

Panduan Pengguna

AWS CodeCommit



Versi API 2015-04-13

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

AWS CodeCommit: Panduan Pengguna

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang menghina atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan kekayaan masing-masing pemiliknya, yang mungkin atau mungkin tidak berafiliasi, terkait dengan, atau disponsori oleh Amazon.

Table of Contents

Apa itu CodeCommit?	1
Memperkenalkan CodeCommit	2
CodeCommitGit, dan memilihAWS layanan yang tepat untuk kebutuhan Anda	3
Bagaimana cara kerja CodeCommit?	6
BagaimanaCodeCommit berbeda dari versioning file di Amazon S3?	8
Bagaimana cara memulai dengan memulai denganCodeCommit?	8
Di mana saya dapat mempelajari lebih lanjut tentang Git?	8
Pengaturan	10
Melihat dan mengelola kredensial Anda	10
Pengaturan menggunakan kredensial Git	11
Pengaturan menggunakan metode lain	12
Kompatibilitas untuk CodeCommit, Git, dan komponen lainnya	14
Untuk pengguna HTTPS menggunakan kredensial Git	15
Langkah 1: Konfigurasi awal untuk CodeCommit	15
Langkah 2: Instal Git	16
Langkah 3: Buat kredensi Git untuk koneksi HTTPS CodeCommit	17
Langkah 4: Connect ke CodeCommit konsol dan kloning repositori	19
Langkah selanjutnya	20
Untuk koneksi HTTPS dengangit-remote-codecommit	21
Langkah 0: Instal prasyarat untukgit-remote-codecommit	22
Langkah 1: Konfigurasi awal untukCodeCommit	23
Langkah 2: Instalgit-remote-codecommit	26
Langkah 3: Hubungkan keCodeCommitkonsol dan kloning repositori	27
Langkah selanjutnya	28
Untuk koneksi dari alat pengembangan	29
Integrasi AWS Cloud9 dengan AWS CodeCommit	32
Mengintegrasikan Visual Studio dengan AWS CodeCommit	36
Mengintegrasikan Eclipse dengan AWS CodeCommit	38
Untuk pengguna SSH yang tidak menggunakan AWS CLI	45
Langkah 1: Kaitkan kunci publik Anda dengan pengguna IAM	45
Langkah 2: Menambahkan CodeCommit ke konfigurasi SSH Anda	46
Langkah selanjutnya	47
Untuk koneksi SSH di Linux, macOS, atau Unix	47
Langkah 1: Konfigurasi awal untukCodeCommit	48

Langkah 2: Instal Git	49
Langkah 3: Konfigurasi kredensi di Linux, macOS, atau Unix	49
Langkah 4: Connect keCodeCommit konsol dan kloning repositori	53
Langkah selanjutnya	55
Untuk koneksi SSH pada Windows	55
Langkah 1: Konfigurasi awal untuk CodeCommit	56
Langkah 2: Instal Git	57
Langkah 3: Siapkan kunci publik dan privat untuk Git dan CodeCommit	57
Langkah 4: Connect ke konsol CodeCommit dan kloning repositori	62
Langkah selanjutnya	63
Untuk koneksi HTTPS di Linux, macOS, atau Unix dengan pembantu kredensial AWS CLI	63
Langkah 1: Konfigurasi awal untukCodeCommit	64
Langkah 2: Instal Git	67
Langkah 3: Mengatur pembantu kredensial	68
Langkah 4: Hubungkan keCodeCommitkonsol dan kloning repositori	70
Langkah selanjutnya	71
Untuk sambungan HTTPS pada Windows dengan pembantu kredensial AWS CLI	71
Langkah 1: Konfigurasi awal untukCodeCommit	72
Langkah 2: Instal Git	76
Langkah 3: Mengatur pembantu kredensial	77
Langkah 4: Hubungkan keCodeCommitkonsol dan kloning repositori	79
Langkah selanjutnya	80
Mulai	81
Memulai dengan CodeCommit	81
Prasyarat	82
Langkah 1: Buat CodeCommit repositori	83
Langkah 2: Tambahkan file ke repositori	86
Langkah 3: Menelusuri isi repositori Anda	88
Langkah 4: Buat dan kolaborasi pada permintaan pull	93
Langkah 5: Bersihkan	100
Langkah 6: Langkah selanjutnya	101
Memulai dengan Git dan CodeCommit	101
Langkah 1: Buat CodeCommit repositori	102
Langkah 2: Buat repo lokal	104
Langkah 3: Buat commit pertama	106
Langkah 4: Kirim commit pertama Anda	107

Langkah 5: Bagikan CodeCommit repositori dan dorong dan tarik komit lain	107
Langkah 6: Buat dan bagikan cabang	110
Langkah 7: Buat dan bagikan tanda	112
Langkah 8: Mengatur izin akses	113
Langkah 9: Bersihkan	116
Integrasi produk dan layanan	119
Integrasi dengan AWS layanan lain	119
Contoh integrasi dari komunitas	128
Unggahan blog	129
Sampel Kode	133
Bekerja dengan repositori	134
Buatlah sebuah repositori	135
Buatlah sebuah (konsol) repositori	136
Buatlah sebuah repositori (AWS CLI)	137
Connect ke sebuah repositori	140
Prasyarat untuk menghubungkan ke repositori CodeCommit	140
Connect ke CodeCommit repositori dengan mengkloning repositori	141
Hubungkan repo lokal ke repositori CodeCommit	143
Bagikan sebuah repositori	144
Pilih protokol koneksi yang akan dibagikan dengan para pengguna Anda	145
Buatlah kebijakan IAM untuk repositori Anda	146
Buatlah sebuah grup IAM untuk para pengguna repositori	148
Bagikan informasi koneksi dengan para pengguna Anda	149
Mengkonfigurasi notifikasi untuk peristiwa repositori	150
Menggunakan aturan notifikasi repositori	152
Membuat aturan notifikasi	152
Mengubah atau menonaktifkan pemberitahuan	156
Hapus notifikasi	157
Menandai sebuah repositori	158
Tambahkan tag ke sebuah repositori	159
Tampilkan tag untuk repositori	161
Edit tag untuk repositori	162
Hapus sebuah tag dari repositori	164
Kelola pemicu untuk repositori	165
Buat sumber daya dan tambahkan izin untuk CodeCommit	166
Buat pemicu untuk topik Amazon SNS	167

Buatlah sebuah pemacu untuk sebuah fungsi Lambda	174
Buatlah sebuah pemacu untuk sebuah fungsi Lambda yang sudah ada	180
Edit pemacu untuk sebuah repositori	188
Uji pemacu untuk sebuah repositori	190
Hapus pemacu dari repositori	192
Mengaitkan atau memisahkan repositori dengan Amazon Reviewer CodeGuru	194
Kaitkan repositori dengan Reviewer CodeGuru	197
Memutuskan repositori dari Reviewer CodeGuru	197
Tampilkan detail repositori	198
Tampilkan detail repositori (konsol)	198
Lihat detail CodeCommit repositori (Git)	199
Lihat detail CodeCommit repositori (AWS CLI)	200
Ubah pengaturan repositori	204
Ubah pengaturan repositori (konsol)	204
Ubah pengaturan AWS CodeCommit repositori (AWS CLI)	206
Sinkronkan perubahan antar repositori	208
Dorong commit ke dua repositori	209
Konfigurasi akses lintas-akun ke sebuah repositori dengan menggunakan peran	214
Akses repositori lintas-akun: Tindakan untuk administrator di AccountA	216
Akses repositori lintas akun: Tindakan untuk administrator di AccountB	219
Akses repositori lintas akun: Tindakan untuk pengguna repositori di AccountB	221
Hapus sebuah repositori	227
Hapus CodeCommit repositori (konsol)	227
Hapus sebuah repo lokal	228
Hapus CodeCommit repositori (AWS CLI)	228
Bekerja dengan file	230
Menelusuri file dalam sebuah repositori	231
MenelusuriCodeCommitrepositori	232
Membuat atau menambahkan file	233
Membuat atau mengunggah file (konsol)	233
Tambahkan file (AWS CLI)	234
Menambahkan file (Git)	236
Mengedit isi file	236
Mengedit file (konsol)	237
Mengedit atau menghapus file (AWS CLI)	238
Edit file (Git)	240

Bekerja dengan permintaan pull	241
Membuat sebuah pull request	245
Membuat sebuah pull request (konsol)	245
Membuat sebuah pull request (AWS CLI)	247
Buat aturan persetujuan	249
Buat aturan persetujuan untuk permintaan pull (konsol)	250
Buat aturan persetujuan untuk permintaan pull (AWS CLI)	252
Tampilkan permintaan pull	254
Tampilkan permintaan pull (konsol)	255
Lihat permintaan pull (AWS CLI)	256
Meninjau permintaan pull	260
Meninjau permintaan pull (konsol)	260
Tinjau permintaan pull (AWS CLI)	266
Memperbarui sebuah permintaan pull	271
Memperbarui sebuah permintaan pull (konsol)	271
Memperbarui permintaan pull (AWS CLI)	272
Mengedit atau menghapus aturan persetujuan	275
Mengedit atau menghapus aturan persetujuan untuk permintaan pull (konsol)	275
Mengedit atau menghapus aturan persetujuan untuk permintaan pull (AWS CLI)	277
Menimpa aturan persetujuan pada sebuah permintaan pull	279
Menimpa aturan persetujuan (konsol)	280
Menimpa aturan persetujuan (AWS CLI)	280
Menggabungkan permintaan pull	282
Menggabungkan permintaan pull (konsol)	283
Menggabungkan permintaan pull (AWS CLI)	286
Menyelesaikan konflik dalam permintaan pull	292
Menyelesaikan konflik dalam permintaan pull (konsol)	293
Menyelesaikan konflik dalam permintaan pull (AWS CLI)	296
Menutup sebuah pull request	304
Menutup sebuah pull request (konsol)	305
Menutup sebuah pull request (AWS CLI)	306
Bekerja dengan templat aturan persetujuan	308
Membuat templat aturan persetujuan	310
Membuat templat aturan persetujuan (konsol)	310
Buat templat aturan persetujuan (AWS CLI)	314
Kaitkan templat aturan persetujuan dengan repositori	316

Mengaitkan templat aturan persetujuan (konsol)	316
Mengaitkan templat aturan persetujuan (AWS CLI)	317
Mengelola templat aturan persetujuan	318
Mengelola templat aturan persetujuan (konsol)	318
Mengelola templat aturan persetujuan (AWS CLI)	319
Melepaskan kaitan templat aturan persetujuan	323
Melepaskan kaitan templat aturan persetujuan (konsol)	323
Memisahkan templat aturan persetujuan (AWS CLI)	324
Menghapus templat aturan persetujuan	325
Menghapus templat aturan persetujuan (konsol)	326
Menghapus templat aturan persetujuan (AWS CLI)	326
Bekerja dengan commit	327
Buatlah sebuah commit	328
Buat komit pertama untuk repositori menggunakan AWS CLI	328
Buatlah sebuah commit menggunakan klien Git	330
Buat komit menggunakan AWS CLI	334
Tampilkan detail commit	337
Telusuri commit dalam sebuah repositori	337
Tampilkan detail commit (AWS CLI)	341
Tampilkan detail commit (Git)	347
Bandingkan commit	350
Bandingkan commit dengan induknya	350
Bandingkan dua penentu commit mana pun	353
Komentari sebuah commit	355
Tampilkan komentar pada commit dalam repositori	356
Tambahkan dan balas komentar pada commit dalam repositori	356
Tampilkan, tambah, perbarui, dan balas komentar (AWS CLI)	361
Buat tag Git	370
Gunakan Git untuk membuat tag	370
Tampilkan detail tag	371
Tampilkan detail tag (konsol)	372
Tampilkan detail tag Git (Git)	373
Hapus tag	375
Gunakan Git untuk menghapus tag Git	375
Bekerja dengan cabang	377
Buatlah sebuah cabang	379

Buatlah sebuah cabang (konsol)	379
Buatlah sebuah cabang (Git)	380
Buatlah sebuah cabang (AWS CLI)	381
Batasi dorongan dan penggabungan ke cabang	383
Mengkonfigurasi kebijakan IAM untuk membatasi dorongan dan penggabungan ke cabang	384
Terapkan kebijakan IAM ke grup atau peran IAM	386
Menguji kebijakan	386
Tampilkan detail cabang	387
Tampilkan detail cabang (konsol)	387
Tampilkan detail cabang (Git)	388
Tampilkan detail cabang (AWS CLI)	389
Bandingkan dan gabungkan cabang	391
Bandingkan sebuah cabang terhadap cabang default	391
Bandingkan dua cabang tertentu	391
Gabungkan dua cabang (AWS CLI)	392
Ubah pengaturan cabang	395
Ubah default branch (konsol)	396
Ubah cabang default (AWS CLI)	396
Hapuslah sebuah cabang	397
Hapuslah sebuah cabang (konsol)	398
Hapuslah sebuah cabang (AWS CLI)	398
Hapuslah sebuah cabang (Git)	399
Bekerja dengan preferensi pengguna	401
Migrasi ke CodeCommit	402
Memigrasikan repositori Git ke AWS CodeCommit	402
Langkah 0: Pengaturan diperlukan untuk akses ke CodeCommit	403
Langkah 1: Buat CodeCommit repositori	409
Langkah 2: Kloning repositori dan dorong ke repositori CodeCommit	411
Langkah 3: Lihat file di CodeCommit	413
Langkah 4: Bagikan CodeCommit repositori	413
Migrasi konten ke CodeCommit	416
Langkah 0: Pengaturan diperlukan untuk akses ke CodeCommit	417
Langkah 1: Buat CodeCommit repositori	423
Langkah 2: Migrasikan konten lokal ke repositori CodeCommit	424
Langkah 3: Lihat file di CodeCommit	425

Langkah 4: Bagikan CodeCommit repositori	426
Memigrasi repositori secara bertahap	428
Langkah 0: Menentukan apakah akan bermigrasi secara bertahap	429
Langkah 1: Instal prasyarat dan tambahkan repositori sebagai CodeCommit remote	430
Langkah 2: Buat skrip untuk digunakan untuk migrasi secara bertahap	432
Langkah 3: Jalankan skrip dan migrasi secara bertahap ke CodeCommit	433
Lampiran: Skrip contoh incremental-repo-migration.py	434
Keamanan	443
Perlindungan data	443
AWS KMS dan enkripsi	444
Menggunakan kredenensi berputar	447
Pengelolaan Identitas dan Akses	452
Audiens	453
Mengautentikasi dengan identitas	453
Mengelola kebijakan menggunakan akses	457
Kontrol autentikasi dan akses	459
Cara kerja AWS CodeCommit dengan IAM	531
CodeCommit Kebijakan berbasis sumber daya	532
Otorisasi berdasarkan tag CodeCommit	532
CodeCommit Peran IAM	535
Contoh kebijakan berbasis identitas	536
Pemecahan Masalah	540
Ketahanan	542
Keamanan infrastruktur	542
Pemantauan CodeCommit	544
Pemantauan CodeCommit acara	545
Peristiwa referenceCreated	546
Peristiwa referenceUpdated	547
Peristiwa referenceDeleted	547
unreferencedMergeCommitPeristiwa	548
commentOnCommitPeristiwa	549
commentOnCommitPeristiwa	550
commentOnPullRequestCreated peristiwa	551
commentOnPullRequestUpdated peristiwa	552
pullRequestCreated peristiwa	553
pullRequestSourceBranchUpdated peristiwa	554

pullRequestStatusBerubah acara	555
pullRequestMergeStatusUpdated peristiwa	556
approvalRuleTemplatePeristiwa	557
approvalRuleTemplatePeristiwa	557
approvalRuleTemplatePeristiwa	558
approvalRuleTemplateAssociatedWithRepository peristiwa	559
approvalRuleTemplateDisassociatedWithRepository peristiwa	560
approvalRuleTemplateBatchAssociatedWithRepositories peristiwa	561
approvalRuleTemplateBatchDisassociatedFromRepositories peristiwa	562
pullRequestApprovalRuleCreated peristiwa	562
pullRequestApprovalRuleDeleted peristiwa	564
pullRequestApprovalRuleOverridden peristiwa	565
pullRequestApprovalStateChanged peristiwa	567
pullRequestApprovalRuleUpdated peristiwa	569
Peristiwa reactionCreated	570
Peristiwa reactionUpdated	571
Mencatat panggilan API AWS CodeCommit dengan AWS CloudTrail	572
CodeCommit informasi dalam CloudTrail	572
Memahami entri berkas CodeCommit log	574
Sumber daya AWS CloudFormation	581
Templat CodeCommit dan AWS CloudFormation	581
Contoh templat	582
AWS CloudFormation, CodeCommit, dan AWS Cloud Development Kit (AWS CDK)	583
Pelajari selengkapnya tentang AWS CloudFormation	584
Memecahkan masalah	585
Mengatasi masalah kredensial Git (HTTPS)	585
Kredensial Git untuk AWS CodeCommit: Saya terus melihat permintaan kredensial ketika saya terhubung ke repositori CodeCommit saya di terminal atau baris perintah	586
Kredensial Git untuk AWS CodeCommit: Saya menyiapkan kredensial Git, tetapi sistem saya tidak menggunakannya	586
Mengatasi masalah git-remote-codecommit	587
Saya melihat kesalahan: git: 'remote-codecommit' bukan perintah git	587
Saya melihat kesalahan: fatal: Tidak dapat menemukan penolong jarak jauh untuk 'codecommit'	588
Kesalahan kloning: Saya tidak dapat mengkloning repositori CodeCommit dari IDE	588

Push atau pull error: Saya tidak dapat melakukan push atau pull commit dari IDE ke repositori CodeCommit	588
Pemecahan masalah koneksi SSH	589
Kesalahan akses: Kunci publik berhasil diunggah ke IAM namun koneksi gagal pada sistem Linux, macOS, atau Unix	589
Kesalahan akses: Kunci publik berhasil diunggah ke IAM dan SSH berhasil diuji tetapi koneksi gagal pada sistem Windows	591
Tantangan otentikasi: Keaslian host tidak dapat dibuat saat menghubungkan keCodeCommitrepositori	591
Galat IAM: 'Format tidak valid' ketika mencoba untuk menambahkan kunci publik ke IAM	598
Saya perlu mengaksesCodeCommitrepositori di beberapa akun Amazon Web Services dengan kredensi SSH	599
Git pada Windows: Bash emulator atau baris perintah membeku ketika mencoba untuk terhubung menggunakan SSH	600
Format kunci publik membutuhkan spesifikasi dalam beberapa distribusi Linux	601
Kesalahan akses: Kunci publik SSH ditolak saat menghubungkan keCodeCommitrepositori	601
Pemecahan masalah pembantu kredensial (HTTPS)	601
Saya menerima galat saat menjalankan perintah <code>git config</code> untuk mengkonfigurasi pembantu kredensial	602
Saya mendapatkan perintah tidak ditemukan kesalahan di Windows saat menggunakan pembantu kredensial	603
Saya diminta untuk memasukkan nama pengguna ketika saya terhubung ke sebuah repositori CodeCommit	604
Git untuk macOS: Saya mengkonfigurasi pembantu kredensial dengan sukses, tapi sekarang saya menolak akses ke repositori saya (403)	604
Git untuk Windows: Saya menginstal Git untuk Windows, tapi akses saya ke repositori saya ditolak (403)	607
Mengatasi masalah klien Git	609
Kesalahan Git: Kesalahan: RPC gagal; hasil = 56, kode HTTP = 200 fatal: Ujung terpencil menutup telepon secara tak terduga	610
Kesalahan Git: Terlalu banyak perintah pembaruan referensi	610
Kesalahan Git: Dorong melalui HTTPS rusak di beberapa versi Git	610
Kesalahan Git: 'gnutls_handshake () gagal'	611
Kesalahan Git: Git tidak dapat menemukan repositori CodeCommit atau tidak memiliki izin untuk mengakses repositori	611

Git pada Windows: Tidak tersedia metode autentikasi yang didukung (publickey)	611
Mengatasi masalah kesalahan akses	612
Kesalahan akses: Saya diminta untuk memasukkan nama pengguna dan sandi ketika saya terhubung ke sebuah repositori CodeCommit dari Windows	612
Kesalahan akses: Kunci publik ditolak saat menghubungkan ke sebuah repositori CodeCommit	613
Kesalahan akses: Pesan “Tingkat Terlampaui” atau “429” saat menghubungkan ke sebuah repositori CodeCommit	613
Mengatasi masalah kesalahan konfigurasi	615
Kesalahan konfigurasi: Tidak dapat mengonfigurasiAWS CLIkredensi pada macOS	615
Mengatasi masalah kesalahan konsol	615
Kesalahan akses: Akses kunci enkripsi ditolak untuk CodeCommit repositori dari konsol atau AWS CLI	613
Kesalahan enkripsi: Repositori tidak dapat didekripsi	616
Kesalahan konsol: Tidak dapat menelusuri kode di CodeCommit repositori dari konsol	616
Kesalahan tampilan: Tidak dapat melihat file atau perbandingan antar file	617
Pemecahan masalah pemicu	617
Kesalahan pemicu: Pemicu repositori tidak berjalan ketika diharapkan	617
Mengaktifkan debugging	618
CodeCommit referensi	620
Wilayah dan titik akhir koneksi Git	620
Didukung Wilayah AWS untuk CodeCommit	620
Titik akhir koneksi Git	622
Sidik jari server untuk CodeCommit	629
Menggunakan AWS CodeCommit dengan antarmuka VPC endpoint	636
Ketersediaan	637
Buat titik akhir VPC untuk CodeCommit	638
Membuat kebijakan titik akhir VPC untuk CodeCommit	639
Kuota	640
Referensi baris perintah	648
Perintah Git Basic	654
Variabel konfigurasi	654
Repositori jarak jauh	655
Berkomitmen	657
Cabang	658
Tanda	659

Riwayat dokumen	661
Pembaruan lebih awal	672
AWSGlosarium	679
.....	dclxxx

Apakah AWS CodeCommit itu?

AWS CodeCommit adalah layanan kontrol versi yang di-hosting oleh Amazon Web Services yang dapat Anda gunakan untuk menyimpan dan mengelola aset (seperti dokumen, kode sumber, dan file biner) secara privat di cloud. Untuk informasi selengkapnya tentang hargaCodeCommit, lihat [Harga](#).

Note

CodeCommitadalah dalam lingkup dengan banyak program kepatuhan. Untuk detail tentang AWS dan upaya kepatuhan, lihat [layanan AWS dalam lingkup berdasarkan program kepatuhan](#).

Ini adalah Layanan yang Memenuhi Syarat HIPAA. Untuk informasi selengkapnya tentang AWS, Undang-Undang Portabilitas dan Akuntabilitas Asuransi Kesehatan A.S. tahun 1996 (Health Insurance Portability and Accountability Act/HIPAA) dan menggunakan layanan AWS untuk memproses, menyimpan, dan mengirimkan informasi kesehatan yang dilindungi (PHI), lihat [Gambaran Umum HIPAA](#).

Untuk informasi tentang layanan ini dan ISO 27001, standar manajemen keamanan yang menentukan praktik terbaik manajemen keamanan, lihat [Gambaran Umum ISO 27001](#).

Untuk informasi tentang layanan ini dan Standar Keamanan Data Industri Kartu Pembayaran (PCI DSS), lihat [Gambaran Umum PCI DSS](#).

Untuk informasi tentang layanan ini dan Federal Information Processing Standard (FIPS) standar pemerintah AS Publikasi 140-2 AS yang menentukan persyaratan keamanan untuk modul kriptografi yang melindungi informasi sensitif, lihat [Gambaran Umum Federal Information Processing Standard \(FIPS\) 140-2](#) dan [Titik akhir koneksi Git](#).

Topik

- [Memperkenalkan CodeCommit](#)
- [CodeCommitGit, dan memilihAWS layanan yang tepat untuk kebutuhan Anda](#)
- [Bagaimana cara kerja CodeCommit?](#)
- [BagaimanaCodeCommit berbeda dari versioning file di Amazon S3?](#)
- [Bagaimana cara memulai dengan memulai denganCodeCommit?](#)
- [Di mana saya dapat mempelajari lebih lanjut tentang Git?](#)

Memperkenalkan CodeCommit

CodeCommit adalah layanan kontrol sumber terkelola yang aman, dan sangat terukur yang meng-hosting repositori Git privat. CodeCommit menghilangkan kebutuhan Anda untuk mengelola sistem kontrol sumber Anda sendiri atau khawatir tentang penskalaan infrastrukturnya. Anda dapat menggunakan CodeCommit untuk menyimpan apa pun dari kode ke binari. Ini mendukung fungsi standar Git, sehingga bekerja mulus dengan alat berbasis Git yang ada.

Dengan CodeCommit, Anda dapat:

- Manfaat dari layanan terkelola penuh yang di-hosting oleh AWS. CodeCommit menyediakan ketersediaan dan daya tahan layanan yang tinggi dan menghilangkan biaya operasional administrasi untuk mengelola perangkat keras dan perangkat lunak Anda sendiri. Tidak ada perangkat keras untuk penyediaan dan penskalaan dan tidak ada perangkat lunak server yang harus diinstal, dikonfigurasi, dan diperbarui.
- Simpan kode Anda dengan aman. CodeCommit repositori dienkripsi saat tidak digunakan dan saat di transit.
- Bekerja secara kolaboratif pada kode. CodeCommit repositori mendukung permintaan pull, di mana pengguna dapat meninjau dan mengomentari perubahan kode masing-masing sebelum menggabungkannya ke cabang; notifikasi yang secara otomatis mengirim email kepada pengguna tentang permintaan pull dan komentar; dan banyak lagi.
- Dengan mudah menskalakan proyek kontrol versi Anda. CodeCommit repositori dapat ditingkatkan skalanya untuk memenuhi kebutuhan pengembangan Anda. Layanan ini dapat menangani repositori dengan sejumlah besar file atau cabang, ukuran file besar, dan riwayat revisi yang panjang.
- Simpan apa saja, kapan saja. CodeCommit tidak memiliki batasan ukuran repositori Anda atau jenis file yang dapat Anda simpan.
- Mengintegrasikan dengan AWS lainnya dan layanan pihak ketiga. CodeCommit menyimpan repositori Anda dekat dengan sumber daya produksi Anda yang lain di AWS Cloud, yang membantu meningkatkan kecepatan dan frekuensi siklus hidup pengembangan Anda. Hal ini terintegrasi dengan IAM dan dapat digunakan dengan layanan AWS lainnya dan secara paralel dengan repositori lainnya. Untuk informasi selengkapnya, lihat [Integrasi produk dan layanan dengan AWS CodeCommit](#).
- Migrasikan file dengan mudah dari repositori jarak jauh lainnya. Anda dapat bermigrasi ke CodeCommit dari repositori berbasis Git.

- Gunakan alat Git yang sudah Anda ketahui. CodeCommit mendukung perintah Git serta AWS CLI perintah dan API.

CodeCommit Git, dan memilih AWS layanan yang tepat untuk kebutuhan Anda

Sebagai layanan berbasis Git, CodeCommit sangat cocok untuk sebagian besar kebutuhan kontrol versi. Tidak ada batasan sewenang-wenang pada ukuran file, jenis file, dan ukuran repositori. Namun, ada keterbatasan yang melekat pada Git yang dapat berdampak negatif terhadap kinerja jenis operasi tertentu, terutama seiring waktu. Anda dapat menghindari potensi degradasi kinerja CodeCommit repositori dengan menghindari menggunakannya untuk kasus penggunaan di mana AWS layanan lainnya lebih cocok untuk tugas tersebut. Anda juga dapat mengoptimalkan kinerja Git untuk repositori kompleks. Berikut adalah beberapa kasus penggunaan di mana Git, dan karena itu CodeCommit, mungkin bukan solusi terbaik untuk Anda, atau di mana Anda mungkin perlu mengambil langkah-langkah tambahan untuk mengoptimalkan Git.

Kasus penggunaan	Deskripsi	Layanan lain yang perlu dipertimbangkan
File besar yang sering berubah	Git menggunakan delta encoding untuk menyimpan perbedaan antara versi file. Misalnya, jika Anda mengubah beberapa kata dalam dokumen, Git hanya akan menyimpan kata-kata yang berubah. Jika Anda memiliki file atau objek lebih dari 5 MB dengan banyak perubahan, Git mungkin perlu merekonstruksi rantai besar delta perbedaan. Ini dapat membutuhkan peningkatan jumlah sumber daya komputasi pada komputer lokal Anda dan CodeCommit	Untuk versi file besar, pertimbangkan Amazon Simple Storage Service (Amazon S3). Untuk informasi selengkapnya, lihat Menggunakan Versi di Panduan Pengguna Amazon Simple Storage Service.

Kasus penggunaan	Deskripsi	Layanan lain yang perlu dipertimbangkan
	karena file-file ini tumbuh seiring waktu.	
Basis Data	<p>Repositori Git tumbuh lebih besar seiring waktu. Karena versioning melacak semua perubahan, perubahan apapun akan meningkatkan ukuran repositori Anda. Dengan kata lain, saat Anda membuat commit data, bahkan jika Anda menghapus data dalam suatu commit, data ditambahkan ke repositori. Karena ada lebih banyak data untuk memproses dan mengirimkan seiring waktu, Git akan melambat. Hal ini secara khusus merugikan kasus penggunaan basis data. Git tidak dirancang sebagai basis data.</p>	<p>Untuk membuat dan menggunakan basis data dengan kinerja yang konsisten terlepas dari ukuran, pertimbangkan Amazon DynamoDB. Untuk informasi selengkapnya, lihat Panduan Memulai Amazon DynamoDB.</p>

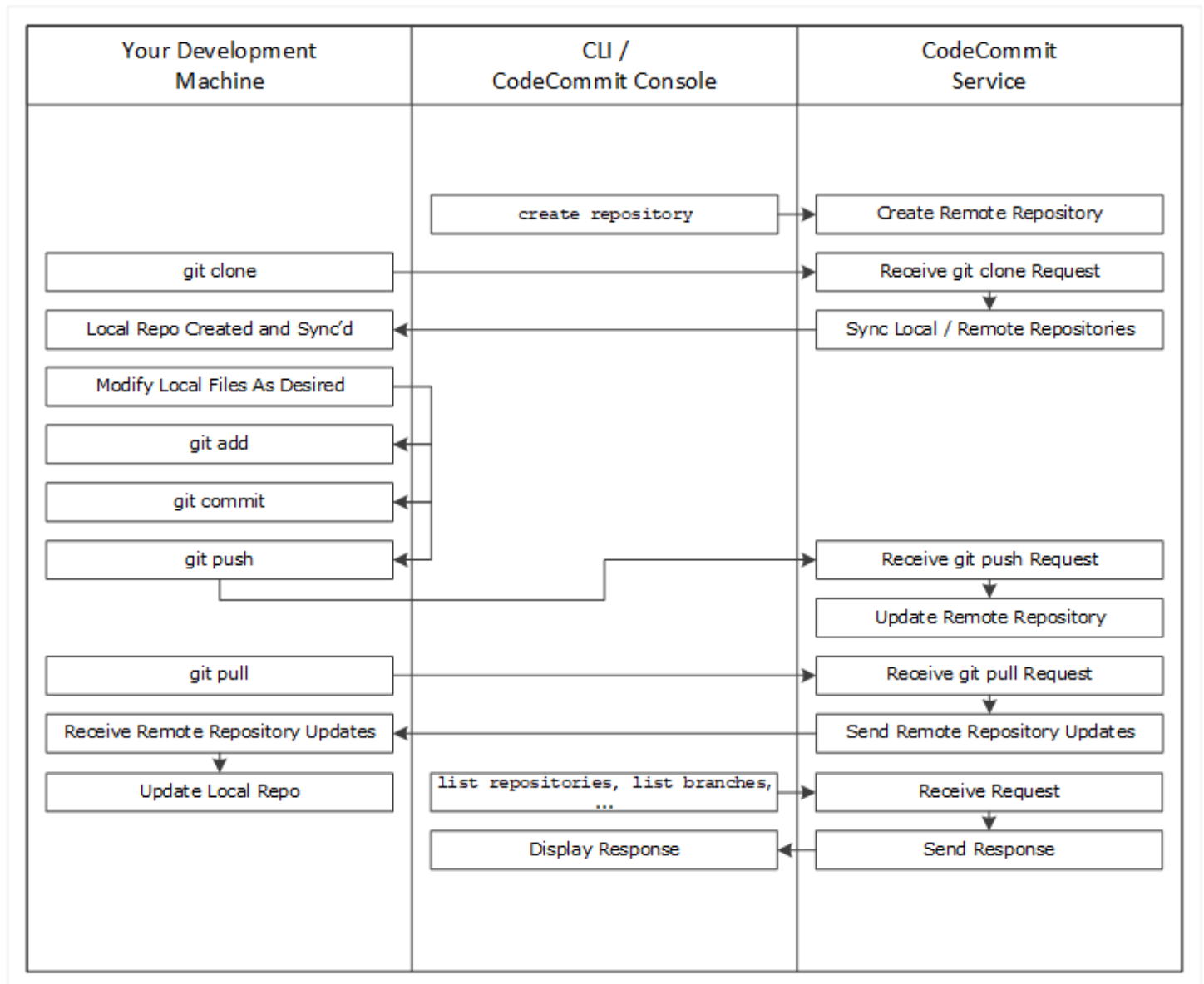
Kasus penggunaan	Deskripsi	Layanan lain yang perlu dipertimbangkan
Jejak audit	<p>Biasanya, jejak audit disimpan dalam jangka waktu yang lama dan terus dihasilkan oleh proses sistem pada irama yang sangat sering. Git dirancang untuk aman menyimpan kode sumber yang dihasilkan oleh grup developer pada siklus pengembangan. Repositori yang berubah dengan cepat yang terus menyimpan perubahan sistem yang dihasilkan pemrograman akan melihat kinerja menurun seiring waktu.</p>	<p>Untuk menyimpan jejak audit, pertimbangkan Amazon Simple Storage Service (Amazon S3).</p> <p>Untuk mengaudit AWS aktivitas, tergantung pada kasus penggunaan Anda, pertimbangkan untuk menggunakan AWS CloudTrail, AWS Config, atau Amazon CloudWatch.</p>
Backup	<p>Git dirancang untuk kode sumber versi yang ditulis oleh developer. Anda dapat mengirim commit ke dua repositori remote, termasuk CodeCommit repositori, sebagai strategi cadangan. Namun, Git tidak dirancang untuk menangani backup dari sistem file komputer Anda, dump basis data, atau konten cadangan serupa. Melakukan hal itu dapat memperlambat sistem Anda dan meningkatkan jumlah waktu yang diperlukan untuk mengkloning dan mengirim repositori.</p>	<p>Untuk informasi tentang pencadangan ke Cloud AWS, lihat Backup & Restore.</p>

Kasus penggunaan	Deskripsi	Layanan lain yang perlu dipertimbangkan
Sejumlah besar cabang atau referensi	Ketika klien Git mengirim atau menarik data repositori, server remote harus mengirim semua cabang dan referensi seperti tanda, bahkan jika Anda hanya tertarik pada satu cabang. Jika Anda memiliki ribuan cabang dan referensi, untuk memproses dan mengirim (negosiasi paket) akan memakan waktu sehingga respon repositori yang dihasilkan kemungkinan akan lambat. Semakin banyak cabang dan tanda yang Anda miliki, semakin lama proses ini dapat dilakukan. Sebaiknya gunakan CodeCommit, namun hapus cabang dan tanda yang tidak diperlukan lagi.	Untuk menganalisis jumlah referensi dalam CodeCommit repositori untuk menentukan mana yang mungkin tidak diperlukan, Anda dapat menggunakan salah satu dari perintah berikut: <ul style="list-style-type: none">Linux, macOS, atau Unix, atau emulator Bash pada Windows:<pre data-bbox="1101 808 1507 886">git ls-remote wc -l</pre>PowerShell:<pre data-bbox="1101 976 1507 1096">git ls-remote Measure-Object -line</pre>

Bagaimana cara kerja CodeCommit?

CodeCommit dikenal oleh pengguna repositori berbasis Git, tetapi bahkan untuk tidak mengenalnya seharusnya menganggap transisi ke CodeCommit relatif sederhana. CodeCommit menyediakan konsol untuk pembuatan repositori yang mudah dan daftar repositori dan cabang yang ada. Dalam beberapa langkah sederhana, pengguna dapat menemukan informasi tentang repositori dan mengkloning ke komputer mereka, membuat repo lokal di mana mereka dapat membuat perubahan dan kemudian mengirimnya ke CodeCommit repositori. Pengguna dapat bekerja dari baris perintah pada mesin lokal mereka atau menggunakan editor berbasis GUI.

Gambar berikut menunjukkan bagaimana Anda menggunakan mesin pengembangan Anda, AWS CLI atau CodeCommit konsol, dan CodeCommit layanan untuk membuat dan mengelola repositori:



1. Gunakan AWS CLI atau CodeCommit konsol untuk membuat CodeCommit repositori.
2. Dari mesin pengembangan Anda, gunakan Git untuk menjalankan Git untuk menjalankan git clone, menentukan nama CodeCommit repositori. Hal ini menciptakan repo lokal yang menghubungkan ke CodeCommit repositori.
3. Menggunakan repo lokal pada mesin pengembangan Anda untuk mengubah (menambah, mengedit, dan menghapus) file, dan kemudian jalankan git add untuk mengatur file yang dimodifikasi secara lokal. Jalankan git commit untuk melakukan file secara lokal, dan kemudian jalankan git push untuk mengirim file ke CodeCommit repositori.

4. Unduh perubahan dari pengguna lain. Jalankan `git pull` untuk menyinkronkan file di CodeCommit repositori dengan repo lokal Anda. Hal ini memastikan Anda bekerja dengan versi terbaru dari file.

Anda dapat menggunakan AWS CLI atau CodeCommit konsol untuk melacak dan mengelola repositori Anda.

Bagaimana CodeCommit berbeda dari versioning file di Amazon S3?

CodeCommit dioptimalkan untuk pengembangan perangkat lunak tim. Ini mengelola batch perubahan di beberapa file, yang dapat terjadi secara paralel dengan perubahan yang dibuat oleh developer lain. Versioning Amazon S3 mendukung pemulihan versi sebelumnya dari file, tetapi tidak terfokus pada fitur pelacakan file kolaboratif yang dibutuhkan tim pengembangan perangkat lunak.

Bagaimana cara memulai dengan CodeCommit?

Untuk memulai dengan CodeCommit:

1. Ikuti langkah-langkah di [Pengaturan](#) untuk mempersiapkan mesin pengembangan Anda.
2. Ikuti langkah-langkah di salah satu atau beberapa tutorial di [Mulai](#).
3. [Buat](#) proyek kontrol versi CodeCommit atau [migrasi](#) proyek kontrol versi ke CodeCommit.

Di mana saya dapat mempelajari lebih lanjut tentang Git?

Jika Anda belum mengetahuinya, Anda harus [mempelajari cara menggunakan Git](#). Berikut adalah beberapa sumber daya yang bermanfaat:

- [Pro Git](#), versi online buku Pro Git. Ditulis oleh Scott Chacon. Diterbitkan oleh Apress.
- [Perendaman Git](#), turtry-it-yourself berpemandu yang memandu Anda melalui dasar-dasar penggunaan Git. Diterbitkan oleh Neo Innovation, Inc.
- [Referensi Git](#), referensi cepat online yang juga dapat digunakan sebagai tutorial Git yang lebih mendalam. Diterbitkan oleh GitHub tim.

- [Lembar Cheat Git](#) dengan sintaks perintah Git dasar. Diterbitkan oleh GitHub tim.
- [Panduan Git](#). Ditulis oleh Richard E. Silverman. Diterbitkan oleh O'Reilly Media, Inc.

Pengaturan untuk AWS CodeCommit

Anda dapat masuk ke AWS Management Console dan [mengunggah, menambah, atau mengedit file](#) ke repositori langsung dari konsol AWS CodeCommit. Ini adalah cara cepat untuk membuat perubahan. Namun, jika Anda ingin bekerja dengan beberapa file, file di seluruh cabang, dan sebagainya, pertimbangkan untuk menyiapkan komputer lokal Anda untuk bekerja dengan repositori. Cara termudah untuk mengatur CodeCommit adalah mengkonfigurasi kredensial Git HTTPS untuk AWS CodeCommit. Metode autentikasi HTTPS ini:

- Menggunakan nama pengguna dan kata sandi statis.
- Bekerja dengan semua sistem operasi yang didukung oleh CodeCommit.
- Juga kompatibel dengan lingkungan pengembangan terpadu (IDE) dan alat pengembangan lainnya yang mendukung kredensial Git.

Anda dapat menggunakan metode lain jika Anda tidak ingin atau tidak dapat menggunakan kredensial Git untuk alasan operasional. Misalnya, jika Anda mengakses repositori CodeCommit menggunakan akses gabungan, kredensial sementara, atau penyedia identitas web, Anda tidak dapat menggunakan kredensial Git. Kami menyarankan agar Anda mengatur komputer lokal Anda menggunakan perintah `git-remote-codecommit`. Tinjau pilihan ini dengan hati-hati, untuk memutuskan metode alternatif mana yang terbaik untuk Anda.

- [Pengaturan menggunakan kredensial Git](#)
- [Pengaturan menggunakan metode lain](#)
- [Kompatibilitas untuk CodeCommit, Git, dan komponen lainnya](#)

Untuk informasi tentang menggunakan CodeCommit dan Amazon Virtual Private Cloud, lihat [Menggunakan AWS CodeCommit dengan antarmuka VPC endpoint](#).

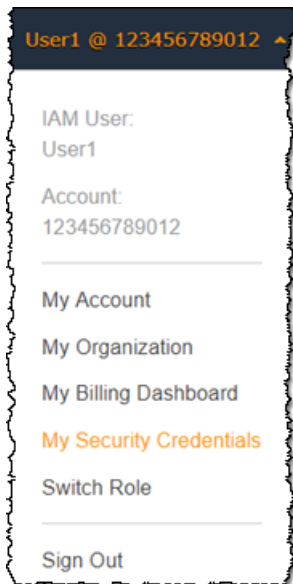
Melihat dan mengelola kredensial Anda

Anda dapat melihat dan mengelola kredensial CodeCommit Anda dari konsol AWS melalui Kredensial Keamanan Saya.

Note

Opsi ini tidak tersedia untuk pengguna yang menggunakan akses gabungan, kredensi sementara, atau penyedia identitas web.

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Di bilah navigasi di kanan atas, pilih nama pengguna Anda, dan kemudian pilih Kredensial Keamanan Saya.



3. Pilih tab kredensial AWS CodeCommit.

Pengaturan menggunakan kredensial Git

Dengan koneksi HTTPS dan kredensial Git, Anda menghasilkan nama pengguna statis dan kata sandi di IAM. Anda kemudian menggunakan kredensial ini dengan Git dan alat pihak ketiga yang mendukung autentikasi nama pengguna dan kata sandi Git. Metode ini didukung oleh sebagian besar IDE dan alat pengembangan. Ini adalah metode koneksi yang paling sederhana dan termudah untuk digunakan dengan CodeCommit.

- [Untuk pengguna HTTPS menggunakan kredensial Git](#): Ikuti petunjuk ini untuk mengatur koneksi antara komputer lokal dan repositori CodeCommit menggunakan kredensial Git.

- [Untuk koneksi dari alat pengembangan](#): Ikuti panduan ini untuk mengatur koneksi antara IDE atau alat pengembangan lainnya dan repositori CodeCommit menggunakan kredensial Git. IDE yang mendukung kredensial Git termasuk (namun tidak terbatas pada) Visual Studio, Eclipse, Xcode, dan IntelliJ.

Pengaturan menggunakan metode lain

Anda dapat menggunakan protokol SSH bukan HTTPS untuk terhubung ke repositori CodeCommit Anda. Dengan koneksi SSH, Anda membuat file kunci publik dan privat di mesin lokal Anda seperti Git dan CodeCommit yang digunakan untuk autentikasi SSH. Anda mengaitkan kunci publik dengan pengguna IAM Anda. Anda menyimpan kunci privat di mesin lokal Anda. Karena SSH memerlukan pembuatan manual dan pengelolaan file kunci publik dan privat, Anda mungkin menemukan kredensial Git lebih sederhana dan lebih mudah digunakan dengan CodeCommit.

Tidak seperti kredensial Git, pengaturan koneksi SSH bervariasi, tergantung pada sistem operasi pada komputer lokal Anda.

- [Untuk pengguna SSH yang tidak menggunakan AWS CLI](#): Ikuti petunjuk singkat ini jika Anda sudah memiliki pasangan kunci publik-privat dan akrab dengan koneksi SSH pada komputer lokal Anda.
- [Untuk koneksi SSH di Linux, macOS, atau Unix](#): Ikuti petunjuk berikut untuk panduan langkah demi langkah untuk membuat pasangan kunci privat publik dan menyiapkan koneksi di sistem operasi Linux, macOS, atau Unix.
- [Untuk koneksi SSH pada Windows](#): Ikuti petunjuk ini untuk panduan langkah demi langkah pembuatan pasangan kunci publik-privat dan menyiapkan koneksi pada sistem operasi Windows.

Jika Anda terhubung ke CodeCommit dan AWS menggunakan akses federasi, penyedia identitas, atau kredensial sementara, atau jika Anda tidak ingin mengkonfigurasi pengguna IAM atau kredensial Git untuk pengguna IAM, Anda dapat mengatur koneksi ke repositori CodeCommit dengan salah satu dari dua cara:

- Instal dan gunakan `git-remote-codecommit` (direkomendasikan).
- Instal dan gunakan pembantu kredensial yang disertakan dalam AWS CLI.

Kedua metode mendukung akses repositori CodeCommit tanpa memerlukan pengguna IAM, artinya Anda dapat terhubung ke repositori menggunakan akses gabungan dan kredensial sementara.

Utilitas `git-remote-codecommit` adalah pendekatan yang disarankan. Ini meluas Git dan kompatibel dengan berbagai versi Git dan pembantu kredensial. Namun, tidak semua IDE mendukung format URL klon yang digunakan oleh `git-remote-codecommit`. Anda mungkin harus secara manual mengkloning repositori ke komputer lokal Anda sebelum Anda dapat bekerja dengan repositori di IDE Anda.

- Ikuti instruksi di [Langkah-Langkah Pengaturan Untuk Koneksi HTTPS ke Repositori AWS CodeCommit dengan git-remote-codecommit](#) untuk menginstal dan mengatur `git-remote-codecommit` di Windows, Linux, macOS, atau Unix.

Pembantu kredensial yang termasuk dalam AWS CLI memungkinkan Git untuk menggunakan HTTPS dan versi kredensi pengguna IAM yang ditandatangani secara kriptografis atau peran instans Amazon EC2 Anda setiap kali Git perlu mengotentikasi dengan AWS untuk berinteraksi dengan repositori CodeCommit. Beberapa sistem operasi dan versi Git memiliki pembantu kredensialnya sendiri, yang bertentangan dengan pembantu kredensial termasuk dalam AWS CLI. Ini dapat menyebabkan masalah konektivitas untuk CodeCommit.

- [Untuk koneksi HTTPS di Linux, macOS, atau Unix dengan pembantu kredensial AWS CLI](#): Ikuti petunjuk berikut untuk panduan langkah demi langkah untuk menginstal dan mengatur pembantu kredensial di sistem Linux, macOS, atau Unix.
- [Untuk sambungan HTTPS pada Windows dengan pembantu kredensial AWS CLI](#): Ikuti petunjuk berikut untuk panduan langkah demi langkah untuk menginstal dan mengatur pembantu kredensial di sistem Windows.

Jika Anda terhubung ke repositori CodeCommit yang di-host di akun Amazon Web Services lain, Anda dapat mengkonfigurasi akses dan mengatur koneksi menggunakan peran, kebijakan, dan pembantu kredensial yang disertakan dalam AWS CLI.

- [Konfigurasi akses lintas akun ke AWS CodeCommit repositori menggunakan peran](#): Ikuti petunjuk ini untuk panduan langkah demi langkah dalam mengkonfigurasi akses lintas akun di salah satu akun Amazon Web Services untuk pengguna dalam kelompok IAM di akun Amazon Web Services lain.

Kompatibilitas untuk CodeCommit, Git, dan komponen lainnya

Ketika Anda bekerja dengan CodeCommit, Anda menggunakan Git. Anda juga dapat menggunakan program lain. Tabel berikut memberikan panduan terbaru untuk versi kompatibilitas. Sebagai praktik terbaik, kami menyarankan Anda menggunakan versi terbaru dari Git, dan perangkat lunak lainnya.

Informasi kompatibilitas versi untuk AWS CodeCommit

Komponen	Versi
Git	CodeCommit support versi Git 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git.
Curl	CodeCommit membutuhkan curl 7.33 dan versi lebih baru. Namun, ada masalah yang diketahui dengan HTTPS dan curl update 7.41.0. Untuk informasi selengkapnya, lihat Memecahkan masalah .
Python (hanya git-remote-codecommit)	git-remote-codecommit memerlukan versi 3 dan versi lebih baru.
Pip (hanya git-remote-codecommit)	git-remote-codecommit memerlukan versi 9.0.3 dan versi lebih baru.
AWS CLI (hanya git-remote-codecommit)	Kami merekomendasikan versi terbaru dari AWS CLI versi 2 untuk semua pengguna CodeCommit. git-remote-codecommit membutuhkan AWS CLI versi 2 untuk mendukung SSO dan koneksi AWS yang memerlukan kredensial sementara, seperti pengguna gabungan.

Pengaturan untuk pengguna HTTPS menggunakan kredensial Git

Cara termudah untuk mengatur koneksi ke AWS CodeCommit repositori adalah dengan mengkonfigurasi kredensial Git untuk CodeCommit di konsol IAM, dan kemudian menggunakan kredensial tersebut untuk koneksi HTTPS. Anda juga dapat menggunakan kredensial yang sama ini dengan alat pihak ketiga atau lingkungan pengembangan terintegrasi (IDE) yang mendukung otentikasi HTTPS menggunakan nama pengguna dan kata sandi statis. Sebagai contoh, lihat [Untuk koneksi dari alat pengembangan](#).

Note

Jika sebelumnya Anda telah mengonfigurasi komputer lokal Anda untuk menggunakan bantuan kredensialnya CodeCommit, Anda harus mengedit file.gitconfig Anda untuk menghapus informasi pembantu kredensial dari file sebelum Anda dapat menggunakan kredensial Git. Jika komputer lokal menjalankan macOS, Anda mungkin harus menghapus kredensial yang disimpan di cache dari Keychain Access.

Langkah 1: Konfigurasi awal untuk CodeCommit

Ikuti langkah-langkah berikut untuk menyiapkan akun Amazon Web Services, membuat pengguna IAM, dan mengonfigurasi akses ke CodeCommit akun.

Untuk membuat dan mengkonfigurasi pengguna IAM untuk mengakses CodeCommit

1. Buat akun Amazon Web Services dengan membuka <http://aws.amazon.com> dan memilih Mendaftar.
2. Membuat pengguna IAM, atau menggunakan yang sudah ada, di akun Amazon Web Services Anda. Pastikan Anda memiliki access key ID dan secret access key yang terkait dengan pengguna IAM. Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di akun Amazon Web Services Anda](#).

Note

CodeCommit membutuhkan AWS Key Management Service. Jika Anda menggunakan pengguna IAM yang sudah ada, pastikan tidak ada kebijakan yang dilampirkan pada

pengguna yang secara tegas menolak AWS KMS tindakan yang diperlukan oleh CodeCommit Untuk informasi selengkapnya, lihat [AWS KMS dan enkripsi](#).

3. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
4. Di konsol IAM, di panel navigasi, pilih Pengguna, lalu pilih pengguna IAM yang ingin Anda konfigurasi untuk akses. CodeCommit
5. Pada tab Izin, pilih Tambahkan izin.
6. Dalam Berikan izin, pilih Lampirkan kebijakan yang ada secara langsung.
7. Dari daftar kebijakan, pilih AWSCodeCommitPowerUseratau kebijakan terkelola lainnya untuk CodeCommit akses. Untuk informasi selengkapnya, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Setelah memilih kebijakan yang ingin dilampirkan, pilih Berikutnya: Tinjau untuk meninjau daftar kebijakan untuk dilampirkan ke pengguna IAM. Jika daftar sudah benar, pilih Tambahkan izin.

Untuk informasi selengkapnya tentang kebijakan CodeCommit terkelola dan berbagi akses ke repositori dengan grup dan pengguna lain, lihat [Bagikan sebuah repositori](#) dan [Kontrol autentikasi dan akses untuk AWS CodeCommit](#)

Jika Anda ingin menggunakan AWS CLI perintah dengan CodeCommit, instal file AWS CLI. Kami menyarankan Anda membuat profil untuk menggunakan AWS CLI dengan CodeCommit. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#) dan [Menggunakan profil bernama](#).

Langkah 2: Instal Git

Untuk bekerja dengan file, commit, dan informasi lainnya di CodeCommit repositori, Anda harus menginstal Git pada mesin lokal Anda. CodeCommit mendukung Git versi 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git.

Untuk menginstal Git, kami merekomendasikan situs web seperti [Pengunduhan Git](#).

Note

Git adalah platform yang berkembang dan diperbarui secara teratur. Terkadang, perubahan fitur dapat memengaruhi cara kerjanya CodeCommit. Jika Anda mengalami masalah dengan versi Git tertentu dan CodeCommit, tinjau informasinya di [Memecahkan masalah](#).

Langkah 3: Buat kredensi Git untuk koneksi HTTPS CodeCommit

Setelah Anda menginstal Git, membuat kredensial Git untuk pengguna IAM Anda di IAM.

Untuk mengatur kredensi HTTPS Git untuk CodeCommit

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.

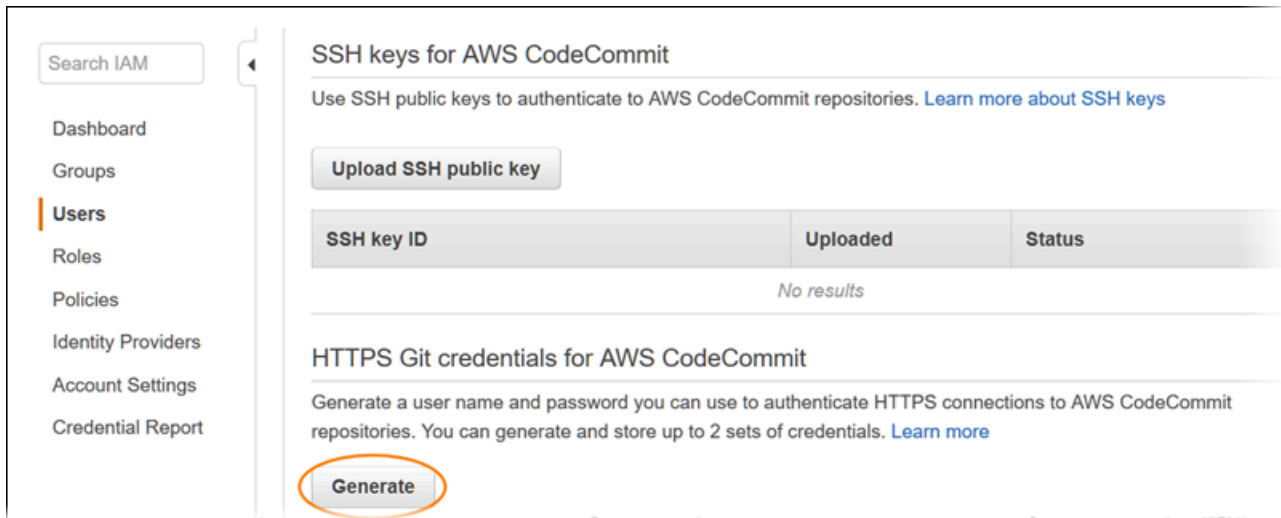
Pastikan untuk masuk sebagai pengguna IAM yang akan membuat dan menggunakan kredensi Git untuk koneksi ke CodeCommit

2. Di konsol IAM, pada panel navigasi, pilih Pengguna, dan dari daftar pengguna, pilih pengguna IAM Anda.

Note

Anda dapat langsung melihat dan mengelola CodeCommit kredensialnya di My Security Credentials. Untuk informasi selengkapnya, lihat [Melihat dan mengelola kredensial Anda](#).

3. Pada halaman detail pengguna, pilih tab Security Credentials, dan di HTTPS Git credentials for AWS CodeCommit, pilih Generate.

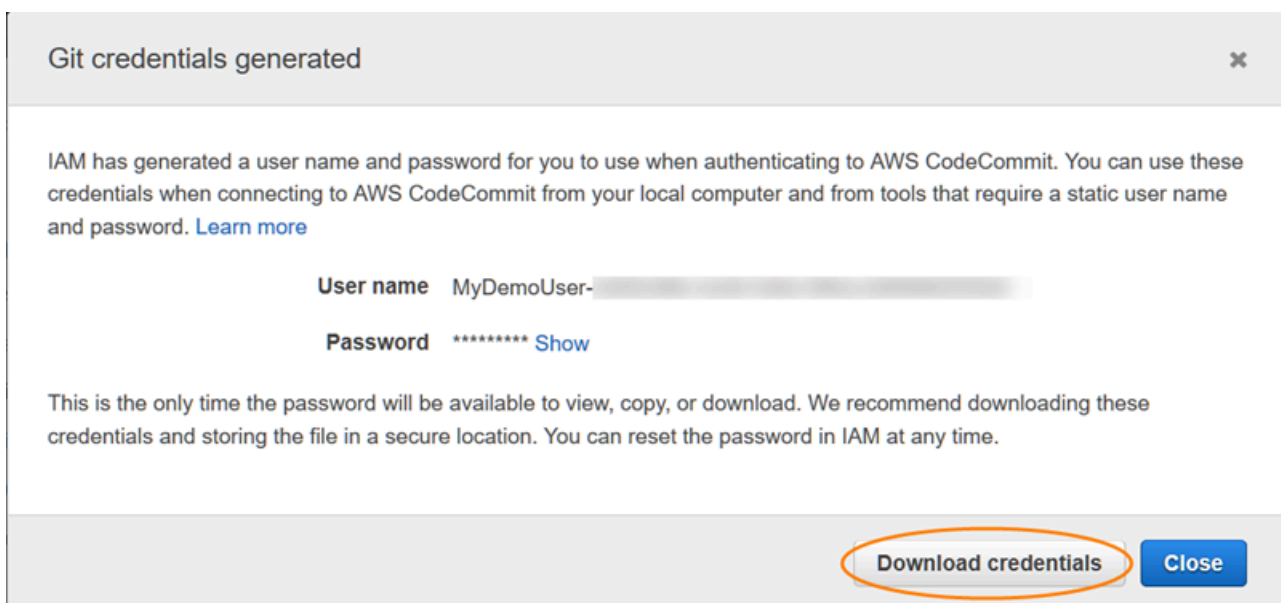


The screenshot shows the AWS IAM console interface. On the left is a navigation menu with options like Dashboard, Groups, Users, Roles, Policies, Identity Providers, Account Settings, and Credential Report. The main content area is titled 'SSH keys for AWS CodeCommit'. It includes a sub-header 'Use SSH public keys to authenticate to AWS CodeCommit repositories.' and a button 'Upload SSH public key'. Below this is a table with columns 'SSH key ID', 'Uploaded', and 'Status', which currently shows 'No results'. Underneath the table is the section 'HTTPS Git credentials for AWS CodeCommit', which includes a sub-header 'Generate a user name and password you can use to authenticate HTTPS connections to AWS CodeCommit repositories. You can generate and store up to 2 sets of credentials.' and a button 'Generate' that is circled in orange.

Note


Anda tidak dapat memilih nama pengguna atau kata sandi Anda sendiri untuk kredensial Git. Untuk informasi selengkapnya, lihat [Menggunakan Kredensial Git dan HTTPS dengan CodeCommit](#).

4. Salin nama pengguna dan sandi yang dihasilkan IAM untuk Anda, baik dengan menampilkan, menyalin, dan kemudian menyisipkan informasi ini ke file aman pada komputer lokal Anda, atau dengan memilih Unduh kredensial untuk mengunduh informasi ini sebagai file .CSV. Anda memerlukan informasi ini untuk terhubung CodeCommit.



The screenshot shows a dialog box titled 'Git credentials generated'. It contains the following text: 'IAM has generated a user name and password for you to use when authenticating to AWS CodeCommit. You can use these credentials when connecting to AWS CodeCommit from your local computer and from tools that require a static user name and password. [Learn more](#)'. Below this, it displays 'User name MyDemoUser-' followed by a redacted area, and 'Password ***** Show'. At the bottom, it states: 'This is the only time the password will be available to view, copy, or download. We recommend downloading these credentials and storing the file in a secure location. You can reset the password in IAM at any time.' At the bottom right, there are two buttons: 'Download credentials' (circled in orange) and 'Close'.

Setelah Anda menyimpan kredensialnya, pilih Tutup.

 Important

Ini adalah satu-satunya kesempatan Anda untuk menyimpan nama pengguna dan kata sandi. Jika Anda tidak menyimpannya, Anda dapat menyalin nama pengguna dari konsol IAM, tetapi Anda tidak dapat mencari kata sandi. Anda harus mengatur ulang kata sandi dan kemudian menyimpannya.

Langkah 4: Connect ke CodeCommit konsol dan kloning repositori

Jika administrator telah mengirimkan Anda nama dan detail koneksi untuk CodeCommit repositori, Anda dapat melewati langkah ini dan mengkloning repositori secara langsung.

Untuk terhubung ke CodeCommit repositori

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di pemilih wilayah, pilih Wilayah AWS tempat repositori dibuat. Repositori khusus untuk file. Wilayah AWS Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Temukan repositori yang ingin Anda sambungkan dari daftar dan pilih. Pilih URL Kloning, lalu pilih protokol yang ingin Anda gunakan saat mengkloning atau menghubungkan ke repositori. Langkah ini akan menyalin URL klon tersebut.
 - Salin URL HTTPS jika Anda menggunakan kredensial Git dengan pengguna IAM Anda atau pembantu kredensial yang disertakan dengan AWS CLI.
 - Salin URL HTTPS (GRC) jika Anda menggunakan perintah `git-remote-codecommit` di komputer lokal Anda.
 - Salin URL SSH jika Anda menggunakan pasangan kunci publik/swasta SSH dengan pengguna IAM Anda.

 Note

Jika Anda melihat halaman Selamat Datang alih-alih daftar repositori, tidak ada repositori yang terkait dengan AWS akun Anda di Wilayah AWS tempat Anda masuk. Untuk membuat repositori, lihat [the section called “Buatlah sebuah repositori”](#) atau ikuti langkah-langkah di tutorial [Memulai dengan Git dan CodeCommit](#).

4. Buka terminal, baris perintah, atau Git shell. Jalankan git clone dengan URL klon HTTPS yang Anda salin untuk mengkloning repositori. Misalnya, untuk mengkloning repositori yang dinamai *MyDemoRepo* lokal yang dinamai *my-demo-repo* di Wilayah Timur AS (Ohio):

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Saat pertama kali Anda terhubung, Anda diminta nama pengguna dan kata sandi untuk repositori. Bergantung pada konfigurasi komputer lokal Anda, prompt ini berasal dari sistem manajemen kredensi untuk sistem operasi, utilitas pengelola kredensi untuk versi Git Anda (misalnya, Manajer Kredensial Git yang disertakan dalam Git untuk Windows), IDE Anda, atau Git itu sendiri. Masukkan nama pengguna dan kata sandi yang dihasilkan untuk kredensial Git di IAM (yang Anda buat di [Langkah 3: Buat kredensi Git untuk koneksi HTTPS CodeCommit](#)). Tergantung pada sistem operasi Anda dan perangkat lunak lainnya, informasi ini mungkin disimpan untuk Anda di penyimpanan kredensial atau utilitas manajemen kredensial. Jika demikian, Anda tidak harus diminta lagi kecuali jika Anda mengubah kata sandi, menonaktifkan kredensial Git, atau menghapus kredensial Git di IAM.

Jika Anda tidak memiliki penyimpanan kredensial atau utilitas manajemen kredensial yang dikonfigurasi pada komputer lokal Anda, Anda dapat menginstal salah satu. Untuk informasi selengkapnya tentang Git dan cara mengelola kredensial, lihat [Penyimpanan Kredensial](#) dalam dokumentasi Git.

Untuk informasi selengkapnya, lihat [Connect ke CodeCommit repositori dengan mengkloning repositori](#) dan [Buatlah sebuah commit](#).

Langkah selanjutnya

Anda telah menyelesaikan prasyaratnya. Ikuti langkah-langkahnya [Memulai dengan CodeCommit](#) untuk mulai menggunakan CodeCommit.

Untuk mempelajari cara membuat dan mendorong commit pertama Anda, lihat [Buat komit di AWS CodeCommit](#). Jika Anda baru mengenal Git, Anda mungkin juga ingin meninjau informasi di [Di mana saya dapat mempelajari lebih lanjut tentang Git?](#) dan [Memulai dengan Git dan AWS CodeCommit](#).

Langkah-langkah persiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit

Jika Anda ingin terhubung keCodeCommitmenggunakan akun root, akses federasi, atau kredensi sementara, Anda harus mengatur akses menggunakangit-remote-codecommit. Utilitas ini menyediakan metode sederhana untuk mendorong dan menarik kode dariCodeCommitrepositori dengan memperluas Git. Ini adalah metode yang direkomendasikan untuk mendukung koneksi yang dibuat dengan akses gabungan, penyedia identitas, dan kredensial sementara. Untuk menetapkan izin ke identitas federasi, Anda membuat peran dan menentukan izin untuk peran tersebut. Ketika identitas federasi mengautentikasi, identitas dikaitkan dengan peran dan diberikan izin yang ditentukan oleh peran. Untuk informasi tentang peran federasi, lihat[Membuat peran untuk Penyedia Identitas pihak ketiga](#)diPanduan Pengguna IAM. Jika Anda menggunakan Pusat Identitas IAM, Anda mengonfigurasi set izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah diautentikasi, IAM Identity Center mengkorelasikan izin yang disetel ke peran dalam IAM. Untuk informasi tentang set izin, lihat[Set izin](#)diAWS IAM Identity CenterPanduan Pengguna.

Anda juga dapat menggunakan git-remote-codecommit dengan pengguna IAM. Tidak seperti metode koneksi HTTPS lainnya, git-remote-codecommit tidak memerlukan pengaturan kredensial Git untuk pengguna.

Note

Beberapa IDE tidak support format URL klon yang digunakan oleh git-remote-codecommit. Anda mungkin harus secara manual mengkloning repositori ke komputer lokal Anda sebelum Anda dapat bekerja dengan repositori di IDE yang Anda pilih. Untuk informasi selengkapnya, lihat [Mengatasi masalah git-remote-codecommit dan AWS CodeCommit](#).

Prosedur ini ditulis dengan asumsi bahwa Anda memiliki akun Amazon Web Services, telah membuat setidaknya satu repositori diCodeCommit, dan gunakan pengguna IAM dengan kebijakan terkelola saat menyambung keCodeCommitrepositori. Untuk informasi tentang cara mengkonfigurasi akses untuk pengguna gabungan dan jenis kredensial berputar lainnya, lihat [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#).

Topik

- [Langkah 0: Instal prasyarat untukgit-remote-codecommit](#)
- [Langkah 1: Konfigurasi awal untukCodeCommit](#)

- [Langkah 2: Instalgit-remote-codecommit](#)
- [Langkah 3: Hubungkan keCodeCommitkonsol dan kloning repositori](#)
- [Langkah selanjutnya](#)

Langkah 0: Instal prasyarat untukgit-remote-codecommit

Sebelum Anda dapat menggunakan git-remote-codecommit, Anda harus menginstal beberapa prasyarat di komputer lokal Anda. Ini termasuk:

- Python (versi 3 atau yang lebih baru) dan manajer pakatnya, pip, jika belum terpasang. Untuk mengunduh dan menginstal versi terbaru dari Python, kunjungi [situs Python](#).
- Git

Note

Ketika Anda menginstal Python pada Windows, pastikan bahwa Anda memilih opsi untuk menambahkan Python ke path.

git-remote-codecommit memerlukan pip versi 9.0.3 atau versi lebih baru. Untuk memeriksa versi pip Anda, buka terminal atau baris perintah dan jalankan perintah berikut:

```
pip --version
```

Anda dapat menjalankan dua perintah berikut untuk memperbarui versi pip Anda ke versi terbaru:

```
curl -O https://bootstrap.pypa.io/get-pip.py  
python3 get-pip.py --user
```

Untuk bekerja dengan file, komit, dan informasi lainnya diCodeCommitrepositori, Anda harus menginstal Git pada mesin lokal Anda. CodeCommitmendukung Git versi 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git.

Untuk menginstal Git, kami merekomendasikan situs web seperti [Pengunduhan Git](#).

Note

Git adalah platform yang berkembang dan diperbarui secara teratur. Terkadang, perubahan fitur dapat memengaruhi cara kerjanya CodeCommit. Jika Anda mengalami masalah dengan versi tertentu dari Git dan CodeCommit, tinjau informasi di [Memecahkan masalah](#).

Langkah 1: Konfigurasi awal untuk CodeCommit

Ikuti langkah-langkah untuk membuat pengguna IAM, mengkonfigurasi dengan kebijakan yang sesuai, mendapatkan access key dan secret key, dan menginstal dan mengkonfigurasi AWS CLI.

Untuk membuat dan mengkonfigurasi pengguna IAM untuk mengakses CodeCommit

1. Buat akun Amazon Web Services dengan membuka <http://aws.amazon.com> dan memilih Mendaftar.
2. Membuat pengguna IAM, atau menggunakan yang sudah ada, di akun Amazon Web Services Anda. Pastikan Anda memiliki access key ID dan secret access key yang terkait dengan pengguna IAM. Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di akun Amazon Web Services Anda](#).

Note

CodeCommit memerlukan AWS Key Management Service. Jika Anda menggunakan pengguna IAM yang sudah ada, pastikan tidak ada kebijakan yang dilampirkan pada pengguna yang secara tegas menolak AWS KMS tindakan yang diperlukan oleh CodeCommit. Untuk informasi selengkapnya, lihat [AWS KMS dan enkripsi](#).

3. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
4. Di konsol IAM, di panel navigasi, pilih Pengguna, dan kemudian pilih pengguna IAM yang ingin Anda konfigurasi untuk akses CodeCommit.
5. Pada tab Izin, pilih Tambahkan izin.
6. Dalam Berikan izin, pilih Lampirkan kebijakan yang ada secara langsung.
7. Dari daftar kebijakan, pilih AWS CodeCommitPowerUser atau kebijakan terkelola lainnya untuk akses CodeCommit. Untuk informasi selengkapnya, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).


Setelah memilih kebijakan yang ingin dilampirkan, pilih Berikutnya: Tinjau untuk meninjau daftar kebijakan untuk dilampirkan ke pengguna IAM. Jika daftar sudah benar, pilih Tambahkan izin.

Untuk informasi lebih lanjut tentang CodeCommit kebijakan terkelola dan berbagi akses ke repositori dengan grup dan pengguna lain, lihat [Bagikan sebuah repositori](#) dan [Kontrol autentikasi dan akses untuk AWS CodeCommit](#).

Untuk menginstal dan mengonfigurasi AWS CLI

1. Pada komputer lokal Anda, unduh dan instal AWS CLI. Ini adalah prasyarat untuk berinteraksi dengan CodeCommit dari baris perintah. Kami merekomendasikan agar Anda menginstal AWS CLI versi 2. Ini adalah versi utama terbaru dari AWS CLI dan mendukung semua fitur terbaru. Ini adalah satu-satunya versi AWS CLI yang mendukung penggunaan akun root, akses federasi, atau kredensial sementara dengan git-remote-codecommit.

Untuk informasi selengkapnya, lihat [Menyiapkan Antarmuka Baris Perintah AWS](#).

 Note

CodeCommit bekerja hanya dengan AWS CLI versi 1.7.38 dan yang lebih baru. Sebagai praktik terbaik, instal atau tingkatkan AWS CLI untuk versi terbaru yang tersedia. Untuk menentukan versi AWS CLI yang telah Anda instal, jalankan perintah `aws --version`. Untuk meng-upgrade versi lama AWS CLI ke versi terbaru, lihat [Menginstal AWS Command Line Interface](#).

2. Jalankan perintah ini untuk memverifikasi bahwa CodeCommit perintah untuk AWS CLI diinstal.

```
aws codecommit help
```

Perintah ini mengembalikan daftar CodeCommit perintah.

3. Konfigurasi AWS CLI dengan profil dengan menggunakan perintah `configure`, sebagai berikut:

```
aws configure
```

Saat diminta, tentukan AWS kunci akses dan AWS kunci akses rahasia pengguna IAM untuk digunakan CodeCommit. Juga, pastikan untuk menentukan Wilayah AWS di mana repositori ada,

seperti `us-east-2`. Saat diminta untuk format output default, tentukan `json`. Misalnya, jika Anda mengonfigurasi profil untuk pengguna IAM:

```
AWS Access Key ID [None]: Type your IAM user AWS access key ID here, and then press Enter
AWS Secret Access Key [None]: Type your IAM user AWS secret access key here, and then press Enter
Default region name [None]: Type a supported region for CodeCommit here, and then press Enter
Default output format [None]: Type json here, and then press Enter
```

Untuk informasi selengkapnya tentang membuat dan mengonfigurasi profil untuk digunakan dengan AWS CLI, lihat berikut ini:

- [Profil Bernama](#)
- [Menggunakan Peran IAM dalam AWS CLI](#)
- [Tetapkan perintah](#)
- [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#)

Untuk terkoneksi ke repositori atau sumber daya di Wilayah AWS lain, Anda harus mengkonfigurasi ulang AWS CLI dengan nama wilayah default. Nama Wilayah default yang didukung untuk CodeCommit termasuk:

- `us-east-2`
- `us-east-1`
- `eu-west-1`
- `us-west-2`
- `ap-northeast-1`
- `ap-southeast-1`
- `ap-southeast-2`
- `ap-southeast-3`
- `me-central-1`
- `eu-central-1`
- `ap-northeast-2`

- `sa-east-1`

- us-west-1
- eu-west-2
- ap-south-1
- ap-south-1
- ca-central-1
- us-gov-west-1
- us-gov-east-1
- eu-north-1
- ap-east-1
- me-south-1
- cn-north-1
- cn-northwest-1
- eu-south-1
- ap-northeast-3
- af-south-1
- pusat-1

Untuk informasi selengkapnya tentang CodeCommit dan Wilayah AWS, lihat [Wilayah dan titik akhir koneksi Git](#). Untuk informasi selengkapnya tentang IAM, access key, dan kunci rahasia, lihat [Bagaimana Saya Mendapatkan Kredensial?](#) dan [Mengelola Access Key untuk Pengguna IAM](#). Untuk informasi selengkapnya tentang AWS CLI dan profil, lihat [Profil Bernama](#).

Langkah 2: Instalgit-remote-codecommit

Ikuti langkah-langkah berikut untuk menginstal git-remote-codecommit.

Untuk menginstal git-remote-codecommit

1. Pada terminal atau baris perintah, jalankan perintah berikut:

```
pip install git-remote-codecommit
```


Note

Bergantung pada sistem operasi dan konfigurasi Anda, Anda mungkin perlu menjalankan perintah ini dengan izin tinggi, seperti `sudo`, atau menggunakan parameter `--user` untuk menginstal ke direktori yang tidak memerlukan hak istimewa khusus, seperti akun pengguna Anda saat ini. Misalnya, pada komputer yang menjalankan Linux, macOS, atau Unix:

```
sudo pip install git-remote-codecommit
```

Di komputer yang menjalankan Windows:

```
pip install --user git-remote-codecommit
```

2. Pantau proses instalasi sampai Anda melihat pesan sukses.

Langkah 3: Hubungkan keCodeCommitkonsol dan kloning repositori

Jika administrator telah mengirimkan Anda URL klon untuk digunakan `git-remote-codecommit` untuk CodeCommit repositori, Anda dapat melewati koneksi ke konsol dan mengkloning repositori secara langsung.

Untuk terhubung ke aCodeCommitrepositori

1. BukaCodeCommitkonsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Dalam pemilih wilayah, pilih Wilayah AWS di mana repositori dibuat. Repositori adalah khusus untuk sebuah Wilayah AWS. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Temukan repositori yang ingin Anda sambungkan dari daftar dan pilih. Pilih URL Kloning, lalu pilih protokol yang ingin Anda gunakan saat mengkloning atau menghubungkan ke repositori. Langkah ini akan menyalin URL klon tersebut.
 - Salin URL HTTPS jika Anda menggunakan kredensial Git dengan pengguna IAM Anda atau pembantu kredensial yang disertakan dengan AWS CLI.
 - Salin URL HTTPS (GRC) jika Anda menggunakan perintah `git-remote-codecommit` di komputer lokal Anda.

- Salin URL SSH jika Anda menggunakan pasangan kunci publik/swasta SSH dengan pengguna IAM Anda.

Note

Jika Anda melihat halaman Selamat Datang bukan daftar repositori, tidak ada repositori yang terkait dengan akun AWS Anda di Wilayah AWS tempat Anda masuk. Untuk membuat repositori, lihat [the section called “Buatlah sebuah repositori”](#) atau ikuti langkah-langkah di tutorial [Memulai dengan Git dan CodeCommit](#).

4. Di terminal atau command prompt, kloning repositori dengan perintah git clone. Gunakan HTTPSgit-remote-codecommitURL yang Anda salin dan namaAWS CLIprofil, jika Anda membuat profil bernama. Jika Anda tidak menentukan profil, perintah mengasumsikan profil default. Repo lokal dibuat di subdirektori direktori di mana Anda menjalankan perintah. Misalnya, untuk mengkloning repositori bernama*MyDemoRepo* ke repo lokal bernama*my-demo-repo*:

```
git clone codecommit://MyDemoRepo my-demo-repo
```

Untuk mengkloning repositori yang sama menggunakan profil bernama*CodeCommitProfile*:

```
git clone codecommit://CodeCommitProfile@MyDemoRepo my-demo-repo
```

Untuk mengkloning repositori dalam Wilayah AWS yang berbeda dari yang dikonfigurasi di profil Anda, termasuk nama Wilayah AWS. Misalnya:

```
git clone codecommit::ap-northeast-1://MyDemoRepo my-demo-repo
```

Langkah selanjutnya

Anda telah menyelesaikan prasyaratnya. Ikuti langkah-langkahnya di [Memulai dengan CodeCommit](#) untuk mulai menggunakanCodeCommit.

Untuk mempelajari cara membuat dan mendorong commit pertama Anda, lihat [Buat komit di AWS CodeCommit](#). Jika Anda baru mengenal Git, Anda mungkin juga ingin meninjau informasi di [Di mana saya dapat mempelajari lebih lanjut tentang Git?](#) dan [Memulai dengan Git dan AWS CodeCommit](#).

Mengatur koneksi dari alat pengembangan menggunakan kredensial Git

Setelah Anda mengonfigurasi kredensi Git untuk AWS CodeCommit di konsol IAM, Anda dapat menggunakan kredensial-kredensi tersebut dengan alat pengembangan apa pun yang mendukung kredensial Git. Misalnya, Anda dapat mengonfigurasi akses ke CodeCommit repositori Anda di AWS Cloud9, Visual Studio, Eclipse, Xcode, IntelliJ, atau lingkungan pengembangan terintegrasi (IDE) apa pun yang mengintegrasikan kredensi Git. Setelah Anda mengkonfigurasi akses, Anda dapat mengedit kode Anda, melakukan perubahan Anda, dan mengirim langsung dari IDE atau alat pengembangan lainnya.

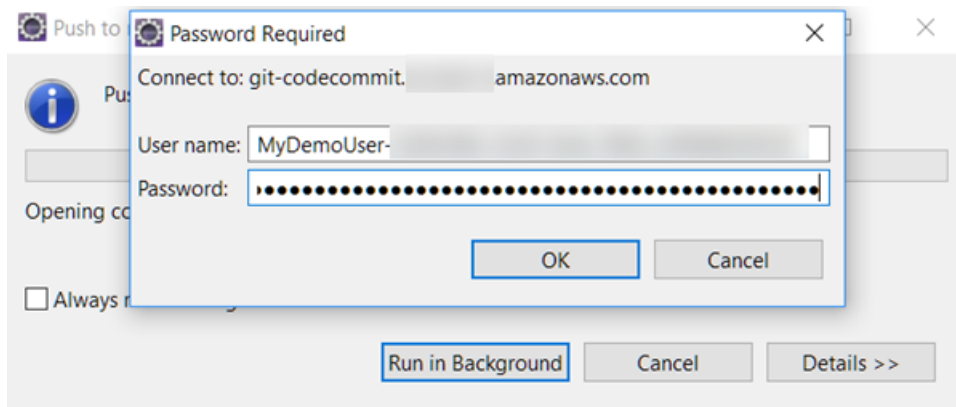
Note

Jika Anda mengakses CodeCommit repositori menggunakan akses federasi, kredensial sementara, atau penyedia identitas web, Anda tidak dapat menggunakan kredensi Git. Kami menyarankan agar Anda mengatur komputer lokal Anda menggunakan perintah `git-remote-codecommit`. Namun, tidak semua IDE sepenuhnya kompatibel dengan pembantu remote Git seperti `git-remote-codecommit`. Jika Anda menemukan masalah, lihat [Mengatasi masalah git-remote-codecommit dan AWS CodeCommit](#).

Topik

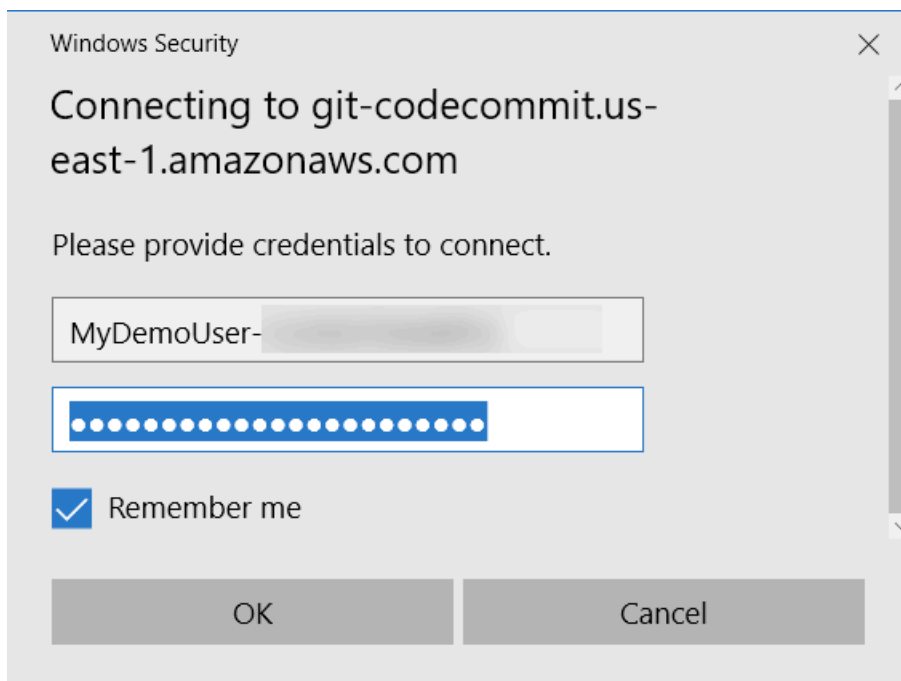
- [Integrasi AWS Cloud9 dengan AWS CodeCommit](#)
- [Mengintegrasikan Visual Studio dengan AWS CodeCommit](#)
- [Mengintegrasikan Eclipse dengan AWS CodeCommit](#)

Ketika diminta oleh IDE atau alat pengembangan untuk nama pengguna dan kata sandi yang digunakan untuk menyambung ke CodeCommit repositori, berikan kredensi Git untuk nama Pengguna dan Kata Sandi yang Anda buat di IAM. Misalnya, jika Anda diminta untuk nama pengguna dan kata sandi di Eclipse, Anda akan memberikan kredensial Git Anda sebagai berikut:



Untuk informasi selengkapnya tentang Wilayah AWS dan titik akhir CodeCommit, lihat [Wilayah dan titik akhir koneksi Git](#).

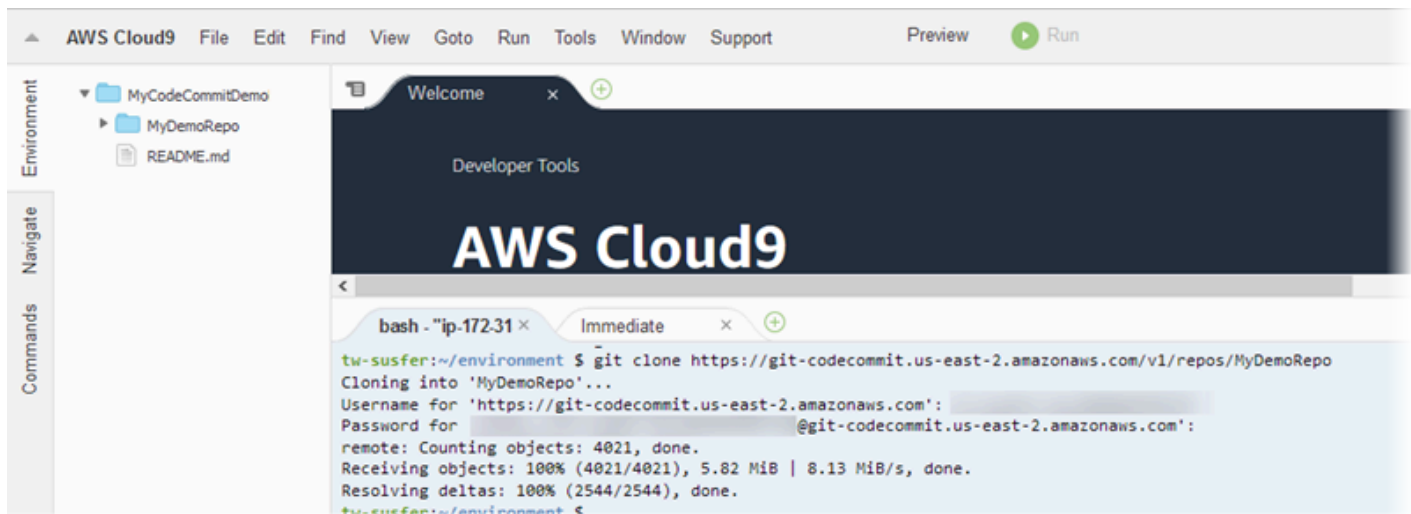
Anda mungkin juga melihat permintaan dari sistem operasi untuk menyimpan nama pengguna dan kata sandi Anda. Misalnya, di Windows, Anda akan memberikan kredensial Git Anda sebagai berikut:



Untuk informasi tentang mengkonfigurasi kredensial Git untuk program perangkat lunak tertentu atau alat pengembangan, lihat dokumentasi produk.

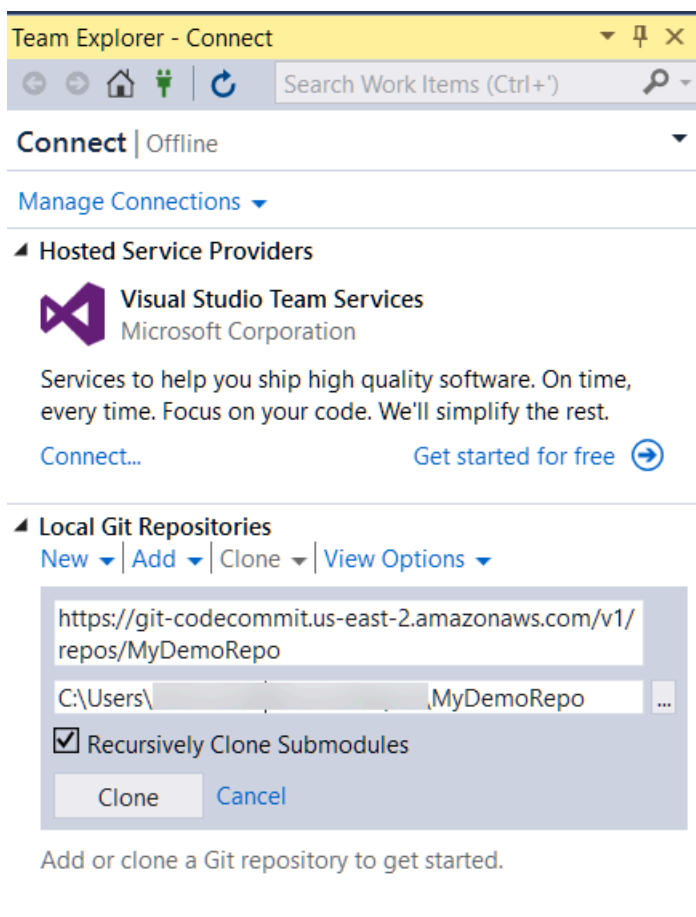
Berikut ini bukan daftar lengkap IDE. Tautan disediakan semata-mata untuk membantu Anda mempelajari lebih lanjut tentang alat ini. AWS tidak bertanggung jawab atas isi dari salah satu topik ini.

- [AWS Cloud9](#)



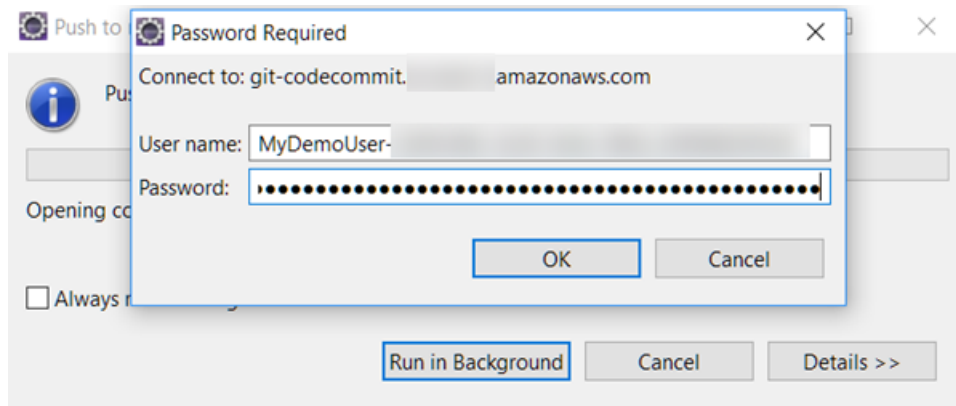
- [Studio Visual](#)

Atau, instal AWS Toolkit for Visual Studio. Untuk informasi selengkapnya, lihat [Mengintegrasikan Visual Studio dengan AWS CodeCommit](#).



- [eGit dengan Eclipse](#)

Atau, instal AWS Toolkit for Eclipse. Untuk informasi selengkapnya, lihat [Mengintegrasikan Eclipse dengan AWS CodeCommit](#).



- [Xcode](#)

Integrasi AWS Cloud9 dengan AWS CodeCommit

Anda dapat menggunakan AWS Cloud9 untuk membuat perubahan kode CodeCommit repositori. AWS Cloud9 berisi kumpulan alat yang dapat Anda gunakan untuk menulis kode dan membangun, menjalankan, menguji, debug, dan merilis perangkat lunak. Anda dapat mengkloning repositori yang ada, membuat repositori, mengkomit dan mengirim perubahan kode ke repositori, dan banyak lagi, semuanya dari lingkungan pengembangan EC2 AWS Cloud9 Anda. Lingkungan pengembangan EC2 AWS Cloud9 umumnya dikonfigurasi dengan AWS CLI, peran Amazon EC2, dan Git, sehingga dalam banyak kasus, Anda dapat menjalankan beberapa perintah sederhana dan mulai berinteraksi dengan repositori Anda.

Untuk menggunakan AWS Cloud9 dengan CodeCommit, Anda memerlukan yang berikut ini:

- Suatu Lingkungan pengembangan EC2 AWS Cloud9 berjalan di Amazon Linux.
- AWS Cloud9 IDE dibuka di peramban web.
- Pengguna IAM dengan salah satu CodeCommit kebijakan terkelola dan salah satu AWS Cloud9 dikelola kebijakan diterapkan untuk itu.

Untuk informasi selengkapnya, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit dan Memahami dan Mendapatkan Kredensial Keamanan Anda](#).

Note

Topik ini menjelaskan pengaturan integrasi dengan CodeCommit dan AWS Cloud9 dengan akses umum dari Internet. Anda dapat mengatur akses ke CodeCommit dan AWS Cloud9 di lingkungan yang terisolasi, tapi itu membutuhkan langkah-langkah tambahan. Untuk informasi selengkapnya, lihat :

- [Menggunakan AWS CodeCommit dengan antarmuka VPC endpoint](#)
- [Mengakses instans Amazon EC2 tanpa AWS Systems Manager](#)
- [Bekerja dengan Lingkungan Bersama](#)
- [Bagikan VPC Anda dengan akun lain](#)
- [Posting blog: Mengisolasi akses jaringan ke AWS Cloud9 lingkungan](#)

Topik

- [Langkah 1: Buat AWS Cloud9 Lingkungan pengembangan](#)
- [Langkah 2: Konfigurasi AWS CLI pembantu kredensial Anda AWS Cloud9 Lingkungan pengembangan EC2](#)
- [Langkah 3: Klon CodeCommit repositori ke dalam AWS Cloud9 Lingkungan pengembangan EC2](#)
- [Langkah selanjutnya](#)

Langkah 1: Buat AWS Cloud9 Lingkungan pengembangan

AWS Cloud9 meng-hosting lingkungan pengembangan Anda pada instans Amazon EC2. Ini adalah cara termudah untuk mengintegrasikan, karena Anda dapat menggunakan AWS kredensial sementara yang dikelola untuk instans untuk terhubung ke CodeCommit repositori. Jika Anda ingin menggunakan server Anda sendiri, lihat [Panduan Pengguna AWS Cloud9](#).

Untuk membuat lingkungan AWS Cloud9

1. Masuk ke AWS sebagai pengguna IAM yang telah Anda konfigurasi dan buka konsol AWS Cloud9.
2. Dalam konsol AWS Cloud9, pilih Buat lingkungan.
3. Di Langkah 1: Lingkungan nama, masukkan nama dan deskripsi opsional untuk lingkungan, lalu pilih Langkah selanjutnya.

4. Di Langkah 2: Konfigurasi Pengaturan, konfigurasi lingkungan Anda sebagai berikut:
 - Dalam Tipe lingkungan, pilih Buat instans baru untuk lingkungan (EC2).
 - Dalam Jenis instans, pilih jenis instans yang sesuai untuk lingkungan pengembangan Anda. Misalnya, jika Anda hanya menjelajahi layanan, Anda dapat memilih default t2.micro. Jika Anda berniat menggunakan lingkungan ini untuk pekerjaan pengembangan, pilih jenis instans yang lebih besar.
 - Menerima pengaturan default lainnya kecuali jika Anda memiliki alasan untuk memilih sebaliknya (misalnya, organisasi Anda menggunakan VPC tertentu, atau akun Amazon Web Services Anda tidak memiliki VPC dikonfigurasi), dan kemudian pilih Langkah selanjutnya.
5. Di Langkah 3: Peninjauan, tinjau pengaturan Anda. Pilih Langkah sebelumnya jika Anda ingin melakukan perubahan. Jika tidak, pilih Buat lingkungan.

Membuat lingkungan dan menghubungkannya untuk pertama kalinya membutuhkan waktu beberapa menit. Jika tampaknya memakan waktu yang sangat lama, lihat [Pemecahan Masalah](#) dalam Panduan Pengguna AWS Cloud9.

6. Setelah Anda terhubung ke lingkungan Anda, periksa untuk melihat apakah Git sudah diinstal dan versi yang didukung dengan menjalankan perintah `git --version` di jendela terminal.

Jika Git tidak diinstal, atau jika bukan versi yang didukung, instal versi yang didukung. CodeCommit mendukung versi Git 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git. Untuk menginstal Git, kami merekomendasikan situs web seperti [Pengunduhan Git](#).

Tip

Bergantung pada sistem operasi lingkungan Anda, Anda mungkin dapat menggunakan perintah `yum` dengan perintah `sudo` untuk menginstal pembaruan, termasuk Git. Sebagai contoh, urutan perintah administratif mungkin menyerupai tiga perintah berikut:

```
sudo yum -y update
sudo yum -y install git
git --version
```

7. Mengkonfigurasi nama pengguna dan email untuk dikaitkan dengan commit Git Anda dengan menjalankan perintah `git config`. Misalnya:


```
git config --global user.name "Mary Major"  
git config --global user.email mary.major@example.com
```

Langkah 2: Konfigurasi AWS CLI pembantu kredensial Anda AWS Cloud9 Lingkungan pengembangan EC2

Setelah Anda membuat AWS Cloud9 lingkungan, Anda dapat mengkonfigurasi AWS CLI kredensial helper untuk mengelola kredensial untuk koneksi ke CodeCommit repositori. Lingkungan pengembangan AWS Cloud9 dilengkapi dengan kredensial sementara yang dikelola AWS yang terkait dengan pengguna IAM Anda. Anda menggunakan kredensial ini dengan pembantu kredensial AWS CLI.

1. Buka jendela terminal dan jalankan perintah berikut untuk memverifikasi bahwa AWS CLI terinstal:

```
aws --version
```

Jika berhasil, perintah ini akan mengembalikan versi AWS CLI. Untuk meng-upgrade versi lama AWS CLI ke versi terbaru, lihat [Menginstal AWS Command Line Interface](#).

2. Di terminal, jalankan perintah berikut untuk mengkonfigurasi pembantu kredensial AWS CLI untuk koneksi HTTPS:

```
git config --global credential.helper '!aws codecommit credential-helper $@'  
git config --global credential.UseHttpPath true
```

Tip

Pembantu kredensial menggunakan peran instans Amazon EC2 default untuk lingkungan pengembangan Anda. Jika Anda berniat untuk menggunakan lingkungan pengembangan untuk terhubung ke repositori yang tidak di-host di CodeCommit, baik mengkonfigurasi koneksi SSH ke repositori tersebut, atau mengkonfigurasi file `.gitconfig` lokal untuk menggunakan sistem manajemen kredensial alternatif saat menghubungkan ke repositori lainnya. Untuk informasi selengkapnya, lihat [Alat Git - Penyimpanan Kredensial](#) di situs web Git.

Langkah 3: Klon CodeCommit repositori ke dalam AWS Cloud9 Lingkungan pengembangan EC2

Setelah Anda mengkonfigurasi AWS CLI, Anda dapat mengkloning repositori CodeCommit Anda ke atasnya. Kemudian Anda bisa mulai bekerja dengan kode.

1. Di terminal, jalankan perintah git clone, menentukan HTTPS klon URL dari repositori yang ingin Anda kloning. Misalnya, jika Anda ingin mengkloning repositori bernama MyDemoRepo di Wilayah Timur AS (Ohio), Anda akan memasukkan:

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
```

Tip

Anda dapat menemukan URL Klon untuk repositori Anda di CodeCommit konsol dengan memilih URL Klon.

2. Saat kloning selesai, di navigasi samping, perluas folder untuk repositori Anda, dan pilih file yang ingin Anda buka untuk diedit. Atau, pilih Berkas lalu pilih Berkas Baru Untuk membuat file.
3. Ketika Anda telah selesai mengedit atau membuat file, di jendela terminal, ubah direktori ke repositori kloning Anda dan kemudian commit dan kirim perubahan Anda. Misalnya, jika Anda menambahkan file baru bernama *MyFile.py*:

```
cd MyDemoRepo
git commit -a MyFile.py
git commit -m "Added a new file with some code improvements"
git push
```

Langkah selanjutnya

Untuk informasi selengkapnya, lihat [AWS Cloud9 Panduan Pengguna](#) dan [Sampel CodeCommit AWS Cloud9](#). Untuk informasi selengkapnya tentang penggunaan Git dengan CodeCommit, lihat [Memulai dengan Git dan AWS CodeCommit](#).

Mengintegrasikan Visual Studio dengan AWS CodeCommit

Anda dapat menggunakan Visual Studio untuk membuat perubahan kode dalam CodeCommit repositori. AWS Toolkit for Visual Studio sekarang memiliki fitur yang membuat bekerja dengan

CodeCommit lebih mudah dan lebih nyaman ketika bekerja di Visual Studio.. Integrasi Toolkit for Visual Studio dirancang untuk bekerja dengan kredensial Git dan pengguna IAM. Anda dapat mengkloning repositori yang ada, membuat repositori, melakukan dan mengirim perubahan kode ke repositori, dan banyak lagi.

⚠ Important

Toolkit for Visual Studio tersedia untuk instalasi pada sistem operasi Windows saja. Jika Anda mencari informasi tentang bekerja dengan Visual Studio Code, lihat [AWS Toolkit for Visual Studio Code](#).

Jika Anda telah menggunakan Toolkit for Visual Studio sebelumnya, Anda mungkin sudah akrab dengan pengaturan kredensial profil AWS yang berisi access key dan secret key. Profil kredensial yang digunakan dalam Toolkit for Visual Studio untuk mengaktifkan panggilan ke API AWS layanan (misalnya, ke Amazon S3 untuk merincikan bucket atau CodeCommit merincikan repositori). Untuk menarik dan mengirim kode ke CodeCommit repositori, Anda juga perlu kredensial Git. Jika Anda tidak memiliki kredensial Git, Toolkit for Visual Studio dapat menghasilkan dan menerapkan kredensialnya untuk Anda. Ini bisa menghemat banyak waktu.

Untuk menggunakan Visual Studio dengan CodeCommit, Anda memerlukan yang berikut ini:

- IAM pengguna dengan satu set kredensial valid (access key dan secret key) yang dikonfigurasi untuk itu. Pengguna IAM ini juga harus memiliki:

Salah satu kebijakan yang CodeCommit dikelola dan kebijakan yang `IAMSelfManageServiceSpecificCredentials` dikelola yang diterapkan padanya.

ATAU

Jika pengguna IAM sudah memiliki kredensial Git yang dikonfigurasi, salah satu kebijakan yang CodeCommit dikelola atau izin yang setara.

Untuk informasi selengkapnya, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit dan Memahami dan Mendapatkan Kredensial Keamanan Anda](#).

- AWS Toolkit for Visual Studio yang diinstal pada komputer di mana Anda telah menginstal Visual Studio. Untuk informasi lebih lanjut, lihat [Mengatur AWS Toolkit for Visual Studio](#).

Untuk informasi selengkapnya tentang penggunaan AWS Toolkit for Visual Studio dengan CodeCommit, lihat [Menggunakan AWS CodeCommit dengan Visual Studio Team Explorer](#) di Toolkit untuk Panduan Pengguna Visual Studio.

Mengintegrasikan Eclipse dengan AWS CodeCommit

Anda dapat menggunakan Eclipse untuk membuat perubahan kode dalam CodeCommit repositori. Integrasi Toolkit for Eclipse dirancang untuk bekerja dengan kredensial Git dan pengguna IAM. Anda dapat mengkloning repositori yang ada, membuat repositori, melakukan dan mengirim perubahan kode ke repositori, dan banyak lagi.

Untuk menggunakan Toolkit for Eclipse dengan CodeCommit, Anda memerlukan yang berikut:

- Eclipse diinstal di komputer lokal Anda.
- IAM pengguna dengan satu set kredensial valid (access key dan secret key) dikonfigurasi untuk itu. Pengguna IAM ini juga harus memiliki:

Salah satu kebijakan yang CodeCommit dikelola dan kebijakan `IAMSelfManageServiceSpecificCredentials` terkelola yang diterapkan padanya.

ATAU

Jika pengguna IAM sudah memiliki kredensial Git yang dikonfigurasi, salah satu kebijakan CodeCommit terkelola atau izin yang setara.

Untuk informasi selengkapnya, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#) dan [Memahami dan Mendapatkan Kredensial Keamanan Anda](#).

- Set kredensial Git aktif yang dikonfigurasi untuk pengguna di IAM. Untuk informasi selengkapnya, lihat [Langkah 3: Buat kredensial Git untuk koneksi HTTPS CodeCommit](#).

Topik

- [Langkah 1: Dapatkan access key dan secret key untuk pengguna IAM Anda](#)
- [Langkah 2: Instal AWS Toolkit for Eclipse dan koneksikan ke CodeCommit](#)
- [CodeCommit Kloning repositori dari Eclipse](#)
- [Buat CodeCommit repositori dari Eclipse](#)
- [Bekerja dengan CodeCommit repositori](#)

Langkah 1: Dapatkan access key dan secret key untuk pengguna IAM Anda

Jika Anda belum memiliki profil kredensial yang diatur pada komputer di mana Eclipse diinstal, Anda dapat [mengkonfigurasikannya dengan perintah AWS CLI dan `aws configure`](#). Atau, Anda dapat mengikuti langkah-langkah dalam prosedur ini untuk membuat dan mengunduh kredensial Anda. Menyediakannya ke Toolkit for Eclipse saat diminta.

Pengguna membutuhkan akses terprogram jika mereka ingin berinteraksi dengan AWS luar AWS Management Console. Cara untuk memberikan akses terprogram tergantung pada jenis pengguna yang mengakses AWS.

Untuk memberikan akses terprogram kepada pengguna, pilih salah satu opsi berikut.

Pengguna mana yang membutuhkan akses terprogram?	Ke	oleh
identitas tenaga kerja (Pengguna dikelola di IAM Identity Center)	Gunakan kredensi sementara untuk menandatangani permintaan terprogram ke AWS CLI, AWS SDK, atau AWS API.	Mengikuti instruksi untuk antarmuka yang ingin Anda gunakan. <ul style="list-style-type: none"> Untuk AWS CLI, lihat Mengkonfigurasi AWS CLI untuk digunakan AWS IAM Identity Center dalam Panduan AWS Command Line Interface Pengguna. Untuk AWS SDK, alat, dan AWS API, lihat otentikasi di IAM Identity Center di Panduan Referensi AWS SDK dan Alat.
IAM	Gunakan kredensi sementara untuk menandatangani permintaan terprogram ke AWS CLI, AWS SDK, atau AWS API.	Mengikuti petunjuk dalam Menggunakan kredensi sementara dengan AWS sumber daya di Panduan Pengguna IAM.

Pengguna mana yang membutuhkan akses terprogram?	Ke	oleh
IAM	(Tidak disarankan) Gunakan kredensi jangka panjang untuk menandatangani permintaan terprogram ke AWS CLI, AWS SDK, atau AWS API.	Mengikuti instruksi untuk antarmuka yang ingin Anda gunakan. <ul style="list-style-type: none"> • Untuk AWS CLI, lihat Mengautentikasi menggunakan kredensi pengguna IAM di Panduan AWS Command Line Interface Pengguna. • Untuk AWS SDK dan alat, lihat Mengautentikasi menggunakan kredensi jangka panjang di Panduan Referensi AWS SDK dan Alat. • Untuk AWS API, lihat Mengelola kunci akses untuk pengguna IAM di Panduan Pengguna IAM.

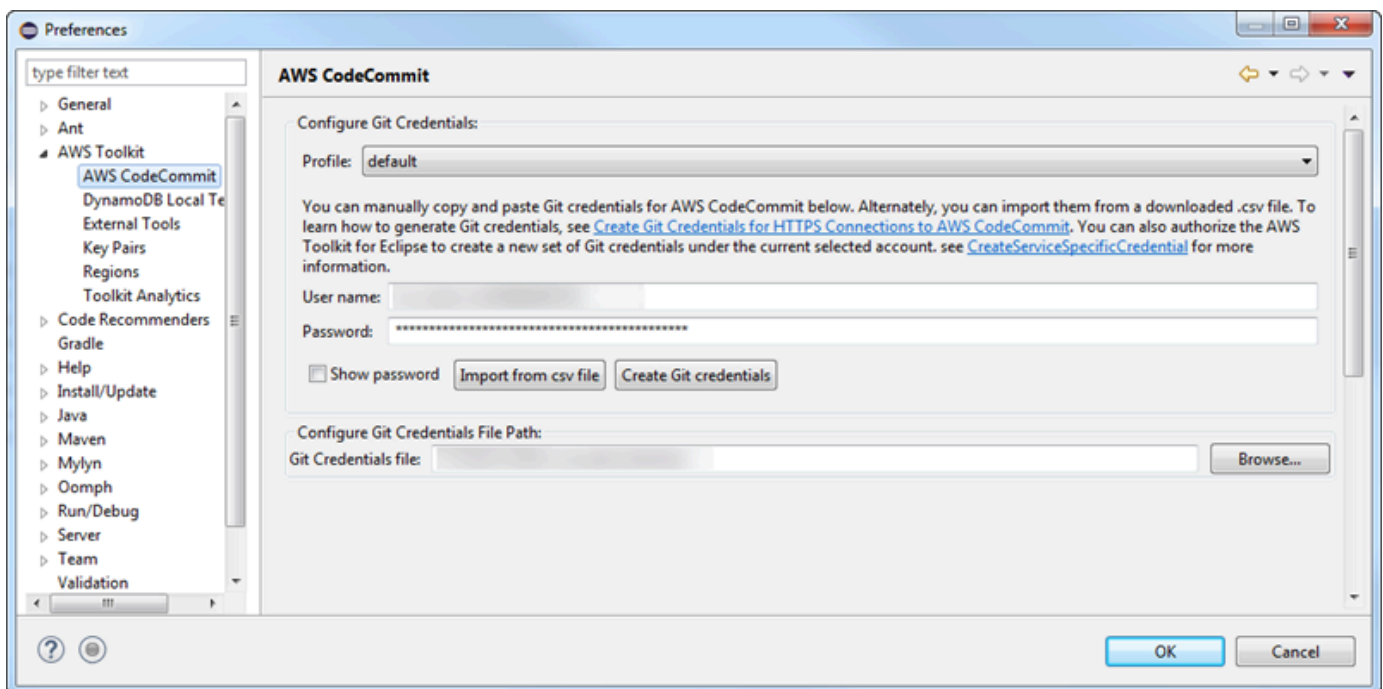
Langkah 2: Instal AWS Toolkit for Eclipse dan koneksikan ke CodeCommit

Toolkit for Eclipse adalah paket perangkat lunak yang dapat Anda tambahkan ke Eclipse. Setelah Anda menginstalnya dan mengonfigurasinya dengan profil AWS kredensi, Anda dapat terhubung ke CodeCommit dari AWS Explorer di Eclipse.

Untuk menginstal Toolkit for Eclipse dengan modul AWS CodeCommit dan mengonfigurasi akses ke repositori proyek Anda

1. Instal Toolkit for Eclipse di komputer lokal Anda jika Anda tidak memiliki versi yang didukung yang sudah diinstal. Jika Anda perlu memperbarui versi Toolkit for Eclipse, ikuti petunjuk dalam [Atur Kit Alat](#).

2. Dalam Eclipse, baik mengikuti pengalaman pengoperasian pertama, atau membuka Preferensi dari sistem menu Eclipse (lokasi bervariasi tergantung pada versi dan sistem operasi Anda) dan pilih Kit Alat AWS.
3. Lakukan salah satu dari berikut:
 - Jika Anda mengikuti pengalaman pertama, berikan kredensial keamanan AWS Anda saat diminta untuk mengatur profil kredensialnya.
 - Jika Anda mengkonfigurasinya di Preferensi dan memiliki profil kredensial yang sudah dikonfigurasi di komputer Anda, pilih dari Profil Default.
 - Jika Anda mengkonfigurasi di Preferensi dan Anda tidak melihat profil yang ingin Anda gunakan, atau jika daftar kosong, pilih Tambahkan profil. Dalam Detail Profil, masukkan nama untuk profil dan kredensial untuk pengguna IAM (access key dan secret key), atau alternatif, masukkan lokasi file kredensial.
 - Jika Anda mengkonfigurasinya di Preferensi dan Anda tidak memiliki profil yang dikonfigurasi, gunakan tautan untuk mendaftar akun atau mengelola kredensial keamanan AWS Anda yang sudah ada.
4. Dalam Eclipse, perluas Kit Alat AWS menu dan pilih AWS CodeCommit. Pilih profil kredentials Anda, dan kemudian masukkan nama pengguna dan kata sandi untuk kredensial Git Anda atau mengimpornya dari file .csv. Pilih Berlakukannya, lalu pilih OKE.



Setelah Anda masuk dengan profil, panel koneksi AWS CodeCommit muncul di Team Explorer dengan opsi untuk mengkloning, membuat, atau keluar. Memilih Klon mengklon CodeCommit repositori yang ada ke komputer lokal Anda, sehingga Anda dapat mulai mengerjakan kode. Ini adalah pilihan yang paling sering digunakan.

Jika Anda tidak memiliki repositori, atau ingin membuat repositori, pilih Buat.

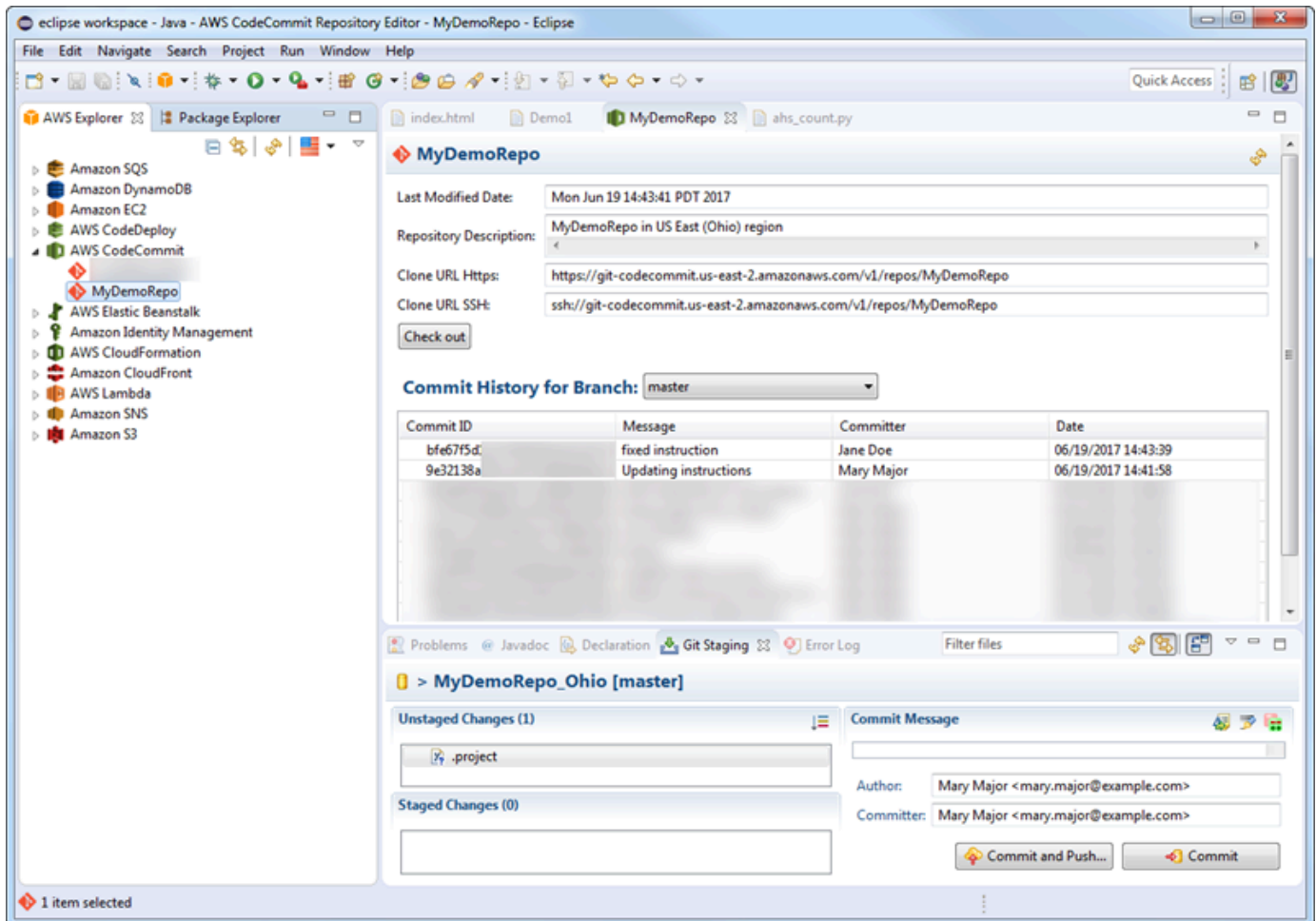
CodeCommit Kloning repositori dari Eclipse

Setelah Anda mengkonfigurasi kredensial Anda, Anda dapat mengkloning repositori ke repo lokal pada komputer Anda dengan memeriksa di Eclipse. Kemudian Anda bisa mulai bekerja dengan kode.

1. Dalam Eclipse, buka AWS Explorer. Untuk informasi tentang tempat menemukannya, lihat [Cara Mengakses AWS Explorer](#). Perluas AWS CodeCommit, dan pilih CodeCommit repositori tempat Anda ingin bekerja. Anda dapat melihat riwayat commit dan detail repositori lainnya, yang dapat membantu Anda menentukan apakah ini adalah repositori dan cabang yang ingin Anda kloning.

Note

Jika Anda tidak melihat repositori Anda, pilih ikon bendera untuk membuka menu Wilayah AWS, dan pilih Wilayah AWS dimana repositori dibuat.



2. Pilih Periksa, dan ikuti petunjuk untuk mengkloning repositori ke komputer lokal Anda.
3. Ketika Anda telah selesai mengkloning proyek, Anda siap untuk mulai mengedit kode Anda di Eclipse dan pentahapan, commit, dan pengiriman perubahan Anda ke repositori proyek Anda CodeCommit.

Buat CodeCommit repositori dari Eclipse

Anda dapat membuat CodeCommit repositori dari Eclipse dengan Toolkit for Eclipse. Sebagai bagian dari pembuatan repositori, Anda juga mengkloning ke repo lokal di komputer Anda, sehingga Anda dapat mulai bekerjanya segera.

1. Dalam AWS Explorer, klik kanan AWS CodeCommit, lalu pilih Buat repositori.

Note

Repositori adalah wilayah khusus. Sebelum Anda membuat repositori, pastikan Anda telah memilih Wilayah AWS. Anda tidak dapat memilih Wilayah AWS setelah Anda memulai proses pembuatan repositori.

2. Dalam Nama Repositori, masukkan nama untuk repositori ini. Nama repositori harus unik dalam akun Amazon Web Services. Ada karakter dan batas panjang. Untuk informasi selengkapnya, lihat [Kuota](#). Dalam Deskripsi Repositori, masukkan deskripsi opsional untuk repositori ini. Ini membantu orang lain memahami untuk apa repositori ini, dan membantu membedakannya dari repositori lain di wilayah ini. Pilih OKE.
3. Dalam AWS Explorer, perluas AWS CodeCommit, lalu pilih CodeCommit repositori yang baru saja Anda buat. Anda melihat bahwa repositori ini tidak memiliki riwayat commit. Pilih Periksa, dan ikuti petunjuk untuk mengkloning repositori ke komputer lokal Anda.

Bekerja dengan CodeCommit repositori

Setelah Anda terhubung ke CodeCommit, Anda dapat melihat daftar repositori yang terkait dengan akun Anda, dengan Wilayah AWS, di AWS Explorer. Pilih menu bendera untuk mengubah wilayah.

Note

CodeCommit mungkin tidak tersedia di semua yang Wilayah AWS didukung oleh Toolkit for Eclipse.

Dalam Toolkit for Eclipse, Anda dapat menelusuri isi repositori ini dari tampilan Navigasi dan Paket Explorer. Untuk membuka file, pilih file dari daftar.

Operasi Git di Toolkit for Eclipse untuk CodeCommit repositori bekerja persis seperti yang mereka lakukan untuk repositori berbasis Git lainnya. Anda dapat membuat perubahan pada kode, menambahkan file, dan membuat commit lokal. Ketika Anda siap untuk berbagi, Anda menggunakan Pementasan Git untuk mengirim commit Anda ke CodeCommit repositori. Jika Anda belum mengkonfigurasi informasi pembuat dan committer Anda dalam profil Git, Anda dapat melakukan ini sebelum Anda memutuskan commit dan pengiriman. Karena kredensial Git Anda untuk pengguna IAM Anda sudah disimpan secara lokal dan terkait dengan terkoneksi AWS Anda, Anda tidak akan diminta untuk memasoknya lagi ketika Anda mengirim ke CodeCommit.

Untuk informasi lebih lanjut tentang Toolkit for Eclipse, lihat [Panduan Memulai AWS Toolkit for Eclipse](#).

Setup untuk pengguna SSH yang tidak menggunakan AWS CLI

Jika Anda ingin menggunakan koneksi SSH untuk repositori Anda, Anda dapat terhubung ke AWS CodeCommit tanpa menginstal AWS CLI. AWS CLI juga mencakup perintah-perintah yang berguna ketika Anda menggunakan dan mengelola repositori CodeCommit, tetapi tidak diperlukan untuk pengaturan awal.

Topik ini mengasumsikan bahwa:

- Anda telah menyiapkan sebuah pengguna IAM dengan kebijakan atau izin yang diperlukan untuk CodeCommit dan kebijakan terkelola IAMUserSSHKeys atau izin setara yang diperlukan untuk mengunggah kunci. Untuk informasi selengkapnya, lihat [Menggunakan kebijakan berbasis identitas \(Kebijakan IAM\) untuk CodeCommit](#).
- Anda sudah memiliki, atau tahu cara membuat, sebuah pasangan kunci publik-privat. Kami sangat menyarankan Anda menggunakan kata sandi yang aman untuk kunci SSH Anda.
- Anda sudah familiar dengan SSH, klien Git Anda, dan file konfigurasinya.
- Jika Anda menggunakan Windows, Anda telah menginstal utilitas baris perintah, seperti Git Bash, yang mengemulasi shell bash.

Jika Anda memerlukan panduan lebih lanjut, ikuti petunjuk yang ada di [Untuk koneksi SSH di Linux, macOS, atau Unix](#) atau [Untuk koneksi SSH pada Windows](#).

Topik

- [Langkah 1: Kaitkan kunci publik Anda dengan pengguna IAM](#)
- [Langkah 2: Menambahkan CodeCommit ke konfigurasi SSH Anda](#)
- [Langkah selanjutnya](#)

Langkah 1: Kaitkan kunci publik Anda dengan pengguna IAM

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Di konsol IAM, pada panel navigasi, pilih Pengguna, dan dari daftar pengguna, pilih pengguna IAM Anda.

Tip

Jika Anda memiliki lebih dari satu konfigurasi SSH, pastikan Anda juga menyertakan baris kosong sebelum dan sesudah konten. Simpan file dengan menekan kunci `Ctrl` dan `d` secara bersamaan.

2. Jalankan perintah berikut ini untuk menguji konfigurasi SSH Anda:

```
ssh git-codecommit.us-east-2.amazonaws.com
```

Masukkan frasa kata sandi untuk file kunci SSH Anda saat diminta. Jika semuanya dikonfigurasi dengan benar, maka Anda akan melihat pesan sukses berikut:

```
You have successfully authenticated over SSH. You can use Git to interact with CodeCommit.
```

Langkah selanjutnya

Anda telah menyelesaikan prasyaratnya. Ikuti langkah-langkah yang ada di [Memulai dengan CodeCommit](#) untuk mulai menggunakan CodeCommit.

Untuk terhubung ke repositori, ikuti langkah-langkah dalam [Connect ke sebuah repositori](#). Untuk membuat sebuah repositori, ikuti langkah-langkah yang ada di [Buatlah sebuah repositori](#).

Langkah-langkah pengaturan untuk koneksi SSH ke repositori AWS CodeCommit di Linux, macOS, atau Unix

Sebelum Anda dapat terhubung ke CodeCommit Untuk pertama kalinya, Anda harus menyelesaikan beberapa langkah konfigurasi awal. Setelah mengatur komputer dan AWS profil Anda, Anda dapat terhubung ke CodeCommit repositori dan mengkloning repositori tersebut ke komputer Anda (juga dikenal sebagai membuat repo lokal). Jika Anda baru mengenal Git, Anda mungkin juga ingin meninjau informasi di [Di mana saya dapat mempelajari lebih lanjut tentang Git?](#)

Topik

- [Langkah 1: Konfigurasi awal untuk CodeCommit](#)

- [Langkah 2: Instal Git](#)
- [Langkah 3: Konfigurasi kredensi di Linux, macOS, atau Unix](#)
- [Langkah 4: Connect keCodeCommit konsol dan kloning repositori](#)
- [Langkah selanjutnya](#)

Langkah 1: Konfigurasi awal untukCodeCommit

Ikuti langkah-langkah untuk mengatur akun Amazon Web Services, membuat pengguna IAM, dan mengkonfigurasi akses keCodeCommit.

Untuk membuat dan mengkonfigurasi pengguna IAM untuk mengaksesCodeCommit

1. Buat akun Amazon Web Services dengan membuka <http://aws.amazon.com> dan memilih Mendaftar.
2. Membuat pengguna IAM, atau menggunakan yang sudah ada, di akun Amazon Web Services Anda. Pastikan Anda memiliki access key ID dan secret access key yang terkait dengan pengguna IAM. Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di akun Amazon Web Services Anda](#).

Note

CodeCommitmembutuhkanAWS Key Management Service. Jika Anda menggunakan pengguna IAM yang ada, pastikan tidak ada kebijakan yang melekat pada pengguna yang secara tegas menolakAWS KMS tindakan yang diperlukan olehCodeCommit. Untuk informasi selengkapnya, lihat [AWS KMS dan enkripsi](#).

3. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
4. Di konsol IAM di panel navigasi, pilih Pengguna, dan kemudian pilih pengguna IAM yang ingin Anda konfigurasi untukCodeCommit akses.
5. Pada tab Izin, pilih Tambahkan izin.
6. Dalam Berikan izin, pilih Lampirkan kebijakan yang ada secara langsung.
7. Dari daftar kebijakan, pilih AWSCodeCommitPowerUserkebijakan terkelola lainnya untukCodeCommit akses. Untuk informasi selengkapnya, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Setelah memilih kebijakan yang ingin dilampirkan, pilih Berikutnya: Tinjau untuk meninjau daftar kebijakan untuk dilampirkan ke pengguna IAM. Jika daftar sudah benar, pilih Tambahkan izin.

Untuk informasi lebih lanjut tentang kebijakanCodeCommit terkelola dan berbagi akses ke repositori dengan grup dan pengguna lain, lihat[Bagikan sebuah repositori](#) dan[Kontrol autentikasi dan akses untuk AWS CodeCommit](#).

Note

Jika Anda ingin menggunakanAWS CLI perintah denganCodeCommit, instalAWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Langkah 2: Instal Git

Untuk bekerja dengan file, commit, dan informasi lainnya diCodeCommit repositori, Anda harus menginstal Git di mesin lokal Anda. CodeCommitmendukung versi Git 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git.

Untuk menginstal Git, kami merekomendasikan situs web seperti [Pengunduhan Git](#).

Note

Git adalah platform yang berkembang dan diperbarui secara teratur. Kadang-kadang, perubahan fitur dapat mempengaruhi cara kerjanyaCodeCommit. Jika Anda mengalami masalah dengan versi Git tertentuCodeCommit, tinjau informasi di[Memecahkan masalah](#).

Langkah 3: Konfigurasikan kredensi di Linux, macOS, atau Unix

SSH dan Linux, macOS, atau Unix: Siapkan kunci publik dan privat untuk Git danCodeCommit

Untuk mengatur kunci publik dan privat untuk Git danCodeCommit

1. Dari terminal pada mesin lokal Anda, jalankan perintah ssh-keygen, dan ikuti petunjuk untuk menyimpan file ke direktori .ssh untuk profil Anda.

Note

Pastikan untuk memeriksa dengan administrator sistem Anda tentang di mana file kunci harus disimpan dan pola penamaan file yang harus digunakan.

Misalnya:

```
$ ssh-keygen

Generating public/private rsa key pair.
Enter file in which to save the key (/home/user-name/.ssh/id_rsa): Type /home/your-user-name/.ssh/ and a file name here, for example /home/your-user-name/.ssh/codecommit_rsa

Enter passphrase (empty for no passphrase): <Type a passphrase, and then press Enter>
Enter same passphrase again: <Type the passphrase again, and then press Enter>

Your identification has been saved in /home/user-name/.ssh/codecommit_rsa.
Your public key has been saved in /home/user-name/.ssh/codecommit_rsa.pub.
The key fingerprint is:
45:63:d5:99:0e:99:73:50:5e:d4:b3:2d:86:4a:2c:14 user-name@client-name
The key's randomart image is:
+--[ RSA 2048]-----+
|      E.+o*.++|
|      .o .=.o.|
|      . .. *. +|
|      ..o . +..|
|      So . . . |
|      .        |
|              |
|              |
|              |
+-----+
```

Ini menghasilkan:

- File *codecommit_rsa*, yang merupakan file kunci privat.
- File *.pub codecommit_rsa*, yang merupakan file kunci publik.

i Tip

Secara default, ssh-keygen menghasilkan kunci 2048 bit. Anda dapat menggunakan parameter `-t` dan `-b` untuk menentukan jenis dan panjang kunci. Jika Anda ingin kunci 4096 bit dalam format `rsa`, Anda akan menentukan ini dengan menjalankan perintah dengan parameter berikut:

```
ssh-keygen -t rsa -b 4096
```

Untuk informasi lebih lanjut tentang format dan panjang yang diperlukan untuk kunci SSH, lihat [Menggunakan IAM dengan CodeCommit](#).

2. Jalankan perintah berikut untuk menampilkan nilai file kunci publik (`codecommit_rsa.pub`):

```
cat ~/.ssh/codecommit_rsa.pub
```

Salin nilai ini. Itu terlihat serupa dengan yang berikut ini:

```
ssh-rsa EXAMPLE-AfICCQD6m7oRw0uX0jANBqkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBA5TC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXRN0Q21sYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcNMTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAGTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDAS=EXAMPLE user-name@ip-192-0-2-137
```

3. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.

i Note

Anda dapat menampilkan dan mengelola CodeCommit kredensi Anda di Kredensi Keamanan Saya. Untuk informasi selengkapnya, lihat [Melihat dan mengelola kredensial Anda](#).

4. Di konsol IAM, pada panel navigasi, pilih Pengguna, dan dari daftar pengguna, pilih pengguna IAM Anda.
5. Pada halaman detail pengguna, pilih tab Kredensial Keamanan, kemudian pilih Unggah kunci publik SSH.

- Tempelkan isi kunci publik SSH Anda ke dalam bidang, dan kemudian pilih Unggah kunci publik SSH.
- Menyalin atau menyimpan informasi di ID Kunci SSH (misalnya, *APKAEIBAERJR2EXAMPLE*).



Note

Jika Anda memiliki lebih dari satu ID kunci SSH yang diunggah, kunci tercantum menurut abjad berdasarkan ID kunci, bukan tanggal pengunggahan. Pastikan bahwa Anda telah menyalin ID kunci yang terkait dengan tanggal pengunggahan yang benar.

- Pada mesin lokal Anda, gunakan editor teks untuk membuat file konfigurasi di direktori `~/.ssh`, dan kemudian tambahkan baris berikut ke file, di mana nilai untuk *Pengguna* adalah ID kunci SSH yang Anda salin sebelumnya:

```
Host git-codecommit.*.amazonaws.com
  User APKAEIBAERJR2EXAMPLE
  IdentityFile ~/.ssh/codecommit_rsa
```

Note

Jika Anda memberi nama file kunci privat Anda selain *codecommit_rsa*, pastikan untuk menggunakannya di sini.

Anda dapat mengatur akses SSH ke repositori di beberapa akun Amazon Web Services, Untuk informasi lebih lanjut, lihat [Pemecahan masalah koneksi SSH ke AWS CodeCommit](#).

Simpan dan beri nama file config ini.

- Dari terminal, jalankan perintah berikut untuk mengubah izin file konfigurasi:

```
chmod 600 config
```

10. Jalankan perintah berikut ini untuk menguji konfigurasi SSH Anda:

```
ssh git-codecommit.us-east-2.amazonaws.com
```

Anda diminta untuk mengonfirmasi koneksi karena `git-codecommit.us-east-2.amazonaws.com` belum termasuk dalam file host Anda yang dikenal. Sidik jari CodeCommit server ditampilkan sebagai bagian dari verifikasi (`a9:6d:03:ed:08:42:21:be:06:e1:e0:2a:d1:75:31:5e` untuk MD5 atau `31B1W2g5xn/NA2Ck6dyeJIrQ0Wvn7n8UES56fG6ZIZQ` SHA256).

Note

CodeCommit sidik jari server unik untuk setiap Wilayah AWS. Untuk melihat sidik jari server untuk Wilayah AWS, lihat [Sidik jari server untuk CodeCommit](#).

Setelah Anda mengkonfirmasi koneksi, Anda akan melihat konfirmasi bahwa Anda telah menambahkan server ke file host yang diketahui dan pesan koneksi yang berhasil. Jika Anda tidak melihat pesan sukses, periksa bahwa Anda menyimpan di direktori `~/.ssh` dari pengguna IAM yang Anda konfigurasi untuk akses ke CodeCommit, dan bahwa Anda menentukan file kunci privat yang benar. `config`

Untuk informasi untuk membantu Anda memecahkan masalah, jalankan perintah `ssh` dengan parameter `-v`. Misalnya:

```
ssh -v git-codecommit.us-east-2.amazonaws.com
```

Untuk informasi untuk membantu Anda memecahkan masalah koneksi, lihat [Pemecahan masalah koneksi SSH ke AWS CodeCommit](#).

Langkah 4: Connect ke CodeCommit konsol dan kloning repositori

Jika administrator telah mengirimkan nama dan rincian koneksi untuk CodeCommit repositori, Anda dapat melewati langkah ini dan mengkloning repositori secara langsung.

Untuk menghubungkan ke CodeCommit repositori

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.

2. Dalam pemilih wilayah, pilih Wilayah AWS di mana repositori dibuat. Repositori adalah khusus untuk sebuah Wilayah AWS. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Temukan repositori yang ingin Anda sambungkan dari daftar dan pilih. Pilih URL Kloning, lalu pilih protokol yang ingin Anda gunakan saat mengkloning atau menghubungkan ke repositori. Langkah ini akan menyalin URL klon tersebut.
 - Salin URL HTTPS jika Anda menggunakan kredensial Git dengan pengguna IAM Anda atau pembantu kredensial yang disertakan dengan AWS CLI.
 - Salin URL HTTPS (GRC) jika Anda menggunakan perintah `git-remote-codecommit` di komputer lokal Anda.
 - Salin URL SSH jika Anda menggunakan pasangan kunci publik/swasta SSH dengan pengguna IAM Anda.

Note

Jika Anda melihat halaman Selamat Datang bukan daftar repositori, tidak ada repositori yang terkait dengan akun AWS Anda di Wilayah AWS tempat Anda masuk. Untuk membuat repositori, lihat [the section called “Buatlah sebuah repositori”](#) atau ikuti langkah-langkah di tutorial [Memulai dengan Git dan CodeCommit](#).

4. Buka terminal. Dari direktori `/tmp`, jalankan `git clone` dengan URL SSH yang Anda salin untuk mengkloning repositori. Misalnya, untuk mengkloning repositori bernama *MyDemoRepo* ke repo lokal bernama *my-demo-repo* di Wilayah US East (Ohio):

```
git clone ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Note

Jika Anda berhasil menguji koneksi Anda, tetapi perintah klon gagal, Anda mungkin tidak memiliki akses yang diperlukan ke file konfigurasi Anda, atau pengaturan lain mungkin bertentangan dengan file konfigurasi Anda. Coba hubungkan lagi, kali ini termasuk ID kunci SSH dalam perintah. Misalnya:

```
git clone ssh://Your-SSH-Key-ID@git-codecommit.us-east-2.amazonaws.com/v1/
repos/MyDemoRepo my-demo-repo
```

Untuk informasi selengkapnya, lihat [Kesalahan akses: Kunci publik berhasil diunggah ke IAM namun koneksi gagal pada sistem Linux, macOS, atau Unix](#).

Untuk informasi selengkapnya tentang cara menghubungkan ke repositori, lihat [Connect ke CodeCommit repositori dengan mengkloning repositori](#).

Langkah selanjutnya

Anda telah menyelesaikan prasyaratnya. Ikuti langkah-langkah [Memulai dengan CodeCommit](#) untuk mulai menggunakan CodeCommit.

Langkah-langkah pengaturan untuk koneksi SSH ke repositori AWS CodeCommit di Windows

Sebelum Anda dapat terhubung ke AWS CodeCommit Untuk pertama kalinya, Anda harus menyelesaikan beberapa langkah konfigurasi awal. Setelah mengatur komputer dan profil AWS Anda, Anda dapat terhubung ke repositori CodeCommit dan mengkloning repositori tersebut ke komputer Anda (juga dikenal sebagai membuat repo lokal). Jika Anda baru mengenal Git, Anda mungkin juga ingin meninjau informasi di [Di mana saya dapat mempelajari lebih lanjut tentang Git?](#).

Topik

- [Langkah 1: Konfigurasi awal untuk CodeCommit](#)
- [Langkah 2: Instal Git](#)
- [Langkah 3: Siapkan kunci publik dan privat untuk Git dan CodeCommit](#)
- [Langkah 4: Connect ke konsol CodeCommit dan kloning repositori](#)
- [Langkah selanjutnya](#)

Langkah 1: Konfigurasi awal untuk CodeCommit

Ikuti langkah-langkah ini untuk menyiapkan akun Amazon Web Services, membuat pengguna IAM, dan mengkonfigurasi akses ke CodeCommit.

Untuk membuat dan mengkonfigurasi pengguna IAM untuk mengakses CodeCommit

1. Buat akun Amazon Web Services dengan membuka <http://aws.amazon.com> dan memilih Mendaftar.
2. Membuat pengguna IAM, atau menggunakan yang sudah ada, di akun Amazon Web Services Anda. Pastikan Anda memiliki access key ID dan secret access key yang terkait dengan pengguna IAM. Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di akun Amazon Web Services Anda](#).

Note

CodeCommit membutuhkan AWS Key Management Service. Jika Anda menggunakan pengguna IAM yang ada, pastikan tidak ada kebijakan yang melekat pada pengguna yang secara tegas menolak tindakan AWS KMS yang diperlukan oleh CodeCommit. Untuk informasi selengkapnya, lihat [AWS KMS dan enkripsi](#).

3. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
4. Pada konsol IAM di panel navigasi, pilih Pengguna, dan kemudian pilih pengguna IAM yang ingin Anda konfigurasi untuk akses CodeCommit.
5. Pada tab Izin, pilih Tambahkan izin.
6. Dalam Berikan izin, pilih Lampirkan kebijakan yang ada secara langsung.
7. Dari daftar kebijakan, pilih AWSCodeCommitPowerUser atau kebijakan terkelola lainnya untuk akses CodeCommit. Untuk informasi selengkapnya, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Setelah memilih kebijakan yang ingin dilampirkan, pilihSelanjutnya: Tinjauuntuk meninjau daftar kebijakan untuk dilampirkan ke pengguna IAM. Jika daftar sudah benar, pilih Tambahkan izin.

Untuk informasi lebih lanjut tentang kebijakan terkelola CodeCommit dan berbagi akses ke repositori dengan grup dan pengguna lain, lihat [Bagikan sebuah repositori](#) dan [Kontrol autentikasi dan akses untuk AWS CodeCommit](#).

Note

Jika Anda ingin menggunakan perintah AWS CLI dengan CodeCommit, instal AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Langkah 2: Instal Git

Untuk bekerja dengan file, commit, dan informasi lainnya di repositori CodeCommit, Anda harus menginstal Git pada mesin lokal Anda. CodeCommit support versi Git 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git.

Untuk menginstal Git, kami merekomendasikan situs web seperti [Pengunduhan Git](#).

Note

Git adalah platform yang berkembang dan diperbarui secara teratur. Kadang-kadang, perubahan fitur dapat mempengaruhi cara kerjanya dengan CodeCommit. Jika Anda mengalami masalah dengan versi Git dan CodeCommit tertentu, tinjau informasi di [Memecahkan masalah](#).

Jika versi Git yang Anda pasang tidak termasuk emulator Bash, seperti Git Bash, pasang satu. Anda menggunakan emulator ini bukan baris perintah Windows ketika Anda mengkonfigurasi koneksi SSH.

Langkah 3: Siapkan kunci publik dan privat untuk Git dan CodeCommit

Untuk mengatur kunci publik dan privat untuk Git dan CodeCommit di Windows

1. Buka emulator Bash.

Note

Anda mungkin perlu menjalankan emulator dengan izin administratif.

Dari emulator, jalankan perintah `ssh-keygen`, dan ikuti petunjuk untuk menyimpan file ke direktori `.ssh` untuk profil Anda.

Misalnya:

```
$ ssh-keygen

Generating public/private rsa key pair.
Enter file in which to save the key (/drive/Users/user-name/.ssh/id_rsa): Type a
file name here, for example /c/Users/user-name/.ssh/codecommit_rsa

Enter passphrase (empty for no passphrase): <Type a passphrase, and then press
Enter>
Enter same passphrase again: <Type the passphrase again, and then press Enter>

Your identification has been saved in drive/Users/user-name/.ssh/codecommit_rsa.
Your public key has been saved in drive/Users/user-name/.ssh/codecommit_rsa.pub.
The key fingerprint is:
45:63:d5:99:0e:99:73:50:5e:d4:b3:2d:86:4a:2c:14 user-name@client-name
The key's randomart image is:
+--[ RSA 2048]-----+
|           E.+o*.++|
|          .o .=.o.|
|         . .. *. +|
|         ..o . +..|
|          So . . . |
|           .       |
|                   |
|                   |
|                   |
+-----+

```

Ini menghasilkan:

- File *codecommit_rsa*, yang merupakan file kunci privat.
- File *.pub codecommit_rsa*, yang merupakan file kunci publik.

Tip

Secara default, ssh-keygen menghasilkan kunci 2048 bit. Anda dapat menggunakan parameter `-t` dan `-b` untuk menentukan jenis dan panjang kunci. Jika Anda ingin kunci 4096 bit dalam format `rsa`, Anda akan menentukan ini dengan menjalankan perintah dengan parameter berikut:


```
ssh-keygen -t rsa -b 4096
```

Untuk informasi lebih lanjut tentang format dan panjang yang diperlukan untuk kunci SSH, lihat [Menggunakan IAM dengan CodeCommit](#).

2. Jalankan perintah berikut untuk menampilkan nilai file kunci publik (`codecommit_rsa.pub`):

```
cd .ssh  
notepad codecommit_rsa.pub
```

Salin konten file, dan kemudian tutup Notepad tanpa menyimpan. Isi file terlihat serupa dengan berikut ini:

```
ssh-rsa EXAMPLE-AfICCQD6m7oRw0uX0jANBgkqhkiG9w0BAQUFADCBiDELMakGA1UEBhMCMVVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC01BTSBDb25zb2x1MRIwEAYDVQQDEw1UZXRhbnQ21sYWMxHzAdBgkqhkiG9w0BCQEWEG5vb251QGFTYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcNMTEwNDI1MjA0NTIxWjCBiDELMakGA1UEBhMCMVVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGx1MQ8wDQYDVQQKEwZBbWF6b24xFDAS=EXAMPLE user-name@computer-name
```

3. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.

Note

Anda dapat menampilkan dan mengelola kredensial CodeCommit Anda di Kredensial Keamanan Saya. Untuk informasi selengkapnya, lihat [Melihat dan mengelola kredensial Anda](#).

4. Di konsol IAM, pada panel navigasi, pilih Pengguna, dan dari daftar pengguna, pilih pengguna IAM Anda.
5. Pada halaman detail pengguna, pilih tab Kredensial Keamanan, kemudian pilih Unggah kunci publik SSH.
6. Tempelkan isi kunci publik SSH Anda ke dalam bidang, dan kemudian pilih Unggah kunci publik SSH.
7. Menyalin atau menyimpan informasi di ID Kunci SSH (misalnya, `APKAEIBAERJR2EXAMPLE`).

SSH keys for AWS CodeCommit

Use SSH public keys to authenticate to AWS CodeCommit repositories. [Learn more about SSH keys.](#)

Upload SSH public key

SSH Key ID	Uploaded	Status	Actions
APKAEIBAERJR2EXAMPLE	2015-07-21 16:32 PDT	Active	Make Inactive Show SSH Key Delete

Note

Jika Anda memiliki lebih dari satu ID kunci SSH yang diunggah, kunci tercantum menurut abjad berdasarkan ID kunci, bukan tanggal pengunggahan. Pastikan bahwa Anda telah menyalin ID kunci yang terkait dengan tanggal pengunggahan yang benar.

8. Dalam emulator Bash, jalankan perintah berikut untuk membuat file konfigurasi di direktori `~/.ssh`, atau edit jika salah satu sudah ada:

```
notepad ~/.ssh/config
```

9. Tambahkan baris berikut ke file, di mana nilai untuk *Pengguna* adalah ID kunci SSH yang Anda salin sebelumnya, dan nilai untuk *IdentityFile* adalah path ke dan nama file kunci privat:

```
Host git-codecommit.*.amazonaws.com
  User APKAEIBAERJR2EXAMPLE
  IdentityFile ~/.ssh/codecommit_rsa
```

Note

Jika Anda memberi nama file kunci privat Anda selain *codecommit_rsa*, pastikan untuk menggunakannya di sini.

Anda dapat mengatur akses SSH ke repositori di beberapa akun Amazon Web Services, Untuk informasi lebih lanjut, lihat [Pemecahan masalah koneksi SSH ke AWS CodeCommit](#).

Simpan file sebagai config (bukan config.txt), dan kemudian tutup Notepad.

⚠ Important

Nama file harus `config` tanpa ekstensi file. Jika tidak, koneksi SSH gagal.

10. Jalankan perintah berikut ini untuk menguji konfigurasi SSH Anda:

```
ssh git-codecommit.us-east-2.amazonaws.com
```

Anda diminta untuk mengonfirmasi koneksi karena `git-codecommit.us-east-2.amazonaws.com` belum termasuk dalam file host Anda yang dikenal. Sidik jari server CodeCommit ditampilkan sebagai bagian dari verifikasi (a9:6d:03:ed:08:42:21:be:06:e1:e0:2a:d1:75:31:5e untuk MD5 atau 31B1W2g5xn/NA2Ck6dyeJIrQ0Wvn7n8UES56fG6ZIZQ untuk SHA256).

i Note

Sidik jari server CodeCommit unik untuk setiap Wilayah AWS. Untuk melihat sidik jari server untuk Wilayah AWS, lihat [Sidik jari server untuk CodeCommit](#).

Setelah Anda mengkonfirmasi koneksi, Anda akan melihat konfirmasi bahwa Anda telah menambahkan server ke file host yang diketahui dan pesan koneksi yang berhasil. Jika Anda tidak melihat pesan sukses, periksa kembali apakah Anda menyimpan file `config` di direktori `~/.ssh` pengguna IAM yang Anda konfigurasi untuk akses ke CodeCommit, bahwa file `config` tidak memiliki ekstensi file (misalnya, itu tidak harus bernama `config.txt`), dan bahwa Anda menentukan file kunci privat yang benar (`codecommit_rsa`, bukan `codecommit_rsa.pub`).

Untuk informasi untuk membantu Anda memecahkan masalah, jalankan perintah `ssh` dengan parameter `-v`. Misalnya:

```
ssh -v git-codecommit.us-east-2.amazonaws.com
```

Untuk informasi untuk membantu Anda memecahkan masalah koneksi, lihat [Pemecahan masalah koneksi SSH ke AWS CodeCommit](#).

Langkah 4: Connect ke konsol CodeCommit dan kloning repositori

Jika administrator telah mengirimkan nama dan rincian koneksi untuk repositori CodeCommit, Anda dapat melewati langkah ini dan mengkloning repositori secara langsung.

Untuk menghubungkan ke sebuah repositori CodeCommit

1. Buka konsol CodeCommit di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Dalam pemilih wilayah, pilih Wilayah AWS di mana repositori dibuat. Repositori adalah khusus untuk sebuah Wilayah AWS. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Temukan repositori yang ingin Anda sambungkan dari daftar dan pilih. Pilih URL Kloning, lalu pilih protokol yang ingin Anda gunakan saat mengkloning atau menghubungkan ke repositori. Langkah ini akan menyalin URL klon tersebut.
 - Salin URL HTTPS jika Anda menggunakan kredensial Git dengan pengguna IAM Anda atau pembantu kredensial yang disertakan dengan AWS CLI.
 - Salin URL HTTPS (GRC) jika Anda menggunakan perintah `git-remote-codecommit` di komputer lokal Anda.
 - Salin URL SSH jika Anda menggunakan pasangan kunci publik/swasta SSH dengan pengguna IAM Anda.

Note

Jika Anda melihat halaman Selamat Datang bukan daftar repositori, tidak ada repositori yang terkait dengan akun AWS Anda di Wilayah AWS tempat Anda masuk. Untuk membuat repositori, lihat [the section called “Buatlah sebuah repositori”](#) atau ikuti langkah-langkah di tutorial [Memulai dengan Git dan CodeCommit](#).

4. Dalam Bash emulator, jalankan `git clone` dengan URL SSH yang Anda salin untuk mengkloning repositori. Perintah ini membuat repo lokal di subdirektori direktori di mana Anda menjalankan perintah. Misalnya, untuk mengkloning repositori bernama *MyDemoRepo* ke repo lokal bernama *my-demo-repo* di Wilayah US East (Ohio):

```
git clone ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Atau, buka prompt perintah, dan gunakan URL dan ID kunci SSH untuk kunci publik yang Anda unggah ke IAM, jalankan perintah git clone. Repo lokal dibuat di subdirektori direktori di mana Anda menjalankan perintah. Misalnya, untuk mengkloning repositori bernama *MyDemoRepo* ke repo lokal bernama *my-demo-repo*:

```
git clone ssh://Your-SSH-Key-ID@git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Untuk informasi selengkapnya, lihat [Connect ke CodeCommit repositori dengan mengkloning repositori](#) dan [Buatlah sebuah commit](#).

Langkah selanjutnya

Anda telah menyelesaikan prasyaratnya. Ikuti langkah-langkah yang ada di [Memulai dengan CodeCommit](#) untuk mulai menggunakan CodeCommit.

Langkah-langkah pengaturan untuk koneksi HTTPS ke AWS CodeCommit di Linux, macOS, atau Unix dengan pembantu kredensial AWS CLI

Sebelum Anda dapat terhubung ke AWS CodeCommit Untuk pertama kalinya, Anda harus menyelesaikan langkah konfigurasi awal. Bagi sebagian besar pengguna, ini bisa dilakukan dengan mudah dengan mengikuti langkah-langkah di [Untuk pengguna HTTPS menggunakan kredensial Git](#). Namun, jika Anda ingin terhubung keCodeCommitmenggunakan akun root, akses federasi, atau kredensi sementara, Anda dapat menggunakan pembantu kredenal yang disertakan dalamAWS CLI.

Note

Meskipun pembantu kredenal adalah metode yang didukung untuk menghubungkan keCodeCommitmenggunakan akses federasi, penyedia identitas, atau kredensi sementara, metode yang disarankan adalah menginstal dan menggunakangit-remote-codecommitutilitas. Untuk informasi selengkapnya, lihat [Langkah-langkah penyiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit](#).

Topik

- [Langkah 1: Konfigurasi awal untukCodeCommit](#)
- [Langkah 2: Instal Git](#)
- [Langkah 3: Mengatur pembantu kredensial](#)
- [Langkah 4: Hubungkan keCodeCommitkonsol dan kloning repositori](#)
- [Langkah selanjutnya](#)

Langkah 1: Konfigurasi awal untukCodeCommit

Ikuti langkah-langkah untuk mengatur akun Amazon Web Services, membuat dan mengkonfigurasi IAM pengguna, dan menginstal AWS CLI.

Untuk membuat dan mengkonfigurasi pengguna IAM untuk mengaksesCodeCommit

1. Buat akun Amazon Web Services dengan membuka <http://aws.amazon.com> dan memilih Mendaftar.
2. Membuat pengguna IAM, atau menggunakan yang sudah ada, di akun Amazon Web Services Anda. Pastikan Anda memiliki access key ID dan secret access key yang terkait dengan pengguna IAM. Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di akun Amazon Web Services Anda](#).

Note

CodeCommitmemerlukanAWS Key Management Service. Jika Anda menggunakan pengguna IAM yang sudah ada, pastikan tidak ada kebijakan yang dilampirkan pada pengguna yang secara tegas menolakAWS KMStindakan yang diperlukan olehCodeCommit. Untuk informasi selengkapnya, lihat [AWS KMS dan enkripsi](#).

3. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
4. Di konsol IAM, di panel navigasi, pilihPengguna, lalu pilih pengguna IAM yang ingin Anda konfigurasiCodeCommitakses.
5. Pada tab Izin, pilih Tambahkan izin.
6. Dalam Berikan izin, pilih Lampirkan kebijakan yang ada secara langsung.
7. Dari daftar kebijakan, pilihAWSCodeCommitPowerUseratau kebijakan terkelola lainnya untukCodeCommitakses. Untuk informasi selengkapnya, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).


Setelah memilih kebijakan yang ingin dilampirkan, pilih Berikutnya: Tinjau untuk meninjau daftar kebijakan untuk dilampirkan ke pengguna IAM. Jika daftar sudah benar, pilih Tambahkan izin.

Untuk informasi lebih lanjut tentang CodeCommit kebijakan terkelola dan berbagi akses ke repositori dengan grup dan pengguna lain, lihat [Bagikan sebuah repositori](#) dan [Kontrol autentikasi dan akses untuk AWS CodeCommit](#).

Untuk menginstal dan mengonfigurasi AWS CLI

1. Pada komputer lokal Anda, unduh dan instal AWS CLI. Ini adalah prasyarat untuk berinteraksi dengan CodeCommit dari baris perintah. Kami merekomendasikan agar Anda menginstal AWS CLI versi 2. Ini adalah versi utama terbaru dari AWS CLI dan mendukung semua fitur terbaru. Ini adalah satu-satunya versi AWS CLI yang mendukung penggunaan akun root, akses federasi, atau kredensial sementara dengan git-remote-codecommit.

Untuk informasi selengkapnya, lihat [Menyiapkan Antarmuka Baris Perintah AWS](#).

 Note

CodeCommit bekerja hanya dengan AWS CLI versi 1.7.38 dan yang lebih baru. Sebagai praktik terbaik, instal atau tingkatkan AWS CLI untuk versi terbaru yang tersedia. Untuk menentukan versi AWS CLI yang telah Anda instal, jalankan perintah `aws --version`. Untuk meng-upgrade versi lama AWS CLI ke versi terbaru, lihat [Menginstal AWS Command Line Interface](#).

2. Jalankan perintah ini untuk memverifikasi bahwa CodeCommit perintah untuk AWS CLI diinstal.

```
aws codecommit help
```

Perintah ini mengembalikan daftar CodeCommit perintah.

3. Konfigurasi AWS CLI dengan profil dengan menggunakan perintah `configure`, sebagai berikut:

```
aws configure
```

Saat diminta, tentukan AWS kunci akses dan AWS kunci akses rahasia pengguna IAM untuk digunakan CodeCommit. Juga, pastikan untuk menentukan Wilayah AWS di mana repositori ada,

seperti `us-east-2`. Saat diminta untuk format output default, tentukan `json`. Misalnya, jika Anda mengonfigurasi profil untuk pengguna IAM:

```
AWS Access Key ID [None]: Type your IAM user AWS access key ID here, and then press Enter
AWS Secret Access Key [None]: Type your IAM user AWS secret access key here, and then press Enter
Default region name [None]: Type a supported region for CodeCommit here, and then press Enter
Default output format [None]: Type json here, and then press Enter
```

Untuk informasi selengkapnya tentang membuat dan mengonfigurasi profil untuk digunakan dengan AWS CLI, lihat berikut ini:

- [Profil Bernama](#)
- [Menggunakan Peran IAM dalam AWS CLI](#)
- [Tetapkan perintah](#)
- [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#)

Untuk terkoneksi ke repositori atau sumber daya di Wilayah AWS lain, Anda harus mengkonfigurasi ulang AWS CLI dengan nama wilayah default. Nama Wilayah default yang didukung untuk CodeCommit termasuk:

- `us-east-2`
- `us-east-1`
- `eu-west-1`
- `us-west-2`
- `ap-northeast-1`
- `ap-southeast-1`
- `ap-southeast-2`
- `ap-southeast-3`
- `me-central-1`
- `eu-central-1`
- `ap-northeast-2`

- `sa-east-1`

- us-west-1
- eu-west-2
- ap-south-1
- ap-south-1
- ca-central-1
- us-gov-west-1
- us-gov-east-1
- eu-north-1
- ap-east-1
- me-south-1
- cn-north-1
- cn-northwest-1
- eu-south-1
- ap-northeast-3
- af-south-1
- Il-sentral-1

Untuk informasi selengkapnya tentang CodeCommit dan Wilayah AWS, lihat [Wilayah dan titik akhir koneksi Git](#). Untuk informasi selengkapnya tentang IAM, access key, dan kunci rahasia, lihat [Bagaimana Saya Mendapatkan Kredensial?](#) dan [Mengelola Access Key untuk Pengguna IAM](#). Untuk informasi selengkapnya tentang AWS CLI dan profil, lihat [Profil Bernama](#).

Langkah 2: Instal Git

Untuk bekerja dengan file, komit, dan informasi lainnya di CodeCommit repositori, Anda harus menginstal Git pada mesin lokal Anda. CodeCommit mendukung Git versi 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git.

Untuk menginstal Git, kami merekomendasikan situs web seperti [Pengunduhan Git](#).

Note

Git adalah platform yang berkembang dan diperbarui secara teratur. Terkadang, perubahan fitur dapat memengaruhi cara kerjanya CodeCommit. Jika Anda mengalami masalah dengan versi tertentu dari Git dan CodeCommit, tinjau informasi di [Memecahkan masalah](#).

Langkah 3: Mengatur pembantu kredensial

1. Dari terminal, gunakan Git untuk menjalankan git config, menentukan penggunaan pembantu kredensial Git dengan profil kredensial AWS, dan yang memungkinkan pembantu kredensial Git untuk mengirim path ke repositori:

```
git config --global credential.helper '!aws codecommit credential-helper $@'
git config --global credential.UseHttpPath true
```

Tip

Pembantu kredensial menggunakan profil kredensial AWS default atau peran instans Amazon EC2. Anda dapat menentukan profil yang akan digunakan, seperti `CodeCommitProfile` jika kamu telah menciptakan AWS profil kredensi untuk digunakan dengan CodeCommit:

```
git config --global credential.helper '!aws --profile CodeCommitProfile
codecommit credential-helper $@'
```

Jika nama profil Anda berisi spasi, pastikan Anda menyertakan nama dalam tanda kutip ("").

Anda dapat mengkonfigurasi profil per repositori bukan secara global dengan menggunakan `--local` bukan `--global`.

Pembantu kredensial Git menulis nilai berikut ke `~/.gitconfig`:

```
[credential]
  helper = !aws --profile CodeCommitProfile codecommit credential-helper $@
  UseHttpPath = true
```

⚠ Important

Jika Anda ingin menggunakan pengguna IAM yang berbeda pada mesin lokal yang sama untuk CodeCommit, Anda harus menjalankan konfigurasi perintah lagi dan tentukan yang berbeda AWS profil kredensi.

2. Jalankan `git config --global --edit` untuk memverifikasi nilai sebelumnya telah ditulis ke `~/.gitconfig`. Jika berhasil, Anda akan melihat nilai-nilai sebelumnya (selain nilai-nilai yang mungkin sudah ada di file konfigurasi global Git). Untuk keluar, biasanya Anda akan mengetik `:q`, lalu tekan Enter.


Jika Anda mengalami masalah setelah Anda mengkonfigurasi pembantu kredensial Anda, lihat [Memecahkan masalah](#).

⚠ Important

Jika Anda menggunakan macOS, gunakan langkah-langkah berikut untuk memastikan pembantu kredensial dikonfigurasi dengan benar.

3. Jika Anda menggunakan macOS, gunakan HTTPS untuk [terhubung ke CodeCommit repositori](#). Setelah Anda terhubung ke CodeCommit repositori dengan HTTPS untuk pertama kalinya, akses berikutnya gagal setelah sekitar 15 menit. Versi Git default di macOS menggunakan utilitas Keychain Access untuk menyimpan kredensial. Untuk langkah-langkah keamanan, kata sandi yang dihasilkan untuk akses ke CodeCommit repositori bersifat sementara, sehingga kredensial yang disimpan di gantungan kunci berhenti bekerja setelah sekitar 15 menit. Untuk mencegah kredensial kedaluwarsa ini digunakan, Anda harus:
 - Menginstal versi Git yang tidak menggunakan keychain secara default.
 - Konfigurasi utilitas Akses Rantai Kunci untuk tidak memberikan kredensi untuk CodeCommit repositori.
 1. Buka utilitas Keychain Access. (Anda dapat menggunakan Finder untuk menemukannya.)
 2. Cari `git-codecommit.us-east-2.amazonaws.com`. Sorot baris, buka menu konteks atau klik kanan, lalu pilih Dapatkan Info.
 3. Pilih tab Kontrol akses.

4. Dalam Konfirmasi sebelum mengizinkan akses, pilih `git-credential-osxkeychain`, lalu pilih tanda minus untuk menghapusnya dari daftar.

 Note

Setelah menghapus `git-credential-osxkeychain` dari daftar, Anda melihat pesan pop-up setiap kali Anda menjalankan perintah Git. Pilih Tolak untuk melanjutkan. Jika menurut Anda pop-up terlalu mengganggu, berikut adalah beberapa pilihan lainnya:

- Terhubung keCodeCommitmenggunakan SSH alih-alih HTTPS. Untuk informasi selengkapnya, lihat [Untuk koneksi SSH di Linux, macOS, atau Unix](#).
- Dalam utilitas Keychain Access, pada tab Kontrol Akses untuk `git-codecommit.us-east-2.amazonaws.com`, pilih Izinkan semua aplikasi mengakses item ini (akses ke item ini tidak dibatasi). Hal ini mencegah pop-up, tetapi kredensialnya akhirnya berakhir (rata-rata, dalam waktu sekitar 15 menit) dan Anda melihat pesan kesalahan 403. Jika ini terjadi, Anda harus menghapus item keychain untuk memulihkan fungsionalitas.
- Untuk informasi selengkapnya, lihat [Git untuk macOS: Saya mengkonfigurasi pembantu kredensial dengan sukses, tapi sekarang saya menolak akses ke repositori saya \(403\)](#).

Langkah 4: Hubungkan keCodeCommitkonsol dan kloning repositori

Jika administrator telah mengirim Anda nama dan detail koneksi untukCodeCommitrepositori, Anda dapat melewati langkah ini dan mengkloning repositori secara langsung.

Untuk terhubung keCodeCommitrepositori

1. BukaCodeCommitkonsol di<https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Dalam pemilih wilayah, pilih Wilayah AWS di mana repositori dibuat. Repositori adalah khusus untuk sebuah Wilayah AWS. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Temukan repositori yang ingin Anda sambungkan dari daftar dan pilih. Pilih URL Kloning, lalu pilih protokol yang ingin Anda gunakan saat mengkloning atau menghubungkan ke repositori. Langkah ini akan menyalin URL klon tersebut.

- Salin URL HTTPS jika Anda menggunakan kredensial Git dengan pengguna IAM Anda atau pembantu kredensial yang disertakan dengan AWS CLI.
- Salin URL HTTPS (GRC) jika Anda menggunakan perintah `git-remote-codecommit` di komputer lokal Anda.
- Salin URL SSH jika Anda menggunakan pasangan kunci publik/swasta SSH dengan pengguna IAM Anda.

Note

Jika Anda melihat halaman Selamat Datang bukan daftar repositori, tidak ada repositori yang terkait dengan akun AWS Anda di Wilayah AWS tempat Anda masuk. Untuk membuat repositori, lihat [the section called “Buatlah sebuah repositori”](#) atau ikuti langkah-langkah di tutorial [Memulai dengan Git dan CodeCommit](#).

4. Buka terminal dan jalankan perintah `git clone` dengan HTTPS URL yang Anda salin. Misalnya, untuk mengkloning repositori bernama *MyDemoRepo* ke repo lokal bernama *my-demo-repo* di Wilayah Timur AS (Ohio):

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Langkah selanjutnya

Anda telah menyelesaikan prasyaratnya. Ikuti langkah-langkahnya di [Memulai dengan CodeCommit](#) untuk mulai menggunakan CodeCommit.

Langkah pengaturan untuk koneksi HTTPS ke repositori AWS CodeCommit pada Windows dengan pembantu kredensial AWS CLI

Sebelum Anda dapat terhubung ke AWS CodeCommit Untuk pertama kalinya, Anda harus menyelesaikan langkah konfigurasi awal. Bagi sebagian besar pengguna, ini bisa dilakukan dengan mudah dengan mengikuti langkah-langkah di [Untuk pengguna HTTPS menggunakan kredensial](#)

[Git](#). Namun, jika Anda ingin terhubung keCodeCommitmenggunakan akun root, akses federasi, atau kredensi sementara, Anda dapat menggunakan pembantu kredensial yang disertakan dalamAWS CLI.

Note

Meskipun pembantu kredensial adalah metode yang didukung untuk menghubungkan keCodeCommitmenggunakan akses federasi, penyedia identitas, atau kredensi sementara, metode yang disarankan adalah menginstal dan menggunakangit-remote-codecommitutilitas. Untuk informasi selengkapnya, lihat [Langkah-langkah penyiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit](#).

Topik ini memandu Anda melalui langkah-langkah untuk menginstalAWS CLI, mengatur komputer Anda danAWSprofil, sambungkan keCodeCommitrepositori, dan kloning repositori itu ke komputer Anda, juga dikenal sebagai membuat repo lokal. Jika Anda baru mengenal Git, Anda mungkin juga ingin meninjau informasi di [Di mana saya dapat mempelajari lebih lanjut tentang Git?](#).

Topik

- [Langkah 1: Konfigurasi awal untukCodeCommit](#)
- [Langkah 2: Instal Git](#)
- [Langkah 3: Mengatur pembantu kredensial](#)
- [Langkah 4: Hubungkan keCodeCommitkonsol dan kloning repositori](#)
- [Langkah selanjutnya](#)


Langkah 1: Konfigurasi awal untukCodeCommit

Ikuti langkah-langkah untuk mengatur akun Amazon Web Services, membuat dan mengkonfigurasi IAM pengguna, dan menginstal AWS CLI. TheAWS CLImenyertakan pembantu kredensial yang Anda konfigurasi untuk koneksi HTTPS keCodeCommitrepositori.

Untuk membuat dan mengkonfigurasi pengguna IAM untuk mengaksesCodeCommit

1. Buat akun Amazon Web Services dengan membuka <http://aws.amazon.com> dan memilih Mendaftar.
2. Membuat pengguna IAM, atau menggunakan yang sudah ada, di akun Amazon Web Services Anda. Pastikan Anda memiliki access key ID dan secret access key yang terkait dengan

pengguna IAM. Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di akun Amazon Web Services Anda](#).

 Note

CodeCommit memerlukan AWS Key Management Service. Jika Anda menggunakan pengguna IAM yang sudah ada, pastikan tidak ada kebijakan yang dilampirkan pada pengguna yang secara tegas menolak AWS KMS tindakan yang diperlukan oleh CodeCommit. Untuk informasi selengkapnya, lihat [AWS KMS dan enkripsi](#).

3. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
4. Di konsol IAM, di panel navigasi, pilih Pengguna, dan kemudian pilih pengguna IAM yang ingin Anda konfigurasi CodeCommit akses.
5. Pada tab Izin, pilih Tambahkan izin.
6. Dalam Berikan izin, pilih Lampirkan kebijakan yang ada secara langsung.
7. Dari daftar kebijakan, pilih AWS CodeCommit Power User atau kebijakan terkelola lainnya untuk CodeCommit akses. Untuk informasi selengkapnya, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Setelah memilih kebijakan yang ingin dilampirkan, pilih Berikutnya: Tinjau untuk meninjau daftar kebijakan untuk dilampirkan ke pengguna IAM. Jika daftar sudah benar, pilih Tambahkan izin.

Untuk informasi lebih lanjut tentang CodeCommit kebijakan terkelola dan berbagi akses ke repositori dengan grup dan pengguna lain, lihat [Bagikan sebuah repositori](#) dan [Kontrol autentikasi dan akses untuk AWS CodeCommit](#).

Untuk menginstal dan mengonfigurasi AWS CLI

1. Pada komputer lokal Anda, unduh dan instal AWS CLI. Ini adalah prasyarat untuk berinteraksi dengan CodeCommit dari baris perintah. Kami merekomendasikan agar Anda menginstal AWS CLI versi 2. Ini adalah versi utama terbaru dari AWS CLI dan mendukung semua fitur terbaru. Ini adalah satu-satunya versi AWS CLI yang mendukung penggunaan akun root, akses federasi, atau kredensial sementara dengan git-remote-codecommit.

Untuk informasi selengkapnya, lihat [Menyiapkan Antarmuka Baris Perintah AWS](#).

Note

CodeCommit bekerja hanya dengan AWS CLI versi 1.7.38 dan yang lebih baru. Sebagai praktik terbaik, instal atau tingkatkan AWS CLI untuk versi terbaru yang tersedia. Untuk menentukan versi AWS CLI yang telah Anda instal, jalankan perintah `aws --version`. Untuk meng-upgrade versi lama AWS CLI ke versi terbaru, lihat [Menginstal AWS Command Line Interface](#).

2. Jalankan perintah ini untuk memverifikasi bahwa CodeCommit perintah untuk AWS CLI diinstal.

```
aws codecommit help
```

Perintah ini mengembalikan daftar CodeCommit perintah.

3. Konfigurasi AWS CLI dengan profil dengan menggunakan perintah `configure`, sebagai berikut:

```
aws configure
```

Saat diminta, tentukan AWS kunci akses dan AWS kunci akses rahasia pengguna IAM untuk digunakan CodeCommit. Juga, pastikan untuk menentukan Wilayah AWS di mana repositori ada, seperti `us-east-2`. Saat diminta untuk format output default, tentukan `json`. Misalnya, jika Anda mengonfigurasi profil untuk pengguna IAM:

```
AWS Access Key ID [None]: Type your IAM user AWS access key ID here, and then press Enter
AWS Secret Access Key [None]: Type your IAM user AWS secret access key here, and then press Enter
Default region name [None]: Type a supported region for CodeCommit here, and then press Enter
Default output format [None]: Type json here, and then press Enter
```

Untuk informasi selengkapnya tentang membuat dan mengonfigurasi profil untuk digunakan dengan AWS CLI, lihat berikut ini:

- [Profil Bernama](#)
- [Menggunakan Peran IAM dalam AWS CLI](#)
- [Tetapkan perintah](#)

- [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#)

Untuk terkoneksi ke repositori atau sumber daya di Wilayah AWS lain, Anda harus mengkonfigurasi ulang AWS CLI dengan nama wilayah default. Nama Wilayah default yang didukung untuk CodeCommit termasuk:

- us-east-2
- us-east-1
- eu-west-1
- us-west-2
- ap-northeast-1
- ap-southeast-1
- ap-southeast-2
- ap-southeast-3
- me-central-1
- eu-central-1
- ap-northeast-2
- sa-east-1
- us-west-1
- eu-west-2
- ap-south-1
- ap-south-1
- ca-central-1
- us-gov-west-1
- us-gov-east-1
- eu-north-1
- ap-east-1
- me-south-1
- cn-north-1
- cn-northwest-1

- ap-northeast-3
- af-south-1
- Il-sentral-1

Untuk informasi selengkapnya tentang CodeCommit dan Wilayah AWS, lihat [Wilayah dan titik akhir koneksi Git](#). Untuk informasi selengkapnya tentang IAM, access key, dan kunci rahasia, lihat [Bagaimana Saya Mendapatkan Kredensial?](#) dan [Mengelola Access Key untuk Pengguna IAM](#). Untuk informasi selengkapnya tentang AWS CLI dan profil, lihat [Profil Bernama](#).

Langkah 2: Instal Git

Untuk bekerja dengan file, komit, dan informasi lainnya di CodeCommit repositori, Anda harus menginstal Git pada mesin lokal Anda. CodeCommit mendukung Git versi 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git.

Untuk menginstal Git, kami merekomendasikan situs web seperti [Git untuk Windows](#). Jika Anda menggunakan link ini untuk menginstal Git, Anda dapat menerima semua pengaturan default instalasi kecuali untuk berikut:

- Ketika diminta selama langkah Menyesuaikan lingkungan PATH Anda, pilih opsi untuk menggunakan Git dari baris perintah.
- (Opsional) Jika Anda bermaksud menggunakan HTTPS dengan pembantu kredensi yang disertakan dalam AWS CLI alih-alih mengonfigurasi kredensi Git untuk CodeCommit, pada konfigurasi opsi tambahan halaman, pastikan Aktifkan Git Credential Manager opsi dihapus. Git Credential Manager hanya kompatibel dengan CodeCommit jika pengguna IAM mengonfigurasi kredensi Git. Untuk informasi selengkapnya, lihat [Untuk pengguna HTTPS menggunakan kredensial Git](#) dan [Git untuk Windows: Saya menginstal Git untuk Windows, tapi akses saya ke repositori saya ditolak \(403\)](#).

Note

Git adalah platform yang berkembang dan diperbarui secara teratur. Terkadang, perubahan fitur dapat memengaruhi cara kerjanya CodeCommit. Jika Anda mengalami masalah dengan versi tertentu dari Git dan CodeCommit, tinjau informasi di [Memecahkan masalah](#).

Langkah 3: Mengatur pembantu kredensial

The AWS CLI termasuk pembantu kredensial Git yang dapat Anda gunakan CodeCommit. Pembantu kredensial Git membutuhkan AWS Profil kredensial, yang menyimpan salinan ID access key AWS dan secret access key AWS milik pengguna IAM (bersama dengan nama Wilayah AWS default dan format output default). Pembantu kredensial Git menggunakan informasi ini untuk mengautentikasi secara otomatis CodeCommit jadi Anda tidak perlu memasukkan informasi ini setiap kali Anda menggunakan Git untuk berinteraksi CodeCommit.

1. Buka prompt perintah dan gunakan Git untuk menjalankan git config, menentukan penggunaan pembantu kredensial Git dengan profil kredensial AWS, yang memungkinkan pembantu kredensial Git untuk mengirim path ke repositori:

```
git config --global credential.helper "!aws codecommit credential-helper $@"
git config --global credential.UseHttpPath true
```

Pembantu kredensial Git menulis berikut ke file .gitconfig:

```
[credential]
  helper = !aws codecommit credential-helper $@
  UseHttpPath = true
```

Important

- Jika Anda menggunakan emulator Bash bukan baris perintah Windows, Anda harus menggunakan tanda kutip tunggal bukan tanda kutip ganda.
- Pembantu kredensial menggunakan profil AWS default atau peran instans Amazon EC2. Jika Anda telah membuat AWS profil kredensial untuk digunakan, seperti *CodeCommitProfile*, Anda dapat memodifikasi perintah sebagai berikut untuk menggunakannya sebagai gantinya:

```
git config --global credential.helper "!aws codecommit credential-helper
--profile CodeCommitProfile $@"
```

Ini menulis hal berikut ke file .gitconfig:

```
[credential]
  helper = !aws codecommit credential-helper --profile=CodeCommitProfile
  $@
  UseHttpPath = true
```

- Jika nama profil Anda berisi spasi, Anda harus mengedit file `.gitconfig` setelah menjalankan perintah ini untuk melampirkannya dalam tanda kutip tunggal (`'`). Jika tidak, pembantu kredensial tidak bekerja.
- Jika instalasi Git untuk Windows menyertakan utilitas Git Credential Manager, Anda melihat 403 kesalahan atau meminta untuk memberikan kredensial ke utilitas Credential Manager setelah beberapa upaya sambungan pertama. Cara yang paling dapat diandalkan untuk mengatasi masalah ini adalah dengan menghapus dan kemudian menginstal ulang Git untuk Windows tanpa opsi untuk utilitas Git Credential Manager, karena tidak kompatibel dengan CodeCommit. Jika Anda ingin menyimpan utilitas Git Credential Manager, Anda harus melakukan langkah-langkah konfigurasi tambahan untuk juga menggunakan CodeCommit, termasuk memodifikasi file `.gitconfig` secara manual untuk menentukan penggunaan pembantu kredensial AWS CodeCommit saat menghubungkan ke CodeCommit. Menghapus kredensial yang disimpan dari utilitas Credential Manager (Anda dapat menemukan utilitas ini di Control Panel). Setelah Anda menghapus kredensial yang disimpan, tambahkan berikut ke file `.gitconfig` Anda, simpan, dan kemudian coba sambungkan lagi dari jendela prompt perintah baru:

```
[credential "https://git-codecommit.us-east-2.amazonaws.com"]
  helper = !aws codecommit credential-helper $@
  UseHttpPath = true

[credential "https://git-codecommit.us-east-1.amazonaws.com"]
  helper = !aws codecommit credential-helper $@
  UseHttpPath = true
```

Anda mungkin juga harus mengkonfigurasi ulang pengaturan git config Anda dengan menentukan `--system` bukan `--global` atau `--local` sebelum semua koneksi bekerja seperti yang diharapkan.

- Jika Anda ingin menggunakan pengguna IAM yang berbeda pada mesin lokal yang sama untuk CodeCommit, Anda harus menentukan git config --local bukannya git config --global, dan jalankan konfigurasi untuk masing-masing AWS profil kredensi.

2. Jalankan git config --global --edit untuk memverifikasi bahwa nilai sebelumnya telah ditulis ke file .gitconfig untuk profil pengguna Anda (secara default, %HOME%\.gitconfig atau *drive*: \Users*UserName*\.gitconfig). Jika berhasil, Anda akan melihat nilai-nilai sebelumnya (selain nilai-nilai yang mungkin sudah ada di file konfigurasi global Git). Untuk keluar, biasanya Anda akan mengetik :q Lalu tekan Enter.

Langkah 4: Hubungkan ke CodeCommit konsol dan kloning repositori

Jika administrator telah mengirim Anda nama dan detail koneksi untuk CodeCommit repositori, Anda dapat melewati langkah ini dan mengkloning repositori secara langsung.

Untuk terhubung ke CodeCommit repositori

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Dalam pemilih wilayah, pilih Wilayah AWS di mana repositori dibuat. Repositori adalah khusus untuk sebuah Wilayah AWS. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Temukan repositori yang ingin Anda sambungkan dari daftar dan pilih. Pilih URL Kloning, lalu pilih protokol yang ingin Anda gunakan saat mengkloning atau menghubungkan ke repositori. Langkah ini akan menyalin URL klon tersebut.
 - Salin URL HTTPS jika Anda menggunakan kredensial Git dengan pengguna IAM Anda atau pembantu kredensial yang disertakan dengan AWS CLI.
 - Salin URL HTTPS (GRC) jika Anda menggunakan perintah git-remote-codecommit di komputer lokal Anda.
 - Salin URL SSH jika Anda menggunakan pasangan kunci publik/swasta SSH dengan pengguna IAM Anda.

Note

Jika Anda melihat halaman Selamat Datang bukan daftar repositori, tidak ada repositori yang terkait dengan akun AWS Anda di Wilayah AWS tempat Anda masuk. Untuk

membuat repositori, lihat [the section called “Buatlah sebuah repositori”](#) atau ikuti langkah-langkah di tutorial [Memulai dengan Git dan CodeCommit](#).

4. Buka prompt perintah dan jalankan perintah git clone dengan HTTPS URL yang Anda salin. Repo lokal dibuat di subdirektori direktori di mana Anda menjalankan perintah. Misalnya, untuk mengkloning repositori bernama *MyDemoRepo* ke repo lokal bernama *my-demo-repo* di Wilayah Timur AS (Ohio):

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Di beberapa versi Windows, Anda mungkin melihat pesan pop-up yang meminta nama pengguna dan kata sandi Anda. Ini adalah sistem manajemen kredensial built-in untuk Windows, namun tidak kompatibel dengan pembantu kredensial untuk AWS CodeCommit. Pilih Batalkan.

Langkah selanjutnya

Anda telah menyelesaikan prasyaratnya. Ikuti langkah-langkahnya di [Memulai dengan CodeCommit](#) untuk mulai menggunakan CodeCommit.

Mulai

Cara termudah untuk memulai dengan CodeCommit adalah dengan mengikuti langkah-langkah dalam [Memulai dengan CodeCommit](#) . Jika Anda baru mengenal Git dan CodeCommit, Anda juga harus mempertimbangkan mengikuti langkah-langkah di [Memulai dengan Git dan CodeCommit](#). Ini akan membantu Anda membiasakan diri dengan CodeCommit dan dasar-dasar menggunakan Git saat berinteraksi dengan repositori CodeCommit Anda.

Anda juga dapat mengikuti tutorial dalam [Panduan Alur Sederhana dengan CodePipeline dan CodeCommit](#) untuk mempelajari cara menggunakan repositori CodeCommit Anda sebagai bagian dari alur pengiriman berkelanjutan.

Tutorial di bagian ini ditulis dengan asumsi bahwa Anda telah menyelesaikan [Prasyarat dan pengaturan](#), termasuk:

- Menetapkan izin kepada pengguna IAM.
- Menyiapkan pengelolaan kredensial untuk koneksi HTTPS atau SSH pada mesin lokal yang Anda gunakan untuk tutorial ini.
- Mengonfigurasi AWS CLI jika Anda ingin menggunakan baris perintah atau terminal untuk semua operasi, termasuk membuat repositori.

Topik

- [Memulai dengan AWS CodeCommit](#)
- [Memulai dengan Git dan AWS CodeCommit](#)

Memulai dengan AWS CodeCommit

Tutorial ini menunjukkan cara menggunakan beberapa CodeCommit fitur utama. Pertama, Anda membuat repositori dan melakukan beberapa perubahan untuk itu. Kemudian, Anda menelusuri file dan menampilkan perubahan. Anda juga dapat membuat permintaan pull sehingga pengguna lain dapat meninjau dan mengomentari perubahan kode Anda.

Jika Anda ingin memigrasikan kode yang ada CodeCommit, lihat [Migrasi ke AWS CodeCommit](#).

Jika Anda belum pernah menggunakan Git, pertimbangkan untuk menyelesaikan [Memulai dengan Git dan CodeCommit](#) juga. Setelah Anda menyelesaikan tutorial ini, Anda harus memiliki cukup latihan untuk mulai menggunakan CodeCommit untuk proyek Anda sendiri dan di lingkungan tim.

CodeCommit Konsol menyertakan informasi bermanfaat di panel yang dapat dilipat yang dapat Anda buka dari ikon informasi



atau tautan Info apa pun di halaman. Anda dapat menutup panel ini kapan saja.

Developer Tools > CodeCommit > Repositories > MyDemoRepo

MyDemoRepo

Notify main Create pull request Clone URL

MyDemoRepo / cl_sample.js Info Edit

```

1 var aws = require('aws-sdk');
2 var codecommit = new aws.CodeCommit({ apiVersion: '2015-04-13'});
3
4 exports.handler = function(event, context) {
5   //Log the updated references from the event
6   var references = event.Records[0].codecommit.references.map(function(reference) {return
7     references.ref;});
8   console.log('References:', references);
9
10  //Get the repository from the event and show its git clone URL
11  var repository = event.Records[0].eventSourceARN.split(":")[5];
12  var params = {
13    repositoryName: repository
14  };
15  codecommit.getRepository(params, function(err, data) {
16    if (err) {
17      console.log(err);
18      var message = "Error getting repository metadata for repository " + repository;
19      console.log(message);
20      context.fail(message);
21    } else {
22      console.log('Clone URL', data.repositoryMetadata.cloneUrlHttp);
23      context.succeed(data.repositoryMetadata.cloneUrlHttp);
24    }
25  });

```

View a file in a repository

You can view the contents of most file types in the console. For Markdown files, you can switch between a rendered and source code view. For binary files, you must confirm before the file content is displayed.

If you want to quickly make a change to the file, you can edit it in the console.

Not all files can be displayed in the console. For example, if the file size exceeds the commit object limit, you cannot view it in the console. You must view it in a local repo.

Learn more

- Browse files
- Edit a file
- Add a file

CodeCommit Konsol juga menyediakan cara untuk mencari sumber daya Anda dengan cepat, seperti repositori, membangun proyek, aplikasi penerapan, dan saluran pipa. Pilih Pergi ke sumber daya atau tekan tombol /, dan kemudian ketik nama sumber daya. Setiap kecocokan akan muncul di daftar. Pencarian peka huruf besar/kecil. Anda hanya melihat sumber daya yang izin untuk menampilkannya Anda memiliki. Untuk informasi selengkapnya, lihat [Menampilkan sumber daya di konsol](#).

Prasyarat

Sebelum memulai, Anda harus menyelesaikan [prasyarat dan pengaturan](#), termasuk:

- Menetapkan izin ke pengguna IAM.

- Menyiapkan manajemen credential untuk koneksi HTTPS atau SSH pada mesin lokal yang Anda gunakan untuk tutorial ini.
- Mengkonfigurasi AWS CLI jika Anda ingin menggunakan baris perintah atau terminal untuk semua operasi, termasuk untuk membuat repositori.

Topik

- [Langkah 1: Buat CodeCommit repositori](#)
- [Langkah 2: Tambahkan file ke repositori](#)
- [Langkah 3: Menelusuri isi repositori Anda](#)
- [Langkah 4: Buat dan kolaborasi pada permintaan pull](#)
- [Langkah 5: Bersihkan](#)
- [Langkah 6: Langkah selanjutnya](#)

Langkah 1: Buat CodeCommit repositori


Anda dapat menggunakan CodeCommit konsol untuk membuat CodeCommit repositori. Jika Anda sudah memiliki repositori yang ingin Anda gunakan untuk tutorial ini, Anda dapat melewati langkah ini.

Note

Tergantung pada penggunaan Anda, Anda mungkin dikenai biaya untuk membuat atau mengakses suatu repositori. Untuk informasi selengkapnya, lihat [Harga](#) di halaman informasi CodeCommit produk.

Untuk membuat CodeCommit repositori

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Gunakan pemilih wilayah untuk memilih Wilayah AWS tempat Anda ingin membuat repositori. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Pada halaman Repositori, pilih Buat repositori.
4. Pada halaman Buat repositori, dalam Nama Repositori, ketikkan nama untuk repositori Anda, (misalnya **MyDemoRepo**).

 Note

Nama repositori peka huruf besar kecil dan tidak boleh lebih dari 100 karakter. Untuk informasi selengkapnya, lihat [Batasan-batasan](#).

5. (Optional) Dalam Deskripsi, masukkan deskripsi (misalnya, **My demonstration repository**). Ini dapat membantu Anda dan pengguna lain mengidentifikasi tujuan repositori.
6. (Opsional) Pilih Tambahkan tag untuk menambahkan satu atau beberapa tag repositori (label atribut khusus yang membantu Anda mengatur dan mengelola AWS sumber daya) ke repositori Anda. Untuk informasi selengkapnya, lihat [Menandai repositori di AWS CodeCommit](#).
7. (Opsional) Perluas konfigurasi tambahan untuk menentukan apakah akan menggunakan default Kunci yang dikelola AWS atau kunci yang dikelola pelanggan Anda sendiri untuk mengenkripsi dan mendekripsi data dalam repositori ini. Jika Anda memilih untuk menggunakan kunci yang dikelola pelanggan Anda sendiri, Anda harus memastikan bahwa itu tersedia di Wilayah AWS tempat Anda membuat repositori, dan bahwa kuncinya aktif. Untuk informasi selengkapnya, lihat [AWS Key Management Service dan enkripsi untuk repositori AWS CodeCommit](#).
8. (Opsional) Pilih Aktifkan Amazon CodeGuru Reviewer untuk Java dan Python jika repositori ini akan berisi kode Java atau Python, dan Anda ingin Reviewer menganalisis kode itu. CodeGuru CodeGuru Reviewer menggunakan beberapa model pembelajaran mesin untuk menemukan cacat kode dan secara otomatis menyarankan perbaikan dan perbaikan dalam permintaan tarik. Untuk informasi selengkapnya, lihat Panduan Pengguna Amazon CodeGuru Reviewer.
9. Pilih Buat.

Create repository

Create a secure repository to store and share your code. Begin by typing a repository name and a description for your repository. Repository names are included in the URLs for that repository.

Repository settings

Repository name

100 characters maximum. Other limits apply.

Description - *optional*

1,000 characters maximum

Tags

Enable Amazon CodeGuru Reviewer for Java and Python - *optional*

Get recommendations to improve the quality of the Java and Python code for all pull requests in this repository.

A service-linked role will be created in IAM on your behalf if it does not exist.

Cancel

Create

Note

Jika Anda menggunakan nama selain MyDemoRepo untuk repositori Anda, pastikan untuk menggunakannya dalam langkah-langkah selanjutnya.

Ketika repositori terbuka, Anda melihat informasi tentang cara menambahkan file langsung dari konsol. CodeCommit

Langkah 2: Tambahkan file ke repositori

Anda dapat menambahkan file ke repositori Anda dengan:

- Membuat file di CodeCommit konsol. Jika Anda membuat file pertama untuk repositori di konsol tersebut, cabang dibuat untuk Anda yang dinamai utama. Cabang ini adalah cabang default untuk repositori Anda.
- Mengunggah file dari komputer lokal Anda menggunakan CodeCommit konsol. Jika Anda mengunggah file pertama untuk repositori dari konsol tersebut, cabang dibuat untuk Anda yang dinamai utama. Cabang ini adalah cabang default untuk repositori Anda.
- Menggunakan klien Git untuk mengkloning repositori ke komputer lokal Anda, dan kemudian menambahkan, melakukan, dan mendorong file ke repositori. CodeCommit Sebuah cabang dibuat untuk Anda sebagai bagian dari commit yang pertama dari Git, dan diatur sebagai cabang default untuk repositori Anda. Nama cabang adalah pilihan default klien Git Anda. Pertimbangkan untuk mengkonfigurasi klien Git Anda untuk menggunakan utama sebagai nama untuk cabang awal.

Note

Anda dapat membuat cabang dan mengubah cabang default untuk repositori kapan saja. Untuk informasi selengkapnya, lihat [Bekerja dengan cabang di AWS CodeCommit repositori](#).

Cara termudah untuk memulai adalah membuka CodeCommit konsol dan menambahkan file. Dengan demikian, Anda juga membuat cabang default untuk repositori Anda yang dinamai utama. Untuk petunjuk tentang cara menambahkan file dan membuat komit pertama ke repositori menggunakan file AWS CLI, lihat [Membuat komit pertama untuk repositori](#) menggunakan. AWS CLI

Untuk menambahkan file ke repositori

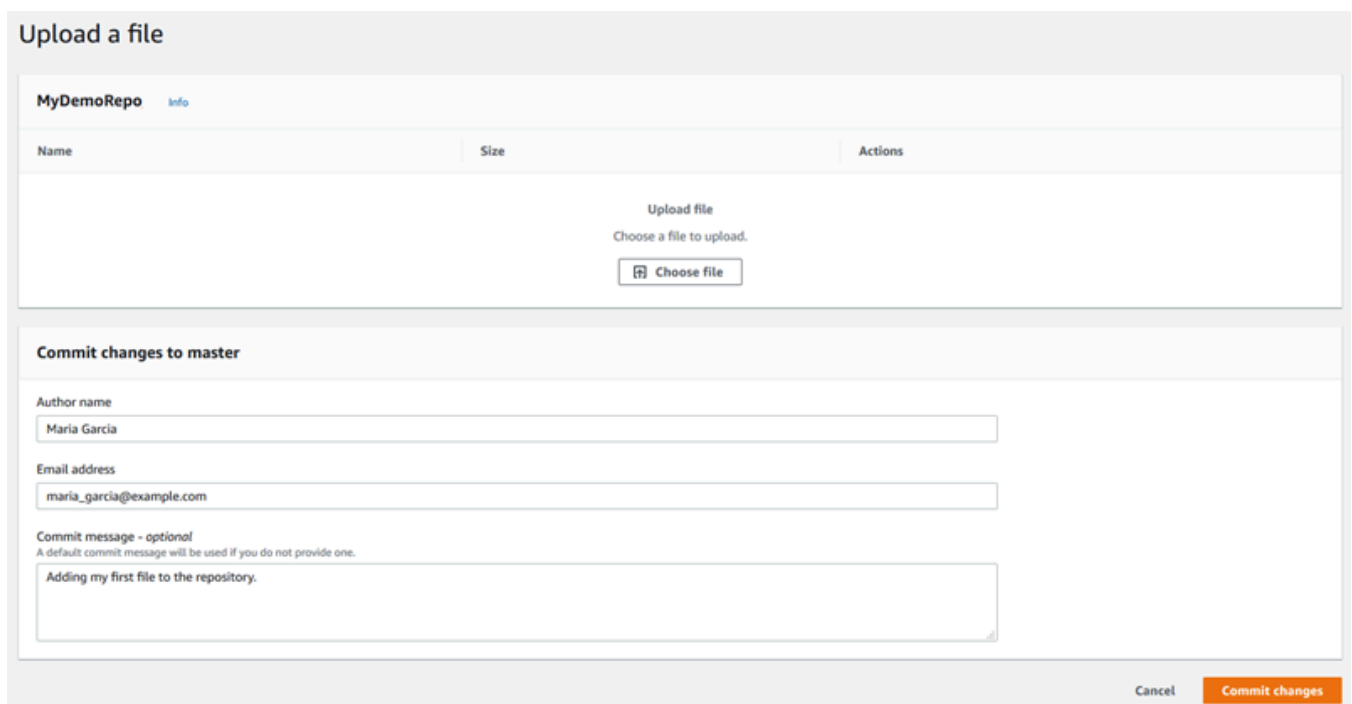
1. Di bilah navigasi untuk repositori, pilih Kode.
2. Pilih Tambah file, lalu pilih apakah akan membuat file atau mengunggah file dari komputer Anda. Tutorial ini menunjukkan cara melakukan keduanya.
3. Untuk menambahkan file, lakukan hal berikut:
 - a. Dalam daftar drop-down cabang, pilih cabang tempat Anda ingin menambahkan file. Cabang default dipilih secara otomatis untuk Anda. Pada contoh yang ditampilkan di sini,

cabang default dinamai *utama*. Jika Anda ingin menambahkan file ke cabang yang berbeda, pilih cabang yang berbeda.

- b. Di Nama file, masukkan nama untuk file tersebut. Di editor kode, masukkan kode untuk file.
- c. Di Nama penulis, masukkan nama yang ingin ditampilkan ke pengguna repositori lainnya.
- d. Di Alamat Email, masukkan alamat email.
- e. (Opsional) Dalam Pesan commit, masukkan pesan singkat. Meskipun ini opsional, kami menyarankan Anda menambahkan pesan commit untuk membantu anggota tim Anda memahami mengapa Anda menambahkan file ini. Jika Anda tidak memasukkan pesan commit, pesan default akan digunakan.
- f. Pilih Perubahan commit.

Untuk mengunggah file, lakukan hal berikut:

- Jika Anda mengunggah file, pilih file yang ingin unggah.



The screenshot displays the 'Upload a file' interface in AWS CodeCommit. At the top, it shows the repository name 'MyDemoRepo' with an 'info' link. Below this is a table with columns for 'Name', 'Size', and 'Actions'. The table is currently empty, and a central 'Upload file' button is visible with the text 'Choose a file to upload.' and a 'Choose file' button. Below the table is a section titled 'Commit changes to master' which contains three input fields: 'Author name' (filled with 'Maria Garcia'), 'Email address' (filled with 'maria_garcia@example.com'), and 'Commit message - optional' (filled with 'Adding my first file to the repository.'). At the bottom right of the form are 'Cancel' and 'Commit changes' buttons.

- Di Nama penulis, masukkan nama yang ingin ditampilkan ke pengguna repositori lainnya.
- Di Alamat Email, masukkan alamat email.
- (Opsional) Dalam Pesan commit, masukkan pesan singkat. Meskipun ini opsional, kami menyarankan Anda menambahkan pesan commit untuk membantu anggota tim Anda memahami mengapa Anda menambahkan file ini. Jika Anda tidak memasukkan pesan commit, pesan default akan digunakan.

- Pilih Perubahan commit.

Untuk informasi selengkapnya, lihat [Bekerja dengan file repositori AWS CodeCommit](#).

Untuk menggunakan klien Git untuk mengkloning repositori, instal Git di komputer lokal Anda, lalu kloning repositori. CodeCommit Tambahkan beberapa file ke repo lokal dan dorong ke CodeCommit repositori. Untuk pengenalan mendalam, coba [Memulai dengan Git dan CodeCommit](#). Jika Anda terbiasa dengan Git, tetapi tidak yakin bagaimana melakukannya dengan CodeCommit repositori, Anda dapat melihat contoh dan instruksi di [Buatlah sebuah commit](#), [Langkah 2: Buat repo lokal](#), atau [Connect ke sebuah repositori](#)

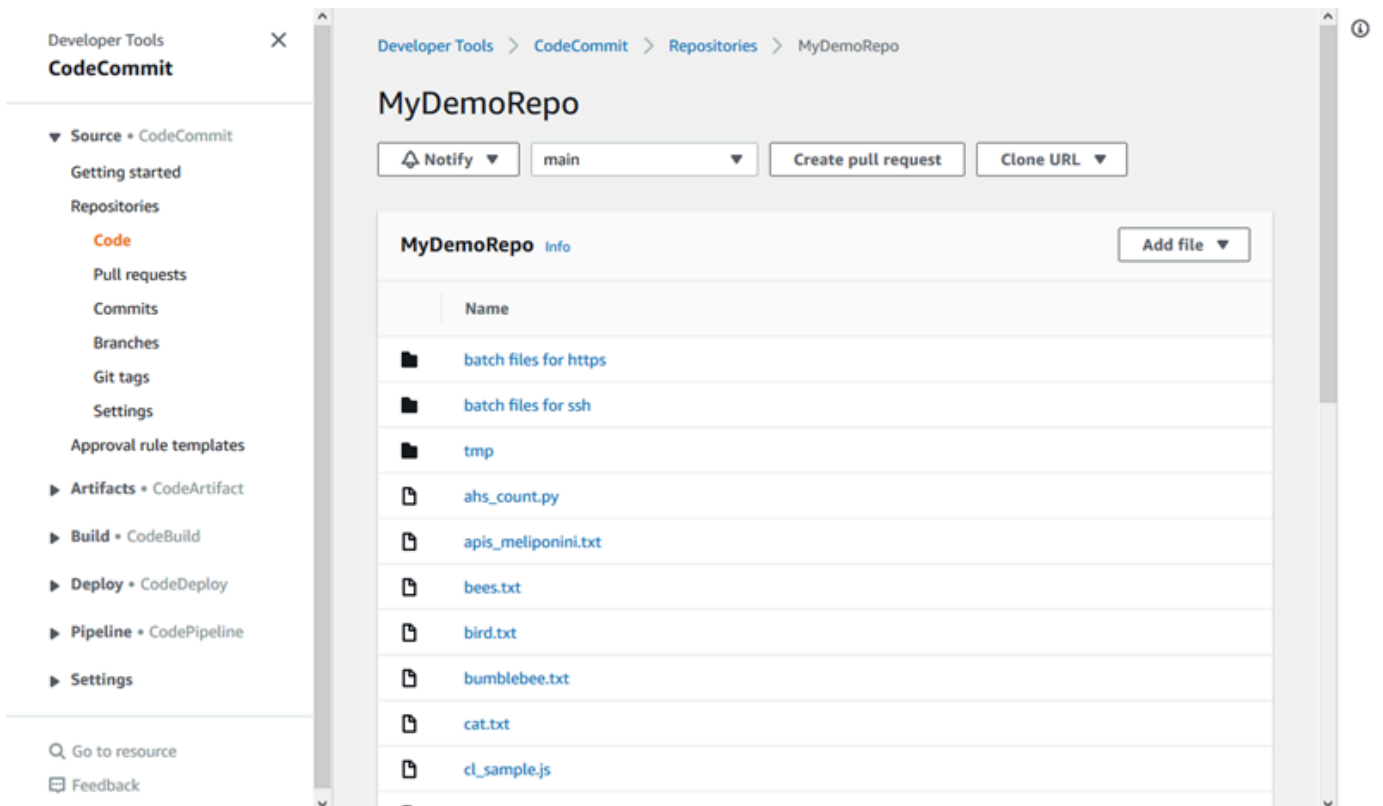
Setelah Anda menambahkan beberapa file ke CodeCommit repositori, Anda dapat melihatnya di konsol.

Langkah 3: Menelusuri isi repositori Anda

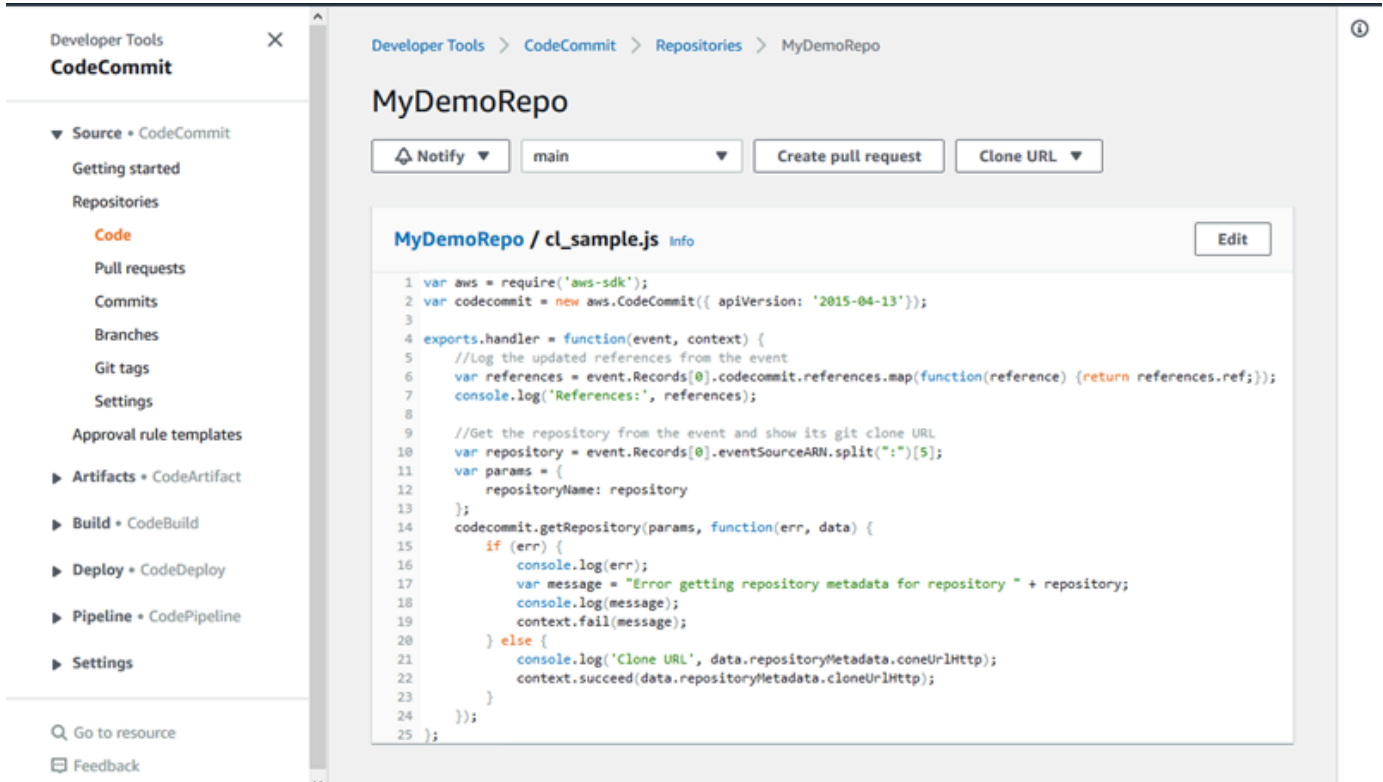
Anda dapat menggunakan CodeCommit konsol untuk meninjau file dalam repositori atau membaca konten file dengan cepat. Ini membantu Anda menentukan cabang mana yang akan diperiksa atau apakah Anda akan membuat salinan lokal dari suatu repositori.

Untuk menelusuri repositori

1. Dari Repositori, pilih. MyDemoRepo
2. Halaman tersebut menampilkan isi di cabang default repositori Anda. Untuk menampilkan cabang lain atau untuk menampilkan kode pada tanda tertentu, pilih cabang atau tanda yang ingin Anda lihat dari daftar. Di tangkapan layar berikut, tampilan diatur ke cabang utama.

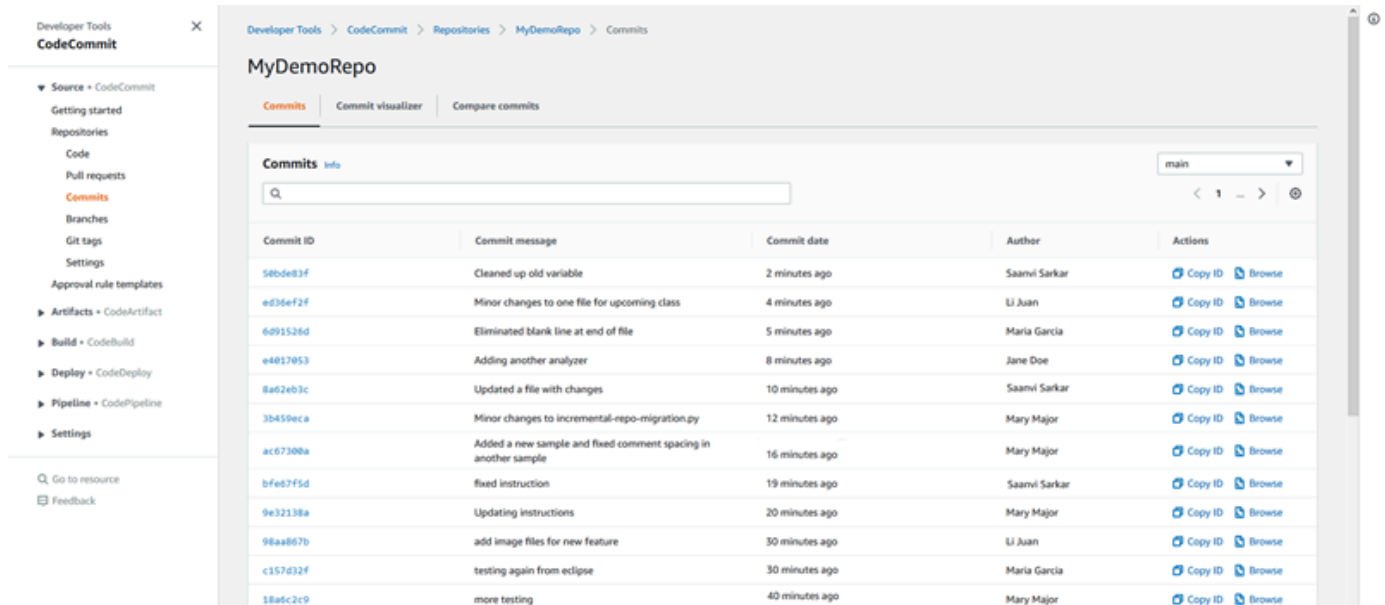


3. Untuk menampilkan isi file di repositori Anda, pilih file tersebut dari daftar. Untuk mengubah warna kode yang ditampilkan, pilih ikon pengaturan.



Untuk informasi selengkapnya, lihat [Menelusuri file dalam sebuah repositori](#).

4. Untuk menelusuri riwayat commit repositori, pilih commit. Konsol menampilkan riwayat commit untuk cabang default, dalam urutan kronologis terbalik. Tinjau detail commit berdasarkan penulis, tanggal, dan lainnya.



5. Untuk menampilkan riwayat commit berdasarkan [cabang](#) atau berdasarkan [Tanda Git](#), pilih cabang atau tanda yang ingin Anda tampilkan dari daftar.
6. Untuk menampilkan perbedaan antara commit dan commit induknya, pilih ID commit disingkat. Anda dapat memilih bagaimana perubahan ditampilkan, termasuk menampilkan atau menyembunyikan perubahan spasi, dan apakah akan menampilkan perubahan yang segaris (tampilan Digabung) atau berdampingan (tampilan Dibagi).

Note

Pilihan Anda untuk menampilkan kode dan pengaturan konsol lainnya disimpan sebagai cookie browser setiap kali Anda mengubahnya. Untuk informasi selengkapnya, lihat [Bekerja dengan preferensi pengguna](#).

Developer Tools > CodeCommit > Repositories > MyDemoRepo > Commits > 7d09e44c

Commit 7d09e44c

▼ Details

Author	Commit date	Parent commit
Mary Major mary_major@example.com	48 minutes ago	e6aca768

Commit message

Adding a readme file to the repository.

< Page 1 of 1 > Hide whitespace changes Unified Split

readme.md

```
1 - This is a readme file that provides a basic description of what's in this repository.
   \ No newline at end of file
1 + Use this repository for code changes to the *Demo* project. The default branch is *master*. Cod
   \ No newline at end of file
```

7. Untuk menampilkan semua komentar pada suatu commit, pilih commit dan kemudian gulir melalui perubahan untuk menampilkannya secara satu baris. Anda juga dapat menambahkan komentar Anda sendiri dan membalas komentar yang dibuat oleh orang lain.

Untuk informasi selengkapnya, lihat [Komentari sebuah commit](#).

8. Untuk menampilkan perbedaan antara dua penspesifikasi commit, termasuk tanda, cabang, dan ID commit, di panel navigasi, pilih commit, lalu pilih Bandingkan commit.

Developer Tools > CodeCommit > Repositories > MyDemoRepo > Compare

MyDemoRepo

Commits | Commit visualizer | **Compare commits**

Destination: AnotherBranch | Source: 6b65eb76 | **Compare** | Cancel

< Page 1 of 1 > | Go to file | Hide whitespace changes | Unified | Split

ahs_count.py | Browse file contents | Comment on file

```
***      ***      @@ -5,6 +5,6 @@
5        5
6        6      total = (ess + z)
7        7      ahs = "Number of alveolar hissing sibilants: {}"
8        8      - print(ahs.format(total))
9        9
10       10     + print(alv.format(total))
10       10     #When using this script, make sure that you ask the subject to use one of the provided texts, such as bumblebee.txt.
```

anothernew/dir2/anotherstest.txt Added | Browse file contents | Comment on file

Untuk informasi selengkapnya, lihat [Telusuri riwayat commit dari sebuah repositori](#) dan [Bandingkan commit](#).

9. Di commit, pilih tab Penampil visual commit.

Developer Tools > CodeCommit > Repositories > MyDemoRepo > Commits

MyDemoRepo

Commits | **Commit visualizer** | Compare commits

Commit visualizer

d615e7ae	Merge branch 'AnotherBranch' into testbranch	2 minutes ago
b6589863	Added another file.	2 minutes ago
73a6e39c	remote-tracking branch 'refs/remotes/origin/jane-branch' into jane-branch	
6bbb6d3c	Another test of the editing feature.	20 minutes ago
edacdffe	Testing this out to see how well it works.	23 minutes ago
70bb94d7	Revised test results with correct information.	36 minutes ago
b78e6d1c	Merge branch 'results' into testbranch	50 minutes ago
84b7d158	Edited ahs_count.py	50 minutes ago

Grafik commit ditampilkan, dengan baris subjek untuk setiap commit yang ditampilkan di samping titik dalam grafik. Tampilan baris subjek dibatasi hingga 80 karakter.

10. Untuk menampilkan detail lebih lanjut tentang commit, pilih ID commit disingkat. Untuk membuat grafik dari commit tertentu, pilih titik itu dalam grafik. Untuk informasi selengkapnya, lihat [Tampilkan sebuah grafik riwayat commit dari sebuah repositori](#).

Langkah 4: Buat dan kolaborasi pada permintaan pull

Ketika Anda bekerja di repositori dengan pengguna lain, Anda dapat berkolaborasi pada kode dan meninjau perubahan. Anda dapat membuat permintaan pull sehingga pengguna lain dapat meninjau dan mengomentari perubahan kode Anda di suatu cabang. Anda juga dapat membuat satu atau lebih aturan persetujuan untuk permintaan pull. Misalnya, Anda dapat membuat aturan persetujuan yang mengharuskan setidaknya dua pengguna lain untuk menyetujui permintaan pull sebelum dapat digabung. Setelah permintaan pull disetujui, Anda dapat menggabungkan perubahan tersebut ke cabang tujuan. Jika Anda mengatur pemberitahuan untuk repositori Anda, pengguna repositori dapat menerima email tentang peristiwa repositori (misalnya, untuk permintaan pull atau ketika seseorang

mengomentari kode). Untuk informasi selengkapnya, lihat [Mengkonfigurasi notifikasi untuk peristiwa dalam sebuah repositori AWS CodeCommit](#).

Important

Sebelum Anda dapat membuat permintaan pull, Anda harus membuat cabang yang berisi perubahan kode yang ingin Anda tinjau. Untuk informasi selengkapnya, lihat [Buatlah sebuah cabang](#).

Untuk membuat dan berkolaborasi pada permintaan pull

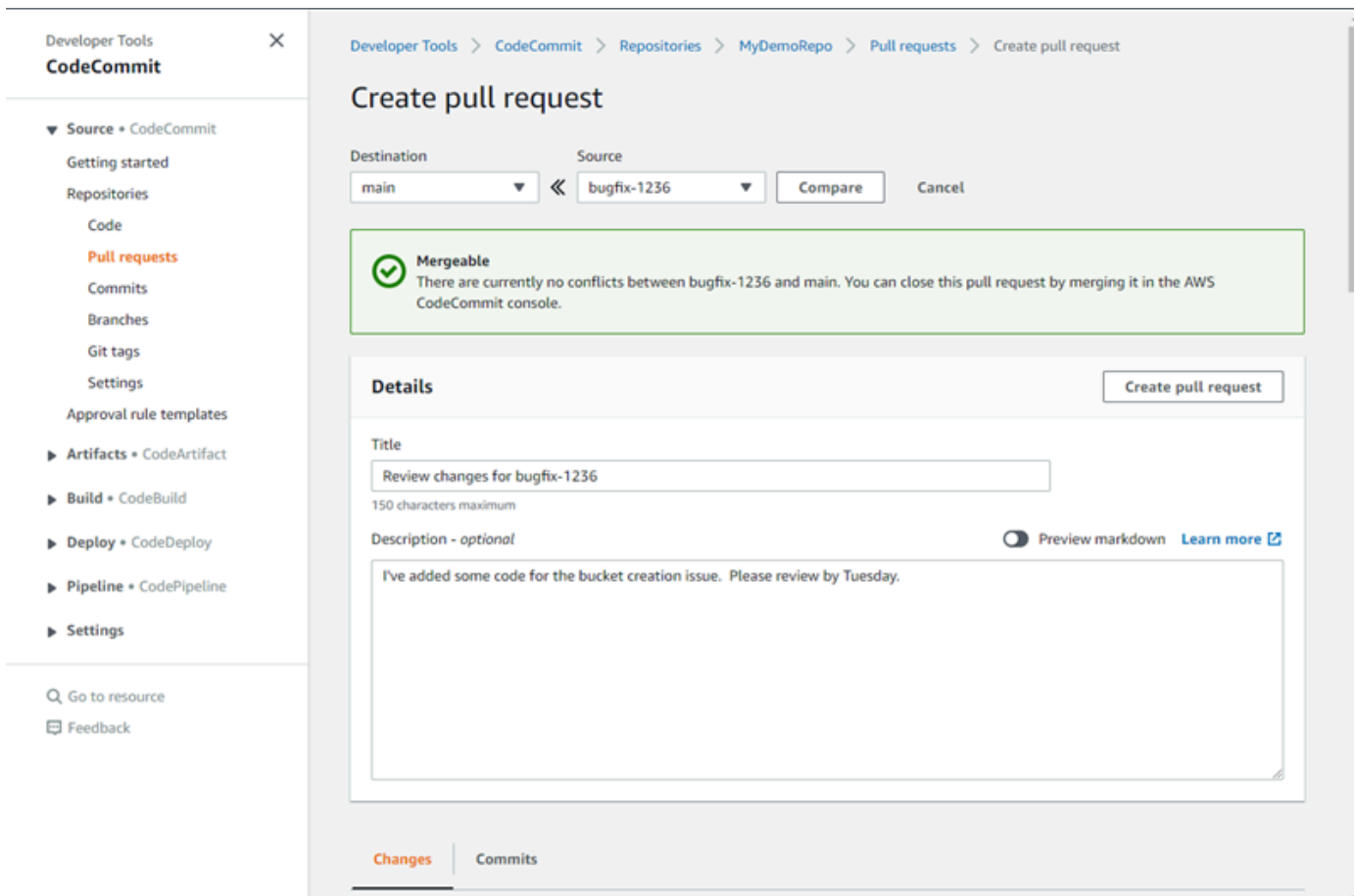
1. Di panel navigasi, pilih Permintaan pull.
2. Di permintaan pull, pilih Buat permintaan pull.

Tip

Anda juga dapat membuat permintaan pull dari Cabang dan Kode.

Di Buat permintaan pull, di Sumber, pilih cabang yang berisi perubahan yang ingin Anda tinjau. Di Tujuan, pilih cabang tempat Anda ingin kode yang ditinjau digabung saat permintaan pull ditutup. Pilih Bandingkan.

3. Meninjau detail gabungan dan perubahan untuk mengkonfirmasi bahwa permintaan pull berisi perubahan dan commit yang ingin Anda tinjau. Jika ya, di Judul, masukkan judul untuk tinjauan ini. Ini adalah judul yang muncul dalam daftar permintaan pull untuk repositori. Di Deskripsi, masukkan detail tentang ulasan ini dan informasi berguna lainnya untuk peninjau. Pilih Buat.

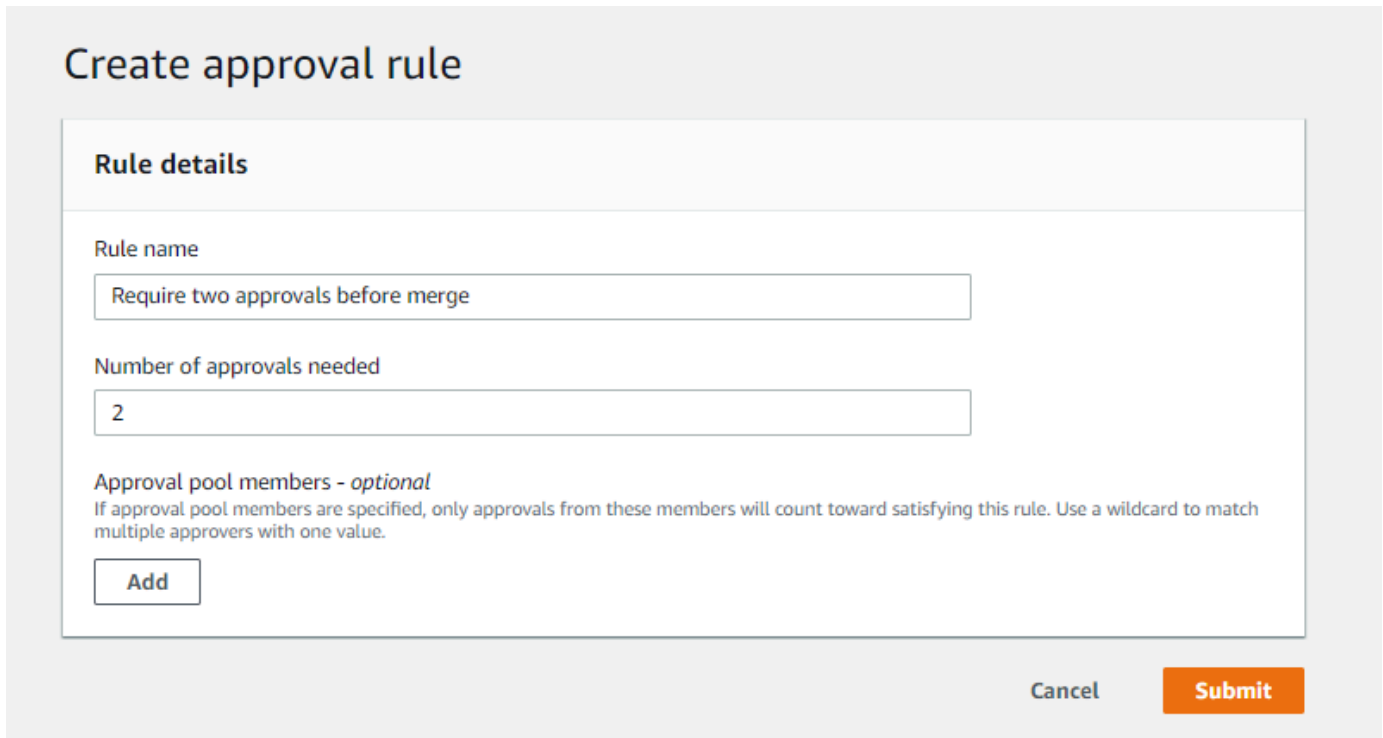


4. Permintaan pull Anda muncul dalam daftar permintaan pull untuk repositori. Anda dapat memfilter tampilan untuk hanya menampilkan permintaan terbuka, permintaan tertutup, permintaan yang Anda buat, dan lainnya.

Pull requests Info					
Pull request	Author	Destination	Last activity	Status	Approval status
31: testing this	Saanvi_Sarkar	preprod	4 minutes ago	Open	No approval rules
25: Updated some of our Java samples	Li_Juan	main	5 minutes ago	Open	0 of 1 rules satisfied
29: Changing duplicate value	Li_Juan	main	19 minutes ago	Open	0 of 1 rules satisfied
22: Test pull request	Saanvi_Sarkar	preprod	20 minutes ago	Open	No approval rules
28: Changes to some of our code samples	Li_Juan	main	1 month ago	Open	0 of 1 rules satisfied
20: A bugfix to add missing examples for S3	Saanvi_Sarkar	main	1 month ago	Open	0 of 1 rules satisfied

5. Anda dapat menambahkan aturan persetujuan untuk permintaan pull Anda untuk memastikan bahwa kondisi tertentu terpenuhi sebelum dapat digabung. Untuk menambahkan aturan persetujuan pada permintaan pull, pilih permintaan pull dari daftar. Pada tab Persetujuan, pilih Buat aturan persetujuan.

- Di Nama aturan, berikan nama deskriptif pada aturan tersebut. Misalnya, jika Anda ingin meminta dua orang untuk menyetujui permintaan pull sebelum dapat digabung, Anda dapat menyebutkan aturan **Require two approvals before merge**. Di Jumlah persetujuan yang dibutuhkan, masukkan **2**, jumlah yang Anda inginkan. Default-nya adalah 1. Pilih Kirim. Untuk mempelajari lebih lanjut tentang aturan persetujuan dan anggota kolam persetujuan, lihat [Buat aturan persetujuan untuk permintaan pull](#).



Create approval rule

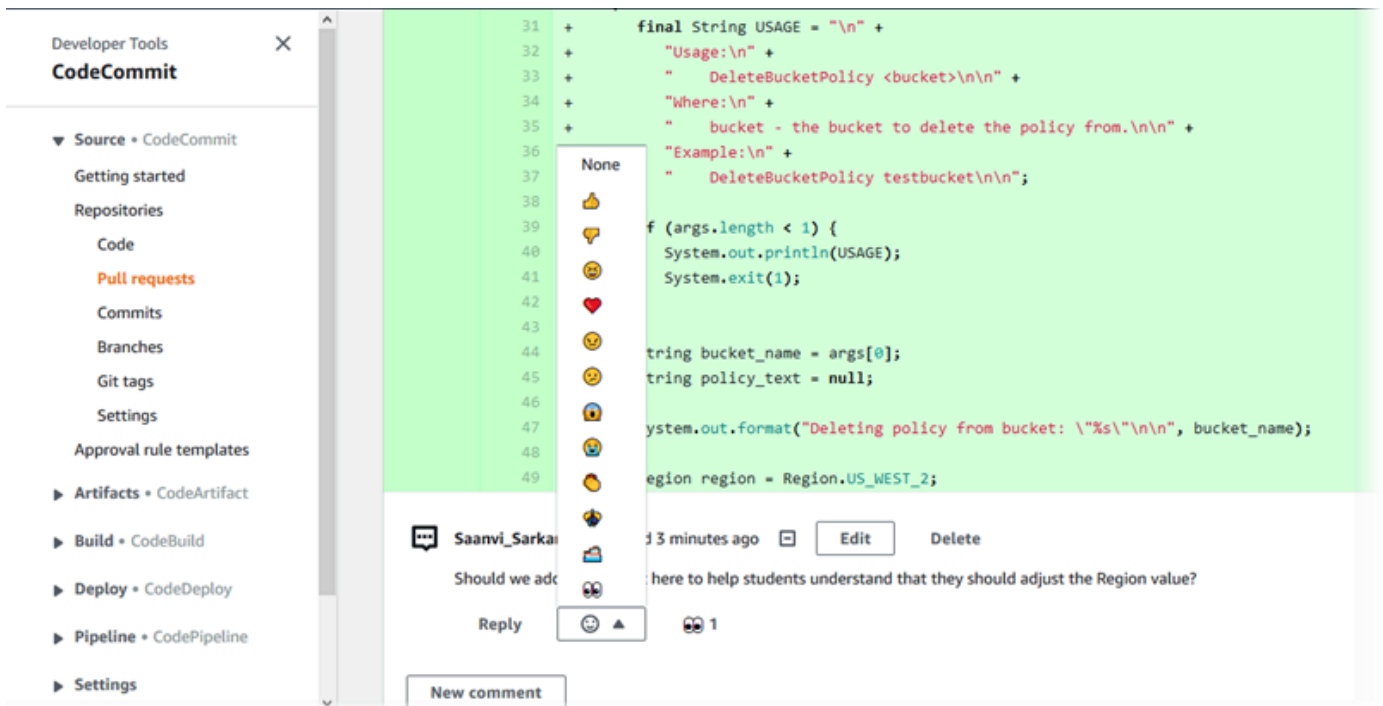
Rule details

Rule name

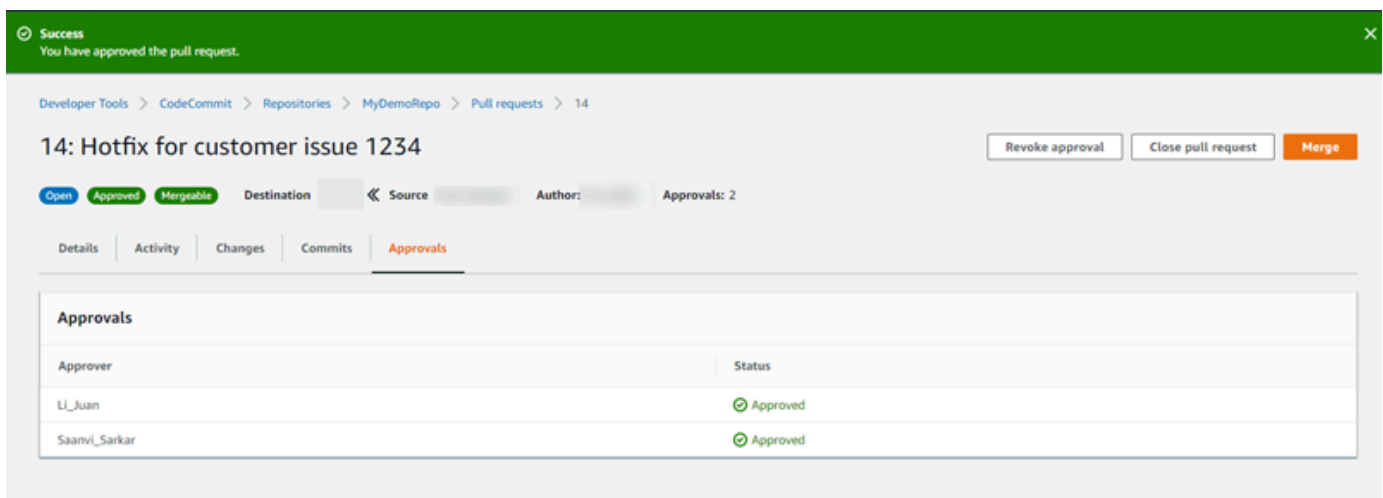
Number of approvals needed

Approval pool members - *optional*
If approval pool members are specified, only approvals from these members will count toward satisfying this rule. Use a wildcard to match multiple approvers with one value.

- Jika Anda mengkonfigurasi pemberitahuan untuk repositori Anda dan memilih untuk memberitahu pengguna peristiwa permintaan pull, pengguna menerima email tentang permintaan pull baru Anda. Pengguna dapat menampilkan perubahan dan komentar pada baris kode tertentu, file, dan permintaan pull itu sendiri. Mereka juga dapat membalas komentar dengan teks dan emoji. Jika perlu, Anda dapat mengirim perubahan ke cabang permintaan pull, yang memperbarui permintaan pull.



8. Jika Anda puas dengan perubahan yang dibuat dalam permintaan, pilih Setujui. Anda dapat memilih untuk menyetujui permintaan pull meskipun tidak ada aturan persetujuan yang dikonfigurasi untuk permintaan pull tersebut. Hal ini memberikan catatan yang jelas tentang Anda yang telah meninjau permintaan pull dan persetujuan Anda atas perubahan tersebut. Anda juga dapat memilih untuk mencabut persetujuan Anda jika Anda berubah pikiran.



Note

Anda tidak dapat menyetujui permintaan pull jika Anda membuatnya.

9. Ketika Anda puas bahwa semua perubahan kode telah ditinjau dan disetujui, dari permintaan pull, lakukan salah satu dari berikut ini:
- Jika Anda ingin menutup permintaan pull tanpa menggabungkan cabang, pilih Tutup permintaan pull.
 - Jika Anda ingin menggabungkan cabang-cabang dan menutup permintaan pull, pilih Gabungkan. Anda dapat memilih antara strategi penggabungan yang tersedia untuk kode Anda, yang bergantung pada perbedaan antara cabang asal dan tujuan, dan apakah akan secara otomatis menghapus cabang sumber setelah penggabungan selesai. Setelah Anda membuat pilihan Anda, pilih Gabungkan permintaan pull untuk menyelesaikan penggabungan.

Merge pull request 9: Bug fix for unhandled exception


Merge request details

Pull request: #9 Bug fix for unhandled exception


Destination main **Source** bugfix-bug1234

Merge strategy [Info](#)
Determines the way in which the current pull request will be merged into the destination branch


Fast forward merge
`git merge --ff-only`
Merges the branches and moves the destination branch pointer to the tip of the source branch. This is the default merge strategy in Git.



Squash and merge
`git merge --squash`
Combine all commits from the source branch into a single merge commit in the destination branch.



3-way merge
`git merge --no-ff`
Create a merge commit and adds individual source commits to the destination branch.



Commit message - optional Preview markdown

Squashed commit of the following

```
commit d49940ad
Author: Li Juan <li_juan@example.com>
Date: Tue May 07 2019 15:12:48 GMT-0700 (Pacific Daylight Time)

Fixing the bug reported in 1234.
```

Author name

Email address

Delete source branch bugfix-bug1234 after merging?

[Cancel](#) [Merge pull request](#)

- Jika ada konflik gabungan di cabang yang tidak dapat diselesaikan secara otomatis, Anda dapat menyelesaikannya di CodeCommit konsol, atau Anda dapat menggunakan klien Git lokal Anda untuk menggabungkan cabang dan kemudian mendorong penggabungan. Untuk informasi selengkapnya, lihat [Menyelesaikan konflik dalam permintaan pull dalam repositori AWS CodeCommit](#).

Note

Anda dapat selalu secara manual menggabungkan cabang, termasuk cabang permintaan pull, dengan menggunakan perintah git merge di repo lokal Anda dan mengirim perubahan Anda.

Untuk informasi selengkapnya, lihat [Bekerja dengan permintaan pull](#) dan [Bekerja dengan templat aturan persetujuan](#).

Langkah 5: Bersihkan

Jika Anda tidak lagi memerlukan CodeCommit repositori, Anda harus menghapus CodeCommit repositori dan sumber daya lain yang Anda gunakan dalam latihan ini sehingga Anda tidak akan terus dikenakan biaya untuk ruang penyimpanan.

⚠ Important

Tindakan ini tidak dapat dibatalkan. Setelah Anda menghapus repositori ini, Anda tidak lagi dapat mengkloningnya ke repo lokal atau repo bersama. Anda juga tidak bisa lagi menarik data dari atau mengirim data ke repositori tersebut, atau melakukan operasi Git, dari repo lokal atau repo bersama.

Jika Anda mengonfigurasi notifikasi untuk repositori, menghapus repositori juga akan menghapus aturan Amazon CloudWatch Events yang dibuat untuk repositori. Ini tidak menghapus topik Amazon SNS yang digunakan sebagai target untuk aturan itu.

Jika Anda mengkonfigurasi pemicu untuk repositori Anda, menghapus repositori tidak menghapus topik Amazon SNS atau fungsi Lambda yang Anda konfigurasi sebagai target dari pemicu tersebut. Pastikan untuk menghapus sumber daya tersebut jika Anda tidak membutuhkannya. Untuk informasi selengkapnya, lihat [Hapus pemicu dari repositori](#).

Untuk menghapus CodeCommit repositori

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori yang ingin Anda hapus. Jika Anda mengikuti konvensi penamaan dalam topik ini, itu dinamai MyDemoRepo.

3. Pada panel navigasi, silakan pilih Pengaturan.
4. Pada halaman Pengaturan, di Hapus repositori, pilih Hapus repositori.
5. Ketik **delete**, lalu pilih Hapus. Repositori dihapus secara permanen.

Langkah 6: Langkah selanjutnya

Sekarang setelah Anda membiasakan diri dengan CodeCommit dan beberapa fitur-fiturnya, pertimbangkan untuk melakukan hal berikut:

- Jika Anda baru mengenal Git dan CodeCommit atau ingin meninjau contoh penggunaan Git dengan CodeCommit, lanjutkan ke [Memulai dengan Git dan CodeCommit](#) tutorial.
- Jika Anda ingin bekerja dengan orang lain di CodeCommit repositori, lihat. [Bagikan sebuah repositori](#) (Jika Anda ingin berbagi repositori Anda dengan pengguna di akun Amazon Web Services lain, lihat [Konfigurasi akses lintas akun ke AWS CodeCommit repositori menggunakan peran.](#))
- Jika Anda ingin memigrasikan repositori ke CodeCommit, ikuti langkah-langkahnya. [Migrasi ke CodeCommit](#)
- Jika Anda ingin menambahkan repositori Anda ke pipeline pengiriman berkelanjutan, ikuti langkah-langkah di [Panduan Pipeline Sederhana](#).
- Jika Anda ingin mempelajari lebih lanjut tentang produk dan layanan yang terintegrasi CodeCommit, termasuk contoh dari komunitas, lihat [Integrasi produk dan layanan](#).

Memulai dengan Git dan AWS CodeCommit

Jika Anda baru mengenal Git dan CodeCommit, tutorial ini membantu Anda mempelajari beberapa perintah sederhana untuk memulai. Jika Anda sudah pernah menggunakan Git, Anda dapat melewati tutorial ini dan pergi ke [Memulai dengan CodeCommit](#).

Dalam tutorial ini, Anda membuat repositori yang mewakili salinan lokal dari CodeCommit repositori, yang kami sebut sebagai repo lokal.

Setelah Anda membuat repo lokal, Anda membuat beberapa perubahan untuk itu. Kemudian Anda mengirim (mendorong) perubahan Anda ke CodeCommit repositori.

Anda juga mensimulasikan lingkungan tim di mana dua pengguna secara independen melakukan perubahan ke repo lokal mereka dan mendorong perubahan tersebut ke repositori. CodeCommit

Pengguna kemudian menarik perubahan dari CodeCommit repositori ke repo lokal mereka sendiri untuk melihat perubahan yang dibuat pengguna lain.

Anda juga membuat cabang dan tag dan mengelola beberapa izin akses di CodeCommit repositori.

Setelah Anda menyelesaikan tutorial ini, Anda harus memiliki cukup latihan dengan inti Git dan CodeCommit konsep untuk menggunakannya untuk proyek Anda sendiri.

Selesaikan [prasyarat dan pengaturan](#), termasuk:

- Tetapkan izin ke pengguna IAM.
- Siapkan CodeCommit untuk terhubung ke repositori menggunakan [HTTPS](#), SSH, atau [git-remote-codecommit](#) Untuk informasi selengkapnya tentang pilihan ini, lihat [Pengaturan untuk AWS CodeCommit](#).
- Konfigurasi AWS CLI jika Anda ingin menggunakan baris perintah atau terminal untuk semua operasi, termasuk membuat repositori.

Topik

- [Langkah 1: Buat CodeCommit repositori](#)
- [Langkah 2: Buat repo lokal](#)
- [Langkah 3: Buat commit pertama](#)
- [Langkah 4: Kirim commit pertama Anda](#)
- [Langkah 5: Bagikan CodeCommit repositori dan dorong dan tarik komit lain](#)
- [Langkah 6: Buat dan bagikan cabang](#)
- [Langkah 7: Buat dan bagikan tanda](#)
- [Langkah 8: Mengatur izin akses](#)
- [Langkah 9: Bersihkan](#)

Langkah 1: Buat CodeCommit repositori

Pada langkah ini, Anda menggunakan CodeCommit konsol untuk membuat repositori.

Anda dapat melewati langkah ini jika Anda sudah memiliki CodeCommit repositori yang ingin Anda gunakan.

Note

Tergantung pada penggunaan Anda, Anda mungkin dikenakan biaya untuk membuat atau mengakses suatu repositori. Untuk informasi selengkapnya, lihat [Harga](#) di halaman informasi CodeCommit produk.

Untuk membuat CodeCommit repositori

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Gunakan pemilih wilayah untuk memilih Wilayah AWS di mana Anda ingin membuat repositori. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Pada halaman Repositori, pilih Buat repositori.
4. Pada halaman Buat repositori, dalam Nama Repositori, ketikkan nama untuk repositori Anda, (misalnya **MyDemoRepo**).

Note

Nama repositori peka huruf besar kecil dan tidak boleh lebih dari 100 karakter. Untuk informasi selengkapnya, lihat [Batasan-batasan](#).

5. (Optional) Dalam Deskripsi, masukkan deskripsi (misalnya, **My demonstration repository**). Ini dapat membantu Anda dan pengguna lain mengidentifikasi tujuan repositori.
6. (Opsional) Pilih Tambahkan tanda untuk menambahkan satu atau beberapa tanda repositori (label atribut khusus yang membantu Anda mengatur dan mengelola sumber daya AWS Anda) ke repositori Anda. Untuk informasi selengkapnya, lihat [Menandai repositori di AWS CodeCommit](#).
7. (Opsional) Perluas konfigurasi tambahan untuk menentukan apakah akan menggunakan default Kunci yang dikelola AWS atau kunci yang dikelola pelanggan Anda sendiri untuk mengenkripsi dan mendekripsi data dalam repositori ini. Jika Anda memilih untuk menggunakan kunci yang dikelola pelanggan Anda sendiri, Anda harus memastikan bahwa itu tersedia di Wilayah AWS tempat Anda membuat repositori, dan bahwa kuncinya aktif. Untuk informasi selengkapnya, lihat [AWS Key Management Service dan enkripsi untuk repositori AWS CodeCommit](#).
8. (Opsional) Pilih Aktifkan Amazon CodeGuru Reviewer untuk Java dan Python jika repositori ini akan berisi kode Java atau Python, dan Anda ingin Reviewer menganalisis kode itu. CodeGuru CodeGuru Reviewer menggunakan beberapa model pembelajaran mesin untuk menemukan

cacat kode dan secara otomatis menyarankan perbaikan dan perbaikan dalam permintaan tarik. Untuk informasi selengkapnya, lihat Panduan Pengguna Amazon CodeGuru Reviewer.

9. Pilih Buat.

Note

Langkah-langkah yang tersisa dalam tutorial ini digunakan MyDemoRepo untuk nama CodeCommit repositori. Jika Anda memilih nama yang berbeda, pastikan untuk menggunakannya di seluruh tutorial ini.

Untuk informasi lebih lanjut tentang membuat repositori, termasuk cara membuat repositori dari terminal atau baris perintah, lihat [Buatlah sebuah repositori](#).

Langkah 2: Buat repo lokal

Pada langkah ini, Anda mengatur repo lokal pada mesin lokal Anda untuk terhubung ke repositori Anda. Untuk melakukannya, Anda memilih direktori pada mesin lokal Anda yang mewakili repo lokal. Anda menggunakan Git untuk mengkloning dan menginisialisasi salinan CodeCommit repositori kosong Anda di dalam direktori itu. Kemudian Anda menentukan nama pengguna Git dan alamat email yang digunakan untuk membubuhi keterangan commit Anda.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Dalam pemilih wilayah, pilih Wilayah AWS di mana repositori dibuat. Repositori adalah khusus untuk sebuah Wilayah AWS. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Temukan repositori yang ingin Anda sambungkan dari daftar dan pilih. Pilih URL Kloning, lalu pilih protokol yang ingin Anda gunakan saat mengkloning atau menghubungkan ke repositori. Langkah ini akan menyalin URL klon tersebut.
 - Salin URL HTTPS jika Anda menggunakan kredensial Git dengan pengguna IAM Anda atau pembantu kredensial yang disertakan dengan AWS CLI.
 - Salin URL HTTPS (GRC) jika Anda menggunakan perintah `git-remote-codecommit` di komputer lokal Anda.
 - Salin URL SSH jika Anda menggunakan pasangan kunci publik/swasta SSH dengan pengguna IAM Anda.

Note

Jika Anda melihat halaman Selamat Datang bukan daftar repositori, tidak ada repositori yang terkait dengan akun AWS Anda di Wilayah AWS tempat Anda masuk. Untuk membuat repositori, lihat [the section called “Buatlah sebuah repositori”](#) atau ikuti langkah-langkah di tutorial [Memulai dengan Git dan CodeCommit](#).

- (Opsional) Kami sarankan Anda mengonfigurasi klien Git lokal Anda untuk menggunakan **main** sebagai nama untuk cabang default untuk repositori Anda. Ini adalah nama yang digunakan untuk cabang default dalam contoh dalam panduan ini. Ini juga merupakan nama cabang default yang sama yang CodeCommit digunakan jika Anda membuat komit pertama Anda di konsol. Jalankan perintah berikut untuk mengkonfigurasi nama cabang default secara global untuk sistem Anda:

```
git config --global init.defaultBranch main
```

Jika Anda lebih suka menggunakan nama cabang default yang berbeda untuk semua repositori Anda, ganti **main** dengan nama pilihan Anda. Tutorial ini mengasumsikan bahwa cabang default Anda bernama utama.

Jika Anda ingin menggunakan nama cabang default yang berbeda untuk repositori yang berbeda, Anda dapat mengatur atribut ini secara lokal (--local) bukan secara global (--global).

- Di terminal atau command prompt, kloning repositori dengan perintah git clone dan berikan URL klon yang Anda salin di langkah 3. URL klon Anda tergantung pada protokol dan konfigurasi yang Anda gunakan. Misalnya, jika Anda menggunakan HTTPS dengan kredensi Git untuk mengkloning repositori bernama *MyDemoRepo* di Wilayah AS Timur (Ohio):

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Jika Anda menggunakan HTTPS dengan git-remote-codecommit:

```
git clone codecommit://MyDemoRepo my-demo-repo
```

Jika Anda menggunakan SSH:

```
git clone ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Note

Jika Anda melihat kesalahan saat mencoba mengkloning repositori, Anda mungkin belum menyelesaikan pengaturan yang diperlukan untuk komputer lokal Anda. Untuk informasi selengkapnya, lihat [Pengaturan untuk AWS CodeCommit](#).

Langkah 3: Buat commit pertama

Pada langkah ini, Anda membuat commit pertama Anda di repo lokal Anda. Untuk melakukan ini, Anda membuat dua contoh file di repo lokal Anda. Anda menggunakan Git untuk tahap perubahan, dan kemudian commit perubahan untuk, repo lokal Anda.

1. Gunakan editor teks untuk membuat dua file teks contoh berikut di direktori Anda. Nama file `cat.txt` dan `dog.txt`:

```
cat.txt
-----
The domestic cat (Felis catus or Felis silvestris catus) is a small, usually furry, domesticated, and carnivorous mammal.
```

```
dog.txt
-----
The domestic dog (Canis lupus familiaris) is a canid that is known as man's best friend.
```

2. Jalankan git config untuk menambahkan nama pengguna dan alamat email yang diwakili oleh placeholder `your-user-name` dan `your-email-address` ke repo lokal Anda. Ini memudahkan untuk mengidentifikasi commit yang Anda buat:

```
git config --local user.name "your-user-name"
git config --local user.email your-email-address
```


3. Jika Anda tidak menetapkan nama cabang default Anda secara global ketika Anda membuat repo lokal, jalankan perintah berikut untuk menetapkan nama cabang default ke **main**:

```
git config --local init.defaultBranch main
```

4. Jalankan git add untuk melancarkan perubahan:

```
git add cat.txt dog.txt
```

5. Jalankan git commit untuk melaksanakan perubahan:

```
git commit -m "Added cat.txt and dog.txt"
```

Tip

Untuk melihat detail tentang commit yang baru saja Anda buat, jalankan git log.

Langkah 4: Kirim commit pertama Anda

Pada langkah ini, Anda mendorong komit dari repo lokal ke CodeCommit repositori Anda.

Jalankan git push untuk mendorong komit Anda melalui nama jarak jauh default yang digunakan Git untuk CodeCommit repositori (`origin`) Anda, dari cabang default di repo lokal Anda (`:`): `main`

```
git push -u origin main
```

Tip

Setelah Anda mendorong file ke CodeCommit repositori Anda, Anda dapat menggunakan CodeCommit konsol untuk melihat konten. Untuk informasi selengkapnya, lihat [Menelusuri file dalam sebuah repositori](#).

Langkah 5: Bagikan CodeCommit repositori dan dorong dan tarik komit lain

Pada langkah ini, Anda berbagi informasi tentang CodeCommit repositori dengan sesama anggota tim. Anggota tim menggunakan informasi ini untuk mendapatkan salinan lokal, membuat beberapa

perubahan padanya, dan kemudian mendorong salinan lokal yang dimodifikasi ke CodeCommit repositori Anda. Anda kemudian menarik perubahan dari CodeCommit repositori ke repo lokal Anda.

Dalam tutorial ini, Anda mensimulasikan sesama pengguna dengan memerintahkan Git membuat direktori terpisah dari yang Anda buat di [Langkah 2](#). (Biasanya, direktori ini ada pada mesin yang berbeda.) Direktori baru ini adalah salinan dari CodeCommit repositori Anda. Setiap perubahan yang Anda buat ke direktori yang ada atau direktori baru ini dibuat secara independen. Satu-satunya cara untuk mengidentifikasi perubahan pada direktori ini adalah dengan menarik dari CodeCommit repositori.

Meskipun mereka berada di mesin lokal yang sama, kami menyebut direktori yang ada sebagai Repo lokal dan direktori baru sebagai repo bersama.

Dari direktori baru, Anda mendapatkan salinan terpisah dari CodeCommit repositori. Anda kemudian menambahkan file contoh baru, mengkomit perubahan ke repo bersama, dan kemudian mendorong komit dari repo bersama ke repositori Anda CodeCommit .

Terakhir, Anda menarik perubahan dari repositori Anda ke repo lokal Anda dan kemudian menjelajahi untuk melihat perubahan yang dilakukan oleh pengguna lain.

1. Beralih ke /tmp atau direktori c:\temp.
2. Jalankan git clone untuk menarik salinan repositori ke repo bersama:

Untuk HTTPS:

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
shared-demo-repo
```

Untuk HTTPS dengan git-remote-codecommit:

```
git clone codecommit://MyDemoRepo shared-demo-repo
```

Untuk SSH:

```
git clone ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo shared-
demo-repo
```

Note

Ketika Anda mengkloning SSH pada sistem operasi Windows, Anda mungkin perlu menambahkan ID kunci SSH ke string koneksi sebagai berikut:

```
git clone ssh://Your-SSH-Key-ID@git-codecommit.us-east-2.amazonaws.com/v1/
repos/MyDemoRepo my-demo-repo
```

Untuk informasi selengkapnya, lihat [Untuk koneksi SSH pada Windows](#).

Dalam perintah ini, MyDemoRepo adalah nama CodeCommit repositori Anda. shared-demo-repo adalah nama direktori yang dibuat Git di /tmp direktori atau c:\temp direktori. Setelah Git membuat direktori, Git menarik salinan repositori CodeCommit Anda ke direktori shared-demo-repo.

3. Beralih ke direktori shared-demo-repo:

```
(For Linux, macOS, or Unix) cd /tmp/shared-demo-repo
(For Windows) cd c:\temp\shared-demo-repo
```

4. Jalankan git config untuk menambahkan nama pengguna dan alamat email lain yang diwakili oleh placeholder *other-user-name* dan *other-email-address*. Ini memudahkan untuk mengidentifikasi commit yang dibuat pengguna lain:

```
git config --local user.name "other-user-name"
git config --local user.email other-email-address
```

5. Gunakan editor teks untuk membuat file teks contoh berikut di direktori shared-demo-repo. Namai file horse.txt:

```
horse.txt
-----
The horse (Equus ferus caballus) is one of two extant subspecies of Equus ferus.
```

6. Jalankan git add untuk melancarkan perubahan ke repo bersama:

```
git add horse.txt
```

7. Jalankan git commit untuk melaksanakan perubahan ke repo bersama:

```
git commit -m "Added horse.txt"
```

8. Jalankan git push untuk mendorong komit awal Anda melalui nama jarak jauh default yang digunakan Git untuk CodeCommit repositori (`origin`) Anda, dari cabang default di repo lokal Anda (`:`): `main`

```
git push -u origin main
```

9. Beralih ke repo lokal Anda dan jalankan git pull untuk menarik ke repo lokal Anda komit repo bersama yang dibuat ke repositori. CodeCommit Kemudian jalankan git log untuk melihat commit yang dimulai dari repo bersama.

Langkah 6: Buat dan bagikan cabang

Pada langkah ini, Anda membuat cabang di repo lokal Anda, membuat beberapa perubahan, dan kemudian mendorong cabang ke CodeCommit repositori Anda. Anda kemudian menarik cabang ke repo bersama dari CodeCommit repositori Anda.

Cabang memungkinkan Anda untuk secara independen mengembangkan versi yang berbeda dari isi repositori (misalnya, untuk bekerja pada fitur perangkat lunak baru tanpa mempengaruhi pekerjaan anggota tim Anda). Ketika fitur tersebut stabil, Anda menggabungkan cabang ke cabang yang lebih stabil dari perangkat lunak.

Anda menggunakan Git untuk membuat cabang dan kemudian mengarahkannya ke commit pertama yang Anda buat. Anda menggunakan Git untuk mendorong cabang ke CodeCommit repositori. Anda kemudian beralih ke repo bersama Anda dan menggunakan Git untuk menarik cabang baru ke repo lokal bersama Anda dan menjelajahi cabang.

1. Dari repo lokal Anda, jalankan git checkout, menentukan nama cabang (misalnya, `MyNewBranch`) dan ID dari commit pertama yang Anda buat di repo lokal.

Jika Anda tidak mengetahui ID commit, jalankan git log Untuk mendapatkannya. Pastikan commit memiliki nama pengguna dan alamat email Anda, bukan nama pengguna dan alamat email dari pengguna lain. Ini untuk mensimulasikan bahwa itu `main` adalah versi stabil dari CodeCommit repositori dan `MyNewBranch` cabang untuk beberapa fitur baru yang relatif tidak stabil:

```
git checkout -b MyNewBranch commit-ID
```

2. Jalankan git push untuk mengirim cabang baru dari repo lokal ke CodeCommit repositori:

```
git push origin MyNewBranch
```

3. Sekarang, tarik cabang ke repo bersama dan periksa hasil Anda:

1. Beralih ke direktori repo bersama (shared-demo-repo).
2. Tarik cabang baru (git fetch origin).
3. Konfirmasikan bahwa cabang telah ditarik (git branch --all menampilkan daftar semua cabang untuk repositori).
4. Beralih ke cabang baru (git checkout MyNewBranch).
5. Konfirmasikan bahwa Anda telah beralih ke cabang MyNewBranch dengan menjalankan git status atau git branch. Output menunjukkan cabang tempat Anda berada. Dalam hal ini, seharusnya MyNewBranch.
6. Tampilkan daftar commit di cabang (git log).

Berikut adalah daftar perintah Git untuk memanggil:

```
git fetch origin  
git branch --all  
git checkout MyNewBranch  
git branch or git status  
git log
```

4. Beralih kembali ke cabang main dan lihat daftar commit. Perintah Git akan terlihat seperti ini:

```
git checkout main  
git log
```

5. Beralih ke cabang main di repo lokal Anda. Anda dapat menjalankan git status atau git branch. Output menunjukkan cabang tempat Anda berada. Dalam hal ini, seharusnya main. Perintah Git akan terlihat seperti ini:

```
git checkout main
```

```
git branch or git status
```

Langkah 7: Buat dan bagikan tanda

Pada langkah ini, Anda membuat dua tag di repo lokal Anda, mengaitkan tag dengan komit, dan kemudian mendorong tag ke repositori Anda CodeCommit . Anda kemudian menarik perubahan dari CodeCommit repositori ke repo bersama.

Tanda digunakan untuk memberikan nama yang dapat dibaca manusia untuk commit (atau cabang atau bahkan tanda lain). Anda akan melakukan ini, misalnya, jika Anda ingin menandai commit sebagai v2.1. Sebuah commit, cabang, atau tanda dapat memiliki sejumlah tanda yang terkait dengannya, tetapi tanda individu dapat dikaitkan dengan hanya satu commit, cabang, atau tanda. Dalam tutorial ini, Anda menandai satu commit sebagai `releas`e dan satu commit sebagai `beta`.

Anda menggunakan Git untuk membuat tanda, mengarahkan tanda `releas`e ke commit pertama yang Anda buat dan tanda `beta` ke commit yang dibuat oleh pengguna lain. Anda kemudian menggunakan Git untuk mendorong tag ke CodeCommit repositori. Kemudian Anda beralih ke repo bersama Anda dan menggunakan Git untuk menarik tanda ke repo lokal Anda bersama dan menjelajahi tanda.

1. Dari repo lokal Anda, jalankan `git tag`, menentukan nama tanda baru (`releas`e) dan ID dari commit pertama yang Anda buat di repo lokal.

Jika Anda tidak mengetahui ID commit, jalankan `git log` Untuk mendapatkannya. Pastikan commit memiliki nama pengguna dan alamat email Anda, bukan nama pengguna dan alamat email dari pengguna lain. Ini untuk mensimulasikan bahwa komit Anda adalah versi stabil dari CodeCommit repositori:

```
git tag release commit-ID
```

Jalankan `git tag` lagi untuk menandai commit dari pengguna lain dengan tanda `beta`. Hal ini untuk mensimulasikan bahwa commit adalah untuk beberapa fitur baru yang relatif tidak stabil:

```
git tag beta commit-ID
```

2. Jalankan `git push --tags` untuk mengirim tag ke CodeCommit repositori.
3. Sekarang tarik tanda ke repo bersama dan periksa hasil Anda:

1. Beralih ke direktori repo bersama (shared-demo-repo).
2. Tarik tanda baru (git fetch origin).
3. Konfirmasikan bahwa tanda telah ditarik (git tag menampilkan daftar tanda untuk repositori).
4. Tampilkan informasi tentang setiap tanda (git log release dan git log beta).

Berikut adalah daftar perintah Git untuk memanggil:

```
git fetch origin
git tag
git log release
git log beta
```

4. Cobalah ini di repo lokal, juga:

```
git log release
git log beta
```

Langkah 8: Mengatur izin akses

Pada langkah ini, Anda memberikan izin pengguna untuk menyinkronkan repo bersama dengan repositori. CodeCommit ini adalah langkah opsional. Disarankan untuk pengguna yang tertarik untuk belajar tentang cara mengontrol akses ke CodeCommit repositori ketika pengguna menggunakan kredensi Git atau pasangan kunci SSH digunakan dengan pengguna IAM untuk akses ke repositori. CodeCommit

Note

Jika Anda menggunakan akses federasi, kredensi sementara, atau penyedia identitas web seperti IAM Identity Center, siapkan pengguna, akses, dan izin untuk penyedia identitas Anda, lalu gunakan. `git-remote-codecommit` Untuk informasi selengkapnya, silakan lihat [Langkah-langkah penyiapan untuk koneksi HTTPS ke AWS CodeCommit bersama git-remote-codecommit](#) dan [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#).

Untuk melakukan ini, Anda menggunakan konsol IAM untuk membuat pengguna, yang, secara default, tidak memiliki izin untuk menyinkronkan repo bersama dengan repositori. CodeCommit Anda dapat menjalankan git pull untuk memverifikasi ini. Jika pengguna baru tidak memiliki izin untuk menyinkronkan, perintah tersebut tidak akan bekerja. Kemudian Anda kembali ke konsol IAM dan menerapkan kebijakan yang memungkinkan pengguna untuk menggunakan git pull. Sekali lagi, Anda dapat menjalankan git pull untuk memverifikasi ini.

Langkah ini ditulis dengan asumsi Anda memiliki izin untuk membuat pengguna IAM di akun Amazon Web Services Anda. Jika Anda tidak memiliki izin ini, Anda tidak dapat melakukan prosedur dalam langkah ini. Lanjutkan ke langkah [Langkah 9: Bersihkan](#) untuk membersihkan sumber daya yang Anda gunakan untuk tutorial Anda.

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.

Pastikan untuk masuk dengan nama pengguna dan kata sandi yang sama dengan yang Anda gunakan di [Pengaturan](#).

2. Di panel navigasi, pilih Pengguna, lalu pilih Buat Pengguna Baru.
3. Di kotak Masukkan Nama Pengguna pertama, masukkan contoh nama pengguna (misalnya, **JaneDoe-CodeCommit**). Pilih kotak Hasilkan sebuah access key untuk setiap pengguna, dan kemudian pilih Buat.
4. Pilih Tampilkan Kredensial Keamanan Pengguna. Catat access key ID dan secret access key atau pilih Unduh Kredensial.
5. Ikuti instruksi di [Untuk pengguna HTTPS menggunakan kredensial Git](#) untuk menghasilkan dan menyediakan kredensial IAM pengguna.

Jika Anda ingin menggunakan SSH, ikuti petunjuk di [SSH dan Linux, macOS, atau Unix: Siapkan kunci publik dan privat untuk Git dan CodeCommit](#) atau [Langkah 3: Siapkan kunci publik dan privat untuk Git dan CodeCommit](#) untuk mengatur pengguna dengan kunci publik dan privat.

6. Jalankan git pull. Kesalahan berikut akan muncul:

Untuk HTTPS:

```
fatal: unable to access 'https://git-codecommit.us-east-2.amazonaws.com/v1/repos/repository-name/' : The requested URL returned error: 403.
```

Untuk SSH:


```
fatal: unable to access 'ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/repository-name': The requested URL returned error: 403.
```

Kesalahan muncul karena pengguna baru tidak memiliki izin untuk menyinkronkan repo bersama dengan repositori. CodeCommit

7. Mengembalikan ke konsol IAM. Pada panel navigasi, pilih Kebijakan, lalu pilih Buat kebijakan. (Jika tombol Memulai muncul, pilihlah, lalu pilih Buat Kebijakan.)
8. Di sebelah Buat Kebijakan Anda Sendiri, pilih Pilih.
9. Di kotak Nama kebijakan, masukkan nama (misalnya, **CodeCommitAccess-GettingStarted**).
10. Di kotak Dokumen Kebijakan, masukkan yang berikut, yang memungkinkan pengguna IAM untuk menarik dari repositori yang terkait dengan pengguna IAM:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "codecommit:GitPull"
      ],
      "Resource": "*"
    }
  ]
}
```

Tip

Jika Anda ingin pengguna IAM untuk dapat mengirim commit untuk setiap repositori yang terkait dengan pengguna IAM, masukkan ini sebagai gantinya:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
```

```
        "codecommit:GitPull",
        "codecommit:GitPush"
    ],
    "Resource": "*"
}
]
```

Untuk informasi tentang CodeCommit tindakan lain dan izin sumber daya yang dapat Anda berikan kepada pengguna, lihat [Kontrol autentikasi dan akses untuk AWS CodeCommit](#).

11. Di panel navigasi, pilih Users (Pengguna).
12. Pilih contoh nama pengguna (misalnya, **JaneDoe-CodeCommit**) yang ingin Anda lampirkan kebijakan.
13. Pilih tab Izin.
14. Di Kebijakan Terkelola, pilih Lampirkan Kebijakan.
15. Pilih kebijakan **CodeCommitAccess-GettingStarted** yang baru saja Anda buat, dan kemudian pilih Lampirkan kebijakan.
16. Jalankan git pull. Kali ini perintah harus bekerja dan pesan Already up-to-date akan muncul.
17. Jika Anda menggunakan HTTPS, beralih ke kredensial Git semula Anda atau, jika menggunakan git-remote-codecommit, profil Anda yang biasa. Untuk informasi lebih lanjut, lihat petunjuk di [Pengaturan untuk pengguna HTTPS menggunakan kredensial Git](#) atau [Langkah-langkah penyiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit](#).

Jika Anda menggunakan SSH, beralih ke tombol asli Anda. Untuk informasi selengkapnya, lihat [SSH dan Linux, macOS, atau Unix: Siapkan kunci publik dan privat untuk Git danCodeCommit](#) atau [Langkah 3: Siapkan kunci publik dan privat untuk Git dan CodeCommit](#).

Anda telah mencapai akhir tutorial ini.

Langkah 9: Bersihkan

Pada langkah ini, Anda menghapus CodeCommit repositori yang Anda gunakan dalam tutorial ini, sehingga Anda tidak akan terus dikenakan biaya untuk ruang penyimpanan.

Anda juga menghapus repo lokal dan repo bersama di mesin lokal Anda karena mereka tidak akan diperlukan setelah Anda menghapus repositori. CodeCommit

Important

Setelah Anda menghapus repositori ini, Anda tidak lagi dapat mengkloningnya ke repo lokal atau repo bersama. Anda juga tidak akan dapat menarik data darinya, atau mengirim data ke sana, dari repo lokal atau repo bersama. Tindakan ini tidak dapat dibatalkan.

Untuk menghapus CodeCommit repositori (konsol)

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Pada halaman Dashboard, dalam daftar repositori, pilih MyDemoRepo
3. Pada panel navigasi, silakan pilih Pengaturan.
4. Pada halaman Pengaturan, di Hapus repositori, pilih Hapus repositori.
5. Dalam kotak di sebelah Ketik nama repositori untuk mengonfirmasi penghapusan, masukkan **MyDemoRepo**, lalu pilih Hapus.

Untuk menghapus CodeCommit repositori () AWS CLI

Jalankan perintah [delete-repository](#):

```
aws codecommit delete-repository --repository-name MyDemoRepo
```

Untuk menghapus repo lokal dan repo bersama

Untuk Linux, macOS, atau Unix:

```
cd /tmp
rm -rf /tmp/my-demo-repo
rm -rf /tmp/shared-demo-repo
```

Untuk Windows:

```
cd c:\temp
rd /s /q c:\temp\my-demo-repo
```

```
rd /s /q c:\temp\shared-demo-repo
```

Integrasi produk dan layanan dengan AWS CodeCommit

Secara default CodeCommit, terintegrasi dengan sejumlah AWS layanan. Anda juga dapat menggunakan CodeCommit produk dan layanan di luar AWS. Informasi berikut dapat membantu Anda mengonfigurasi CodeCommit untuk berintegrasi dengan produk dan layanan yang Anda gunakan.

Note

Anda dapat secara otomatis membangun dan menerapkan komit ke CodeCommit repositori dengan mengintegrasikan dengan CodePipeline. Untuk mempelajari lebih lanjut, ikuti langkah-langkah di [Panduan AWS DevOps Memulai](#).

Topik

- [Integrasi dengan AWS layanan lain](#)
- [Contoh integrasi dari komunitas](#)

Integrasi dengan AWS layanan lain

CodeCommit terintegrasi dengan AWS layanan berikut:

AWS Amplify

[AWS Amplify](#) memudahkan membuat, mengkonfigurasi, dan menerapkan aplikasi seluler terukur yang didukung oleh AWS. Amplify ketentuan mulus dan mengelola backend ponsel Anda dan menyediakan kerangka kerja sederhana untuk dengan mudah mengintegrasikan backend Anda dengan frontend iOS, Android, Web, dan React Native. Amplify juga mengotomatiskan proses rilis aplikasi dari kedua frontend dan backend Anda, yang memungkinkan Anda untuk memberikan fitur lebih cepat.

Anda dapat menghubungkan CodeCommit repository Anda di konsol Amplify. Setelah Anda mengotorisasi konsol Amplify, Amplify mengambil token akses dari penyedia repository, tetapi token tersebut tidak menyimpan token di server. AWS Amplify mengakses repository Anda menggunakan kunci deploy yang terinstal di repository tertentu saja.

Pelajari selengkapnya:

- [AWS Amplify Panduan Pengguna](#)
- [Memulai](#)

AWS Cloud9

[AWS Cloud9](#) berisi koleksi alat yang Anda gunakan untuk kode, membangun, menjalankan, menguji, debug, dan melepaskan perangkat lunak di cloud. Kumpulan alat ini disebut sebagai lingkungan pengembangan AWS Cloud9 terintegrasi, atau IDE.

Anda mengakses AWS Cloud9 IDE melalui browser web. IDE menawarkan pengalaman pengeditan kode yang kaya dengan dukungan untuk beberapa bahasa pemrograman dan runtime debugger, dan terminal built-in.

Pelajari selengkapnya:

- [Panduan Pengguna AWS Cloud9](#)
- [AWS CodeCommit Sampel untuk AWS Cloud9](#)
- [Integrasi AWS Cloud9 dengan AWS CodeCommit](#)

AWS CloudFormation

[AWS CloudFormation](#) adalah layanan yang membantu Anda memodelkan dan mengatur AWS sumber daya Anda sehingga Anda dapat menghabiskan lebih sedikit waktu mengelola sumber daya tersebut dan lebih banyak waktu untuk berfokus pada aplikasi Anda. Anda membuat templat yang menjelaskan sumber daya, termasuk CodeCommit repositori, dan AWS CloudFormation menangani penyediaan dan konfigurasi sumber daya tersebut untuk Anda.

Pelajari selengkapnya:

- [AWS CodeCommit Halaman sumber daya repositori](#)

AWS CloudTrail

[CloudTrail](#) menangkap panggilan AWS API dan peristiwa terkait yang dibuat oleh atau atas nama akun Amazon Web Services dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Anda dapat mengonfigurasi CloudTrail untuk menangkap panggilan API dari AWS CodeCommit konsol, CodeCommit perintah dari AWS CLI, klien Git lokal, dan dari CodeCommit API.

Pelajari selengkapnya:

- [Mencatat panggilan API AWS CodeCommit dengan AWS CloudTrail](#)

CloudWatch Acara Amazon

[CloudWatch Peristiwa](#) memberikan aliran peristiwa sistem yang mendekati waktu nyata yang menggambarkan perubahan AWS sumber daya. Dengan menggunakan aturan sederhana yang dapat Anda atur dengan cepat, Anda dapat mencocokkan acara dan mengarahkannya ke satu atau lebih fungsi atau aliran target. CloudWatch Peristiwa menjadi sadar akan perubahan operasional saat terjadi. CloudWatch Peristiwa merespons perubahan operasional ini dan mengambil tindakan seperlunya, dengan mengirim pesan untuk merespons lingkungan, mengaktifkan fungsi, membuat perubahan, dan menangkap informasi negara.

Anda dapat mengonfigurasi CloudWatch Peristiwa untuk memantau CodeCommit repositori dan merespons peristiwa repositori dengan menargetkan aliran, fungsi, tugas, atau proses lain di layanan lain, seperti Amazon Simple Queue Service AWS, Amazon Kinesis, dan banyak lagi. AWS Lambda

Pelajari selengkapnya:

- [CloudWatch Panduan Pengguna Acara](#)
- [AWS CodeCommit Acara](#)
- Posting blog: [Membangun AWS CodeCommit Alur Kerja Tanpa Server menggunakan Amazon CloudWatch Events dan JGit](#)

AWS CodeBuild

[CodeBuild](#) adalah layanan build yang dikelola sepenuhnya di cloud yang mengkompilasi kode sumber Anda, menjalankan pengujian unit, dan menghasilkan artefak yang siap diterapkan. Anda dapat menyimpan kode sumber yang akan dibangun dan spesifikasi build dalam CodeCommit repositori. Anda dapat menggunakan CodeBuild langsung dengan CodeCommit, atau Anda dapat menggabungkan keduanya CodeBuild dan CodeCommit dalam pipa pengiriman berkelanjutan dengan CodePipeline.

Pelajari selengkapnya:

- [Rencanakan Bangun](#)
- [Buat Proyek Bangun](#)
- [Gunakan CodePipeline dengan AWS CodeBuild untuk Menjalankan Builds](#)

CodeGuru Peninjau Amazon

Amazon CodeGuru Reviewer adalah layanan peninjauan kode otomatis yang menggunakan analisis program dan pembelajaran mesin untuk mendeteksi masalah umum dan merekomendasikan perbaikan dalam kode Java atau Python Anda. Anda dapat mengaitkan repositori di akun Amazon Web Services Anda dengan CodeGuru Reviewer. Ketika Anda melakukannya, CodeGuru Reviewer akan membuat peran terkait layanan yang memungkinkan CodeGuru Reviewer menganalisis kode di semua permintaan tarik yang dibuat setelah asosiasi dibuat.

Pelajari selengkapnya:

- [Kaitkan atau pisahkan AWS CodeCommit repositori dengan Amazon Reviewer CodeGuru](#)
- [Panduan Pengguna Amazon CodeGuru Reviewer](#)

AWS CodePipeline

[CodePipeline](#) adalah layanan pengiriman berkelanjutan yang dapat Anda gunakan untuk memodelkan, memvisualisasikan, dan mengotomatiskan langkah-langkah yang diperlukan untuk merilis perangkat lunak Anda. Anda dapat mengonfigurasi CodePipeline untuk menggunakan CodeCommit repositori sebagai tindakan sumber dalam pipeline, dan mengotomatiskan pembuatan, pengujian, dan penerapan perubahan Anda.

Pelajari selengkapnya:

- [Panduan Pipa Sederhana dengan dan CodePipeline AWS CodeCommit](#)
- [Migrasi ke Amazon CloudWatch Events Change Detection for Pipelines dengan Repositori CodeCommit](#)
- [Metode Deteksi Perubahan yang Digunakan untuk Memulai Pipa Secara Otomatis](#)

AWS CodeStar

[AWS CodeStar](#) adalah layanan berbasis cloud untuk membuat, mengelola, dan bekerja dengan proyek pengembangan perangkat lunak di AWS. Anda dapat dengan cepat mengembangkan, membangun, dan menyebarkan aplikasi AWS dengan sebuah AWS CodeStar proyek. Sebuah AWS CodeStar proyek membuat dan mengintegrasikan AWS layanan untuk rantai alat pengembangan proyek Anda, termasuk CodeCommit repositori untuk proyek tersebut. AWS CodeStar juga memberikan izin kepada anggota tim untuk proyek itu. Izin ini diterapkan secara otomatis, termasuk izin untuk mengakses, membuat dan mengelola kredensi Git CodeCommit, dan banyak lagi.

Anda dapat mengonfigurasi repositori yang dibuat untuk AWS CodeStar proyek seperti halnya CodeCommit repositori lainnya dengan menggunakan AWS CodeCommit konsol, CodeCommit perintah dari, klien Git lokal, dan dari API. AWS CLI CodeCommit

Pelajari selengkapnya:

- [Bekerja dengan repositori](#)
- [Bekerja dengan AWS CodeStar Proyek](#)
- [Bekerja dengan AWS CodeStar Tim](#)

AWS Elastic Beanstalk

[Elastic](#) Beanstalk adalah layanan terkelola yang memudahkan untuk menyebarkan dan mengelola aplikasi AWS di cloud tanpa mengkhawatirkan infrastruktur yang menjalankan aplikasi tersebut. Anda dapat menggunakan antarmuka baris perintah Elastic Beanstalk (EB CLI) untuk menyebarkan aplikasi Anda langsung dari repositori baru atau yang sudah ada. CodeCommit

Pelajari selengkapnya:

- [Menggunakan EB CLI dengan AWS CodeCommit](#)
- [Menggunakan AWS CodeCommit Repositori yang Ada](#)
- [sumber kode eb \(perintah EB CLI\)](#)

AWS Key Management Service

[AWS KMS](#) adalah layanan terkelola yang memungkinkan Anda membuat dan mengendalikan kunci enkripsi yang digunakan untuk mengenkripsi data Anda. Secara default, CodeCommit menggunakan AWS KMS untuk mengenkripsi repositori.

Pelajari selengkapnya:

- [AWS KMS dan enkripsi](#)

AWS Lambda

[Lambda](#) – Memungkinkan Anda menjalankan kode tanpa penyediaan atau pengelolaan server. Anda dapat mengonfigurasi pemicu untuk CodeCommit repositori yang memanggil fungsi Lambda sebagai respons terhadap peristiwa repositori.

Pelajari selengkapnya:

- [Buatlah sebuah pemicu untuk sebuah fungsi Lambda](#)
- [AWS Lambda Panduan Pengembang](#)

Layanan Pemberitahuan Sederhana Amazon

[Amazon SNS](#) adalah layanan web yang memungkinkan aplikasi, pengguna akhir, dan perangkat untuk langsung mengirim dan menerima notifikasi dari cloud. Anda dapat mengonfigurasi pemicu untuk CodeCommit repositori yang mengirim notifikasi Amazon SNS sebagai respons terhadap peristiwa repositori. Anda juga dapat menggunakan notifikasi Amazon SNS untuk berintegrasi dengan layanan lain AWS. Misalnya, Anda dapat menggunakan notifikasi Amazon SNS untuk mengirim pesan ke antrian Amazon Simple Queue Service.

Pelajari selengkapnya:

- [Buat pemicu untuk topik Amazon SNS](#)
- [Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon](#)

Contoh integrasi dari komunitas

Bagian berikut menyediakan tautan ke unggahan blog, artikel, dan contoh yang disediakan komunitas.

Note

Tautan ini disediakan hanya untuk tujuan informasi, dan tidak boleh dianggap sebagai daftar lengkap atau dukungan atas isi contoh. AWS tidak bertanggung jawab atas konten atau keakuratan konten eksternal.

Topik

- [Unggahan blog](#)
- [Sampel Kode](#)

Unggahan blog

- [Mengintegrasikan SonarQube sebagai Penyetuju Permintaan Tarik pada AWS CodeCommit](#)

Pelajari cara membuat CodeCommit repositori yang memerlukan analisis SonarQube kualitas yang sukses sebelum permintaan tarik dapat digabungkan.

Diterbitkan 12 Desember 2019

- [Migrasi ke AWS CodeCommit, AWS CodePipeline,, dan AWS CodeBuild Dari GitLab](#)

Pelajari cara memigrasikan beberapa repositori ke AWS CodeCommit from GitLab dan menyiapkan pipeline CI/CD menggunakan dan. AWS CodePipeline AWS CodeBuild

Diterbitkan 22 November 2019

- [Menerapkan GitFlow Menggunakan AWS CodePipeline, AWS CodeCommit,, AWS CodeBuild, dan AWS CodeDeploy](#)

Pelajari cara menerapkan GitFlow menggunakan AWS CodePipeline, AWS CodeCommit, AWS CodeBuild, dan AWS CodeDeploy.

Diterbitkan 22 Februari 2019

- [Menggunakan Git AWS CodeCommit dengan Beberapa AWS Akun](#)

Pelajari cara mengelola konfigurasi Git Anda di beberapa akun Amazon Web Services.

Diterbitkan 12 Februari 2019

- [Memvalidasi Permintaan AWS CodeCommit Tarik dengan AWS CodeBuild dan AWS Lambda](#)

Pelajari cara memvalidasi permintaan tarik dengan AWS CodeCommit, AWS CodeBuild, and AWS Lambda. Dengan menjalankan tes terhadap perubahan yang diusulkan sebelum menggabungkan mereka ke cabang default, Anda dapat membantu memastikan tingkat tinggi kualitas dalam permintaan tarik, menangkap potensi masalah, dan meningkatkan kepercayaan dari developer dalam kaitannya dengan perubahan mereka.

Diterbitkan 11 Februari 2019

- [Menggunakan Identitas Federasi dengan AWS CodeCommit](#)

Pelajari cara mengakses repositori dalam AWS CodeCommit menggunakan identitas yang digunakan dalam bisnis Anda.

Diterbitkan 5 Oktober 2018

- [Menyempurnakan Akses ke Cabang di AWS CodeCommit](#)

Pelajari cara membatasi commit ke cabang repositori dengan membuat dan menerapkan kebijakan IAM yang menggunakan kunci konteks.

Diterbitkan 16 Mei 2018

- [Replikasi AWS CodeCommit Repositori Antar Wilayah Menggunakan Fargate AWS](#)

Pelajari cara mengatur replikasi berkelanjutan dari CodeCommit repositori dari satu AWS wilayah ke wilayah lain menggunakan arsitektur tanpa server.

Diterbitkan 11 April 2018

- [Mendistribusikan Infrastruktur Anda AWS OpsWorks for Chef Automate](#)

Pelajari cara menggunakan CodePipeline,, CodeCommit CodeBuild, dan AWS Lambda untuk memastikan bahwa buku masak dan konfigurasi lainnya digunakan secara konsisten di dua atau lebih Server Chef yang berada di satu atau lebih. Wilayah AWS

Diterbitkan 9 Maret 2018

- [Selai Kacang dan Cokelat: Azure Fungsi Pipa CI/CD dengan AWS CodeCommit](#)

Pelajari cara membuat pipeline CI/CD Azure Functions PowerShell berbasis tempat kode disimpan dalam repositori. CodeCommit

Diterbitkan 19 Februari 2018

- [Penerapan Berkelanjutan ke Kubernetes Menggunakan AWS CodePipeline,, AWS CodeBuild Amazon ECR AWS CodeCommit, dan AWS Lambda](#)

Pelajari cara menggunakan Kubernetes dan AWS bersama-sama membuat pipeline deployment berkelanjutan yang dikelola sepenuhnya untuk aplikasi berbasis container.

Diterbitkan pada 11 Januari 2018

- [Gunakan Permintaan AWS CodeCommit Tarik untuk Meminta Ulasan Kode dan Mendiskusikan Kode](#)

Pelajari cara menggunakan permintaan tarik untuk meninjau, mengomentari, dan secara interaktif mengulangi perubahan kode dalam repositori. CodeCommit

Diterbitkan 20 November 2017

- [Membangun AWS CodeCommit Alur Kerja Tanpa Server Menggunakan Amazon CloudWatch Events dan JGit](#)

Pelajari cara membuat aturan CloudWatch Acara yang memproses perubahan dalam repositori menggunakan peristiwa CodeCommit repositori dan tindakan target di layanan lain. AWS Contohnya termasuk AWS Lambda fungsi yang menerapkan kebijakan pesan komit Git pada komit, mereplikasi repositori, dan mencadangkan CodeCommit repositori ke Amazon S3. CodeCommit

Diterbitkan 3 Agustus 2017

- [Migrasi ke AWS CodeCommit](#)

Pelajari cara mendorong kode ke dua repositori sebagai bagian dari migrasi dari menggunakan repositori Git lain ke saat menggunakan. CodeCommit SourceTree

Diterbitkan 6 September 2016

- [Mengatur Pengujian Berkelanjutan dengan Appium,, Jenkins AWS CodeCommit, dan AWS Device Farm](#)

Pelajari cara membuat proses pengujian berkelanjutan untuk perangkat seluler menggunakan Appium, Jenkins CodeCommit, dan Device Farm.

Diterbitkan 2 Februari 2016

- [Mengggunakan AWS CodeCommit Repositori Git di Beberapa akun Amazon Web Services](#)

Pelajari cara mengkloning CodeCommit repositori Anda dan, dalam satu perintah, konfigurasi pembantu kredensial untuk menggunakan peran IAM tertentu untuk koneksi ke repositori itu.

Diterbitkan pada November 2015

- [Integrasi dan AWS OpsWorksAWS CodeCommit](#)

Pelajari cara AWS OpsWorks mengambil buku masak Aplikasi dan Chef secara otomatis. CodeCommit

Diterbitkan 25 Agustus 2015

- [Mengggunakan AWS CodeCommit dan GitHub Credential Helpers](#)

Pelajari cara mengonfigurasi file gitconfig Anda agar berfungsi dengan keduanya CodeCommit dan pembantu GitHub kredensial.

Diterbitkan September 2015

- [Mengggunakan AWS CodeCommit dari Eclipse](#)

Pelajari cara menggunakan alat eGit di Eclipse untuk bekerja dengannya. CodeCommit

Diterbitkan Agustus 2015

- [AWS CodeCommit dengan Kredensial Peran Amazon EC2](#)

Pelajari cara menggunakan profil instans untuk Amazon EC2 saat mengonfigurasi akses agen otomatis ke repositori. CodeCommit

Diterbitkan Juli 2015

- [Integrasi AWS CodeCommit dengan Jenkins](#)

Pelajari cara menggunakan CodeCommit dan Jenkins untuk mendukung dua skenario integrasi berkelanjutan (CI) sederhana.

Diterbitkan Juli 2015

- [Integrasi AWS CodeCommit dengan Review Board](#)

Pelajari cara mengintegrasikan CodeCommit ke dalam alur kerja pengembangan menggunakan sistem peninjauan kode [Review Board](#).

Diterbitkan Juli 2015

Sampel Kode

Berikut ini adalah contoh kode yang mungkin menarik bagi CodeCommit pengguna.

- [Mac OS X Script untuk Menghapus Cached Credentials secara berkala di OS X Certificate Store](#)

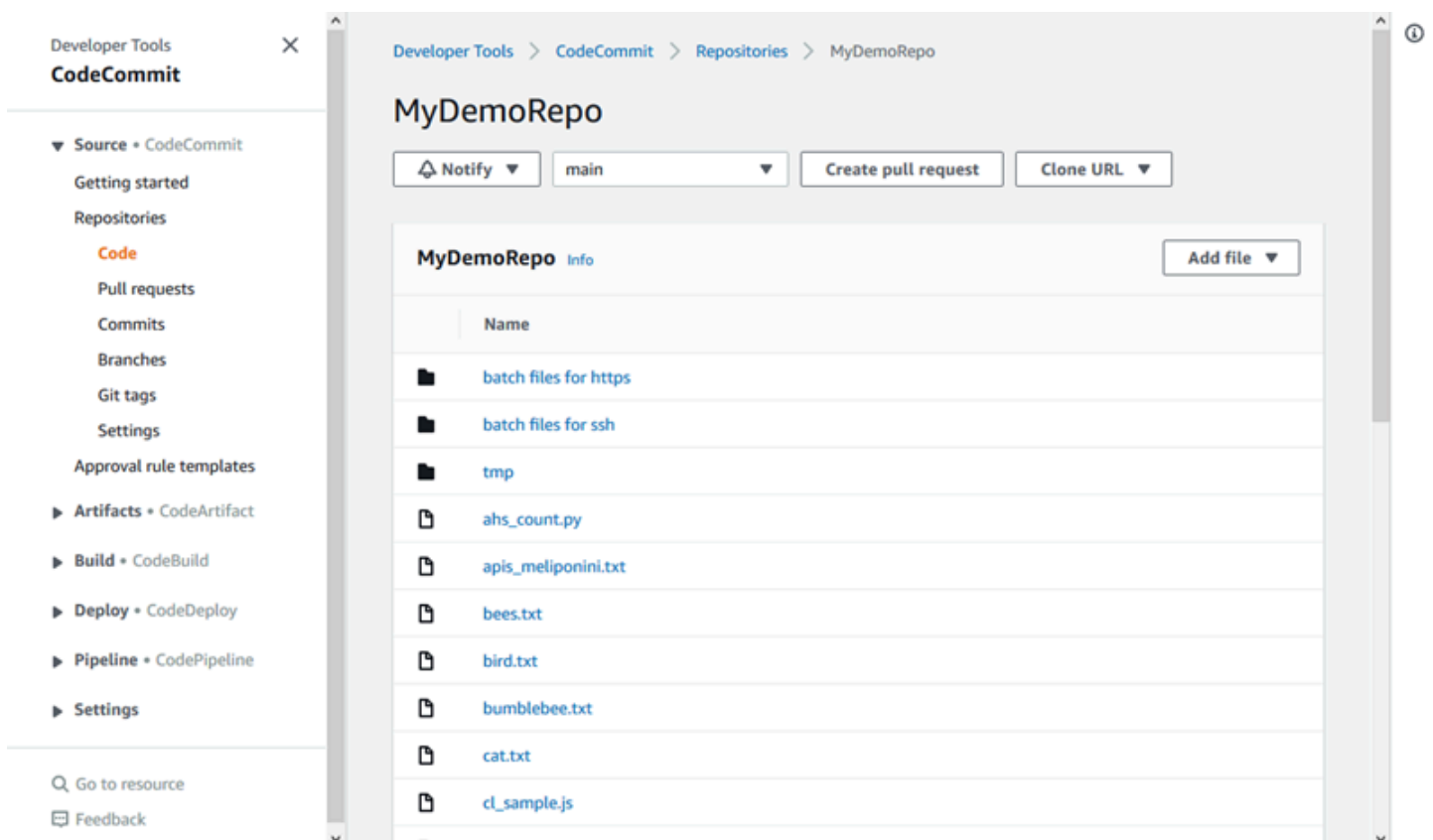
Jika Anda menggunakan pembantu kredenal untuk CodeCommit di Mac OS X, Anda mungkin terbiasa dengan masalah dengan kredenal yang di-cache. Skrip ini menunjukkan satu solusi.

Penulis: Nico Coetzea

Diterbitkan Februari 2016

Bekerja dengan repositori di AWS CodeCommit

Repositori adalah objek kontrol versi fundamental di CodeCommit. Di situlah Anda menyimpan kode dan file dengan aman untuk proyek Anda. Ini juga menyimpan riwayat proyek Anda, dari commit pertama hingga perubahan terbaru. Anda dapat berbagi repositori Anda dengan pengguna lain sehingga Anda dapat bekerja sama pada suatu proyek. Jika Anda menambahkan AWS tag ke repositori, Anda dapat mengatur notifikasi sehingga pengguna repositori menerima email tentang peristiwa (misalnya, pengguna lain yang mengomentari kode). Anda juga dapat mengubah pengaturan default untuk repositori, menelusuri isinya, dan lainnya. Anda dapat membuat pemicu untuk repositori Anda sehingga kode mendorong atau peristiwa-peristiwa lainnya memicu tindakan, seperti email atau kode fungsi. Anda bahkan dapat mengkonfigurasi repositori di komputer lokal Anda (repo lokal) untuk mendorong perubahan Anda ke lebih dari satu repositori.



Sebelum Anda dapat mendorong perubahan ke CodeCommit repositori, Anda harus mengonfigurasi pengguna IAM di akun Amazon Web Services Anda, atau mengatur akses untuk akses federasi atau kredensial sementara. Untuk informasi selengkapnya, lihat [Langkah 1: Konfigurasi awal untuk CodeCommit](#) dan [Langkah-langkah penyiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit](#).

Untuk informasi tentang bekerja dengan aspek lain dari repositori Anda di CodeCommit, lihat [Bekerja dengan file](#), [Bekerja dengan permintaan pull](#), [Bekerja dengan commit](#), [Bekerja dengan cabang](#), dan [Bekerja dengan preferensi pengguna](#). Untuk informasi tentang migrasi ke CodeCommit, lihat [Migrasi ke CodeCommit](#).

Topik

- [Buat AWS CodeCommit repositori](#)
- [Connect ke AWS CodeCommit repositori](#)
- [Bagikan AWS CodeCommit repositori](#)
- [Mengkonfigurasi notifikasi untuk peristiwa dalam sebuah repositori AWS CodeCommit](#)
- [Menandai repositori di AWS CodeCommit](#)
- [Mengelola pemicu untuk repositori AWS CodeCommit](#)
- [Kaitkan atau pisahkan AWS CodeCommit repositori dengan Amazon Reviewer CodeGuru](#)
- [Lihat detail CodeCommit repositori](#)
- [Ubah AWS CodeCommit pengaturan repositori](#)
- [Sinkronkan perubahan antara sebuah repo lokal dengan sebuah repositori AWS CodeCommit](#)
- [Dorong commit ke repositori Git tambahan](#)
- [Konfigurasi akses lintas akun ke AWS CodeCommit repositori menggunakan peran](#)
- [Hapus AWS CodeCommit repositori](#)

Buat AWS CodeCommit repositori

Gunakan AWS CodeCommit konsol atau AWS Command Line Interface (AWS CLI) untuk membuat CodeCommit repositori kosong. Untuk menambahkan tag ke repositori setelah Anda membuatnya, lihat [Tambahkan tag ke sebuah repositori](#).

Petunjuk ini berasumsi bahwa Anda telah menyelesaikan langkah-langkah di [Pengaturan](#).

Note

Tergantung pada penggunaan Anda, Anda mungkin dikenakan biaya untuk membuat atau mengakses suatu repositori. Untuk informasi selengkapnya, lihat [Harga](#) di halaman informasi CodeCommit produk.

Topik

- [Buatlah sebuah \(konsol\) repositori](#)
- [Buatlah sebuah repositori \(AWS CLI\)](#)

Buatlah sebuah (konsol) repositori

Untuk membuat CodeCommit repositori

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di pemilih wilayah, pilih Wilayah AWS tempat Anda ingin membuat repositori. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Pada halaman Repositori, pilih Buat repositori.
4. Pada halaman Buat repositori, di Nama repositori, masukkan nama untuk repositori tersebut.

Note

Nama repositori sensitif huruf besar dan kecil. Nama harus unik di Wilayah AWS untuk akun Amazon Web Services Anda.

5. (Opsional) Di Deskripsi, masukkan deskripsi untuk repositori tersebut. Ini dapat membantu Anda dan pengguna lain mengidentifikasi tujuan repositori tersebut.

Note

Bidang deskripsi menampilkan Markdown di konsol dan menerima semua karakter HTML dan karakter Unicode yang valid. Jika Anda adalah pengembang aplikasi yang menggunakan `BatchGetRepositories` API `GetRepository` atau dan Anda berencana untuk menampilkan bidang deskripsi repositori di browser web, lihat [Referensi CodeCommit API](#).

6. (Opsional) Pilih Tambahkan tag untuk menambahkan satu atau beberapa tag repositori (label atribut khusus yang membantu Anda mengatur dan mengelola AWS sumber daya) ke repositori Anda. Untuk informasi selengkapnya, lihat [Menandai repositori di AWS CodeCommit](#).
7. (Opsional) Perluas konfigurasi tambahan untuk menentukan apakah akan menggunakan default Kunci yang dikelola AWS atau kunci yang dikelola pelanggan Anda sendiri untuk mengenkripsi dan mendekripsi data dalam repositori ini. Jika Anda memilih untuk menggunakan kunci yang

dikelola pelanggan Anda sendiri, Anda harus memastikan bahwa itu tersedia di Wilayah AWS tempat Anda membuat repositori, dan bahwa kuncinya aktif. Untuk informasi selengkapnya, lihat [AWS Key Management Service dan enkripsi untuk repositori AWS CodeCommit](#).

8. (Opsional) Pilih Aktifkan Amazon CodeGuru Reviewer untuk Java dan Python jika repositori ini berisi kode Java atau Python, dan Anda ingin Reviewer menganalisisnya. CodeGuru CodeGuru Reviewer menggunakan beberapa model pembelajaran mesin untuk menemukan cacat kode dan menyarankan perbaikan dan perbaikan dalam permintaan tarik. Untuk informasi selengkapnya, lihat [Panduan Pengguna Amazon CodeGuru Reviewer](#).
9. Pilih Buat.

Setelah Anda membuat repositori, Anda dapat terhubung ke sana dan mulai menambahkan kode baik melalui CodeCommit konsol atau klien Git lokal, atau dengan mengintegrasikan CodeCommit repositori Anda dengan IDE favorit Anda. Untuk informasi selengkapnya, lihat [Pengaturan untuk AWS CodeCommit](#). Anda juga dapat menambahkan repositori Anda ke alur pengiriman berkelanjutan. Untuk informasi lebih lanjut, lihat [Panduan Alur Sederhana](#).

Untuk mendapatkan informasi tentang CodeCommit repositori baru, seperti URL yang akan digunakan saat mengkloning repositori, pilih nama repositori dari daftar, atau cukup pilih protokol koneksi yang ingin Anda gunakan di sebelah nama repositori.

Untuk berbagi repositori ini dengan orang lain, Anda harus mengirimkan tautan HTTPS atau SSH untuk digunakan untuk mengkloning repositori tersebut. Pastikan mereka memiliki izin yang diperlukan untuk mengakses repositori tersebut. Untuk informasi selengkapnya, lihat [Bagikan sebuah repositori](#) dan [Kontrol autentikasi dan akses untuk AWS CodeCommit](#).

Buatlah sebuah repositori (AWS CLI)

Anda dapat menggunakan AWS CLI untuk membuat CodeCommit repositori. Berbeda dengan konsol, Anda dapat menambahkan tag ke repositori jika Anda membuatnya menggunakan AWS CLI.

1. Pastikan bahwa Anda telah mengkonfigurasi AWS CLI dengan Wilayah AWS tempat repositori ada. Untuk memverifikasi Wilayah, jalankan perintah berikut pada baris perintah atau terminal dan tinjau informasi untuk nama wilayah default.

```
aws configure
```

Nama wilayah default harus cocok dengan repositori di. Wilayah AWS CodeCommit Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).

2. Jalankan perintah `create-repository`, dengan menentukan:

- Nama yang secara unik mengidentifikasi CodeCommit repositori (dengan opsi). `--repository-name`

Note

Nama ini harus unik di seluruh akun Amazon Web Services.

- Komentar opsional tentang CodeCommit repositori (dengan `--repository-description` opsi).
- Pasangan atau pasangan nilai kunci opsional untuk digunakan sebagai tag untuk CodeCommit repositori (dengan opsi). `--tags`
- Kunci yang dikelola pelanggan opsional untuk digunakan saat mengenkripsi dan mendekripsi repositori ini. Semua repositori dienkripsi dalam perjalanan dan diam menggunakan kunci masuk. AWS KMS Jika tidak ada kunci yang ditentukan, kunci AWS terkelola default `aws/codecommit` digunakan.

Misalnya, untuk membuat CodeCommit repositori bernama `MyDemoRepo` dengan deskripsi `"My demonstration repository"` dan tag dengan kunci bernama `Team` dengan nilai `Saanvi`, gunakan perintah ini.

```
aws codecommit create-repository --repository-name MyDemoRepo --repository-description "My demonstration repository" --tags Team=Saanvi
```

Note

Bidang deskripsi menampilkan Markdown di konsol dan menerima semua karakter HTML dan karakter Unicode yang valid. Jika Anda adalah pengembang aplikasi yang menggunakan `BatchGetRepositories` API `GetRepository` atau dan Anda berencana untuk menampilkan bidang deskripsi repositori di browser web, lihat Referensi [CodeCommit API](#).

3. Jika berhasil, perintah ini menghasilkan objek `repositoryMetadata` dengan informasi berikut:

- Deskripsi (`repositoryDescription`).
- ID unik yang dihasilkan oleh sistem (`repositoryId`).

- Nama (`repositoryName`).
- ID akun Amazon Web Services yang terkait dengan CodeCommit repositori (`accountId`).

Berikut ini adalah contoh output, berdasarkan contoh perintah sebelumnya.

```
{
  "repositoryMetadata": {
    "repositoryName": "MyDemoRepo",
    "cloneUrlSsh": "ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo",
    "lastModifiedDate": 1446071622.494,
    "repositoryDescription": "My demonstration repository",
    "cloneUrlHttp": "https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo",
    "defaultBranch": main,
    "kmsKeyId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "creationDate": 1446071622.494,
    "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
    "Arn": "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",
    "accountId": "111111111111"
  }
}
```

Note

Tag yang ditambahkan ketika repositori dibuat tidak dikembalikan dalam output. Untuk menampilkan daftar tanda yang terkait dengan repositori, jalankan perintah [list-tags-for-resource](#).

4. Catat nama dan ID CodeCommit repositori. Anda membutuhkan mereka untuk memantau dan mengubah informasi tentang CodeCommit repositori, terutama jika Anda menggunakannya.
AWS CLI

Jika Anda lupa nama atau ID, ikuti petunjuk di [Lihat detail CodeCommit repositori \(\)AWS CLI](#).

Setelah Anda membuat repositori, Anda dapat menghubungkannya dan mulai menambahkan kode. Untuk informasi selengkapnya, lihat [Connect ke sebuah repositori](#). Anda juga dapat menambahkan repositori Anda ke alur pengiriman berkelanjutan. Untuk informasi lebih lanjut, lihat [Panduan Alur Sederhana](#).

Connect ke AWS CodeCommit repositori

Ketika Anda terhubung ke CodeCommit repositori untuk pertama kalinya, Anda biasanya mengkloning isinya ke mesin lokal Anda. Anda juga dapat [menambahkan file](#) ke dan [mengedit file](#) di repositori langsung dari konsol. CodeCommit Atau, jika Anda sudah memiliki repo lokal, Anda dapat menambahkan CodeCommit repositori sebagai remote. Topik ini memberikan instruksi untuk menghubungkan ke CodeCommit repositori. Jika Anda ingin memigrasikan repositori yang ada, lihat. CodeCommit [Migrasi ke CodeCommit](#)

Note

Tergantung pada penggunaan Anda, Anda mungkin dikenakan biaya untuk membuat atau mengakses suatu repositori. Untuk informasi selengkapnya, lihat [Harga](#) di halaman informasi CodeCommit produk.

Topik

- [Prasyarat untuk menghubungkan ke repositori CodeCommit](#)
- [Connect ke CodeCommit repositori dengan mengkloning repositori](#)
- [Hubungkan repo lokal ke repositori CodeCommit](#)

Prasyarat untuk menghubungkan ke repositori CodeCommit

Sebelum Anda dapat mengkloning CodeCommit repositori atau menghubungkan repo lokal ke repositori: CodeCommit

- Anda harus telah mengonfigurasi komputer lokal Anda dengan perangkat lunak dan pengaturan yang diperlukan untuk terhubung CodeCommit. Ini termasuk menginstal dan mengkonfigurasi Git. Untuk informasi selengkapnya, lihat [Pengaturan](#) dan [Memulai dengan Git dan AWS CodeCommit](#).
- Anda harus memiliki URL klon dari CodeCommit repositori yang ingin Anda sambungkan. Untuk informasi selengkapnya, lihat [Tampilkan detail repositori](#).

Jika Anda belum membuat CodeCommit repositori, ikuti instruksi di [Buatlah sebuah repositori](#), salin URL klon CodeCommit repositori, dan kembali ke halaman ini.

Jika Anda memiliki CodeCommit repositori tetapi Anda tidak tahu namanya, ikuti instruksi di [Tampilkan detail repositori](#)

- Anda harus memiliki lokasi di komputer lokal Anda untuk menyimpan salinan lokal dari CodeCommit repositori yang Anda sambungkan. (Salinan lokal CodeCommit repositori ini dikenal sebagai repo lokal.) Anda kemudian beralih ke dan menjalankan perintah Git dari lokasi tersebut. Misalnya, Anda dapat menggunakan `/tmp` (untuk Linux, macOS, atau Unix) atau `c:\temp` (untuk Windows) jika Anda membuat kloning sementara untuk tujuan pengujian. Itu adalah jalur direktori yang digunakan dalam contoh-contoh ini.

Note

Anda dapat menggunakan direktori apa pun yang Anda inginkan. Jika Anda mengkloning repositori untuk penggunaan jangka panjang, pertimbangkan untuk membuat klon dari direktori yang bekerja dan bukan yang digunakan untuk file sementara. Jika Anda menggunakan direktori yang berbeda dari `/tmp` atau `c:\temp`, pastikan untuk mengganti direktori tersebut untuk direktori kami saat Anda mengikuti petunjuk ini.

Connect ke CodeCommit repositori dengan mengkloning repositori

Jika Anda belum memiliki repo lokal, ikuti langkah-langkah dalam prosedur ini untuk mengkloning CodeCommit repositori ke mesin lokal Anda.

1. Lengkapi prasyaratnya, termasuk [Pengaturan](#).

Important

Jika Anda belum menyelesaikan pengaturan, Anda tidak dapat terhubung ke atau mengkloning repositori tersebut.

2. Dari direktori `/tmp` atau direktori `c:\temp`, gunakan Git untuk menjalankan perintah `clone`. Contoh berikut menunjukkan cara mengkloning repositori yang dinamai *MyDemoRepo* di Wilayah Timur AS (Ohio).

Untuk HTTPS yang menggunakan [kredensial Git](#) atau pembantu kredensial yang disertakan dengan AWS CLI:

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Untuk HTTPS yang menggunakan [git-remote-codecommit](#), dengan asumsi profil default dan Wilayah AWS dikonfigurasi dalam AWS CLI:

```
git clone codecommit://MyDemoRepo my-demo-repo
```

Untuk SSH:

```
git clone ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Dalam contoh ini, `git-codecommit.us-east-2.amazonaws.com` adalah titik koneksi Git untuk Wilayah Timur AS (Ohio) di mana repositori ada, `MyDemoRepo` mewakili nama CodeCommit repositori Anda, dan `my-demo-repo` mewakili nama direktori yang dibuat Git di direktori atau direktori. /tmp c:\temp Untuk informasi selengkapnya tentang dukungan Wilayah AWS tersebut CodeCommit dan koneksi Git untuk mereka Wilayah AWS, lihat [Wilayah dan titik akhir koneksi Git](#).

Note

Ketika Anda menggunakan SSH pada sistem operasi Windows untuk mengkloning repositori, Anda mungkin perlu menambahkan ID kunci SSH ke string koneksi sebagai berikut:

```
git clone ssh://Your-SSH-Key-ID@git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Untuk informasi selengkapnya, lihat [Untuk koneksi SSH pada Windows](#) dan [Memecahkan masalah](#).

Setelah Git membuat direktori, ia menarik salinan CodeCommit repositori Anda ke direktori yang baru dibuat.

Jika CodeCommit repositori baru atau kosong, Anda melihat pesan bahwa Anda mengkloning repositori kosong. Ini yang diharapkan.

Note

Jika Anda menerima kesalahan bahwa Git tidak dapat menemukan CodeCommit repositori atau bahwa Anda tidak memiliki izin untuk terhubung ke CodeCommit repositori, pastikan Anda menyelesaikan [prasyarat](#), termasuk menetapkan izin kepada pengguna IAM dan menyiapkan kredensi pengguna IAM Anda untuk Git dan pada mesin lokal. CodeCommit Selain itu, pastikan Anda menentukan nama repositori yang benar.

Setelah Anda berhasil menghubungkan repo lokal Anda ke CodeCommit repositori Anda, Anda sekarang siap untuk mulai menjalankan perintah Git dari repo lokal untuk membuat commit, cabang, dan tag dan push ke dan pull dari repositori. CodeCommit

Hubungkan repo lokal ke repositori CodeCommit

Selesaikan langkah-langkah berikut jika Anda sudah memiliki repo lokal dan ingin menambahkan repositori sebagai CodeCommit repositori jarak jauh. Jika Anda sudah memiliki repositori jarak jauh dan ingin mendorong komit Anda ke CodeCommit dan repositori jarak jauh lainnya, ikuti langkah-langkahnya. [Dorong commit ke dua repositori](#)

1. Lengkapi [prasyarat](#).
2. Dari command prompt atau terminal, beralih ke direktori repo lokal Anda dan jalankan git remote add perintah untuk menambahkan repositori sebagai CodeCommit repositori jarak jauh untuk repo lokal Anda.

Misalnya, perintah berikut menambahkan remote yang dijuluki **origin** `https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo`:

Untuk HTTPS:

```
git remote add origin https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
```

Untuk SSH:

```
git remote add origin ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
```

Perintah ini tidak mengembalikan apa pun.

3. Untuk memverifikasi bahwa Anda telah menambahkan CodeCommit repositori sebagai remote untuk repo lokal Anda, jalankan `git remote -v` perintah, yang seharusnya membuat output yang mirip dengan berikut ini:

Untuk HTTPS:

```
origin https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (fetch)
origin https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (push)
```

Untuk SSH:

```
origin ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (fetch)
origin ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (push)
```

Setelah Anda berhasil menghubungkan repo lokal Anda ke CodeCommit repositori Anda, Anda siap untuk mulai menjalankan perintah Git dari repo lokal untuk membuat commit, cabang, dan tag, dan untuk mendorong dan menarik dari repositori. CodeCommit

Bagikan AWS CodeCommit repositori

Setelah Anda membuat CodeCommit repositori, Anda dapat membagikannya dengan pengguna lain. Pertama, tentukan apakah Anda akan menggunakan akses federasi, kredensial sementara, atau penyedia identitas web seperti IAM Identity Center saat mengakses CodeCommit, atau jika Anda ingin menggunakan kredensial Git atau pasangan kunci SSH dengan pengguna IAM. Jika Anda menggunakan yang pertama, Anda perlu mengatur pengguna, akses, dan izin untuk penyedia identitas Anda, dan kemudian memberikan instruksi untuk digunakan `git-remote-codecommit` pengguna Anda. Untuk informasi selengkapnya, lihat [Langkah-langkah penyiapan untuk koneksi HTTPS ke AWS CodeCommit bersama git-remote-codecommit](#) dan [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#).

Anda tidak dapat menggunakan kredensial Git atau pasangan kunci SSH dengan akses federasi atau penyedia identitas, tetapi banyak IDE bekerja paling baik dengan kredensial ini. Dalam hal ini,

tentukan protokol mana (HTTPS atau SSH) yang akan direkomendasikan kepada pengguna saat mengkloning dan menggunakan klien Git atau IDE untuk terhubung ke repositori Anda. Kemudian kirim URL dan informasi koneksi ke pengguna dengan siapa Anda ingin berbagi repositori tersebut. Bergantung pada persyaratan keamanan Anda, berbagi repositori mungkin juga memerlukan pembuatan grup IAM, menerapkan kebijakan terkelola ke grup tersebut, dan mengedit kebijakan IAM untuk menyempurnakan akses, atau membuat dan menggunakan peran IAM.

Note

Setelah Anda telah memberikan akses konsol pengguna ke repositori tersebut, mereka dapat menambahkan atau mengedit file langsung di konsol tanpa harus mengatur klien Git atau koneksi lainnya. Untuk informasi selengkapnya, lihat [Membuat atau menambahkan file ke repositori AWS CodeCommit](#) dan [Mengedit isi file dalam repositori AWS CodeCommit](#).

Instruksi ini ditulis dengan asumsi bahwa Anda telah menyelesaikan langkah-langkah di [Pengaturan](#) dan [Buatlah sebuah repositori](#).

Note

Tergantung pada penggunaan Anda, Anda mungkin dikenakan biaya untuk membuat atau mengakses suatu repositori. Untuk informasi selengkapnya, lihat [Harga](#) di halaman informasi CodeCommit produk.

Topik

- [Pilih protokol koneksi yang akan dibagikan dengan para pengguna Anda](#)
- [Buatlah kebijakan IAM untuk repositori Anda](#)
- [Buatlah sebuah grup IAM untuk para pengguna repositori](#)
- [Bagikan informasi koneksi dengan para pengguna Anda](#)

Pilih protokol koneksi yang akan dibagikan dengan para pengguna Anda

Saat Anda membuat repositori di CodeCommit, dua titik akhir dihasilkan: satu untuk koneksi HTTPS dan satu untuk koneksi SSH. Keduanya menyediakan koneksi yang aman melalui sebuah jaringan. Para pengguna Anda dapat menggunakan salah satu protokol. Kedua endpoint tetap aktif terlepas dari protokol mana yang Anda rekomendasikan untuk para pengguna Anda.

Koneksi HTTPS mensyaratkan salah satu:

- Kredensial Git, yang dapat dihasilkan oleh para pengguna IAM untuk mereka sendiri di IAM. Kredensial Git adalah metode termudah bagi para pengguna repositori Anda untuk mengatur dan menggunakan.
- Kunci AWS akses atau peran untuk diasumsikan, yang harus dikonfigurasi oleh pengguna repositori Anda di profil kredensialnya. Anda dapat mengonfigurasi `git-remote-codecommit` (disarankan) atau penolong kredensial yang disertakan dalam AWS CLI. Ini adalah satu-satunya metode yang tersedia untuk akun root atau para pengguna gabungan.

Koneksi SSH mensyaratkan pengguna Anda untuk:

- Menghasilkan sebuah pasangan kunci publik-pribadi.
- Simpan kunci publik.
- Kaitkan kunci publik dengan pengguna IAM mereka.
- Konfigurasi file host mereka yang telah dikenal pada komputer lokal mereka.
- Buat dan pelihara file konfigurasi pada komputer lokal mereka.

Karena ini adalah proses konfigurasi yang lebih kompleks, kami sarankan Anda memilih kredensi HTTPS dan Git untuk koneksi ke CodeCommit

Untuk informasi lebih lanjut tentang HTTPS, SSH, Git, `git-remote-codecommit`, dan repositori jarak jauh, lihat [Pengaturan](#), [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#), atau konsultasikan dengan dokumentasi Git Anda. Untuk gambaran umum protokol komunikasi dan bagaimana masing-masing berkomunikasi dengan repositori jarak jauh, lihat [Git pada Server - Protokol](#).


Note

Meskipun Git mendukung berbagai protokol koneksi, CodeCommit tidak mendukung koneksi dengan protokol yang tidak aman, seperti protokol lokal atau HTTP generik.

Buatlah kebijakan IAM untuk repositori Anda

AWS menyediakan tiga kebijakan terkelola di IAM untuk CodeCommit. Kebijakan ini tidak dapat diedit dan berlaku untuk semua repositori yang terkait dengan akun Amazon Web Services Anda. Namun,

Anda dapat menggunakan kebijakan ini sebagai templat untuk membuat kebijakan terkelola kustom Anda sendiri yang hanya berlaku untuk repositori yang ingin Anda bagikan. Kebijakan terkelola pelanggan Anda dapat diterapkan secara khusus ke repositori yang ingin Anda bagikan. Untuk informasi selengkapnya, lihat [Kebijakan terkelola](#) dan [Pengguna dan Grup IAM](#).

 Tip

Untuk kontrol yang lebih halus atas akses ke repositori Anda, Anda dapat membuat lebih dari satu kebijakan terkelola pelanggan dan menerapkan kebijakan untuk pengguna dan grup IAM yang berbeda.

Untuk informasi tentang meninjau konten kebijakan terkelola dan menggunakan kebijakan untuk membuat dan menerapkan izin, lihat [Kontrol autentikasi dan akses untuk AWS CodeCommit](#).

Buatlah kebijakan terkelola pelanggan untuk repositori Anda

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Pada panel navigasi Dasbor, pilih Kebijakan, dan kemudian pilih Buat Kebijakan.
3. Pada halaman Buat Kebijakan, pilih Impor kebijakan terkelola.
4. Pada halaman Impor kebijakan terkelola, di Kebijakan filter, masukkan **AWSCodeCommitPowerUser**. Pilih tombol di samping nama kebijakan, lalu pilih Impor.
5. Di halaman Buat kebijakan, pilih JSON. Ganti bagian "*" dari Resource baris untuk CodeCommit tindakan dengan Nama Sumber Daya Amazon (ARN) dari CodeCommit repositori, seperti yang ditunjukkan di sini:

```
"Resource": [  
  "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo"  
]
```

 Tip

Untuk menemukan ARN untuk CodeCommit repositori, buka CodeCommit konsol, pilih nama repositori dari daftar, lalu pilih Pengaturan. Untuk informasi selengkapnya, lihat [Tampilkan detail repositori](#).

Jika Anda ingin kebijakan ini diterapkan ke lebih dari satu repositori, tambahkan setiap repositori sebagai sumber daya dengan menentukan ARN-nya. Sertakan koma antara setiap pernyataan sumber daya, seperti yang ditunjukkan di sini:

```
"Resource": [  
  "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",  
  "arn:aws:codecommit:us-east-2:111111111111:MyOtherDemoRepo"  
]
```

Setelah Anda selesai mengedit, pilih Tinjau kebijakan.


6. Pada halaman Kebijakan Tinjauan, di Nama, masukkan nama baru untuk kebijakan (misalnya, *AWSCodeCommitPowerUser- MyDemoRepo*). Secara opsional berikan deskripsi untuk kebijakan ini.
7. Pilih Buat Kebijakan.

Buatlah sebuah grup IAM untuk para pengguna repositori

Untuk mengelola akses ke repositori Anda, buatlah sebuah grup IAM untuk para penggunanya, tambahkan pengguna IAM ke grup tersebut, lalu lampirkan kebijakan terkelola pelanggan yang Anda buat pada langkah sebelumnya. Atau, Anda dapat membuat peran dengan kebijakan terkelola pelanggan terlampir, dan meminta pengguna mengambil peran itu.


Jika Anda menggunakan SSH, Anda harus melampirkan kebijakan terkelola lain ke grup `lamusersShKeys`, kebijakan terkelola IAM yang memungkinkan pengguna mengunggah kunci publik SSH mereka dan mengaitkannya dengan pengguna IAM yang mereka gunakan untuk terhubung. CodeCommit

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Dalam area navigasi Dasbor, pilih Grup, dan kemudian pilih Buat Grup Baru.
3. Pada halaman Setel Nama Grup, di Nama Grup, masukkan nama untuk grup (misalnya, *MyDemoRepoGroup*), lalu pilih Langkah Berikutnya. Pertimbangkan memasukkan nama repositori sebagai bagian dari nama grup.

 Note

Nama ini harus unik di seluruh akun Amazon Web Services.

4. Pilih kotak di sebelah kebijakan terkelola pelanggan yang Anda buat di bagian sebelumnya (misalnya, AWSCodeCommitPowerUser- MyDemoRepo).
5. Pada halaman Tinjauan, pilih Buat Grup. IAM membuat grup ini dengan kebijakan tertentu yang sudah terlampir. Grup tersebut muncul dalam daftar grup yang terkait dengan akun Amazon Web Services Anda.
6. Pilih nama grup Anda dari daftar.
7. Di halaman ringkasan grup, pilih tab Pengguna, dan kemudian pilih Tambahkan Pengguna ke Grup. Pada daftar yang menampilkan semua pengguna yang terkait dengan akun Amazon Web Services Anda, pilih kotak di sebelah pengguna yang ingin Anda izinkan akses ke CodeCommit repositori, lalu pilih Tambah Pengguna.

 Tip

Anda dapat menggunakan kotak Pencarian untuk menemukan pengguna dengan cepat berdasarkan nama.

8. Bila Anda telah menambahkan pengguna Anda, tutup konsol IAM tersebut.

Bagikan informasi koneksi dengan para pengguna Anda

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di pemilih wilayah, pilih Wilayah AWS tempat repositori dibuat. Repositori khusus untuk file. Wilayah AWS Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Pada halaman Repositori, pilih repositori yang Anda ingin bagikan.
4. Dalam URL Klon, pilih protokol yang Anda inginkan agar digunakan oleh para pengguna Anda. Ini menyalin URL klon untuk protokol koneksi.
5. Kirim URL klon kepada pengguna Anda bersama dengan instruksi lainnya, seperti menginstal AWS CLI, mengonfigurasi profil, atau menginstal Git. Pastikan untuk menyertakan informasi konfigurasi untuk protokol koneksi (misalnya, HTTPS).

Contoh email berikut memberikan informasi bagi pengguna yang terhubung ke MyDemoRepo repositori dengan protokol koneksi HTTPS dan kredensial Git di Wilayah AS Timur (Ohio) (us-east-2). Email ini ditulis dengan asumsi pengguna telah menginstal Git dan akrab dengan menggunakannya.

```
I've created a CodeCommit repository for us to use while working on our project.
The name of the repository is MyDemoRepo, and
it is in the US East (Ohio) (us-east-2) region.
Here's what you need to do in order to get started using it:
```

1. Make sure that your version of Git on your local computer is 1.7.9 or later.
2. Generate Git credentials for your IAM user by signing into the IAM console here: <https://console.aws.amazon.com/iam/>.

```
Switch to the Security credentials tab for your IAM user and choose the Generate button
in HTTPS Git credentials for CodeCommit.
```

```
Make sure to save your credentials in a secure location!
```

3. Switch to a directory of your choice and clone the CodeCommit repository to your local machine by running the following command:

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```
4. When prompted for user name and password, use the Git credentials you just saved.

```
That's it! If you'd like to learn more about using CodeCommit, you can start with the
tutorial here.
```

Anda dapat menemukan petunjuk penyiapan lengkap di [Pengaturan](#) .

Mengkonfigurasi notifikasi untuk peristiwa dalam sebuah repositori AWS CodeCommit

Anda dapat mengatur aturan notifikasi untuk repositori sehingga pengguna repositori menerima email tentang jenis peristiwa repositori yang Anda tentukan. Notifikasi dikirim ketika peristiwa cocok dengan pengaturan aturan peristiwa tersebut. Anda dapat membuat topik Amazon SNS untuk digunakan untuk notifikasi atau menggunakan topik yang sudah ada di akun Amazon Web Services Anda. Anda dapat menggunakan CodeCommit konsol dan AWS CLI untuk mengkonfigurasi aturan notifikasi.

Developer Tools > CodeCommit > Repositories > MyDemoRepo > Settings

Create notification rule

Notification rules set up a subscription to events that happen with your resources. When these events occur, you will receive notifications sent to the targets you designate. You can manage your notification preferences in Settings. [Info](#)

Notification rule settings

Notification name

Detail type

Choose the level of detail you want in notifications. [Learn more about notifications and security](#)

Full
Includes any supplemental information about events provided by the resource or the notifications feature.

Basic
Includes only information provided in resource events.

Events that trigger notifications

Comments	Pull request	Branches and tags
<input type="checkbox"/> On Commits	<input checked="" type="checkbox"/> Source Updated	<input type="checkbox"/> Created
<input checked="" type="checkbox"/> On Pull requests	<input checked="" type="checkbox"/> Created	<input checked="" type="checkbox"/> Deleted
	<input checked="" type="checkbox"/> Status Changed	<input type="checkbox"/> Updated
	<input checked="" type="checkbox"/> Merged	

Targets

Choose an SNS topic to use as the target for the notification rule. Users can subscribe to the notification topic to receive emails about events. You can also configure integration between the SNS topic and AWS Chatbot, so that users receive notifications in Slack channels or Amazon Chime chatrooms.

You can also configure integration between the SNS topic and AWS Chatbot, so that users receive notifications in Slack channels or Amazon Chime chatrooms. [Learn more](#)

Amazon SNS topic ARN

Topik

- [Menggunakan aturan notifikasi repositori](#)
- [Membuat aturan notifikasi](#)

- [Mengubah atau menonaktifkan pemberitahuan](#)
- [Hapus notifikasi](#)

Menggunakan aturan notifikasi repositori

Mengkonfigurasi aturan notifikasi membantu pengguna repositori Anda dengan mengirimkan email ketika seseorang mengambil tindakan yang mempengaruhi pengguna lain. Misalnya, Anda dapat mengkonfigurasi aturan notifikasi untuk mengirim pemberitahuan ketika komentar dibuat pada commit. Dalam konfigurasi ini, ketika pengguna repositori berkomentar pada baris kode dalam sebuah commit, pengguna repositori lain menerima email. Mereka dapat masuk dan melihat komentar tersebut. Tanggapan terhadap komentar juga menghasilkan email, sehingga pengguna repositori terus mendapat informasi.

Aturan notifikasi berbeda dari pemicu repositori, dan juga berbeda dari notifikasi yang dapat Anda konfigurasi di CodeCommit konsol sebelum 5 November 2019.

- Meskipun Anda dapat mengkonfigurasi pemicu untuk menggunakan Amazon SNS untuk mengirim email tentang beberapa peristiwa repositori, peristiwa tersebut terbatas pada peristiwa operasional, seperti menciptakan cabang dan mendorong kode ke suatu cabang. Pemicu tidak menggunakan aturan CloudWatch Peristiwa untuk mengevaluasi peristiwa repositori. Mereka memiliki lingkup yang lebih terbatas. Untuk informasi lebih lanjut tentang menggunakan pemicu lihat [Kelola pemicu untuk repositori](#).
- Notifikasi yang dikonfigurasi sebelum 5 November 2019 memiliki lebih sedikit jenis peristiwa yang tersedia, dan tidak dapat dikonfigurasi untuk integrasi dengan ruang obrolan Amazon Chime atau saluran Slack. Anda dapat terus menggunakan notifikasi yang dikonfigurasi sebelum 5 November 2019, tetapi Anda tidak dapat membuat notifikasi jenis ini. Alih-alih, buatlah dan gunakan aturan notifikasi. Kami merekomendasikan Anda menggunakan aturan notifikasi dan nonaktifkan atau hapus pemberitahuan yang dibuat sebelum 5 November 2019. Untuk informasi selengkapnya, lihat [Membuat aturan notifikasi](#) dan [Hapus notifikasi](#).

Membuat aturan notifikasi

Anda dapat menggunakan aturan notifikasi untuk memberitahu pengguna perubahan penting, seperti ketika permintaan tarik dibuat dalam sebuah repositori. Aturan notifikasi menentukan peristiwa dan topik Amazon SNS yang digunakan untuk mengirim notifikasi. Untuk informasi selengkapnya, lihat [Apa itu notifikasi?](#)

Note

Fitur ini tidak tersedia di Wilayah Eropa (Milan). Untuk mempelajari cara mengonfigurasi notifikasi dalam pengalaman yang tersedia di Wilayah tersebut, lihat [Mengkonfigurasi Notifikasi Repository](#).

Anda dapat menggunakan konsol atau AWS CLI untuk membuat aturan notifikasi untuk AWS CodeCommit.

Untuk membuat aturan notifikasi (konsol)

1. Masuk ke AWS Management Console dan buka CodeCommit konsol di <https://console.aws.amazon.com/codecommit/>.
2. Pilih Repository, dan kemudian pilih repository tempat Anda ingin menambahkan aturan notifikasi.
3. Pada halaman repository, pilih Beritahu, lalu pilih Buat aturan notifikasi. Anda juga dapat pergi ke Pengaturan untuk repository tersebut dan pilih Buat aturan notifikasi.
4. Di Nama notifikasi, masukkan nama untuk aturan.
5. Di Jenis detail, pilih Dasar jika Anda hanya menginginkan informasi yang diberikan ke Amazon yang EventBridge disertakan dalam notifikasi. Pilih Lengkap jika Anda ingin menyertakan informasi yang diberikan ke Amazon EventBridge dan informasi yang mungkin diberikan oleh CodeCommit atau pengelola notifikasi.

Untuk informasi selengkapnya, lihat [Memahami Konten Notifikasi dan Keamanan](#).

6. Di Peristiwa yang memicu notifikasi, pilih peristiwa yang ingin Anda kirimkan notifikasi. Untuk informasi selengkapnya, lihat [Peristiwa untuk Aturan Notifikasi di Repository](#).
7. Di Target, lakukan salah satu langkah berikut:
 - Jika Anda telah mengonfigurasi sumber daya untuk digunakan dengan notifikasi, di Pilih tipe target, pilih salah satu dari AWS Chatbot (Slack) atau Topik SNS. Di Pilih target, pilih nama klien (untuk klien Slack yang dikonfigurasi AWS Chatbot) atau Nama Sumber Daya Amazon (ARN) dari topik Amazon SNS (untuk topik Amazon SNS yang sudah dikonfigurasi dengan kebijakan yang diperlukan untuk pemberitahuan).
 - Jika Anda belum mengonfigurasi sumber daya untuk digunakan dengan notifikasi, pilih Buat target, lalu pilih Topik SNS. Berikan nama untuk topik setelah codestar-notifications-, lalu pilih Buat.

Note

- Jika Anda membuat topik Amazon SNS sebagai bagian dari membuat aturan notifikasi, kebijakan yang memungkinkan fitur notifikasi untuk mempublikasikan peristiwa untuk topik diterapkan untuk Anda. Menggunakan topik yang dibuat untuk aturan notifikasi membantu memastikan bahwa Anda hanya berlangganan pengguna yang ingin Anda terima notifikasinya tentang sumber daya ini.
- Anda tidak dapat membuat AWS Chatbot klien sebagai bagian dari pembuatan aturan notifikasi. Jika Anda memilih AWS Chatbot (Slack), Anda akan melihat tombol yang mengarahkan Anda untuk mengkonfigurasi klien. AWS Chatbot Memilih opsi itu membuka AWS Chatbot konsol. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Integrasi Antara Pemberitahuan dan AWS Chatbot](#).
- Jika ingin menggunakan topik Amazon SNS yang ada sebagai target, Anda harus menambahkan kebijakan yang diperlukan untuk AWS CodeStar Pemberitahuan selain kebijakan lain yang mungkin ada untuk topik tersebut. Untuk informasi selengkapnya, lihat [Konfigurasi Topik Amazon SNS untuk Notifikasi](#) dan [Memahami Konten Notifikasi dan Keamanan](#).

8. Untuk menyelesaikan pembuatan aturan, pilih Kirim.
9. Anda harus membuat pengguna berlangganan ke topik Amazon SNS untuk aturan tersebut sebelum mereka dapat menerima notifikasi. Untuk informasi selengkapnya, lihat [Buat Pengguna Berlangganan ke Topik Amazon SNS yang Merupakan Target](#). Anda juga dapat mengatur integrasi antara notifikasi dan mengirim notifikasi AWS Chatbot ke ruang obrolan Amazon Chime. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Integrasi Antara Pemberitahuan dan AWS Chatbot](#).

Membuat aturan notifikasi (AWS CLI)

1. Di terminal atau prompt perintah, jalankan perintah create-notification rule untuk menghasilkan kerangka JSON:

```
aws codestar-notifications create-notification-rule --generate-cli-skeleton  
> rule.json
```


Anda dapat memberi nama file apa pun yang Anda inginkan. Dalam contoh ini, file diberi nama *rule.json*.

2. Buka file JSON dalam editor plaintext dan edit untuk memasukkan sumber daya, jenis peristiwa, dan target yang Anda inginkan untuk aturan tersebut. *Contoh berikut menunjukkan aturan notifikasi bernama **MyNotificationRule** untuk repositori bernama **MyDemoRepo** dalam AWS account dengan ID **123456789012**. Pemberitahuan dengan tipe detail lengkap dikirim ke topik Amazon SNS yang diberi nama **MyNotificationTopic** saat cabang dan tag dibuat:*

```
{
  "Name": "MyNotificationRule",
  "EventTypes": [
    "codecommit-repository-branches-and-tags-created"
  ],
  "Resource": "arn:aws:codecommit:us-east-1:123456789012:MyDemoRepo",
  "Targets": [
    {
      "TargetType": "SNS",
      "TargetAddress": "arn:aws:sns:us-east-1:123456789012:MyNotificationTopic"
    }
  ],
  "Status": "ENABLED",
  "DetailType": "FULL"
}
```

Simpan file tersebut.

3. Menggunakan file yang baru saja Anda edit, di terminal atau baris perintah, jalankan `create-notification-rule` perintah lagi untuk membuat aturan notifikasi:

```
aws codestar-notifications create-notification-rule --cli-input-json
file://rule.json
```

4. Jika berhasil, perintah tersebut mengembalikan ARN aturan notifikasi, yang serupa dengan berikut ini:

```
{
  "Arn": "arn:aws:codestar-notifications:us-east-1:123456789012:notificationrule/
dc82df7a-EXAMPLE"
```

```
}
```

Mengubah atau menonaktifkan pemberitahuan

Anda dapat menggunakan AWS CodeCommit konsol untuk mengubah cara pemberitahuan yang dibuat sebelum 5 November 2019 dikonfigurasi, termasuk jenis acara yang mengirim email ke pengguna dan topik Amazon SNS yang digunakan untuk mengirim email tentang repositori. Anda juga dapat menggunakan CodeCommit konsol untuk mengelola daftar alamat email dan titik akhir yang berlangganan topik atau untuk menonaktifkan notifikasi.

Untuk mengubah pengaturan notifikasi

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin mengkonfigurasi pemberitahuan yang dibuat sebelum 5 November 2019.
3. Di bilah navigasi, pilih Pengaturan, lalu pilih Notifikasi. Jika Anda melihat banner yang menginformasikan bahwa Anda memiliki notifikasi, alih-alih aturan notifikasi, pilih Kelola notifikasi yang ada.
4. Pilih Edit.
5. Buat perubahan Anda dan kemudian pilih Simpan.

Menonaktifkan notifikasi adalah cara mudah untuk mencegah sementara pengguna menerima email tentang peristiwa repositori.

Untuk menghapus notifikasi yang dibuat secara permanen sebelum 5 November 2019, ikuti langkah-langkah di [Hapus notifikasi](#).

Untuk menonaktifkan notifikasi

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori yang Anda ingin menonaktifkan notifikasi.
3. Di bilah navigasi, pilih Pengaturan, lalu pilih Notifikasi. Pilih Kelola notifikasi yang ada.
4. Pilih Edit, dan di Status peristiwa, gunakan slider untuk mematikan Aktifkan notifikasi. Pilih Simpan.

5. Status peristiwa berubah menjadi Nonaktif. Tidak ada email tentang peristiwa yang dikirim. Saat Anda menonaktifkan notifikasi, aturan CloudWatch Acara untuk repositori dinonaktifkan secara otomatis. Jangan mengubah statusnya secara manual di konsol CloudWatch Acara.

Hapus notifikasi

Jika Anda tidak lagi ingin menggunakan notifikasi yang dibuat untuk repositori sebelum 5 November 2019, Anda dapat menghapus aturan Amazon CloudWatch Events yang terkait dengan notifikasi. Itu akan secara otomatis akan menghapus notifikasi. Ini tidak menghapus langganan atau topik Amazon SNS yang digunakan untuk notifikasi.

Note

Jika Anda mengubah nama repositori dari konsol, notifikasi yang dibuat sebelum 5 November 2019 terus bekerja tanpa modifikasi. Namun, jika Anda mengubah nama repositori Anda dari baris perintah atau dengan menggunakan API, notifikasi tidak lagi bekerja. Cara paling mudah untuk memulihkan notifikasi adalah menghapus pengaturan notifikasi dan kemudian mengkonfigurasinya lagi.

Untuk menghapus setelan notifikasi

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori yang Anda ingin hapus notifikasi yang dibuat sebelum 5 November 2019.
3. Di bilah navigasi, pilih Pengaturan, lalu pilih Notifikasi. Jika Anda melihat banner yang menginformasikan bahwa Anda memiliki notifikasi, alih-alih aturan notifikasi, pilih Kelola notifikasi yang ada.
4. Dalam aturan CloudWatch acara, salin nama aturan yang dibuat untuk pemberitahuan.
5. Masuk ke AWS Management Console dan buka CloudWatch konsol di <https://console.aws.amazon.com/cloudwatch/>.
6. Di Peristiwa, pilih Aturan. Di Nama, tempelkan nama peraturan yang dibuat untuk notifikasi tersebut. Pilih aturan dan dalam Tindakan, pilih Hapus.
7. (Opsional) Untuk mengubah atau menghapus topik Amazon SNS yang digunakan untuk notifikasi setelah Anda menghapus pengaturan notifikasi, buka konsol Amazon SNS di <https://>

console.aws.amazon.com/sns/v3/home. Untuk informasi lebih lanjut, lihat [Bersihkan](#) dalam [Panduan Developer Amazon Simple Notification Service](#).

Menandai repositori di AWS CodeCommit

Tag adalah label atribut kustom yang Anda atau AWS tetapkan ke AWS sumber daya. AWS tag berbeda dari tag Git, yang dapat diterapkan untuk commit. Setiap AWS tag memiliki dua bagian:

- Kunci tag (misalnya, `CostCenter`, `Environment`, `Project`, atau `Secret`). Kunci tanda peka terhadap huruf besar dan kecil.
- Bidang opsional yang dikenal sebagai nilai tag (misalnya, `111122223333`, `Production`, atau nama tim). Mengabaikan nilai tag sama dengan menggunakan rangkaian kosong. Seperti kunci tanda, nilai tanda peka huruf besar dan kecil.

Bersama-sama ini dikenal sebagai pasangan nilai-kunci. Untuk batas jumlah tag yang dapat Anda miliki di repositori dan batasan pada kunci dan nilai tag, lihat [Batasan](#).

Tag membantu Anda mengidentifikasi dan mengatur AWS sumber daya Anda. Banyak AWS layanan mendukung penandaan, sehingga Anda dapat menetapkan tag yang sama ke sumber daya dari layanan yang berbeda untuk menunjukkan bahwa sumber daya terkait. Misalnya, Anda dapat menetapkan tag yang sama ke CodeCommit repositori yang Anda tetapkan ke bucket Amazon S3. Untuk informasi selengkapnya tentang strategi penandaan, lihat [AWS Menandai](#) Sumber Daya.

Di CodeCommit, sumber daya utama adalah repositori. Anda dapat menggunakan CodeCommit konsol, CodeCommit API AWS CLI, atau AWS SDK untuk menambahkan, mengelola, dan menghapus tag untuk repositori. Selain mengidentifikasi, mengatur, dan melacak repositori Anda dengan tag, Anda dapat menggunakan tag dalam kebijakan IAM untuk membantu mengontrol siapa yang dapat melihat dan berinteraksi dengan repositori Anda. Untuk contoh kebijakan akses berbasis tag, lihat [Contoh 5: Tolak atau izinkan tindakan pada repositori dengan tag](#).

Topik

- [Tambahkan tag ke sebuah repositori](#)
- [Tampilkan tag untuk repositori](#)
- [Edit tag untuk repositori](#)
- [Hapus sebuah tag dari repositori](#)

Tambahkan tag ke sebuah repositori

Menambahkan tag ke repositori dapat membantu Anda mengidentifikasi dan mengatur AWS sumber daya Anda dan mengelola akses ke sana. Pertama, Anda menambahkan satu atau lebih tag (pasangan kunci-nilai) ke sebuah repositori. Perlu diingat bahwa ada batasan pada jumlah tag yang dapat Anda miliki di sebuah repositori. Ada pembatasan pada karakter yang dapat Anda gunakan di bidang kunci dan nilai. Untuk informasi selengkapnya, lihat [Batasan-batasan](#). Setelah Anda memiliki tag, Anda dapat membuat kebijakan IAM untuk mengelola akses ke repositori berdasarkan tag ini. Anda dapat menggunakan CodeCommit konsol atau AWS CLI untuk menambahkan tag ke repositori.

Important

Menambahkan tag ke repositori dapat memengaruhi akses ke repositori tersebut. Sebelum Anda menambahkan tag ke repositori, pastikan untuk meninjau kebijakan IAM yang mungkin menggunakan tag untuk mengontrol akses ke sumber daya seperti repositori. Untuk contoh kebijakan akses berbasis tag, lihat [Contoh 5: Tolak atau izinkan tindakan pada repositori dengan tag](#).

Untuk informasi selengkapnya tentang menambahkan tag ke repositori saat Anda membuatnya, lihat [Buatlah sebuah \(konsol\) repositori](#).

Topik

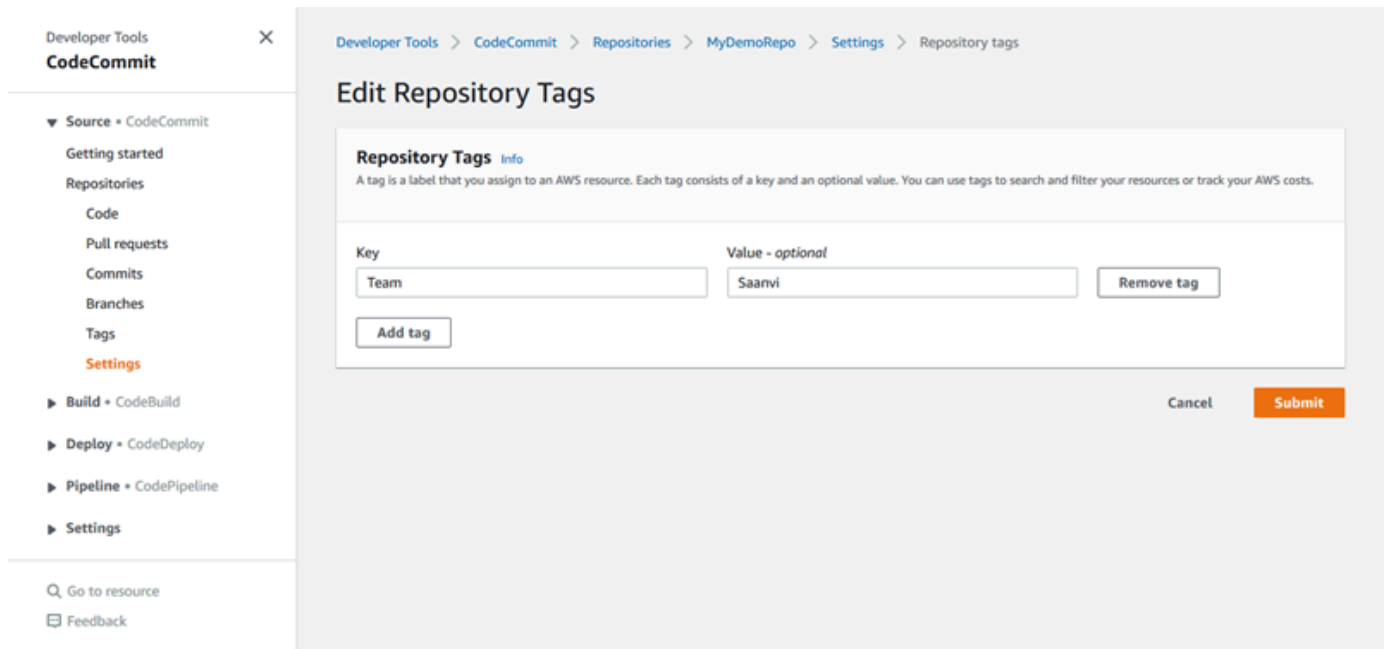
- [Tambahkan tag ke sebuah repositori \(konsol\)](#)
- [Tambahkan tag ke sebuah repositori \(AWS CLI\)](#)

Tambahkan tag ke sebuah repositori (konsol)

Anda dapat menggunakan CodeCommit konsol untuk menambahkan satu atau beberapa tag ke CodeCommit repositori.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin tambahkan tag.
3. Pada panel navigasi, silakan pilih Pengaturan. Pilih Tag repositori.
4. Jika tidak ada tag yang ditambahkan ke repositori, pilih Tambahkan tag. Jika tidak, pilih Edit, lalu pilih Tambahkan tanda.

- Di Kunci, masukkan sebuah nama untuk tag tersebut. Anda dapat menambahkan nilai opsional untuk tag di Nilai.



- (Opsional) Untuk menambahkan tag lain, pilih Tambahkan tag lagi.
- Setelah Anda selesai menambahkan tag, pilih Kirim.

Tambahkan tag ke sebuah repositori (AWS CLI)

Ikuti langkah-langkah ini untuk menggunakan AWS CLI untuk menambahkan tag ke CodeCommit repositori. Untuk menambahkan tag ke repositori saat Anda membuatnya, lihat [Buatlah sebuah repositori \(AWS CLI\)](#).

Dalam langkah-langkah ini, kami menganggap bahwa Anda telah menginstal versi terbaru dari AWS CLI atau diperbarui ke versi terkini. Untuk informasi lebih lanjut, lihat [Menginstal AWS Command Line Interface](#).

Pada terminal atau baris perintah, jalankan perintah `tag-resource`, dengan menentukan Amazon Resource Name (ARN) dari repositori tempat Anda ingin menambahkan tag dan kunci dan nilai dari tanda yang ingin ditambahkan. Anda dapat menambahkan lebih dari satu tag ke repositori. *Misalnya, untuk menandai repositori bernama `MyDemoRepo` dengan dua tag, kunci tag bernama `Status` dengan nilai tag `Secret`, dan kunci tag bernama `Team` dengan nilai tag `Saanvi`:*

```
aws codecommit tag-resource --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo --tags Status=Secret,Team=Saanvi
```

Jika berhasil, perintah ini tidak mengembalikan apa pun.

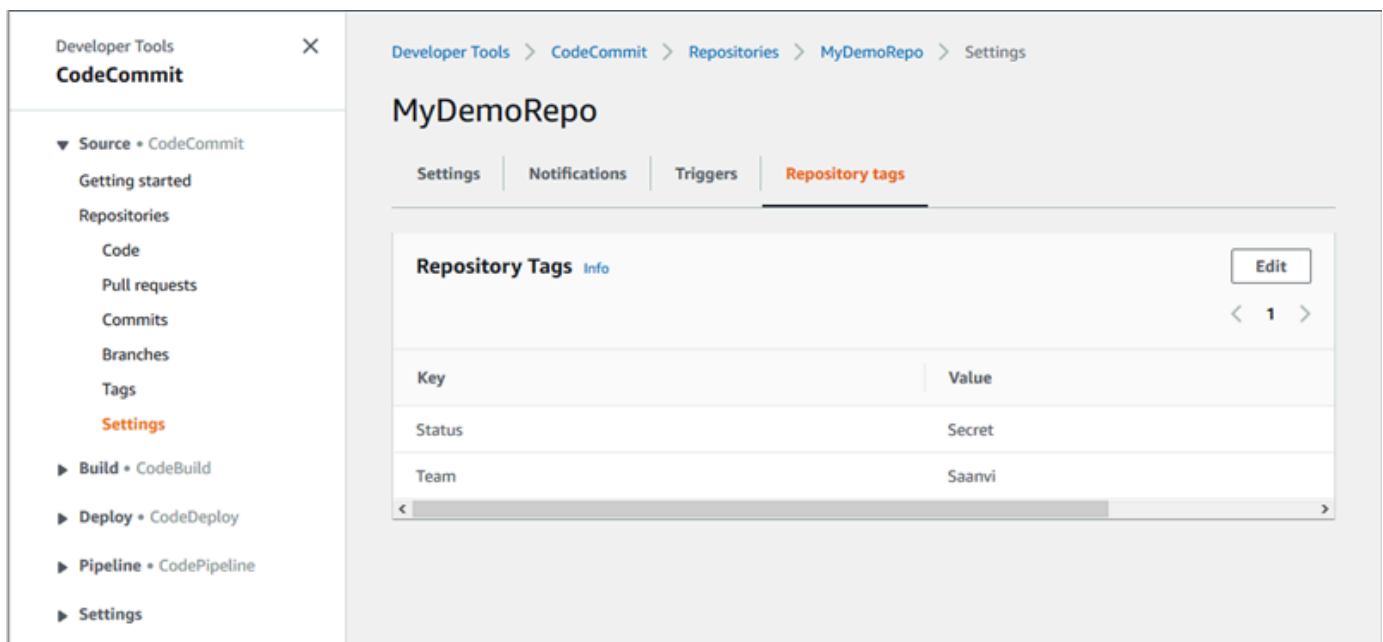
Tampilkan tag untuk repositori

Tag dapat membantu Anda mengidentifikasi dan mengatur AWS sumber daya Anda dan mengelola akses ke sana. Untuk informasi selengkapnya tentang strategi penandaan, lihat [AWS Menandai Sumber Daya](#). Untuk contoh kebijakan akses berbasis tag, lihat [Contoh 5: Tolak atau izinkan tindakan pada repositori dengan tag](#).

Tampilkan tag untuk repositori (konsol)

Anda dapat menggunakan CodeCommit konsol untuk melihat tag yang terkait dengan CodeCommit repositori.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori yang Anda ingin tampilkan tag.
3. Pada panel navigasi, silakan pilih Pengaturan. Pilih Tag repositori.



The screenshot shows the AWS CodeCommit console interface. On the left is a navigation sidebar with a 'Developer Tools' header and a 'CodeCommit' sub-header. Under 'CodeCommit', there is a 'Source' section with a dropdown arrow, and a list of options: 'Getting started', 'Repositories', 'Code', 'Pull requests', 'Commits', 'Branches', 'Tags', 'Settings' (highlighted in orange), 'Build' (with a dropdown arrow), 'Deploy' (with a dropdown arrow), 'Pipeline' (with a dropdown arrow), and 'Settings' (with a dropdown arrow). The main content area shows the breadcrumb 'Developer Tools > CodeCommit > Repositories > MyDemoRepo > Settings'. Below the breadcrumb is the title 'MyDemoRepo' and a set of tabs: 'Settings', 'Notifications', 'Triggers', and 'Repository tags' (highlighted in orange). Under the 'Repository tags' tab, there is a section titled 'Repository Tags' with an 'Info' link and an 'Edit' button. Below this is a table with two columns: 'Key' and 'Value'. The table contains three rows: 'Status' with value 'Secret', and 'Team' with value 'Saanvi'. There is a pagination indicator '< 1 >' and a scrollbar at the bottom of the table.

Key	Value
Status	Secret
Team	Saanvi

Tampilkan tag untuk repositori (AWS CLI)

Ikuti langkah-langkah ini untuk menggunakan AWS CLI untuk melihat AWS tag untuk CodeCommit repositori. Jika tidak ada tanda yang telah ditambahkan, daftar yang ditampilkan kosong.

Pada terminal atau baris perintah, jalankan perintah `list-tags-for-resource`. Misalnya, untuk melihat daftar kunci tag dan nilai tag untuk repositori bernama ARN `arn:aws:codecommit:us-east-2:111111111111::MyDemoRepo`

```
aws codecommit list-tags-for-resource --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo
```

Jika berhasil, perintah ini menampilkan informasi yang serupa dengan yang berikut:

```
{
  "tags": {
    "Status": "Secret",
    "Team": "Saanvi"
  }
}
```

Edit tag untuk repositori

Anda dapat mengubah nilai untuk tag yang terkait dengan repositori. Anda juga dapat mengubah nama kunci, yang setara dengan menghapus tag saat ini dan menambahkan tag yang berbeda dengan nama baru dan nilai yang sama dengan kunci lainnya. Perlu diingat bahwa ada batasan pada karakter yang dapat Anda gunakan di bidang kunci dan nilai. Untuk informasi selengkapnya, lihat [Batasan-batasan](#).

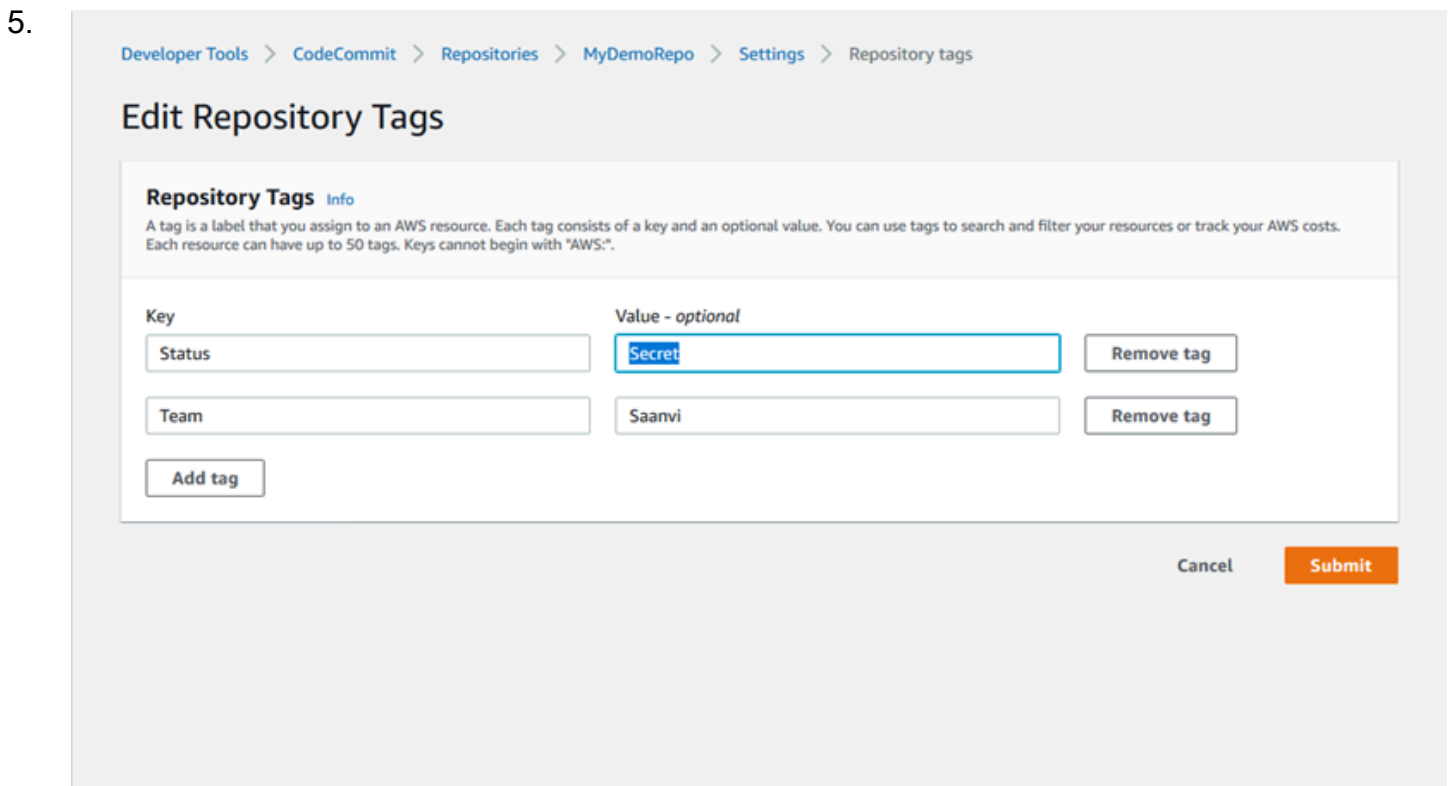
Important

Mengedit tag untuk repositori dapat memengaruhi akses ke repositori tersebut. Sebelum Anda mengedit nama (kunci) atau nilai tag untuk repositori, pastikan untuk meninjau kebijakan IAM yang mungkin menggunakan kunci atau nilai tag untuk mengontrol akses ke sumber daya seperti repositori. Untuk contoh kebijakan akses berbasis tag, lihat [Contoh 5: Tolak atau izinkan tindakan pada repositori dengan tag](#).

Edit tag untuk repositori (konsol)

Anda dapat menggunakan CodeCommit konsol untuk mengedit tag yang terkait dengan CodeCommit repositori.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin mengedit tag.
3. Pada panel navigasi, silakan pilih Pengaturan. Pilih Tag repositori.
4. Pilih Edit.



Lakukan salah satu hal berikut ini:

- Untuk mengubah tag, masukkan nama baru di Kunci. Mengubah nama tag adalah setara dengan menghapus tag dan menambahkan tag baru dengan nama kunci baru.
 - Untuk mengubah nilai sebuah tag, masukkan nilai baru. Jika Anda ingin mengubah nilai menjadi nol, hapus nilai saat ini dan biarkan bidang kosong.
6. Setelah Anda selesai mengedit tag, pilih Kirim.

Edit tag untuk repositori (AWS CLI)

Ikuti langkah-langkah ini untuk menggunakan AWS CLI untuk memperbarui tag untuk CodeCommit repositori. Anda dapat mengubah nilai untuk kunci yang ada, atau menambahkan kunci lain.

Pada terminal atau baris perintah, jalankan perintah `tag-resource`, dengan menentukan Amazon Resource Name (ARN) dari repositori tempat Anda ingin memperbarui tanda dan menentukan kunci tanda dan nilai tanda:

```
aws codecommit tag-resource --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo --tags Team=Li
```

Hapus sebuah tag dari repositori

Anda dapat menghapus satu atau lebih tag yang terkait dengan repositori. Menghapus tag tidak menghapus tag dari AWS sumber daya lain yang terkait dengan tag tersebut.

Important

Menghapus tag untuk repositori dapat memengaruhi akses ke repositori tersebut. Sebelum menghapus tag dari repositori, pastikan untuk meninjau kebijakan IAM yang mungkin menggunakan kunci atau nilai tag untuk mengontrol akses ke sumber daya seperti repositori. Untuk contoh kebijakan akses berbasis tag, lihat [Contoh 5: Tolak atau izinkan tindakan pada repositori dengan tag](#).

Hapus tag dari repositori (konsol)

Anda dapat menggunakan CodeCommit konsol untuk menghapus asosiasi antara tag dan CodeCommit repositori.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin menghapus tag.
3. Pada panel navigasi, silakan pilih Pengaturan. Pilih Tag repositori.
4. Pilih Edit.
5. Temukan tag yang ingin Anda hapus, lalu pilih Hapus tag.
6. Setelah Anda selesai menghapus tag, pilih Kirim.

Hapus sebuah tag dari repositori (AWS CLI)

Ikuti langkah-langkah ini untuk menggunakan AWS CLI untuk menghapus tag dari CodeCommit repositori. Menghapus sebuah tag tidak menghapusnya, tetapi hanya menghapus hubungan antara tag dan repositori.

Note

Jika Anda menghapus CodeCommit repositori, semua asosiasi tag akan dihapus dari repositori yang dihapus. Anda tidak perlu menghapus tag sebelum menghapus repositori.

Pada terminal atau baris perintah, jalankan perintah `untag-resource`, dengan menentukan Amazon Resource Name (ARN) dari repositori tempat Anda ingin menghapus tanda dan kunci tanda dari tanda yang ingin Anda hapus. *Misalnya, untuk menghapus tag pada repositori bernama `MyDemoRepo` dengan kunci tag `Status`:*

```
aws codecommit untag-resource --resource-arn arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo --tag-keys Status
```

Jika berhasil, perintah ini tidak mengembalikan apa pun. Untuk memverifikasi tanda yang terkait dengan repositori, jalankan perintah `list-tags-for-resource`.

Mengelola pemicu untuk repositori AWS CodeCommit

Anda dapat mengonfigurasi CodeCommit repositori sehingga mendorong kode atau peristiwa lain memicu tindakan, seperti mengirim pemberitahuan dari Amazon Simple Notification Service (Amazon SNS) atau menjalankan fungsi. AWS Lambda Anda dapat membuat hingga 10 pemicu untuk setiap CodeCommit repositori.

Pemicu biasanya dikonfigurasi untuk:

- Mengirim email ke pengguna berlangganan setiap kali seseorang mendorong ke repositori.
- Memberitahu sistem build eksternal untuk memulai membangun setelah seseorang mendorong ke cabang utama repositori.

Skenario seperti memberitahukan sistem build eksternal memerlukan menulis fungsi Lambda untuk berinteraksi dengan aplikasi lain. Skenario email hanya memerlukan pembuatan sebuah topik Amazon SNS.

Topik ini menunjukkan cara menyetel izin yang memungkinkan CodeCommit untuk memicu tindakan di Amazon SNS dan Lambda. Ini juga mencakup link ke contoh untuk membuat, mengedit, menguji, dan menghapus pemicu.

Topik

- [Buat sumber daya dan tambahkan izin untuk CodeCommit](#)
- [Contoh: Buat AWS CodeCommit pemicu untuk topik Amazon SNS](#)
- [Contoh: Buat AWS CodeCommit pemicu untuk suatu AWS Lambda fungsi](#)
- [Contoh: Buat pemicu AWS CodeCommit untuk AWS Lambda fungsi yang ada](#)
- [Edit pemicu untuk repositori AWS CodeCommit](#)
- [Uji pemicu untuk repositori AWS CodeCommit](#)
- [Hapus pemicu dari repositori AWS CodeCommit](#)

Buat sumber daya dan tambahkan izin untuk CodeCommit

Anda dapat mengintegrasikan topik Amazon SNS dan fungsi Lambda dengan pemicu CodeCommit, tetapi Anda harus terlebih dahulu membuat dan kemudian mengonfigurasi sumber daya dengan kebijakan yang memberikan izin untuk berinteraksi dengan sumber daya CodeCommit tersebut. Anda harus membuat sumber daya Wilayah AWS sama dengan CodeCommit repositori. Sebagai contoh, jika repositori berada di US East (Ohio) (us-east-2), topik Amazon SNS atau fungsi Lambda harus di US East (Ohio).

- Untuk topik Amazon SNS, Anda tidak perlu mengonfigurasi kebijakan atau izin IAM tambahan jika topik Amazon SNS dibuat menggunakan akun yang sama dengan repositori. CodeCommit Anda dapat membuat CodeCommit pemicu segera setelah Anda membuat dan berlangganan topik Amazon SNS.
 - Untuk informasi selengkapnya tentang membuat topik di Amazon SNS, lihat [Dokumentasi Amazon SNS](#).
 - Untuk informasi tentang menggunakan Amazon SNS untuk mengirim pesan ke antrian Amazon SQS, lihat [Mengirim Pesan ke Antrian Amazon SQS](#) di Panduan Developer Amazon SNS.

- Untuk informasi tentang menggunakan Amazon SNS untuk memanggil fungsi Lambda, lihat [Memanggil Fungsi Lambda](#) dalam Panduan Developer Amazon SNS.
- Jika Anda ingin mengonfigurasi pemicu untuk menggunakan topik Amazon SNS di AWS akun lain, Anda harus terlebih dahulu mengonfigurasi topik tersebut dengan kebijakan yang memungkinkan CodeCommit untuk mempublikasikan topik tersebut. Untuk informasi selengkapnya, lihat [Contoh 1: Buat kebijakan yang memungkinkan akses lintas-akun ke topik Amazon SNS](#).
- Anda dapat mengkonfigurasi fungsi Lambda dengan menciptakan pemicu di konsol Lambda sebagai bagian dari fungsi. Ini adalah metode yang paling sederhana, karena pemicu yang dibuat di konsol Lambda secara otomatis menyertakan izin yang diperlukan CodeCommit untuk menjalankan fungsi Lambda. Jika Anda membuat pemicu CodeCommit, Anda harus menyertakan kebijakan CodeCommit untuk mengizinkan menjalankan fungsi. Untuk informasi selengkapnya, lihat [Buatlah sebuah pemicu untuk sebuah fungsi Lambda yang sudah ada](#) dan [Contoh 3: Membuat kebijakan untuk AWS Lambda integrasi dengan CodeCommit pemicu](#).

Contoh: Buat AWS CodeCommit pemicu untuk topik Amazon SNS

Anda dapat membuat pemicu untuk CodeCommit repositori sehingga peristiwa di repositori tersebut memicu notifikasi dari topik Amazon Simple Notification Service (Amazon SNS). Anda mungkin ingin membuat pemicu untuk topik Amazon SNS untuk memungkinkan pengguna untuk berlangganan notifikasi tentang peristiwa repositori, seperti penghapusan cabang. Anda juga dapat memanfaatkan integrasi topik Amazon SNS dengan layanan lain, seperti Amazon Simple Queue Service (Amazon Simple Queue Service) dan AWS Lambda

Note

- Anda harus mengarahkan pemicu ke topik Amazon SNS yang ada yang tindakan yang diambil dalam menanggapi peristiwa repositori. Untuk informasi selengkapnya tentang membuat dan berlangganan topik Amazon SNS, lihat [Memulai Amazon Simple Notification Service](#).
- Topik Amazon SNS FIFO (masuk pertama, keluar pertama) tidak didukung untuk pemicu CodeCommit

Topik

- [Buat pemicu ke topik Amazon SNS untuk CodeCommit repositori \(konsol\)](#)

- [Buat pemicu ke topik Amazon SNS untuk CodeCommit repositori \(AWS CLI\)](#)

Buat pemicu ke topik Amazon SNS untuk CodeCommit repositori (konsol)

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori tempat Anda ingin membuat pemicu untuk peristiwa repositori.
3. Di panel navigasi untuk repositori, pilih Pengaturan, lalu pilih Pemicu.
4. Pilih Buat pemicu, dan kemudian lakukan hal berikut:
 - Di nama Pemicu, masukkan nama untuk pemicu (misalnya, *MyFirstTrigger*).
 - Di Peristiwa, pilih peristiwa repositori yang memicu topik Amazon SNS untuk mengirim notifikasi.

Jika Anda memilih Semua peristiwa repositori, Anda tidak dapat memilih peristiwa lainnya. Untuk memilih sebuah subset peristiwa, hapus Semua peristiwa repositori, lalu pilih satu atau lebih peristiwa dari daftar. Misalnya, jika Anda ingin pemicu berjalan hanya ketika pengguna membuat cabang atau tag di CodeCommit repositori, hapus Semua peristiwa repositori, lalu pilih Buat cabang atau tag.

- Jika Anda ingin pemicu untuk diterapkan ke semua cabang repositori, di Cabang, biarkan pilihan kosong, karena opsi default ini berlaku pemicu untuk semua cabang secara otomatis. Jika Anda ingin pemicu ini berlaku untuk cabang tertentu saja, pilih hingga 10 nama cabang dari daftar cabang repositori.
- Di Pilih layanan yang akan digunakan, pilih Amazon SNS.
- Di Amazon SNS, pilih nama topik dari daftar atau masukkan ARN untuk topik tersebut.

Note

Topik Amazon SNS FIFO (masuk pertama, keluar pertama) tidak didukung untuk pemicu. CodeCommit Anda harus memilih topik Amazon SNS yang jenisnya telah ditetapkan untuk Standar. Jika Anda ingin menggunakan topik Amazon SNS FIFO, Anda harus mengonfigurasi aturan Amazon Eventbridge untuk CodeCommit acara yang memiliki topik SNS FIFO yang dikonfigurasi sebagai targetnya.

- Di Data khusus, berikan informasi opsional yang ingin Anda sertakan dalam notifikasi yang dikirim oleh topik Amazon SNS (misalnya, developer nama saluran IRC digunakan saat

membahas pengembangan di repositori ini). Bidang ini adalah string. Ini tidak dapat digunakan untuk melewati parameter dinamis.

5. (Opsional) Pilih Pemicu uji. Langkah ini membantu Anda mengonfirmasi telah mengonfigurasi akses dengan benar antara CodeCommit dan topik Amazon SNS. Menggunakan topik Amazon SNS untuk mengirim notifikasi uji menggunakan data dari repositori Anda, jika tersedia. Jika tidak ada data nyata yang tersedia, notifikasi uji berisi data sampel.
6. Pilih Buat pemicu untuk menyelesaikan pembuatan pemicu.

Buat pemicu ke topik Amazon SNS untuk CodeCommit repositori (AWS CLI)

Anda juga dapat menggunakan baris perintah untuk membuat pemicu untuk topik Amazon SNS dalam menanggapi peristiwa CodeCommit repositori, seperti ketika seseorang mendorong komit ke repositori Anda.

Buatlah sebuah pemicu untuk topik Amazon SNS

1. Buka editor teks biasa dan buatlah file JSON yang menentukan:

- Nama topik Amazon SNS.

Note

Topik Amazon SNS FIFO (masuk pertama, keluar pertama) tidak didukung untuk pemicu. CodeCommit Anda harus memilih topik Amazon SNS yang jenisnya telah ditetapkan untuk Standar. Jika Anda ingin menggunakan topik Amazon SNS FIFO, Anda harus mengonfigurasi aturan Amazon Eventbridge untuk CodeCommit acara yang memiliki topik SNS FIFO yang dikonfigurasi sebagai targetnya.

- Repositori dan cabang yang ingin Anda pantau dengan pemicu ini. (Jika Anda tidak menentukan cabang apa pun, pemicu berlaku untuk semua cabang di repositori.)
- Peristiwa yang mengaktifkan pemicu ini.

Simpan file tersebut.

Misalnya, untuk membuat pemicu repositori bernama *MyDemoRepo* menerbitkan semua peristiwa repositori ke topik Amazon SNS bernama *MySnstopic* untuk dua cabang, *main* dan *preprod*:

```
{
  "repositoryName": "MyDemoRepo",
  "triggers": [
    {
      "name": "MyFirstTrigger",
      "destinationArn": "arn:aws:sns:us-east-2:111122223333:MySNSTopic",
      "customData": "",
      "branches": [
        "main", "preprod"
      ],
      "events": [
        "all"
      ]
    }
  ]
}
```

Harus ada blok pemicu di JSON untuk setiap pemicu untuk sebuah repositori. Untuk membuat lebih dari satu pemicu untuk repositori, termasuk lebih dari satu blok pemicu di JSON. Ingat bahwa semua pemicu yang dibuat dalam file ini adalah untuk repositori yang telah ditentukan. Anda tidak dapat membuat pemicu untuk beberapa repositori dalam file JSON tunggal. Misalnya, jika Anda ingin membuat dua pemicu untuk repositori, Anda dapat membuat file JSON dengan dua blok pemicu. Pada contoh berikut, tidak ada cabang yang ditentukan untuk pemicu kedua, sehingga pemicu berlaku untuk semua cabang:

```
{
  "repositoryName": "MyDemoRepo",
  "triggers": [
    {
      "name": "MyFirstTrigger",
      "destinationArn": "arn:aws:sns:us-east-2:111122223333:MySNSTopic",
      "customData": "",
      "branches": [
        "main", "preprod"
      ],

```



```
        "events": [
            "all"
        ]
    },
    {
        "name": "MySecondTrigger",
        "destinationArn": "arn:aws:sns:us-east-2:111122223333:MySNSTopic2",
        "customData": "",
        "branches": [],
        "events": [
            "updateReference", "deleteReference"
        ]
    }
]
}
```

Anda dapat membuat pemicu untuk peristiwa yang Anda tentukan, seperti ketika commit didorong ke repositori. Jenis-jenis peristiwa meliputi:

- `all` untuk semua peristiwa di repositori dan cabang yang telah ditentukan.
- `updateReference` untuk ketika commit didorong ke repositori dan cabang yang ditentukan.
- `createReference` untuk ketika cabang baru atau tag dibuat dalam repositori yang ditentukan.
- `deleteReference` untuk ketika cabang atau tag dihapus dalam repositori yang ditentukan.

Note

Anda dapat menggunakan lebih dari satu jenis peristiwa di sebuah pemicu. Namun, jika Anda menentukan `all`, Anda tidak dapat menentukan peristiwa lainnya.

Untuk menampilkan daftar lengkap jenis peristiwa yang valid, pada terminal atau prompt perintah, masukkan `aws codecommit put-repository-triggers help`.

Selain itu, Anda dapat menyertakan sebuah string dalam `customData` (misalnya, developer nama saluran IRC digunakan saat membahas pengembangan di repositori ini). Bidang ini adalah string. Ini tidak dapat digunakan untuk melewati parameter dinamis. String ini ditambahkan sebagai atribut ke CodeCommit JSON yang dikembalikan sebagai respons terhadap pemicu.

- (Opsional) Pada terminal atau prompt perintah, jalankan perintah `test-repository-triggers`. Uji ini menggunakan data sampel dari repositori (atau menghasilkan data sampel jika tidak ada data yang tersedia) untuk mengirim notifikasi kepada pelanggan topik Amazon SNS. Misalnya, berikut ini digunakan untuk menguji bahwa JSON dalam file pemicu bernama `trigger.json` valid dan yang CodeCommit dapat mempublikasikan ke topik Amazon SNS:

```
aws codecommit test-repository-triggers --cli-input-json file://trigger.json
```

Jika berhasil, perintah ini menampilkan informasi yang serupa dengan yang berikut:

```
{
  "successfulExecutions": [
    "MyFirstTrigger"
  ],
  "failedExecutions": []
}
```

- Pada terminal atau command prompt, jalankan `put-repository-triggers` perintah untuk membuat pemicu di CodeCommit. Misalnya, untuk menggunakan file JSON yang bernama `trigger.json` untuk membuat pemicu:

```
aws codecommit put-repository-triggers --cli-input-json
file://trigger.json
```

Perintah ini mengembalikan [ID konfigurasi](#), yang serupa dengan yang berikut ini:

```
{
  "configurationId": "0123456-I-AM-AN-EXAMPLE"
}
```

- Untuk menampilkan konfigurasi pemicu, jalankan perintah `get-repository-triggers`, dengan menentukan nama repositori:

```
aws codecommit get-repository-triggers --repository-name MyDemoRepo
```

Perintah ini mengembalikan struktur semua pemicu dikonfigurasi untuk repositori, yang serupa dengan berikut ini:

```
{
  "configurationId": "0123456-I-AM-AN-EXAMPLE",
```

```

"triggers": [
  {
    "events": [
      "all"
    ],
    "destinationArn": "arn:aws:sns:us-east-2:111122223333:MySNSTopic",
    "branches": [
      "main",
      "preprod"
    ],
    "name": "MyFirstTrigger",
    "customData": "Project ID 12345"
  }
]
}

```

- Untuk menguji fungsi dari pemicu itu sendiri, buat dan doronglah commit ke repositori di mana Anda mengkonfigurasi pemicu. Anda harus melihat respons dari topik Amazon SNS. Misalnya, jika Anda mengkonfigurasi topik Amazon SNS untuk mengirim email, Anda akan melihat email dari Amazon SNS di akun email yang berlangganan topik tersebut.

Berikut ini adalah contoh output dari email yang dikirim dari Amazon SNS sebagai tanggapan terhadap push ke repositori: CodeCommit

```

{
  "Records": [
    {
      "awsRegion": "us-east-2",
      "codecommit": {
        "references": [
          {
            "commit": "317f8570EXAMPLE",
            "created": true,
            "ref": "refs/heads/NewBranch"
          },
          {
            "commit": "4c925148EXAMPLE",
            "ref": "refs/heads/preprod",
          }
        ]
      },
      "eventId": "11111-EXAMPLE-ID",
      "eventName": "ReferenceChange",
    }
  ]
}

```

```
"eventPartNumber":1,
"eventSource":"aws:codecommit",
"eventSourceARN":"arn:aws:codecommit:us-east-2:111122223333:MyDemoRepo",
"eventTime":"2016-02-09T00:08:11.743+0000",
"eventTotalParts":1,
"eventTriggerConfigId":"0123456-I-AM-AN-EXAMPLE",
"eventTriggerName":"MyFirstTrigger",
"eventVersion":"1.0",
"customData":"Project ID 12345",
"userIdentityARN":"arn:aws:iam:111122223333:user/JaneDoe-CodeCommit",
}
]
}
```

Contoh: Buat AWS CodeCommit pemacu untuk suatu AWS Lambda fungsi

Anda dapat membuat pemacu untuk CodeCommit repositori sehingga peristiwa di repositori memanggil fungsi Lambda. Dalam contoh ini, Anda membuat fungsi Lambda yang mengembalikan URL yang digunakan untuk mengkloning repositori ke log Amazon. CloudWatch

Topik

- [Buat fungsi Lambda](#)
- [Tampilkan pemacu untuk fungsi Lambda di repositori AWS CodeCommit](#)

Buat fungsi Lambda

Saat Anda menggunakan konsol Lambda untuk membuat fungsi, Anda juga dapat membuat CodeCommit pemacu untuk fungsi Lambda. Langkah-langkah berikut termasuk contoh fungsi Lambda. Sampel tersedia dalam dua bahasa: JavaScript dan Python. Fungsi mengembalikan URL yang digunakan untuk mengkloning repositori ke log. CloudWatch

Buatlah sebuah fungsi Lambda menggunakan cetak biru Lambda


1. Masuk ke AWS Management Console dan buka AWS Lambda konsol di <https://console.aws.amazon.com/lambda/>.
2. Pada halaman Fungsi Lambda, pilih Buat fungsi. (Jika Anda belum pernah menggunakan Lambda sebelumnya, pilih Mulai Sekarang.)

3. Pilih halaman Buat fungsi, pilih Penulis dari scratch. Di Nama Fungsi, berikan nama untuk fungsi tersebut, misalnya *MyLambdaFunctionforCodeCommit*. Di Runtime, pilih bahasa yang ingin Anda gunakan untuk menulis fungsi Anda, lalu pilih Create function.
4. Pada tab Konfigurasi, pilih Tambah pemicu.
5. Di konfigurasi Trigger, pilih CodeCommit dari daftar drop-down layanan.

Lambda > Add trigger

Add trigger

Trigger configuration

 CodeCommit
aws developer-tools git

Repository name
Select the repository to add a trigger to.

MyDemoRepo

Trigger name
Provide a name for the trigger that will invoke this function.

MyLambdaFunctionTrigger

Events
Choose one or more events to listen for. If you choose "All repository events", you cannot choose other event types.

Branch names
This trigger will be configured for all repository branches and tags by default. For a more specific configuration, choose up to 10 branches. If you choose "All branches", you cannot choose specific branches.

Custom data - optional
Custom data is additional contextual information used to distinguish this trigger from other triggers that run for the same event, refer to external resources, or group triggers from different repositories. For example, you could include the channel ID # for a chat room used by your team to collaborate on development.

#1

Lambda will add the necessary permissions for AWS CodeCommit to invoke your Lambda function from this trigger.
[Learn more](#) about the Lambda permissions model.

- Di Nama repositori, pilih nama repositori di mana Anda ingin mengkonfigurasi pemicu yang menggunakan fungsi Lambda dalam menanggapi peristiwa repositori.

- Di nama Pemicu, masukkan nama untuk pemicu (misalnya, *MyLambdaFunctionTrigger*).
- Di Peristiwa, pilih peristiwa repositori yang memicu fungsi Lambda. Jika Anda memilih Semua peristiwa repositori, Anda tidak dapat memilih peristiwa lainnya. Jika Anda ingin memilih subset dari peristiwa, bersihkan Semua peristiwa repositori, dan kemudian pilih peristiwa yang Anda inginkan dari daftar. Misalnya, jika Anda ingin pemicu berjalan hanya ketika pengguna membuat tag atau cabang di AWS CodeCommit repositori, hapus Semua peristiwa repositori, lalu pilih Buat cabang atau tag.
- Jika Anda ingin pemicu untuk diterapkan ke semua cabang repositori, di Cabang, pilih Semua cabang. Jika tidak, pilih Cabang khusus. Cabang default untuk repositori ditambahkan secara default. Anda dapat menyimpan atau menghapus cabang ini dari daftar. Pilih hingga 10 nama cabang dari daftar cabang repositori.
- (Opsional) Di Data khusus, masukkan informasi yang ingin Anda sertakan dalam fungsi Lambda (misalnya, nama saluran IRC yang digunakan oleh developer untuk membahas pengembangan di repositori tersebut). Bidang ini adalah string. Ini tidak dapat digunakan untuk melewati parameter dinamis.

Pilih Tambahkan.

6. Pada halaman Konfigurasi, di Kode Fungsi, di Jenis entri kode, pilih Edit kode sebaris.. Di Runtime, pilih Node.js. Jika anda ingin membuat sebuah sampel fungsi Python, pilih Python.
7. Di Jenis entri kode, pilih Edit kode inline, dan kemudian gganti kode hello world dengan salah satu dari dua contoh berikut.

Untuk Node.js:

```
import {
  CodeCommitClient,
  GetRepositoryCommand,
} from "@aws-sdk/client-codecommit";

const codecommit = new CodeCommitClient({ region: "your-region" });

/**
 * @param {{ Records: { codecommit: { references: { ref: string }[] },
  eventSourceARN: string }[] } event
 */
export const handler = async (event) => {
  // Log the updated references from the event
  const references = event.Records[0].codecommit.references.map(
```

```
(reference) => reference.ref,
);
console.log("References:", references);

// Get the repository from the event and show its git clone URL
const repository = event.Records[0].eventSourceARN.split(":")[5];
const params = {
  repositoryName: repository,
};

try {
  const data = await codecommit.send(new GetRepositoryCommand(params));
  console.log("Clone URL:", data.repositoryMetadata.cloneUrlHttp);
  return data.repositoryMetadata.cloneUrlHttp;
} catch (error) {
  console.error("Error:", error);
  throw new Error(
    `Error getting repository metadata for repository ${repository}`,
  );
}
};
```

Untuk Python:

```
import json
import boto3

codecommit = boto3.client("codecommit")

def lambda_handler(event, context):
    # Log the updated references from the event
    references = {
        reference["ref"]
        for reference in event["Records"][0]["codecommit"]["references"]
    }
    print("References: " + str(references))

    # Get the repository from the event and show its git clone URL
    repository = event["Records"][0]["eventSourceARN"].split(":")[5]
    try:
        response = codecommit.get_repository(repositoryName=repository)
```



```
print("Clone URL: " + response["repositoryMetadata"]["cloneUrlHttp"])
return response["repositoryMetadata"]["cloneUrlHttp"]
except Exception as e:
    print(e)
    print(
        "Error getting repository {}. Make sure it exists and that your
repository is in the same region as this function.".format(
            repository
        )
    )
    raise e
```

8. Di tab Izin, di peran Eksekusi, pilih peran untuk membukanya di konsol IAM. Edit kebijakan terlampir untuk menambahkan GetRepository izin untuk repositori yang ingin Anda gunakan pemicu.

Tampilkan pemicu untuk fungsi Lambda di repositori AWS CodeCommit

Setelah Anda membuat fungsi Lambda, Anda dapat melihat dan menguji pemicu di AWS CodeCommit. Pengujian pemicu menjalankan fungsi dalam menanggapi peristiwa repositori yang Anda tentukan.

Untuk menampilkan dan menguji pemicu untuk fungsi Lambda

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori yang Anda ingin tampilkan pemicu.
3. Di panel navigasi untuk repositori, pilih Pengaturan, lalu pilih Pemicu.
4. Tinjau daftar pemicu untuk repositori. Anda akan melihat pemicu yang Anda buat di konsol Lambda. Pilih dari daftar dan kemudian pilih Pemicu uji. Opsi ini mencoba untuk memanggil fungsi dengan data sampel tentang repositori Anda, termasuk terbaru ID commit untuk repositori. (Jika tidak ada riwayat commit, nilai sampel yang terdiri dari nol dihasilkan sebagai gantinya.) Ini membantu Anda mengonfirmasi bahwa Anda telah mengonfigurasi akses dengan benar antara AWS CodeCommit dan fungsi Lambda.
5. Untuk lebih jauh memverifikasi fungsi pemicu, membuat dan mendorong sebuah commit ke repositori di mana Anda dikonfigurasi pemicu. Anda akan melihat respons dari fungsi Lambda pada tab Pemantauan untuk fungsi di konsol Lambda. Dari tab Monitoring, pilih Lihat log masuk CloudWatch. CloudWatch Konsol terbuka di tab baru dan menampilkan acara untuk fungsi Anda.

Pilih aliran log dari daftar yang sesuai dengan waktu Anda mendorong commit Anda. Anda akan melihat data peristiwa yang serupa dengan yang berikut ini:

```
START RequestId: 70afdc9a-EXAMPLE Version: $LATEST
2015-11-10T18:18:28.689Z 70afdc9a-EXAMPLE References: [ 'refs/heads/main' ]
2015-11-10T18:18:29.814Z 70afdc9a-EXAMPLE Clone URL: https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
END RequestId: 70afdc9a-EXAMPLE
REPORT RequestId: 70afdc9a-EXAMPLE Duration: 1126.87 ms Billed Duration: 1200 ms
Memory Size: 128 MB Max Memory Used: 14 MB
```

Contoh: Buat pemicu AWS CodeCommit untuk AWS Lambda fungsi yang ada

Cara termudah untuk membuat pemicu yang memanggil fungsi Lambda adalah membuat pemicu itu di konsol Lambda. Integrasi bawaan ini memastikan bahwa CodeCommit memiliki izin yang diperlukan untuk menjalankan fungsi. Untuk menambahkan pemicu fungsi Lambda yang ada, buka konsol Lambda, dan pilih fungsi tersebut. Pada tab Pemicu untuk fungsi, ikuti langkah-langkah di Tambahkan pemicu. Langkah-langkah ini mirip dengan yang ada di [Buat fungsi Lambda](#).

Anda juga dapat membuat pemicu untuk fungsi Lambda di repositori. CodeCommit Melakukan hal demikian mensyaratkan Anda memilih fungsi Lambda yang ada untuk dipanggil. Hal ini juga mengharuskan Anda secara manual mengkonfigurasi izin yang diperlukan CodeCommit untuk menjalankan fungsi.

Topik

- [Konfigurasi izin secara manual CodeCommit untuk memungkinkan menjalankan fungsi Lambda](#)
- [Buat pemicu untuk fungsi Lambda di CodeCommit repositori \(konsol\)](#)
- [Buat pemicu ke fungsi Lambda untuk CodeCommit repositori \(AWS CLI\)](#)

Konfigurasi izin secara manual CodeCommit untuk memungkinkan menjalankan fungsi Lambda

Jika Anda membuat pemicu CodeCommit yang memanggil fungsi Lambda, Anda harus mengonfigurasi izin secara manual yang CodeCommit memungkinkan untuk menjalankan fungsi Lambda. Untuk menghindari konfigurasi manual ini, pertimbangkan untuk membuat pemicu untuk fungsi tersebut di konsol Lambda sebagai gantinya.

Untuk CodeCommit memungkinkan menjalankan fungsi Lambda

1. Buka editor teks biasa dan buat file JSON yang menentukan nama fungsi Lambda, detail CodeCommit repositori, dan tindakan yang ingin Anda izinkan di Lambda, mirip dengan yang berikut ini:

```
{
  "FunctionName": "MyCodeCommitFunction",
  "StatementId": "1",
  "Action": "lambda:InvokeFunction",
  "Principal": "codecommit.amazonaws.com",
  "SourceArn": "arn:aws:codecommit:us-east-1:111122223333:MyDemoRepo",
  "SourceAccount": "111122223333"
}
```

2. Simpan file sebagai file JSON dengan nama yang mudah diingat (misalnya, *AllowAccessfromMyDemoRepo.json*).
3. Dengan menggunakan file JSON yang baru saja Anda buat, di terminal (Linux, MacOS, atau Unix) atau baris perintah (Windows), jalankan perintah `aws lambda add-permissions` untuk menambahkan izin ke kebijakan sumber daya yang terkait dengan fungsi Lambda Anda:

```
aws lambda add-permission --cli-input-json file://AllowAccessfromMyDemoRepo.json
```

Perintah ini mengembalikan JSON dari pernyataan kebijakan Anda hanya menambahkan, yang serupa dengan berikut ini:

```
{
  "Statement": "{ \"Condition\": { \"StringEquals\": { \"AWS:SourceAccount\": \"111122223333\" }, \"ArnLike\": { \"AWS:SourceArn\": \"arn:aws:codecommit:us-east-1:111122223333:MyDemoRepo\" } }, \"Action\": [ \"lambda:InvokeFunction\" ], \"Resource\": \"arn:aws:lambda:us-east-1:111122223333:function:MyCodeCommitFunction\", \"Effect\": \"Allow\", \"Principal\": { \"Service\": \"codecommit.amazonaws.com\" }, \"Sid\": \"1\" } }
```

Untuk informasi selengkapnya tentang kebijakan sumber daya untuk fungsi Lambda, lihat [AddPermission](#) dan [Model Peristiwa Tarik/Dorong di Panduan Pengguna](#). AWS Lambda

4. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.

5. Di panel navigasi Dasbor, pilih Peran, dan di daftar peran, pilih *lambda_basic_execution*.
6. Pada halaman ringkasan untuk peran tersebut, pilih tab Izin, dan di Kebijakan inline, pilih Buat Kebijakan Peran.
7. Pada halaman Atur Izin, pilih Generator Kebijakan, dan kemudian pilih Pilih.
8. Di halaman Edit Izin, lakukan hal berikut:
 - Di Efek, pilih Izinkan.
 - Di Layanan AWS , pilih AWS CodeCommit.
 - Di Tindakan, pilih GetRepository.
 - Di Amazon Resource Name (ARN), masukkan ARN untuk repositori (misalnya, `arn:aws:codecommit:us-east-1:111122223333:MyDemoRepo`).

Pilih Tambah Pernyataan, dan kemudian pilih Langkah Selanjutnya.

9. Pada halaman Tinjau kebijakan, pilih Terapkan Kebijakan.

Pernyataan kebijakan Anda akan terlihat serupa dengan contoh berikut:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt11111111",
      "Effect": "Allow",
      "Action": [
        "codecommit:GetRepository"
      ],
      "Resource": [
        "arn:aws:codecommit:us-east-1:111122223333:MyDemoRepo"
      ]
    }
  ]
}
```

Buat pemicu untuk fungsi Lambda di CodeCommit repositori (konsol)

Setelah Anda membuat fungsi Lambda, Anda dapat membuat pemicu CodeCommit yang menjalankan fungsi sebagai respons terhadap peristiwa repositori yang Anda tentukan.

Note

Sebelum Anda berhasil menguji atau menjalankan pemicu untuk contoh, Anda harus mengonfigurasi kebijakan yang memungkinkan CodeCommit untuk memanggil fungsi dan fungsi Lambda untuk mendapatkan informasi tentang repositori. Untuk informasi selengkapnya, lihat [Untuk CodeCommit memungkinkan menjalankan fungsi Lambda](#).

Buatlah sebuah pemicu untuk sebuah fungsi Lambda

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori tempat Anda ingin membuat pemicu untuk peristiwa repositori.
3. Di panel navigasi untuk repositori, pilih Pengaturan, lalu pilih Pemicu.
4. Pilih Buat pemicu.
5. Di Buat pemicu, lakukan hal berikut:
 - Di nama Pemicu, masukkan nama untuk pemicu (misalnya, *MyLambdaFunctionTrigger*).
 - Di Peristiwa, pilih peristiwa repositori yang memicu fungsi Lambda.

Jika Anda memilih Semua peristiwa repositori, Anda tidak dapat memilih peristiwa lainnya. Jika Anda ingin memilih subset dari peristiwa, bersihkan Semua peristiwa repositori, dan kemudian pilih peristiwa yang Anda inginkan dari daftar. Misalnya, jika Anda ingin pemicu berjalan hanya ketika pengguna membuat tag atau cabang di CodeCommit repositori, hapus Semua peristiwa repositori, lalu pilih Buat cabang atau tag.

- Jika Anda ingin pemicu untuk diterapkan ke semua cabang repositori, di Cabang, biarkan pilihan kosong, karena opsi default ini berlaku pemicu untuk semua cabang secara otomatis. Jika Anda ingin pemicu ini berlaku untuk cabang tertentu saja, pilih hingga 10 nama cabang dari daftar cabang repositori.
- Di Pilih layanan yang akan digunakan, pilih AWS Lambda.
- Di Fungsi Lambda, pilih nama fungsi dari daftar, atau masukkan ARN untuk fungsi tersebut.
- (Opsional) Di Data khusus, masukkan informasi yang ingin Anda sertakan dalam fungsi Lambda (misalnya, nama saluran IRC yang digunakan oleh developer untuk membahas pengembangan di repositori tersebut). Bidang ini adalah string. Ini tidak dapat digunakan untuk melewati parameter dinamis.

6. (Opsional) Pilih Pemicu uji. Opsi ini mencoba untuk memanggil fungsi dengan data sampel tentang repositori Anda, termasuk terbaru ID commit untuk repositori. (Jika tidak ada riwayat commit, nilai sampel yang terdiri dari nol dihasilkan sebagai gantinya.) Ini membantu Anda mengonfirmasi bahwa Anda telah mengonfigurasi akses dengan benar antara CodeCommit dan fungsi Lambda.
7. Pilih Buat pemicu untuk menyelesaikan pembuatan pemicu.
8. Untuk memverifikasi fungsi pemicu, buat dan doronglah commit ke repositori di mana Anda mengkonfigurasi pemicu. Anda akan melihat respons dari fungsi Lambda pada tab Pemantauan untuk fungsi di konsol Lambda.

Buat pemicu ke fungsi Lambda untuk CodeCommit repositori ()AWS CLI

Anda juga dapat menggunakan baris perintah untuk membuat pemicu fungsi Lambda sebagai respons terhadap peristiwa CodeCommit repositori, seperti ketika seseorang mendorong komit ke repositori Anda.

Buatlah sebuah pemicu untuk sebuah fungsi Lambda

1. Buka editor teks biasa dan buatlah file JSON yang menentukan:
 - Nama fungsi Lambda.
 - Repositori dan cabang yang ingin Anda pantau dengan pemicu ini. (Jika Anda tidak menentukan cabang apa pun, pemicu berlaku untuk semua cabang di repositori.)
 - Peristiwa yang mengaktifkan pemicu ini.

Simpan file tersebut.

Misalnya, jika Anda ingin membuat pemicu untuk repositori bernama yang menerbitkan semua peristiwa repositori ke fungsi Lambda MyDemoRepo yang diberi nama MyCodeCommitFunction untuk dua cabang, main dan preprod:

```
{
  "repositoryName": "MyDemoRepo",
  "triggers": [
    {
      "name": "MyLambdaFunctionTrigger",
      "destinationArn": "arn:aws:lambda:us-east-1:111122223333:function:MyCodeCommitFunction",
```

```

        "customData": "",
        "branches": [
            "main", "preprod"
        ],
        "events": [
            "all"
        ]
    }
]
}

```

Harus ada blok pemicu di JSON untuk setiap pemicu untuk sebuah repositori. Untuk membuat lebih dari satu pemicu untuk repositori, termasuk blok tambahan di JSON. Ingat bahwa semua pemicu yang dibuat dalam file ini adalah untuk repositori yang telah ditentukan. Anda tidak dapat membuat pemicu untuk beberapa repositori dalam file JSON tunggal. Misalnya, jika Anda ingin membuat dua pemicu untuk repositori, Anda dapat membuat file JSON dengan dua blok pemicu. Pada contoh berikut, tidak ada cabang yang ditentukan di blok pemicu kedua, sehingga pemicu berlaku untuk semua cabang:

```

{
  "repositoryName": "MyDemoRepo",
  "triggers": [
    {
      "name": "MyLambdaFunctionTrigger",
      "destinationArn": "arn:aws:lambda:us-
east-1:111122223333:function:MyCodeCommitFunction",
      "customData": "",
      "branches": [
        "main", "preprod"
      ],
      "events": [
        "all"
      ]
    },
    {
      "name": "MyOtherLambdaFunctionTrigger",
      "destinationArn": "arn:aws:lambda:us-
east-1:111122223333:function:MyOtherCodeCommitFunction",
      "customData": "",
      "branches": [],
      "events": [

```

```
        "updateReference", "deleteReference"
    ]
}
]
```

Anda dapat membuat pemacu untuk peristiwa yang Anda tentukan, seperti ketika commit didorong ke repositori. Jenis-jenis peristiwa meliputi:

- `all` untuk semua peristiwa di repositori dan cabang yang telah ditentukan.
- `updateReference` untuk ketika commit didorong ke repositori dan cabang yang ditentukan.
- `createReference` untuk ketika cabang baru atau tag dibuat dalam repositori yang ditentukan.
- `deleteReference` untuk ketika cabang atau tag dihapus dalam repositori yang ditentukan.

Note

Anda dapat menggunakan lebih dari satu jenis peristiwa di sebuah pemacu. Namun, jika Anda menentukan `all`, Anda tidak dapat menentukan peristiwa lainnya.

Untuk menampilkan daftar lengkap jenis peristiwa yang valid, pada terminal atau prompt perintah, masukkan `aws codecommit put-repository-triggers help`.

Selain itu, Anda dapat menyertakan sebuah string dalam `customData` (misalnya, developer nama saluran IRC digunakan saat membahas pengembangan di repositori ini). Bidang ini adalah string. Ini tidak dapat digunakan untuk melewati parameter dinamis. String ini ditambahkan sebagai atribut ke CodeCommit JSON yang dikembalikan sebagai respons terhadap pemacu.

2. (Opsional) Pada terminal atau prompt perintah, jalankan perintah `test-repository-triggers`. Misalnya, berikut ini digunakan untuk menguji bahwa file JSON bernama *trigger.json* valid dan yang dapat memicu fungsi CodeCommit Lambda. Uji ini menggunakan data sampel untuk memicu fungsi jika tidak ada data nyata tersedia.

```
aws codecommit test-repository-triggers --cli-input-json file://trigger.json
```

Jika berhasil, perintah ini menampilkan informasi yang serupa dengan yang berikut:


```
{
  "successfulExecutions": [
    "MyLambdaFunctionTrigger"
  ],
  "failedExecutions": []
}
```

3. Pada terminal atau command prompt, jalankan `put-repository-triggers` perintah untuk membuat pemicu di CodeCommit. Misalnya, untuk menggunakan file JSON yang bernama *trigger.json* untuk membuat pemicu:

```
aws codecommit put-repository-triggers --cli-input-json
file://trigger.json
```

Perintah ini mengembalikan ID konfigurasi, yang serupa dengan yang berikut:

```
{
  "configurationId": "0123456-I-AM-AN-EXAMPLE"
}
```

4. Untuk menampilkan konfigurasi pemicu, jalankan perintah `get-repository-triggers`, dengan menentukan nama repositori:

```
aws codecommit get-repository-triggers --repository-name MyDemoRepo
```

Perintah ini mengembalikan struktur semua pemicu dikonfigurasi untuk repositori, yang serupa dengan berikut ini:

```
{
  "configurationId": "0123456-I-AM-AN-EXAMPLE",
  "triggers": [
    {
      "events": [
        "all"
      ],
      "destinationArn": "arn:aws:lambda:us-
east-1:111122223333:MyCodeCommitFunction",
      "branches": [
        "main",
        "preprod"
      ],
    }
  ],
}
```

```
        "name": "MyLambdaFunctionTrigger",
        "customData": "Project ID 12345"
    }
]
}
```

5. Untuk menguji fungsi pemicu, buat dan doronglah sebuah commit ke repositori di mana Anda mengkonfigurasi pemicu tersebut. Anda akan melihat respons dari fungsi Lambda pada tab Pemantauan untuk fungsi di konsol Lambda.

Edit pemicu untuk repositori AWS CodeCommit

Anda dapat mengedit pemicu yang telah dibuat untuk CodeCommit repositori. Anda dapat mengubah peristiwa dan cabang untuk pemicu, tindakan yang diambil dalam menanggapi peristiwa, dan pengaturan lainnya.

Topik

- [Edit sebuah pemicu untuk repositori \(konsol\)](#)
- [Edit pemicu untuk repositori \(AWS CLI\)](#)

Edit sebuah pemicu untuk repositori (konsol)

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori tempat Anda ingin mengedit pemicu untuk peristiwa repositori.
3. Di panel navigasi untuk repositori, pilih Pengaturan, lalu pilih Pemicu.
4. Dari daftar pemicu untuk repositori, pilih pemicu yang ingin Anda edit, lalu pilih Edit.
5. Buat perubahan yang Anda inginkan untuk pemicu, dan kemudian pilih Simpan.

Edit pemicu untuk repositori (AWS CLI)

1. Pada terminal (Linux, macOS, atau Unix) atau prompt perintah (Windows), jalankan perintah `get-repository-triggers` untuk membuat file JSON dengan struktur semua pemicu yang dikonfigurasi untuk repositori Anda. Misalnya, untuk membuat file JSON bernama *MyTriggers.json* dengan struktur semua pemicu yang dikonfigurasi untuk repositori bernama: *MyDemoRepo*

```
aws codecommit get-repository-triggers --repository-name MyDemoRepo
>MyTriggers.json
```

Perintah ini tidak mengembalikan apa pun, tetapi file bernama *MyTriggers.json* dibuat di direktori tempat Anda menjalankan perintah.

2. Edit file JSON dalam editor teks biasa dan buat perubahan pada blok pemicu dari pemicu yang ingin Anda edit. Ganti pasangan `configurationId` dengan pasangan `repositoryName`. Simpan file tersebut.

Misalnya, jika Anda ingin mengedit pemicu bernama *MyFirstTrigger* dalam repositori bernama *MyDemoRepo* sehingga berlaku untuk semua cabang, ganti `configurationId` dengan `repositoryName`, dan hapus yang ditentukan `main` dan `preprod` cabang dalam teks *miring merah*. Secara default, jika tidak ada cabang yang ditentukan, pemicu tersebut berlaku untuk semua cabang di repositori:

```
{
  "repositoryName": "MyDemoRepo",
  "triggers": [
    {
      "destinationArn": "arn:aws:sns:us-
east-2:111122223333:MyCodeCommitTopic",
      "branches": [
        "main",
        "preprod"
      ],
      "name": "MyFirstTrigger",
      "customData": "",
      "events": [
        "all"
      ]
    }
  ]
}
```

3. Di terminal atau baris perintah, jalankan perintah `put-repository-triggers`. Ini memperbarui semua pemicu untuk repositori, termasuk perubahan yang Anda buat pada pemicu: *MyFirstTrigger*

```
aws codecommit put-repository-triggers --repository-name MyDemoRepo
file://MyTriggers.json
```

Perintah ini mengembalikan ID konfigurasi, yang serupa dengan yang berikut ini:

```
{  
  "configurationId": "0123456-I-AM-AN-EXAMPLE"  
}
```

Uji pemicu untuk repositori AWS CodeCommit

Anda dapat menguji pemicu yang telah dibuat untuk CodeCommit repositori. Pengujian melibatkan menjalankan pemicu dengan data sampel dari repositori Anda, termasuk ID commit terbaru. Jika tidak ada riwayat commit untuk repositori tersebut, nilai sampel yang terdiri dari angka nol dihasilkan sebagai gantinya. Pemicu pengujian membantu Anda mengonfirmasi bahwa Anda telah mengonfigurasi akses dengan benar antara CodeCommit dan target pemicu, apakah itu AWS Lambda fungsi atau pemberitahuan Amazon Simple Notification Service.

Topik

- [Uji pemicu untuk repositori \(konsol\)](#)
- [Uji pemicu untuk repositori \(AWS CLI\)](#)

Uji pemicu untuk repositori (konsol)

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori tempat Anda ingin menguji pemicu untuk peristiwa repositori.
3. Di panel navigasi untuk repositori, pilih Pengaturan, lalu pilih Pemicu.
4. Pilih pemicu yang ingin Anda uji, lalu pilih Uji pemicu. Anda akan melihat pesan berhasil atau gagal. Jika berhasil, Anda juga akan melihat respon tindakan yang sesuai dari fungsi Lambda atau topik Amazon SNS.

Uji pemicu untuk repositori (AWS CLI)

1. Pada terminal (Linux, macOS, atau Unix) atau prompt perintah (Windows), jalankan perintah `get-repository-triggers` untuk membuat file JSON dengan struktur semua pemicu yang dikonfigurasi untuk repositori Anda. Misalnya, untuk membuat file JSON bernama *TestTrigger.json* dengan struktur semua pemicu yang dikonfigurasi untuk repositori bernama: MyDemoRepo

```
aws codecommit get-repository-triggers --repository-name MyDemoRepo
>TestTrigger.json
```

Perintah ini membuat file bernama *TestTriggers.json* di direktori tempat Anda menjalankan perintah.

2. Edit file JSON dalam editor teks biasa dan membuat perubahan pada pernyataan pemicu. Ganti pasangan `configurationId` dengan pasangan `repositoryName`. Simpan file tersebut.

Misalnya, jika Anda ingin menguji pemicu bernama `MyFirstTrigger` dalam repositori bernama `MyDemoRepo` sehingga berlaku untuk semua cabang, ganti `configurationId` dengan `repositoryName` dan kemudian simpan file yang terlihat mirip dengan berikut ini seperti `TestTrigger`

```
{
  "repositoryName": "MyDemoRepo",
  "triggers": [
    {
      "destinationArn": "arn:aws:sns:us-
east-2:111122223333:MyCodeCommitTopic",
      "branches": [
        "main",
        "preprod"
      ],
      "name": "MyFirstTrigger",
      "customData": "",
      "events": [
        "all"
      ]
    }
  ]
}
```

3. Di terminal atau baris perintah, jalankan perintah `test-repository-triggers`. Ini memperbarui semua pemicu untuk repositori, termasuk perubahan yang Anda buat pada pemicu: *MyFirstTrigger*

```
aws codecommit test-repository-triggers --cli-input-json file://TestTrigger.json
```

Perintah ini mengembalikan respons yang serupa dengan yang berikut:

```
{
  "successfulExecutions": [
    "MyFirstTrigger"
  ],
  "failedExecutions": []
}
```

Hapus pemicu dari repositori AWS CodeCommit

Anda mungkin ingin menghapus pemicu jika tidak lagi digunakan. Anda tidak dapat membatalkan penghapusan pemicu, tetapi Anda dapat membuat pemicu lain lagi.

Note

Jika Anda mengkonfigurasi satu atau lebih pemicu untuk repositori Anda, menghapus repositori tidak menghapus topik Amazon SNS atau fungsi Lambda yang Anda konfigurasi sebagai target dari pemicu tersebut. Pastikan untuk menghapus sumber daya tersebut juga, jika tidak lagi diperlukan.

Topik

- [Hapus pemicu dari repositori \(konsol\)](#)
- [Hapus pemicu dari repositori \(AWS CLI\)](#)

Hapus pemicu dari repositori (konsol)

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori tempat Anda ingin menghapus pemicu untuk peristiwa repositori.
3. Di panel navigasi untuk repositori, pilih Pengaturan. Di Pengaturan, pilih Pemicu.
4. Pilih pemicu yang ingin Anda hapus dari daftar pemicu, lalu pilih Hapus.
5. Di kotak dialog, pilih hapus untuk mengonfirmasi.

Hapus pemicu dari repositori (AWS CLI)

1. Pada terminal (Linux, macOS, atau Unix) atau prompt perintah (Windows), jalankan perintah `get-repository-triggers` untuk membuat file JSON dengan struktur semua pemicu yang dikonfigurasi untuk repositori Anda. Misalnya, untuk membuat file JSON bernama *MyTriggers.json* dengan struktur semua pemicu yang dikonfigurasi untuk repositori bernama: MyDemoRepo

```
aws codecommit get-repository-triggers --repository-name MyDemoRepo
>MyTriggers.json
```

Perintah ini membuat file bernama *MyTriggers.json* di direktori tempat Anda menjalankan perintah.

2. Edit file JSON dalam editor teks biasa dan hapus blok pemicu untuk pemicu yang ingin Anda hapus. Ganti pasangan `configurationId` dengan pasangan `repositoryName`. Simpan file tersebut.

Misalnya, jika Anda ingin menghapus pemicu bernama *MyFirstTrigger* dari repositori bernama *MyDemoRepo*, Anda akan mengganti `configurationId` dengan `repositoryName`, dan menghapus pernyataan dalam teks *miring merah*:

```
{
  "repositoryName": "MyDemoRepo",
  "triggers": [
    {
      "destinationArn": "arn:aws:sns:us-
east-2:111122223333:MyCodeCommitTopic",
      "branches": [
        "main",
        "preprod"
      ],
      "name": "MyFirstTrigger",
      "customData": "",
      "events": [
        "all"
      ]
    },
    {
      "destinationArn": "arn:aws:lambda:us-
east-2:111122223333:function:MyCodeCommitJSFunction",
      "branches": [],
```

```
        "name": "MyLambdaTrigger",
        "events": [
            "all"
        ]
    }
]
```

3. Di terminal atau baris perintah, jalankan perintah `put-repository-triggers`. Ini memperbarui pemicu untuk repositori dan menghapus pemicu: *MyFirstTrigger*

```
aws codecommit put-repository-triggers --repository-name MyDemoRepo
file://MyTriggers.json
```

Perintah ini mengembalikan ID konfigurasi, yang serupa dengan yang berikut ini:

```
{
  "configurationId": "0123456-I-AM-AN-EXAMPLE"
}
```

Note

Untuk menghapus semua pemicu repositori bernama *MyDemoRepo*, file JSON Anda akan terlihat mirip dengan ini:

```
{
  "repositoryName": "MyDemoRepo",
  "triggers": []
}
```

Kaitkan atau pisahkan AWS CodeCommit repositori dengan Amazon Reviewer CodeGuru

Amazon CodeGuru Reviewer adalah layanan peninjauan kode otomatis yang menggunakan analisis program dan pembelajaran mesin untuk mendeteksi masalah umum dan merekomendasikan perbaikan dalam kode Java atau Python Anda. Anda dapat mengaitkan repositori di akun Amazon Web Services Anda dengan CodeGuru Reviewer. Ketika Anda melakukannya, CodeGuru Reviewer

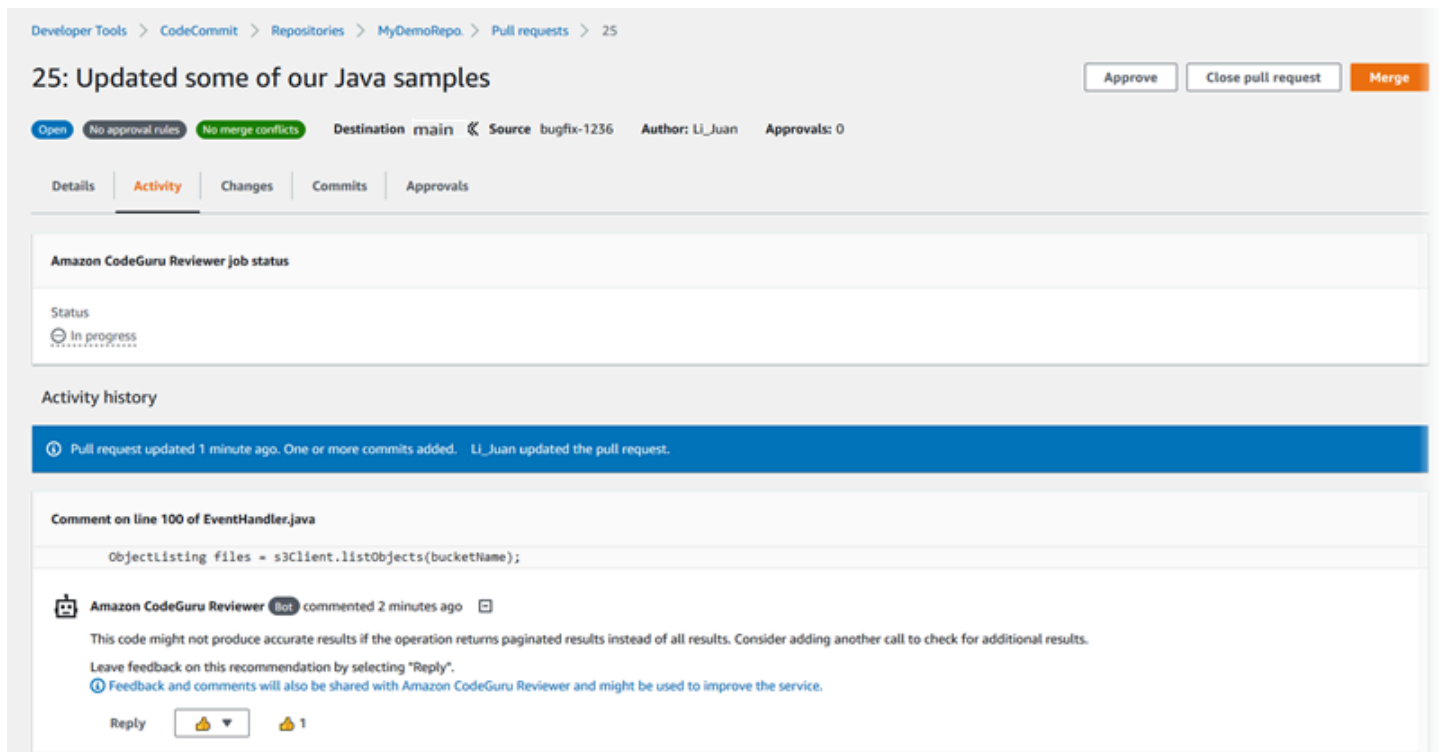
akan membuat peran terkait layanan yang memungkinkan CodeGuru Reviewer menganalisis kode di semua permintaan tarik yang dibuat setelah asosiasi dibuat.

Setelah Anda mengaitkan repositori, CodeGuru Reviewer menganalisis dan mengomentari masalah apa pun yang ditemukan saat Anda membuat permintaan tarik. Setiap komentar ditandai dengan jelas berasal dari CodeGuru Reviewer dengan sebutan Amazon CodeGuru Reviewer. Anda dapat membalas komentar ini seperti yang Anda lakukan untuk komentar lain dalam permintaan tarik, dan Anda juga dapat memberikan umpan balik pada kualitas saran. Umpan balik ini dibagikan dengan CodeGuru Reviewer dan dapat membantu meningkatkan layanan dan sarannya.

Note

Anda tidak akan melihat komentar dari CodeGuru Reviewer dalam permintaan tarik yang dibuat sebelum repositori dikaitkan dengannya. Anda mungkin tidak melihat komentar dalam permintaan tarik yang dibuat setelah pengaitan tersebut karena alasan berikut:

- Permintaan tarik tidak mengandung Java atau kode Python.
- CodeGuru Peninjau belum memiliki cukup waktu untuk menjalankan dan meninjau kode dalam permintaan tarik. Proses ini dapat memakan waktu hingga 30 menit. Komentar dapat muncul saat tinjauan berlangsung, tetapi komentar tidak lengkap sampai status lowongan ditampilkan sebagai Selesai.
- CodeGuru Peninjau tidak menemukan masalah apa pun dalam kode Java atau Python dalam permintaan tarik.
- Tugas tinjauan kode gagal untuk berjalan. Untuk meninjau status tinjauan untuk permintaan tarik, lihat tab Aktivitas dari permintaan tarik.
- Anda melihat perubahan pada permintaan tarik di tab Perubahan, permintaan tarik telah diperbarui, dan Amazon CodeGuru Reviewer tidak menemukan masalah apa pun dalam perubahan tersebut. Komentar Amazon CodeGuru Reviewer hanya muncul di tab Perubahan jika komentar dibuat pada revisi terbaru dari permintaan tarik. Mereka selalu muncul di tab Aktivitas.



Developer Tools > CodeCommit > Repositories > MyDemoRepo > Pull requests > 25

25: Updated some of our Java samples

Approve Close pull request Merge

Open No approval rules No merge conflicts Destination main Source bugfix-1236 Author: Li_Juan Approvals: 0

Details Activity Changes Commits Approvals

Amazon CodeGuru Reviewer job status

Status
In progress

Activity history

Pull request updated 1 minute ago. One or more commits added. Li_Juan updated the pull request.

Comment on line 100 of EventHandler.java

```
ObjectListing files = s3Client.listObjects(bucketName);
```

Amazon CodeGuru Reviewer commented 2 minutes ago

This code might not produce accurate results if the operation returns paginated results instead of all results. Consider adding another call to check for additional results.

Leave feedback on this recommendation by selecting "Reply".

Feedback and comments will also be shared with Amazon CodeGuru Reviewer and might be used to improve the service.

Reply 1

Untuk informasi selengkapnya [Bekerja dengan permintaan pull di repositori AWS CodeCommit](#), lihat [Meninjau permintaan pull](#), dan [Panduan Pengguna Amazon CodeGuru Reviewer](#).

Note

Anda harus masuk dengan pengguna IAM atau peran yang memiliki izin yang cukup untuk mengaitkan atau memisahkan repositori dengan Reviewer. CodeGuru Untuk informasi tentang kebijakan terkelola CodeCommit yang menyertakan izin ini, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#) dan [AWS CodeCommit kebijakan terkelola dan Amazon CodeGuru Reviewer](#). Untuk informasi tentang izin dan keamanan CodeGuru Peninjau, lihat [Panduan Pengguna Amazon CodeGuru Reviewer](#).

Topik

- [Kaitkan repositori dengan Reviewer CodeGuru](#)
- [Memutuskan repositori dari Reviewer CodeGuru](#)

Kaitkan repositori dengan Reviewer CodeGuru

Gunakan AWS CodeCommit konsol untuk mengaitkan repositori dengan Reviewer dengan CodeGuru cepat. Untuk metode lain, lihat Panduan Pengguna Amazon CodeGuru Reviewer.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori untuk dikaitkan dengan Reviewer. CodeGuru
3. Pilih Pengaturan, lalu pilih Amazon CodeGuru Reviewer.
4. Pilih Kaitkan repositori.

Note

Diperlukan waktu hingga 10 menit untuk sepenuhnya mengaitkan repositori dengan CodeGuru Reviewer. Status tidak akan diperbarui secara otomatis. Untuk menampilkan status saat ini, pilih tombol refresh.

The screenshot shows the AWS CodeCommit console interface. At the top, there is a breadcrumb navigation: Developer Tools > CodeCommit > Repositories > MyDemoRepo > Settings. Below this, the title 'MyDemoRepo' is displayed. There are five tabs: General, Notifications, Triggers, Repository tags, and Amazon CodeGuru Reviewer (which is selected and highlighted in orange). The main content area shows the 'Amazon CodeGuru Reviewer for Java and Python' section. It includes an 'Info' icon and a refresh button. The text reads: 'When you associate this repository with Amazon CodeGuru Reviewer, you will receive recommendations to help improve Java and Python code in all pull requests.' Below this, the 'Status' is shown as 'Associated' with a green checkmark icon. At the bottom of this section, there is a 'Disassociate repository' button.

Memutuskan repositori dari Reviewer CodeGuru

Gunakan AWS CodeCommit konsol untuk memisahkan repositori dari Reviewer dengan cepat. CodeGuru Untuk metode lain, lihat Panduan Pengguna Amazon CodeGuru Reviewer.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori yang ingin Anda pisahkan dari Reviewer. CodeGuru
3. Pilih Pengaturan, lalu pilih Amazon CodeGuru Reviewer.
4. Pilih pisahkan repositori.

Lihat detail CodeCommit repositori

Anda dapat menggunakan AWS CodeCommit konsol, AWS CLI, atau Git dari repo lokal yang terhubung ke repositori untuk melihat informasi tentang CodeCommit repositori yang tersedia.

Sebelum Anda mengikuti petunjuk ini, selesaikan langkah-langkah di [Pengaturan](#) .

Topik

- [Tampilkan detail repositori \(konsol\)](#)
- [Lihat detail CodeCommit repositori \(Git\)](#)
- [Lihat detail CodeCommit repositori \(AWS CLI\)](#)

Tampilkan detail repositori (konsol)

Gunakan AWS CodeCommit konsol untuk dengan cepat melihat semua repositori yang dibuat dengan akun Amazon Web Services Anda.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, lihat detail tentang repositori di Wilayah AWS tempat Anda masuk. Gunakan pemilih Wilayah untuk memilih Wilayah AWS t yang berbeda untuk melihat repositori di Wilayah itu.
3. Pilih nama repoistori yang ingin Anda lihat lebih detail, lalu lakukan salah satu hal berikut:
 - Untuk menampilkan URL untuk mengkloning repositori, pilih URL Kloning, lalu pilih protokol yang ingin Anda gunakan saat mengkloning repositori tersebut. Ini menyalin URL klon tersebut. Untuk meninjaunya, tempelkan ke editor teks biasa.
 - Untuk melihat opsi yang dapat dikonfigurasi untuk repositori serta detail seperti ARN repositori dan ID repositori, di panel navigasi, pilih Pengaturan.

Note

Jika Anda masuk sebagai pengguna IAM, Anda dapat mengonfigurasi dan menyimpan preferensi Anda untuk menampilkan kode dan pengaturan konsol lainnya. Untuk informasi selengkapnya, lihat [Bekerja dengan preferensi pengguna](#).

Lihat detail CodeCommit repositori (Git)

Untuk menggunakan Git dari repo lokal untuk melihat detail tentang CodeCommit repositori, jalankan perintah. `git remote show`

Sebelum Anda melakukan langkah-langkah ini, sambungkan repo lokal ke CodeCommit repositori. Untuk petunjuk, lihat [Connect ke sebuah repositori](#).

1. Jalankan `git remote show remote-name` perintah, di mana *remote-name* adalah alias CodeCommit repositori (secara default). `origin`

Tip

Untuk mendapatkan daftar nama CodeCommit repositori dan URL mereka, jalankan perintah. `git remote -v`

Misalnya, untuk melihat detail tentang CodeCommit repositori dengan alias: `origin`

```
git remote show origin
```

2. Untuk HTTPS:

```
* remote origin
Fetch URL: https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
Push URL: https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
HEAD branch: (unknown)
Remote branches:
  MyNewBranch tracked
  main tracked
Local ref configured for 'git pull':
  MyNewBranch merges with remote MyNewBranch (up to date)
Local refs configured for 'git push':
```

```
MyNewBranch pushes to MyNewBranch (up to date)
main pushes to main (up to date)
```

Untuk SSH:

```
* remote origin
Fetch URL: ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
Push URL: ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
HEAD branch: (unknown)
Remote branches:
  MyNewBranch tracked
  main tracked
Local ref configured for 'git pull':
  MyNewBranch merges with remote MyNewBranch (up to date)
Local refs configured for 'git push':
  MyNewBranch pushes to MyNewBranch (up to date)
  main pushes to main (up to date)
```

Tip

Untuk mencari ID kunci SSH untuk pengguna IAM Anda, buka konsol IAM dan perluas Kredensial Keamanan pada halaman detail pengguna IAM. ID kunci SSH dapat ditemukan di SSH Keys untuk AWS CodeCommit

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Lihat detail CodeCommit repositori (AWS CLI)

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal file AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk menggunakan AWS CLI untuk melihat rincian repositori, jalankan perintah berikut:

- [Untuk melihat daftar nama CodeCommit repositori dan ID yang sesuai, jalankan list-repositori.](#)
- [Untuk melihat informasi tentang satu CodeCommit repositori, jalankan get-repository.](#)
- Untuk melihat informasi tentang beberapa repositori di CodeCommit, jalankan [batch-get-repositories](#)

Untuk melihat daftar CodeCommit repositori

1. Jalankan perintah `list-repositories`:

```
aws codecommit list-repositories
```

Anda dapat menggunakan opsi `--sort-by` atau `--order` opsional untuk mengubah urutan informasi yang dikembalikan.

2. Jika berhasil, perintah ini mengeluarkan `repositories` objek yang berisi nama dan ID semua repositori yang CodeCommit terkait dengan akun Amazon Web Services.

Berikut adalah beberapa contoh output berdasarkan perintah sebelumnya:

```
{
  "repositories": [
    {
      "repositoryName": "MyDemoRepo",
      "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE"
    },
    {
      "repositoryName": "MyOtherDemoRepo",
      "repositoryId": "cfc29ac4-b0cb-44dc-9990-f6f51EXAMPLE"
    }
  ]
}
```

Untuk melihat detail tentang satu CodeCommit repositori

1. Jalankan `get-repository` perintah, tentukan nama CodeCommit repositori dengan opsi. `--repository-name`

Tip

Untuk mendapatkan nama repositori, jalankan perintah CodeCommit [list-repositori](#).

Misalnya, untuk melihat detail tentang CodeCommit repositori bernama: `MyDemoRepo`

```
aws codecommit get-repository --repository-name MyDemoRepo
```

2. Jika berhasil, perintah ini menghasilkan objek `repositoryMetadata` dengan informasi berikut:

- Nama repositori (`repositoryName`).
- Deskripsi repositori (`repositoryDescription`).
- ID repositori yang unik dan dihasilkan oleh sistem (`repositoryId`).
- ID akun Amazon Web Services yang terkait dengan repositori (`accountId`).

Berikut adalah beberapa contoh output berdasarkan perintah sebelumnya:

```
{
  "repositoryMetadata": {
    "creationDate": 1429203623.625,
    "defaultBranch": "main",
    "repositoryName": "MyDemoRepo",
    "cloneUrlSsh": "ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo",
    "lastModifiedDate": 1430783812.0869999,
    "repositoryDescription": "My demonstration repository",
    "cloneUrlHttp": "https://codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo",
    "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
    "Arn": "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",
    "accountId": "111111111111"
  }
}
```

Untuk melihat detail tentang beberapa CodeCommit repositori

1. Jalankan perintah `batch-get-repositories` dengan opsi `--repository-names`. Tambahkan spasi di antara setiap nama CodeCommit repositori.

Tip

Untuk mendapatkan nama repositori CodeCommit, jalankan perintah [list-repositori](#).

Misalnya, untuk melihat detail tentang dua CodeCommit repositori bernama `MyDemoRepo` dan `MyOtherDemoRepo`

```
aws codecommit batch-get-repositories --repository-names MyDemoRepo MyOtherDemoRepo
```

2. Jika berhasil, perintah ini menghasilkan objek dengan informasi berikut:

- Daftar CodeCommit repositori apa pun yang tidak dapat ditemukan (`repositoriesNotFound`).
- Daftar CodeCommit repositori (`repositories`). Setiap nama CodeCommit repositori diikuti oleh:
 - Deskripsi repositori (`repositoryDescription`).
 - ID repositori yang unik dan dihasilkan oleh sistem (`repositoryId`).
 - ID akun Amazon Web Services yang terkait dengan repositori (`accountId`).

Berikut adalah beberapa contoh output berdasarkan perintah sebelumnya:

```
{
  "repositoriesNotFound": [],
  "repositories": [
    {
      "creationDate": 1429203623.625,
      "defaultBranch": "main",
      "repositoryName": "MyDemoRepo",
      "cloneUrlSsh": "ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo",
      "lastModifiedDate": 1430783812.0869999,
      "repositoryDescription": "My demonstration repository",
      "cloneUrlHttp": "https://codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo",
      "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE",
      "Arn": "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",
      "accountId": "111111111111"
    },
    {
      "creationDate": 1429203623.627,
      "defaultBranch": "main",
      "repositoryName": "MyOtherDemoRepo",
```

```
        "cloneUrlSsh": "ssh://git-codecommit.us-east-2.amazonaws.com/v1/
repos/MyOtherDemoRepo",
        "lastModifiedDate": 1430783812.0889999,
        "repositoryDescription": "My other demonstration repository",
        "cloneUrlHttp": "https://codecommit.us-east-2.amazonaws.com/v1/
repos/MyOtherDemoRepo",
        "repositoryId": "cfc29ac4-b0cb-44dc-9990-f6f51EXAMPLE",
        "Arn": "arn:aws:codecommit:us-east-2:111111111111:MyOtherDemoRepo",
        "accountId": "111111111111"
    }
],
"repositoriesNotFound": []
}
```

Ubah AWS CodeCommit pengaturan repositori

Anda dapat menggunakan AWS CLI dan AWS CodeCommit konsol untuk mengubah pengaturan CodeCommit repositori, seperti deskripsi atau namanya.

Important

Mengubah nama repositori dapat merusak repository's lokal yang menggunakan nama lama di URL jarak jauh mereka. Jalankan perintah `git remote set-url` untuk memperbarui URL jarak jauh untuk menggunakan nama repositori baru.

Topik

- [Ubah pengaturan repositori \(konsol\)](#)
- [Ubah pengaturan AWS CodeCommit repositori \(AWS CLI\)](#)

Ubah pengaturan repositori (konsol)

Untuk menggunakan AWS CodeCommit konsol untuk mengubah pengaturan CodeCommit repositori AWS CodeCommit, ikuti langkah-langkah ini.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin mengubah pengaturan.
3. Pada panel navigasi, silakan pilih Pengaturan.

4. Untuk mengganti nama repositori, di Nama repositori, masukkan nama baru di kotak teks Nama dan pilih Simpan. Saat diminta, verifikasi pilihan Anda.

Important

Mengubah nama AWS CodeCommit repositori akan mengubah URL SSH dan HTTPS yang harus dihubungkan pengguna ke repositori. Pengguna tidak akan dapat terhubung ke repositori ini sampai mereka memperbarui pengaturan koneksi mereka. Selain itu, karena ARN repositori akan berubah, yang mengubah nama repositori akan membatalkan setiap kebijakan pengguna IAM yang bergantung pada ARN repositori ini. Untuk menyambung ke repositori setelah nama diubah, setiap pengguna harus menggunakan perintah git remote set-url dan menentukan URL baru untuk digunakan. Misalnya, jika Anda mengubah nama repositori dari menjadi MyRenamedDemoRepo, pengguna yang menggunakan HTTPS MyDemoRepo untuk terhubung ke repositori akan menjalankan perintah Git berikut:

```
git remote set-url origin https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyRenamedDemoRepo
```

Para pengguna yang menggunakan SSH untuk terhubung ke repositori akan menjalankan perintah Git berikut:

```
git remote set-url origin ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyRenamedDemoRepo
```

Untuk opsi lainnya, lihat dokumentasi Git Anda.

5. Untuk mengubah deskripsi repositori, modifikasi teks dalam kotak teks Deskripsi, dan kemudian pilih Simpan.

Note

Bidang deskripsi menampilkan Markdown di konsol dan menerima semua karakter HTML dan karakter Unicode yang valid. Jika Anda adalah pengembang aplikasi yang menggunakan BatchGetRepositories API GetRepository atau dan Anda berencana untuk menampilkan bidang deskripsi repositori di browser web, lihat Referensi [CodeCommit API](#).

6. Untuk mengubah cabang default, di Cabang default, pilih daftar drop-down cabang dan pilih cabang yang berbeda. Pilih Simpan.
7. Untuk mengubah kunci AWS KMS enkripsi yang digunakan untuk mengenkripsi dan mendekripsi data dalam repositori, di kunci enkripsi Repositori, pilih salah satu Kunci yang dikelola AWS atau kunci yang dikelola Pelanggan untuk menentukan jenis kunci yang akan digunakan. Jika memilih kunci yang dikelola pelanggan, masukkan ARN kunci. Pilih Simpan.
8. Untuk menghapus repositori, pilih Hapus repositori. Dalam kotak di sebelah Ketik nama repositori untuk mengonfirmasi penghapusan, masukkan **delete**, lalu pilih Hapus.

Important

Setelah Anda menghapus repositori ini AWS CodeCommit, Anda tidak akan lagi dapat mengkloningnya ke repo lokal atau repo bersama. Anda juga akan tidak lagi dapat menarik data darinya, atau mendorong data kepadanya, dari repo lokal atau repo bersama. Tindakan ini tidak dapat dibatalkan.

Ubah pengaturan AWS CodeCommit repositori ()AWS CLI

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal file AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk digunakan AWS CLI untuk mengubah pengaturan CodeCommit repositori AWS CodeCommit, jalankan satu atau beberapa perintah berikut:

- [update-repository-description](#) untuk mengubah deskripsi CodeCommit repositori.
- [update-repository-name](#) untuk mengubah nama CodeCommit repositori.

Untuk mengubah deskripsi CodeCommit repositori

1. Jalankan perintah `update-repository-description`, dengan menentukan:
 - Nama CodeCommit repositori (dengan `--repository-name` opsi).

Tip

Untuk mendapatkan nama CodeCommit repositori, jalankan perintah. [list-repositories](#)

- Deskripsi repositori baru (dengan opsi `--repository-description`).

Note

Bidang deskripsi menampilkan Markdown di konsol dan menerima semua karakter HTML dan karakter Unicode yang valid. Jika Anda adalah pengembang aplikasi yang menggunakan BatchGetRepositories API GetRepository atau dan Anda berencana untuk menampilkan bidang deskripsi repositori di browser web, lihat Referensi [CodeCommit API](#).

Misalnya, untuk mengubah deskripsi CodeCommit repositori bernama MyDemoRepo: This description was changed

```
aws codecommit update-repository-description --repository-name MyDemoRepo --repository-description "This description was changed"
```

Perintah ini menghasilkan output hanya jika ada kesalahan.

2. Untuk memverifikasi deskripsi yang diubah, jalankan get-repository perintah, tentukan nama CodeCommit repositori yang deskripsinya Anda ubah dengan opsi. `--repository-name`

Output dari perintah menunjukkan teks yang diubah dalam repositoryDescription.

Untuk mengubah nama CodeCommit repositori

1. Jalankan perintah update-repository-name, dengan menentukan:

- Nama CodeCommit repositori saat ini (dengan `--old-name` opsi).

Tip

Untuk mendapatkan nama CodeCommit repositori, jalankan perintah [list-repositori](#).

- Nama baru CodeCommit repositori (dengan `--new-name` opsi).

Misalnya, untuk mengubah repositori yang bernama MyDemoRepo menjadi MyRenamedDemoRepo:

```
aws codecommit update-repository-name --old-name MyDemoRepo --new-name
MyRenamedDemoRepo
```

Perintah ini menghasilkan output hanya jika ada kesalahan.

Important

Mengubah nama AWS CodeCommit repositori mengubah URL SSH dan HTTPS yang perlu dihubungkan pengguna ke repositori. Pengguna tidak dapat terhubung ke repositori ini sampai mereka memperbarui pengaturan koneksi mereka. Selain itu, karena perubahan ARN repositori, mengubah nama repositori membatalkan setiap kebijakan pengguna IAM yang bergantung pada ARN repositori ini.

2. Untuk memverifikasi nama yang berubah, jalankan perintah `list-repositories` dan tinjau daftar nama repositori.

Sinkronkan perubahan antara sebuah repo lokal dengan sebuah repositori AWS CodeCommit

Anda menggunakan Git untuk menyinkronkan perubahan antara repo lokal dan CodeCommit repositori yang terhubung ke repo lokal.

Untuk mendorong perubahan dari repo lokal ke CodeCommit repositori, jalankan. `git push remote-name branch-name`

Untuk menarik perubahan pada repo lokal dari CodeCommit repositori, jalankan. `git pull remote-name branch-name`

Untuk mendorong dan menarik, nama *jarak jauh adalah nama* panggilan yang digunakan repo lokal untuk repositori. CodeCommit *nama-cabang* adalah nama cabang pada CodeCommit repositori untuk mendorong atau menarik dari.

Tip

Untuk mendapatkan nama panggilan yang digunakan repo lokal untuk CodeCommit repositori, jalankan. `git remote` Untuk mendapatkan daftar nama cabang, jalankan `git branch`.

Tanda bintang (*) muncul di sebelah nama cabang saat ini. (Anda juga dapat menjalankan git status untuk menampilkan nama cabang saat ini.)

Note

Jika Anda mengkloning repositori, dari perspektif repo lokal, nama *jarak jauh bukanlah nama repositori*. CodeCommit Saat Anda mengkloning repositori, *Nama jarak jauh* diatur secara otomatis ke `origin`.

Misalnya, untuk mendorong perubahan dari repo lokal ke `main` cabang di CodeCommit repositori dengan nama panggilan: `origin`

```
git push origin main
```

Demikian pula, untuk menarik perubahan pada repo lokal dari `main` cabang di CodeCommit repositori dengan nama panggilan: `origin`

```
git pull origin main
```

Tip

Jika Anda menambahkan opsi `-u` untuk `git push`, Anda mengatur informasi pelacakan hulu. Sebagai contoh, jika Anda menjalankan `git push -u origin main`, di masa depan Anda dapat menjalankan `git push` dan `git pull` tanpa *Nama jarak jauh Nama cabang*. Untuk mendapatkan informasi pelacakan hulu, jalankan `git remote show remote-name` (sebagai contoh, `git remote show origin`).

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Dorong commit ke repositori Git tambahan

Anda dapat mengkonfigurasi repo lokal Anda untuk mendorong perubahan ke dua repositori jarak jauh. Sebagai contoh, Anda mungkin ingin terus menggunakan solusi repositori Git yang sudah ada

saat Anda mencoba AWS CodeCommit. Ikuti langkah-langkah dasar ini untuk mendorong perubahan dalam repo lokal Anda ke CodeCommit dan repositori Git terpisah.

Tip

Jika Anda tidak memiliki repositori Git, Anda dapat membuat repositori kosong di layanan selain CodeCommit dan kemudian memigrasikan repositori Anda ke CodeCommit repositori tersebut. Anda harus mengikuti langkah-langkah yang serupa dengan yang ada di [Migrasi ke CodeCommit](#).

1. Dari prompt perintah atau terminal, beralihlah ke direktori repo lokal Anda dan jalankan perintah `git remote -v`. Anda akan melihat output yang serupa dengan yang berikut:

Untuk HTTPS:

```
origin https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (fetch)
origin https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (push)
```

Untuk SSH:

```
origin ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (fetch)
origin ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (push)
```

2. Jalankan `git remote set-url --add --push origin git-repository-name` perintah di `git-repository-name` mana URL dan nama repositori Git tempat Anda ingin meng-host kode Anda. Ini mengubah tujuan dorong `origin` ke repositori Git itu.

Note

`git remote set-url --add --push` menimpa URL default untuk dorongan, sehingga Anda harus menjalankan perintah ini dua kali, seperti yang ditunjukkan dalam langkah-langkah selanjutnya.

Misalnya, perintah berikut mengubah push of `origin` ke `some-URL/`: `MyDestinationRepo`


```
git remote set-url --add --push origin some-URL/MyDestinationRepo
```

Perintah ini tidak mengembalikan apa pun.

 Tip

Jika Anda mendorong ke repositori Git yang mensyaratkan kredensial, pastikan Anda mengkonfigurasi kredensial di kredensial-pembantu atau di konfigurasi string *some-URL*. Jika tidak, dorongan Anda ke repositori itu gagal.

3. Jalankan perintah `git remote -v` ini lagi, yang harus membuat output yang serupa dengan yang berikut:

Untuk HTTPS:

```
origin https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (fetch)
origin some-URL/MyDestinationRepo (push)
```

Untuk SSH:

```
origin ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (fetch)
origin some-URL/MyDestinationRepo (push)
```

4. Sekarang tambahkan CodeCommit repositori. Jalankan `git remote set-url --add --push origin` lagi, kali ini dengan URL dan nama repositori repositori Anda CodeCommit.

Misalnya, perintah berikut menambahkan push of origin ke `https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo`:

Untuk HTTPS:

```
git remote set-url --add --push origin https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
```

Untuk SSH:

```
git remote set-url --add --push origin ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
```

Perintah ini tidak mengembalikan apa pun.

5. Jalankan perintah `git remote -v` ini lagi, yang harus membuat output yang serupa dengan yang berikut:

Untuk HTTPS:

```
origin https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (fetch)
origin some-URL/MyDestinationRepo (push)
origin https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (push)
```

Untuk SSH:

```
origin ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (fetch)
origin some-URL/MyDestinationRepo (push)
origin ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo (push)
```

*Anda sekarang memiliki dua repositori Git sebagai tujuan untuk push Anda, tetapi push Anda pergi ke *some-URL*/terlebih dahulu.*

MyDestinationRepo Jika dorongan ke repositori itu gagal, commit Anda tidak didorong ke repositori juga.

Tip

Jika repositori lain memerlukan kredensial yang ingin Anda masukkan secara manual, pertimbangkan untuk mengubah urutan dorongan sehingga Anda mendorong terlebih dahulu. CodeCommit jalankan `git remote set-url --delete` untuk menghapus repositori yang didorong pertama, dan kemudian jalankan `git remote set-url --add` untuk menambahkannya lagi sehingga menjadi tujuan dorongan kedua dalam daftar. Untuk opsi lainnya, lihat dokumentasi Git Anda.

6. Untuk memverifikasi Anda sedang mendorong untuk kedua repositori jarak jauh, gunakan editor teks untuk membuat file teks berikut di repo lokal Anda:

```
bees.txt
```

```
-----
```

```
Bees are flying insects closely related to wasps and ants, and are known for their  
role in pollination and for producing honey and beeswax.
```

7. Jalankan git add untuk mengadakan perubahan di repo lokal Anda:

```
git add bees.txt
```

8. Jalankan git commit untuk melakukan perubahan di repo lokal Anda:

```
git commit -m "Added bees.txt"
```

9. Untuk mendorong commit dari repo lokal ke repositori jarak jauh Anda, jalankan git push -u **remote-name branch-name** di mana *nama jarak jauh* adalah julukan yang digunakan oleh repo lokal untuk repositori jarak jauh tersebut dan *nama cabang* adalah nama cabang untuk mendorong ke repositori.

 Tip

Anda hanya harus menggunakan opsi -u pertama kali Anda mendorong. Kemudian informasi pelacakan hulu diatur.

Sebagai contoh, menjalankan git push -u origin main akan menunjukkan dorongan pergi ke kedua repositori jarak jauh di cabang-cabang yang diharapkan, dengan output yang serupa dengan berikut:

Untuk HTTPS:

```
Counting objects: 5, done.  
Delta compression using up to 4 threads.  
Compressing objects: 100% (3/3), done.  
Writing objects: 100% (3/3), 5.61 KiB | 0 bytes/s, done.  
Total 3 (delta 1), reused 0 (delta 0)  
To some-URL/MyDestinationRepo  
  a5ba4ed..250f6c3  main -> main  
Counting objects: 5, done.  
Delta compression using up to 4 threads.  
Compressing objects: 100% (3/3), done.
```

```
Writing objects: 100% (3/3), 5.61 KiB | 0 bytes/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote:
To https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
a5ba4ed..250f6c3 main -> main
```

Untuk SSH:

```
Counting objects: 5, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 5.61 KiB | 0 bytes/s, done.
Total 3 (delta 1), reused 0 (delta 0)
To some-URL/MyDestinationRepo
a5ba4ed..250f6c3 main -> main
Counting objects: 5, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 5.61 KiB | 0 bytes/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote:
To ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
a5ba4ed..250f6c3 main -> main
```

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Konfigurasi akses lintas akun ke AWS CodeCommit repositori menggunakan peran

Anda dapat mengonfigurasi akses ke CodeCommit repositori untuk pengguna dan grup IAM di akun lain. AWS Ini sering disebut sebagai akses lintas akun. Bagian ini memberikan contoh dan step-by-step instruksi untuk mengonfigurasi akses lintas akun untuk repositori bernama *MySharedDemoRepo* di Wilayah AS Timur (Ohio) di AWS akun (disebut sebagai *accountA*) kepada pengguna IAM yang termasuk dalam grup IAM yang disebutkan di akun *DevelopersWithCrossAccountRepositoryAccess* lain (disebut sebagai *accountB*). AWS

Bab ini dibagi menjadi tiga bagian:

- Tindakan untuk Administrator di *AccountA*.

- Tindakan untuk Administrator di AccountB.
- Tindakan untuk pengguna repositori di AccountB.

Untuk mengonfigurasi akses lintas akun:

- Administrator di AccountA masuk sebagai pengguna IAM dengan izin yang diperlukan untuk membuat dan mengelola repositori dan membuat peran di CodeCommit IAM. Jika Anda menggunakan kebijakan terkelola, terapkan IAM FullAccess dan AWSCodeCommitFullAccess ke pengguna IAM ini.

Contoh ID akun untuk AccountA adalah **111122223333**.

- Administrator di AccountB masuk sebagai pengguna IAM dengan izin yang diperlukan untuk membuat dan mengelola pengguna dan grup IAM, dan untuk mengkonfigurasi kebijakan untuk pengguna dan grup. Jika Anda menggunakan kebijakan terkelola, terapkan IAM FullAccess ke pengguna IAM ini.

Contoh ID akun untuk AccountB adalah **888888888888**.

- Pengguna repositori di accountB, untuk meniru aktivitas pengembang, masuk sebagai pengguna IAM yang merupakan anggota grup IAM yang dibuat untuk memungkinkan akses ke repositori di accountA. CodeCommit Akun ini harus dikonfigurasi dengan:
 - AWS Akses Konsol Manajemen.
 - Kunci akses dan kunci rahasia untuk digunakan saat menghubungkan ke AWS sumber daya dan ARN dari peran yang akan diasumsikan saat mengakses repositori di AccountA.
 - Utilitas git-remote-codecommit pada komputer lokal di mana repositori tersebut dikloning. Utilitas ini mensyaratkan Python dan installer-nya, pip. Anda dapat mengunduh utilitas dari [git-remote-codecommit](#) di situs web Indeks Paket Python.

Untuk informasi lebih lanjut, lihat [Langkah-langkah penyiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit](#) dan [pengguna IAM](#).

Topik

- [Akses repositori lintas-akun: Tindakan untuk administrator di AccountA](#)
- [Akses repositori lintas akun: Tindakan untuk administrator di AccountB](#)
- [Akses repositori lintas akun: Tindakan untuk pengguna repositori di AccountB](#)

Akses repositori lintas-akun: Tindakan untuk administrator di AccountA

Untuk mengizinkan pengguna atau grup di AccountB mengakses repositori di AccountA, administrator AccountA harus:

- Membuat kebijakan di AccountA yang memberikan akses ke repositori.
- Membuat peran di AccountA yang dapat diambil oleh pengguna IAM dan grup di AccountB.
- Melampirkan kebijakan pada peran tersebut.

Bab berikut memberikan langkah dan contoh.

Topik

- [Langkah 1: Buatlah kebijakan untuk akses repositori di AccountA](#)
- [Langkah 2: Buat peran untuk akses repositori di AccountA](#)

Langkah 1: Buatlah kebijakan untuk akses repositori di AccountA

Anda dapat membuat kebijakan di AccountA yang memberikan akses ke repositori di AccountB. Dengan bergantung pada tingkat akses yang ingin Anda izinkan, lakukan salah satu hal berikut:

- Konfigurasi kebijakan untuk mengizinkan pengguna AccountB mengakses repositori tertentu, namun jangan izinkan mereka Untuk menampilkan daftar semua repositori di AccountA.
- Konfigurasi akses tambahan untuk mengizinkan pengguna AccountB memilih repositori dari daftar semua repositori di AccountA.

Untuk membuat kebijakan untuk akses repositori

1. Masuk ke AWS Management Console sebagai pengguna IAM dengan izin untuk membuat kebijakan di AccountA.
2. Buka konsol IAM di <https://console.aws.amazon.com/iam/>.
3. Di panel navigasi, pilih Kebijakan.
4. Pilih Buat kebijakan.
5. Pilih tab JSON dan tempelkan dokumen kebijakan JSON berikut ke dalam kotak teks JSON. Ganti *us-east-2* dengan untuk repositori, *111122223333* dengan Wilayah AWS ID akun

untuk AccountA, dan dengan nama untuk repositori Anda di AccountA: *MySharedDemoRepo*
CodeCommit

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "codecommit:BatchGet*",
        "codecommit:Create*",
        "codecommit>DeleteBranch",
        "codecommit:Get*",
        "codecommit:List*",
        "codecommit:Describe*",
        "codecommit:Put*",
        "codecommit:Post*",
        "codecommit:Merge*",
        "codecommit:Test*",
        "codecommit:Update*",
        "codecommit:GitPull",
        "codecommit:GitPush"
      ],
      "Resource": [
        "arn:aws:codecommit:us-east-2:111122223333:MySharedDemoRepo"
      ]
    }
  ]
}
```

Jika Anda ingin pengguna yang menjalankan peran ini dapat melihat daftar repositori di beranda CodeCommit konsol, tambahkan pernyataan tambahan ke kebijakan, sebagai berikut:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "codecommit:BatchGet*",
        "codecommit:Create*",
        "codecommit>DeleteBranch",
```

```

        "codecommit:Get*",
        "codecommit:List*",
        "codecommit:Describe*",
        "codecommit:Put*",
        "codecommit:Post*",
        "codecommit:Merge*",
        "codecommit:Test*",
        "codecommit:Update*",
        "codecommit:GitPull",
        "codecommit:GitPush"
    ],
    "Resource": [
        "arn:aws:codecommit:us-east-2:111122223333:MySharedDemoRepo"
    ]
},
{
    "Effect": "Allow",
    "Action": "codecommit:ListRepositories",
    "Resource": "*"
}
]
}

```

Akses ini memudahkan pengguna yang mengambil peran ini dengan kebijakan ini untuk menemukan repositori tempat mereka memiliki akses. Mereka dapat memilih nama repositori dari daftar dan diarahkan ke beranda repositori bersama (Code). Pengguna tidak dapat mengakses salah satu repositori lain yang mereka lihat dalam daftar, tetapi mereka dapat menampilkan repositori di AccountA pada halaman Dasbor.

Jika Anda tidak ingin mengizinkan pengguna yang mengambil peran untuk dapat melihat daftar semua repositori di AccountA, gunakan contoh kebijakan pertama, tetapi pastikan Anda mengirim pengguna tersebut tautan langsung ke halaman beranda repositori bersama di konsol CodeCommit

6. Pilih Tinjau kebijakan. Validator kebijakan melaporkan kesalahan sintaks (misalnya, jika Anda lupa mengganti ID akun Amazon Web Services dan nama repositori contoh dengan ID akun Amazon Web Services dan nama repositori Anda).
7. Pada halaman Kebijakan ulasan, masukkan nama untuk kebijakan (misalnya, *CrossAccountAccessForMySharedDemoRepo*). Anda juga dapat memberikan deskripsi opsional untuk kebijakan ini. Pilih Buat kebijakan.

Langkah 2: Buat peran untuk akses repositori di AccountA

Setelah Anda mengkonfigurasi kebijakan, buat peran yang dapat diambil oleh pengguna dan grup IAM di AccountB, dan lampirkan kebijakan ke peran tersebut.

Untuk membuat peran untuk akses repositori

1. Di konsol IAM, pilih Peran.
2. Pilih Buat peran.
3. Pilih akun Amazon Web Services lainnya.
4. Di ID Akun, masukkan ID akun Amazon Web Services untuk AccountB (misalnya, **888888888888**). Pilih Berikutnya: Izin.
5. Di Lampirkan kebijakan izin, pilih kebijakan yang Anda buat di prosedur sebelumnya (***CrossAccountAccessForMySharedDemoRepo***). Pilih Berikutnya: Tinjau.
6. Di Nama peran, masukkan nama untuk peran (misalnya, ***MyCrossAccountRepositoryContributorRole***). Anda juga dapat memasukkan deskripsi opsional untuk membantu orang lain memahami tujuan dari peran tersebut.
7. Pilih Buat peran.
8. Buka peran yang baru saja Anda buat, dan salin peran ARN (misalnya, **`arn:aws:iam::111122223333:role/MyCrossAccountRepositoryContributorRole`**). Anda harus memberikan ARN ini ke administrator AccountB.

Akses repositori lintas akun: Tindakan untuk administrator di AccountB

Untuk mengizinkan pengguna atau grup di AccountB mengakses repositori di AccountA, administrator AccountB harus membuat sebuah grup di AccountB. Grup ini harus dikonfigurasi dengan kebijakan yang memungkinkan anggota grup untuk mengambil peran yang dibuat oleh administrator AccountA.

Bab berikut memberikan langkah dan contoh.

Topik

- [Langkah 1: Buatlah sebuah grup IAM untuk akses repositori bagi pengguna AccountB](#)
- [Langkah 2: Buatlah sebuah kebijakan dan tambahkan pengguna ke grup IAM](#)

Langkah 1: Buatlah sebuah grup IAM untuk akses repositori bagi pengguna AccountB

Cara termudah untuk mengelola pengguna IAM di AccountB yang dapat mengakses repositori AccountA adalah dengan membuat sebuah grup IAM di AccountB yang memiliki izin untuk mengambil peran di AccountA, dan kemudian menambahkan pengguna IAM ke grup tersebut.

Untuk membuat sebuah grup untuk akses repositori lintas-akun

1. Masuk ke Konsol AWS Manajemen sebagai pengguna IAM dengan izin yang diperlukan untuk membuat grup dan kebijakan IAM serta mengelola pengguna IAM di AccountB.
2. Buka konsol IAM di <https://console.aws.amazon.com/iam/>.
3. Di konsol IAM, pilih Grup.
4. Pilih Buat Grup Baru.
5. Di Nama Grup, masukkan nama untuk grup (misalnya, *DevelopersWithCrossAccountRepositoryAccess*). Pilih Langkah Selanjutnya.
6. di Lampirkan kebijakan, pilih Langkah Selanjutnya. Anda membuat kebijakan lintas-akun dalam prosedur berikutnya. Selesaikan membuat grup.

Langkah 2: Buatlah sebuah kebijakan dan tambahkan pengguna ke grup IAM

Sekarang setelah Anda memiliki grup, buatlah kebijakan yang memungkinkan anggota grup ini bisa mengambil peran yang memberi mereka akses ke repositori di AccountA. Kemudian tambahkan ke grup pengguna IAM di AccountB yang ingin Anda izinkan akses di AccountA.

Untuk membuat sebuah kebijakan grup dan menambahkan pengguna ke grup tersebut

1. Di konsol IAM, pilih Grup, lalu pilih nama grup yang baru saja Anda buat (misalnya, *DevelopersWithCrossAccountRepositoryAccess*).
2. Pilih tab Izin. Perluas Kebijakan Inline, lalu pilih tautan untuk membuat sebuah kebijakan inline. (Jika Anda mengkonfigurasi grup yang sudah memiliki kebijakan inline, pilih Buat Kebijakan Grup.)
3. Pilih Kebijakan Kustom, lalu pilih Pilihan.
4. Di Nama Kebijakan, masukkan nama untuk kebijakan (misalnya, *AccessPolicyForSharedRepository*).
5. Di Dokumen Kebijakan, tempelkan kebijakan berikut. ***DiResource, ganti ARN dengan ARN kebijakan yang dibuat oleh administrator di AccountA (misalnya,***

arn:aws:iam:: 111122223333:role/), lalu pilih Terapkan Kebijakan.

MyCrossAccountRepositoryContributorRole Untuk informasi selengkapnya tentang kebijakan yang dibuat oleh administrator di AccountA, lihat. [Langkah 1: Buatlah kebijakan untuk akses repositori di AccountA.](#)

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": "sts:AssumeRole",
    "Resource":
      "arn:aws:iam::111122223333:role/MyCrossAccountRepositoryContributorRole"
  }
}
```

6. Pilih tab Pengguna. Pilih Tambahkan Pengguna ke Grup, dan kemudian tambahkan pengguna IAM AccountB. Sebagai contoh, Anda dapat menambahkan pengguna IAM dengan nama pengguna *Saanvi_Sarkar* ke grup.

Note

Pengguna di accountB harus memiliki akses terprogram, termasuk kunci akses dan kunci rahasia, untuk mengkonfigurasi komputer lokal mereka untuk akses ke repositori bersama. CodeCommit Jika Anda membuat pengguna IAM, pastikan untuk menyimpan access key dan kunci rahasia. Untuk memastikan keamanan akun AWS Anda, secret access key hanya dapat diakses pada saat Anda membuatnya.

Akses repositori lintas akun: Tindakan untuk pengguna repositori di AccountB

Untuk mengakses repositori di AccountA, pengguna di grup AccountB harus mengkonfigurasi komputer lokal mereka untuk akses repositori. Bab berikut memberikan langkah dan contoh.

Topik

- [Langkah 1: Konfigurasi AWS CLI dan Git untuk pengguna AccountB untuk mengakses repositori di AccountA](#)
- [Langkah 2: Kloning dan akses CodeCommit repositori di AccountA](#)

Langkah 1: Konfigurasi AWS CLI dan Git untuk pengguna AccountB untuk mengakses repositori di AccountA

Anda tidak dapat menggunakan kunci SSH atau kredensial Git untuk mengakses repositori di akun Amazon Web Services lain. Pengguna AccountB harus mengonfigurasi komputer mereka untuk menggunakan `git-remote-codecommit` (disarankan) atau pembantu kredensial untuk mengakses CodeCommit repositori bersama di accountA. Namun, Anda dapat terus menggunakan kunci SSH atau kredensial Git saat mengakses repositori di AccountB.

Ikuti langkah-langkah ini untuk mengonfigurasi akses menggunakan `git-remote-codecommit`. Jika Anda belum menginstal `git-remote-codecommit`, unduh dari [git-remote-codecommit](#) situs web Python Package Index.

Untuk mengkonfigurasi AWS CLI dan Git untuk akses lintas akun

1. Instal AWS CLI di komputer lokal. Lihat petunjuk untuk sistem operasi Anda di [Menginstal AWS CLI](#).
2. Instal Git pada komputer lokal. Untuk menginstal Git, kami merekomendasikan situs web seperti [Unduh Git](#) atau [Git untuk Windows](#).

Note


CodeCommit mendukung Git versi 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami merekomendasikan penggunaan versi terbaru Git. Git adalah platform yang berkembang dan diperbarui secara teratur. Terkadang, perubahan fitur dapat memengaruhi cara kerjanya CodeCommit. Jika Anda mengalami masalah dengan versi Git tertentu dan CodeCommit, tinjau informasinya di [Memecahkan masalah](#).

3. Dari terminal atau baris perintah, di lokasi direktori tempat Anda ingin mengkloning repositori, jalankan perintah `git config --local user.name` dan `git config --local user.email` untuk mengatur nama pengguna dan email untuk commit yang akan Anda buat ke repositori. Sebagai contoh:

```
git config --local user.name "Saanvi Sarkar"  
git config --local user.email saanvi_sarkar@example.com
```

Perintah ini tidak mengembalikan apa pun, tetapi email dan nama pengguna yang Anda tentukan terkait dengan commit yang Anda buat ke repositori di AccountA.

4. Jalankan perintah `aws configure --profile` untuk mengkonfigurasi profil default untuk digunakan saat menyambung ke sumber daya di AccountB. Saat diminta, berikan access key dan kunci rahasia untuk pengguna IAM Anda.

 Note

Jika Anda telah menginstal AWS CLI dan mengkonfigurasi profil, Anda dapat melewati langkah ini.

Misalnya, jalankan perintah berikut untuk membuat AWS CLI profil default yang Anda gunakan untuk mengakses AWS sumber daya di accountB di US East (Ohio) (`us-east-2`):

```
aws configure
```

Saat diminta, berikan informasi berikut:

```
AWS Access Key ID [None]: Your-IAM-User-Access-Key
AWS Secret Access Key ID [None]: Your-IAM-User-Secret-Access-Key
Default region name ID [None]: us-east-2
Default output format [None]: json
```

5. Jalankan perintah `aws configure --profile` lagi untuk mengkonfigurasi profil bernama untuk digunakan ketika menghubungkan ke repositori di AccountA. Saat diminta, berikan access key dan kunci rahasia untuk pengguna IAM Anda. Misalnya, jalankan perintah berikut untuk membuat AWS CLI profil bernama *MyCrossAccountAccessProfile* yang Anda gunakan untuk mengakses repositori di AccountA di US East (Ohio) (`us-east-2`):

```
aws configure --profile MyCrossAccountAccessProfile
```

Saat diminta, berikan informasi berikut:

```
AWS Access Key ID [None]: Your-IAM-User-Access-Key
AWS Secret Access Key ID [None]: Your-IAM-User-Secret-Access-Key
Default region name ID [None]: us-east-2
Default output format [None]: json
```

6. Dalam editor teks biasa, buka file config, yang juga dikenal sebagai file konfigurasi AWS CLI . Bergantung pada sistem operasi Anda, file ini mungkin berada di `~/ .aws/config` di Linux, macOS, atau Unix, atau di `drive:\Pengguna\NAMA PENGGUNA\.aws\config` pada Windows.
7. Dalam file, cari entri yang sesuai dengan profil default yang Anda konfigurasi untuk akses ke repositori di AccountB. Itu terlihat serupa dengan yang berikut ini:

```
[default]
region = us-east-2
output = json
```

Tambahkan account ke konfigurasi profil. Berikan ID akun AWS AccountB. Sebagai contoh:

```
[default]
account = 888888888888
region = us-east-2
output = json
```

8. Dalam file, temukan entri yang sesuai dengan `MyCrossAccountAccessProfile` profil yang baru saja Anda buat. Itu terlihat serupa dengan yang berikut ini:

```
[profile MyCrossAccountAccessProfile]
region = us-east-2
output = json
```

Tambahkan account, role_arn dan source_profile ke konfigurasi profil. Sediakan ID akun Amazon Web Services AccountA, ARN peran di AccountA yang Anda ambil untuk mengakses repositori di akun lain, dan nama AWS CLI default di AccountB. Sebagai contoh:

```
[profile MyCrossAccountAccessProfile]
region = us-east-2
account = 111122223333
role_arn = arn:aws:iam::111122223333:role/MyCrossAccountRepositoryContributorRole
source_profile = default
output = json
```

Simpan perubahan Anda, dan tutup editor teks biasa.

Langkah 2: Kloning dan akses CodeCommit repositori di AccountA

Jalankan `git clone`, `git push`, dan `git pull` untuk mengkloning, mendorong ke, dan menarik dari repositori lintas akun CodeCommit. Anda juga dapat masuk ke Konsol AWS Manajemen, beralih peran, dan menggunakan CodeCommit konsol untuk berinteraksi dengan repositori di akun lain.

Note

Bergantung pada bagaimana peran IAM dikonfigurasi, Anda mungkin dapat melihat repositori pada halaman default untuk CodeCommit. Jika Anda tidak dapat melihat repositori, minta administrator repositori untuk mengirimkan Anda tautan URL ke halaman Kode untuk repositori bersama di konsol. CodeCommit URL tersebut serupa dengan yang berikut ini:

```
https://console.aws.amazon.com/codecommit/home?region=us-east-2#/repository/MySharedDemoRepo/browse/HEAD/--/
```

Untuk mengkloning repositori lintas-akun ke komputer lokal Anda

1. Pada baris perintah atau terminal, di direktori tempat Anda ingin mengkloning repositori, jalankan `git clone` perintah dengan URL klon HTTPS (GRC). Sebagai contoh:

```
git clone codecommit://MyCrossAccountAccessProfile@MySharedDemoRepo
```

Kecuali Anda menentukan sebaliknya, repositori dikloning ke subdirektori dengan nama yang sama dengan repositori.

2. Ubah direktori ke repositori yang dikloning, dan tambahkan atau buat perubahan pada file. Misalnya, Anda dapat menambahkan file bernama *NewFile.txt*.
3. Tambahkan file ke perubahan yang dilacak untuk repo lokal, komit perubahan, dan dorong file ke repositori. CodeCommit Sebagai contoh:


```
git add NewFile.txt  
git commit -m "Added a file to test cross-account access to this repository"  
git push
```

Untuk informasi selengkapnya, lihat [Memulai dengan Git dan AWS CodeCommit](#).

Sekarang setelah Anda menambahkan file, buka CodeCommit konsol untuk melihat komit Anda, tinjau perubahan pengguna lain pada repo, berpartisipasi dalam permintaan tarik, dan banyak lagi.

Untuk mengakses repositori lintas akun di konsol CodeCommit

1. Masuk ke akunTB (**88888888888888**) sebagai pengguna IAM yang telah diberikan akses lintas akun ke repositori di accountA. AWS Management Console
2. Pilih nama pengguna Anda di bilah navigasi, dan di daftar menurun, pilih Alihkan Peran.

 Note

Jika ini adalah pertama kalinya Anda telah memilih opsi ini, tinjau informasi pada halaman, dan kemudian pilih Alihkan Peran lagi.

3. Pada halaman Alihkan Peran, lakukan hal berikut:
 - Di Akun, masukkan ID akun untuk AccountA (misalnya, **111122223333**).
 - Dalam Peran, masukkan nama peran yang ingin Anda asumsikan untuk akses ke repositori di AccountA (misalnya, **MyCrossAccountRepositoryContributorRole**).
 - Di Nama Tampilan, masukkan nama yang ramah untuk peran ini. Nama ini muncul di konsol ketika Anda mengambil peran ini. Ini juga muncul dalam daftar peran yang diambil saat berikutnya Anda ingin beralih peran di konsol.
 - (Opsional) Di Warna, pilih label warna untuk nama tampilan tersebut.
 - Pilih Ganti Peran.

Untuk informasi selengkapnya, lihat [Beralih ke Peran \(AWS Management Console\)](#).

4. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.

Jika peran yang diambil memiliki izin untuk menampilkan nama repositori di AccountA, Anda melihat daftar repositori dan pesan kesalahan yang memberitahu Anda bahwa Anda tidak memiliki izin Untuk menampilkan status mereka. Ini adalah perilaku yang diharapkan. Pilih nama repositori yang dibagikan dari daftar.

Jika peran yang diambil tidak memiliki izin untuk menampilkan nama-nama repositori di AccountA, Anda melihat pesan kesalahan dan daftar kosong tanpa repositori. Tempelkan link URL ke repositori atau modifikasi tautan konsol dan ubah `/list` untuk nama repositori yang dibagikan (misalnya, `/MySharedDemoRepo`).

5. Di Kode, cari nama file yang ditambahkan dari komputer lokal Anda. Pilih untuk menelusuri kode dalam file, dan kemudian menelusuri sisa repositori dan mulai menggunakan fitur-fiturnya.

Untuk informasi selengkapnya, lihat [Memulai dengan AWS CodeCommit](#).

Hapus AWS CodeCommit repositori

Anda dapat menggunakan CodeCommit konsol atau AWS CLI untuk menghapus CodeCommit repositori.

Note

Menghapus repositori tidak menghapus salinan lokal dari repositori tersebut (repos lokal). Untuk menghapus repo lokal, gunakan direktori mesin lokal Anda dan alat pengelola file.

Topik

- [Hapus CodeCommit repositori \(konsol\)](#)
- [Hapus sebuah repo lokal](#)
- [Hapus CodeCommit repositori \(AWS CLI\)](#)

Hapus CodeCommit repositori (konsol)

Ikuti langkah-langkah ini untuk menggunakan CodeCommit konsol untuk menghapus CodeCommit repositori.

Important

Setelah Anda menghapus CodeCommit repositori, Anda tidak lagi dapat mengkloningnya ke repo lokal atau repo bersama. Anda juga tidak lagi dapat menarik data darinya, atau mendorong data kepadanya, dari repo lokal atau repo bersama. Tindakan ini tidak dapat dibatalkan.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori yang ingin Anda hapus.

3. Pada panel navigasi, silakan pilih Pengaturan.
4. Pada tab Umum, di Hapus repositori, pilih Hapus repositori. Masukkan **delete**, lalu pilih Hapus. Repositori dihapus secara permanen.

Note

Menghapus repositori di CodeCommit tidak menghapus repo lokal apa pun.

Hapus sebuah repo lokal

Gunakan direktori mesin lokal Anda dan alat pengelola file untuk menghapus direktori yang berisi repo lokal.

Menghapus repo lokal tidak menghapus CodeCommit repositori apa pun yang mungkin terhubung.

Hapus CodeCommit repositori ()AWS CLI

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal file AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk menggunakan AWS CLI untuk menghapus CodeCommit repositori, jalankan `delete-repository` perintah, tentukan nama CodeCommit repositori yang akan dihapus (dengan opsi). `--repository-name`

Important

Setelah Anda menghapus CodeCommit repositori, Anda tidak lagi dapat mengkloningnya ke repo lokal atau repo bersama. Anda juga tidak lagi dapat menarik data darinya, atau mendorong data kepadanya, dari repo lokal atau repo bersama. Tindakan ini tidak dapat dibatalkan.

Tip

Untuk mendapatkan nama repositori, jalankan perintah CodeCommit [list-repositori](#).

Misalnya, untuk menghapus repositori yang bernama MyDemoRepo:

```
aws codecommit delete-repository --repository-name MyDemoRepo
```

Jika berhasil, ID CodeCommit repositori yang dihapus secara permanen muncul di output:

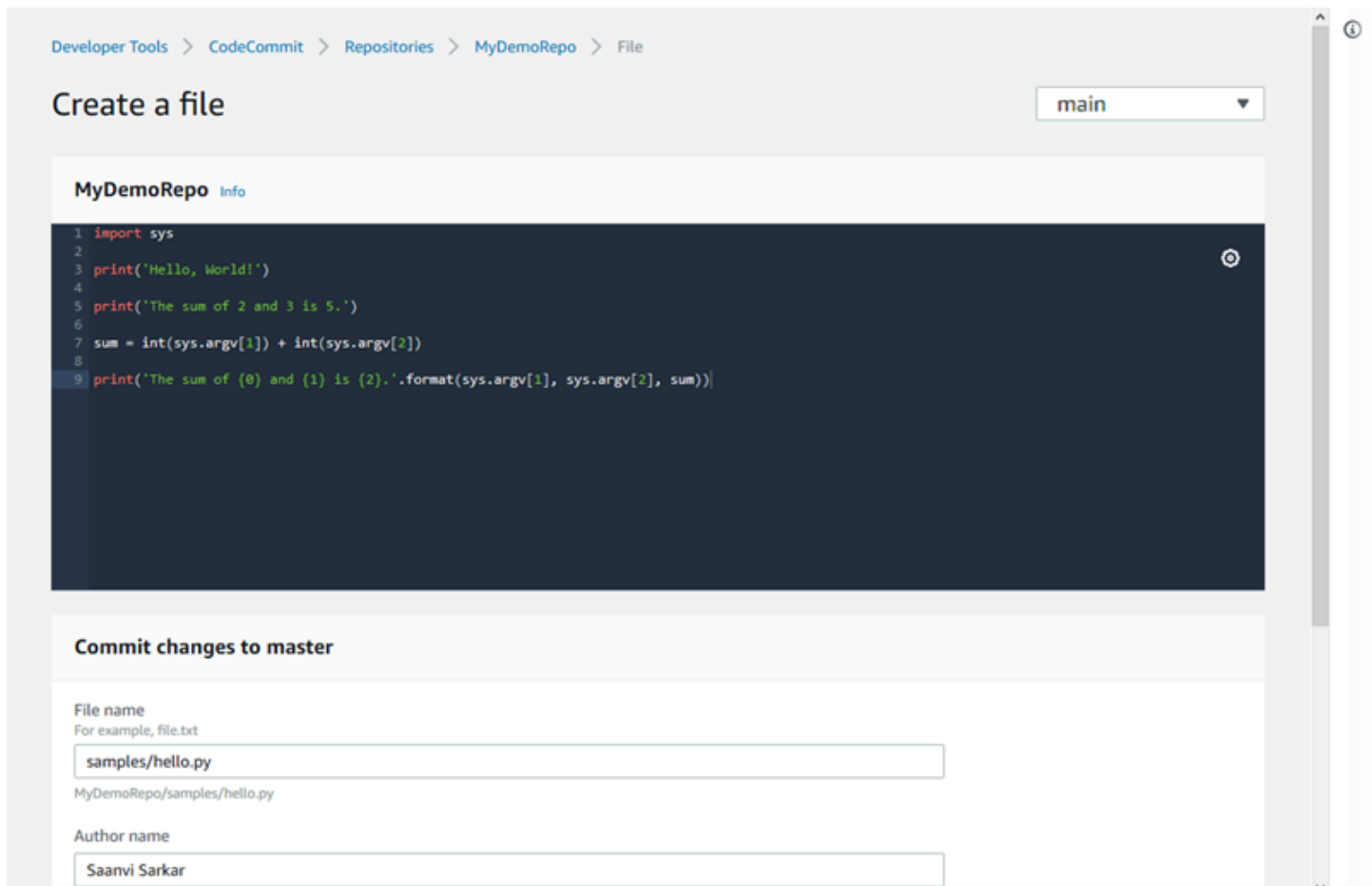
```
{
  "repositoryId": "f7579e13-b83e-4027-aaef-650c0EXAMPLE"
}
```

Menghapus CodeCommit repositori tidak menghapus repo lokal apa pun yang mungkin terhubung dengannya.

Bekerja dengan file repositori AWS CodeCommit

Dalam CodeCommit, file adalah versi terkendali, sebuah informasi mandiri yang tersedia untuk Anda dan pengguna lain dari repositori dan cabang tempat file disimpan. Anda dapat mengatur file repositori Anda dengan struktur direktori, seperti yang Anda lakukan di komputer. Tidak seperti komputer Anda, CodeCommit secara otomatis melacak setiap perubahan ke file. Anda dapat membandingkan versi file dan menyimpan versi file yang berbeda di cabang repositori yang berbeda.

Untuk menambah atau mengedit file dalam repositori, Anda dapat menggunakan klien Git. Anda juga dapat menggunakan konsol CodeCommit, AWS CLI, atau API CodeCommit.



Untuk informasi tentang bekerja dengan aspek-aspek lain dari repositori Anda di CodeCommit, lihat [Bekerja dengan repositori](#), [Bekerja dengan permintaan pull](#), [Bekerja dengan cabang](#), [Bekerja dengan commit](#), dan [Bekerja dengan preferensi pengguna](#).

Topik

- [Menelusuri file dalam sebuah repositori AWS CodeCommit](#)

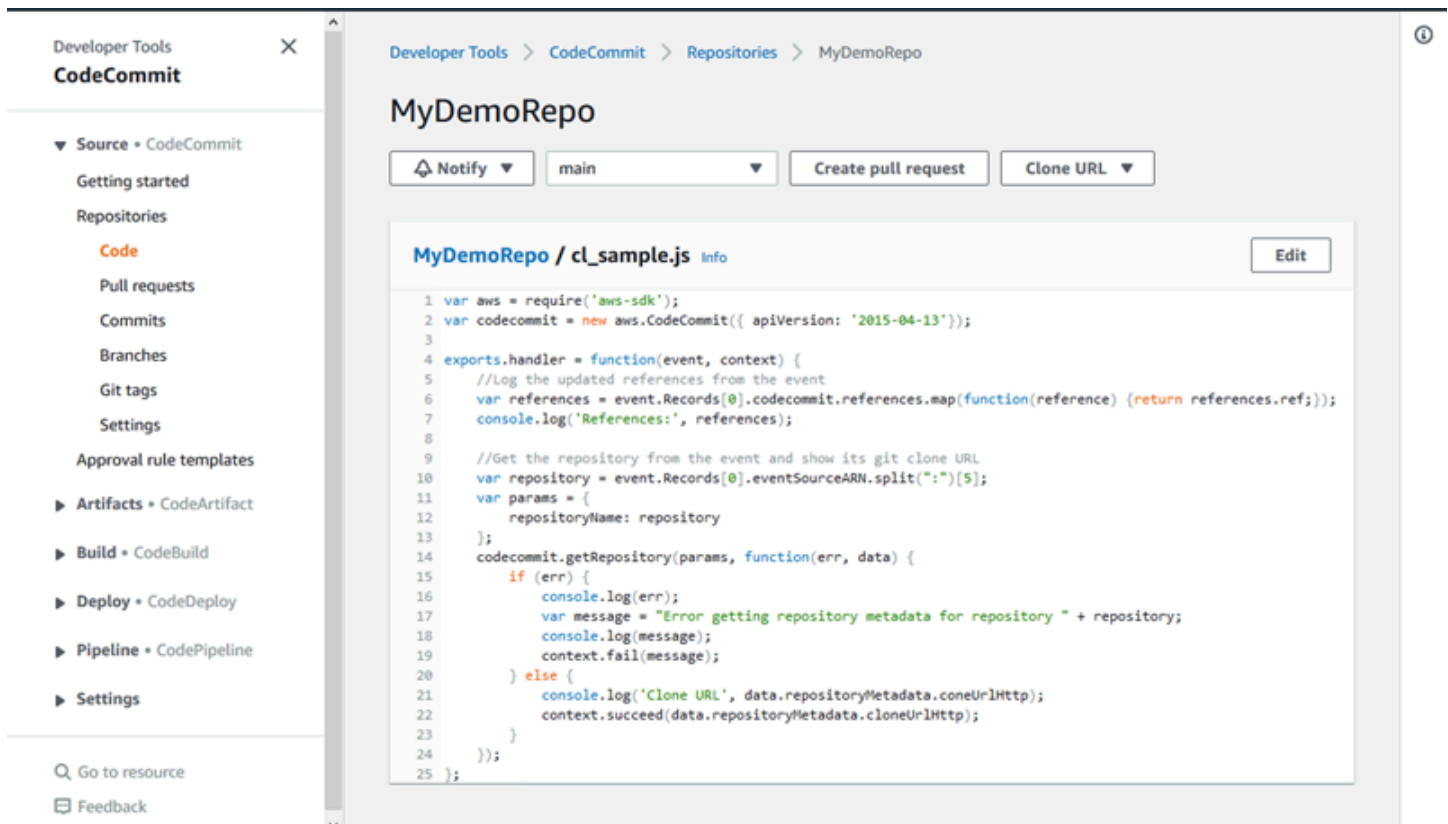
- [Membuat atau menambahkan file ke repositori AWS CodeCommit](#)
- [Mengedit isi file dalam repositori AWS CodeCommit](#)

Menelusuri file dalam sebuah repositori AWS CodeCommit

Setelah Anda terhubung ke CodeCommit repositori, Anda dapat mengkloning ke repo lokal atau menggunakan CodeCommit konsol untuk menelusuri isinya. Topik ini menjelaskan cara menggunakan CodeCommit konsol untuk menelusuri isi dari CodeCommit repositori.

Note

Untuk aktif CodeCommit pengguna, tidak ada biaya untuk menelusuri kode dari CodeCommit konsol. Untuk informasi tentang kapan biaya mungkin berlaku, lihat [Harga](#).



The screenshot displays the AWS CodeCommit console interface. On the left, a sidebar menu shows navigation options under 'Developer Tools' and 'CodeCommit', including 'Source', 'Artifacts', 'Build', 'Deploy', 'Pipeline', and 'Settings'. The main area shows the 'MyDemoRepo' repository page. At the top, there are buttons for 'Notify', a branch selector set to 'main', 'Create pull request', and 'Clone URL'. Below this, the file 'MyDemoRepo / cl_sample.js' is selected, with an 'Edit' button. The code content is displayed in a monospace font, showing a JavaScript handler function that logs references and clones a repository.

```
1 var aws = require('aws-sdk');
2 var codecommit = new aws.CodeCommit({ apiVersion: '2015-04-13'});
3
4 exports.handler = function(event, context) {
5     //Log the updated references from the event
6     var references = event.Records[0].codecommit.references.map(function(reference) {return reference.ref;});
7     console.log('References:', references);
8
9     //Get the repository from the event and show its git clone URL
10    var repository = event.Records[0].eventSourceARN.split(":")[5];
11    var params = {
12        repositoryName: repository
13    };
14    codecommit.getRepository(params, function(err, data) {
15        if (err) {
16            console.log(err);
17            var message = "Error getting repository metadata for repository " + repository;
18            console.log(message);
19            context.fail(message);
20        } else {
21            console.log('Clone URL', data.repositoryMetadata.cloneUrlHttp);
22            context.succeed(data.repositoryMetadata.cloneUrlHttp);
23        }
24    });
25 };
```

Menelusuri CodeCommit repositori

Anda dapat menggunakan CodeCommit konsol untuk meninjau file yang ada dalam sebuah repositori atau membaca isi file dengan cepat.

Untuk menelusuri isi dari sebuah repositori

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Pada Repositori, dari daftar repositori, pilih repositori yang ingin Anda telusuri.
3. Di Kode, telusuri isi dari cabang default untuk repositori Anda.

Untuk mengubah tampilan ke cabang atau tag yang berbeda, pilih tombol pemilih tampilan.

Pilih nama sebuah cabang atau tag dari daftar drop-down, atau di kotak filter, masukkan nama cabang atau tag, lalu pilih dari daftar.

4. Lakukan salah satu dari berikut:
 - Untuk melihat isi sebuah direktori, pilih direktori tersebut dari daftar. Anda dapat memilih salah satu direktori dalam daftar navigasi untuk kembali ke tampilan direktori itu. Anda juga dapat menggunakan panah atas di bagian atas daftar direktori.
 - Untuk melihat isi dari sebuah file, pilih file tersebut dari daftar. Jika file lebih besar dari batas objek commit, maka file itu tidak dapat ditampilkan di konsol dan harus dilihat di repo lokal sebagai gantinya. Untuk informasi selengkapnya, lihat [Kuota](#). Untuk keluar dari tampilan file, dari bilah navigasi kode, pilih direktori yang ingin Anda lihat.

Note

Tidak semua file biner dapat dilihat di konsol. Jika Anda memilih file biner dan berpotensi dapat dilihat, muncul pesan peringatan, yang meminta Anda untuk mengkonfirmasi bahwa Anda ingin menampilkan konten. Untuk melihat file, pilih Tampilkan isi file. Jika Anda tidak ingin melihat file, dari bilah navigasi kode, Anda pilih direktori yang ingin Anda lihat.

Jika Anda memilih file penurunan harga (.md), gunakan Markdown yang Diberikan dan Sumber Markdown tombol untuk beralih antara tampilan yang diberikan dan sintaks. Untuk informasi selengkapnya, lihat [Menggunakan Markdown di Konsol](#).

Membuat atau menambahkan file ke repositori AWS CodeCommit

Anda dapat menggunakan konsol CodeCommit, AWS CLI, atau klien Git untuk menambahkan file ke repositori. Anda dapat mengunggah file dari komputer lokal Anda ke repositori, atau Anda dapat menggunakan editor kode di konsol untuk membuat file. Editor adalah cara cepat dan mudah untuk menambahkan file sederhana, seperti file readme.md, ke cabang di repositori.

Upload a file

MyDemoRepo [Info](#)

Name	Size	Actions
<p>Upload file</p> <p>Choose a file to upload.</p> <p><input type="button" value="Choose file"/></p>		

Commit changes to master

Author name

Email address

Commit message - optional
A default commit message will be used if you do not provide one.

Topik

- [Membuat atau mengunggah file \(konsol\)](#)
- [Tambahkan file \(AWS CLI\)](#)
- [Menambahkan file \(Git\)](#)

Membuat atau mengunggah file (konsol)

Anda dapat menggunakan konsol CodeCommit untuk membuat file dan menembarkannya ke sebuah cabang di repositori CodeCommit. Sebagai bagian dari pembuatan file, Anda dapat memberikan nama pengguna dan alamat email. Anda juga dapat menambahkan pesan commit sehingga pengguna lain memahami siapa yang menambahkan file dan mengapa. Anda juga dapat mengunggah file langsung dari komputer lokal Anda ke cabang di repositori.

Untuk menambahkan file ke repositori

1. Buka konsol CodeCommit di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori tempat Anda ingin menambahkan file.
3. Di tampilan Kode, pilih cabang tempat Anda ingin menambahkan file. Secara default, isi dari cabang default ditampilkan ketika Anda membuka tampilan Kode.

Untuk mengubah tampilan ke cabang yang berbeda, pilih tombol pemilih tampilan. Pilih nama cabang dari daftar drop-down, atau di kotak filter, masukkan nama cabang, lalu pilih dari daftar.

4. Pilih Tambahkan file, kemudian pilih salah satu pilihan berikut:
 - Untuk menggunakan editor kode untuk membuat konten file dan menambahkannya ke repositori, pilih Buat file.
 - Untuk mengunggah file dari komputer lokal Anda ke repositori, pilih Unggah file.
5. Berikan informasi kepada pengguna lain tentang siapa yang menambahkan file ini ke repositori dan mengapa.
 - Di Nama penulis, masukkan nama Anda. Nama ini digunakan sebagai nama penulis dan nama pembuat commit dalam informasi commit. Pengaturan default CodeCommit untuk penggunaan nama pengguna IAM Anda atau derivasi login konsol Anda sebagai nama penulis.
 - Dalam Alamat Email, masukkan alamat email sehingga pengguna repositori lain dapat menghubungi Anda tentang perubahan ini.
 - Dalam Pesan commit, masukkan deskripsi singkat. Ini memang opsional, tetapi sangat direkomendasikan. Jika tidak, pesan commit default akan digunakan.
6. Lakukan salah satu dari berikut:
 - Jika Anda mengunggah file, pilih file dari komputer lokal Anda.
 - Jika Anda membuat file, masukkan konten yang ingin Anda tambahkan di editor kode, dan berikan nama untuk file tersebut.
7. Pilih Perubahan commit.

Tambahkan file (AWS CLI)

Anda dapat menggunakan perintah AWS CLI dan `put-file` untuk menambahkan file dalam repositori CodeCommit. Anda juga dapat menggunakan perintah `put-file` untuk menambahkan direktori atau struktur jalan untuk file.

Note

Untuk menggunakan perintah AWS CLI dengan CodeCommit, instal AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk menambahkan file ke repositori

1. Di komputer lokal Anda, buat file yang ingin Anda tambahkan ke repositori CodeCommit.
2. Pada terminal atau baris perintah, jalankan perintah `put-file`, dengan menentukan:
 - Repositori tempat Anda ingin menambahkan file.
 - Cabang tempat Anda ingin menambahkan file.
 - ID commit penuh dari commit terbaru yang dibuat untuk cabang, juga dikenal sebagai ujung atau kepala commit.
 - Lokasi lokal dari file tersebut. Sintaks yang digunakan untuk lokasi ini bervariasi, tergantung pada sistem operasi lokal Anda.
 - Nama file yang ingin Anda tambahkan, termasuk jalur di mana file yang diperbarui disimpan dalam repositori, jika ada.
 - Nama pengguna dan email yang ingin Anda kaitkan dengan file ini.
 - Sebuah pesan commit yang menjelaskan mengapa Anda menambahkan file ini.

Nama pengguna, alamat email, dan pesan commit bersifat opsional, tetapi membantu pengguna lain mengetahui siapa yang membuat perubahan dan mengapa. Jika Anda tidak menyediakan sebuah nama pengguna, CodeCommit default untuk menggunakan nama pengguna IAM Anda atau derivasi login konsol Anda sebagai nama penulis.

Misalnya, untuk menambahkan file bernama *ExampleSolution.py* ke repositori bernama *MyDemoRepo* ke cabang bernama *feature-randomizationfeature* yang perbaikannya terbaru memiliki ID *4c925148EXAMPLE*:

```
aws codecommit put-file --repository-name MyDemoRepo --branch-name feature-
randomizationfeature --file-content file://MyDirectory/ExampleSolution.py --file-
path /solutions/ExampleSolution.py --parent-commit-id 4c925148EXAMPLE --name "María
García" --email "maría_garcía@example.com" --commit-message "I added a third
randomization routine."
```

Note

Ketika Anda menambahkan file biner, pastikan bahwa Anda menggunakan `fileb://` untuk menentukan lokasi lokal file.

Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "blobId": "2eb4af3bEXAMPLE",
  "commitId": "317f8570EXAMPLE",
  "treeId": "347a3408EXAMPLE"
}
```

Menambahkan file (Git)

Anda dapat menambahkan file dalam repo lokal dan mengirim perubahan Anda ke repositori CodeCommit. Untuk informasi selengkapnya, lihat [Memulai dengan Git dan AWS CodeCommit](#).

Mengedit isi file dalam repositori AWS CodeCommit

Anda dapat menggunakan CodeCommit konsolAWS CLI, atau klien Git untuk mengedit isi file dalam repositori file dalam CodeCommit repositori.

2. Di Repositori, pilih repositori tempat Anda ingin mengedit file.
3. Di tampilan Kode, pilih cabang tempat Anda ingin mengedit file. Secara default, isi dari cabang default ditampilkan ketika Anda membuka tampilan Kode.

Untuk mengubah tampilan ke cabang yang berbeda, pilih tombol pemilih tampilan. Pilih nama cabang dari daftar drop-down, atau di kotak filter, masukkan nama cabang, lalu pilih dari daftar.

4. Navigasikan isi cabang dan pilih file yang ingin Anda edit. Dalam tampilan file, pilih Edit.

Note

Jika Anda memilih file biner, muncul pesan peringatan yang meminta Anda untuk mengkonfirmasi bahwa Anda ingin menampilkan konten. Anda tidak harus menggunakan konsol untuk mengedit file file biner untuk menggunakan file biner untuk menggunakan CodeCommit konsol untuk mengedit file biner.

5. Mengedit file, dan memberikan informasi kepada pengguna lain tentang siapa yang membuat perubahan ini dan mengapa.
 - Di Nama penulis, masukkan nama Anda. Nama ini digunakan sebagai nama penulis dan nama pembuat commit dalam informasi commit. CodeCommit Pengaturan default untuk penggunaan nama pengguna IAM Anda atau derivasi login konsol Anda sebagai nama penulis.
 - Dalam Alamat Email, masukkan alamat email sehingga pengguna repositori lain dapat menghubungi Anda tentang perubahan ini.
 - Dalam Pesan commit, masukkan deskripsi singkat tentang perubahan Anda.
6. Pilih Commit perubahan untuk menyimpan perubahan Anda ke file dan commit perubahan tersebut ke repositori.

Mengedit atau menghapus file (AWS CLI)

Anda dapat menggunakan `put-file` perintah AWS CLI dan untuk membuat perubahan pada suatu file dalam repositori suatu file dalam CodeCommit repositori. Anda juga dapat menggunakan perintah `put-file` untuk menambahkan direktori atau struktur jalan untuk file yang diubah, jika Anda ingin menyimpan file yang diubah di lokasi yang berbeda dari aslinya. Jika Anda ingin menghapus file seluruhnya, Anda dapat menggunakan perintah `delete-file`.

Note

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk mengedit file dalam repositori

1. Menggunakan salinan lokal dari file, membuat perubahan yang ingin Anda tambahkan ke CodeCommit repositori.
2. Pada terminal atau baris perintah, jalankan perintah `put-file`, dengan menentukan:
 - Repositori tempat Anda ingin menambah file yang diedit.
 - Cabang tempat Anda ingin menambahkan file yang diedit.
 - ID commit penuh dari commit terbaru yang dibuat untuk cabang, juga dikenal sebagai ujung atau kepala commit.
 - Lokasi lokal dari file tersebut.
 - Nama file yang diperbarui yang ingin Anda tambahkan, termasuk jalur di mana file yang diperbarui disimpan dalam repositori, jika ada.
 - Nama pengguna dan email yang ingin Anda kaitkan dengan perubahan file ini.
 - Pesan commit yang menjelaskan perubahan yang Anda buat.

Nama pengguna, alamat email, dan pesan commit bersifat opsional, tetapi membantu pengguna lain mengetahui siapa yang membuat perubahan dan mengapa. Jika Anda tidak menyediakan sebuah nama pengguna, CodeCommit penggunaan nama pengguna IAM Anda atau derivasi login konsol Anda.

Misalnya, untuk menambahkan hasil editan yang dibuat pada file bernama

ExampleSolution.py ke repositori bernama *MyDemoRepo* cabang bernama *feature-randomizationfeature* yang perbaikan terbarunya memiliki ID *4c925148EXAMPLE*:

```
aws codecommit put-file --repository-name MyDemoRepo --branch-name feature-  
randomizationfeature --file-content file://MyDirectory/ExampleSolution.py --file-  
path /solutions/ExampleSolution.py --parent-commit-id 4c925148EXAMPLE --name "María  
García" --email "maría_garcía@example.com" --commit-message "I fixed the bug Mary  
found."
```

Note

Jika Anda ingin menambahkan file biner yang diubah, pastikan untuk menggunakan `--file-content` dengan notasi `fileb://MyDirectory/MyFile.raw`.

Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "blobId": "2eb4af3bEXAMPLE",
  "commitId": "317f8570EXAMPLE",
  "treeId": "347a3408EXAMPLE"
}
```

Untuk menghapus file, gunakan perintah `delete-file`. Misalnya, untuk menghapus file bernama `README.md` di sebuah cabang bernama `utama` dengan ID commit terbaru dari `c5709475EXAMPLE` dalam repositori bernama `MyDemoRepo`:

```
aws codecommit delete-file --repository-name MyDemoRepo --branch-name main --file-path README.md --parent-commit-id c5709475EXAMPLE
```

Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "blobId": "559b44fEXAMPLE",
  "commitId": "353cf655EXAMPLE",
  "filePath": "README.md",
  "treeId": "6bc824cEXAMPLE"
}
```

Edit file (Git)

Anda dapat mengedit file dalam repo lokal dan mengirim perubahan Anda ke repositori Anda ke CodeCommit repositori. Untuk informasi selengkapnya, lihat [Memulai dengan Git dan AWS CodeCommit](#).

Bekerja dengan permintaan pull di repositori AWS CodeCommit

Permintaan pull adalah cara utama Anda dan pengguna repositori lainnya dapat meninjau, mengomentari, dan menggabungkan perubahan kode dari satu cabang ke cabang lainnya. Anda dapat menggunakan permintaan pull untuk secara kolaboratif meninjau perubahan kode untuk perubahan kecil atau perbaikan, penambahan fitur utama, atau versi baru dari perangkat lunak yang dirilis. Berikut adalah salah satu alur kerja yang mungkin untuk permintaan pull:

Li Juan, developer yang bekerja di repo bernama MyDemoRepo, ingin mengerjakan fitur baru untuk versi produk yang akan datang. Untuk menjaga pekerjaan terpisah dari kode siap produksi, ini menciptakan cabang terpisah dari cabang default dan menamakannya *feature-randomizationfeature*. Dia menulis kode, membuat commit, dan mendorong kode fitur baru ke cabang ini. Dia menginginkan pengguna repositori lain untuk meninjau kode untuk kualitas sebelum dia menggabungkan perubahannya ke cabang default. Untuk melakukan ini, dia membuat permintaan pull. Permintaan pull berisi perbandingan antara cabang kerjanya dan cabang kode di mana dia bermaksud untuk menggabungkan perubahannya (dalam hal ini, cabang default). Dia juga dapat membuat aturan persetujuan yang mengharuskan sejumlah pengguna tertentu untuk menyetujui permintaan pull-nya. Dia bahkan bisa menentukan grup persetujuan pengguna. Pengguna lain meninjau kode dan perubahannya, menambahkan komentar dan saran. Dia mungkin memperbarui cabang kerjanya beberapa kali dengan perubahan kode dalam menanggapi komentar. Perubahannya dimasukkan ke dalam permintaan pull setiap kali dia mengirimnya ke cabang di CodeCommit. Dia mungkin juga menggabungkan perubahan yang telah dibuat di cabang tujuan yang dimaksudkan sementara permintaan pull terbuka, sehingga pengguna dapat yakin mereka meninjau semua perubahan yang diusulkan dalam konteks. Ketika dia dan pengulasnya puas, dan kondisi untuk aturan persetujuan (jika ada) telah puas, dia atau salah satu pengulasnya menggabungkan kode dan menutup permintaan pull.

Developer Tools > CodeCommit > Repositories > MyDemoRepo > Pull requests > Create pull request

Create pull request

Destination: main Source: bugfix-1236 Compare Cancel

Mergeable
There are currently no conflicts between bugfix-1236 and main. You can close this pull request by merging it in the AWS CodeCommit console.

Details Create pull request

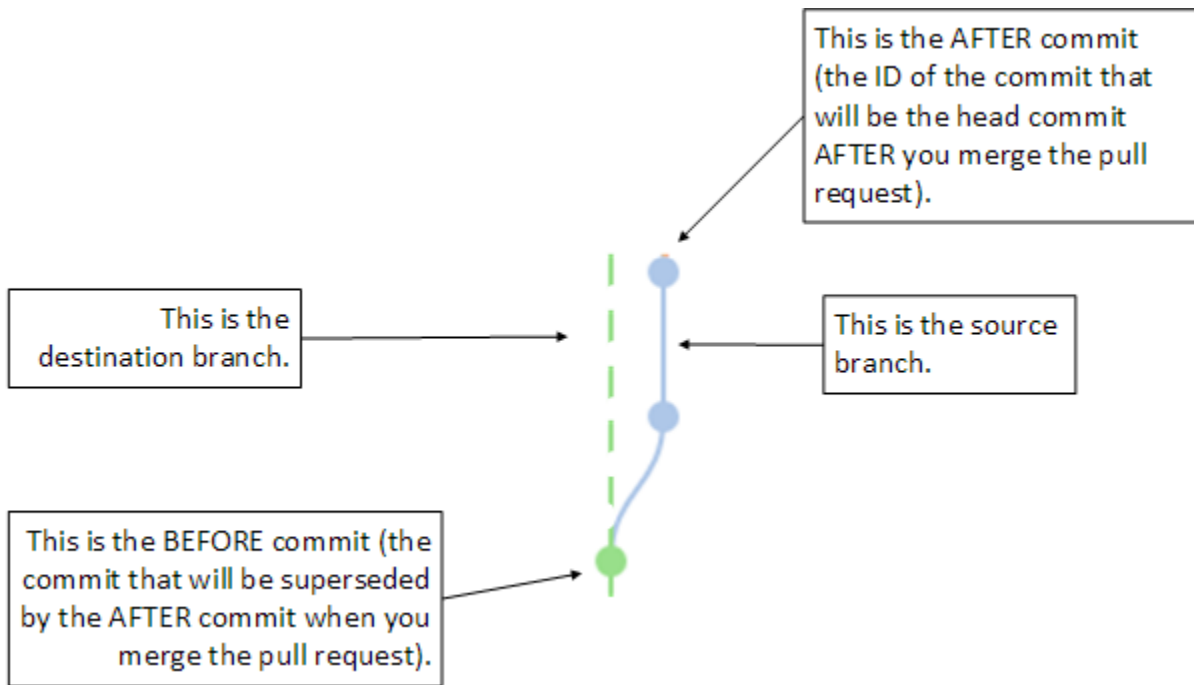
Title
Review changes for bugfix-1236
150 characters maximum

Description - optional Preview markdown [Learn more](#)

I've added some code for the bucket creation issue. Please review by Tuesday.

Changes | Commits

Permintaan pull memerlukan dua cabang: cabang sumber yang berisi kode yang ingin Anda tinjau, dan cabang tujuan, di mana Anda menggabungkan kode yang ditinjau. Cabang sumber berisi commit **SETELAH**, yang merupakan komit yang berisi perubahan yang ingin Anda gabungkan ke cabang tujuan. Cabang tujuan berisi commit **SEBELUM**, yang mewakili keadaan kode sebelum cabang permintaan pull digabung ke cabang tujuan. Pilihan strategi penggabungan mempengaruhi rincian bagaimana melakukan digabung antara sumber dan cabang tujuan di konsol CodeCommit. Untuk informasi selengkapnya tentang strategi penggabungan di CodeCommit, lihat [Menggabungkan permintaan pull \(konsol\)](#).



Permintaan pull menampilkan perbedaan antara ujung cabang sumber dan commit terbaru pada cabang tujuan ketika permintaan pull dibuat, sehingga pengguna dapat melihat dan mengomentari perubahan. Anda dapat memperbarui permintaan pull dalam menanggapi komentar dengan melakukan dan mengirim perubahan ke cabang sumber.

Ketika kode Anda telah ditinjau, dan persyaratan aturan persetujuan (jika ada) telah terpenuhi, Anda dapat menutup permintaan pull dengan salah satu dari beberapa cara:

- Menggabungkan cabang-cabang lokal dan mengirim perubahan Anda. Ini menutup permintaan secara otomatis jika strategi penggabungan cepat-maju digunakan dan tidak ada konflik gabungan.
- Gunakan konsol AWS CodeCommit untuk menutup permintaan pull tanpa penggabungan, menyelesaikan konflik dalam gabungan, atau, jika tidak ada konflik, menutup dan menggabungkan cabang menggunakan salah satu strategi penggabungan yang tersedia.
- Gunakan AWS CLI.

Sebelum Anda membuat permintaan pull:

- Pastikan bahwa Anda telah membuat commit dan mengirim perubahan kode yang ingin Anda tinjau ke cabang (sumber cabang).
- Mengatur pemberitahuan untuk repositori Anda, sehingga pengguna lain dapat diberitahu tentang permintaan pull dan perubahan untuk itu. (Langkah ini memang opsional, tetapi direkomendasikan.)
- Buat dan kaitkan templat aturan persetujuan dengan repositori Anda, sehingga aturan persetujuan dibuat secara otomatis untuk permintaan pull untuk membantu memastikan kualitas kode. Untuk informasi selengkapnya, lihat [Bekerja dengan templat aturan persetujuan](#).

Permintaan pull lebih efektif ketika Anda telah mengatur pengguna IAM untuk pengguna repositori Anda di akun Amazon Web Services Anda. Lebih mudah untuk mengidentifikasi pengguna mana yang membuat komentar. Keuntungan lainnya adalah bahwa pengguna IAM dapat menggunakan kredensial Git untuk akses repositori. Untuk informasi selengkapnya, lihat [Langkah 1: Konfigurasi awal untuk CodeCommit](#). Anda dapat menggunakan permintaan pull dengan jenis pengguna lainnya, termasuk pengguna akses gabungan.

Untuk informasi tentang bekerja dengan aspek lain dari repositori Anda di CodeCommit, lihat [Bekerja dengan repositori](#), [Bekerja dengan templat aturan persetujuan](#), [Bekerja dengan file](#), [Bekerja dengan commit](#), [Bekerja dengan cabang](#), dan [Bekerja dengan preferensi pengguna](#).

Topik

- [Membuat sebuah pull request](#)
- [Buat aturan persetujuan untuk permintaan pull](#)
- [Tampilkan permintaan pull dalam repositori AWS CodeCommit](#)
- [Meninjau permintaan pull](#)
- [Memperbarui sebuah permintaan pull](#)

- [Mengedit atau menghapus aturan persetujuan untuk permintaan pull](#)
- [Menimpa aturan persetujuan pada sebuah permintaan pull](#)
- [Menggabungkan permintaan pull dalam repositori AWS CodeCommit](#)
- [Menyelesaikan konflik dalam permintaan pull dalam repositori AWS CodeCommit](#)
- [Menutup sebuah pull request dalam sebuah repositori AWS CodeCommit](#)

Membuat sebuah pull request

Membuat pull request akan membantu pengguna lain melihat dan meninjau perubahan kode Anda sebelum Anda menggabungkan mereka ke cabang lain. Pertama, Anda harus membuat sebuah cabang untuk perubahan kode Anda. Hal ini disebut sebagai cabang sumber untuk sebuah pull request. Setelah Anda melakukan commit dan push pada perubahan ke repositori, Anda dapat membuat pull request yang membandingkan isi dari cabang tersebut (sumber cabang) ke cabang di mana Anda ingin menggabungkan perubahan Anda setelah pull request ditutup (cabang tujuan).

Anda dapat menggunakan konsol AWS CodeCommit atau AWS CLI untuk membuat pull request untuk repositori Anda.

Topik

- [Membuat sebuah pull request \(konsol\)](#)
- [Membuat sebuah pull request \(AWS CLI\)](#)

Membuat sebuah pull request (konsol)

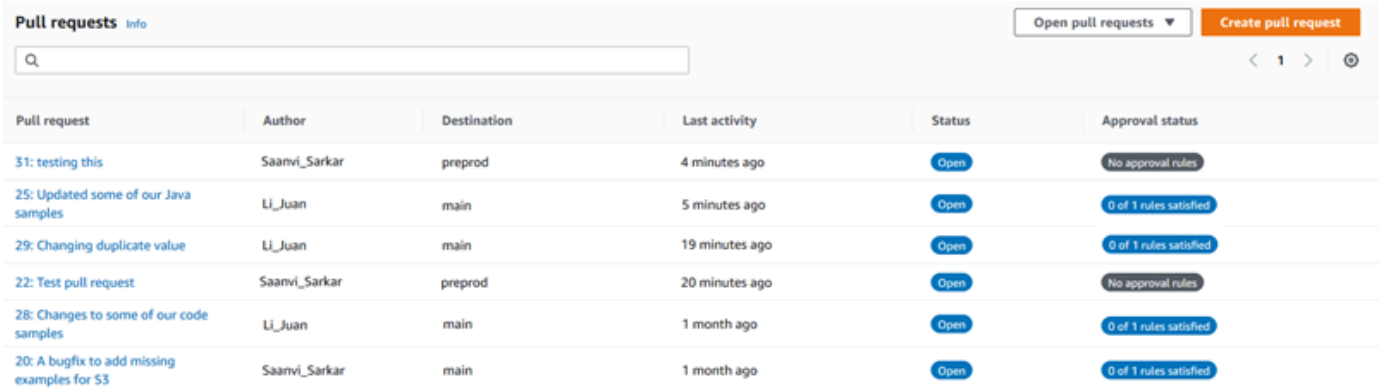
Anda dapat menggunakan konsol CodeCommit untuk membuat sebuah pull request dalam sebuah repositori CodeCommit. Jika repositori Anda [dikonfigurasi dengan notifikasi](#), pengguna yang berlangganan akan menerima email saat Anda membuat sebuah pull request.

1. Buka konsol CodeCommit di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin membuat sebuah pull request.
3. Di panel navigasi, pilih Permintaan Pull.

Tip

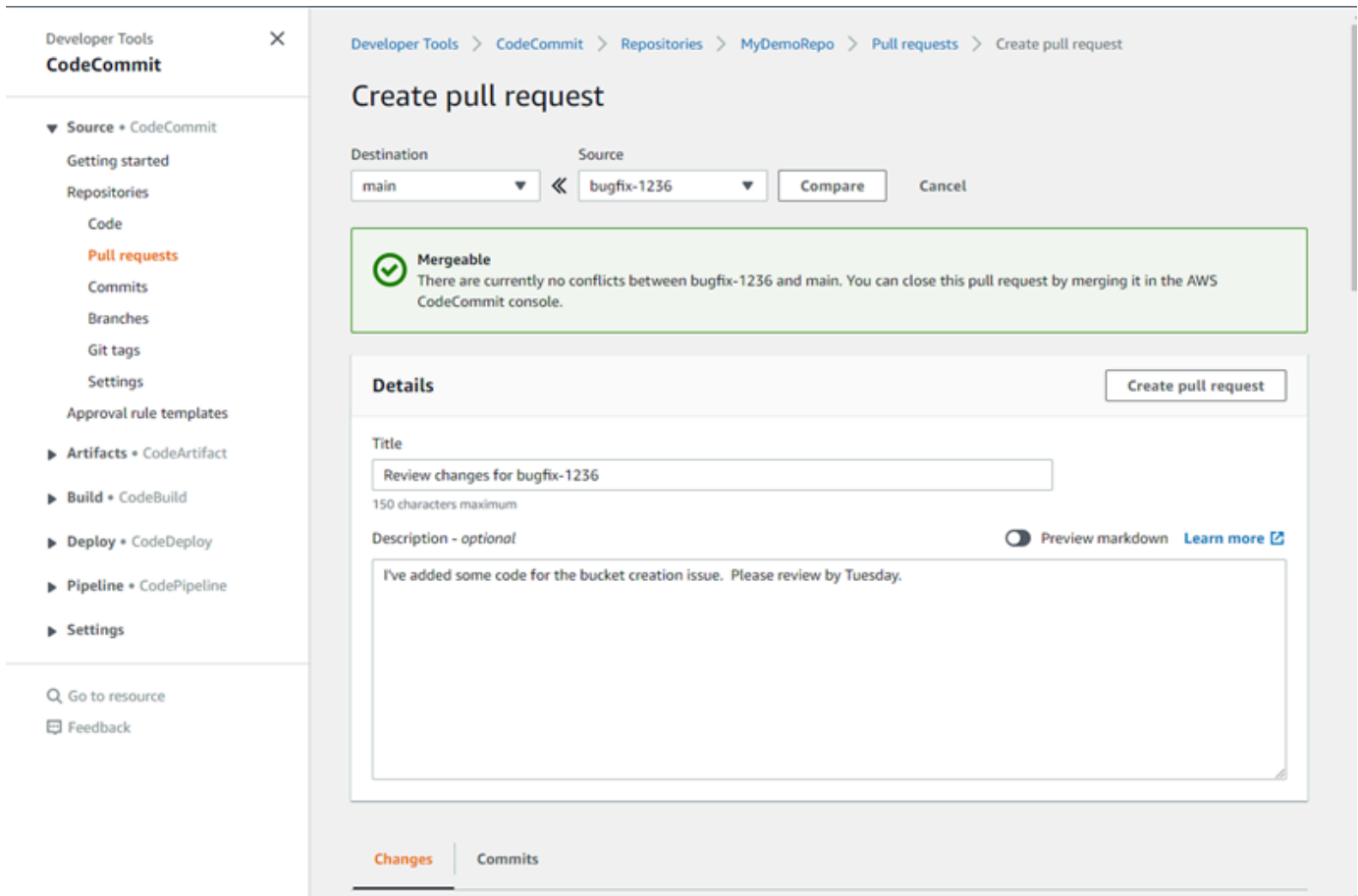
Anda juga dapat membuat pull request dari Cabang dan Kode.

4. Pilih Membuat permintaan pull.



Pull request	Author	Destination	Last activity	Status	Approval status
31: testing this	Saanvi_Sarkar	preprod	4 minutes ago	Open	No approval rules
25: Updated some of our Java samples	Li_Juan	main	5 minutes ago	Open	0 of 1 rules satisfied
29: Changing duplicate value	Li_Juan	main	19 minutes ago	Open	0 of 1 rules satisfied
22: Test pull request	Saanvi_Sarkar	preprod	20 minutes ago	Open	No approval rules
28: Changes to some of our code samples	Li_Juan	main	1 month ago	Open	0 of 1 rules satisfied
20: A bugfix to add missing examples for S3	Saanvi_Sarkar	main	1 month ago	Open	0 of 1 rules satisfied

5. Di Buat permintaan pull, di Sumber, pilih cabang yang berisi perubahan yang ingin Anda tinjau.
6. Di Tujuan, pilih cabang di mana Anda berniat untuk menggabungkan perubahan kode Anda ketika permintaan pull ditutup.
7. Pilih Bandingkan. Perbandingan berjalan pada dua cabang, dan perbedaan di antara keduanya akan ditampilkan. Analisis juga dilakukan untuk menentukan apakah dua cabang tersebut dapat digabung secara otomatis ketika permintaan pull ditutup.
8. Tinjau detail perbandingan dan perubahan untuk memastikan bahwa permintaan pull berisi perubahan dan perbaikan yang ingin Anda tinjau. Jika tidak, sesuaikan pilihan Anda untuk cabang sumber dan tujuan, lalu pilih Bandingkan sekali lagi.
9. Ketika Anda puas dengan hasil perbandingan pull request, pada Judul, masukkan deskripsi singkat namun deskriptif untuk tinjauan ini. Ini adalah judul yang muncul dalam daftar permintaan pull untuk repositori.
10. (Opsional) Di Deskripsi, masukkan detail tentang tinjauan ini dan informasi berguna lainnya untuk peninjau.
11. Pilih Create (Buat).



Permintaan pull Anda muncul dalam daftar permintaan pull untuk repositori. Jika Anda [telah mengkonfigurasi notifikasi](#), pelanggan untuk topik Amazon SNS akan menerima email untuk menginformasikan mereka tentang permintaan pull yang baru saja dibuat.

Membuat sebuah pull request (AWS CLI)

Untuk menggunakan perintah AWS CLI dengan CodeCommit, instal AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk menggunakan AWS CLI untuk membuat sebuah pull request dalam repositori CodeCommit

1. Jalankan perintah `create-pull-request`, dengan menentukan:
 - Nama pull request (dengan opsi `--title`).
 - Deskripsi pull request (dengan opsi `--description`).
 - Daftar target untuk perintah `create-pull-request`, termasuk:

- Nama repositori CodeCommit dimana permintaan pull dibuat (dengan atribut `repositoryName`).
- Nama cabang yang berisi perubahan kode yang ingin Anda tinjau, juga dikenal sebagai cabang sumber (dengan atribut `sourceReference`).
- (Opsional) Nama cabang di mana Anda berniat untuk menggabungkan perubahan kode Anda, juga dikenal sebagai cabang tujuan, jika Anda tidak ingin menggabungkan ke cabang default (dengan atribut `destinationReference`).
- Sebuah token idempotensi yang unik dan dihasilkan klien (dengan opsi `--client-request-token`).

Contoh ini membuat sebuah permintaan pull bernama *Peng analisis kesulitan pengucapan* dengan deskripsi *Silakan tinjau perubahan ini pada hari Selasa* yang menargetkan cabang sumber *jane-branch*. Pull request tersebut akan digabung ke cabang default *utama* dalam repositori CodeCommit bernama MyDemoRepo:

```
aws codecommit create-pull-request --title "Pronunciation difficulty analyzer"
--description "Please review these changes by Tuesday" --client-request-token
123Example --targets repositoryName=MyDemoRepo,sourceReference=jane-branch
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\",
        \"DestinationReferences\": [\"refs/heads/main\"], \"Statements\": [{\"Type
        \": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":
        [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "originApprovalRuleTemplate": {
          "approvalRuleTemplateId": "dd3d22fe-EXAMPLE",
          "approvalRuleTemplateName": "2-approver-rule-for-main"
        },
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ]
  }
}
```

```
    ],
    "authorArn": "arn:aws:iam::111111111111:user/Jane_Doe",
    "description": "Please review these changes by Tuesday",
    "title": "Pronunciation difficulty analyzer",
    "pullRequestTargets": [
      {
        "destinationCommit": "5d036259EXAMPLE",
        "destinationReference": "refs/heads/main",
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "317f8570EXAMPLE",
        "sourceReference": "refs/heads/jane-branch",
        "mergeMetadata": {
          "isMerged": false
        }
      }
    ],
    "lastActivityDate": 1508962823.285,
    "pullRequestId": "42",
    "clientRequestToken": "123Example",
    "pullRequestStatus": "OPEN",
    "creationDate": 1508962823.285
  }
}
```

Buat aturan persetujuan untuk permintaan pull

Membuat aturan persetujuan untuk permintaan pull Anda membantu memastikan kualitas kode Anda dengan mewajibkan pengguna untuk menyetujui permintaan pull sebelum kode dapat digabung ke cabang tujuan. Anda dapat menentukan jumlah pengguna yang harus menyetujui permintaan pull. Anda juga dapat menentukan grup persetujuan pengguna untuk aturan tersebut. Jika Anda melakukannya, hanya persetujuan dari pengguna yang dihitung terhadap jumlah persetujuan yang diperlukan untuk aturan tersebut.

Note

Anda juga dapat membuat templat aturan persetujuan, yang dapat membantu Anda mengotomatisasi pembuatan aturan persetujuan di seluruh repositori yang akan berlaku untuk setiap permintaan pull. Untuk informasi selengkapnya, lihat [Bekerja dengan templat aturan persetujuan](#).

Anda dapat menggunakan konsol AWS CodeCommit atau AWS CLI untuk membuat aturan persetujuan untuk repositori Anda.

Topik

- [Buat aturan persetujuan untuk permintaan pull \(konsol\)](#)
- [Buat aturan persetujuan untuk permintaan pull \(AWS CLI\)](#)

Buat aturan persetujuan untuk permintaan pull (konsol)

Anda dapat menggunakan CodeCommit konsol untuk membuat aturan persetujuan untuk permintaan pull dalam CodeCommit repositori.

1. Membuka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Dalam Repositori, pilih nama repositori tempat Anda ingin membuat aturan persetujuan untuk permintaan pull.
3. Di panel navigasi, pilih Permintaan Pull.
4. Pilih permintaan pull yang ingin Anda buat aturan persetujuan dari daftar. Anda hanya dapat membuat aturan persetujuan untuk permintaan pull terbuka.

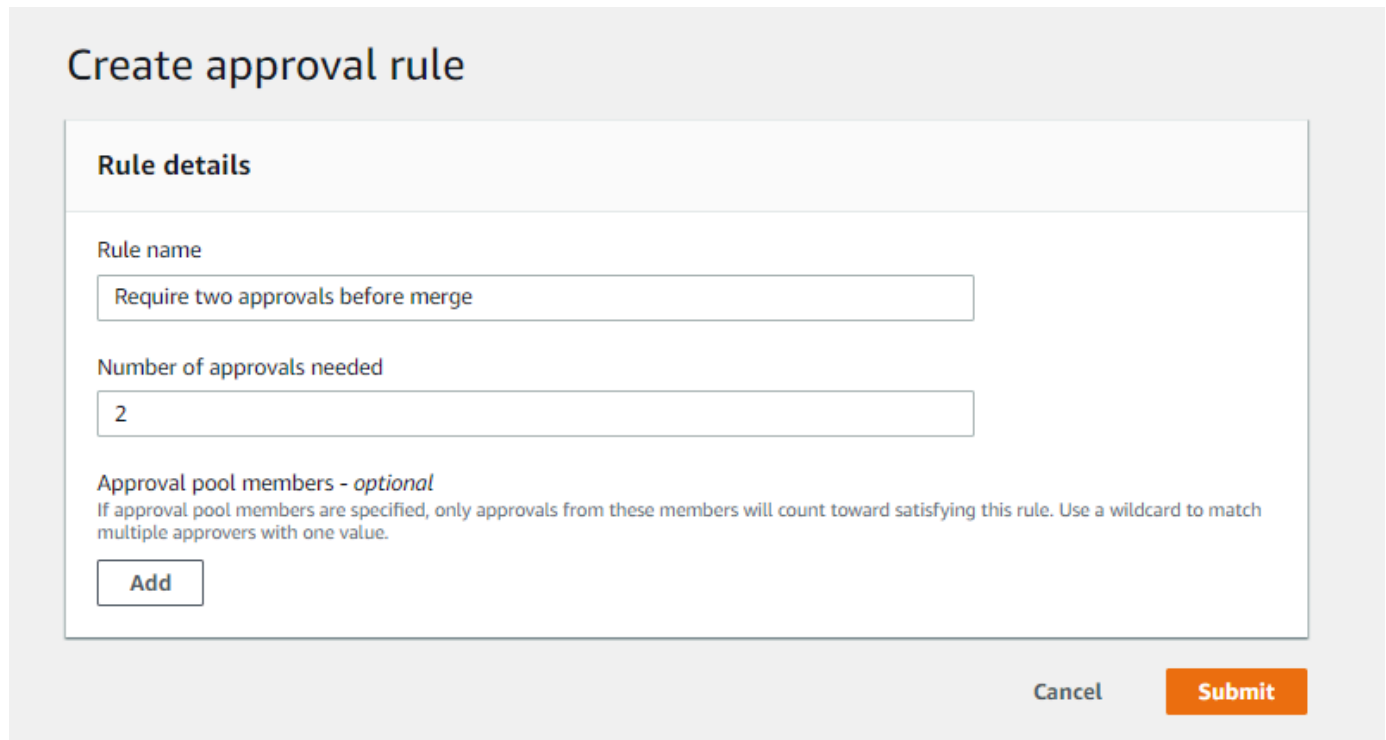
Pull request	Author	Destination	Last activity	Status	Approval status
31: testing this	Saanvi_Sarkar	preprod	4 minutes ago	Open	No approval rules
25: Updated some of our Java samples	Li_Juan	main	5 minutes ago	Open	0 of 1 rules satisfied
29: Changing duplicate value	Li_Juan	main	19 minutes ago	Open	0 of 1 rules satisfied
22: Test pull request	Saanvi_Sarkar	preprod	20 minutes ago	Open	No approval rules
28: Changes to some of our code samples	Li_Juan	main	1 month ago	Open	0 of 1 rules satisfied
20: A bugfix to add missing examples for S3	Saanvi_Sarkar	main	1 month ago	Open	0 of 1 rules satisfied

5. Dalam permintaan pull, pilih Persetujuan, lalu pilih Buat aturan persetujuan.
6. Dalam Nama aturan, berikan aturan nama deskriptif sehingga Anda tahu untuk apa itu. Misalnya, jika Anda ingin meminta dua orang untuk menyetujui permintaan pull sebelum dapat digabung, Anda dapat menyebutkan aturan **Require two approvals before merge**.

Note

Anda tidak dapat mengubah nama grup persetujuan setelah Anda membuatnya.

Dalam Jumlah persetujuan yang dibutuhkan, masukkan jumlah yang Anda inginkan. Default-nya adalah 1.



Create approval rule

Rule details

Rule name
Require two approvals before merge

Number of approvals needed
2

Approval pool members - *optional*
If approval pool members are specified, only approvals from these members will count toward satisfying this rule. Use a wildcard to match multiple approvers with one value.

Add

Cancel Submit

7. (Opsional) Jika Anda ingin meminta persetujuan untuk permintaan pull yang berasal dari grup pengguna tertentu, dalam Anggota aturan persetujuan, pilih Tambahkan. Dalam Jenis penyetuju, pilih salah satu langkah berikut:

- Nama pengguna IAM atau peran yang diasumsikan: Opsi ini prepopulates AWSID akun dengan akun yang Anda gunakan untuk masuk, dan hanya memerlukan nama. Ini dapat digunakan untuk baik oleh pengguna IAM maupun pengguna akses gabungan yang namanya cocok dengan nama yang disediakan. Ini adalah pilihan yang sangat kuat yang menawarkan fleksibilitas tinggi. Misalnya, jika Anda masuk dengan akun Amazon Web Services 123456789012 dan memilih opsi ini, dan Anda menentukan **Mary_Major**, semua hal berikut dihitung sebagai persetujuan yang berasal dari pengguna tersebut:
 - Pengguna IAM di akun (`arn:aws:iam::123456789012:user/Mary_Major`)
 - Seorang pengguna gabungan yang diidentifikasi dalam IAM sebagai `Mary_Major` (`arn:aws:sts::123456789012:federated-user/Mary_Major`)

Opsi ini tidak akan mengenali sesi aktif dari seseorang yang memiliki peran **CodeCommitReview** dengan nama sesi peran `Mary_Major`

(arn:aws:sts::123456789012:assumed-role/CodeCommitReview/Mary_Major) kecuali Anda menyertakan wildcard (*Mary_Major). Anda juga dapat menentukan nama peran secara eksplisit (CodeCommitReview/Mary_Major).

- ARN yang sepenuhnya memenuhi syarat: Opsi ini memungkinkan Anda untuk menentukan Amazon Resource Name (ARN) yang memenuhi syarat dari pengguna atau peran IAM. Opsi ini juga mendukung peran yang dimiliki yang digunakan oleh layanan AWS, seperti AWS Lambda dan AWS CodeBuild. Untuk peran yang dimiliki, format ARN seharusnya arn:aws:sts::*AccountID*:assumed-role/*RoleName* untuk peran dan arn:aws:sts::*AccountID*:assumed-role/*FunctionName* untuk fungsi.

Jika Anda memilih nama pengguna atau peran yang dimiliki IAM sebagai tipe penyetuju, di Nilai, masukkan nama pengguna IAM atau peran atau ARN yang memenuhi syarat dari pengguna atau peran. Pilih Tambahkan lagi untuk menambahkan lebih banyak pengguna atau peran, hingga Anda menambahkan semua pengguna atau peran yang persetujuannya dihitung untuk jumlah persetujuan yang diperlukan.

Kedua jenis penyetuju memungkinkan Anda menggunakan wildcard (*) dalam nilainya. Misalnya, jika Anda memilih nama pengguna atau peran yang dimiliki IAM, dan Anda menentukan **CodeCommitReview/***, semua pengguna yang mengambil peran **CodeCommitReview** dihitung dalam grup persetujuan. Nama sesi peran masing-masing dihitung terhadap jumlah penyetuju yang diperlukan. Dengan cara ini, baik Mary_Major maupun Li_Juan dihitung sebagai persetujuan ketika masuk dan memiliki peran CodeCommitReview. Untuk informasi lebih lanjut tentang IAM ARN, wildcard, dan format, lihat [Pengidentifikasi IAM](#).

Note

Aturan persetujuan tidak mendukung persetujuan lintas akun.

8. Setelah selesai mengonfigurasi aturan persetujuan, pilih Kirimkan.

Buat aturan persetujuan untuk permintaan pull (AWS CLI)

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk membuat aturan persetujuan untuk permintaan pull dalam CodeCommit repositori

1. Jalankan perintah `create-pull-request-approval-rule`, dengan menentukan:

- ID dari permintaan pull (dengan pilihan `--id`).
- Nama aturan persetujuan (dengan pilihan `--approval-rule-name`).
- Isi aturan persetujuan (dengan pilihan `--approval-rule-content`).

Saat membuat aturan persetujuan, Anda dapat menentukan penyetuju di grup persetujuan dengan salah satu dari dua cara:

- `CodeCommitApprovers`: Opsi ini hanya memerlukan akun dan sumber daya Amazon Web Services. Ini dapat digunakan baik untuk pengguna IAM maupun pengguna akses gabungan yang namanya cocok dengan nama sumber daya yang disediakan. Ini adalah pilihan yang sangat kuat yang menawarkan fleksibilitas tinggi. Misalnya, jika Anda menentukan akun Amazon Web Services 123456789012 dan **Mary_Major**, semua hal berikut dihitung sebagai persetujuan yang berasal dari pengguna tersebut:

- Pengguna IAM di akun (`arn:aws:iam::123456789012:user/Mary_Major`)
- Seorang pengguna gabungan yang diidentifikasi dalam IAM sebagai `Mary_Major` (`arn:aws:sts::123456789012:federated-user/Mary_Major`)

Opsi ini tidak akan mengenali sesi aktif dari seseorang yang memiliki peran **CodeCommitReview** dengan nama sesi peran `Mary_Major` (`arn:aws:sts::123456789012:assumed-role/CodeCommitReview/Mary_Major`) kecuali Anda menyertakan wildcard (`*Mary_Major`).

- ARN yang sepenuhnya memenuhi syarat: Opsi ini memungkinkan Anda untuk menentukan Amazon Resource Name (ARN) yang memenuhi syarat dari pengguna atau peran IAM.

Untuk informasi lebih lanjut tentang IAM ARN, wildcard, dan format, lihat [Pengidentifikasi IAM](#).

Contoh berikut membuat aturan persetujuan bernama `Require two approved approvers` untuk permintaan pull dengan ID dari 27. Aturan menetapkan dua persetujuan diperlukan dari grup persetujuan. Kolam renang mencakup semua pengguna yang mengakses CodeCommit dan mengasumsikan peran **CodeCommitReview** di dalam 123456789012 Akun Amazon Web Services. Ini juga mencakup pengguna IAM atau pengguna gabungan bernama `Nikhil_Jayashankar` di akun Amazon Web Services yang sama:

```
aws codecommit create-pull-request-approval-rule --pull-request-id 27
--approval-rule-name "Require two approved approvers" --approval-
rule-content "{\"Version\": \"2018-11-08\", \"Statements\": [{\"Type\":
 \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers
\": [\"CodeCommitApprovers:123456789012:Nikhil_Jayashankar\",
 \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "approvalRule": {
    "approvalRuleName": "Require two approved approvers",
    "lastModifiedDate": 1570752871.932,
    "ruleContentSha256": "7c44e6ebEXAMPLE",
    "creationDate": 1570752871.932,
    "approvalRuleId": "aac33506-EXAMPLE",
    "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\":
 [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers
\": [\"CodeCommitApprovers:123456789012:Nikhil_Jayashankar\",
 \"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major"
  }
}
```

Tampilkan permintaan pull dalam repositori AWS CodeCommit

Anda dapat menggunakan konsol AWS CodeCommit atau AWS CLI untuk menampilkan permintaan pull untuk repositori anda. Secara default, Anda hanya melihat permintaan pull terbuka, tetapi Anda dapat mengubah filter untuk melihat semua permintaan pull, hanya permintaan tertutup, hanya menarik permintaan yang Anda buat, dan lainnya.

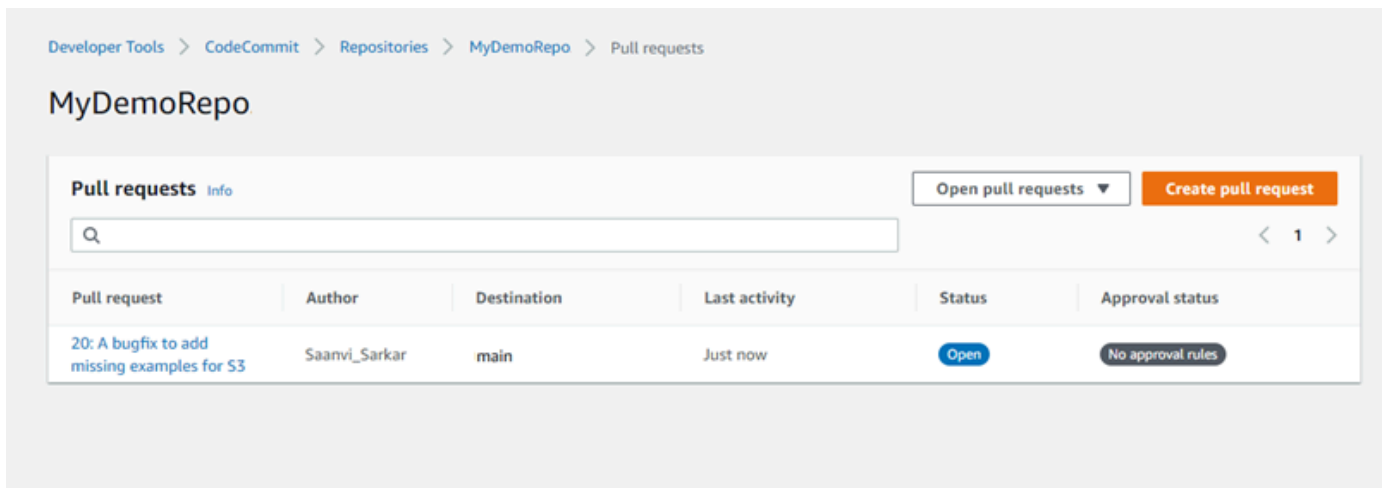
Topik

- [Tampilkan permintaan pull \(konsol\)](#)
- [Lihat permintaan pull \(AWS CLI\)](#)

Tampilkan permintaan pull (konsol)

Anda dapat menggunakan konsol AWS CodeCommit untuk menampilkan daftar permintaan pull dalam repositori CodeCommit. Dengan mengubah filter, Anda dapat mengubah tampilan daftar untuk hanya menampilkan serangkaian permintaan pull tertentu. Misalnya, Anda dapat melihat daftar permintaan pull yang Anda buat dengan status Terbuka, atau Anda dapat memilih filter lain dan melihat permintaan pull yang Anda buat dengan status Ditutup.

1. Buka konsol CodeCommit di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Dalam Repositori, pilih nama repositori tempat Anda ingin menampilkan permintaan pull.
3. Di panel navigasi, pilih Permintaan Pull.
4. Secara default, daftar semua permintaan pull terbuka ditampilkan.



5. Untuk mengubah filter tampilan, pilih dari daftar filter yang tersedia:
 - Buka permintaan pull(default): Menampilkan semua permintaan pull dengan statusBuka.
 - Semua permintaan pull: Menampilkan semua permintaan pull.
 - Permintaan pull ditutup: Menampilkan semua permintaan pull dengan statusTutup.
 - Permintaan pull saya: Menampilkan semua permintaan pull yang Anda buat, apa pun statusnya. Ini tidak menampilkan ulasan yang telah Anda komentari atau yang terdapat partisipasi Anda.
 - Permintaan pull terbuka saya: Menampilkan semua permintaan pull yang Anda buat dengan statusBuka.
 - Permintaan pull ditutup: Menampilkan semua permintaan pull yang Anda buat dengan statusTutup.

6. Jika Anda menemukan permintaan pull dalam daftar yang ditampilkan yang ingin Anda lihat, pilihlah permintaan pull tersebut.

Lihat permintaan pull (AWS CLI)

Untuk menggunakan perintah AWS CLI dengan CodeCommit, instal AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Ikuti langkah-langkah ini untuk menggunakan AWS CLI untuk menampilkan permintaan pull dalam repositori CodeCommit.

1. Untuk menampilkan daftar permintaan pull dalam repositori, jalankan perintah `list-pull-requests`, tentukan:
 - Nama repositori CodeCommit di mana Anda ingin menampilkan permintaan pull (dengan opsi `--repository-name`).
 - (Opsional) Status permintaan pull (dengan opsi `--pull-request-status`).
 - (Opsional) Amazon Resource Name (ARN) dari IAM pengguna yang menciptakan permintaan pull (dengan opsi `--author-arn`).
 - (Opsional) Sebuah token enumerasi yang dapat digunakan untuk mengembalikan batch berikutnya dari hasil (dengan opsi `--next-token`).
 - (Opsional) Batas jumlah hasil yang dikembalikan per permintaan (dengan opsi `--max-results`).

Misalnya, untuk daftar permintaan pull yang dibuat oleh pengguna IAM dengan ARN `arn:aws:iam::111111111111:user/Li_Juan` dan status *DITUTUP* dalam repositori CodeCommit bernama MyDemoRepo:

```
aws codecommit list-pull-requests --author-arn arn:aws:iam::111111111111:user/Li_Juan --pull-request-status CLOSED --repository-name MyDemoRepo
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "nextToken": "",
  "pullRequestIds": ["2", "12", "16", "22", "23", "35", "30", "39", "47"]
}
```

ID permintaan pull ditampilkan dalam urutan aktivitas terbaru.

2. Untuk melihat rincian permintaan pull, jalankan perintah `get-pull-request` dengan opsi `--pull-request-id`, menentukan ID dari permintaan pull. Misalnya, untuk melihat informasi tentang permintaan pull dengan ID **27**:

```
aws codecommit get-pull-request --pull-request-id 27
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\": [{\"Type\": \"Approver\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ],
    "lastActivityDate": 1562619583.565,
    "pullRequestTargets": [
      {
        "sourceCommit": "ca45e279EXAMPLE",
        "sourceReference": "refs/heads/bugfix-1234",
        "mergeBase": "a99f5ddbEXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
          "isMerged": false
        },
        "destinationCommit": "2abfc6beEXAMPLE",
        "repositoryName": "MyDemoRepo"
      }
    ],
    "revisionId": "e47def21EXAMPLE",
    "title": "Quick fix for bug 1234",
    "authorArn": "arn:aws:iam::123456789012:user/Nikhil_Jayashankar",
  }
}
```

```
"clientRequestToken": "d8d7612e-EXAMPLE",
"creationDate": 1562619583.565,
"pullRequestId": "27",
"pullRequestStatus": "OPEN"
}
}
```

3.

Untuk menampilkan persetujuan pada permintaan pull, jalankan perintah `get-pull-request-approval-state`, menentukan:

- ID dari permintaan pull (menggunakan opsi `--pull-request-id`).
- ID revisi dari permintaan pull (menggunakan opsi `--revision-id` option). Anda bisa mendapatkan ID revisi saat ini untuk permintaan pull dengan menggunakan perintah [get-pull-request](#).

Misalnya, untuk melihat persetujuan permintaan pull dengan ID `8` dan ID revisi `9f29d167EXAMPLE`:

```
aws codecommit get-pull-request-approval-state --pull-request-id 8 --revision-id 9f29d167EXAMPLE
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "approvals": [
    {
      "userArn": "arn:aws:iam::123456789012:user/Mary_Major",
      "approvalState": "APPROVE"
    }
  ]
}
```

4. Untuk menampilkan peristiwa dalam permintaan pull, jalankan perintah `describe-pull-request-events` dengan `--pull-request-id`, menentukan ID dari permintaan pull. Misalnya, untuk melihat peristiwa untuk permintaan pull dengan ID `8`:

```
aws codecommit describe-pull-request-events --pull-request-id 8
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:


```
{
  "pullRequestEvents": [
    {
      "pullRequestId": "8",
      "pullRequestEventType": "PULL_REQUEST_CREATED",
      "eventDate": 1510341779.53,
      "actor": "arn:aws:iam::111111111111:user/Zhang_Wei"
    },
    {
      "pullRequestStatusChangedEventMetadata": {
        "pullRequestStatus": "CLOSED"
      },
      "pullRequestId": "8",
      "pullRequestEventType": "PULL_REQUEST_STATUS_CHANGED",
      "eventDate": 1510341930.72,
      "actor": "arn:aws:iam::111111111111:user/Jane_Doe"
    }
  ]
}
```

5. Untuk menampilkan apakah ada konflik penggabungan untuk permintaan pull, jalankan perintah `get-merge-conflicts`, menentukan:
- Nama repositori CodeCommit (dengan opsi `--repository-name`).
 - Cabang, tanda, HEAD, atau referensi memenuhi syarat lainnya untuk sumber perubahan untuk digunakan dalam evaluasi penggabungan (dengan opsi `--source-commit-specifier`).
 - Cabang, tanda, HEAD, atau referensi memenuhi syarat lainnya untuk tujuan perubahan untuk digunakan dalam evaluasi penggabungan (dengan opsi `--destination-commit-specifier`).
 - Opsi penggabungan yang akan digunakan (dengan opsi `--merge-option`)

Misalnya, untuk menampilkan apakah ada konflik penggabungan antara ujung cabang sumber bernama *my-feature-branch* dan cabang tujuan bernama *utama* dalam repositori bernama MyDemoRepo:

```
aws codecommit get-merge-conflicts --repository-name MyDemoRepo --source-commit-specifier my-feature-branch --destination-commit-specifier main --merge-option FAST_FORWARD_MERGE
```

Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "destinationCommitId": "fac04518EXAMPLE",
  "mergeable": false,
  "sourceCommitId": "16d097f03EXAMPLE"
}
```

Meninjau permintaan pull

Anda dapat menggunakan AWS CodeCommit konsol untuk meninjau perubahan yang disertakan dalam permintaan tarik. Anda dapat menambahkan komentar untuk permintaan, file, dan baris kode individu. Anda juga dapat membalas komentar yang dibuat oleh pengguna lain. Jika repositori Anda adalah [dikonfigurasi dengan notifikasi](#), Anda menerima email ketika pengguna membalas komentar Anda atau ketika pengguna mengomentari permintaan pull.

Anda dapat menggunakan AWS CLI untuk mengomentari permintaan tarik dan membalas komentar. Untuk meninjau perubahan, Anda harus menggunakan CodeCommit konsol, git diff perintah, atau alat diff.

Topik

- [Meninjau permintaan pull \(konsol\)](#)
- [Tinjau permintaan pull \(AWS CLI\)](#)

Meninjau permintaan pull (konsol)

Anda dapat menggunakan CodeCommit konsol untuk meninjau permintaan tarik di CodeCommit repositori.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori.
3. Di panel navigasi, pilih Permintaan pull.
4. Secara default, daftar semua permintaan pull terbuka ditampilkan. Pilih permintaan pull terbuka yang ingin Anda tinjau.

Developer Tools > CodeCommit > Repositories > MyDemoRepo > Pull requests

MyDemoRepo

Pull requests Info Open pull requests ▾ Create pull request

< 1 >

Pull request	Author	Destination	Last activity	Status	Approval status
20: A bugfix to add missing examples for S3	Saarvi_Sarkar	main	Just now	Open	No approval rules

Note

Anda dapat mengomentari permintaan pull tertutup atau digabung, tetapi Anda tidak dapat menggabungkan atau membukanya kembali.

5. Dalam permintaan pull, pilih Perubahan.
6. Lakukan salah satu hal berikut ini:
 - Untuk menambahkan komentar umum untuk seluruh permintaan pull, di Komentar perubahan, di Komentar baru, masukkan komentar, dan kemudian pilih Simpan. Anda dapat menggunakan [Markdown](#), atau Anda dapat memasukkan komentar Anda di plaintext.

Comments on changes

New comment Preview markdown [Learn more](#)

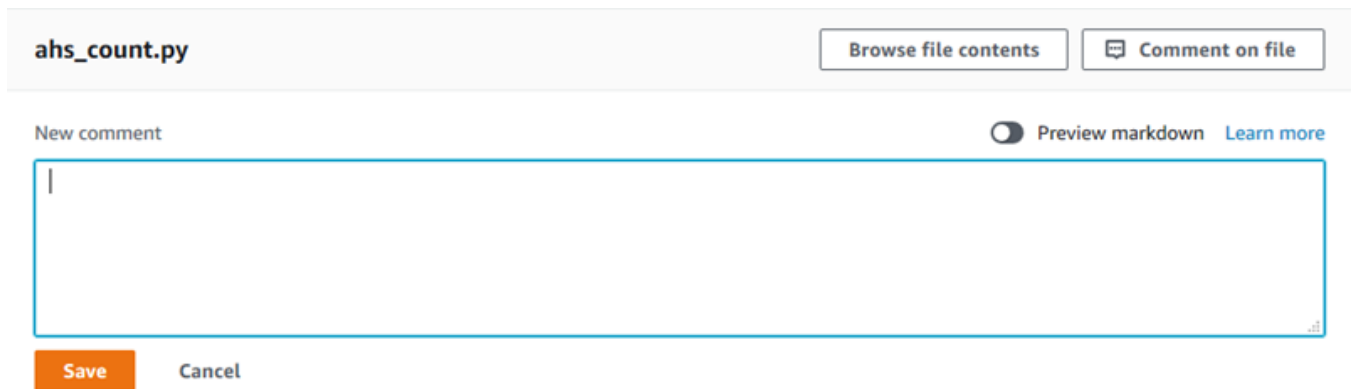
Did we also change the variable name in blf.py and concat.py?

Save

- Untuk menambahkan komentar ke file dalam commit, dalam Perubahan, cari nama file. Pilih ikon komentar



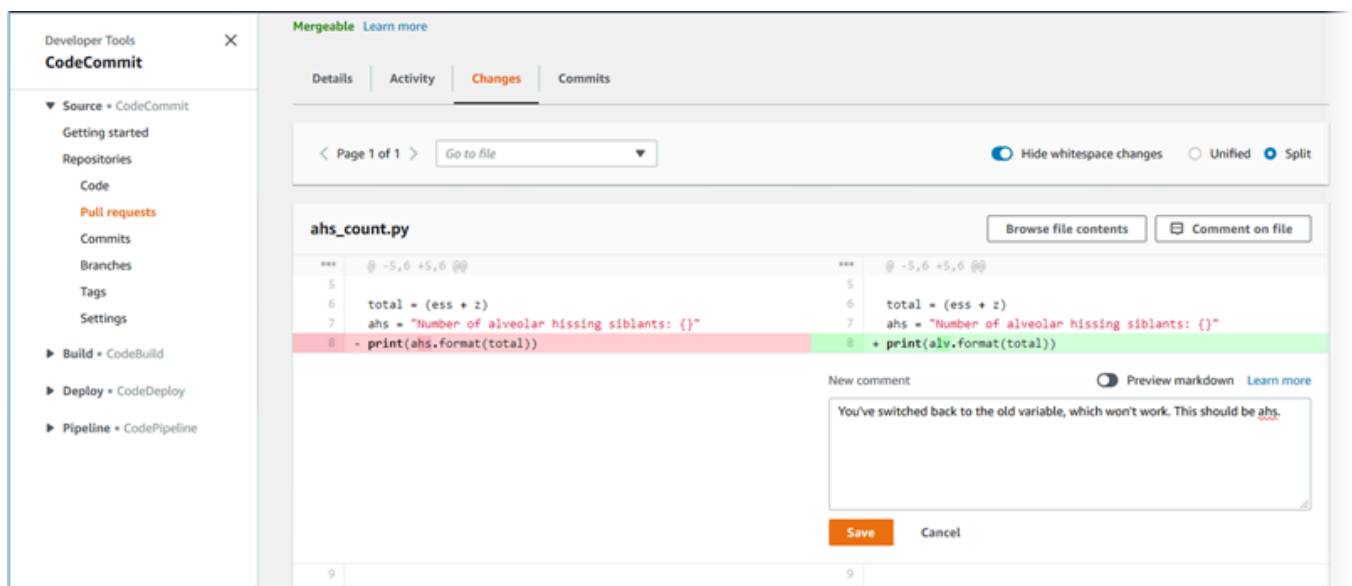
yang muncul di samping nama file, masukkan komentar, dan kemudian pilih Simpan.



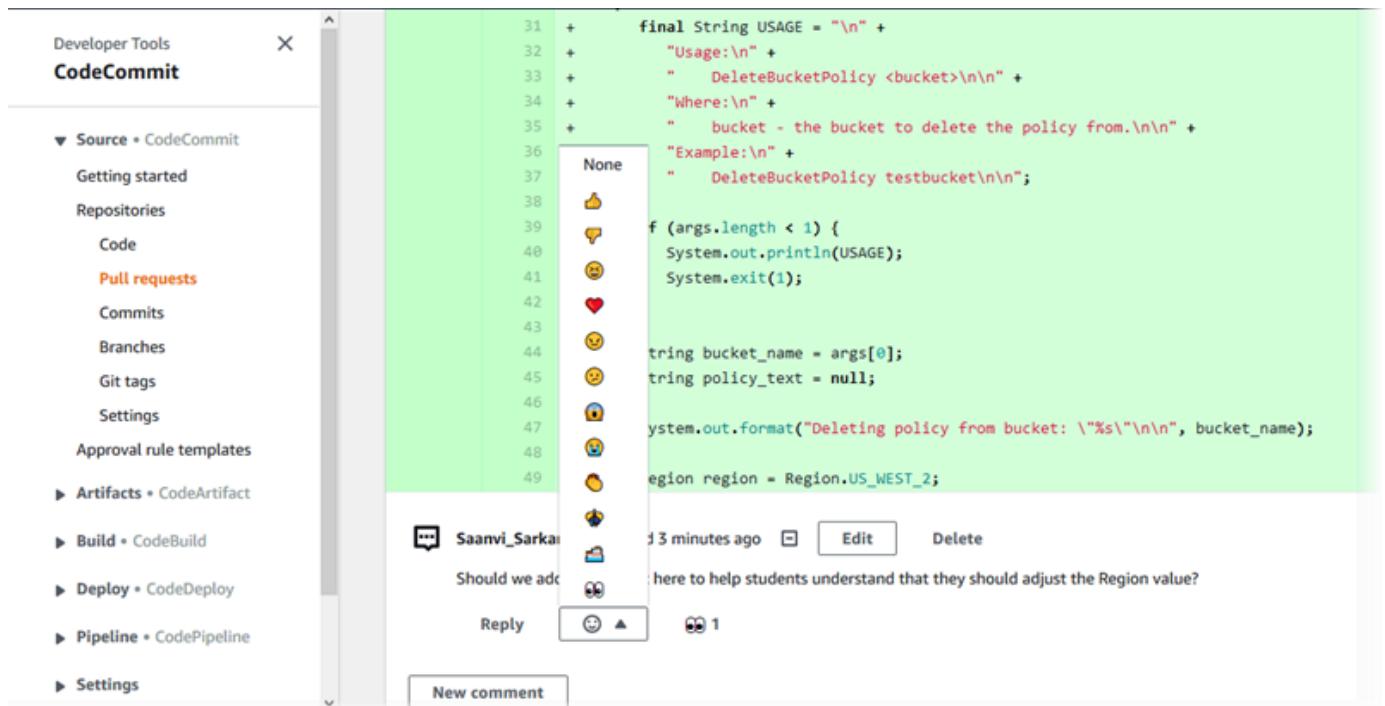
- Untuk menambahkan komentar ke baris yang diubah dalam permintaan pull, di Perubahan, pergi ke baris yang ingin Anda komentari. Pilih ikon komentar



yang muncul untuk baris tersebut, masukkan komentar, dan kemudian pilih Simpan.



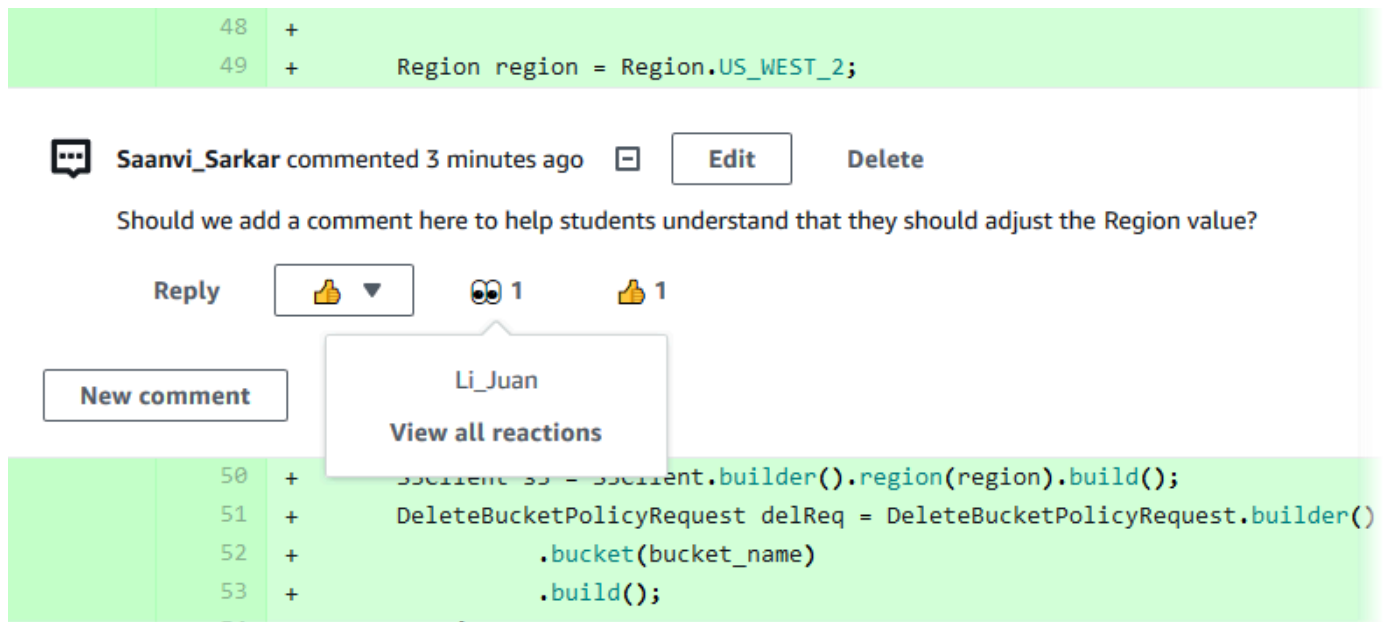
7. Untuk membalas komentar pada komit, di Perubahan atau Aktivitas, pilih Balas. Anda dapat membalas dengan teks dan dengan emoji.



Anda dapat melihat nama-nama mereka yang menanggapi dengan jawaban reaksi emoji tertentu dengan memilihnya. Untuk melihat semua reaksi emoji dan informasi tentang siapa yang menanggapi emoji, pilih Lihat semua reaksi. Jika Anda menanggapi dengan emoji pada komentar, respons Anda akan ditampilkan di ikon tombol reaksi emoji.

Note

Jumlah reaksi yang ditampilkan di konsol akurat sejak halaman dimuat. Untuk informasi terkini tentang jumlah reaksi emoji, segarkan halaman, atau pilih Lihat semua reaksi.



8. (Opsional) Untuk membalas rekomendasi yang dibuat oleh Amazon CodeGuru Reviewer, termasuk memberikan umpan balik tentang kualitas rekomendasi, pilih Balas. Gunakan tombol reaksi untuk memberikan informasi umum tentang apakah Anda menyetujui atau tidak menyetujui rekomendasi tersebut. Gunakan kolom komentar untuk memberikan rincian lebih lanjut tentang reaksi Anda.

i Note

Amazon CodeGuru Reviewer adalah layanan peninjauan kode otomatis yang menggunakan analisis program dan pembelajaran mesin untuk mendeteksi masalah umum dan merekomendasikan perbaikan dalam kode Java atau Python Anda.

- Anda hanya melihat komentar Amazon CodeGuru Reviewer jika Anda telah mengaitkan repositori dengan Amazon CodeGuru Reviewer, jika analisisnya selesai, dan jika kode dalam permintaan tarik adalah kode Java atau Python. Untuk informasi selengkapnya, lihat [Kaitkan atau pisahkan AWS CodeCommit repositori dengan Amazon Reviewer CodeGuru](#).
- Komentar Amazon CodeGuru Reviewer hanya muncul di tab Perubahan jika komentar dibuat pada revisi terbaru dari permintaan tarik. Mereka selalu muncul di tab Aktivitas.
- Meskipun Anda dapat merespons dengan reaksi emoji apa pun yang tersedia terhadap rekomendasi Amazon CodeGuru Reviewer, hanya reaksi emoji jempol

ke atas dan jempol ke bawah yang digunakan untuk mengevaluasi kegunaan rekomendasi tersebut.

The screenshot shows the AWS CodeCommit pull request interface for a pull request titled "25: Updated some of our Java samples". The interface includes navigation tabs for "Details", "Activity", "Changes", "Commits", and "Approvals". The "Activity" tab is selected, showing a comment from "Amazon CodeGuru Reviewer" on line 100 of "EventHandler.java". The comment text reads: "ObjectListing files = s3Client.listObjects(bucketName);". Below the code snippet, the reviewer provides a recommendation: "This code might not produce accurate results if the operation returns paginated results instead of all results. Consider adding another call to check for additional results. Leave feedback on this recommendation by selecting 'Reply'." The interface also shows a status of "In progress" and a "Merge" button.

9. Untuk menyetujui perubahan yang dibuat dalam permintaan pull, pilih Setujui.

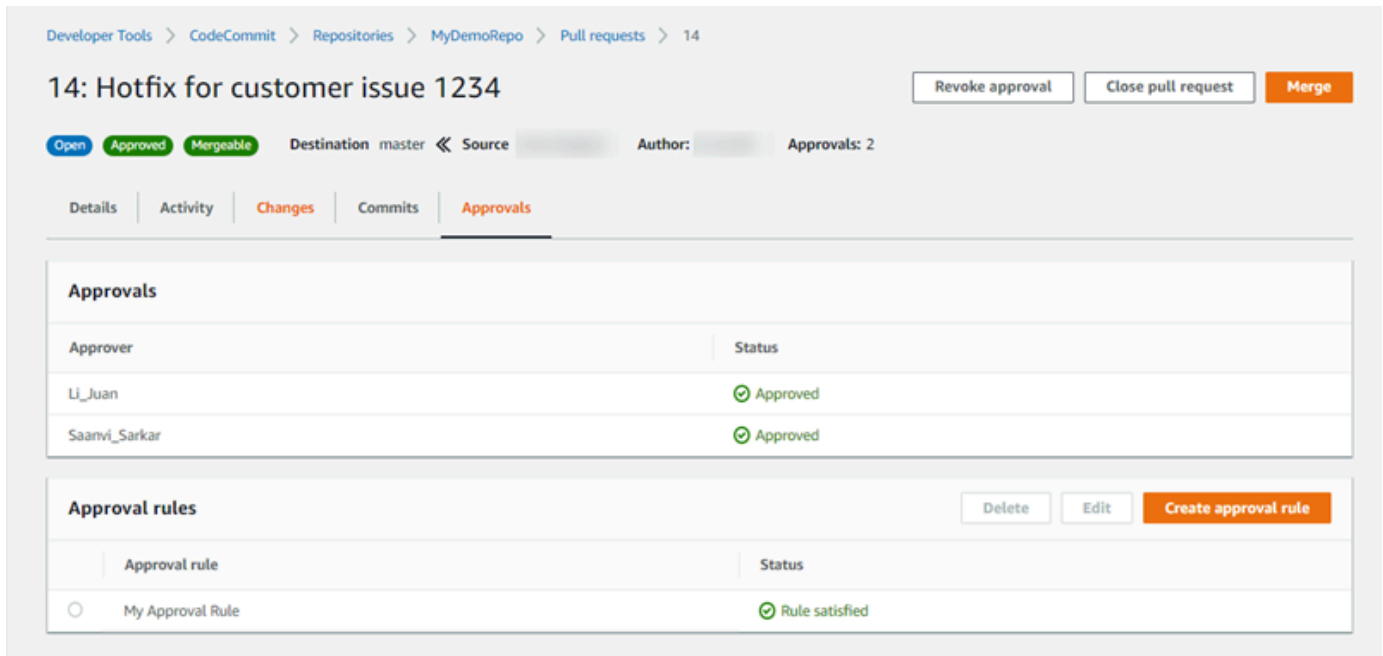
Note

Anda tidak dapat menyetujui permintaan pull yang Anda buat.

Anda dapat melihat persetujuan, aturan persetujuan untuk permintaan pull, dan persetujuan aturan yang dibuat oleh templat aturan persetujuan di Persetujuan. Jika Anda memutuskan Anda tidak ingin menyetujui permintaan pull setelah semua, Anda dapat memilih Cabut persetujuan.

Note

Anda hanya dapat menyetujui atau mencabut persetujuan atas permintaan pull terbuka. Anda tidak dapat menyetujui atau mencabut persetujuan permintaan pull yang statusnya digabung atau ditutup.



Developer Tools > CodeCommit > Repositories > MyDemoRepo > Pull requests > 14

14: Hotfix for customer issue 1234

Revoke approval Close pull request Merge

Open Approved Mergeable Destination master << Source Author: Approvals: 2

Details Activity Changes Commits Approvals

Approvals

Approver	Status
Li_Juan	Approved
Saarvi_Sarkar	Approved

Approval rules

Delete Edit Create approval rule

Approval rule	Status
<input type="radio"/> My Approval Rule	Rule satisfied

Tinjau permintaan pull (AWS CLI)

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal file AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Anda dapat meninjau permintaan tarik dengan AWS CLI perintah berikut:

- [post-comment-for-pull-request](#), untuk menambahkan komentar ke permintaan pull
- [get-comments-for-pull-request](#), untuk melihat komentar yang tersisa pada permintaan pull
- [update-pull-request-approval-state](#), untuk menyetujui atau mencabut persetujuan untuk permintaan pull
- [post-comment-reply](#), untuk membalas komentar dalam permintaan pull

Anda juga dapat menggunakan emoji dengan komentar dalam permintaan pull dengan perintah berikut:

- Untuk membalas komentar dengan emoji, jalankan [put-comment-reaction](#).
- Untuk menampilkan reaksi emoji pada komentar, jalankan [get-comment-reactions](#).

Untuk menggunakan AWS CLI untuk meninjau permintaan tarik dalam CodeCommit repositori

1. Untuk menambahkan komentar untuk permintaan pull dalam repositori, jalankan perintah `post-comment-for-pull-request`, menentukan:
 - ID dari permintaan pull (dengan opsi `--pull-request-id`).
 - Nama repositori yang berisi permintaan pull (dengan opsi `--repository-name`).
 - ID commit penuh dari commit di cabang tujuan dimana permintaan pull digabung (dengan opsi `--before-commit-id`).
 - ID commit penuh commit di cabang sumber yang ujung cabang saat ini untuk permintaan pull ketika Anda mengirim komentar (dengan opsi `--after-commit-id`).
 - Token idempotensi yang unik dan dihasilkan klien (dengan opsi `--client-request-token`).
 - Isi dari komentar Anda (dengan opsi `--content`).
 - Daftar informasi lokasi tentang tempat untuk menempatkan komentar, termasuk:
 - Nama file yang dibandingkan, termasuk ekstensi dan subdirektori, jika ada (dengan opsi `filePath`).
 - Nomor baris perubahan dalam file dibandingkan (dengan opsi `filePosition`).
 - Apakah komentar pada perubahan adalah "sebelum" atau "setelah" dalam perbandingan antara sumber dan cabang tujuan (dengan opsi `relativeFileVersion`).

Misalnya, gunakan perintah ini untuk menambahkan komentar *"Ini tampaknya tidak digunakan di mana saja. Bisakah kita menghapusnya?"* pada perubahan ke file *ahs_count.py* dalam permintaan tarik dengan ID *47* di repositori bernama *MyDemoRepo*

```
aws codecommit post-comment-for-pull-request --pull-request-id "47" --
repository-name MyDemoRepo --before-commit-id 317f8570EXAMPLE --after-
commit-id 5d036259EXAMPLE --client-request-token 123Example --content
"These don't appear to be used anywhere. Can we remove them?" --location
filePath=ahs_count.py,filePosition=367,relativeFileVersion=AFTER
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut.

```
{
  "afterBlobId": "1f330709EXAMPLE",
  "afterCommitId": "5d036259EXAMPLE",
  "beforeBlobId": "80906a4cEXAMPLE",
```

```

    "beforeCommitId": "317f8570EXAMPLE",
    "comment": {
      "authorArn": "arn:aws:iam::111111111111:user/Saanvi_Sarkar",
      "clientRequestToken": "123Example",
      "commentId": "abcd1234EXAMPLEb5678efgh",
      "content": "These don't appear to be used anywhere. Can we remove
them?",
      "creationDate": 1508369622.123,
      "deleted": false,
      "lastModifiedDate": 1508369622.123,
      "callerReactions": [],
      "reactionCounts": []
    }
    "location": {
      "filePath": "ahs_count.py",
      "filePosition": 367,
      "relativeFileVersion": "AFTER"
    },
    "repositoryName": "MyDemoRepo",
    "pullRequestId": "47"
  }

```

2. Untuk melihat komentar untuk permintaan pull, jalankan perintah `get-comments-for-pull-request`, menentukan:

- Nama CodeCommit repositori (dengan `--repository-name` opsi).
- ID yang dihasilkan sistem dari permintaan pull (dengan opsi `--pull-request-id`).
- (Opsional) Sebuah enumerasi token untuk mengembalikan batch berikutnya dari hasil (dengan opsi `--next-token`).
- (Opsional) Sebuah bilangan bulat non-negatif untuk membatasi jumlah hasil yang dikembalikan (dengan opsi `--max-results`).

Misalnya, gunakan perintah ini untuk melihat komentar untuk permintaan pull dengan ID 42.

```
aws codecommit get-comments-for-pull-request --pull-request-id 42
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut.

```
{
  "commentsForPullRequestData": [
```

```

{
  "afterBlobId": "1f330709EXAMPLE",
  "afterCommitId": "5d036259EXAMPLE",
  "beforeBlobId": "80906a4cEXAMPLE",
  "beforeCommitId": "317f8570EXAMPLE",
  "comments": [
    {
      "authorArn": "arn:aws:iam::111111111111:user/Saanvi_Sarkar",
      "clientRequestToken": "",
      "commentId": "abcd1234EXAMPLEb5678efgh",
      "content": "These don't appear to be used anywhere. Can we remove
them?",
      "creationDate": 1508369622.123,
      "deleted": false,
      "lastModifiedDate": 1508369622.123,
      "callerReactions": [],
      "reactionCounts":
      {
        "THUMBSUP" : 6,
        "CONFUSED" : 1
      }
    },
    {
      "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
      "clientRequestToken": "",
      "commentId": "442b498bEXAMPLE5756813",
      "content": "Good catch. I'll remove them.",
      "creationDate": 1508369829.104,
      "deleted": false,
      "lastModifiedDate": 150836912.273,
      "callerReactions": ["THUMBSUP"]
      "reactionCounts":
      {
        "THUMBSUP" : 14
      }
    }
  ],
  "location": {
    "filePath": "ahs_count.py",
    "filePosition": 367,
    "relativeFileVersion": "AFTER"
  },
  "repositoryName": "MyDemoRepo",
  "pullRequestId": "42"
}

```

```
    }  
  ],  
  "nextToken": "exampleToken"  
}
```

3.

Untuk menyetujui atau mencabut persetujuan untuk permintaan pull, jalankan perintah `update-pull-request-approval-state`, menentukan:

- ID dari permintaan pull (menggunakan `--pull-request-id`).
- ID revisi dari permintaan pull (menggunakan opsi `--revision-id` option). Anda bisa mendapatkan ID revisi saat ini untuk permintaan tarik dengan menggunakan [get-pull-request](#) perintah.
- Status persetujuan yang ingin Anda terapkan pilihan (menggunakan `--approval-state`). Status persetujuan yang valid mencakup APPROVE dan REVOKE.

Misalnya, gunakan perintah ini untuk menyetujui permintaan pull dengan ID `27` dan ID revisi `9f29d167EXAMPLE`.

```
aws codecommit update-pull-request-approval-state --pull-request-id 27 --revision-id 9f29d167EXAMPLE --approval-state "APPROVE"
```

Jika berhasil, perintah ini tidak mengembalikan apa pun.

4. Untuk mengirim balasan ke komentar dalam permintaan pull, jalankan perintah `post-comment-reply`, menentukan:

- ID yang dihasilkan sistem dari komentar yang ingin Anda balas (dengan opsi `--in-reply-to`).
- Sebuah token idempotensi yang unik dan dihasilkan klien (dengan opsi `--client-request-token`).
- Isi balasan Anda (dengan opsi `--content`).

Misalnya, gunakan perintah ini untuk menambahkan balasan *"Tangkapan bagus. Aku akan menghapusnya."* untuk komentar dengan ID yang dihasilkan sistem `abcd1234EXAMPLEb5678efgh`.

```
aws codecommit post-comment-reply --in-reply-to abcd1234EXAMPLEb5678efgh --content "Good catch. I'll remove them." --client-request-token 123Example
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut.

```
{
  "comment": {
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
    "clientRequestToken": "123Example",
    "commentId": "442b498bEXAMPLE5756813",
    "content": "Good catch. I'll remove them.",
    "creationDate": 1508369829.136,
    "deleted": false,
    "lastModifiedDate": 150836912.221,
    "callerReactions": [],
    "reactionCounts": []
  }
}
```

Memperbarui sebuah permintaan pull

Anda dapat memperbarui sebuah permintaan pull dengan perubahan kode lebih lanjut dengan mendorong commit ke cabang sumber permintaan pull yang terbuka. Untuk informasi selengkapnya, lihat [Buat komit di AWS CodeCommit](#).

Anda dapat menggunakan konsol AWS CodeCommit atau AWS CLI untuk memperbarui judul atau deskripsi sebuah permintaan pull. Anda mungkin ingin memperbarui judul atau deskripsi sebuah permintaan pull karena:


- Pengguna lain tidak memahami deskripsi, atau judul aslinya tidak begitu tepat.
- Anda ingin judul atau deskripsi agar mencerminkan perubahan yang dibuat ke cabang sumber dari permintaan pull terbuka.

Memperbarui sebuah permintaan pull (konsol)

Anda dapat menggunakan konsol CodeCommit untuk memperbarui judul dan deskripsi dari sebuah permintaan pull dalam repositori CodeCommit. Untuk memperbarui kode dalam permintaan pull, Anda harus melakukan push commit ke cabang sumber dari sebuah permintaan pull terbuka.

1. Buka konsol CodeCommit di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin memperbarui sebuah permintaan pull.
3. Di panel navigasi, pilih Permintaan pull.

4. Secara default, daftar semua permintaan pull terbuka ditampilkan. Pilih permintaan pull terbuka yang ingin Anda perbarui.
5. Dalam permintaan pull, pilih Detail, lalu pilih Edit detail untuk mengedit judul atau deskripsi.

 Note

Anda tidak dapat memperbarui judul atau deskripsi permintaan pull yang sudah tertutup atau digabung.

Memperbarui permintaan pull (AWS CLI)

Untuk menggunakan perintah AWS CLI dengan CodeCommit, instal AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Anda mungkin juga tertarik dengan perintah-perintah berikut:

- [update-pull-request-approval-state](#), untuk menyetujui atau mencabut persetujuan pada sebuah permintaan pull.
- [create-pull-request-approval-rule](#), untuk membuat aturan persetujuan untuk sebuah permintaan pull.
- [delete-pull-request-approval-rule](#), untuk menghapus aturan persetujuan untuk sebuah permintaan pull.
- [Buat komit menggunakan AWS CLI](#) atau [Buatlah sebuah commit menggunakan klien Git](#), untuk membuat dan mendorong perubahan kode tambahan ke cabang sumber permintaan pull terbuka.

Untuk menggunakan AWS CLI untuk memperbarui pull request dalam sebuah repositori CodeCommit

1. Untuk memperbarui judul permintaan pull dalam sebuah repositori, jalankan perintah `update-pull-request-title`, dengan menentukan:
 - ID dari permintaan pull (dengan pilihan `--pull-request-id`).
 - Judul permintaan pull (dengan pilihan `--title`).

Misalnya, untuk memperbarui judul dari sebuah permintaan pull dengan ID **47**:

```
aws codecommit update-pull-request-title --pull-request-id 47 --title
"Consolidation of global variables - updated review"
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"],\"Statements\": [{\"Type
\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":
[\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "originApprovalRuleTemplate": {
          "approvalRuleTemplateId": "dd8b26gr-EXAMPLE",
          "approvalRuleTemplateName": "2-approver-rule-for-main"
        },
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ],
    "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
    "clientRequestToken": "",
    "creationDate": 1508530823.12,
    "description": "Review the latest changes and updates to the global
variables. I have updated this request with some changes, including removing some
unused variables.",
    "lastActivityDate": 1508372657.188,
    "pullRequestId": "47",
    "pullRequestStatus": "OPEN",
    "pullRequestTargets": [
      {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
          "isMerged": false,
        },
        "repositoryName": "MyDemoRepo",
      }
    ]
  }
}
```

```

        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
    }
  ],
  "title": "Consolidation of global variables - updated review"
}
}

```

2. Untuk memperbarui deskripsi dari sebuah permintaan pull, jalankan perintah `update-pull-request-description`, dengan menentukan:

- ID dari permintaan pull (dengan pilihan `--pull-request-id`).
- Deskripsi (dengan pilihan `--description`).

Misalnya, untuk memperbarui deskripsi dari sebuah permintaan pull dengan ID **47**:

```
aws codecommit update-pull-request-description --pull-request-id 47 --description
"Updated the pull request to remove unused global variable."
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```

{
  "pullRequest": {
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
    "clientRequestToken": "",
    "creationDate": 1508530823.155,
    "description": "Updated the pull request to remove unused global variable.",
    "lastActivityDate": 1508372423.204,
    "pullRequestId": "47",
    "pullRequestStatus": "OPEN",
    "pullRequestTargets": [
      {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
          "isMerged": false,
        },
      },
      {
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
      }
    ]
  }
}

```



```
    ],  
    "title": "Consolidation of global variables"  
  }  
}
```

Mengedit atau menghapus aturan persetujuan untuk permintaan pull

Ketika Anda memiliki sebuah aturan persetujuan pada permintaan pull, Anda tidak dapat menggabungkan permintaan pull tersebut sampai syarat-syaratnya telah terpenuhi. Anda dapat mengubah aturan persetujuan untuk permintaan pull untuk membuatnya menjadi lebih mudah untuk memenuhi syarat-syaratnya, atau untuk meningkatkan ketelitian tinjauan. Anda dapat mengubah jumlah pengguna yang harus menyetujui sebuah permintaan pull. Anda juga dapat menambahkan, menghapus, atau mengubah keanggotaan dalam kolom persetujuan pengguna untuk aturan tersebut. Yang terakhir, jika Anda tidak lagi ingin menggunakan aturan persetujuan untuk sebuah permintaan pull, Anda dapat menghapusnya.

Note

Anda juga dapat mengganti aturan persetujuan untuk sebuah permintaan pull. Untuk informasi selengkapnya, lihat [Menimpa aturan persetujuan pada sebuah permintaan pull](#).

Anda dapat menggunakan konsol AWS CodeCommit atau AWS CLI untuk mengedit atau menghapus aturan persetujuan untuk repositori Anda.

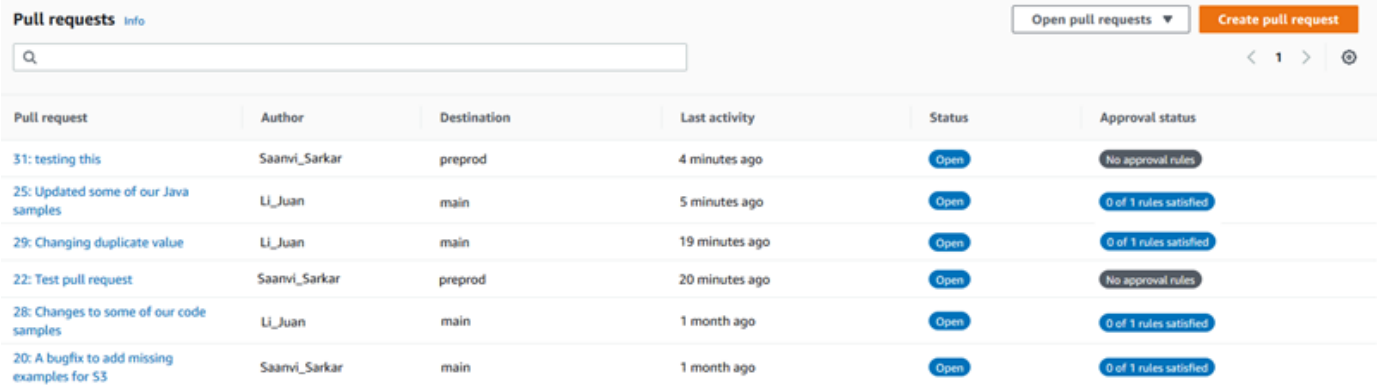
Topik

- [Mengedit atau menghapus aturan persetujuan untuk permintaan pull \(konsol\)](#)
- [Mengedit atau menghapus aturan persetujuan untuk permintaan pull \(AWS CLI\)](#)

Mengedit atau menghapus aturan persetujuan untuk permintaan pull (konsol)

Anda dapat menggunakan konsol CodeCommit untuk mengedit atau menghapus aturan persetujuan untuk sebuah permintaan pull dalam repositori CodeCommit.

1. Buka konsol CodeCommit di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Dalam Repositori, pilih nama repositori tempat Anda ingin mengedit atau menghapus aturan persetujuan untuk permintaan pull.
3. Di panel navigasi, pilih Permintaan Pull.
4. Pilih permintaan pull tempat Anda ingin mengedit atau menghapus aturan persetujuan. Anda hanya dapat mengedit dan menghapus aturan persetujuan untuk permintaan pull yang terbuka.



Pull request	Author	Destination	Last activity	Status	Approval status
31: testing this	Saanvi_Sarkar	preprod	4 minutes ago	Open	No approval rules
25: Updated some of our Java samples	Li_Juan	main	5 minutes ago	Open	0 of 1 rules satisfied
29: Changing duplicate value	Li_Juan	main	19 minutes ago	Open	0 of 1 rules satisfied
22: Test pull request	Saanvi_Sarkar	preprod	20 minutes ago	Open	No approval rules
28: Changes to some of our code samples	Li_Juan	main	1 month ago	Open	0 of 1 rules satisfied
20: A bugfix to add missing examples for S3	Saanvi_Sarkar	main	1 month ago	Open	0 of 1 rules satisfied

5. Dalam permintaan pull, pilih Persetujuan, dan kemudian pilih aturan yang ingin Anda edit atau hapus dari daftar. Lakukan salah satu dari berikut:
 - Jika Anda ingin mengedit aturan, pilih Edit.
 - Jika Anda ingin menghapus aturan, pilih Hapus, lalu ikuti petunjuk untuk memverifikasi penghapusan aturan.
6. Di Edit aturan persetujuan, buat perubahan yang Anda inginkan pada aturan tersebut, lalu pilih Kirim.

Edit approval rule

Rule details

Rule name

My Approval Rule

Number of approvals needed

Approval pool members - *optional*

If approval pool members are specified, only approvals from these members will count toward satisfying this rule. Use a wildcard to match multiple approvers with one value.

Approver type [Info](#)

Value

CodeCommit ▼

Mary_Major

Remove

CodeCommit ▼

Li_Juan

Remove

CodeCommit ▼

Saanvi_Sarkar

Remove

Fully qualified ARN ▼

arn:aws:iam:::user/

Remove

Add

Cancel

Submit

7. Setelah selesai mengonfigurasi aturan persetujuan, pilih Kirim.

Mengedit atau menghapus aturan persetujuan untuk permintaan pull (AWS CLI)

Untuk menggunakan perintah AWS CLI dengan CodeCommit, instal AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Anda dapat menggunakan AWS CLI untuk mengedit isi aturan persetujuan dan menghapus sebuah aturan persetujuan.

Note

Anda mungkin juga tertarik dengan perintah-perintah berikut:

- [update-pull-request-approval-state](#), untuk menyetujui atau mencabut persetujuan pada sebuah permintaan pull.
- [get-pull-request-approval-states](#), untuk melihat persetujuan pada permintaan pull.
- [evaluate-pull-request-approval-rules](#), untuk menentukan apakah aturan persetujuan untuk permintaan pull syarat-syaratnya telah terpenuhi.

Untuk menggunakan AWS CLI untuk mengedit atau menghapus aturan persetujuan untuk sebuah permintaan pull dalam repositori CodeCommit

1. Untuk mengedit aturan persetujuan, jalankan perintah `update-pull-request-approval-rule-content`, dengan menentukan:

- ID dari permintaan pull (dengan pilihan `--id`).
- Nama aturan persetujuan (dengan pilihan `--approval-rule-name`).
- Isi aturan persetujuan (dengan pilihan `--approval-rule-content`).

Contoh ini memperbarui aturan persetujuan bernama *Mengharuskan dua penyetuju yang disetujui* untuk sebuah permintaan pull dengan ID `27`. Aturan memerlukan satu persetujuan pengguna dari kolam persetujuan yang mencakup setiap pengguna IAM di akun `123456789012` Amazon Web Services:

```
aws codecommit update-pull-request-approval-rule-content --pull-request-id 27
--approval-rule-name "Require two approved approvers" --approval-rule-content
"{Version: 2018-11-08, Statements: [{Type: \"Approvers\", NumberOfApprovalsNeeded:
1, ApprovalPoolMembers:[\"CodeCommitApprovers:123456789012:user/*\"]}]}"
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "approvalRule": {
    "approvalRuleContent": "{Version: 2018-11-08, Statements:
  [{Type: \"Approvers\", NumberOfApprovalsNeeded: 1, ApprovalPoolMembers:
  [\"CodeCommitApprovers:123456789012:user/*\"]}]}"
```

```
"approvalRuleId": "aac33506-EXAMPLE",
"originApprovalRuleTemplate": {},
"creationDate": 1570752871.932,
"lastModifiedDate": 1570754058.333,
"approvalRuleName": Require two approved approvers",
"lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
"ruleContentSha256": "cd93921cEXAMPLE",
}
}
```

3.

Untuk menghapus aturan persetujuan, jalankan perintah `delete-pull-request-approval-rule`, dengan menentukan:

- ID dari permintaan pull (dengan pilihan `--id`).
- Nama aturan persetujuan (dengan pilihan `--approval-rule-name`).

Misalnya, untuk menghapus sebuah aturan persetujuan dengan nama *Aturan Persetujuan Saya* untuk sebuah permintaan pull dengan ID *15*:

```
aws codecommit delete-pull-request-approval-rule --pull-request-id 15 --approval-
rule-name "My Approval Rule"
```

Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "approvalRuleId": "077d8e8a8-EXAMPLE"
}
```

Menimpa aturan persetujuan pada sebuah permintaan pull

Dalam pengembangan normal, Anda ingin pengguna untuk memenuhi syarat aturan persetujuan sebelum Anda menggabungkan permintaan pull. Namun, mungkin ada kalanya Anda perlu mempercepat penggabungan sebuah permintaan pull. Misalnya, Anda mungkin ingin menempatkan perbaikan bug ke dalam produksi, tetapi tidak ada seorang pun di kolam persetujuan yang tersedia untuk menyetujui permintaan pull. Dalam kasus seperti ini, Anda dapat memilih untuk menimpa aturan persetujuan pada sebuah permintaan pull. Anda dapat mengganti semua aturan persetujuan untuk sebuah permintaan pull, termasuk yang dibuat khusus untuk permintaan pull tersebut dan

dihasilkan dari templat aturan persetujuan. Anda tidak dapat secara selektif menimpa aturan persetujuan tertentu, harus semua aturan. Setelah Anda telah menimpa persyaratan aturan persetujuan dengan menimpa aturan, Anda dapat menggabungkan permintaan pull ke cabang tujuan.

Ketika Anda mengganti aturan persetujuan pada sebuah permintaan pull, informasi tentang pengguna yang menimpa aturan akan dicatat dalam aktivitas untuk permintaan pull tersebut. Dengan cara ini Anda dapat kembali ke dalam riwayat permintaan pull dan meninjau siapa yang menimpa aturan. Anda juga dapat memilih untuk mencabut penimpaan jika permintaan pull masih terbuka. Setelah permintaan pull telah digabung, Anda tidak dapat lagi mencabut penimpaan tersebut.

Topik

- [Menimpa aturan persetujuan \(konsol\)](#)
- [Menimpa aturan persetujuan \(AWS CLI\)](#)

Menimpa aturan persetujuan (konsol)

Anda dapat menimpa persyaratan aturan persetujuan pada sebuah permintaan pull di konsol, sebagai bagian dari peninjauan permintaan pull. Jika Anda berubah pikiran, Anda dapat mencabut penimpaan Anda, dan persyaratan aturan persetujuan tersebut kemudian diterapkan kembali. Anda hanya dapat menimpa aturan persetujuan atau mencabut penimpaan jika permintaan pull masih terbuka. Jika ia sudah digabung atau ditutup, Anda tidak dapat mengubah status penimpaan-nya.

1. Buka konsol CodeCommit di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori.
3. Di panel navigasi, pilih Permintaan pull. Pilih permintaan pull dimana anda ingin menimpa persyaratan aturan persetujuan, atau mencabut penimpaan.
4. Pada tab Persetujuan, pilih Menimpa aturan persetujuan. Persyaratan kemudian disisihkan, dan teks tombol berubah menjadi Cabut penimpaan. Untuk menerapkan kembali persyaratan aturan persetujuan, pilih Cabut penimpaan.

Menimpa aturan persetujuan (AWS CLI)

Anda dapat menggunakan AWS CLI untuk menimpa persyaratan aturan persetujuan. Anda juga dapat menggunakannya untuk melihat status penimpaan untuk sebuah permintaan pull.

Untuk menimpa persyaratan aturan persetujuan pada sebuah permintaan pull

1. Pada sebuah terminal atau baris perintah, jalankan perintah `override-pull-request-approval-rules`, dengan menentukan:
 - ID permintaan pull yang dihasilkan sistem.
 - ID permintaan pull revisi terbaru. Untuk melihat informasi ini, gunakan `get-pull-request`.
 - Status yang Anda inginkan untuk penimpaan, `OVERRIDE` atau `REVOKE`. Status `REVOKE` menghapus status `OVERRIDE` tetapi tidak disimpan.

Sebagai contoh, untuk menimpa aturan persetujuan pada permintaan pull dengan ID **34** dan ID revisi **927df8d8EXAMPLE**:

```
aws codecommit override-pull-request-approval-rules --pull-request-id 34 --  
revision-id 927df8d8EXAMPLE --override-status OVERRIDE
```

2. Jika berhasil, perintah ini tidak mengembalikan apa pun.
3. Untuk mencabut penimpaan pada permintaan pull dengan ID **34** dan ID revisi **927df8d8EXAMPLE**:

```
aws codecommit override-pull-request-approval-rules --pull-request-id 34 --  
revision-id 927df8d8EXAMPLE --override-status REVOKE
```

Untuk mendapatkan informasi tentang status penimpaan dari sebuah permintaan pull

1. Pada sebuah terminal atau baris perintah, jalankan perintah `get-pull-request-override-state`, dengan menentukan:
 - ID permintaan pull yang dihasilkan sistem.
 - ID permintaan pull revisi terbaru. Untuk melihat informasi ini, gunakan `get-pull-request`.

Misalnya, untuk melihat status penimpaan untuk sebuah permintaan pull dengan ID **34** dan ID revisi **927df8d8EXAMPLE**:

```
aws codecommit get-pull-request-override-state --pull-request-id 34 --revision-  
id 927df8d8EXAMPLE
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "overridden": true,
  "overrider": "arn:aws:iam::123456789012:user/Mary_Major"
}
```

Menggabungkan permintaan pull dalam repositori AWS CodeCommit

Setelah kode Anda ditinjau dan semua aturan persetujuan (jika ada) pada permintaan pull telah terpenuhi, Anda dapat menggabungkan permintaan pull dengan salah satu dari beberapa cara:

- Anda dapat menggunakan konsol untuk menggabungkan cabang sumber Anda ke cabang tujuan menggunakan salah satu strategi penggabungan yang tersedia, yang juga menutup permintaan pull. Anda juga dapat menyelesaikan konflik penggabungan di konsol tersebut. Konsol menampilkan pesan yang menunjukkan jika permintaan pull dapat digabungkan atau jika konflik harus diselesaikan. Ketika semua konflik diselesaikan dan Anda memilih Gabungkan, penggabungan dilakukan menggunakan strategi penggabungan yang Anda pilih. Fast-forward adalah strategi merge default, yang merupakan pilihan default untuk Git. Tergantung pada status kode di cabang sumber dan tujuan, strategi tersebut mungkin tidak tersedia, tetapi pilihan lain mungkin tersedia, seperti squash atau 3-way.
- Anda dapat menggunakan AWS CLI untuk menggabungkan dan menutup permintaan pull menggunakan strategi penggabungan fast-forward, squash, atau 3-way.
- Pada komputer lokal Anda, Anda dapat menggunakan perintah git merge untuk menggabungkan cabang sumber ke cabang tujuan, dan kemudian mendorong kode gabungan ke cabang tujuan. Pendekatan ini memiliki kekurangan yang harus Anda pertimbangkan dengan hati-hati. Ini menggabungkan permintaan pull terlepas jika persyaratan untuk aturan persetujuan pada permintaan pull telah dipenuhi atau tidak, yang menghindari kontrol tersebut. Penggabungan dan pengiriman ke cabang tujuan juga menutup permintaan pull secara otomatis jika permintaan pull digabung menggunakan strategi penggabungan fast-forward. Salah satu keuntungan dari pendekatan ini adalah bahwa perintah git merge memungkinkan Anda untuk memilih opsi penggabungan atau strategi yang tidak tersedia di konsol CodeCommit. Untuk informasi lebih lanjut tentang git merge dan opsi penggabungan, lihat [git-merge](#) atau dokumentasi Git Anda.

CodeCommit menutup permintaan pull secara otomatis jika salah sumber atau cabang tujuan permintaan pull dihapus.

Topik

- [Menggabungkan permintaan pull \(konsol\)](#)
- [Menggabungkan permintaan pull \(AWS CLI\)](#)

Menggabungkan permintaan pull (konsol)

Anda dapat menggunakan konsol CodeCommit untuk menggabungkan permintaan pull dalam repositori CodeCommit. Setelah status permintaan pull diubah menjadi Digabungkan, maka tidak akan lagi muncul di daftar permintaan pull terbuka. Permintaan pull gabungan dikategorikan sebagai tertutup. Hal ini tidak dapat diubah kembali ke Buka, namun pengguna masih dapat mengomentari perubahan dan membalas komentar. Setelah permintaan pull digabung atau ditutup, Anda tidak dapat menyetujuinya, mencabut persetujuan untuk itu, atau mengesampingkan aturan persetujuan yang diterapkan pada permintaan pull.

1. Buka konsol CodeCommit di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori.
3. Di panel navigasi, pilih Permintaan pull.
4. Secara default, daftar semua permintaan pull terbuka ditampilkan. Pilih permintaan pull terbuka yang ingin Anda gabungkan.
5. Dalam permintaan pull, pilih Persetujuan. Tinjau daftar pemberi persetujuan, dan verifikasi bahwa semua aturan persetujuan (jika ada) telah memenuhi syarat. Anda tidak dapat menggabungkan permintaan pull jika satu atau lebih persetujuan aturan memiliki status Aturan tidak terpenuhi. Jika tidak ada yang menyetujui permintaan pull, pertimbangkan apakah Anda ingin menggabungkannya, atau apakah Anda ingin menunggu persetujuan.

Note

Jika aturan persetujuan dibuat untuk permintaan pull, Anda dapat mengeditnya atau menghapusnya untuk membuka blokir penggabungan. Jika aturan persetujuan dibuat dengan templat aturan persetujuan, Anda tidak dapat mengedit atau menghapusnya. Anda hanya dapat memilih untuk mengesampingkan persyaratan. Untuk informasi selengkapnya, lihat [Menimpa aturan persetujuan pada sebuah permintaan pull](#).

The screenshot displays the AWS CodeCommit pull request interface. At the top, the breadcrumb navigation shows: Developer Tools > CodeCommit > Repositories > MyDemoRepo > Pull requests > 15. The pull request title is "15: Quick fix to one of the code samples to include onomatopoeia". On the right, there are buttons for "Close pull request" and "Merge". Below the title, the status is "Open", "0 of 1 rules satisfied", and "Mergeable". The destination is "main", the source is "bugfix-codesample", the author is "Saamvi_Sarkar", and there are "Approvals: 0". The "Approvals" tab is selected, showing a table with columns "Approver" and "Status". The table is empty, with a message: "No results. There are no results to display." Below the table, the "Approval rules" section shows a table with columns "Approval rule" and "Status". There is one rule: "Require two approvals before merge" with a status of "Rule not satisfied". Buttons for "Delete", "Edit", and "Create approval rule" are visible.

6. Pilih Gabungkan.
7. Dalam permintaan pull, pilih antara strategi penggabungan yang tersedia. Menggabungkan strategi yang tidak dapat diterapkan muncul berwarna abu-abu. Jika tidak ada strategi penggabungan tersedia, Anda dapat memilih untuk secara manual menyelesaikan konflik di konsol CodeCommit, atau Anda dapat mengatasinya secara lokal menggunakan klien Git Anda. Untuk informasi selengkapnya, lihat [Menyelesaikan konflik dalam permintaan pull dalam repositori AWS CodeCommit](#).

Merge pull request 9: Bug fix for unhandled exception


Merge request details

Pull request: #9 Bug fix for unhandled exception


Destination main **Source** bugfix-bug1234

Merge strategy [Info](#)
Determines the way in which the current pull request will be merged into the destination branch


Fast forward merge
`git merge --ff-only`
Merges the branches and moves the destination branch pointer to the tip of the source branch. This is the default merge strategy in Git.



Squash and merge
`git merge --squash`
Combine all commits from the source branch into a single merge commit in the destination branch.



3-way merge
`git merge --no-ff`
Create a merge commit and adds individual source commits to the destination branch.



Commit message - optional Preview markdown

Squashed commit of the following

```
commit d49940ad
Author: Li Juan <li_juan@example.com>
Date: Tue May 07 2019 15:12:48 GMT-0700 (Pacific Daylight Time)

    Fixing the bug reported in 1234.
```

Author name

Email address

Delete source branch bugfix-bug1234 after merging?

Cancel
Merge pull request

- Penggabungan fast-forward menggerakkan referensi untuk cabang tujuan maju ke commit terbaru cabang sumber. Ini adalah perilaku default Git bila memungkinkan. Tidak ada commit penggabungan yang dibuat, tetapi semua riwayat commit dari cabang sumber disimpan seolah-olah telah terjadi di cabang tujuan. Penggabungan fast-forward tidak muncul sebagai gabungan cabang dalam tampilan visualizer commit riwayat cabang tujuan karena tidak ada gabungan commit dibuat. Ujung cabang sumber cepat diteruskan ke ujung cabang tujuan.
- Penggabungan squash menciptakan satu commit yang berisi perubahan di cabang sumber dan memberlakukan satu commit squash ke cabang tujuan. Secara default, pesan commit

untuk commit squash berisi semua pesan commit dari perubahan di cabang sumber. Tidak ada riwayat commit individu perubahan cabang yang disimpan. Hal ini dapat membantu menyederhanakan riwayat repositori Anda dengan terus mempertahankan representasi grafis dari gabungan dalam tampilan visualizer commit riwayat cabang tujuan.

- Penggabungan tiga arah menciptakan commit gabungan untuk penggabungan di cabang tujuan, tetapi juga mempertahankan commit individu yang dibuat di cabang sumber sebagai bagian dari riwayat cabang tujuan. Hal ini dapat membantu mempertahankan riwayat perubahan lengkap pada repositori Anda.
8. Jika Anda memilih strategi penggabungan squash atau 3-way, tinjau pesan commit yang dihasilkan secara otomatis dan ubah jika Anda ingin mengubah informasi. Tambahkan nama dan alamat email Anda untuk riwayat commit.
 9. (Opsional) Hapus opsi untuk menghapus cabang sumber sebagai bagian dari penggabungan. Default-nya adalah menghapus cabang sumber ketika permintaan pull digabung.
 10. Pilih Gabungkan permintaan pull untuk menyelesaikan penggabungan.

Menggabungkan permintaan pull (AWS CLI)

Untuk menggunakan perintah AWS CLI dengan CodeCommit, instal AWS CLI. Untuk informasi lebih lanjut, lihat [Referensi baris perintah](#).

Untuk menggunakan AWS CLI untuk menggabungkan pull request dalam sebuah repositori CodeCommit

1. Untuk mengevaluasi apakah permintaan pull telah memenuhi semua aturan persetujuan dan siap untuk digabung, jalankan perintah `evaluate-pull-request-approval-rules`, menentukan:
 - ID dari permintaan pull (menggunakan opsi `--pull-request-id`).
 - ID revisi dari permintaan pull (menggunakan `--revision-id` option). Anda bisa mendapatkan ID revisi saat ini untuk permintaan pull dengan menggunakan perintah [get-pull-request](#).

Misalnya, untuk mengevaluasi keadaan aturan persetujuan pada permintaan pull dengan ID `27` dan ID revisi `9f29d167EXAMPLE`:

```
aws codecommit evaluate-pull-request-approval-rules --pull-request-id 27 --  
revision-id 9f29d167EXAMPLE
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "evaluation": {
    "approved": false,
    "approvalRulesNotSatisfied": [
      "Require two approved approvers"
    ],
    "overridden": false,
    "approvalRulesSatisfied": []
  }
}
```

Note

Output ini menunjukkan bahwa permintaan pull tidak dapat digabungkan karena persyaratan aturan persetujuan belum dipenuhi. Untuk menggabungkan permintaan pull ini, Anda dapat meminta pengulas menyetujuinya untuk memenuhi persyaratan aturan. Tergantung pada izin dan cara aturan dibuat, Anda mungkin juga dapat mengedit, menimpa, atau menghapus aturan. Untuk informasi lebih lanjut, lihat [Meninjau permintaan pull](#), [Menimpa aturan persetujuan pada sebuah permintaan pull](#), dan [Mengedit atau menghapus aturan persetujuan untuk permintaan pull](#).

- Untuk menggabungkan dan menutup permintaan pull menggunakan strategi penggabungan fast-forward, jalankan perintah `merge-pull-request-by-fast-forward`, menentukan:
 - ID dari permintaan pull (dengan opsi `--pull-request-id`).
 - ID commit penuh dari ujung cabang sumber (dengan opsi `--source-commit-id`).
 - Nama repositori (dengan opsi `--repository-name`).

Misalnya, untuk menggabungkan dan menutup permintaan pull dengan ID **47** dan sebuah ID commit sumber **99132ab0EXAMPLE** dalam repositori bernama **MyDemoRepo**:

```
aws codecommit merge-pull-request-by-fast-forward --pull-request-id 47 --source-commit-id 99132ab0EXAMPLE --repository-name MyDemoRepo
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```

{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "I want one approver for this pull request",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ],
    "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
    "clientRequestToken": "",
    "creationDate": 1508530823.142,
    "description": "Review the latest changes and updates to the global variables",
    "lastActivityDate": 1508887223.155,
    "pullRequestId": "47",
    "pullRequestStatus": "CLOSED",
    "pullRequestTargets": [
      {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
          "isMerged": true,
          "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
      }
    ],
    "title": "Consolidation of global variables"
  }
}

```

- Untuk menggabungkan dan menutup permintaan pull menggunakan strategi penggabungan squash, jalankan perintah `merge-pull-request-by-squash`, menentukan:

- ID permintaan pull (dengan opsi `--pull-request-id`).
- ID commit penuh dari ujung cabang sumber (dengan opsi `--source-commit-id`).
- Nama repositori (dengan opsi `--repository-name`).
- Tingkat detail konflik yang ingin Anda gunakan (dengan opsi `--conflict-detail-level`). Jika tidak ditentukan, default **FILE_LEVEL** digunakan.
- Strategi resolusi konflik yang ingin Anda gunakan (dengan opsi `--conflict-resolution-strategy`). Jika tidak ditentukan, secara default diatur ke **NONE**, dan konflik harus diselesaikan secara manual.
- Pesan commit yang harus disertakan (dengan opsi `--commit-message`).
- Nama yang digunakan untuk commit (dengan opsi `--author-name`).
- Alamat email yang digunakan untuk commit (dengan opsi `--email`).
- Apakah menyimpan folder kosong (dengan opsi `--keep-empty-folders`).

Contoh berikut menggabungkan dan menutup permintaan pull dengan ID *47* dan sebuah ID commit sumber *99132ab0EXAMPLE* dalam repositori bernama *MyDemoRepo*. Ini menggunakan detail konflik **LINE_LEVEL** dan strategi penyelesaian konflik **ACCEPT_SOURCE**:

```
aws codecommit merge-pull-request-by-squash --pull-request-id 47 --source-commit-id 99132ab0EXAMPLE --repository-name MyDemoRepo --conflict-detail-level LINE_LEVEL --conflict-resolution-strategy ACCEPT_SOURCE --author-name "Jorge Souza" --email "jorge_souza@example.com" --commit-message "Merging pull request 47 by squash and accepting source in merge conflicts"
```

Jika berhasil, perintah ini menghasilkan jenis output yang sama sebagai penggabungan dengan `fast-forward`, output mirip dengan berikut ini:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"DestinationReferences\": [\"refs/heads/main\"], \"Statements\": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approver-rule-for-main",
```

```
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "originApprovalRuleTemplate": {
            "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
            "approvalRuleTemplateName": "2-approver-rule-for-main"
        },
        "ruleContentSha256": "4711b576EXAMPLE"
    }
],
"authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
"clientRequestToken": "",
"creationDate": 1508530823.142,
"description": "Review the latest changes and updates to the global
variables",
"lastActivityDate": 1508887223.155,
"pullRequestId": "47",
"pullRequestStatus": "CLOSED",
"pullRequestTargets": [
    {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
            "isMerged": true,
            "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
    }
],
"title": "Consolidation of global variables"
}
```

4. Untuk menggabungkan dan menutup permintaan pull menggunakan strategi penggabungan tiga arah, jalankan perintah `merge-pull-request-by-three-way`, menentukan:
- ID dari permintaan pull (dengan opsi `--pull-request-id`).
 - ID commit penuh dari ujung cabang sumber (dengan opsi `--source-commit-id`).
 - Nama repositori (dengan opsi `--repository-name`).

- Tingkat detail konflik yang ingin Anda gunakan (dengan opsi `--conflict-detail-level`). Jika tidak ditentukan, default **FILE_LEVEL** digunakan.
- Strategi resolusi konflik yang ingin Anda gunakan (dengan opsi `--conflict-resolution-strategy`). Jika tidak ditentukan, secara default diatur ke **NONE**, dan konflik harus diselesaikan secara manual.
- Pesan commit yang harus disertakan (dengan opsi `--commit-message`).
- Nama yang digunakan untuk commit (dengan opsi `--author-name`).
- Alamat email yang digunakan untuk commit (dengan opsi `--email`).
- Apakah menyimpan folder kosong (dengan opsi `--keep-empty-folders`).

Contoh berikut menggabungkan dan menutup permintaan pull dengan ID **47** dan sebuah ID commit sumber dari **99132ab0EXAMPLE** dalam repositori **MyDemoRepo**. Ini menggunakan opsi default untuk detail konflik dan strategi penyelesaian konflik:

```
aws codecommit merge-pull-request-by-three-way --pull-request-id 47 --source-commit-id 99132ab0EXAMPLE --repository-name MyDemoRepo --author-name "Maria Garcia" --email "maria_garcia@example.com" --commit-message "Merging pull request 47 by three-way with default options"
```

Jika berhasil, perintah ini menghasilkan jenis yang sama output sebagai penggabungan dengan `fast-forward`, mirip dengan berikut ini:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"DestinationReferences\": [\"refs/heads/main\"], \"Statements\": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approver-rule-for-main",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "originApprovalRuleTemplate": {
          "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
          "approvalRuleTemplateName": "2-approver-rule-for-main"
        }
      }
    ]
  }
}
```

```
        },
        "ruleContentSha256": "4711b576EXAMPLE"
    }
],
"authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
"clientRequestToken": "",
"creationDate": 1508530823.142,
"description": "Review the latest changes and updates to the global
variables",
"lastActivityDate": 1508887223.155,
"pullRequestId": "47",
"pullRequestStatus": "CLOSED",
"pullRequestTargets": [
    {
        "destinationCommit": "9f31c968EXAMPLE",
        "destinationReference": "refs/heads/main",
        "mergeMetadata": {
            "isMerged": true,
            "mergedBy": "arn:aws:iam::123456789012:user/Mary_Major"
        },
        "repositoryName": "MyDemoRepo",
        "sourceCommit": "99132ab0EXAMPLE",
        "sourceReference": "refs/heads/variables-branch"
    }
],
"title": "Consolidation of global variables"
}
}
```

Menyelesaikan konflik dalam permintaan pull dalam repositori AWS CodeCommit

Jika permintaan pull Anda memiliki konflik dan tidak dapat digabung, Anda dapat mencoba menyelesaikan konflik dengan salah satu dari beberapa cara:

- Pada komputer lokal Anda, Anda dapat menggunakan perintah git diff untuk menemukan konflik antara dua cabang dan membuat perubahan untuk mengatasinya. Anda juga dapat menggunakan alat perbandingan atau perangkat lunak lain untuk membantu Anda menemukan dan menyelesaikan perbedaan. Setelah Anda menyelesaikannya sesuai keinginan Anda, Anda dapat mengirim

cabang sumber Anda dengan perubahan yang berisi konflik yang diselesaikan, yang memperbarui permintaan pull. Untuk informasi selengkapnya tentang git diff dan git difftool, lihat dokumentasi Git.

- Di konsol, Anda dapat memilih Selesaikan konflik. Ini akan membuka editor teks biasa yang menunjukkan konflik dengan cara yang sama seperti perintah git diff. Anda dapat secara manual meninjau konflik di setiap file yang berisi konflik, membuat perubahan, dan kemudian memperbarui permintaan pull dengan perubahan Anda.
- Di AWS CLI, Anda dapat menggunakan AWS CLI untuk mendapatkan informasi tentang konflik penggabungan dan membuat commit gabungan yang belum dirujuk untuk menguji penggabungan.

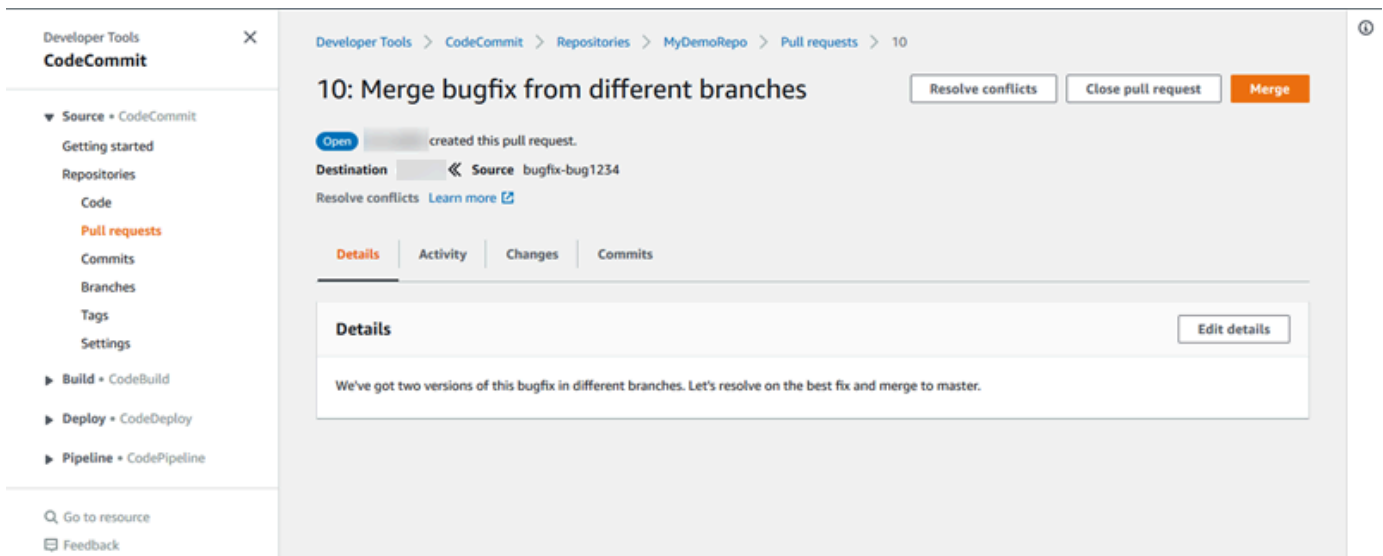
Topik

- [Menyelesaikan konflik dalam permintaan pull \(konsol\)](#)
- [Menyelesaikan konflik dalam permintaan pull \(AWS CLI\)](#)

Menyelesaikan konflik dalam permintaan pull (konsol)

Anda dapat menggunakan konsol CodeCommit untuk menyelesaikan konflik dalam permintaan pull di repositori CodeCommit.

1. Buka konsol CodeCommit di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori.
3. Di panel navigasi, pilih Permintaan pull.
4. Secara default, daftar semua permintaan pull terbuka ditampilkan. Pilih permintaan pull terbuka yang ingin Anda gabungkan tetapi berisi konflik.
5. Dalam permintaan pull, pilih Selesaikan konflik. Opsi ini hanya muncul jika ada konflik yang harus diselesaikan sebelum permintaan pull dapat digabung.



6. Jendela penyelesaian konflik membuka daftar setiap file yang memiliki konflik yang harus diselesaikan. Pilih setiap file dalam daftar untuk meninjau konflik, dan membuat perubahan yang diperlukan sampai semua konflik telah diselesaikan.

Developer Tools > CodeCommit > Repositories > MyDemoRepo > Pull requests > 10 > Resolve conflicts

Resolve conflicts

Resolve conflicts in each of the files in the list. When you have resolved all conflicts, update the pull request and review the merge strategies available. [Info](#)

Destination << Source bugfix-bug1234

Editing: helloworld.py

helloworld.py	1
<pre>1 import sys 2 3 print('Hello, World!') 4 5 <<<<<< HEAD:helloworld.py 6 print('The sum of 2 and 3 is 5.') 7 ===== 8 print('The sum of 3 and 2 is 5.') 9 >>>>>> bugfix-bug1234:helloworld.py 10 11 sum = int(sys.argv[1]) + int(sys.argv[2]) 12 13 print('The sum of {0} and {1} is {2}.'.format(sys.argv[1], sys.argv[2], sum))</pre>	

Allow the merge to proceed with Git conflict markers still present.

- Anda dapat memilih untuk menggunakan isi file sumber, isi file tujuan, atau jika file bukan file biner, untuk mengedit isi file secara manual sehingga hanya berisi perubahan yang Anda inginkan. Penanda diff git standar digunakan untuk menunjukkan konflik antara tujuan (HEAD) dan cabang sumber dalam file.
- Jika suatu file adalah file biner, Git submodule, atau jika ada konflik nama file/folder, Anda harus memilih untuk menggunakan file sumber atau file tujuan untuk menyelesaikan konflik. Anda tidak dapat melihat atau mengedit file biner di konsol CodeCommit.
- Jika ada konflik mode file, Anda melihat opsi untuk menyelesaikan konflik tersebut dengan memilih antara mode file dari file sumber dan mode file dari file tujuan.

- Jika Anda memutuskan ingin membuang perubahan untuk file dan mengembalikannya ke keadaan konflik, pilih Atur ulang file. Hal ini memungkinkan Anda menyelesaikan konflik dengan cara yang berbeda.
7. Jika Anda puas dengan perubahan Anda, pilih Perbarui permintaan pull.

Note

Anda harus menyelesaikan semua konflik di semua file sebelum Anda dapat berhasil memperbarui permintaan pull dengan perubahan Anda.

8. Permintaan pull diperbarui dengan perubahan Anda dan dapat digabung. Anda melihat halaman penggabungan. Anda dapat memilih untuk menggabungkan permintaan pull saat ini, atau Anda dapat kembali ke daftar permintaan pull.

Menyelesaikan konflik dalam permintaan pull (AWS CLI)

Untuk menggunakan perintah AWS CLI dengan CodeCommit, instal AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Tidak ada satu perintah AWS CLI pun yang membuat Anda dapat menyelesaikan konflik dalam permintaan pull dan menggabungkan permintaan itu. Namun, Anda dapat menggunakan perintah individu untuk menemukan konflik, mencoba untuk menyelesaikannya, dan menguji apakah permintaan pull dapat digabung. Anda dapat menggunakan:

- `get-merge-options`, untuk mengetahui opsi gabungan apa yang tersedia untuk penggabungan antara dua penentu commit.
- `get-merge-conflicts`, untuk mengembalikan daftar file dengan konflik penggabungan dalam suatu penggabungan antara dua penentu commit.
- `batch-describe-merge-conflicts`, untuk mendapatkan informasi tentang semua konflik penggabungan dalam file dalam penggabungan antara dua commit menggunakan strategi penggabungan tertentu.
- `describe-merge-conflicts`, untuk mendapatkan informasi tentang konflik penggabungan untuk file khusus antara dua commit menggunakan strategi penggabungan yang ditentukan.
- `create-unreferenced-merge-commit`, untuk menguji hasil penggabungan dua penentu commit menggunakan strategi gabungan yang ditentukan.

1.

Untuk menemukan opsi penggabungan apa yang tersedia untuk penggabungan antara dua penentu commit, jalankan perintah `get-merge-options`, menentukan:

- Sebuah penentu commit untuk sumber penggabungan (dengan opsi `--source-commit-specifier`).
- Sebuah penentu commit untuk tujuan untuk penggabungan (dengan opsi `--destination-commit-specifier`).
- Nama repositori (dengan opsi `--repository-name`).
- (Opsional) Strategi penyelesaian konflik untuk digunakan (dengan opsi `--conflict-resolution-strategy`).
- (Opsional) Tingkat detail yang Anda inginkan tentang konflik apapun (dengan opsi `--conflict-detail-level`).

Misalnya, untuk menentukan strategi penggabungan yang tersedia untuk menggabungkan cabang sumber yang bernama *bugfix-1234* dengan cabang tujuan yang bernama *utama* di sebuah repositori yang bernama *MyDemoRepo*:

```
aws codecommit get-merge-options --source-commit-specifier bugfix-1234 --  
destination-commit-specifier main --repository-name MyDemoRepo
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{  
  "mergeOptions": [  
    "FAST_FORWARD_MERGE",  
    "SQUASH_MERGE",  
    "THREE_WAY_MERGE"  
  ],  
  "sourceCommitId": "d49940adEXAMPLE",  
  "destinationCommitId": "86958e0aEXAMPLE",  
  "baseCommitId": "86958e0aEXAMPLE"  
}
```

2.

Untuk mendapatkan daftar file yang berisi konflik penggabungan dalam suatu penggabungan antara dua penentu commit, jalankan perintah `get-merge-conflicts`, menentukan:

- Sebuah penentu commit untuk sumber penggabungan (dengan opsi `--source-commit-specifier`).
- Sebuah penentu commit untuk tujuan untuk penggabungan (dengan opsi `--destination-commit-specifier`).
- Nama repositori (dengan opsi `--repository-name`).
- Opsi penggabungan yang ingin Anda gunakan (dengan opsi `--merge-option`).
- (Opsional) Tingkat detail yang Anda inginkan tentang konflik apapun (dengan opsi `--conflict-detail-level`).
- (Opsional) Strategi penyelesaian konflik untuk digunakan (dengan opsi `--conflict-resolution-strategy`).
- (Opsional) Jumlah maksimum file dengan konflik untuk kembali (dengan opsi `--max-conflict-files`).

Misalnya, untuk mendapatkan daftar file yang berisi konflik dalam suatu penggabungan antara cabang sumber bernama `fitur-randomizationfeature` dan cabang tujuan bernama utama menggunakan strategi penggabungan tiga arah dalam repositori bernama `MyDemoRepo`:

```
aws codecommit get-merge-conflicts --source-commit-specifier feature-
randomizationfeature --destination-commit-specifier main --merge-option
THREE_WAY_MERGE --repository-name MyDemoRepo
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "mergeable": false,
  "destinationCommitId": "86958e0aEXAMPLE",
  "sourceCommitId": "6ccd57fdEXAMPLE",
  "baseCommitId": "767b6958EXAMPLE",
  "conflictMetadataList": [
    {
      "filePath": "readme.md",
      "fileSizes": {
        "source": 139,
        "destination": 230,
        "base": 85
      },
      "fileModes": {
```



```
        "source": "NORMAL",
        "destination": "NORMAL",
        "base": "NORMAL"
    },
    "objectTypes": {
        "source": "FILE",
        "destination": "FILE",
        "base": "FILE"
    },
    "numberOfConflicts": 1,
    "isBinaryFile": {
        "source": false,
        "destination": false,
        "base": false
    },
    "contentConflict": true,
    "fileModeConflict": false,
    "objectTypeConflict": false,
    "mergeOperations": {
        "source": "M",
        "destination": "M"
    }
}
]
```

3.

Untuk mendapatkan informasi tentang konflik penggabungan di semua file atau subset dari file dalam penggabungan antara dua penentu commit, jalankan perintah `batch-describe-merge-conflicts`, menentukan:

- Sebuah penentu commit untuk sumber penggabungan (dengan opsi `--source-commit-specifier`).
- Sebuah penentu commit untuk tujuan untuk penggabungan (dengan opsi `--destination-commit-specifier`).
- Opsi gabungan yang ingin Anda gunakan (dengan opsi `--merge-option`).
- Nama repositori (dengan opsi `--repository-name`).
- (Opsional) Strategi penyelesaian konflik untuk digunakan (dengan opsi `--conflict-resolution-strategy`).
- (Opsional) Tingkat detail yang Anda inginkan tentang konflik apapun (dengan opsi `--conflict-detail-level`).

- (Opsional) Jumlah maksimum hunk penggabungan yang akan dikembalikan (dengan opsi `--max-merge-hunks`).
- (Opsional) Jumlah maksimum file dengan konflik yang akan dikembalikan (dengan opsi `--max-conflict-files`).
- (Opsional) Path dari file target untuk digunakan untuk menggambarkan konflik (dengan opsi `--file-paths`).

Misalnya, untuk menentukan konflik penggabungan untuk penggabungan cabang sumber bernama `feature-randomizationfeature` dengan cabang tujuan bernama `utama` menggunakan strategi `THREE_WAY_MERGE` dalam repositori bernama `MyDemoRepo`:

```
aws codecommit batch-describe-merge-conflicts --source-commit-specifier feature-randomizationfeature --destination-commit-specifier main --merge-option THREE_WAY_MERGE --repository-name MyDemoRepo
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "conflicts": [
    {
      "conflictMetadata": {
        "filePath": "readme.md",
        "fileSizes": {
          "source": 139,
          "destination": 230,
          "base": 85
        },
        "fileModes": {
          "source": "NORMAL",
          "destination": "NORMAL",
          "base": "NORMAL"
        },
        "objectTypes": {
          "source": "FILE",
          "destination": "FILE",
          "base": "FILE"
        },
        "numberOfConflicts": 1,
        "isBinaryFile": {
          "source": false,
```

```

        "destination": false,
        "base": false
    },
    "contentConflict": true,
    "fileModeConflict": false,
    "objectTypeConflict": false,
    "mergeOperations": {
        "source": "M",
        "destination": "M"
    }
},
"mergeHunks": [
    {
        "isConflict": true,
        "source": {
            "startLine": 0,
            "endLine": 3,
            "hunkContent": "VGhpcyBpEXAMPLE=="
        },
        "destination": {
            "startLine": 0,
            "endLine": 1,
            "hunkContent": "VXNlIHRoEXAMPLE="
        }
    }
]
}
],
"errors": [],
"destinationCommitId": "86958e0aEXAMPLE",
"sourceCommitId": "6ccd57fdEXAMPLE",
"baseCommitId": "767b6958EXAMPLE"
}

```

4.

Untuk mendapatkan informasi rinci tentang konflik penggabungan untuk file tertentu dalam gabungan antara dua penentu commit, jalankan perintah `describe-merge-conflicts`, menentukan:

- Sebuah penentu commit untuk sumber penggabungan (dengan opsi `--source-commit-specifier`).
- Sebuah penentu commit untuk tujuan untuk penggabungan (dengan opsi `--destination-commit-specifier`).
- Opsi penggabungan yang ingin Anda gunakan (dengan opsi `--merge-option`).

- Path dari file target untuk digunakan untuk menggambarkan konflik (dengan opsi `--file-path`).
- Nama repositori (dengan opsi `--repository-name`).
- (Opsional) Strategi penyelesaian konflik untuk digunakan (dengan opsi `--conflict-resolution-strategy`).
- (Opsional) Tingkat detail yang Anda inginkan tentang konflik apapun (dengan opsi `--conflict-detail-level`).
- (Opsional) Jumlah maksimum hunk penggabungan yang akan dikembalikan (dengan opsi `--max-merge-hunks`).
- (Opsional) Jumlah maksimum file dengan konflik yang akan dikembalikan (dengan opsi `--max-conflict-files`).

Sebagai contoh, untuk menentukan konflik penggabungan untuk file bernama `readme.md` di cabang sumber bernama `feature-randomizationfeature` dengan cabang tujuan bernama `utama` menggunakan strategi `THREE_WAY_MERGE` dalam repositori bernama `MyDemoRepo`:

```
aws codecommit describe-merge-conflicts --source-commit-specifier feature-randomizationfeature --destination-commit-specifier main --merge-option THREE_WAY_MERGE --file-path readme.md --repository-name MyDemoRepo
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "conflictMetadata": {
    "filePath": "readme.md",
    "fileSizes": {
      "source": 139,
      "destination": 230,
      "base": 85
    },
    "fileModes": {
      "source": "NORMAL",
      "destination": "NORMAL",
      "base": "NORMAL"
    },
    "objectTypes": {
      "source": "FILE",
      "destination": "FILE",
```

```
    "base": "FILE"
  },
  "numberOfConflicts": 1,
  "isBinaryFile": {
    "source": false,
    "destination": false,
    "base": false
  },
  "contentConflict": true,
  "fileModeConflict": false,
  "objectTypeConflict": false,
  "mergeOperations": {
    "source": "M",
    "destination": "M"
  }
},
"mergeHunks": [
  {
    "isConflict": true,
    "source": {
      "startLine": 0,
      "endLine": 3,
      "hunkContent": "VGhpcyBpEXAMPLE=="
    },
    "destination": {
      "startLine": 0,
      "endLine": 1,
      "hunkContent": "VXNlIHRoEXAMPLE="
    }
  }
],
"destinationCommitId": "86958e0aEXAMPLE",
"sourceCommitId": "6ccd57fdEXAMPLE",
"baseCommitId": "767b69580EXAMPLE"
}
```

5.

Untuk membuat commit yang tidak direferensikan yang mewakili hasil penggabungan dua penentu commit, jalankan perintah `create-unreferenced-merge-commit`, menentukan:

- Sebuah penentu commit untuk sumber penggabungan (dengan opsi `--source-commit-specifier`).

- Sebuah penentu commit untuk tujuan untuk penggabungan (dengan opsi `--destination-commit-specifier`).
- Opsi gabungan yang ingin Anda gunakan (dengan opsi `--merge-option`).
- Nama repositori (dengan opsi `--repository-name`).
- (Opsional) Strategi penyelesaian konflik untuk digunakan (dengan opsi `--conflict-resolution-strategy`).
- (Opsional) Tingkat detail yang Anda inginkan tentang konflik apapun (dengan opsi `--conflict-detail-level`).
- (Opsional) Pesan commit yang harus disertakan (dengan opsi `--commit-message`).
- (Opsional) Nama yang digunakan untuk commit (dengan opsi `--name`).
- (Opsional) Alamat email yang akan digunakan untuk commit (dengan opsi `--email`).
- (Opsional) Apakah akan menyimpan folder kosong (dengan opsi `--keep-empty-folders`).

Misalnya, untuk menentukan konflik penggabungan untuk penggabungan cabang sumber bernama *bugfix-1234* dengan cabang tujuan bernama *utama* menggunakan strategi ACCET_SOURCE dalam repositori bernama *MyDemoRepo*:

```
aws codecommit create-unreferenced-merge-commit --source-commit-specifier bugfix-1234 --destination-commit-specifier main --merge-option THREE_WAY_MERGE --repository-name MyDemoRepo --name "Maria Garcia" --email "maria_garcia@example.com" --commit-message "Testing the results of this merge."
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "commitId": "4f178133EXAMPLE",
  "treeId": "389765daEXAMPLE"
}
```

Menutup sebuah pull request dalam sebuah repositori AWS CodeCommit

Jika Anda ingin menutup pull request tanpa menggabungkan kode, Anda dapat melakukannya dengan salah satu dari beberapa cara berikut:

- Di konsol, Anda dapat menutup sebuah pull request tanpa menggabungkan kode. Anda mungkin ingin melakukan ini jika Anda ingin menggunakan perintah git merge untuk menggabungkan cabang secara manual, atau jika kode dalam cabang sumber pull request bukanlah kode yang Anda ingin gabungkan ke cabang tujuan.
- Anda dapat menghapus cabang sumber yang ditentukan dalam pull request tersebut. CodeCommit menutup sebuah pull request secara otomatis jika sumber atau cabang tujuan dari pull request tersebut dihapus.
- Di AWS CLI, Anda dapat memperbarui status pull request dari OPEN menjadi CLOSED. Hal ini akan menutup pull request tanpa menggabungkan kode.

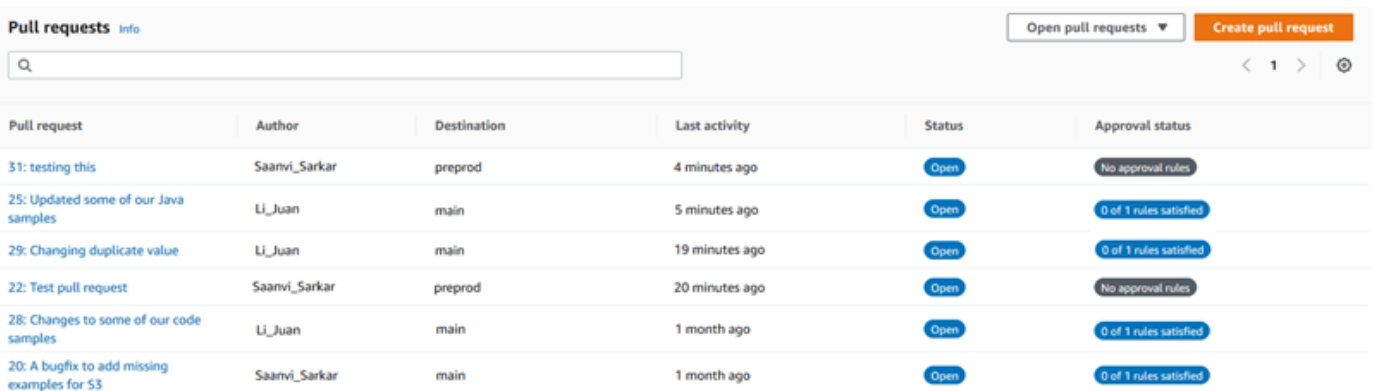
Topik

- [Menutup sebuah pull request \(konsol\)](#)
- [Menutup sebuah pull request \(AWS CLI\)](#)

Menutup sebuah pull request (konsol)

Anda dapat menggunakan konsol CodeCommit untuk menutup sebuah pull request dalam sebuah repositori CodeCommit. Setelah status pull request diubah menjadi Tutup, maka pull request itu tidak dapat diubah statusnya kembali ke Buka, namun pengguna masih dapat mengomentari perubahan dan membalas komentar.

1. Buka konsol CodeCommit di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori.
3. Di panel navigasi, pilih Permintaan pull.
4. Secara default, daftar semua pull request terbuka ditampilkan. Pilih pull request terbuka yang ingin Anda tutup.



Pull request	Author	Destination	Last activity	Status	Approval status
31: testing this	Saanvi_Sarkar	preprod	4 minutes ago	Open	No approval rules
25: Updated some of our Java samples	Li_Juan	main	5 minutes ago	Open	0 of 1 rules satisfied
29: Changing duplicate value	Li_Juan	main	19 minutes ago	Open	0 of 1 rules satisfied
22: Test pull request	Saanvi_Sarkar	preprod	20 minutes ago	Open	No approval rules
28: Changes to some of our code samples	Li_Juan	main	1 month ago	Open	0 of 1 rules satisfied
20: A bugfix to add missing examples for S3	Saanvi_Sarkar	main	1 month ago	Open	0 of 1 rules satisfied

5. Dalam pull request, pilih Tutup pull request. Opsi ini menutup pull request tanpa mencoba untuk menggabungkan cabang sumber ke cabang tujuan. Opsi ini tidak menyediakan cara untuk menghapus cabang sumber sebagai bagian dari menutup pull request, tetapi Anda dapat melakukannya sendiri setelah permintaan itu ditutup.

Menutup sebuah pull request (AWS CLI)

Untuk menggunakan perintah AWS CLI dengan CodeCommit, instal AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk menggunakan AWS CLI untuk menutup pull request dalam sebuah repositori CodeCommit

- Untuk memperbarui status pull request dalam repositori dari OPEN ke CLOSED, jalankan perintah `update-pull-request-status`, yang menentukan:
 - ID dari pull request (dengan opsi `--pull-request-id`).
 - Status pull request (dengan opsi `--pull-request-status`).

Misalnya, untuk memperbarui status pull request dengan ID `42` ke status `TUTUP` dalam repositori CodeCommit bernama `MyDemoRepo`:

```
aws codecommit update-pull-request-status --pull-request-id 42 --pull-request-status CLOSED
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "pullRequest": {
    "approvalRules": [
      {
        "approvalRuleContent": "{\"Version\": \"2018-11-08\", \"Statements\": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
        "approvalRuleId": "dd8b17fe-EXAMPLE",
        "approvalRuleName": "2-approvers-needed-for-this-change",
        "creationDate": 1571356106.936,
        "lastModifiedDate": 571356106.936,
        "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
        "ruleContentSha256": "4711b576EXAMPLE"
      }
    ]
  }
}
```



```
    }
  ],
  "authorArn": "arn:aws:iam::123456789012:user/Li_Juan",
  "clientRequestToken": "",
  "creationDate": 1508530823.165,
  "description": "Updated the pull request to remove unused global
variable.",
  "lastActivityDate": 1508372423.12,
  "pullRequestId": "47",
  "pullRequestStatus": "CLOSED",
  "pullRequestTargets": [
    {
      "destinationCommit": "9f31c968EXAMPLE",
      "destinationReference": "refs/heads/main",
      "mergeMetadata": {
        "isMerged": false,
      },
      "repositoryName": "MyDemoRepo",
      "sourceCommit": "99132ab0EXAMPLE",
      "sourceReference": "refs/heads/variables-branch"
    }
  ],
  "title": "Consolidation of global variables"
}
}
```

Bekerja dengan templat aturan persetujuan

Anda dapat membuat aturan persetujuan untuk permintaan pull. Untuk secara otomatis menerapkan aturan persetujuan untuk beberapa atau semua permintaan pull yang dibuat dalam repositori, gunakan templat aturan persetujuan. Templat aturan persetujuan membantu Anda menyesuaikan alur kerja pengembangan Anda di seluruh repositori sehingga cabang yang berbeda memiliki tingkat persetujuan dan kontrol yang sesuai. Anda dapat menentukan aturan yang berbeda untuk cabang produksi dan pengembangan. Aturan-aturan tersebut diterapkan setiap kali permintaan pull yang cocok dengan syarat aturan dibuat. Untuk informasi selengkapnya tentang kebijakan terkelola dan izin untuk templat aturan persetujuan, lihat [izin untuk tindakan pada templat aturan persetujuan](#) dan [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Anda dapat mengaitkan templat aturan persetujuan dengan satu atau lebih repositori di Wilayah AWS di mana mereka dibuat. Ketika templat dikaitkan dengan repositori, templat secara otomatis membuat aturan persetujuan untuk permintaan pull di repositori itu sebagai bagian dari pembuatan permintaan pull. Sama seperti aturan persetujuan tunggal, templat aturan persetujuan menentukan struktur aturan persetujuan, termasuk jumlah persetujuan yang diperlukan dan grup pengguna opsional tempat asal persetujuan. Tidak seperti aturan persetujuan, Anda juga dapat menentukan referensi tujuan (cabang atau cabang-cabang), juga dikenal sebagai filter cabang. Jika Anda menentukan referensi tujuan, maka hanya permintaan pull yang nama cabang tujuannya cocok dengan nama cabang yang ditentukan (referensi tujuan) dalam templat yang memiliki aturan yang dibuat untuknya. Jadi, misalnya, jika Anda menentukan **refs/heads/main** sebagai referensi tujuan, aturan persetujuan yang didefinisikan dalam templat hanya diterapkan untuk permintaan pull jika cabang tujuan `main`.

Approval rule template

Approval rule template name

Description - *optional*

Before a pull request can be merged to the main branch, at least one senior developer must give an approval to the changes.

Number of approvals needed

Approval pool members - *optional*

If approval pool members are specified, only approvals from these members will count toward satisfying this rule. You can use wildcards to match multiple approvers with one value.

Approver type [Info](#)

Value

IAM user name or assumed role ▼

Remove

Fully qualified ARN ▼

Remove

Add

Branch filters - *optional*

Use branch filters to only apply this template to a pull request if the destination branch name matches a name in the filter list.

Branch name

Remove

Add

▼ Associated repositories

Repositories - *optional*

MyDemoRepo ✕

MyTestRepo ✕

Topik

- [Membuat templat aturan persetujuan](#)
- [Kaitkan templat aturan persetujuan dengan repositori](#)
- [Mengelola templat aturan persetujuan](#)
- [Melepaskan kaitan templat aturan persetujuan](#)
- [Menghapus templat aturan persetujuan](#)

Membuat templat aturan persetujuan

Anda dapat membuat satu atau beberapa templat aturan persetujuan untuk membantu Anda menyesuaikan alur kerja pengembangan di seluruh repositori. Dengan membuat beberapa templat, Anda dapat mengkonfigurasi pembuatan otomatis aturan persetujuan sehingga cabang yang berbeda memiliki tingkat persetujuan dan kontrol yang sesuai. Misalnya, Anda dapat membuat templat yang berbeda untuk cabang produksi dan pengembangan dan menerapkan templat ini ke satu atau lebih repositori. Ketika pengguna membuat permintaan pull di repositori tersebut, permintaan dievaluasi terhadap templat tersebut. Jika permintaan cocok dengan syarat dalam templat yang diterapkan, aturan persetujuan dibuat untuk permintaan pull.

Anda bisa menggunakan konsol atau AWS CLI untuk membuat templat aturan persetujuan. Untuk informasi selengkapnya tentang kebijakan terkelola dan izin untuk templat aturan persetujuan, lihat [izin untuk tindakan pada templat aturan persetujuan](#) dan [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Topik

- [Membuat templat aturan persetujuan \(konsol\)](#)
- [Buat templat aturan persetujuan \(AWS CLI\)](#)

Membuat templat aturan persetujuan (konsol)

Templat aturan persetujuan tidak dikaitkan dengan repositori apa pun secara default. Anda dapat membuat hubungan antara templat dan satu repositori atau lebih ketika Anda membuat templat, atau Anda dapat menambahkan hubungan di lain waktu.

Untuk membuat templat aturan persetujuan (Konsol)

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Pilih Templat aturan persetujuan, lalu pilih Buat templat.


3. Dalam Nama templat aturan persetujuan, berikan nama deskriptif untuk templat sehingga Anda tahu untuk apa templat dibuat. Misalnya, jika Anda ingin meminta satu orang dari sekumpulan developer senior untuk menyetujui permintaan pull sebelum dapat digabungkan, Anda dapat menyebutkan aturan **Require 1 approver from a senior developer**.
4. (Opsional) Dalam Deskripsi, berikan deskripsi tentang tujuan templat ini. Hal ini dapat membantu orang lain memutuskan apakah templat ini sesuai untuk repositori mereka.
5. Dalam Jumlah persetujuan yang dibutuhkan, masukkan jumlah yang Anda inginkan. Nilai default-nya adalah 1.
6. (Opsional) Jika Anda ingin mengharuskan bahwa persetujuan untuk permintaan pull berasal dari grup pengguna tertentu, di Anggota aturan persetujuan, pilih Tambahkan. Dalam Jenis penyetuju, pilih salah satu dari berikut ini:
 - Nama pengguna atau peran yang dimiliki IAM: Opsi ini mem-prepopulasikan ID akun Amazon Web Services untuk akun yang Anda gunakan untuk masuk, dan hanya memerlukan nama. Ini dapat digunakan untuk baik oleh pengguna IAM maupun pengguna akses gabungan yang namanya cocok dengan nama yang disediakan. Ini adalah pilihan yang sangat kuat yang menawarkan fleksibilitas tinggi. Sebagai contoh, jika Anda memilih opsi ini dan masuk dengan akun Amazon Web Services 123456789012, dan Anda menentukan **Mary_Major**, semua hal berikut dihitung sebagai persetujuan yang berasal dari pengguna tersebut:
 - Pengguna IAM di akun (`arn:aws:iam::123456789012:user/Mary_Major`)
 - Seorang pengguna gabungan yang diidentifikasi dalam IAM sebagai `Mary_Major` (`arn:aws:sts::123456789012:federated-user/Mary_Major`)

Opsi ini tidak mengenali sesi aktif seseorang dengan memiliki peran **CodeCommitReview** dengan nama sesi peran `Mary_Major` (`arn:aws:sts::123456789012:assumed-role/CodeCommitReview/Mary_Major`) kecuali Anda menyertakan wildcard (`*Mary_Major`). Anda juga dapat menentukan nama peran secara eksplisit (`CodeCommitReview/Mary_Major`).

- ARN memenuhi syarat: Opsi ini memungkinkan Anda untuk menentukan Amazon Resource Name (ARN) yang memenuhi syarat dari pengguna atau peran IAM. Opsi ini juga mendukung peran yang dimiliki yang digunakan oleh layanan AWS, seperti AWS Lambda dan AWS CodeBuild. Untuk peran yang dimiliki, format ARN seharusnya `arn:aws:sts::AccountID:assumed-role/RoleName` untuk peran dan `arn:aws:sts::AccountID:assumed-role/FunctionName` untuk fungsi.

Jika Anda memilih nama pengguna atau peran yang dimiliki IAM sebagai tipe penyetuju, di Nilai, masukkan nama pengguna IAM atau peran atau ARN yang memenuhi syarat dari pengguna atau peran. Pilih Tambahkan lagi untuk menambahkan lebih banyak pengguna atau peran, hingga Anda menambahkan semua pengguna atau peran yang persetujuannya dihitung untuk jumlah persetujuan yang diperlukan.

Kedua jenis penyetuju memungkinkan Anda menggunakan wildcard (*) dalam nilainya. Misalnya, jika Anda memilih nama pengguna atau peran yang dimiliki IAM, dan Anda menentukan **CodeCommitReview/***, semua pengguna yang mengambil peran **CodeCommitReview** dihitung dalam grup persetujuan. Nama sesi peran masing-masing dihitung terhadap jumlah penyetuju yang diperlukan. Dengan cara ini, baik Mary_Major maupun Li_Juan dihitung sebagai persetujuan ketika masuk dan memiliki peran CodeCommitReview. Untuk informasi lebih lanjut tentang IAM ARN, wildcard, dan format, lihat [Pengidentifikasi IAM](#).

 Note

Aturan persetujuan tidak mendukung persetujuan lintas akun.

7. (Opsional) Dalam Filter cabang, masukkan nama cabang tujuan untuk digunakan untuk memfilter pembuatan aturan persetujuan. Misalnya, jika Anda menentukan *utama*, aturan persetujuan dibuat untuk permintaan pull dalam repositori terkait hanya jika cabang tujuan untuk permintaan pull adalah sebuah cabang bernama *utama*. Anda dapat menggunakan wildcard (*) dalam nama cabang untuk menerapkan aturan persetujuan untuk semua nama cabang yang cocok dengan kasus wildcard. Namun, Anda tidak dapat menggunakan wildcard di awal nama cabang. Anda dapat menentukan hingga 100 nama cabang. Jika Anda tidak menentukan filter apapun, templat berlaku untuk semua cabang di repositori terkait.
8. (Opsional) Repositori terkait, di daftar Repositori, pilih repositori dalam Wilayah AWS ini yang ingin Anda kaitkan dengan aturan persetujuan ini.

 Note

Anda dapat memilih untuk mengaitkan repositori setelah membuat templat. Untuk informasi selengkapnya, lihat [Kaitkan templat aturan persetujuan dengan repositori](#).

9. Pilih Buat.

Approval rule template

Approval rule template name

Require 1 approver from a senior developer for main branch

Description - *optional*

Before a pull request can be merged to the main branch, at least one senior developer must give an approval to the changes.

Number of approvals needed

1

Approval pool members - *optional*

If approval pool members are specified, only approvals from these members will count toward satisfying this rule. You can use wildcards to match multiple approvers with one value.

Approver type [Info](#)

Value

IAM user name or assumed role ▼

SeniorDevelopers/*

Remove

Fully qualified ARN ▼

:iam::123456789012:user/Mary_Major

Remove

Add

Branch filters - *optional*

Use branch filters to only apply this template to a pull request if the destination branch name matches a name in the filter list.

Branch name

main

Remove

Add

▼ Associated repositories

Repositories - *optional*

MyDemoRepo ✕

MyTestRepo ✕

Buat templat aturan persetujuan (AWS CLI)

Anda dapat menggunakan AWS CLI untuk membuat templat aturan persetujuan. Ketika Anda menggunakan AWS CLI, Anda dapat menentukan referensi tujuan untuk templat, sehingga hanya berlaku untuk permintaan pull cabang tujuan yang cocok dengan yang ada di templat.

Untuk membuat templat aturan persetujuan (AWS CLI)

1. Pada sebuah terminal atau baris perintah, jalankan perintah `create-approval-rule-template`, dengan menentukan:
 - Nama untuk templat aturan persetujuan. Pertimbangkan untuk menggunakan nama yang menggambarkan tujuannya.
 - Deskripsi templat aturan persetujuan. Seperti namanya, pertimbangkan untuk memberikan penjelasan mendetail.
 - Struktur JSON dari templat aturan persetujuan. Struktur ini dapat mencakup persyaratan untuk referensi tujuan, yang merupakan cabang tujuan untuk permintaan pull yang aturan persetujuannya diterapkan, dan anggota grup persetujuan, yang merupakan pengguna yang persetujuannya dihitung terhadap jumlah persetujuan yang diperlukan.

Saat membuat isi aturan persetujuan, Anda dapat menentukan penyetuju di grup persetujuan dengan salah satu dari dua cara berikut:

- `CodeCommitApprovers`: Opsi ini hanya memerlukan akun dan sumber daya Amazon Web Services dan sumber daya. Ini dapat digunakan baik untuk pengguna IAM maupun pengguna akses gabungan yang namanya cocok dengan nama sumber daya yang disediakan. Ini adalah pilihan yang sangat kuat yang menawarkan fleksibilitas tinggi. Sebagai contoh, jika Anda menentukan akun AWS 123456789012 dan **Mary_Major**, semua hal berikut dihitung sebagai persetujuan yang berasal dari pengguna tersebut:
 - Pengguna IAM di akun (`arn:aws:iam::123456789012:user/Mary_Major`)
 - Seorang pengguna gabungan yang diidentifikasi dalam IAM sebagai `Mary_Major` (`arn:aws:sts::123456789012:federated-user/Mary_Major`)

Opsi ini tidak mengenali sesi aktif seseorang dengan peran *SeniorDevelopers* dengan nama sesi peran *Mary_Major* (`arn:aws:sts::123456789012:assumed-role/SeniorDevelopers/Mary_Major`) kecuali Anda menyertakan wildcard (`*Mary_Major`).

- ARN memenuhi syarat: Opsi ini memungkinkan Anda untuk menentukan Amazon Resource Name (ARN) yang memenuhi syarat dari pengguna atau peran IAM.

Untuk informasi lebih lanjut tentang IAM ARN, wildcard, dan format, lihat [Pengidentifikasi IAM](#).

Contoh berikut membuat templat aturan persetujuan bernama **2-approver-rule-for-main** dan deskripsi **Requires two developers from the team to approve the pull request if the destination branch is main**. Templat membutuhkan dua pengguna yang memiliki peran **CodeCommitReview** untuk menyetujui permintaan pull apapun sebelum dapat digabungkan ke cabang **main**:

```
aws codecommit create-approval-rule-template --approval-rule-template-name 2-approver-rule-for-main --approval-rule-template-description "Requires two developers from the team to approve the pull request if the destination branch is main" --approval-rule-template-content "{\"Version\": \"2018-11-08\", \"DestinationReferences\": [\"refs/heads/main\"], \"Statements\": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"
```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "approvalRuleTemplate": {
    "approvalRuleTemplateName": "2-approver-rule-for-main",
    "creationDate": 1571356106.936,
    "approvalRuleTemplateId": "dd8b17fe-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"DestinationReferences\": [\"refs/heads/main\"], \"Statements\": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
    "approvalRuleTemplateDescription": "Requires two developers from the team to approve the pull request if the destination branch is main",
    "lastModifiedDate": 1571356106.936,
    "ruleContentSha256": "4711b576EXAMPLE"
  }
}
```

Kaitkan templat aturan persetujuan dengan repositori

Templat aturan persetujuan dibuat dalam Wilayah AWS, tetapi templat tidak memengaruhi repositori apa pun dalam Wilayah AWS sampai templat terkait. Untuk menerapkan templat ke satu atau lebih repositori, Anda harus mengaitkan templat dengan satu atau beberapa repositori. Anda dapat menerapkan templat tunggal untuk beberapa repositori dalam Wilayah AWS. Hal ini membantu Anda mengotomatisasi dan menstandarisasi alur kerja pengembangan di repositori Anda dengan membuat kondisi yang konsisten untuk menyetujui dan menggabungkan permintaan pull.

Anda hanya dapat mengaitkan templat aturan persetujuan dengan repositori di Wilayah AWS tempat templat aturan persetujuan dibuat.

Untuk informasi selengkapnya tentang kebijakan terkelola dan izin untuk templat aturan persetujuan, lihat [izin untuk tindakan pada templat aturan persetujuan](#) dan [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Topik

- [Mengaitkan templat aturan persetujuan \(konsol\)](#)
- [Mengaitkan templat aturan persetujuan \(AWS CLI\)](#)

Mengaitkan templat aturan persetujuan (konsol)

Anda mungkin telah mengaitkan repositori dengan templat aturan persetujuan saat membuatnya. (Langkah itu bersifat opsional.) Anda dapat menambahkan atau menghapus pengaitan dengan mengedit templat.

Untuk mengaitkan templat aturan persetujuan dengan repositori

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Pilih Templat aturan persetujuan. Pilih templat, dan kemudian pilih Edit.
3. Dalam Repositori terkait, pilih repositori dari daftar Repositori. Setiap repositori terkait muncul di bawah kotak daftar.
4. Pilih Save (Simpan). Aturan persetujuan sekarang diterapkan untuk setiap permintaan pull yang dibuat dalam repositori terkait tersebut.

Mengaitkan templat aturan persetujuan (AWS CLI)

Anda dapat menggunakan AWS CLI untuk mengaitkan templat aturan persetujuan dengan satu atau lebih repositori.

Untuk mengaitkan templat dengan satu repositori

1. Pada terminal atau baris perintah, jalankan perintah `associate-approval-rule-template-with-repository`, dengan menentukan:
 - Nama aturan persetujuan yang ingin Anda kaitkan dengan repositori.
 - Nama repositori yang akan dikaitkan dengan templat aturan persetujuan.

Sebagai contoh, untuk mengaitkan templat dengan repositori bernama `2` dengan repositori bernama `2` dengan repositori bernama `2approver-rule-for-main` dengan repositori bernama `2` dengan repositori bernama `MyDemoRepo2` dengan repositori

```
aws codecommit associate-approval-rule-template-with-repository --repository-name MyDemoRepo --approval-rule-template-name 2-approver-rule-for-main
```

2. Jika berhasil, perintah ini tidak mengembalikan apa pun.

Untuk mengaitkan templat dengan beberapa repositori

1. Pada terminal atau baris perintah, jalankan perintah `batch-associate-approval-rule-template-with-repositories`, dengan menentukan:
 - Nama aturan persetujuan yang ingin Anda kaitkan dengan repositori.
 - Nama-nama repositori untuk dikaitkan dengan templat aturan persetujuan.

Sebagai contoh, untuk mengaitkan templat aturan persetujuan bernama `2-approver-rule-for-main` dengan repositori bernama `MyDemoRepo` dan `MyOtherDemoRepo`:

```
aws codecommit batch-associate-approval-rule-template-with-repositories --repository-names "MyDemoRepo", "MyOtherDemoRepo" --approval-rule-template-name 2-approver-rule-for-main
```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "associatedRepositoryNames": [
    "MyDemoRepo",
    "MyOtherDemoRepo"
  ],
  "errors": []
}
```

Mengelola templat aturan persetujuan

Anda dapat mengelola templat aturan persetujuan dalam Wilayah AWS untuk membantu memahami bagaimana templat digunakan dan tujuan penggunaannya. Misalnya, Anda dapat mengedit nama dan deskripsi templat aturan persetujuan untuk membantu orang lain memahami tujuan mereka. Anda dapat mencantumkan semua templat aturan persetujuan di Wilayah AWS, dan mendapatkan informasi tentang isi dan struktur templat. Anda dapat meninjau templat mana yang dikaitkan dengan repositori, dan repositori mana yang dikaitkan dengan templat.

Untuk informasi selengkapnya tentang kebijakan terkelola dan izin untuk templat aturan persetujuan, lihat [izin untuk tindakan pada templat aturan persetujuan](#) dan [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Mengelola templat aturan persetujuan (konsol)

Anda dapat melihat dan mengelola konsol konsol di CodeCommit konsol konsol.

Untuk mengelola templat aturan persetujuan

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Pilih Templat aturan persetujuan untuk melihat daftar templat aturan persetujuan dalam Wilayah AWS tempat Anda masuk.

Note

Templat aturan persetujuan hanya tersedia dalam Wilayah AWS tempat templat dibuat.

3. Jika Anda ingin membuat perubahan pada templat, pilih dari daftar, dan kemudian pilih Edit.
4. Buat perubahan Anda, dan kemudian pilih Simpan.

Mengelola templat aturan persetujuan (AWS CLI)

Anda dapat mengelola templat aturan persetujuan Anda dengan perintah AWS CLI:

- [list-approval-rule-templates](#), untuk menampilkan daftar semua templat aturan persetujuan dalam Wilayah AWS
- [get-approval-rule-template](#), untuk melihat isi dari templat aturan persetujuan
- [update-approval-rule-template-content](#), untuk mengubah isi dari templat aturan persetujuan
- [update-approval-rule-template-name](#), untuk mengubah nama templat aturan persetujuan
- [update-approval-rule-template-description](#), untuk mengubah deskripsi templat aturan persetujuan
- [list-repositories-for-approval-rule-template](#), untuk melihat semua repositori yang terkait dengan templat aturan persetujuan
- [list-associated-approval-rule-templates-for-repository](#), untuk melihat semua templat aturan persetujuan yang dikaitkan dengan repositori

Untuk mencantumkan semua templat aturan persetujuan di Wilayah AWS

1. Di terminal atau baris perintah, jalankan perintah `list-approval-rule-templates`. Misalnya, untuk mencantumkan semua templat aturan persetujuan di Wilayah US East (Ohio):

```
aws codecommit list-approval-rule-templates --region us-east-2
```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "approvalRuleTemplateName": [
    "2-approver-rule-for-main",
    "1-approver-rule-for-all-pull-requests"
  ]
}
```

Untuk mendapatkan isi dari templat aturan persetujuan

1. Pada terminal atau baris perintah, jalankan perintah `get-approval-rule-template`, tentukan nama templat aturan persetujuan:

```
aws codecommit get-approval-rule-template --approval-rule-template-name 1-approver-
rule-for-all-pull-requests
```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "approvalRuleTemplate": {
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements
\": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers
\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "ruleContentSha256": "621181bbEXAMPLE",
    "lastModifiedDate": 1571356106.936,
    "creationDate": 1571356106.936,
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan",
    "approvalRuleTemplateId": "a29abb15-EXAMPLE",
    "approvalRuleTemplateDescription": "All pull requests must be approved by
one developer on the team."
  }
}
```

Untuk memperbarui isi templat aturan persetujuan

1. Pada terminal atau jendela perintah, jalankan perintah `update-approval-rule-template-content`, menentukan nama templat dan konten yang diubah. Sebagai contoh, untuk mengubah isi dari templat aturan persetujuan bernama **1-approver-rule** untuk mendefinisikan kembali grup persetujuan untuk pengguna yang memiliki peran **CodeCommitReview**:

```
aws codecommit update-approval-rule-template-content --approval-rule-template-
name 1-approver-rule --new-rule-content "{\"Version\": \"2018-11-08\",
\"DestinationReferences\": [\"refs/heads/main\"], \"Statements\": [{\"Type
\": \"Approvers\", \"NumberOfApprovalsNeeded\": 2, \"ApprovalPoolMembers\":
[\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}"
```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "approvalRuleTemplate": {
    "creationDate": 1571352720.773,
```

```

    "approvalRuleTemplateDescription": "Requires 1 approval for all pull
    requests from the CodeCommitReview pool",
    "lastModifiedDate": 1571358728.41,
    "approvalRuleTemplateId": "41de97b7-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements
    \": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers
    \": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]]}]\"",
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "ruleContentSha256": "2f6c21a5EXAMPLE",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan"
  }
}

```

Untuk memperbarui nama templat aturan persetujuan

1. Pada terminal atau jendela perintah, jalankan perintah `update-approval-rule-template-name`, menentukan nama saat ini dan nama yang ingin Anda ubah. Sebagai contoh, untuk mengubah nama templat aturan persetujuan dari **1-approver-rule** ke **1-approver-rule-for-all-pull-requests**:

```

aws codecommit update-approval-rule-template-name --old-approval-rule-template-name
  "1-approver-rule" --new-approval-rule-template-name "1-approver-rule-for-all-pull-
  requests"

```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```

{
  "approvalRuleTemplate": {
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "lastModifiedDate": 1571358241.619,
    "approvalRuleTemplateId": "41de97b7-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements
    \": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers
    \": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]]}]\"",
    "creationDate": 1571352720.773,
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Mary_Major",
    "approvalRuleTemplateDescription": "All pull requests must be approved by
    one developer on the team.",
    "ruleContentSha256": "2f6c21a5cEXAMPLE"
  }
}

```

, untuk mengubah deskripsi templat aturan persetujuan

1. Pada terminal atau baris perintah, jalankan perintah `update-approval-rule-template-description`, menentukan nama templat aturan persetujuan dan deskripsi baru:

```
aws codecommit update-approval-rule-template-description --approval-rule-template-name "1-approver-rule-for-all-pull-requests" --approval-rule-template-description "Requires 1 approval for all pull requests from the CodeCommitReview pool"
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "approvalRuleTemplate": {
    "creationDate": 1571352720.773,
    "approvalRuleTemplateDescription": "Requires 1 approval for all pull requests from the CodeCommitReview pool",
    "lastModifiedDate": 1571358728.41,
    "approvalRuleTemplateId": "41de97b7-EXAMPLE",
    "approvalRuleTemplateContent": "{\"Version\": \"2018-11-08\", \"Statements\": [{\"Type\": \"Approvers\", \"NumberOfApprovalsNeeded\": 1, \"ApprovalPoolMembers\": [\"arn:aws:sts::123456789012:assumed-role/CodeCommitReview/*\"]}]}",
    "approvalRuleTemplateName": "1-approver-rule-for-all-pull-requests",
    "ruleContentSha256": "2f6c21a5EXAMPLE",
    "lastModifiedUser": "arn:aws:iam::123456789012:user/Li_Juan"
  }
}
```

Untuk mencantumkan semua repositori yang dikaitkan dengan templat

1. Pada baris perintah atau terminal, jalankan perintah `list-repositories-for-approval-rule-template`, menentukan nama templat:

```
aws codecommit list-repositories-for-approval-rule-template --approval-rule-template-name 2-approver-rule-for-main
```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "repositoryNames": [
    "MyDemoRepo",
    "MyClonedRepo"
  ]
}
```



```
]
}
```

Untuk mencantumkan semua templat yang dikaitkan dengan repositori

1. Pada baris perintah atau terminal, jalankan perintah `list-associated-approval-rule-templates-for-repository`, menentukan nama repositori:

```
aws codecommit list-associated-approval-rule-templates-for-repository --repository-name MyDemoRepo
```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "approvalRuleTemplateName": [
    "2-approver-rule-for-main",
    "1-approver-rule-for-all-pull-requests"
  ]
}
```

Melepaskan kaitan templat aturan persetujuan

Jika aturan persetujuan yang dihasilkan oleh templat aturan persetujuan tidak lagi masuk akal untuk alur kerja tim Anda dalam repositori, Anda dapat memisahkan templat dari repositori tersebut. Memisahkan templat tidak akan menghapus aturan persetujuan apapun yang dibuat ketika templat dikaitkan dengan repositori.

Untuk informasi selengkapnya tentang kebijakan terkelola dan izin untuk templat aturan persetujuan, lihat [izin untuk tindakan pada templat aturan persetujuan](#) dan [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Melepaskan kaitan templat aturan persetujuan (konsol)

Anda dapat menggunakan konsol untuk menghapus pengaitan antara repositori dan templat aturan persetujuan.

Untuk memisahkan templat aturan persetujuan dari repositori

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.

2. Pilih Templat aturan persetujuan. Pilih templat yang ingin Anda pisahkan dari repositori atau repositori, lalu pilih Edit.
3. Dalam Repositori terkait, pilih X di samping repositori yang ingin Anda pisahkan. Nama repositori tidak lagi muncul.
4. Pilih Save (Simpan). Aturan persetujuan tidak diterapkan untuk permintaan pull yang dibuat di repositori tersebut. Aturan masih diterapkan untuk permintaan pull yang dibuat saat pengaitan masih terjadi.

Memisahkan templat aturan persetujuan (AWS CLI)

Anda dapat menggunakan AWS CLI untuk memisahkan satu atau lebih repositori dari templat aturan persetujuan.

Untuk memisahkan templat aturan persetujuan dari repositori

1. Pada terminal atau baris perintah, jalankan perintah `disassociate-approval-rule-template-from-repository`, dengan menentukan:
 - Nama templat aturan persetujuan.
 - Nama repositori.

Sebagai contoh, untuk memisahkan templat aturan persetujuan bernama **1-approver-rule-for-all-pull-requests** dari repositori bernama **MyDemoRepo**:

```
aws codecommit disassociate-approval-rule-template-from-repository --repository-name MyDemoRepo --approval-rule-template-name 1-approver-rule-for-all-pull-requests
```

2. Jika berhasil, perintah ini tidak mengembalikan apa pun.

Untuk memisahkan templat aturan persetujuan dari beberapa repositori

1. Pada terminal atau baris perintah, jalankan perintah `batch-disassociate-approval-rule-template-from-repositories`, dengan menentukan:
 - Nama templat aturan persetujuan.
 - Nama-nama repositori.

Sebagai contoh, untuk memisahkan templat aturan persetujuan bernama **1-approver-rule-for-all-pull-requests** dari repositori bernama **MyDemoRepo** dan repositori bernama **MyOtherDemoRepo**:

```
aws codecommit batch-disassociate-approval-rule-template-from-repositories --
repository-names "MyDemoRepo", "MyOtherDemoRepo" --approval-rule-template-name 1-
approver-rule-for-all-pull-requests
```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "disassociatedRepositoryNames": [
    "MyDemoRepo",
    "MyOtherDemoRepo"
  ],
  "errors": []
}
```

Menghapus templat aturan persetujuan

Anda dapat menghapus templat aturan persetujuan jika Anda tidak menggunakannya dalam repositori. Menghapus templat aturan persetujuan yang tidak digunakan membantu menjaga templat Anda terorganisir dan membuatnya lebih mudah untuk menemukan templat yang masuk akal untuk alur kerja Anda.

Untuk informasi selengkapnya tentang kebijakan terkelola dan izin untuk templat aturan persetujuan, lihat [izin untuk tindakan pada templat aturan persetujuan](#) dan [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Topik

- [Menghapus templat aturan persetujuan \(konsol\)](#)
- [Menghapus templat aturan persetujuan \(AWS CLI\)](#)

Menghapus templat aturan persetujuan (konsol)

Anda dapat menghapus templat aturan persetujuan jika tidak lagi relevan dengan pekerjaan pengembangan Anda. Ketika Anda menggunakan konsol untuk menghapus templat aturan persetujuan, templat dipisahkan dari repositori apapun selama proses penghapusan.

Untuk menghapus templat aturan persetujuan

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Pilih Templat aturan persetujuan. Pilih templat yang ingin Anda hapus, dan pilih Hapus.

Menghapus templat aturan persetujuan (AWS CLI)

Anda dapat menggunakan AWS CLI untuk menghapus aturan persetujuan jika telah dipisahkan dari semua repositori. Untuk informasi selengkapnya, lihat [Memisahkan templat aturan persetujuan \(AWS CLI\)](#).

Untuk menghapus templat aturan persetujuan

1. Pada terminal atau baris perintah, jalankan `delete-approval-rule-template`, menentukan nama templat aturan persetujuan yang ingin Anda hapus:

```
aws codecommit delete-approval-rule-template --approval-rule-template-name 1-approver-for-all-pull-requests
```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut. Jika templat aturan persetujuan telah dihapus, perintah ini tidak mengembalikan apa pun.

```
{
  "approvalRuleTemplateId": "41de97b7-EXAMPLE"
}
```

Bekerja dengan komit di repositori AWS CodeCommit

Commit adalah snapshot dari isi dan perubahan isi repositori Anda. Setiap kali pengguna melakukan dan mendorong perubahan, informasi yang di-save dan disimpan. Demikian pula, informasi yang mencakup siapa yang melakukan perubahan, tanggal dan waktu commit, dan perubahan yang dibuat sebagai bagian dari commit tersebut. Anda juga dapat menambahkan tag ke commit, untuk dengan mudah mengidentifikasi commit tertentu. Di CodeCommit, Anda dapat:

- Meninjau commit.
- Menampilkan sejarah commit dalam sebuah grafik.
- Membandingkan sebuah commit terhadap induknya atau penentu lain.
- Menambahkan komentar ke commit Anda dan balas komentar yang dibuat oleh orang lain.

The screenshot displays a diff view for the file `ahs_count.py`. The diff shows a change on line 7: the original code (left) has `- alv = "Number of alveolar hissing sibilants: {}"`, and the new code (right) has `+ ahs = "Number of alveolar hissing sibilants: {}"`. A comment box is open below the diff, containing the text: "You've reverted to the old value here, which won't work. This should remain `alv`." The comment box has "Save" and "Cancel" buttons. The interface also includes buttons for "Browse file contents" and "Comment on file".

Sebelum Anda dapat mendorong commit ke CodeCommit repositori, Anda harus mengatur komputer lokal Anda untuk terhubung ke repositori. Untuk metode yang paling sederhana, lihat [Untuk pengguna HTTPS menggunakan kredensial Git](#).

Untuk informasi tentang bekerja dengan aspek lain dari repositori Anda di CodeCommit, lihat, [Bekerja dengan repositori](#), [Bekerja dengan file](#) [Bekerja dengan permintaan pull](#) [Bekerja dengan cabang](#), dan [Bekerja dengan preferensi pengguna](#)

Topik

- [Buat komit di AWS CodeCommit](#)
- [Lihat detail komit di AWS CodeCommit](#)
- [Bandingkan komit di AWS CodeCommit](#)
- [Komentari komit di AWS CodeCommit](#)
- [Buat tag Git di AWS CodeCommit](#)
- [Lihat detail tag Git di AWS CodeCommit](#)
- [Hapus tag Git di AWS CodeCommit](#)

Buat komit di AWS CodeCommit

Saat Anda membuat komit pertama untuk repositori baru, Anda menggunakan perintah AWS CLI dan perintah `put-file`. Ini membuat commit pertama dan memungkinkan Anda untuk membuat dan menentukan cabang default untuk repositori baru Anda. Anda dapat menggunakan Git atau AWS CLI untuk membuat komit dalam CodeCommit repositori. Jika repo lokal terhubung ke CodeCommit repositori, Anda menggunakan Git untuk mendorong komit dari repo lokal ke repositori. CodeCommit Untuk membuat komit langsung di CodeCommit konsol, lihat [Membuat atau menambahkan file ke repositori AWS CodeCommit](#) dan [Mengedit isi file dalam repositori AWS CodeCommit](#).

Note

Sebagai praktik terbaik, kami menyarankan Anda menggunakan versi terbaru yang didukung dari AWS CLI, Git, dan perangkat lunak lainnya. Jika Anda menggunakan AWS CLI, pastikan Anda memiliki versi terbaru yang diinstal untuk memastikan bahwa Anda menggunakan versi yang berisi `create-commit` perintah.

Topik

- [Buat komit pertama untuk repositori menggunakan AWS CLI](#)
- [Buatlah sebuah commit menggunakan klien Git](#)
- [Buat komit menggunakan AWS CLI](#)

Buat komit pertama untuk repositori menggunakan AWS CLI

Anda dapat menggunakan `put-file` perintah AWS CLI dan untuk membuat komit pertama Anda untuk repositori. Menggunakan `put-file` membuat commit pertama yang menambahkan file ke

repositori kosong Anda, dan membuat cabang dengan nama yang Anda tentukan. Ini menunjuk cabang baru sebagai cabang default untuk repositori Anda.

Note

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal file AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk membuat komit pertama untuk repositori menggunakan AWS CLI

1. Di komputer lokal Anda, buat file yang ingin Anda tambahkan sebagai file pertama ke CodeCommit repositori. Praktik yang umum adalah untuk menciptakan sebuah file markdown README .md yang menjelaskan tujuan dari repositori ini untuk pengguna repositori lainnya. Jika Anda menyertakan README .md file, konten file ditampilkan secara otomatis di bagian bawah halaman Kode untuk repositori Anda di CodeCommit konsol.
2. Pada terminal atau baris perintah, jalankan perintah put-file, dengan menentukan:
 - Nama repositori tempat Anda ingin menambahkan file pertama.
 - Nama cabang yang ingin Anda buat sebagai cabang default.
 - Lokasi lokal dari file tersebut. Sintaks yang digunakan untuk lokasi ini bervariasi, tergantung pada sistem operasi lokal Anda.
 - Nama file yang ingin Anda tambahkan, termasuk jalur di mana file yang diperbarui disimpan dalam repositori.
 - Nama pengguna dan email yang ingin Anda kaitkan dengan file ini.
 - Sebuah pesan commit yang menjelaskan mengapa Anda menambahkan file ini.

Nama pengguna, alamat email, dan pesan commit bersifat opsional, tetapi dapat membantu pengguna lain mengetahui siapa yang membuat perubahan dan mengapa. Jika Anda tidak memberikan nama pengguna, CodeCommit default menggunakan nama pengguna IAM Anda atau turunan dari login konsol Anda sebagai nama penulis.

Sebagai contoh, untuk menambahkan file yang bernama *README.md* dengan isi "Selamat datang di repositori tim kami!" *ke repositori bernama MyDemoRepoke cabang bernama development:*

```
aws codecommit put-file --repository-name MyDemoRepo --branch-name development --file-path README.md --file-content "Welcome to our team repository!" --name "Mary Major" --email "mary_major@example.com" --commit-message "I added a quick readme for our new team repository."
```

Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "commitId": "724caa36EXAMPLE",
  "blobId": "a8a94062EXAMPLE",
  "treeId": "08b2fc73EXAMPLE"
}
```

Buatlah sebuah commit menggunakan klien Git

Anda dapat membuat commit menggunakan klien Git yang diinstal pada komputer lokal Anda, dan kemudian mendorong commit tersebut ke repositori Anda CodeCommit .

1. Lengkapi prasyaratnya, termasuk [Pengaturan](#) .

Important

Jika Anda belum menyelesaikan penyiapan, Anda tidak dapat menghubungkan atau melakukan commit ke repositori menggunakan Git.

2. Pastikan Anda membuat sebuah commit di cabang yang benar. Untuk menampilkan daftar cabang yang tersedia dan mencari tahu cabang mana yang sedang Anda atur untuk digunakan, jalankan `git branch`. Semua cabang ditampilkan. Tanda bintang (*) muncul di sebelah cabang Anda saat ini. Untuk beralih ke cabang lain, jalankan `git checkout branch-name`. Jika ini adalah commit pertama Anda, jalankan perintah `git config` untuk mengkonfigurasi klien Git Anda untuk membuat cabang awal dengan nama yang ingin Anda gunakan untuk cabang itu. Misalnya, jika Anda ingin cabang default Anda memiliki nama *Pengembangan*:

```
git config --local init.defaultBranch development
```


i Tip

Perintah ini hanya tersedia di Git v.2.28 dan lebih baru.

Anda juga dapat menjalankan perintah ini untuk mengatur nama cabang default Anda untuk **development** untuk semua repositori yang baru dibuat:

```
git config --global init.defaultBranch development
```

3. Buatlah sebuah perubahan pada cabang (seperti menambahkan, memodifikasi, atau menghapus file).

Misalnya, dalam repo lokal, buatlah sebuah file dengan nama `bird.txt` dengan teks berikut:

```
bird.txt
-----
Birds (class Aves or clade Avialae) are feathered, winged, two-legged, warm-
blooded, egg-laying vertebrates.
```

4. Jalankan `git status`, yang seharusnya menunjukkan bahwa `bird.txt` belum dimasukkan dalam commit yang tertunda apa pun:

```
...
Untracked files:
  (use "git add <file>..." to include in what will be committed)

    bird.txt
```

5. Jalankan `git add bird.txt` untuk menyertakan file baru dalam commit yang tertunda.
6. Saat Anda menjalankan `git status` lagi, Anda akan melihat output yang serupa dengan yang berikut. Ini menunjukkan bahwa `bird.txt` sekarang menjadi bagian dari commit yang tertunda atau yang diadakan untuk commit:

```
...
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   bird.txt
```

- Untuk menyelesaikan commit tersebut, jalankan git commit dengan opsi -m (misalnya, git commit -m "**Adding bird.txt to the repository.**") Opsi -m membuat pesan commit.
- Saat Anda menjalankan git status lagi, Anda akan melihat output yang serupa dengan yang berikut. Ini menunjukkan bahwa komit siap didorong dari repo lokal ke CodeCommit repositori:

```
...
nothing to commit, working directory clean
```

- Sebelum Anda mendorong komit yang telah diselesaikan dari repo lokal ke CodeCommit repositori, Anda dapat melihat apa yang Anda dorong dengan menjalankangit diff --stat remote-name/branch-name, di mana nama jarak jauh adalah nama panggilan yang digunakan repo lokal untuk CodeCommit repositori dan nama cabang adalah nama cabang untuk membandingkan.*

Tip

Untuk mendapatkan julukan, jalankan git remote. Untuk mendapatkan daftar nama cabang, jalankan git branch. Tanda bintang (*) muncul di sebelah cabang saat ini. Anda juga dapat menjalankan git status untuk mendapatkan nama cabang saat ini.

Note

Jika Anda mengkloning repositori, dari perspektif repo lokal, nama *jarak jauh bukanlah nama repositori*. CodeCommit Saat Anda mengkloning repositori, *Nama jarak jauh* diatur secara otomatis ke origin.

Misalnya, git diff --stat origin/main akan menampilkan output yang serupa dengan berikut ini:

```
bird.txt | 1 +
1 file changed, 1 insertion(+)
```

Outputnya mengasumsikan Anda telah menghubungkan repo lokal ke repositori. CodeCommit (Untuk instruksi, lihat [Connect ke sebuah repositori](#).)

- Saat Anda siap untuk mendorong komit dari repo lokal ke CodeCommit repositori, jalankangit push **remote-name branch-name**, di mana nama *jarak jauh adalah nama panggilan*

yang digunakan repo lokal untuk repositori dan nama cabang adalah *nama cabang* untuk mendorong ke CodeCommit repositori. CodeCommit

Sebagai contoh, menjalankan git push origin main akan menampilkan output yang serupa dengan berikut ini:

Untuk HTTPS:

```
Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 516 bytes | 0 bytes/s, done.
Total 5 (delta 2), reused 0 (delta 0)
remote:
To https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
    b9e7aa6..3dbf4dd main -> main
```

Untuk SSH:

```
Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 516 bytes | 0 bytes/s, done.
Total 5 (delta 2), reused 0 (delta 0)
remote:
To ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo
    b9e7aa6..3dbf4dd main -> main
```

Tip

Jika Anda menambahkan opsi `-u` ke git push (misalnya, `git push -u origin main`), maka anda hanya perlu menjalankan git push di masa depan karena informasi pelacakan hulu telah ditetapkan. Untuk mendapatkan informasi pelacakan hulu, jalankan `git remote show remote-name` (sebagai contoh, `git remote show origin`).

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Buat komit menggunakan AWS CLI

Anda dapat menggunakan `create-commit` perintah AWS CLI dan untuk membuat komit untuk repositori di ujung cabang tertentu. Anda juga dapat membuat commit gabungan yang tidak direferensikan untuk mewakili hasil penggabungan dua penentu commit. Untuk informasi selengkapnya, lihat [Buatlah sebuah commit yang tidak direferensikan](#).

Note

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal file AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk membuat sebuah commit

1. Di komputer lokal Anda, buat perubahan yang ingin Anda komit ke CodeCommit repositori.
2. Pada terminal atau baris perintah, jalankan perintah `create-commit`, dengan menentukan:
 - Repositori di mana Anda ingin melakukan perubahan tersebut.
 - Cabang di mana Anda ingin melakukan perubahan tersebut.
 - ID commit penuh dari commit terbaru dibuat ke cabang itu, yang juga dikenal sebagai commit ujung atau kepala atau ID commit induk.
 - Apakah untuk menyimpan folder kosong jika perubahan yang Anda buat hapus isi folder tersebut. Secara default, nilai ini adalah salah.
 - Informasi tentang file yang ingin ditambahkan, diubah, atau dihapus.
 - Nama pengguna dan email yang Anda ingin kaitkan dengan perubahan ini.
 - Sebuah pesan commit yang menjelaskan mengapa Anda membuat perubahan ini.

Nama pengguna, alamat email, dan pesan commit adalah opsional, tetapi membantu pengguna lain mengetahui siapa yang membuat perubahan dan mengapa. Jika Anda tidak memberikan nama pengguna, CodeCommit default menggunakan nama pengguna IAM Anda atau turunan dari login konsol Anda sebagai nama penulis.

Misalnya, untuk membuat komit untuk repositori yang menambahkan README.md file ke repositori bernama MyDemoRepod cabang utama. Isi file ada di Base64 dan berbunyi "Selamat datang di repositori tim kami!" :

```
aws codecommit create-commit --repository-name MyDemoRepo --  
branch-name main --parent-commit-id 4c925148EXAMPLE --put-files  
"filePath=README.md,fileContent=V2VsY29tZSB0byBvdXIgdGVhbSBvZXBvc2l0b3J5IQo="
```

i Tip

Untuk mendapatkan ID commit induk, jalankan perintah [get-branch](#).

Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{  
  "commitId": "4df8b524-EXAMPLE",  
  "treeId": "55b57003-EXAMPLE",  
  "filesAdded": [  
    {  
      "blobId": "5e1c309dEXAMPLE",  
      "absolutePath": "meeting.md",  
      "fileMode": "NORMAL"  
    }  
  ],  
  "filesDeleted": [],  
  "filesUpdated": []  
}
```

Untuk membuat komit yang membuat perubahan pada file bernama file1.py dan file2.txt, ganti nama file dari picture.png menjadi image1.png dan memindahkannya dari direktori bernama pictures ke direktori bernama, images, dan menghapus file bernama ExampleSolution.py dalam repositori bernama MyDemoRepo pada cabang MyFeatureBranch bernama yang komit terbarunya memiliki ID 4C925148Example:

```
aws codecommit create-commit --repository-name MyDemoRepo --branch-  
name MyFeatureBranch --parent-commit-id 4c925148EXAMPLE --name "Saanvi Sarkar"  
--email "saanvi_sarkar@example.com" --commit-message "I'm creating this commit to  
update a variable name in a number of files."  
--keep-empty-folders false --put-files '{"filePath": "file1.py", "fileMode":  
"EXECUTABLE", "fileContent": "bucket_name = sys.argv[1] region = sys.argv[2]"}'
```

```
'{"filePath": "file2.txt", "fileMode": "NORMAL", "fileContent": "//Adding a comment to explain the variable changes in file1.py"}' '{"filePath": "images/image1.png", "fileMode": "NORMAL", "sourceFile": {"filePath": "pictures/picture.png", "isMove": true}}' --delete-files filePath="ExampleSolution.py"
```

Note

Sintaks untuk segmen `--put-files` bervariasi tergantung pada sistem operasi Anda. Contoh di atas dioptimalkan untuk pengguna Linux, MacOS, atau Unix dan pengguna Windows dengan emulator Bash. Pengguna Windows pada baris perintah atau di Powershell harus menggunakan sintaks yang sesuai untuk sistem tersebut.

Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{
  "commitId": "317f8570EXAMPLE",
  "treeId": "347a3408EXAMPLE",
  "filesAdded": [
    {
      "absolutePath": "images/image1.png",
      "blobId": "d68ba6ccEXAMPLE",
      "fileMode": "NORMAL"
    }
  ],
  "filesUpdated": [
    {
      "absolutePath": "file1.py",
      "blobId": "0a4d55a8EXAMPLE",
      "fileMode": "EXECUTABLE"
    },
    {
      "absolutePath": "file2.txt",
      "blobId": "915766bbEXAMPLE",
      "fileMode": "NORMAL"
    }
  ],
  "filesDeleted": [
    {
      "absolutePath": "ExampleSolution.py",
      "blobId": "4f9cebe6aEXAMPLE",

```

```
    "fileMode": "EXECUTABLE"
  },
  {
    "absolutePath": "pictures/picture.png",
    "blobId": "fb12a539EXAMPLE",
    "fileMode": "NORMAL"
  }
]
}
```

Lihat detail komit di AWS CodeCommit

Anda dapat menggunakan AWS CodeCommit konsol untuk menelusuri riwayat komit di repositori. Ini dapat membantu Anda mengidentifikasi perubahan yang dibuat dalam repositori, termasuk:

- Kapan dan oleh siapa perubahan itu dibuat.
- Ketika commit spesifik digabung menjadi cabang.

Menampilkan riwayat commit untuk cabang juga dapat membantu Anda memahami perbedaan antara cabang. Jika Anda menggunakan penandaan, Anda juga dapat dengan cepat melihat commit yang diberi label dengan tag dan induk dari komit yang ditandai tersebut. Pada baris perintah, Anda dapat menggunakan Git untuk melihat detail tentang komit dalam repo lokal atau repositori. CodeCommit

Telusuri commit dalam sebuah repositori

Anda dapat menggunakan AWS CodeCommit konsol untuk menelusuri riwayat komit ke repositori. Anda juga dapat menampilkan grafik dari commit di repositori dan cabang-cabangnya dari waktu ke waktu. Ini dapat membantu Anda memahami riwayat dari repositori tersebut, termasuk saat perubahan dilakukan.

Note

Menggunakan git rebase untuk merebase sebuah repositori mengubah riwayat repositori, yang mungkin menyebabkan commit muncul rusak. Untuk informasi selengkapnya, lihat [Git Percabangan-Rebasing](#) atau dokumentasi Git Anda.

Topik

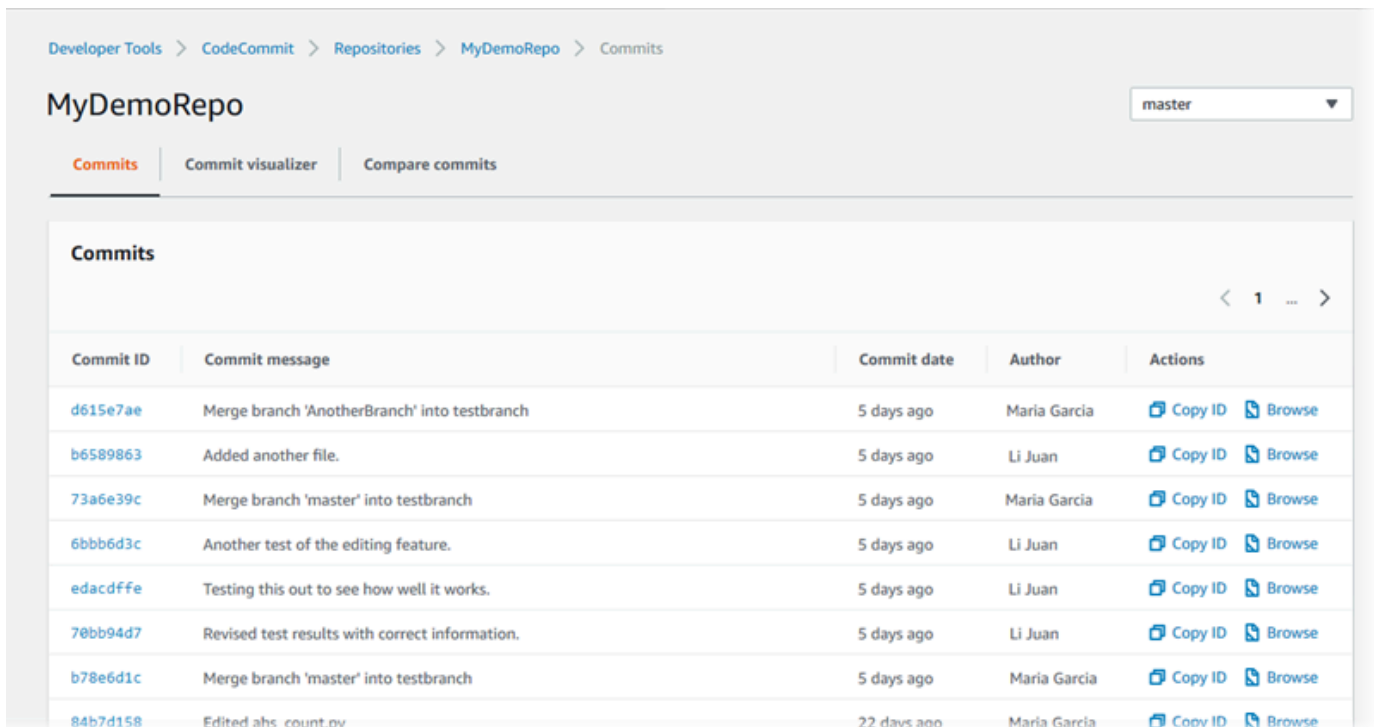
- [Telusuri riwayat commit dari sebuah repositori](#)
- [Tampilkan sebuah grafik riwayat commit dari sebuah repositori](#)

Telusuri riwayat commit dari sebuah repositori

Anda dapat menelusuri riwayat komit untuk cabang tertentu atau tag dari repositori, termasuk informasi tentang committer dan pesan commit. Anda juga dapat menampilkan kode untuk sebuah commit.

Untuk menelusuri riwayat commit

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori yang ingin Anda tinjau riwayat commit-nya.
3. Di panel navigasi, pilih Commit. Dalam tampilan riwayat commit, riwayat commit untuk repositori di cabang default ditampilkan, dalam urutan kronologis terbalik dari tanggal commit tersebut. Tanggal dan waktu berada dalam Waktu Universal Terkoordinasi (UTC). Anda dapat melihat riwayat commit dari cabang yang berbeda dengan memilih tombol pemilih tampilan dan kemudian memilih cabang dari daftar. Jika Anda menggunakan tag dalam repositori Anda, Anda dapat menampilkan sebuah commit dengan tag tertentu dan induknya dengan memilih tag tersebut di tombol pemilih tampilan.



Commit ID	Commit message	Commit date	Author	Actions
d615e7ae	Merge branch 'AnotherBranch' into testbranch	5 days ago	Maria Garcia	Copy ID Browse
b6589863	Added another file.	5 days ago	Li Juan	Copy ID Browse
73a6e39c	Merge branch 'master' into testbranch	5 days ago	Maria Garcia	Copy ID Browse
6bbb6d3c	Another test of the editing feature.	5 days ago	Li Juan	Copy ID Browse
edacdffe	Testing this out to see how well it works.	5 days ago	Li Juan	Copy ID Browse
70bb94d7	Revised test results with correct information.	5 days ago	Li Juan	Copy ID Browse
b78e6d1c	Merge branch 'master' into testbranch	5 days ago	Maria Garcia	Copy ID Browse
84b7d158	Edited ahs_count.py	22 days ago	Maria Garcia	Copy ID Browse

4. Untuk menampilkan perbedaan antara sebuah commit dan induknya, dan Untuk menampilkan komentar tentang perubahan, pilih ID commit yang disingkat. Untuk informasi selengkapnya, lihat [Bandingkan commit dengan induknya](#) dan [Komentari sebuah commit](#). Untuk menampilkan perbedaan antara commit dan penentu commit lainnya, termasuk cabang, tag, atau ID commit, lihat [Bandingkan dua penentu commit mana pun](#).
5. Lakukan salah satu atau beberapa hal berikut:
 - Untuk menampilkan tanggal dan waktu perubahan dibuat, arahkan kursor ke tanggal commit.
 - Untuk menampilkan ID commit penuh, salin dan tempelkan ke editor teks atau lokasi lainnya. Untuk menyalinnya, pilih Salin ID.
 - Untuk menampilkan kode seperti apa adanya pada saat suatu commit, pilih Browse. Isi dari repositori seperti apa adanya pada saat commit tersebut ditampilkan dalam tampilan Kode. Tombol pemilih tampilan menampilkan ID commit yang disingkat alih-laih cabang atau tag.

Tampilkan sebuah grafik riwayat commit dari sebuah repositori

Anda dapat menampilkan grafik commit yang dibuat ke sebuah repositori. Tampilan Pemvisualisasi Commit adalah representasi grafik asiklik terarah (DAG) dari semua commit yang dibuat ke cabang repositori. Representasi grafis ini dapat membantu Anda memahami kapan commit dan fitur terkait ditambahkan atau digabungkan. Ini juga dapat membantu Anda melakukan pinpoint kapan perubahan dibuat dalam kaitannya dengan perubahan lainnya.

Note

Commit yang digabung menggunakan metode fast-forward tidak muncul sebagai baris terpisah dalam grafik commit.

Untuk menampilkan grafik commit

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori yang ingin Anda lihat grafik commit.
3. Di panel navigasi, pilih Commit, dan kemudian pilih tab Pemvisualisasi commit.

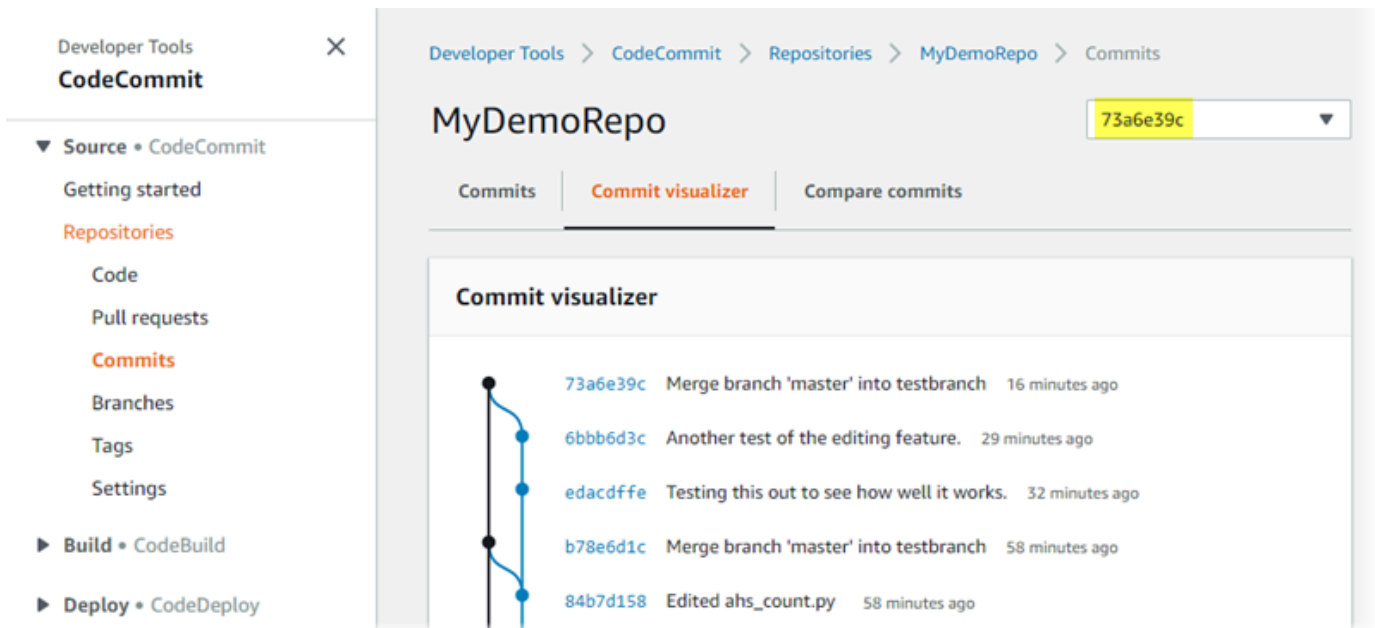
Dalam grafik commit, ID commit yang disingkat dan subjek untuk setiap pesan commit muncul di sebelah titik itu dalam grafik.

Note

Grafik dapat menampilkan hingga 35 cabang pada satu halaman. Jika ada lebih dari 35 cabang, maka grafik terlalu rumit untuk ditampilkan. Anda dapat menyederhanakan tampilan dengan dua cara:

- Dengan menggunakan tombol pemilih tampilan untuk menampilkan grafik untuk cabang tertentu.
- Dengan menempelkan ID commit penuh ke dalam kotak pencarian untuk merender grafik dari commit itu.

4. Untuk membuat grafik baru dari sebuah commit, pilih titik dalam grafik yang sesuai dengan commit itu. Tombol pemilih tampilan berubah menjadi ID commit yang disingkat.



Tampilkan detail commit (AWS CLI)

Git memungkinkan Anda melihat detail tentang commit. Anda juga dapat menggunakan AWS CLI untuk melihat detail tentang komit di repo lokal atau di CodeCommit repositori dengan menjalankan perintah berikut:

- Untuk menampilkan informasi tentang commit, jalankan [aws codecommit get-commit](#).
- Untuk menampilkan informasi tentang beberapa commit, jalankan [aws codecommit batch-get-commits](#).
- Untuk menampilkan informasi tentang commit gabungan, jalankan [aws codecommit get-merge-commit](#).
- Untuk menampilkan informasi tentang perubahan untuk penentu commit (cabang, tag, HEAD, atau referensi yang memenuhi syarat lainnya, seperti ID commit), jalankan [aws codecommit get-differences](#).
- Untuk menampilkan isi terkode base64 dari objek gumpalan Git dalam repositori, jalankan [aws codecommit get-blob](#).

Untuk menampilkan informasi tentang sebuah commit

1. Jalankan perintah `aws codecommit get-commit`, dengan menentukan:

- Nama CodeCommit repositori (dengan `--repository-name` opsi).
- ID commit penuh.

Misalnya, untuk melihat informasi tentang komit dengan ID `317f8570EXAMPLE` dalam CodeCommit repositori bernama: `MyDemoRepo`

```
aws codecommit get-commit --repository-name MyDemoRepo --commit-id
317f8570EXAMPLE
```

2. Jika berhasil, output dari perintah ini meliputi hal berikut:

- Informasi tentang penulis commit tersebut (seperti yang dikonfigurasi dalam Git), termasuk tanggal dalam format timestamp dan offset waktu universal terkoordinasi (UTC).
- Informasi tentang committer (seperti yang dikonfigurasi dalam Git) termasuk tanggal dalam format timestamp dan offset UTC.
- ID pohon Git di mana commit tersebut ada.
- ID commit dari commit induk.
- Pesan commit.

Berikut adalah beberapa contoh output berdasarkan perintah sebelumnya:

```
{
  "commit": {
    "additionalData": "",
    "committer": {
      "date": "1484167798 -0800",
      "name": "Mary Major",
      "email": "mary_major@example.com"
    },
    "author": {
      "date": "1484167798 -0800",
      "name": "Mary Major",
      "email": "mary_major@example.com"
    },
    "treeId": "347a3408EXAMPLE",
    "parents": [
      "4c925148EXAMPLE"
    ],
  },
}
```

```
    "message": "Fix incorrect variable name"
  }
}
```

Untuk menampilkan informasi tentang commit gabungan

1. Jalankan perintah `get-merge-commit`, dengan menentukan:

- Sebuah penentu commit untuk sumber penggabungan (dengan opsi `--source-commit-specifier`).
- Sebuah penentu commit untuk tujuan untuk penggabungan (dengan opsi `--destination-commit-specifier`).
- Opsi gabungan yang ingin Anda gunakan (dengan opsi `--merge-option`).
- Nama dari repositori tersebut (dengan opsi `--repository-name`).

Misalnya, untuk melihat informasi tentang komit gabungan untuk cabang sumber bernama `bugfix-bug1234` dengan cabang tujuan bernama `main` menggunakan strategi `THREE_WAY_MERGE` dalam repositori bernama: `MyDemoRepo`

```
aws codecommit get-merge-commit --source-commit-specifier bugfix-bug1234 --
destination-commit-specifier main --merge-option THREE_WAY_MERGE --repository-
name MyDemoRepo
```

2. Jika berhasil, output ini menampilkan informasi yang serupa dengan yang berikut:

```
{
  "sourceCommitId": "c5709475EXAMPLE",
  "destinationCommitId": "317f8570EXAMPLE",
  "baseCommitId": "fb12a539EXAMPLE",
  "mergeCommitId": "ffc4d608eEXAMPLE"
}
```

Untuk menampilkan informasi tentang beberapa commit

1. Jalankan perintah `batch-get-commits`, dengan menentukan:

- Nama CodeCommit repositori (dengan `--repository-name` opsi).
- Daftar ID commit penuh untuk setiap commit yang Anda ingin lihat informasinya.

Misalnya, untuk melihat informasi tentang komit dengan ID `317f8570EXAMPLE` dan `4c925148EXAMPLE` dalam CodeCommit repositori bernama: `MyDemoRepo`

```
aws codecommit batch-get-commits --repository-name MyDemoRepo --commit-ids
317f8570EXAMPLE 4c925148EXAMPLE
```

2. Jika berhasil, output dari perintah ini meliputi hal berikut:

- Informasi tentang penulis commit (seperti dikonfigurasi dalam Git), termasuk tanggal dalam format timestamp dan offset waktu universal terkoordinasi (UTC).
- Informasi tentang committer (seperti yang dikonfigurasi dalam Git) termasuk tanggal dalam format timestamp dan offset UTC.
- ID pohon Git di mana commit tersebut ada.
- ID commit dari commit induk.
- Pesan commit.

Berikut adalah beberapa contoh output berdasarkan perintah sebelumnya:

```
{
  "commits": [
    {
      "additionalData": "",
      "committer": {
        "date": "1508280564 -0800",
        "name": "Mary Major",
        "email": "mary_major@example.com"
      },
      "author": {
        "date": "1508280564 -0800",
        "name": "Mary Major",
        "email": "mary_major@example.com"
      },
      "commitId": "317f8570EXAMPLE",
      "treeId": "1f330709EXAMPLE",
      "parents": [
```

```
        "6e147360EXAMPLE"
    ],
    "message": "Change variable name and add new response element"
  },
  {
    "additionalData": "",
    "committer": {
      "date": "1508280542 -0800",
      "name": "Li Juan",
      "email": "li_juan@example.com"
    },
    "author": {
      "date": "1508280542 -0800",
      "name": "Li Juan",
      "email": "li_juan@example.com"
    },
    "commitId": "4c925148EXAMPLE",
    "treeId": "1f330709EXAMPLE",
    "parents": [
      "317f8570EXAMPLE"
    ],
    "message": "Added new class"
  }
}
```

Untuk menampilkan informasi tentang perubahan untuk penentu commit

1. Jalankan perintah `aws codecommit get-differences`, dengan menentukan:

- Nama CodeCommit repositori (dengan `--repository-name` opsi).
- penentu commit yang Anda ingin dapatkan informasi tentangnya. Hanya `--after-commit-specifier` disyaratkan. Jika Anda tidak menentukan `--before-commit-specifier`, semua file saat ini sebagai `--after-commit-specifier` ditampilkan.

Misalnya, untuk melihat informasi tentang perbedaan antara komit dengan ID `317f8570EXAMPLE` dan `4c925148EXAMPLE` dalam CodeCommit repositori bernama: `MyDemoRepo`

```
aws codecommit get-differences --repository-name MyDemoRepo --before-commit-specifier 317f8570EXAMPLE --after-commit-specifier 4c925148EXAMPLE
```

2. Jika berhasil, output dari perintah ini meliputi hal berikut:

- Daftar perbedaan, termasuk jenis perubahan (A untuk ditambahkan, D untuk dihapus, atau M untuk dimodifikasi).
- Mode jenis perubahan file.
- ID dari objek gumpalan Git yang berisi perubahan.

Berikut adalah beberapa contoh output berdasarkan perintah sebelumnya:

```
{
  "differences": [
    {
      "afterBlob": {
        "path": "blob.txt",
        "blobId": "2eb4af3bEXAMPLE",
        "mode": "100644"
      },
      "changeType": "M",
      "beforeBlob": {
        "path": "blob.txt",
        "blobId": "bf7fcf28fEXAMPLE",
        "mode": "100644"
      }
    }
  ]
}
```

Untuk menampilkan informasi tentang objek gumpalan Git

1. Jalankan perintah `aws codecommit get-blob`, dengan menentukan:

- Nama CodeCommit repositori (dengan `--repository-name` opsi).
- ID dari gumpalan Git (dengan opsi `--blob-id`).

Misalnya, untuk melihat informasi tentang gumpalan Git dengan ID dari 2eb4af3bEXAMPLE dalam CodeCommit repositori bernama: MyDemoRepo

```
aws codecommit get-blob --repository-name MyDemoRepo --blob-id 2eb4af3bEXAMPLE
```

2. Jika berhasil, output dari perintah ini meliputi hal berikut:

- Konten terkode base64 dari gumpalan, biasanya sebuah file.

Misalnya, output dari perintah sebelumnya mungkin serupa dengan yang berikut ini:

```
{
  "content": "QSBCaW5hcnkgTGFyToEXAMPLE="
}
```

Tampilkan detail commit (Git)

Sebelum Anda mengikuti langkah-langkah ini, Anda seharusnya sudah menghubungkan repo lokal ke CodeCommit repositori dan melakukan perubahan. Untuk petunjuk, lihat [Connect ke sebuah repositori](#).

Untuk menampilkan perubahan untuk commit terbaru ke repositori, jalankan perintah git show.

```
git show
```

Perintah harus menghasilkan output yang serupa dengan yang berikut ini:

```
commit 4f8c6f9d
Author: Mary Major <mary.major@example.com>
Date:   Mon May 23 15:56:48 2016 -0700

    Added bumblebee.txt

diff --git a/bumblebee.txt b/bumblebee.txt
new file mode 100644
index 0000000..443b974
--- /dev/null
+++ b/bumblebee.txt
```

```
@@ -0,0 +1 @@
+A bumblebee, also written bumble bee, is a member of the bee genus Bombus, in the
  family Apidae.
\ No newline at end of file
```

Note

Dalam contoh ini dan contoh berikut, ID commit telah disingkat. ID commit penuh tidak ditampilkan.

Untuk menampilkan perubahan yang terjadi, gunakan perintah git show dengan ID commit:

```
git show 94ba1e60

commit 94ba1e60
Author: John Doe <johndoe@example.com>
Date:   Mon May 23 15:39:14 2016 -0700

    Added horse.txt

diff --git a/horse.txt b/horse.txt
new file mode 100644
index 0000000..080f68f
--- /dev/null
+++ b/horse.txt
@@ -0,0 +1 @@
+The horse (Equus ferus caballus) is one of two extant subspecies of Equus ferus.
```

Untuk menampilkan perbedaan antara dua commit, jalankan perintah git diff dan sertakan dua ID commit.

```
git diff ce22850d 4f8c6f9d
```

Dalam contoh ini, perbedaan antara dua commit adalah bahwa dua file ditambahkan. Perintah harus menghasilkan output yang serupa dengan yang berikut ini:

```
diff --git a/bees.txt b/bees.txt
new file mode 100644
index 0000000..cf57550
```

```
--- /dev/null
+++ b/bees.txt
@@ -0,0 +1 @@
+Bees are flying insects closely related to wasps and ants, and are known for their
  role in pollination and for producing honey and beeswax.
diff --git a/bumblebee.txt b/bumblebee.txt
new file mode 100644
index 0000000..443b974
--- /dev/null
+++ b/bumblebee.txt
@@ -0,0 +1 @@
+A bumblebee, also written bumble bee, is a member of the bee genus Bombus, in the
  family Apidae.
\ No newline at end of file
```

Untuk menggunakan Git Untuk menampilkan detail tentang perbaikan di repo lokal, jalankan perintah git log:

```
git log
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
commit 94ba1e60
Author: John Doe <johndoe@example.com>
Date:   Mon May 23 15:39:14 2016 -0700

    Added horse.txt

commit 4c925148
Author: Jane Doe <janedoe@example.com>
Date:   Mon May 22 14:54:55 2014 -0700

    Added cat.txt and dog.txt
```

Untuk menampilkan hanya ID dan pesan commit, jalankan perintah git log --pretty=oneline:

```
git log --pretty=oneline
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
94ba1e60 Added horse.txt
```

```
4c925148 Added cat.txt and dog.txt
```

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Bandingkan komit di AWS CodeCommit

Anda dapat menggunakan CodeCommit konsol untuk melihat perbedaan antara penentu komit dalam CodeCommit repositori. Anda dapat dengan cepat melihat perbedaan antara commit dan induknya. Anda juga dapat membandingkan dua referensi, termasuk ID commit.

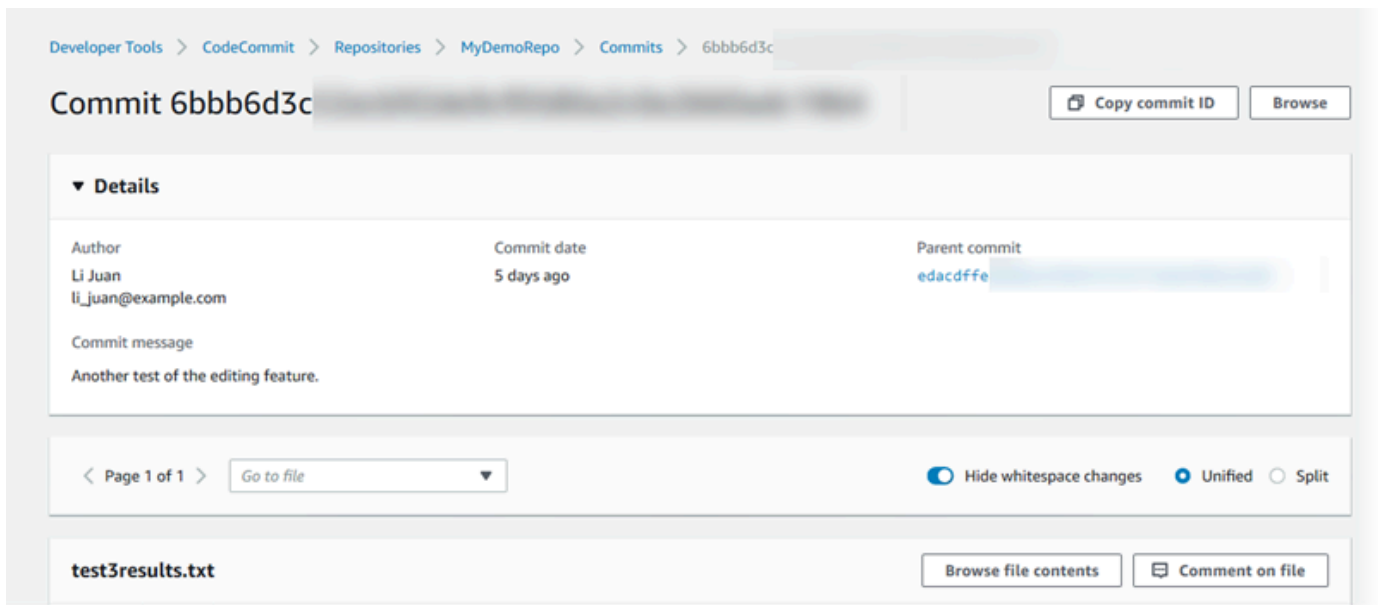
Topik

- [Bandingkan commit dengan induknya](#)
- [Bandingkan dua penentu commit mana pun](#)

Bandingkan commit dengan induknya

Anda dapat dengan cepat melihat perbedaan antara commit dan induknya untuk meninjau pesan commit, committer, dan apa yang berubah.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Pada halaman Repositori, pilih repositori di mana Anda ingin melihat perbedaan antara commit dan induknya.
3. Di panel navigasi, pilih Commit.
4. Pilih ID commit yang disingkat dari setiap commit dalam daftar. Tampilan berubah untuk menampilkan detail untuk commit ini, termasuk perbedaan antara commit tersebut dan commit induknya.



The screenshot displays the AWS CodeCommit interface for a specific commit. At the top, a breadcrumb trail shows the path: Developer Tools > CodeCommit > Repositories > MyDemoRepo > Commits > 6bbb6d3c. The main heading is "Commit 6bbb6d3c", with a "Copy commit ID" button and a "Browse" button to its right. Below this is a "Details" section with a dropdown arrow. The details include: Author (Li Juan, li_juan@example.com), Commit date (5 days ago), and Parent commit (edacdffe). The commit message is "Another test of the editing feature." Below the details is a navigation bar with "Page 1 of 1", a "Go to file" dropdown, and toggle options for "Hide whitespace changes" (checked), "Unified" (checked), and "Split" (unchecked). At the bottom, the file "test3results.txt" is listed, with "Browse file contents" and "Comment on file" buttons.

Anda dapat menampilkan perubahan secara berdampingan (tampilan Split) atau (tampilan Terpadu) inline. Anda juga dapat menyembunyikan atau menampilkan perubahan spasi putih. Anda juga dapat menambahkan komentar. Untuk informasi selengkapnya, lihat [Komentari sebuah commit](#).

Note

Preferensi Anda Untuk menampilkan kode dan pengaturan konsol lainnya disimpan sebagai cookie peramban setiap kali Anda mengubahnya. Untuk informasi selengkapnya, lihat [Bekerja dengan preferensi pengguna](#).

Developer Tools > CodeCommit > Repositories > MyDemoRepo > Commits > 7d09e44c

Commit 7d09e44c

[Copy commit ID](#) [Browse](#)

▼ Details

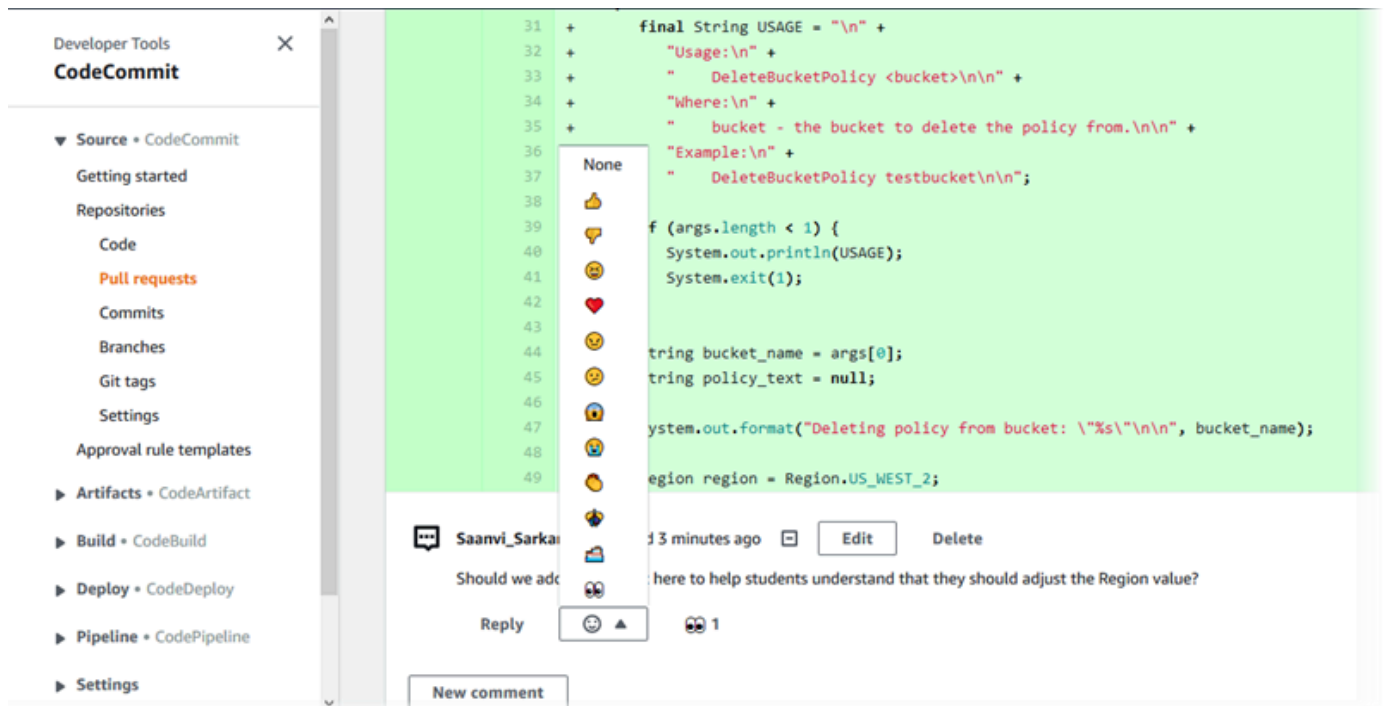
Author Mary Major mary_major@example.com	Commit date 48 minutes ago	Parent commit e6aca768
---	--------------------------------------	--

Commit message
Adding a readme file to the repository.

< Page 1 of 1 > Hide whitespace changes Unified Split

readme.md [Browse file contents](#) [Comment on file](#)

```
1 - This is a readme file that provides a basic description of what's in this repository.
  \ No newline at end of file
1 + Use this repository for code changes to the *Demo* project. The default branch is *master*. Cod
  \ No newline at end of file
```



Note

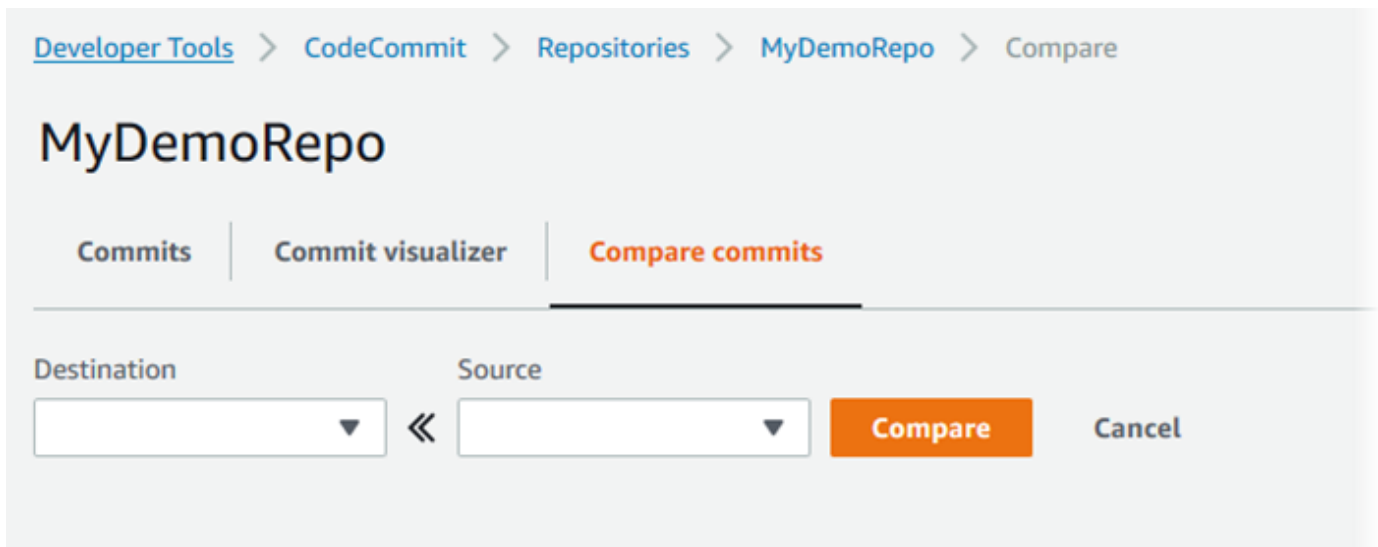
Tergantung pada gaya akhir baris, editor kode, dan faktor lainnya, Anda mungkin melihat seluruh baris ditambahkan atau dihapus, bukan perubahan tertentu dalam satu baris. Tingkat detail cocok dengan apa yang dikembalikan dalam perintah git show atau git diff.

5. Untuk membandingkan commit dengan induknya, dari tab Pemvisualisasi commit, pilih ID commit yang disingkat. detail komit, termasuk perubahan antara commit dan induknya, ditampilkan.

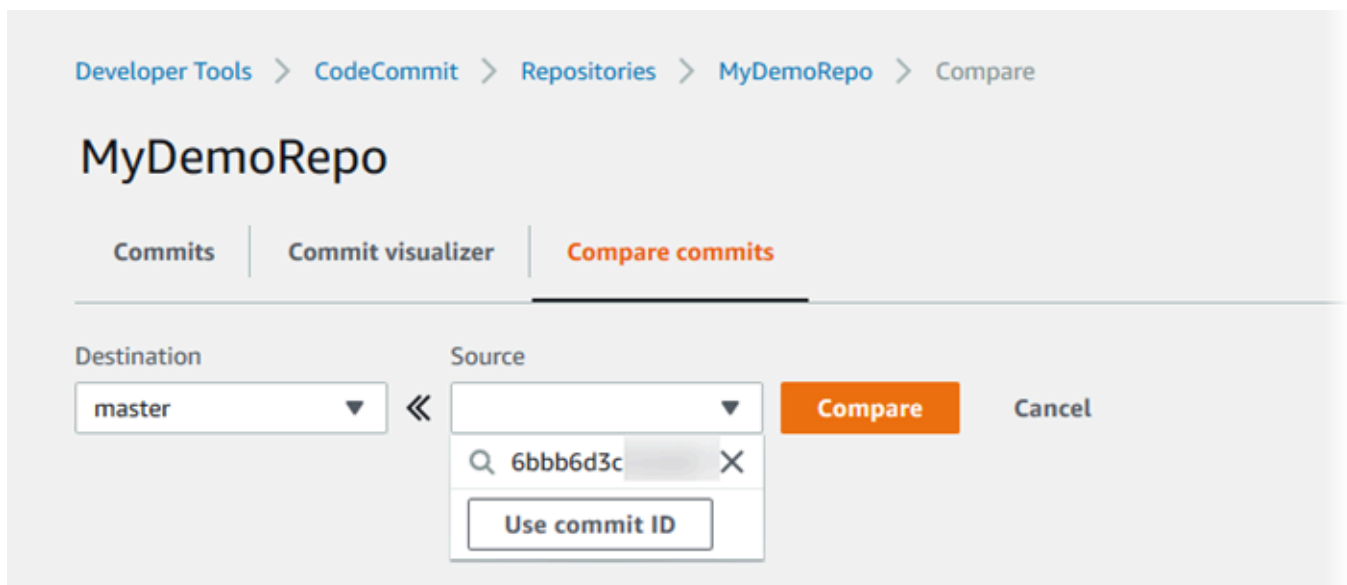
Bandungkan dua penentu commit mana pun

Anda dapat melihat perbedaan antara dua penentu komit di CodeCommit konsol. Penentu commit adalah referensi, seperti cabang, tag, dan ID commit.

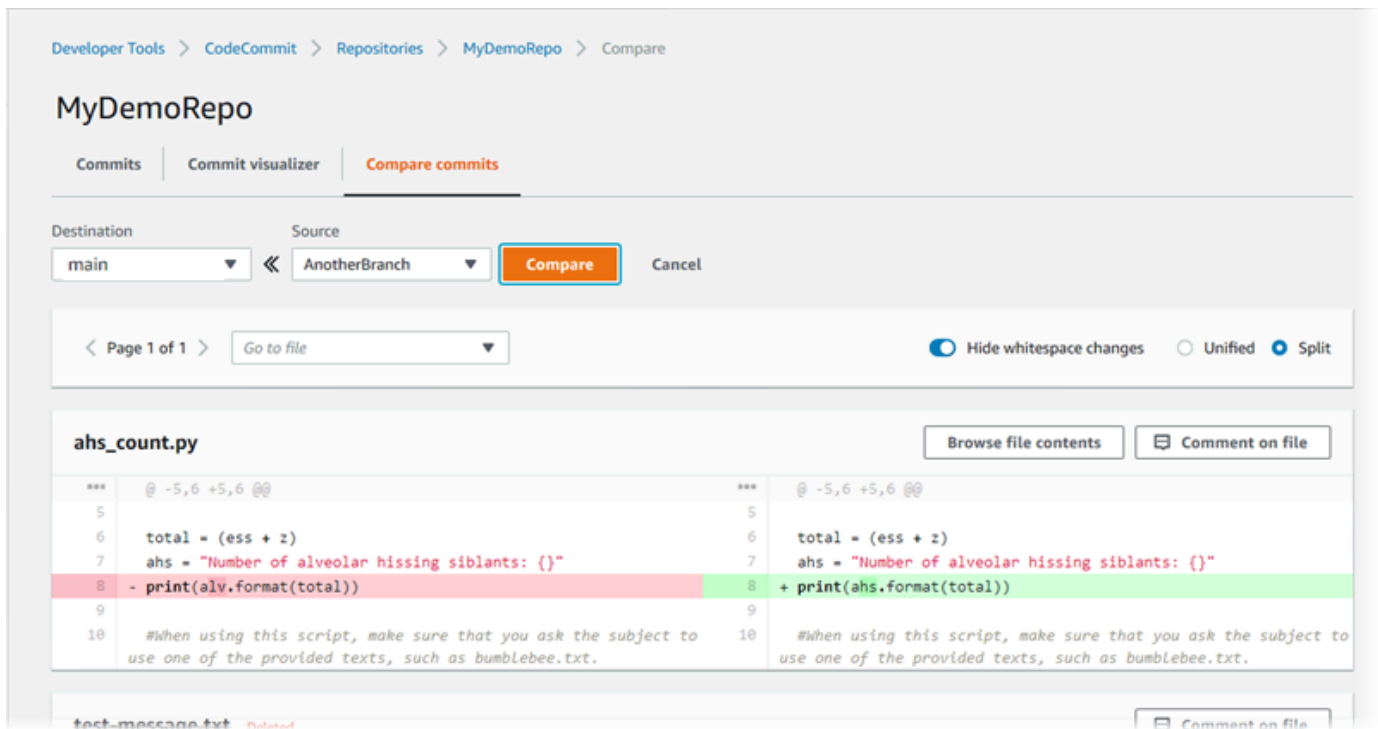
1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Pada halaman Repositori, pilih repositori tempat Anda ingin membandingkan komit, cabang, atau commit yang ditandai.
3. Di panel navigasi, pilih Commit, lalu pilih Bandungkan commit.



4. Gunakan kotak untuk membandingkan dua penentu commit.
 - Untuk membandingkan ujung cabang, pilih nama cabang dari daftar. Ini memilih commit terbaru dari cabang tersebut untuk perbandingan.
 - Untuk membandingkan commit dengan tag tertentu yang terkait dengannya, pilih nama tag dari daftar, jika ada. Ini memilih komit yang ditandai untuk perbandingan.
 - Untuk membandingkan commit tertentu, masukkan atau tempelkan ID commit di dalam kotak. Untuk mendapatkan ID commit penuh, pilih Commit di bilah navigasi, dan salin ID commit dari daftar. Pada halaman Bandingkan commit, tempelkan ID commit penuh di kotak teks, dan pilih Gunakan ID commit.



5. Setelah Anda memilih penentu, pilih Bandingkan.



Anda dapat menampilkan perbedaan secara berdampingan (tampilan Split) atau (tampilan Terpadu) inline. Anda juga dapat menyembunyikan atau menampilkan perubahan spasi putih.

6. Untuk menghapus pilihan perbandingan Anda, pilih Batalkan.

Komentari komit di AWS CodeCommit

Anda dapat menggunakan CodeCommit konsol untuk mengomentari komit di repositori, dan melihat serta membalas komentar pengguna lain tentang komit. Ini dapat membantu Anda mendiskusikan perubahan yang dibuat dalam repositori, termasuk:

- Mengapa perubahan dibuat.
- Apakah lebih banyak perubahan diperlukan.
- Apakah perubahan harus digabung ke cabang lain.

Anda dapat mengomentari commit keseluruhan, file dalam commit, atau baris tertentu atau perubahan dalam file. Anda juga dapat menautkan ke baris kode dengan memilih baris dan kemudian menyalin URL yang dihasilkan di browser Anda.

Note

Untuk hasil terbaik, gunakan komentar saat Anda masuk sebagai pengguna IAM. Fungsi komentar tidak dioptimalkan untuk pengguna yang masuk dengan kredensial akun akar, akses federasi, atau kredensial sementara.

Topik

- [Tampilkan komentar pada commit dalam repositori](#)
- [Tambahkandan balas komentar pada commit dalam repositori](#)
- [Tampilkan, tambah, perbarui, dan balas komentar \(AWS CLI\)](#)

Tampilkan komentar pada commit dalam repositori

Anda dapat menggunakan CodeCommit konsol untuk melihat komentar pada komit.

Untuk menampilkan komentar pada commit

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori yang ingin Anda tinjau komentarnya pada commit.
3. Di panel navigasi, pilih Commit. Pilih ID commit dari commit tersebut di mana Anda ingin melihat komentar apapun.

Halaman untuk commit yang ditampilkan, bersama dengan komentar apapun.

Tambahkandan balas komentar pada commit dalam repositori

Anda dapat menggunakan CodeCommit konsol untuk menambahkan komentar ke perbandingan komit dan induk, atau perbandingan antara dua komit yang ditentukan. Anda juga dapat membalas komentar dengan emoji, dengan komentar Anda sendiri, atau keduanya.

Menambah dan membalas komentar pada commit (konsol)

Anda dapat menambah dan membalas komentar untuk commit dengan teks dan dengan emoji. Komentar dan emoji Anda ditandai sebagai milik pengguna IAM atau peran yang Anda gunakan untuk masuk ke konsol tersebut.

Untuk menambah dan membalas komentar pada commit

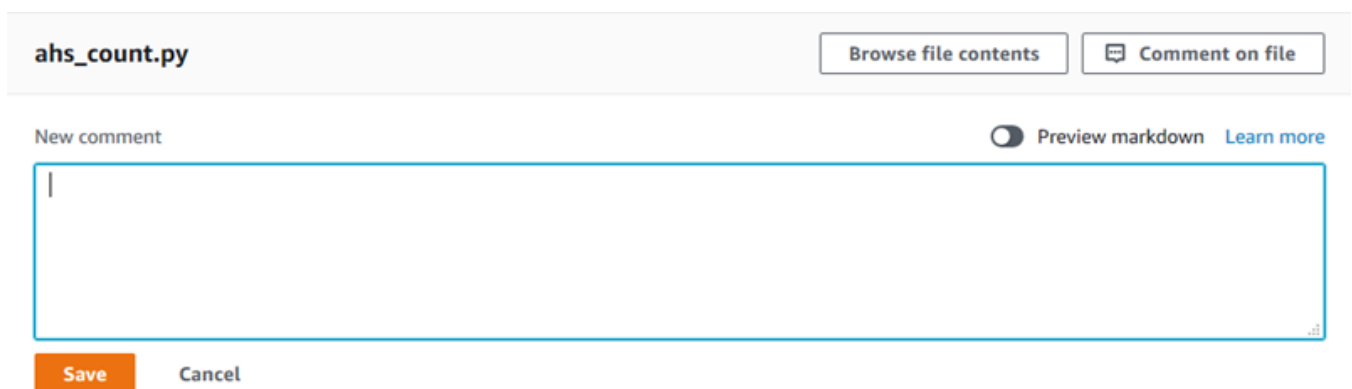
1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori yang ingin Anda komentari pada commit.
3. Di panel navigasi, pilih Commit. Pilih ID commit dari commit tersebut di mana Anda ingin menambahkan atau membalas komentar.

Halaman untuk commit tersebut ditampilkan, bersama dengan komentar apapun.

4. Untuk menambahkan komentar, lakukan salah satu hal berikut:
 - Untuk menambahkan komentar umum, di Komentar tentang perubahan, masukkan komentar Anda, dan kemudian pilih Simpan. Anda dapat menggunakan [Markdown](#), atau Anda dapat memasukkan komentar Anda di plaintext.



- Untuk menambahkan komentar ke file dalam commit, cari nama file. Pilih Komentar pada file, masukkan komentar Anda, dan kemudian pilih Simpan.



- Untuk menambahkan komentar ke baris yang berubah dalam commit, pergi ke baris di mana perubahan tersebut muncul. Pilih gelembung komentar



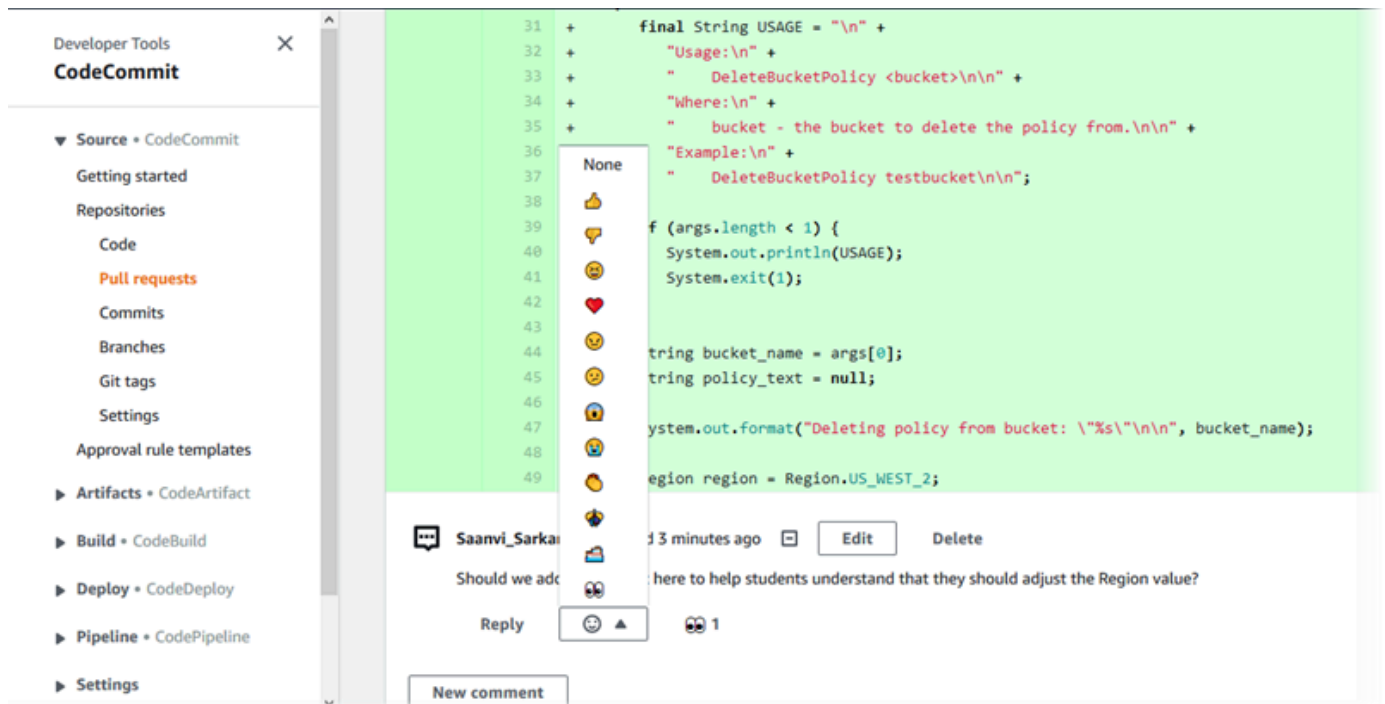
masukkan komentar Anda, dan kemudian pilih Simpan.

The screenshot shows a diff view for the file `ahs_count.py`. The code is shown in two columns, with line numbers 4, 5, 6, 7, and 8. Line 7 shows a change from `- alv = "Number of alveolar hissing sibilants: {}"` to `+ ahs = "Number of alveolar hissing sibilants: {}"`. A comment box is open, containing the text: "You've reverted to the old value here, which won't work. This should remain alv." Below the comment box are buttons for "Save" and "Cancel".

Note

Anda dapat mengedit komentar Anda setelah menyimpannya. Anda juga dapat menghapus isinya. Komentar akan tetap dengan pesan yang mengatakan bahwa isi telah dihapus. Pertimbangkan untuk menggunakan Pratinjau markdown untuk komentar Anda sebelum Anda menyimpannya.

5. Untuk membalas komentar pada commit, pilih Balas. Untuk membalas komentar dengan emoji, pilih emoji yang diinginkan dari daftar. Anda hanya dapat memilih satu emoji per komentar. Jika Anda ingin mengubah reaksi emoji, pilih yang berbeda dari daftar, atau pilih Tidak ada untuk menghilangkan reaksi Anda.

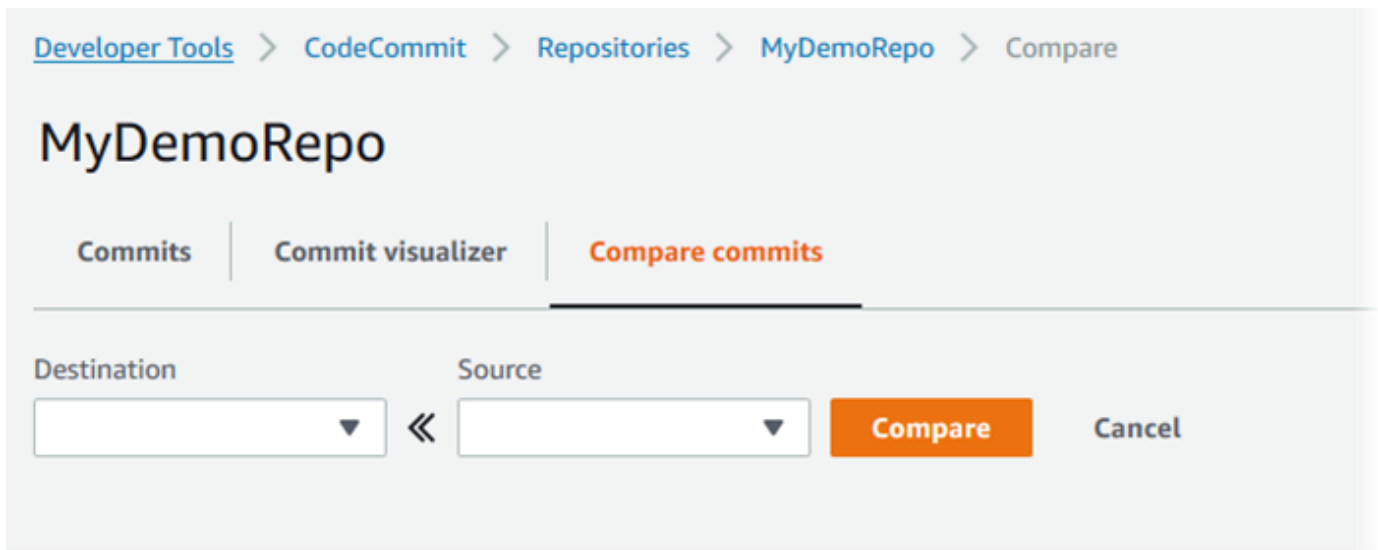


Tambahkan dan balas ke komentar ketika membandingkan dua penentu commit

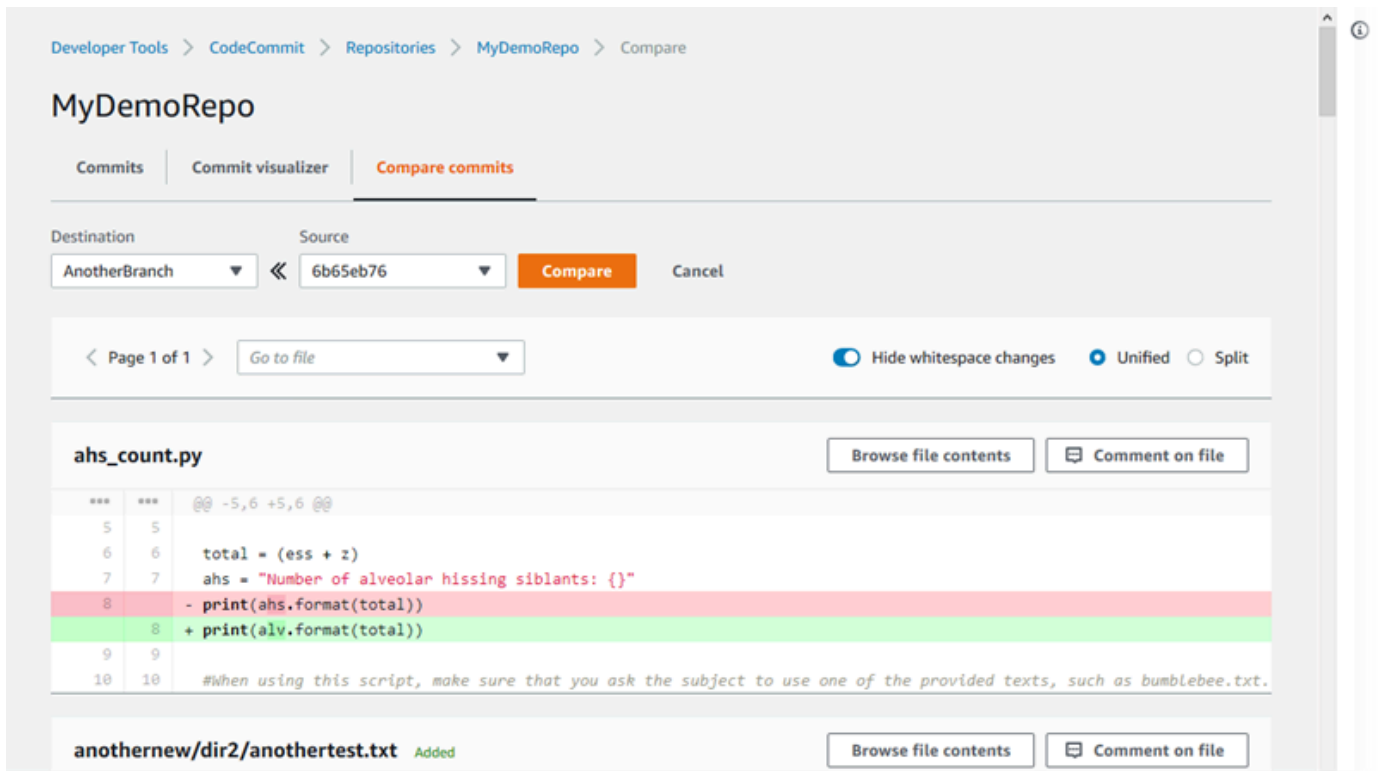
Anda dapat menambahkan komentar untuk perbandingan antara cabang, tag, atau commit.

Untuk menambah atau membalas komentar ketika membandingkan penentu commit

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih repositori tempat Anda ingin membandingkan komit, cabang, atau commit yang ditandai.
3. Di panel navigasi, pilih Commit, dan kemudian pilih tab Bandingkan commit.



- Gunakan Tujuan dan Sumber untuk membandingkan dua penentu commit. Gunakan daftar drop-down atau tempel di ID commit. Pilih Bandingkan.



- Lakukan salah satu atau beberapa hal berikut:
 - Untuk menambahkan komentar ke file atau baris, pilih gelembung komentar



- Untuk menambahkan komentar umum pada perubahan yang dibandingkan, buka Komentar tentang perubahan.

Tampilkan, tambah, perbarui, dan balas komentar (AWS CLI)

Anda dapat menampilkan, menambah, membalas, memperbarui, dan menghapus isi komentar dengan menjalankan perintah berikut:

- Untuk menampilkan komentar pada perbandingan antara dua commit, jalankan [get-comments-for-compared-commit](#).
- Untuk menampilkan detail pada sebuah komentar, jalankan [get-comment](#).
- Untuk menghapus konten komentar yang Anda buat, jalankan [delete-comment-content](#).
- Untuk membuat komentar pada perbandingan antara dua commit, jalankan [post-comment-for-compared-commit](#).
- Untuk memperbarui komentar, jalankan [update-comment](#).
- Untuk membalas komentar, jalankan [post-comment-reply](#).
- Untuk membalas komentar dengan emoji, jalankan [put-comment-reaction](#).
- Untuk menampilkan reaksi emoji pada komentar, jalankan [get-comment-reactions](#).

Untuk menampilkan komentar pada commit

1. Jalankan perintah `get-comments-for-compared-commit`, dengan menentukan:
 - Nama CodeCommit repositori (dengan `--repository-name` opsi).
 - ID commit penuh dari komit setelahnya, untuk menetapkan keterarahan dari perbandingan tersebut (dengan `--after-commit-id` option).
 - ID commit penuh dari komit sebelumnya, untuk menetapkan keterarahan dari perbandingan tersebut (dengan opsi `--before-commit-id`).
 - (Opsional) Sebuah enumerasi token untuk mengembalikan batch berikutnya dari hasil (dengan opsi `--next-token`).
 - (Opsional) Sebuah bilangan bulat non-negatif untuk membatasi jumlah hasil yang dikembalikan (dengan opsi `--max-results`).

Misalnya, untuk melihat komentar yang dibuat pada perbandingan antara dua komit dalam repositori bernama: *MyDemoRepo*

```
aws codecommit get-comments-for-compared-commit --repository-name MyDemoRepo --before-commit-ID 6e147360EXAMPLE --after-commit-id 317f8570EXAMPLE
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "commentsForComparedCommitData": [
    {
      "afterBlobId": "1f330709EXAMPLE",
      "afterCommitId": "317f8570EXAMPLE",
      "beforeBlobId": "80906a4cEXAMPLE",
      "beforeCommitId": "6e147360EXAMPLE",
      "comments": [
        {
          "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
          "clientRequestToken": "123Example",
          "commentId": "ff30b348EXAMPLEb9aa670f",
          "content": "Whoops - I meant to add this comment to the line, not
the file, but I don't see how to delete it.",
          "creationDate": 1508369768.142,
          "deleted": false,
          "CommentId": "123abc-EXAMPLE",
          "lastModifiedDate": 1508369842.278,
          "callerReactions": [],
          "reactionCounts":
            {
              "SMILE" : 6,
              "THUMBSUP" : 1
            }
        },
        {
          "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
          "clientRequestToken": "123Example",
          "commentId": "553b509bEXAMPLE56198325",
          "content": "Can you add a test case for this?",
          "creationDate": 1508369612.240,
          "deleted": false,
          "commentId": "456def-EXAMPLE",
```



```
        "lastModifiedDate": 1508369612.240,  
        "callerReactions": [],  
        "reactionCounts":  
        {  
            "THUMBSUP" : 2  
        }  
    },  
    "location": {  
        "filePath": "cl_sample.js",  
        "filePosition": 1232,  
        "relativeFileVersion": "after"  
    },  
    "repositoryName": "MyDemoRepo"  
}  
],  
"nextToken": "exampleToken"  
}
```

Untuk menampilkan detail komentar pada commit

1. Jalankan perintah `get-comment`, dengan menentukan ID komentar yang dihasilkan sistem. Sebagai contoh:

```
aws codecommit get-comment --comment-id ff30b348EXAMPLEb9aa670f
```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
{  
  "comment": {  
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
    "clientRequestToken": "123Example",  
    "commentId": "ff30b348EXAMPLEb9aa670f",  
    "content": "Whoops - I meant to add this comment to the line, but I don't see  
how to delete it.",  
    "creationDate": 1508369768.142,  
    "deleted": false,  
    "commentId": "",  
    "lastModifiedDate": 1508369842.278,  
    "callerReactions": [],  
    "reactionCounts":  
    {  

```

```
        "SMILE" : 6,  
        "THUMBSUP" : 1  
    }  
}  
}
```

Untuk menghapus isi komentar pada commit

1. Jalankan perintah `delete-comment-content`, dengan menentukan ID komentar yang dihasilkan sistem. Sebagai contoh:

```
aws codecommit delete-comment-content --comment-id ff30b348EXAMPLEb9aa670f
```

Note

Anda hanya dapat menghapus konten komentar jika `AWSCodeCommitFullAccess` kebijakan diterapkan, atau jika Anda memiliki `DeleteCommentContent` izin yang disetel ke Izinkan.

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{  
  "comment": {  
    "creationDate": 1508369768.142,  
    "deleted": true,  
    "lastModifiedDate": 1508369842.278,  
    "clientRequestToken": "123Example",  
    "commentId": "ff30b348EXAMPLEb9aa670f",  
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",  
    "callerReactions": [],  
    "reactionCounts":  
      {  
        "CLAP" : 1  
      }  
  }  
}
```

Untuk membuat komentar pada commit

1. Jalankan perintah `post-comment-for-compared-commit`, dengan menentukan:
 - Nama CodeCommit repositori (dengan `--repository-name` opsi).
 - ID commit penuh dari komit setelahnya, untuk menetapkan keterarahan dari perbandingan tersebut (dengan opsi `--after-commit-id`).
 - ID commit penuh dari komit sebelumnya, untuk menetapkan keterarahan dari perbandingan tersebut (dengan opsi `--before-commit-id`).
 - Sebuah token idempotensi yang unik dan dihasilkan klien (dengan opsi `--client-request-token`).
 - Isi dari komentar Anda (dengan opsi `--content`).
 - Daftar informasi lokasi tentang tempat untuk menempatkan komentar, termasuk:
 - Nama file yang dibandingkan, termasuk ekstensi dan subdirektori-nya, jika ada (dengan atribut `filePath`).
 - Nomor baris perubahan dalam file dibandingkan (dengan atribut `filePosition`).
 - Apakah komentar pada perubahan adalah sebelum atau sesudah dalam perbandingan antara sumber dan cabang tujuan (dengan atribut `relativeFileVersion`).

Misalnya, untuk menambahkan komentar *"Dapatkan Anda menambahkan kasus uji untuk ini?"* pada perubahan ke file `cl_sample.js` dalam perbandingan antara dua komit dalam repositori bernama: *MyDemoRepo*

```
aws codecommit post-comment-for-compared-commit --repository-name MyDemoRepo
--before-commit-id 317f8570EXAMPLE --after-commit-id 5d036259EXAMPLE --client-
request-token 123Example --content "Can you add a test case for this?" --location
filePath=cl_sample.js,filePosition=1232,relativeFileVersion=AFTER
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "afterBlobId": "1f330709EXAMPLE",
  "afterCommitId": "317f8570EXAMPLE",
  "beforeBlobId": "80906a4cEXAMPLE",
  "beforeCommitId": "6e147360EXAMPLE",
  "comment": {
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
    "clientRequestToken": "",
```

```
        "commentId": "553b509bEXAMPLE56198325",
        "content": "Can you add a test case for this?",
        "creationDate": 1508369612.203,
        "deleted": false,
        "commentId": "abc123-EXAMPLE",
        "lastModifiedDate": 1508369612.203,
        "callerReactions": [],
        "reactionCounts": []
    },
    "location": {
        "filePath": "cl_sample.js",
        "filePosition": 1232,
        "relativeFileVersion": "AFTER"
    },
    "repositoryName": "MyDemoRepo"
}
```

Untuk memperbarui komentar pada commit

1. Jalankan perintah `update-comment`, dengan menentukan ID komentar yang dihasilkan sistem dan konten untuk menggantikan konten yang ada.

Misalnya, untuk menambahkan konten *"Tetap seperti yang diminta. Saya akan memperbarui permintaan tarik."* untuk komentar dengan ID **442B498BEXAMPLE5756813**:

```
aws codecommit update-comment --comment-id 442b498bEXAMPLE5756813 --content "Fixed as requested. I'll update the pull request."
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "comment": {
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
    "clientRequestToken": "",
    "commentId": "442b498bEXAMPLE5756813",
    "content": "Fixed as requested. I'll update the pull request.",
    "creationDate": 1508369929.783,
    "deleted": false,
    "lastModifiedDate": 1508369929.287,
    "callerReactions": [],
```

```
    "reactionCounts":
      {
        "THUMBSUP" : 2
      }
  }
}
```

Untuk membalas komentar pada commit

1. Untuk mengirim balasan ke komentar dalam permintaan tarik, jalankan perintah `post-comment-reply`, dengan menentukan:
 - ID yang dihasilkan sistem dari komentar yang ingin Anda balas (dengan opsi `--in-reply-to`).
 - Sebuah token idempotensi yang unik dan dihasilkan klien (dengan opsi `--client-request-token`).
 - Isi dari komentar Anda (dengan opsi `--content`).

Misalnya, untuk menambahkan balasan *"Tangkapan bagus. Aku akan menghapusnya."* untuk komentar dengan ID yang dihasilkan sistem *ABCD1234exampleB5678efgh*:

```
aws codecommit post-comment-reply --in-reply-to abcd1234EXAMPLEb5678efgh --
content "Good catch. I'll remove them." --client-request-token 123Example
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "comment": {
    "authorArn": "arn:aws:iam::111111111111:user/Li_Juan",
    "clientRequestToken": "123Example",
    "commentId": "442b498bEXAMPLE5756813",
    "content": "Good catch. I'll remove them.",
    "creationDate": 1508369829.136,
    "deleted": false,
    "CommentId": "abcd1234EXAMPLEb5678efgh",
    "lastModifiedDate": 150836912.221,
    "callerReactions": [],
    "reactionCounts": []
  }
}
```

Untuk membalas komentar pada commit dengan emoji

- Untuk membalas komentar dalam permintaan tarik dengan emoji, atau untuk mengubah nilai reaksi emoji Anda, jalankan perintah `put-comment-reaction`, dengan menentukan:
 - ID yang dihasilkan sistem dari komentar yang ingin Anda balas dengan emoji.
 - Nilai dari reaksi yang ingin Anda tambahkan atau perbarui. Nilai yang dapat diterima termasuk emoji, shortcode, dan nilai-nilai Unicode yang didukung.

Nilai berikut didukung untuk emoji di: CodeCommit

Emoji	Shortcode	Unicode
#	:jempol ke atas:	U+1F44D
#	:jempol ke bawah:	U+1F44E
#	:senyum:	U+1F604
♥	:hati:	U+2764
#	:marah:	U+1F620
#	:bingung:	U+1F615
#	:menjerit:	U+1F631
#	:terisak:	U+1F62D
#	:tepuk tangan:	U+1F44F
#	:confetti_ball:	U+1F38A
#	:kapal:	U+1F6A2
#	:mata:	U+1F440
	tidak ada	U+0000

Misalnya, untuk menambahkan emoji *:jempol ke atas:* untuk komentar dengan ID yang dihasilkan sistem *abcd1234EXAMPLEb5678efgh*:

```
aws codecommit put-comment-reaction --comment-id abcd1234EXAMPLEb5678efgh --  
reaction-value :thumbsup:
```

2. Jika berhasil, perintah ini tidak menghasilkan output.

Untuk menampilkan reaksi emoji terhadap pkomentar

1. Untuk menampilkan reaksi emoji terhadap komentar, termasuk pengguna yang bereaksi dengan emoji tersebut, jalankan perintah `get-comment-reactions`, dengan menentukan ID yang dihasilkan sistem komentar.

Misalnya, untuk menampilkan reaksi emoji terhadap komentar dengan ID yang dihasilkan sistem *ABCD1234exampleB5678efgh*:

```
aws codecommit get-comment-reactions --comment-id abcd1234EXAMPLEb5678efgh
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{  
  "reactionsForComment": [  
    {  
      "reaction": {  
        "emoji": "#",  
        "shortCode": "thumbsup",  
        "unicode": "U+1F44D"  
      },  
      "users": [  
        "arn:aws:iam::123456789012:user/Li_Juan",  
        "arn:aws:iam::123456789012:user/Mary_Major",  
        "arn:aws:iam::123456789012:user/Jorge_Souza"  
      ]  
    },  
    {  
      "reaction": {  
        "emoji": "#",  
        "shortCode": "thumbsdown",  
        "unicode": "U+1F44E"  
      },  
      "users": [  
        "arn:aws:iam::123456789012:user/Li_Juan",  
        "arn:aws:iam::123456789012:user/Mary_Major",  
        "arn:aws:iam::123456789012:user/Jorge_Souza"  
      ]  
    }  
  ]  
}
```

```
        "unicode": "U+1F44E"
      },
      "users": [
        "arn:aws:iam::123456789012:user/Nikhil_Jayashankar"
      ]
    },
    {
      "reaction": {
        "emoji": "#",
        "shortCode": "confused",
        "unicode": "U+1F615"
      },
      "users": [
        "arn:aws:iam::123456789012:user/Saanvi_Sarkar"
      ]
    }
  ]
}
```

Buat tag Git di AWS CodeCommit

Anda dapat menggunakan tag Git untuk menandai commit dengan label yang membantu pengguna repositori lain memahami pentingnya. Untuk membuat tag Git di CodeCommit repositori, Anda dapat menggunakan Git dari repo lokal yang terhubung ke repositori. CodeCommit Setelah Anda membuat tag Git di repo lokal, Anda dapat menggunakannya git push --tags untuk mendorongnya ke CodeCommit repositori.

Untuk informasi selengkapnya, lihat [Tampilkan detail tag](#).

Gunakan Git untuk membuat tag

Ikuti langkah-langkah ini untuk menggunakan Git dari repo lokal untuk membuat tag Git di CodeCommit repositori.

Dalam langkah-langkah ini, kami berasumsi bahwa Anda telah menghubungkan repo lokal ke CodeCommit repositori. Untuk petunjuk, lihat [Connect ke sebuah repositori](#).

1. Jalankan git tag ***new-tag-name commit-id*** perintah, di ***new-tag-name*** mana nama tag Git baru dan ***commit-id*** adalah ID dari komit untuk dikaitkan dengan tag Git.

Misalnya, perintah berikut membuat tag Git yang bernama beta dan mengaitkannya dengan ID commit dc082f9a...af873b88:

```
git tag beta dc082f9a...af873b88
```

2. Untuk mendorong tag Git baru dari repo lokal ke CodeCommit repositori, jalankan git push **remote-name new-tag-name** perintah, di mana nama *jarak jauh adalah nama* CodeCommit repositori dan merupakan *new-tag-nam* nama tag Git baru.

Misalnya, untuk mendorong tag Git baru bernama beta ke CodeCommit repositori bernama: origin

```
git push origin beta
```

Note

Untuk mendorong semua tag Git baru dari repo lokal Anda ke CodeCommit repositori, jalankan. git push --tags

Untuk memastikan repo lokal Anda diperbarui dengan semua tag Git di CodeCommit repositori, jalankan git fetch diikuti oleh. git fetch --tags

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Lihat detail tag Git di AWS CodeCommit

Dalam Git, tag adalah label Anda yang dapat diterapkan untuk referensi seperti commit untuk menandainya dengan informasi yang mungkin penting untuk pengguna repositori lainnya. Misalnya, Anda mungkin menandai commit yang merupakan titik rilis beta untuk proyek dengan tag **beta**. Untuk informasi selengkapnya, lihat [Gunakan Git untuk membuat tag](#). Tag Git berbeda dari tag repositori. Untuk informasi selengkapnya tentang cara menggunakan tag repositori, lihat [Tambahkan tag ke sebuah repositori](#).

Anda dapat menggunakan AWS CodeCommit konsol untuk melihat informasi tentang tag Git di repositori Anda, termasuk tanggal dan pesan komit dari komit yang direferensikan oleh setiap tag Git. Dari konsol tersebut, Anda dapat membandingkan commit yang direferensikan oleh tag dengan

kepala cabang default repositori Anda. Seperti komit lainnya, Anda juga dapat melihat kode pada titik tag Git tersebut.

Anda juga dapat menggunakan Git dari terminal atau baris perintah Untuk menampilkan detail tentang tag Git di repo lokal.

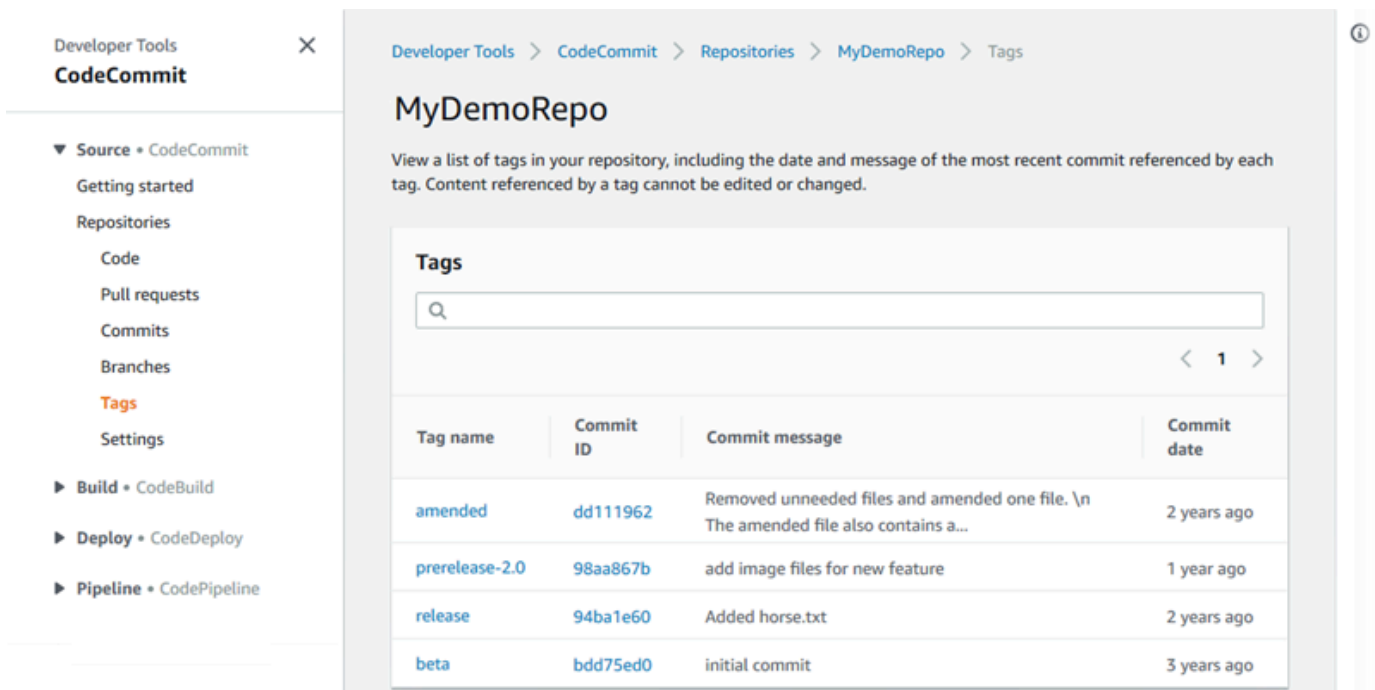
Topik

- [Tampilkan detail tag \(konsol\)](#)
- [Tampilkan detail tag Git \(Git\)](#)

Tampilkan detail tag (konsol)

Gunakan AWS CodeCommit konsol untuk dengan cepat melihat daftar tag Git untuk repositori Anda dan detail tentang komit yang direferensikan oleh tag Git.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori yang Anda ingin tampilkan tag.
3. Di panel navigasi, pilih tag Git.



The screenshot shows the AWS CodeCommit console interface. On the left is a navigation sidebar with 'CodeCommit' selected. The main content area shows the 'Tags' page for a repository named 'MyDemoRepo'. Below the title, there is a search bar and a table listing tags. The table has four columns: 'Tag name', 'Commit ID', 'Commit message', and 'Commit date'. The tags listed are 'amended', 'prerelease-2.0', 'release', and 'beta'.

Tag name	Commit ID	Commit message	Commit date
amended	dd111962	Removed unneeded files and amended one file. \n The amended file also contains a...	2 years ago
prerelease-2.0	98aa867b	add image files for new feature	1 year ago
release	94ba1e60	Added horse.txt	2 years ago
beta	bdd75ed0	initial commit	3 years ago

4. Lakukan salah satu hal berikut ini:
 - Untuk menampilkan kode seperti apa adanya pada commit itu, pilih nama tag Git.

- Untuk menampilkan detail commit, termasuk pesan commit penuh, committer, dan penulis, pilih ID commit yang disingkat.

Tampilkan detail tag Git (Git)

Untuk menggunakan Git Untuk menampilkan detail tentang tag Git di repo lokal, jalankan salah satu dari perintah berikut ini:

- [git tag](#) untuk menampilkan daftar nama tag Git.
- [git show](#) untuk menampilkan informasi tentang tag Git tertentu.
- [git ls-remote](#) untuk melihat informasi tentang tag Git di repositori. CodeCommit

Note

Untuk memastikan bahwa repo lokal Anda diperbarui dengan semua tag Git di CodeCommit repositori, jalankan git fetch diikuti oleh. `git fetch --tags`

Pada langkah-langkah berikut, kami berasumsi bahwa Anda telah menghubungkan repo lokal ke CodeCommit repositori. Untuk petunjuk, lihat [Connect ke sebuah repositori](#).

Untuk menampilkan daftar tag Git di repo lokal

1. Jalankan perintah git tag:

```
git tag
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
beta  
release
```

Note

Jika tidak ada tag yang didefinisikan, git tag tidak mengembalikan apa pun.

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Untuk menampilkan informasi tentang tag Git di repo lokal

1. Jalankan perintah git show **tag-name**. Misalnya, Untuk menampilkan informasi tentang tag Git yang bernama beta, jalankan:

```
git show beta
```

2. Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
commit 317f8570...ad9e3c09
Author: John Doe <johndoe@example.com>
Date:   Tue Sep 23 13:49:51 2014 -0700

    Added horse.txt

diff --git a/horse.txt b/horse.txt
new file mode 100644
index 0000000..df42ff1
--- /dev/null
+++ b/horse.txt
@@ -0,0 +1 @@
+The horse (Equus ferus caballus) is one of two extant subspecies of Equus ferus
\ No newline at end of file
```

Note

Untuk keluar dari output informasi tag Git, ketik :q.

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Untuk melihat informasi tentang tag Git di CodeCommit repositori

1. Jalankan perintah git ls-remote --tags.

```
git ls-remote --tags
```

2. Jika berhasil, perintah ini menghasilkan sebagai output daftar tag Git dalam CodeCommit repositori:

```
129ce87a...70fbffba    refs/tags/beta
785de9bd...59b402d8    refs/tags/release
```

Jika tidak ada tanda Git yang telah didefinisikan, `git ls-remote --tags` mengembalikan sebuah baris kosong.

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Hapus tag Git di AWS CodeCommit

Untuk menghapus tag Git di CodeCommit repositori, gunakan Git dari repo lokal yang terhubung ke repositori. CodeCommit

Gunakan Git untuk menghapus tag Git

Ikuti langkah-langkah ini untuk menggunakan Git dari repo lokal untuk menghapus tag Git di CodeCommit repositori.

Langkah-langkah ini ditulis dengan asumsi bahwa Anda telah menghubungkan repo lokal ke repositori. CodeCommit Untuk petunjuk, lihat [Connect ke sebuah repositori](#).

1. Untuk menghapus tag Git dari repo lokal, jalankan `git tag -d tag-name` di mana *tag-name* adalah nama Git tag yang ingin Anda hapus.

Tip

Untuk mendapatkan daftar nama tag Git, jalankan `git tag`.

Misalnya, untuk menghapus tag Git di repo lokal yang bernama beta:

```
git tag -d beta
```

2. Untuk menghapus tag Git dari CodeCommit repositori, jalankan `git push remote-name --delete tag-name` perintah di mana nama *jarak jauh adalah nama panggilan yang* digunakan repo lokal untuk CodeCommit repositori dan nama tag *adalah* nama tag Git yang ingin Anda hapus dari repositori. CodeCommit

 Tip

Untuk mendapatkan daftar nama CodeCommit repositori dan URL mereka, jalankan perintah. `git remote -v`

Misalnya, untuk menghapus tag Git bernama beta dalam CodeCommit repositori bernama: `origin`

```
git push origin --delete beta
```

Bekerja dengan cabang di AWS CodeCommit repositori

Apa itu cabang? Dalam Git, cabang adalah pointer atau referensi ke commit. Dalam pengembangan, mereka adalah cara yang nyaman untuk mengatur pekerjaan Anda. Anda dapat menggunakan cabang untuk memisahkan pekerjaan pada versi baru atau berbeda dari file tanpa mempengaruhi pekerjaan di cabang-cabang lain. Anda dapat menggunakan cabang untuk mengembangkan fitur baru, menyimpan versi tertentu dari proyek Anda dari commit tertentu, dan banyak lagi. Saat Anda membuat perbaikan pertama Anda, cabang default dibuat untuk Anda. Cabang default ini adalah cabang yang digunakan sebagai dasar atau cabang default dalam repositori lokal (repos) ketika pengguna mengkloning repositori. Nama cabang default itu bervariasi tergantung pada bagaimana Anda membuat commit pertama Anda. Jika Anda menambahkan file pertama ke repositori Anda dengan menggunakan CodeCommit konsol, AWS CLI, atau salah satu SDK, nama cabang default tersebut adalah main. Ini adalah nama cabang default yang digunakan dalam contoh dalam panduan ini. Jika Anda mendorong commit pertama Anda menggunakan klien Git, nama cabang default adalah apa yang klien Git tentukan sebagai default. Pertimbangkan untuk mengkonfigurasi klien Git anda untuk menggunakan utama sebagai nama untuk cabang awal.

Di CodeCommit, Anda dapat mengubah cabang default untuk repositori Anda. Anda juga dapat membuat dan menghapus cabang dan melihat detail tentang cabang. Anda dapat dengan cepat membandingkan perbedaan antara cabang dan cabang default (atau dua cabang mana pun). Untuk menampilkan riwayat cabang dan penggabungan di repositori Anda, Anda dapat menggunakan [pemvisualisasi commit](#), yang ditunjukkan dalam grafik berikut.

Developer Tools > CodeCommit > Repositories > MyDemoRepo > Commits

MyDemoRepo

Commits | **Commit visualizer** | Compare commits

Commit visualizer

Commit Hash	Description	Time Ago
d615e7ae	Merge branch 'AnotherBranch' into testbranch	2 minutes ago
b6589863	Added another file.	2 minutes ago
73a6e39c	remote-tracking branch 'refs/remotes/origin/jane-branch' into jane-branch	
6bbb6d3c	Another test of the editing feature.	20 minutes ago
edacdfef	Testing this out to see how well it works.	23 minutes ago
70bb94d7	Revised test results with correct information.	36 minutes ago
b78e6d1c	Merge branch 'results' into testbranch	50 minutes ago
84b7d158	Edited ahs_count.py	50 minutes ago

Untuk informasi tentang bekerja dengan aspek lain dari repositori Anda di CodeCommit, lihat, [Bekerja dengan repositori](#), [Bekerja dengan file](#) [Bekerja dengan permintaan pull](#) [Bekerja dengan commit](#), dan [Bekerja dengan preferensi pengguna](#)

Topik

- [Buat cabang di AWS CodeCommit](#)
- [Batasi dorongan dan penggabungan ke cabang di AWS CodeCommit](#)
- [Lihat detail cabang di AWS CodeCommit](#)
- [Bandingkan dan gabungkan cabang di AWS CodeCommit](#)
- [Ubah pengaturan cabang di AWS CodeCommit](#)
- [Hapus cabang di AWS CodeCommit](#)

Buat cabang di AWS CodeCommit

Anda dapat menggunakan CodeCommit konsol atau AWS CLI untuk membuat cabang untuk repositori Anda. Ini adalah cara cepat untuk memisahkan pekerjaan pada versi baru atau berbeda dari file tanpa mempengaruhi pekerjaan di cabang default. Setelah Anda membuat cabang di CodeCommit konsol, Anda harus menarik perubahan itu ke repo lokal Anda. Atau, Anda dapat membuat cabang secara lokal dan kemudian menggunakan Git dari repo lokal yang terhubung ke CodeCommit repositori untuk mendorong perubahan itu.

Topik

- [Buatlah sebuah cabang \(konsol\)](#)
- [Buatlah sebuah cabang \(Git\)](#)
- [Buatlah sebuah cabang \(AWS CLI\)](#)

Buatlah sebuah cabang (konsol)

Anda dapat menggunakan CodeCommit konsol untuk membuat cabang di CodeCommit repositori. Lain kali pengguna menarik perubahan dari repositori, mereka melihat cabang baru.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin membuat cabang.
3. Di panel navigasi, pilih Cabang.
4. Pilih Buat cabang.

Create branch [X]

Branch name
feature_advanced-class

Branch from
[Dropdown menu]

Branches

- main
Default branch
- bugfix-1236
- feature-lambdafunctions
- feature-new-wizard
- feature-randomizationfeature

Git tags

- release
- prerelease-2.0
- beta
- amended

Create branch

Di Nama cabang, masukkan nama untuk cabang tersebut. Di Cabang dari, pilih cabang atau tag dari daftar, atau tempelkan ID commit. Pilih Buat cabang.

Buatlah sebuah cabang (Git)

Ikuti langkah-langkah ini untuk menggunakan Git dari repo lokal untuk membuat cabang di repo lokal dan kemudian dorong cabang itu ke repositori. CodeCommit

Langkah-langkah ini ditulis dengan asumsi bahwa Anda telah menghubungkan repo lokal ke repositori. CodeCommit Untuk petunjuk, lihat [Connect ke sebuah repositori](#).

1. Buat cabang di repo lokal Anda dengan menjalankan git checkout -b ***new-branch-name*** perintah, di ***new-branch-name*** mana nama cabang baru.

Misalnya, perintah berikut membuat cabang yang bernama MyNewBranch di repo lokal:

```
git checkout -b MyNewBranch
```

2. Untuk mendorong cabang baru dari repo lokal ke CodeCommit repositori, jalankan git push perintah, dengan menentukan file dan file. ***remote-name new-branch-name***

Misalnya, untuk mendorong cabang baru di repo lokal yang dinamai MyNewBranch ke CodeCommit repositori dengan nama panggilan: origin

```
git push origin MyNewBranch
```

Note

Jika Anda menambahkan opsi -u untuk git push (Misalnya, git push -u origin main), maka di masa depan Anda bisa menjalankan git push tanpa ***remote-name branch-name***. Informasi pelacakan hulu diatur. Untuk mendapatkan informasi pelacakan hulu, jalankan git remote show ***remote-name*** (sebagai contoh, git remote show origin).

Untuk menampilkan daftar semua cabang pelacakan lokal dan jarak jauh, jalankan git branch --all.

Untuk menyiapkan cabang di repo lokal yang terhubung ke cabang di CodeCommit repositori, jalankan. git checkout ***remote-branch-name***

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Buatlah sebuah cabang (AWS CLI)

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal file AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Ikuti langkah-langkah ini untuk menggunakan AWS CLI untuk membuat cabang di CodeCommit repositori dan kemudian dorong cabang itu ke repositori. CodeCommit Untuk langkah-langkah untuk membuat commit awal dan menentukan nama cabang default untuk repositori kosong, lihat [Buat komit pertama untuk repositori menggunakan AWS CLI](#).

1. Jalankan perintah `create-branch`, dengan menentukan:

- Nama CodeCommit repositori tempat cabang dibuat (dengan `--repository-name` opsi).

Note

Untuk mendapatkan nama repositori, jalankan perintah CodeCommit [list-repositori](#).

- Nama cabang baru (dengan opsi `--branch-name`).
- ID dari commit yang ditunjuk oleh cabang baru tersebut (dengan opsi `--commit-id`).

Misalnya, untuk membuat cabang bernama `MyNewBranch` yang menunjuk ke ID komit `317f8570EXAMPLE` dalam CodeCommit repositori bernama: `MyDemoRepo`

```
aws codecommit create-branch --repository-name MyDemoRepo --branch-name MyNewBranch
--commit-id 317f8570EXAMPLE
```

Perintah ini menghasilkan output hanya jika ada kesalahan.

2. Untuk memperbarui daftar cabang CodeCommit repositori yang tersedia di repo lokal Anda dengan nama cabang jarak jauh yang baru, jalankan `git remote update remote-name`

Misalnya, untuk memperbarui daftar cabang yang tersedia untuk CodeCommit repositori dengan nama panggilan: `origin`

```
git remote update origin
```

Note

Atau, Anda dapat menjalankan perintah `git fetch`. Anda juga dapat melihat semua cabang jarak jauh dengan menjalankan `git branch --all`, tapi sampai Anda memperbarui daftar repo lokal Anda, cabang jarak jauh yang Anda buat tersebut tidak muncul dalam daftar. Untuk opsi lainnya, lihat dokumentasi Git Anda.

3. Untuk menyiapkan cabang di repo lokal yang terhubung ke cabang baru di CodeCommit repositori, jalankan `git checkout remote-branch-name`

Note

Untuk mendapatkan daftar nama CodeCommit repositori dan URL mereka, jalankan perintah.
`git remote -v`

Batasi dorongan dan penggabungan ke cabang di AWS CodeCommit

Secara default, setiap pengguna CodeCommit repositori yang memiliki izin yang cukup untuk mendorong kode ke repositori dapat berkontribusi ke cabang mana pun di repositori itu. Ini benar tidak peduli bagaimana Anda menambahkan cabang ke repositori: dengan menggunakan konsol, baris perintah, atau Git. Namun, Anda mungkin ingin mengkonfigurasi cabang sehingga hanya beberapa pengguna repositori yang dapat mendorong atau menggabungkan kode ke cabang tersebut. Misalnya, Anda mungkin ingin mengkonfigurasi cabang yang digunakan untuk kode produksi sehingga hanya subset dari developer senior dapat mendorong atau menggabungkan perubahan ke cabang tersebut. Developer lain masih dapat menarik dari cabang, membuat cabang mereka sendiri, dan membuat permintaan tarik, tetapi mereka tidak dapat mendorong atau menggabungkan perubahan ke cabang tersebut. Anda dapat mengkonfigurasi akses ini dengan membuat kebijakan bersyarat yang menggunakan kunci konteks untuk satu atau lebih cabang di IAM.

Note

Untuk menyelesaikan beberapa prosedur dalam topik ini, Anda harus masuk dengan pengguna administratif yang memiliki izin yang memadai untuk mengkonfigurasi dan menerapkan kebijakan IAM. Untuk informasi selengkapnya, lihat [Membuat pengguna dan grup admin IAM pertama Anda](#).

Topik

- [Mengkonfigurasi kebijakan IAM untuk membatasi dorongan dan penggabungan ke cabang](#)
- [Terapkan kebijakan IAM ke grup atau peran IAM](#)
- [Menguji kebijakan](#)

Mengkonfigurasi kebijakan IAM untuk membatasi dorongan dan penggabungan ke cabang

Anda dapat membuat kebijakan di IAM yang mencegah pengguna memperbarui cabang, termasuk mendorong commit ke cabang dan penggabungan permintaan tarik ke cabang. Untuk melakukannya, kebijakan Anda menggunakan pernyataan bersyarat, sehingga efek dari pernyataan Deny hanya berlaku jika syarat tersebut terpenuhi. API yang Anda sertakan dalam pernyataan Deny menentukan tindakan yang tidak diperbolehkan. Anda dapat mengkonfigurasi kebijakan ini untuk menerapkan hanya satu cabang di repositori, sejumlah cabang di repositori, atau untuk semua cabang yang cocok dengan kriteria di semua repositori di akun Amazon Web Services.

Untuk membuat kebijakan bersyarat untuk cabang

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Di panel navigasi, pilih Kebijakan.
3. Pilih Buat kebijakan.
4. Pilih JSON, dan kemudian tempelkan kebijakan contoh berikut. Ganti nilai `Resource` dengan ARN dari repositori yang berisi cabang yang Anda ingin membatasi akses. Ganti nilai `codecommit:References` dengan referensi ke cabang atau cabang yang Anda ingin membatasi akses. Misalnya, kebijakan ini menolak mendorong commit, menggabungkan cabang, menghapus cabang, menggabungkan permintaan tarik, dan menambahkan file ke cabang bernama dan cabang bernama dalam *main* repositori bernama *prod: MyDemoRepo*

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "codecommit:GitPush",
        "codecommit>DeleteBranch",
        "codecommit:PutFile",
        "codecommit:MergeBranchesByFastForward",
        "codecommit:MergeBranchesBySquash",
        "codecommit:MergeBranchesByThreeWay",
        "codecommit:MergePullRequestByFastForward",
        "codecommit:MergePullRequestBySquash",
        "codecommit:MergePullRequestByThreeWay"
      ]
    }
  ]
}
```

```

    ],
    "Resource": "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",
    "Condition": {
      "StringEqualsIfExists": {
        "codecommit:References": [
          "refs/heads/main",
          "refs/heads/prod"
        ]
      },
      "Null": {
        "codecommit:References": "false"
      }
    }
  }
]
}

```

Cabang di Git hanya (referensi) pointer ke nilai SHA-1 dari kepala commit, yang mengapa syarat tersebut menggunakan References. Parameter pernyataan Null diperlukan dalam setiap kebijakan yang efeknya adalah Deny dan di mana GitPush adalah salah satu tindakan. Ini diperlukan karena cara Git dan `git-receive-pack` bekerja saat mendorong perubahan dari repo lokal ke CodeCommit.

Tip

Untuk membuat kebijakan yang berlaku untuk semua cabang yang bernama utama di semua repositori di akun Amazon Web Services, ubah nilai Resource dari repositori ARN ke sebuah tanda bintang (*).

5. Pilih Tinjau kebijakan. Perbaiki kesalahan dalam pernyataan kebijakan Anda, dan kemudian lanjutkan Buat kebijakan.
6. Ketika JSON divalidasi, halaman Buat kebijakan ditampilkan. Sebuah peringatan muncul di bagian Ringkasan, yang memberitahu Anda bahwa kebijakan ini tidak memberikan izin. Ini yang diharapkan.
 - Di Nama, masukkan nama untuk kebijakan ini, seperti **DenyChangesToMain**.
 - (Opsional) Untuk Deskripsi, masukkan deskripsi tujuan kebijakan. Ini memang opsional, tetapi direkomendasikan.
 - Pilih Buat kebijakan.

Terapkan kebijakan IAM ke grup atau peran IAM

Anda telah membuat kebijakan yang membatasi penggabungan dan penggabungan ke cabang, namun kebijakan tersebut tidak berpengaruh hingga Anda menerapkannya ke pengguna, grup, atau peran IAM. Sebagai praktik terbaik, pertimbangkan untuk menerapkan kebijakan ke grup atau peran IAM. Menerapkan kebijakan untuk pengguna IAM individu tidak menskalakan dengan baik.

Untuk menerapkan kebijakan bersyarat ke sebuah grup atau peran

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Di panel navigasi, jika Anda ingin menerapkan kebijakan ke grup IAM, pilih Grup. Jika Anda ingin menerapkan kebijakan untuk peran yang diambil oleh pengguna, pilih Peran. Pilih nama grup atau peran.
3. Di tab Izin, pilih Lampirkan kebijakan.
4. Pilih kebijakan bersyarat yang Anda buat dari daftar kebijakan, dan kemudian pilih Lampirkan kebijakan.

Untuk informasi selengkapnya, lihat [Melampirkan dan Melepaskan Lampiran Kebijakan IAM](#).

Menguji kebijakan

Anda harus menguji efek dari kebijakan yang telah Anda terapkan pada grup atau peran untuk memastikan tindakan tersebut sesuai harapan. Ada banyak cara Anda dapat melakukan ini. Misalnya, untuk menguji kebijakan yang serupa dengan yang ditunjukkan di atas, Anda dapat:

- Masuk ke CodeCommit konsol dengan pengguna IAM yang merupakan anggota grup IAM yang menerapkan kebijakan tersebut, atau mengambil peran yang menerapkan kebijakan tersebut. Di konsol, tambahkan file di cabang tempat pembatasan berlaku. Anda akan melihat pesan kesalahan ketika Anda mencoba untuk menyimpan atau mengunggah file ke cabang tersebut. Tambahkan file ke cabang yang berbeda. Operasi tersebut seharusnya berhasil.
- Masuk ke CodeCommit konsol dengan pengguna IAM yang merupakan anggota grup IAM yang menerapkan kebijakan tersebut, atau mengambil peran yang menerapkan kebijakan tersebut. Buatlah sebuah permintaan tarik yang bergabung ke cabang di mana pembatasan berlaku. Anda harus dapat membuat permintaan tarik, tetapi mendapatkan kesalahan jika Anda mencoba untuk menggabungkannya.

- Dari terminal atau baris perintah, buat komit di cabang tempat pembatasan berlaku, lalu dorong komit itu ke CodeCommit repositori. Anda akan melihat pesan kesalahan. Commit dan dorongan yang dibuat dari cabang lain seharusnya bekerja seperti biasa.

Lihat detail cabang di AWS CodeCommit

Anda dapat menggunakan CodeCommit konsol untuk melihat detail tentang cabang di CodeCommit repositori. Anda dapat menampilkan tanggal commit terakhir ke cabang, pesan commit, dan lainnya. Anda juga dapat menggunakan AWS CLI atau Git dari repo lokal yang terhubung ke CodeCommit repositori.

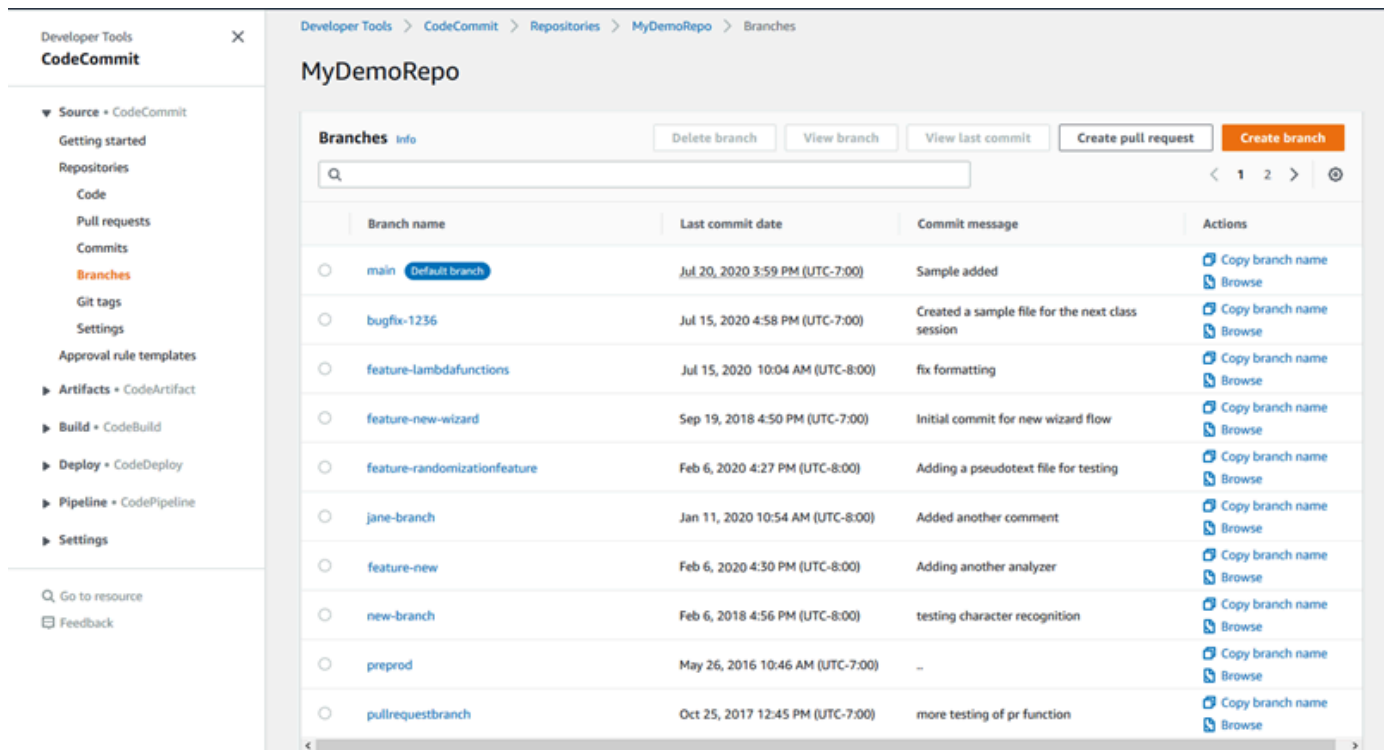
Topik

- [Tampilkan detail cabang \(konsol\)](#)
- [Tampilkan detail cabang \(Git\)](#)
- [Tampilkan detail cabang \(AWS CLI\)](#)

Tampilkan detail cabang (konsol)

Gunakan CodeCommit konsol untuk dengan cepat melihat daftar cabang untuk repositori Anda dan detail tentang cabang.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin menampilkan detail cabang.
3. Di panel navigasi, pilih Cabang.



4. Nama cabang yang digunakan sebagai default untuk repositori ditampilkan di sebelah Cabang default. Untuk menampilkan detail tentang commit terbaru untuk suatu cabang, pilih cabang, lalu pilih Tampilkan commit terakhir. Untuk menampilkan file dan kode di cabang, pilih nama cabang.

Tampilkan detail cabang (Git)

Untuk menggunakan Git dari repo lokal untuk melihat detail tentang cabang pelacakan lokal dan jarak jauh untuk CodeCommit repositori, jalankan perintah. `git branch`

Langkah-langkah berikut ditulis dengan asumsi bahwa Anda telah menghubungkan repo lokal ke repositori. CodeCommit Untuk petunjuk, lihat [Connect ke sebuah repositori](#).

1. Jalankan perintah `git branch`, dengan menentukan opsi `--all`:

```
git branch --all
```

2. Jika berhasil, perintah ini mengembalikan output yang serupa dengan yang berikut:

```
MyNewBranch
* main
remotes/origin/MyNewBranch
```

```
remotes/origin/main
```

Tanda bintang (*) muncul di sebelah cabang yang sedang terbuka. Entri setelah itu adalah referensi pelacakan jarak jauh.

Tip

git branch menunjukkan cabang lokal.

git branch -r menunjukkan cabang jarak jauh.

git checkout **existing-branch-name** mengalihkan ke nama cabang yang ditentukan dan, jika git branch dijalankan segera sesudahnya, menampilkannya dengan tanda bintang (*).

git remote update **remote-name** memperbarui repo lokal Anda dengan daftar cabang CodeCommit repositori yang tersedia. (Untuk mendapatkan daftar nama CodeCommit repositori dan URL-URL mereka, jalankan perintah.) git remote -v

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Tampilkan detail cabang (AWS CLI)

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal file AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk menggunakan AWS CLI untuk melihat detail tentang cabang dalam CodeCommit repositori, jalankan satu atau beberapa perintah berikut:

- Untuk menampilkan daftar nama cabang, jalankan [daftar-cabang](#).
- Untuk menampilkan informasi tentang cabang tertentu, jalankan [get-branch](#).

Untuk menampilkan daftar nama cabang

1. Jalankan list-branches perintah, tentukan nama CodeCommit repositori (dengan opsi). -- repository-name

Tip

Untuk mendapatkan nama repositori, jalankan perintah CodeCommit [list-repositori](#).

Misalnya, untuk melihat detail tentang cabang dalam CodeCommit repositori bernama: MyDemoRepo

```
aws codecommit list-branches --repository-name MyDemoRepo
```

2. Jika berhasil, perintah ini menghasilkan objek `branchNameList`, dengan entri untuk setiap cabang.

Berikut adalah beberapa contoh output berdasarkan perintah sebelumnya:

```
{
  "branches": [
    "MyNewBranch",
    "main"
  ]
}
```

Untuk menampilkan informasi tentang sebuah cabang

1. Jalankan perintah `get-branch`, dengan menentukan:
 - Nama repositori (dengan opsi `--repository-name`).
 - Nama cabang baru (dengan opsi `--branch-name`).

Misalnya, untuk melihat informasi tentang cabang bernama MyNewBranch dalam CodeCommit repositori bernama: MyDemoRepo

```
aws codecommit get-branch --repository-name MyDemoRepo --branch-name MyNewBranch
```

2. Jika berhasil, perintah ini mengeluarkan nama cabang dan ID dari commit terakhir yang dibuat untuk cabang tersebut.

Berikut adalah beberapa contoh output berdasarkan perintah sebelumnya:

```
{
  "branch": {
    "branchName": "MyNewBranch",
    "commitID": "317f8570EXAMPLE"
  }
}
```

Bandingkan dan gabungkan cabang di AWS CodeCommit

Anda dapat menggunakan CodeCommit konsol untuk membandingkan cabang di CodeCommit repositori. Membandingkan cabang membantu Anda dengan cepat melihat perbedaan antara sebuah cabang dan cabang default, atau melihat perbedaan antara dua cabang.

Topik

- [Bandingkan sebuah cabang terhadap cabang default](#)
- [Bandingkan dua cabang tertentu](#)
- [Gabungkan dua cabang \(AWS CLI\)](#)

Bandingkan sebuah cabang terhadap cabang default

Gunakan CodeCommit konsol untuk dengan cepat melihat perbedaan antara cabang dan cabang default untuk repositori Anda.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin membandingkan cabang.
3. Di panel navigasi, pilih Commit, dan kemudian pilih tab Bandingkan commit.
4. Di Tujuan, pilih nama cabang default. Di Sumber, pilih cabang yang ingin Anda bandingkan dengan cabang default. Pilih Bandingkan.

Bandingkan dua cabang tertentu

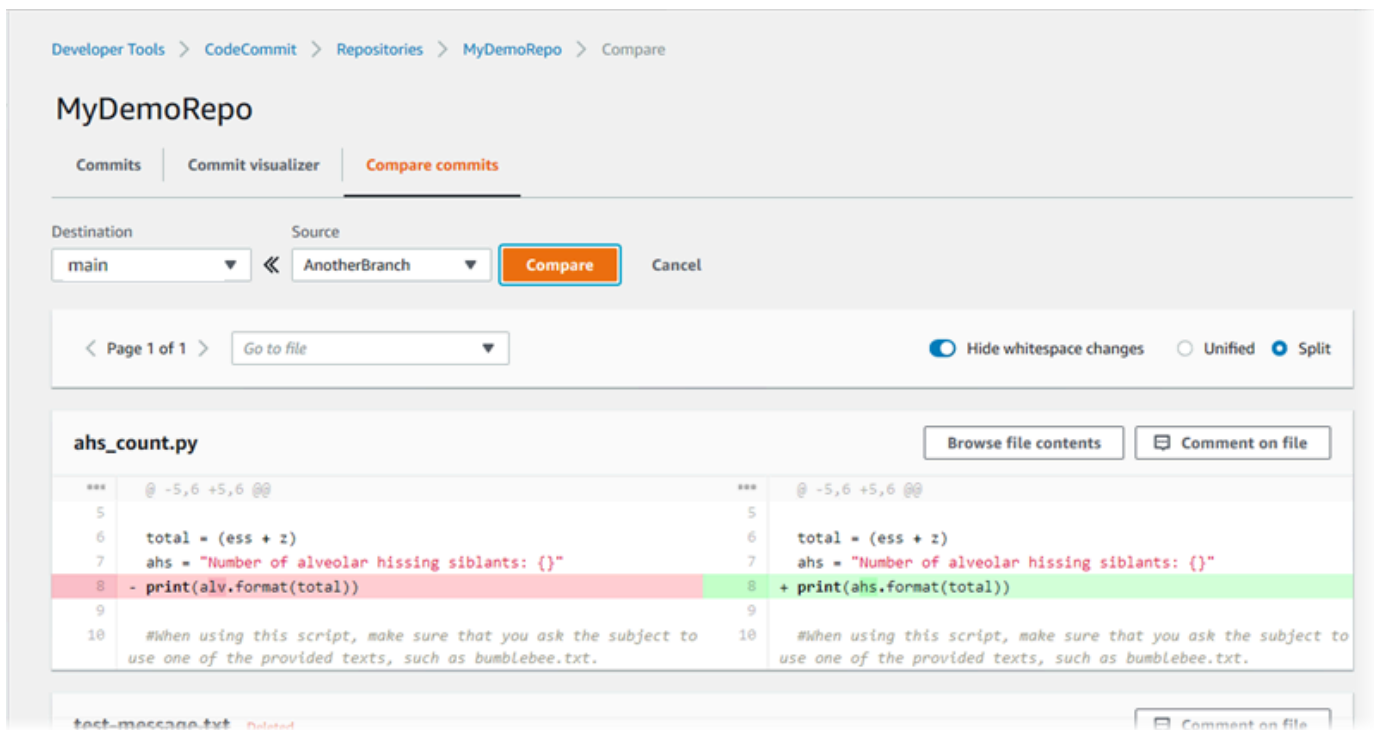
Gunakan CodeCommit konsol untuk melihat perbedaan antara dua cabang yang ingin Anda bandingkan.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.

2. Di Repositori, pilih nama repositori tempat Anda ingin membandingkan cabang.
3. Di panel navigasi, pilih Commit, dan kemudian pilih tab Bandingkan commit.
4. Di Tujuan dan Sumber, pilih dua cabang untuk dibandingkan, lalu pilih Bandingkan. Untuk menampilkan daftar file yang diubah, perluas daftar file yang diubah. Anda dapat menampilkan perubahan pada file yang berdampingan (Split view) atau inline (Unified view).

Note

Jika Anda masuk sebagai pengguna IAM, Anda dapat mengonfigurasi dan menyimpan preferensi Anda untuk menampilkan kode dan pengaturan konsol lainnya. Untuk informasi selengkapnya, lihat [Bekerja dengan preferensi pengguna](#).



Gabungkan dua cabang (AWS CLI)

Anda dapat menggabungkan dua cabang dalam CodeCommit repositori menggunakan AWS CLI menggunakan salah satu strategi penggabungan yang tersedia dengan menjalankan salah satu perintah berikut:

- Untuk menggabungkan dua cabang menggunakan strategi gabungan maju cepat, jalankan perintah [merge-branches-by-fast-forward](#).
- Untuk menggabungkan dua cabang menggunakan strategi gabungan squash, jalankan perintah [merge-branches-by-squash](#).
- Untuk menggabungkan dua cabang menggunakan strategi gabungan tiga arah, jalankan perintah [merge-branches-by-three-way](#).

Anda juga dapat menguji penggabungan dengan menjalankan perintah `create-unreferenced-merge-commit`. Untuk informasi selengkapnya, lihat [Selesaikan Konflik dalam Permintaan Tarik](#).

Note

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal file AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk menggunakan AWS CLI untuk menggabungkan dua cabang dalam repositori CodeCommit

1.

Untuk menggabungkan dua cabang menggunakan strategi gabungan maju cepat, jalankan perintah `merge-branches-by-fast-forward`, dengan menentukan:

- Nama cabang sumber yang berisi perubahan yang ingin Anda gabungkan (dengan opsi `--source-commit-specifier`).
- Nama cabang tujuan di mana Anda ingin menggabungkan perubahan Anda (dengan opsi `--destination-commit-specifier`).
- Nama dari repositori tersebut (dengan opsi `--repository-name`).

Misalnya, untuk menggabungkan cabang sumber bernama *bugfix-1234* ke dalam cabang tujuan bernama *preprod* dalam repositori bernama: *MyDemoRepo*

```
aws codecommit merge-branches-by-fast-forward --source-commit-specifier bugfix-  
bug1234 --destination-commit-specifier preprod --repository-name MyDemoRepo
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
```

```
"commitId": "4f178133EXAMPLE",
"treeId": "389765daEXAMPLE"
}
```

2.

Untuk menggabungkan dua cabang menggunakan strategi gabungan squash, jalankan perintah `merge-branches-by-squash`, dengan menentukan:

- Nama cabang sumber yang berisi perubahan yang ingin Anda gabungkan (dengan opsi `--source-commit-specifier`).
- Nama cabang tujuan di mana Anda ingin menggabungkan perubahan Anda (dengan opsi `--destination-commit-specifier`).
- Nama dari repositori tersebut (dengan opsi `--repository-name`).
- Pesan commit yang harus disertakan (dengan opsi `--commit-message`).
- Nama yang digunakan untuk commit (dengan opsi `--name`).
- Alamat email yang digunakan untuk commit (dengan opsi `--email`).

Misalnya, untuk menggabungkan cabang sumber bernama `bugfix-bug1234` dengan cabang tujuan bernama `bugfix-quarterly` dalam repositori bernama: `MyDemoRepo`

```
aws codecommit merge-branches-by-squash --source-commit-specifier bugfix-bug1234 --
destination-commit-specifier bugfix-quarterly --author-name "Maria Garcia" --email
"maria_garcia@example.com" --commit-message "Merging in fix branches to prepare
for a general patch." --repository-name MyDemoRepo
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "commitId": "4f178133EXAMPLE",
  "treeId": "389765daEXAMPLE"
}
```

3.

Untuk menggabungkan dua cabang menggunakan strategi gabungan tiga arah, jalankan perintah `merge-branches-by-three-way`, dengan menentukan:

- Nama cabang sumber yang berisi perubahan yang ingin Anda gabungkan (dengan opsi `--source-commit-specifier`).

- Nama cabang tujuan di mana Anda ingin menggabungkan perubahan Anda (dengan opsi `--destination-commit-specifier`).
- Nama dari repositori tersebut (dengan opsi `--repository-name`).
- Pesan commit yang harus disertakan (dengan opsi `--commit-message`).
- Nama yang digunakan untuk commit (dengan opsi `--name`).
- Alamat email yang digunakan untuk commit (dengan opsi `--email`).

Misalnya, untuk menggabungkan cabang sumber bernama *main* dengan cabang tujuan bernama *bugfix-1234* dalam repositori bernama: *MyDemoRepo*

```
aws codecommit merge-branches-by-three-way --source-commit-specifier main --
destination-commit-specifier bugfix-bug1234 --author-name "Jorge Souza" --email
"jorge_souza@example.com" --commit-message "Merging changes from main to bugfix
branch before additional testing." --repository-name MyDemoRepo
```

Jika berhasil, perintah ini menghasilkan output yang serupa dengan yang berikut:

```
{
  "commitId": "4f178133EXAMPLE",
  "treeId": "389765daEXAMPLE"
}
```

Ubah pengaturan cabang di AWS CodeCommit

Anda dapat mengubah cabang mana yang akan digunakan sebagai cabang default di AWS CodeCommit konsol atau dengan AWS CLI. Sebagai contoh, jika Anda membuat commit pertama Anda menggunakan klien Git yang mengatur cabang default untuk utama, anda dapat membuat sebuah cabang yang bernama utama, dan kemudian mengubah pengaturan cabang sehingga cabang baru ditetapkan sebagai cabang default untuk repositori tersebut. Untuk mengubah pengaturan cabang lainnya, Anda dapat menggunakan Git dari repo lokal yang terhubung ke CodeCommit repositori.

Topik

- [Ubah default branch \(konsol\)](#)
- [Ubah cabang default \(AWS CLI\)](#)

Ubah default branch (konsol)

Anda dapat menentukan cabang mana yang merupakan cabang default di CodeCommit repositori di konsol. AWS CodeCommit

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin mengubah pengaturan.
3. Pada panel navigasi, silakan pilih Pengaturan.
4. Di Cabang default, pilih daftar drop-down cabang dan pilih cabang yang berbeda. Pilih Simpan.

Tip

- Jika Anda tidak melihat cabang lain dalam daftar drop-down, Anda belum membuat cabang tambahan. Anda tidak dapat mengubah cabang default dari repositori jika repositori hanya memiliki satu cabang. Untuk informasi selengkapnya, lihat [Buat cabang di AWS CodeCommit](#).
- Jika Anda tidak melihat bagian cabang Default tetapi melihat item untuk aturan dan koneksi notifikasi, Anda berada di menu pengaturan umum untuk konsol. Menu pengaturan untuk repositori tercantum di bawah Repositori pada tingkat yang sama dengan permintaan Kode dan Tarik.

Ubah cabang default (AWS CLI)

Untuk menggunakan AWS CLI perintah dengan CodeCommit, instal file AWS CLI. Untuk informasi selengkapnya, lihat [Referensi baris perintah](#).

Untuk menggunakan AWS CLI untuk mengubah pengaturan cabang repositori di CodeCommit repositori, jalankan perintah berikut:

- [update-default-branch](#) untuk mengubah cabang default.

Untuk mengubah cabang default

1. Jalankan perintah `update-default-branch`, dengan menentukan:

- Nama CodeCommit repositori tempat cabang default diperbarui (dengan `--repository-name` opsi).

 Tip

Untuk mendapatkan nama repositori, jalankan perintah CodeCommit [list-repositori](#).

- Nama dari cabang default baru (dengan opsi `--default-branch-name`).

 Tip

Untuk mendapatkan nama cabang, jalankan perintah [list-branches](#).

2. Misalnya, untuk mengubah cabang default ke MyNewBranch dalam CodeCommit repositori bernama: MyDemoRepo

```
aws codecommit update-default-branch --repository-name MyDemoRepo --default-branch-name MyNewBranch
```

Perintah ini menghasilkan output hanya jika ada kesalahan.

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Hapus cabang di AWS CodeCommit

Anda dapat menggunakan CodeCommit konsol untuk menghapus cabang di repositori. Menghapus cabang di CodeCommit tidak menghapus cabang itu di repo lokal, jadi pengguna mungkin terus memiliki salinan cabang itu hingga saat berikutnya mereka menarik perubahan. Untuk menghapus cabang secara lokal dan mendorong perubahan itu ke CodeCommit repositori, gunakan Git dari repo lokal yang terhubung ke repositori. CodeCommit

Menghapus cabang tidak menghapus commit apa pun, tetapi menghapus semua referensi ke commit di cabang itu. Jika Anda menghapus cabang yang berisi commit yang belum digabung ke cabang lain di repositori, Anda tidak dapat mengambil commit tersebut kecuali Anda memiliki ID commit penuh mereka.

Note

Anda tidak dapat menggunakan petunjuk dalam topik ini untuk menghapus cabang default repositori. Jika Anda ingin menghapus cabang default, Anda harus membuat sebuah cabang, menjadikan cabang baru tersebut cabang default, dan kemudian menghapus cabang lama. Untuk informasi selengkapnya, lihat [Buatlah sebuah cabang](#) dan [Ubah pengaturan cabang](#).

Topik

- [Hapuslah sebuah cabang \(konsol\)](#)
- [Hapuslah sebuah cabang \(AWS CLI\)](#)
- [Hapuslah sebuah cabang \(Git\)](#)

Hapuslah sebuah cabang (konsol)

Anda dapat menggunakan CodeCommit konsol untuk menghapus cabang di CodeCommit repositori.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori tempat Anda ingin menghapus cabang.
3. Di panel navigasi, pilih Cabang.
4. Temukan nama cabang yang ingin Anda hapus, pilih Hapus cabang, dan konfirmasi pilihan Anda.

Hapuslah sebuah cabang (AWS CLI)

Anda dapat menggunakan AWS CLI untuk menghapus cabang di CodeCommit repositori, jika cabang itu bukan cabang default untuk repositori. Untuk informasi lebih lanjut tentang menginstal dan menggunakan AWS CLI, lihat [Referensi baris perintah](#).

1. Pada terminal atau baris perintah, jalankan perintah delete-branch, dengan menentukan:
 - Nama CodeCommit repositori tempat cabang akan dihapus (dengan --repository-name opsi).

Tip

Untuk mendapatkan nama repositori, jalankan perintah CodeCommit [list-repositori](#).

- Nama cabang yang akan dihapus (dengan opsi `branch-name`).

 Tip

Untuk mendapatkan nama cabang, jalankan perintah [list-branches](#).

2. Misalnya, untuk menghapus cabang bernama `MyNewBranch` dalam CodeCommit repositori bernama: `MyDemoRepo`

```
aws codecommit delete-branch --repository-name MyDemoRepo --branch-name MyNewBranch
```

Perintah ini mengembalikan informasi tentang cabang yang dihapus, termasuk nama cabang yang dihapus dan ID commit penuh dari commit yang dulunya merupakan kepala cabang. Sebagai contoh:

```
"deletedBranch": {
  "branchName": "MyNewBranch",
  "commitId": "317f8570EXAMPLE"
}
```

Hapuslah sebuah cabang (Git)

Ikuti langkah-langkah ini untuk menggunakan Git dari repo lokal untuk menghapus cabang dalam CodeCommit repositori.

Langkah-langkah ini ditulis dengan asumsi bahwa Anda telah menghubungkan repo lokal ke repositori. CodeCommit Untuk petunjuk, lihat [Connect ke sebuah repositori](#).

1. Untuk menghapus cabang dari repo lokal, jalankan `git branch -D branch-name` di mana ***branch-name*** adalah nama dari cabang yang ingin Anda hapus.

 Tip

Untuk mendapatkan daftar nama cabang, jalankan `git branch --all`.

Misalnya, untuk menghapus cabang di repo lokal yang bernama `MyNewBranch`:

```
git branch -D MyNewBranch
```

2. Untuk menghapus cabang dari CodeCommit repositori, jalankan git push **remote-name** --delete **branch-name** perintah di mana nama *jarak jauh adalah nama panggilan yang* digunakan repo lokal untuk repositori dan nama cabang adalah *nama cabang* yang ingin Anda hapus dari CodeCommit repositori. CodeCommit

 Tip

Untuk mendapatkan daftar nama CodeCommit repositori dan URL mereka, jalankan perintah. git remote -v

Misalnya, untuk menghapus cabang bernama MyNewBranch dalam CodeCommit repositori bernama: origin

```
git push origin --delete MyNewBranch
```

 Tip

Perintah ini tidak menghapus cabang jika cabang tersebut adalah cabang default.

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Bekerja dengan preferensi pengguna

Anda dapat menggunakan konsol AWS CodeCommit untuk mengkonfigurasi beberapa pengaturan default. Misalnya, Anda dapat mengubah preferensi Anda untuk melihat perubahan kode baik secara inline atau dalam tampilan terpisah. Ketika Anda membuat perubahan pada salah satu pengaturan ini, konsol AWS CodeCommit mengatur cookie di peramban Anda yang menyimpan dan menerapkan pilihan Anda setiap kali Anda menggunakan konsol tersebut. Preferensi ini diterapkan ke semua repositori di semua wilayah setiap kali Anda mengakses konsol AWS CodeCommit menggunakan peramban tersebut. Preferensi pengaturan ini tidak spesifik-repositori atau spesifik-wilayah. Mereka tidak memiliki efek apapun pada interaksi Anda dengan API AWS CLI, AWS CodeCommit, atau layanan lain yang berinteraksi dengan AWS CodeCommit.

Note

Cookie preferensi pengguna spesifik untuk peramban. Jika Anda menghapus cookie dari peramban, maka preferensi Anda akan dihapus. Demikian pula, jika Anda menggunakan peramban yang berbeda untuk mengakses sebuah repositori, maka peramban tersebut tidak memiliki akses ke cookie peramban lain. Preferensi Anda tidak dipertahankan.

Preferensi pengguna meliputi:

- Ketika melihat perubahan dalam kode, apakah akan menggunakan tampilan Terpadu atau Split, dan apakah akan menampilkan atau menyembunyikan perubahan spasi.
- Saat melihat, mengedit, atau menulis kode, apakah akan menggunakan latar belakang yang terang atau latar belakang gelap di jendela editor kode.

Tidak ada halaman untuk mengatur preferensi Anda. Sebagai gantinya, di mana pun Anda mengubah preferensi di konsol, seperti bagaimana cara melihat perubahan kode, perubahan itu disimpan dan diterapkan di mana pun yang sesuai.

Migrasi ke AWS CodeCommit

Anda dapat memigrasi sebuah repositori Git ke repositori CodeCommit dengan beberapa cara: dengan melakukan kloning, melakukan mirroring, memigrasi semua atau hanya beberapa cabang saja, dan seterusnya. Anda juga dapat memigrasi konten lokal yang tidak di-versioning di komputer Anda ke CodeCommit.

Topik berikut menunjukkan beberapa cara untuk memigrasi sebuah repositori. Langkah-langkah yang Anda lakukan mungkin berbeda-beda, tergantung pada jenis, gaya, atau kompleksitas repositori Anda dan keputusan yang Anda buat tentang apa dan bagaimana Anda ingin bermigrasi. Untuk repositori yang sangat besar, Anda mungkin ingin mempertimbangkan untuk melakukan [migrasi secara bertahap](#).

Note

Anda dapat bermigrasi ke CodeCommit dari sistem kontrol versi lain, seperti Perforce, Subversion, atau TFS, tetapi Anda harus terlebih dahulu bermigrasi ke Git.

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Atau, Anda dapat meninjau informasi tentang [migrasi ke Git](#) di buku Pro Git yang ditulis oleh Scott Chacon dan Ben Straub.

Topik

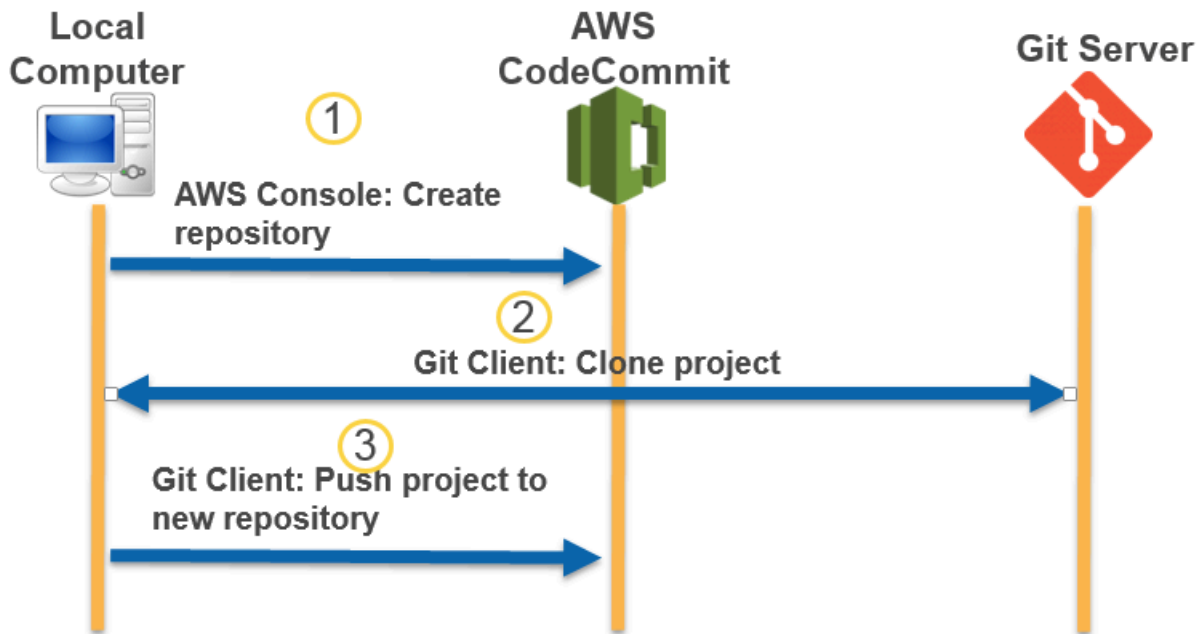
- [Memigrasikan repositori Git ke AWS CodeCommit](#)
- [Memigrasi konten lokal atau tanpa versi ke AWS CodeCommit](#)
- [Memigrasi repositori secara bertahap](#)

Memigrasikan repositori Git ke AWS CodeCommit

Anda dapat memigrasikan repositori Git yang ada ke repositori. CodeCommit Prosedur dalam topik ini menunjukkan kepada Anda cara memigrasikan proyek yang dihosting di repositori Git lain ke CodeCommit. Sebagai bagian dari proses ini, Anda:

- Selesaikan pengaturan awal yang diperlukan untuk CodeCommit.
- Buat CodeCommit repositori.
- Kloning repositori dan dorong ke CodeCommit

- Lihat file di CodeCommit repositori.
- Bagikan CodeCommit repositori dengan tim Anda.



Topik

- [Langkah 0: Pengaturan diperlukan untuk akses ke CodeCommit](#)
- [Langkah 1: Buat CodeCommit repositori](#)
- [Langkah 2: Kloning repositori dan dorong ke repositori CodeCommit](#)
- [Langkah 3: Lihat file di CodeCommit](#)
- [Langkah 4: Bagikan CodeCommit repositori](#)


Langkah 0: Pengaturan diperlukan untuk akses ke CodeCommit

Sebelum Anda dapat memigrasikan repositori ke CodeCommit, Anda harus membuat dan mengkonfigurasi pengguna IAM untuk CodeCommit dan mengkonfigurasi komputer lokal Anda untuk akses. Anda juga harus menginstal AWS CLI to manage CodeCommit. Meskipun Anda dapat melakukan sebagian besar CodeCommit tugas tanpa itu, AWS CLI menawarkan fleksibilitas saat bekerja dengan Git di baris perintah atau terminal.

Jika Anda sudah siap CodeCommit, Anda dapat melompat ke depan [Langkah 1: Buat CodeCommit repositori](#).

Untuk membuat dan mengkonfigurasi pengguna IAM untuk mengakses CodeCommit

1. Buat akun Amazon Web Services dengan membuka <http://aws.amazon.com> dan memilih Mendaftar.
2. Membuat pengguna IAM, atau menggunakan yang sudah ada, di akun Amazon Web Services Anda. Pastikan Anda memiliki access key ID dan secret access key yang terkait dengan pengguna IAM. Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di akun Amazon Web Services Anda](#).

 Note

CodeCommit membutuhkan AWS Key Management Service. Jika Anda menggunakan pengguna IAM yang sudah ada, pastikan tidak ada kebijakan yang dilampirkan pada pengguna yang secara tegas menolak AWS KMS tindakan yang diperlukan oleh CodeCommit Untuk informasi selengkapnya, lihat [AWS KMS dan enkripsi](#).

3. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
4. Di konsol IAM, di panel navigasi, pilih Pengguna, lalu pilih pengguna IAM yang ingin Anda konfigurasi untuk akses. CodeCommit
5. Pada tab Izin, pilih Tambahkan izin.
6. Dalam Berikan izin, pilih Lampirkan kebijakan yang ada secara langsung.
7. Dari daftar kebijakan, pilih AWSCodeCommitPowerUser atau kebijakan terkelola lainnya untuk CodeCommit akses. Untuk informasi selengkapnya, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Setelah memilih kebijakan yang ingin dilampirkan, pilih Berikutnya: Tinjau untuk meninjau daftar kebijakan untuk dilampirkan ke pengguna IAM. Jika daftar sudah benar, pilih Tambahkan izin.

Untuk informasi selengkapnya tentang kebijakan CodeCommit terkelola dan berbagi akses ke repositori dengan grup dan pengguna lain, lihat [Bagikan sebuah repositori](#) dan [Kontrol autentikasi dan akses untuk AWS CodeCommit](#)

Untuk menginstal dan mengonfigurasi AWS CLI

1. Pada komputer lokal Anda, unduh dan instal AWS CLI. Ini adalah prasyarat untuk berinteraksi dengan CodeCommit dari baris perintah. Kami merekomendasikan agar Anda menginstal AWS CLI versi 2. Ini adalah versi utama terbaru dari AWS CLI dan mendukung semua fitur terbaru. Ini adalah satu-satunya versi AWS CLI yang mendukung penggunaan akun root, akses federasi, atau kredensial sementara dengan git-remote-codecommit.

Untuk informasi selengkapnya, lihat [Menyiapkan Antarmuka Baris Perintah AWS](#).

Note

CodeCommit hanya berfungsi dengan AWS CLI versi 1.7.38 dan yang lebih baru. Sebagai praktik terbaik, instal atau tingkatkan AWS CLI untuk versi terbaru yang tersedia. Untuk menentukan versi AWS CLI yang telah Anda instal, jalankan perintah `aws --version`.

Untuk meng-upgrade versi lama AWS CLI ke versi terbaru, lihat [Menginstal AWS Command Line Interface](#).

2. Jalankan perintah ini untuk memverifikasi bahwa CodeCommit perintah AWS CLI untuk diinstal.

```
aws codecommit help
```

Perintah ini mengembalikan daftar CodeCommit perintah.

3. Konfigurasi AWS CLI dengan profil dengan menggunakan perintah `configure`, sebagai berikut:

```
aws configure
```

Saat diminta, tentukan kunci AWS akses dan kunci akses AWS rahasia pengguna IAM untuk digunakan. CodeCommit Juga, pastikan untuk menentukan Wilayah AWS di mana repositori ada, seperti `us-east-2`. Saat diminta untuk format output default, tentukan `json`. Misalnya, jika Anda mengonfigurasi profil untuk pengguna IAM:

```
AWS Access Key ID [None]: Type your IAM user AWS access key ID here, and then press Enter
```

```
AWS Secret Access Key [None]: Type your IAM user AWS secret access key here, and then press Enter
```

Default region name [None]: *Type a supported region for CodeCommit here, and then press Enter*

Default output format [None]: *Type json here, and then press Enter*

Untuk informasi selengkapnya tentang membuat dan mengonfigurasi profil untuk digunakan dengan AWS CLI, lihat berikut ini:

- [Profil Bernama](#)
- [Menggunakan Peran IAM dalam AWS CLI](#)
- [Tetapkan perintah](#)
- [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#)

Untuk terkoneksi ke repositori atau sumber daya di Wilayah AWS lain, Anda harus mengonfigurasi ulang AWS CLI dengan nama wilayah default. Nama Wilayah default yang didukung untuk CodeCommit menyertakan:

- us-east-2
- us-east-1
- eu-west-1
- us-west-2
- ap-northeast-1
- ap-southeast-1
- ap-southeast-2
- ap-southeast-3
- eu-central-1
- eu-central-1
- ap-northeast-2
- sa-east-1
- us-west-1
- eu-west-2
- ap-south-1
- ap-south-1

- us-gov-west-1
- us-gov-east-1
- eu-north-1
- ap-east-1
- me-south-1
- cn-north-1
- cn-northwest-1
- eu-south-1
- ap-northeast-3
- af-south-1
- ll-sentral-1

Untuk informasi lebih lanjut tentang CodeCommit dan Wilayah AWS, lihat [Wilayah dan titik akhir koneksi Git](#). Untuk informasi selengkapnya tentang IAM, access key, dan kunci rahasia, lihat [Bagaimana Saya Mendapatkan Kredensial?](#) dan [Mengelola Access Key untuk Pengguna IAM](#). Untuk informasi selengkapnya tentang AWS CLI dan profil, lihat [Profil Bernama](#).

Selanjutnya, Anda harus menginstal Git.

- Untuk Linux, macOS, atau Unix:

Untuk bekerja dengan file, commit, dan informasi lainnya di CodeCommit repositori, Anda harus menginstal Git pada mesin lokal Anda. CodeCommit mendukung Git versi 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git.

Untuk menginstal Git, kami merekomendasikan situs web seperti [Pengunduhan Git](#).

Note


Git adalah platform yang berkembang dan diperbarui secara teratur. Terkadang, perubahan fitur dapat memengaruhi cara kerjanya CodeCommit. Jika Anda mengalami masalah dengan versi Git tertentu dan CodeCommit, tinjau informasinya di [Memecahkan masalah](#).

- Untuk Windows:

Untuk bekerja dengan file, commit, dan informasi lainnya di CodeCommit repositori, Anda harus menginstal Git pada mesin lokal Anda. CodeCommit mendukung Git versi 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git.

Untuk menginstal Git, kami merekomendasikan situs web seperti [Git untuk Windows](#). Jika Anda menggunakan link ini untuk menginstal Git, Anda dapat menerima semua pengaturan default instalasi kecuali untuk berikut:

- Ketika diminta selama langkah Menyesuaikan lingkungan PATH Anda, pilih opsi untuk menggunakan Git dari baris perintah.
- (Opsional) Jika Anda bermaksud menggunakan HTTPS dengan pembantu kredensi yang disertakan dalam AWS CLI alih-alih mengonfigurasi kredensi Git CodeCommit untuk, pada halaman Mengonfigurasi opsi tambahan, pastikan opsi Aktifkan Pengelola Kredensial Git dihapus. Git Credential Manager hanya kompatibel dengan CodeCommit jika pengguna IAM mengonfigurasi kredensi Git. Untuk informasi selengkapnya, silakan lihat [Untuk pengguna HTTPS menggunakan kredensial Git](#) dan [Git untuk Windows: Saya menginstal Git untuk Windows, tapi akses saya ke repositori saya ditolak \(403\)](#).

 Note

Git adalah platform yang berkembang dan diperbarui secara teratur. Terkadang, perubahan fitur dapat memengaruhi cara kerjanya CodeCommit. Jika Anda mengalami masalah dengan versi Git tertentu dan CodeCommit, tinjau informasinya di [Memecahkan masalah](#).

CodeCommit mendukung otentikasi HTTPS dan SSH. Untuk menyelesaikan penyiapan, Anda harus mengonfigurasi kredensi Git untuk CodeCommit (HTTPS, direkomendasikan untuk sebagian besar pengguna), key pair SSH yang akan digunakan saat mengakses CodeCommit (SSH), git-remote-codecommit (direkomendasikan untuk pengguna yang menggunakan akses federasi), atau pembantu kredensial yang disertakan dalam (HTTPS). AWS CLI

- Untuk kredensial Git pada semua sistem operasi yang didukung, lihat [Langkah 3: Buat kredensi Git untuk koneksi HTTPS CodeCommit](#).

- Untuk SSH di Linux, macOS, atau Unix, lihat [SSH dan Linux, macOS, atau Unix: Siapkan kunci publik dan privat untuk Git dan CodeCommit](#).
- Untuk SSH di Windows, lihat [Langkah 3: Siapkan kunci publik dan privat untuk Git dan CodeCommit](#).
- Untuk git-remote-codecommit, lihat [Langkah-langkah penyiapan untuk koneksi HTTPS ke AWS CodeCommit bersama git-remote-codecommit](#).
- Untuk pembantu kredensial di Linux, macOS, atau Unix, lihat [Mengatur Credential Helper \(Linux, macOS, atau Unix\)](#).
- Untuk pembantu kredensial di Windows, lihat [Mengatur Credential Helper \(Windows\)](#).

Langkah 1: Buat CodeCommit repositori

Di bagian ini, Anda menggunakan CodeCommit konsol untuk membuat CodeCommit repositori yang Anda gunakan untuk sisa tutorial ini. Untuk menggunakan AWS CLI untuk membuat repositori, lihat [Buatlah sebuah repositori \(AWS CLI\)](#).

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di pemilih wilayah, pilih Wilayah AWS tempat Anda ingin membuat repositori tersebut. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Pada halaman Repositori, pilih Buat repositori.
4. Pada halaman Buat repositori, di Nama repositori, masukkan nama untuk repositori tersebut.

Note

Nama repositori sensitif huruf besar dan kecil. Nama harus unik di Wilayah AWS untuk akun Amazon Web Services Anda.

5. (Opsional) Di Deskripsi, masukkan deskripsi untuk repositori tersebut. Ini dapat membantu Anda dan pengguna lain mengidentifikasi tujuan repositori tersebut.

Note

Bidang deskripsi menampilkan Markdown di konsol dan menerima semua karakter HTML dan karakter Unicode yang valid. Jika Anda adalah pengembang aplikasi yang menggunakan BatchGetRepositories API GetRepository atau dan Anda

berencana untuk menampilkan bidang deskripsi repositori di browser web, lihat Referensi [CodeCommit API](#).

6. (Opsional) Pilih Tambahkan tag untuk menambahkan satu atau lebih tag repositori (label atribut khusus yang membantu Anda mengatur dan mengelola sumber daya AWS Anda) ke repositori Anda. Untuk informasi selengkapnya, lihat [Menandai repositori di AWS CodeCommit](#).
7. (Opsional) Perluas konfigurasi tambahan untuk menentukan apakah akan menggunakan default Kunci yang dikelola AWS atau kunci yang dikelola pelanggan Anda sendiri untuk mengenkripsi dan mendekripsi data dalam repositori ini. Jika Anda memilih untuk menggunakan kunci yang dikelola pelanggan Anda sendiri, Anda harus memastikan bahwa itu tersedia di Wilayah AWS tempat Anda membuat repositori, dan bahwa kuncinya aktif. Untuk informasi selengkapnya, lihat [AWS Key Management Service dan enkripsi untuk repositori AWS CodeCommit](#).
8. (Opsional) Pilih Aktifkan Amazon CodeGuru Reviewer untuk Java dan Python jika repositori ini berisi kode Java atau Python, dan Anda ingin Reviewer menganalisisnya. CodeGuru CodeGuru Reviewer menggunakan beberapa model pembelajaran mesin untuk menemukan cacat kode dan menyarankan perbaikan dan perbaikan dalam permintaan tarik. Untuk informasi selengkapnya, lihat [Panduan Pengguna Amazon CodeGuru Reviewer](#).
9. Pilih Buat.

[Developer Tools](#) > [CodeCommit](#) > [Repositories](#) > Create repository

Create repository

Create a secure repository to store and share your code. Begin by typing a repository name and a description for your repository. Repository names are included in the URLs for that repository.

Repository settings

Repository name

100 characters maximum. Other limits apply.

createRepository.form.field.repositoryDescription.label - optional

1,000 characters maximum

[Cancel](#) [Create](#)

Setelah dibuat, repositori muncul di daftar Repositori. Di kolom URL, pilih ikon salin, lalu pilih protokol (SSH atau HTTPS) yang akan digunakan untuk CodeCommit terhubung. Salin URL.

Misalnya, jika Anda menamai repositori Anda *MyClonedRepository* dan Anda menggunakan kredensi Git dengan HTTPS di Wilayah AS Timur (Ohio), URL akan terlihat seperti berikut:

```
https://git-codecommit.us-east-2.amazonaws.com/MyClonedRepository
```

Anda membutuhkan URL ini nanti di [Langkah 2: Kloning repositori dan dorong ke repositori CodeCommit](#).

Langkah 2: Kloning repositori dan dorong ke repositori CodeCommit

Pada bagian ini, Anda mengkloning repositori Git ke komputer lokal Anda, menciptakan apa yang disebut repo lokal. Anda kemudian mendorong konten repo lokal ke CodeCommit repositori yang Anda buat sebelumnya.

1. Dari terminal atau command prompt di komputer lokal Anda, jalankan git clone perintah dengan `--mirror` opsi untuk mengkloning salinan kosong dari repositori jarak jauh ke folder baru bernama `aws-codecommit-demo`. Ini adalah repo polos dimaksudkan hanya untuk migrasi. Ini bukan repo lokal untuk berinteraksi dengan repositori yang dimigrasi di CodeCommit Anda dapat membuatnya nanti, setelah migrasi ke CodeCommit selesai.

Contoh berikut mengkloning aplikasi demo yang dihosting di GitHub (<https://github.com/aws-labs/aws-demo-php-simple-app.git>) ke repo lokal di direktori bernama `aws-codecommit-demo`

```
git clone --mirror https://github.com/aws-labs/aws-demo-php-simple-app.git aws-codecommit-demo
```

2. Ubah direktori ke direktori tempat Anda membuat klon.

```
cd aws-codecommit-demo
```

3. Jalankan git push perintah, tentukan URL dan nama CodeCommit repositori tujuan dan opsi. `--all` (Ini adalah URL yang Anda salin di [Langkah 1: Buat CodeCommit repositori](#)).

Misalnya, jika Anda menamai repositori Anda `MyClonedRepository` dan Anda diatur untuk menggunakan HTTPS, Anda akan menjalankan perintah berikut:

```
git push https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyClonedRepository --all
```

Note

Opsi `--all` hanya mengirim semua cabang untuk repositori. Opsi tersebut tidak mengirim referensi lain, seperti tanda. Jika Anda ingin mengirim tanda, tunggu sampai pengiriman awal selesai, dan kemudian kirim lagi, kali ini menggunakan pilihan `--tags`:

```
git push ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyClonedRepository --tags
```

Untuk informasi lebih lanjut, lihat [Pengiriman git](#) di situs web Git. Untuk informasi tentang mengirim repositori besar, terutama ketika mengirim semua referensi sekaligus (misalnya, dengan opsi `--mirror`), lihat [Memigrasi repositori secara bertahap](#).

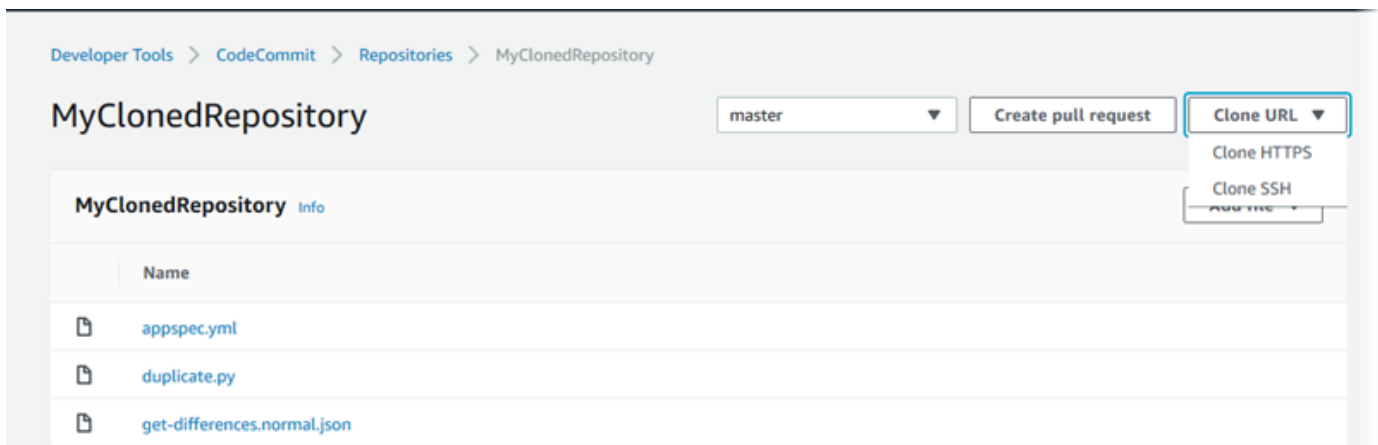
Anda dapat menghapus `aws-codecommit-demo` folder dan isinya setelah Anda memigrasikan repositori ke CodeCommit Untuk membuat repo lokal dengan semua referensi yang benar untuk bekerja dengan repositori di CodeCommit, jalankan `git clone` perintah tanpa opsi: `--mirror`

```
git clone https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyClonedRepository
```

Langkah 3: Lihat file di CodeCommit

Setelah Anda mendorong konten direktori Anda, Anda dapat menggunakan CodeCommit konsol untuk dengan cepat melihat semua file di repositori itu.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori (misalnya, `MyClonedRepository`)
3. Lihat file dalam repositori untuk cabang, URL klon, pengaturan, dan banyak lagi.



Langkah 4: Bagikan CodeCommit repositori

Saat Anda membuat repositori di CodeCommit, dua titik akhir dihasilkan: satu untuk koneksi HTTPS dan satu untuk koneksi SSH. Keduanya menyediakan koneksi yang aman melalui sebuah jaringan. Para pengguna Anda dapat menggunakan salah satu protokol. Kedua titik akhir tetap aktif terlepas dari protokol mana yang Anda rekomendasikan untuk para pengguna Anda. Sebelum Anda dapat membagikan repositori dengan orang lain, Anda harus membuat kebijakan IAM yang memungkinkan pengguna lain mengakses repositori Anda. Berikan petunjuk akses tersebut kepada pengguna Anda.

Buatlah kebijakan terkelola pelanggan untuk repositori Anda

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Pada panel navigasi Dasbor, pilih Kebijakan, dan kemudian pilih Buat Kebijakan.
3. Pada halaman Buat Kebijakan, pilih Impor kebijakan terkelola.
4. Pada halaman Impor kebijakan terkelola, di Kebijakan filter, masukkan **AWSCodeCommitPowerUser**. Pilih tombol di samping nama kebijakan, lalu pilih Impor.
5. Di halaman Buat kebijakan, pilih JSON. Ganti bagian "*" dari Resource baris untuk CodeCommit tindakan dengan Nama Sumber Daya Amazon (ARN) dari CodeCommit repositori, seperti yang ditunjukkan di sini:

```
"Resource": [  
  "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo"  
]
```

Tip

Untuk menemukan ARN untuk CodeCommit repositori, buka CodeCommit konsol, pilih nama repositori dari daftar, lalu pilih Pengaturan. Untuk informasi selengkapnya, lihat [Tampilkan detail repositori](#).

Jika Anda ingin kebijakan ini diterapkan ke lebih dari satu repositori, tambahkan setiap repositori sebagai sumber daya dengan menentukan ARN-nya. Sertakan koma antara setiap pernyataan sumber daya, seperti yang ditunjukkan di sini:

```
"Resource": [  
  "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",  
  "arn:aws:codecommit:us-east-2:111111111111:MyOtherDemoRepo"  
]
```

Setelah Anda selesai mengedit, pilih Tinjau kebijakan.

6. Pada halaman Kebijakan Tinjauan, di Nama, masukkan nama baru untuk kebijakan (misalnya, **AWSCodeCommitPowerUser- MyDemoRepo**). Secara opsional berikan deskripsi untuk kebijakan ini.

7. Pilih Buat Kebijakan.

Untuk mengelola akses ke repositori Anda, buatlah sebuah grup IAM untuk para penggunanya, tambahkan pengguna IAM ke grup tersebut, lalu lampirkan kebijakan terkelola pelanggan yang Anda buat pada langkah sebelumnya. Lampirkan kebijakan lain yang diperlukan untuk akses, seperti `IamUsersShkeys` atau `IAMSelfManageServiceSpecificCredentials`

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Dalam area navigasi Dasbor, pilih Grup, dan kemudian pilih Buat Grup Baru.
3. Pada halaman Setel Nama Grup, di Nama Grup, masukkan nama untuk grup (misalnya, *MyDemoRepoGroup*), lalu pilih Langkah Berikutnya. Pertimbangkan memasukkan nama repositori sebagai bagian dari nama grup.

Note

Nama ini harus unik di seluruh akun Amazon Web Services.

4. Pilih kotak di sebelah kebijakan terkelola pelanggan yang Anda buat di bagian sebelumnya (misalnya, `AWSCodeCommitPowerUser- MyDemoRepo`).
5. Pada halaman Tinjauan, pilih Buat Grup. IAM membuat grup ini dengan kebijakan tertentu yang sudah terlampir. Grup tersebut muncul dalam daftar grup yang terkait dengan akun Amazon Web Services Anda.
6. Pilih nama grup Anda dari daftar.
7. Di halaman ringkasan grup, pilih tab Pengguna, dan kemudian pilih Tambahkan Pengguna ke Grup. Pada daftar yang menampilkan semua pengguna yang terkait dengan akun Amazon Web Services Anda, pilih kotak di sebelah pengguna yang ingin Anda izinkan akses ke CodeCommit repositori, lalu pilih Tambah Pengguna.

Tip

Anda dapat menggunakan kotak Pencarian untuk menemukan pengguna dengan cepat berdasarkan nama.

8. Bila Anda telah menambahkan pengguna Anda, tutup konsol IAM tersebut.

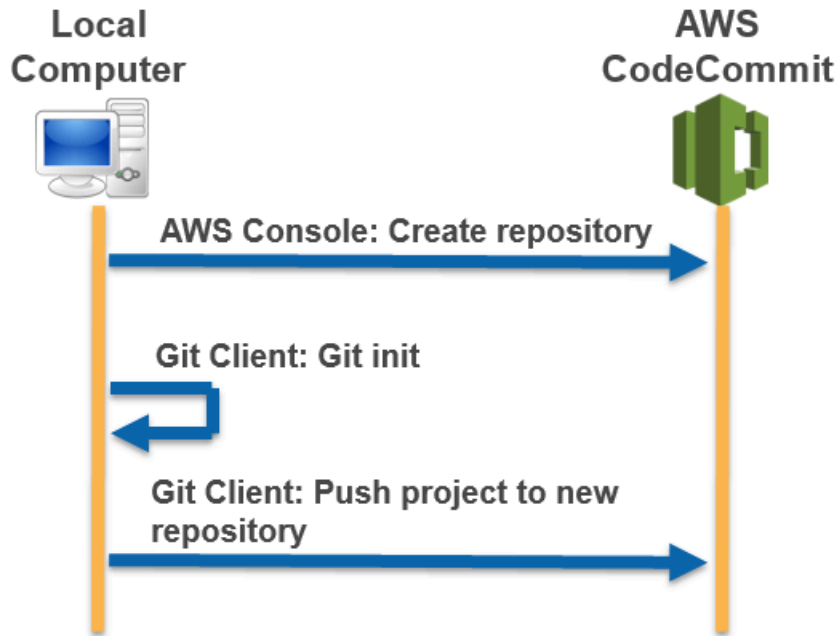
Setelah Anda membuat pengguna IAM untuk mengakses CodeCommit menggunakan grup kebijakan dan kebijakan yang Anda konfigurasi, kirimkan informasi yang diperlukan kepada pengguna tersebut untuk terhubung ke repositori.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Dalam pemilih wilayah, pilih Wilayah AWS di mana repositori dibuat. Repositori adalah khusus untuk sebuah Wilayah AWS. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Pada halaman Repositori, pilih repositori yang Anda ingin bagikan.
4. Dalam URL Klon, pilih protokol yang Anda inginkan agar digunakan oleh para pengguna Anda. Ini menyalin URL klon untuk protokol koneksi.
5. Kirimkan URL klon kepada para pengguna Anda beserta petunjuk lainnya, seperti menginstal AWS CLI, mengkonfigurasi profil, atau menginstal Git. Pastikan untuk menyertakan informasi konfigurasi untuk protokol koneksi (misalnya, HTTPS).

Memigrasi konten lokal atau tanpa versi ke AWS CodeCommit

Prosedur dalam topik ini menunjukkan cara memigrasikan proyek atau konten lokal yang ada di komputer Anda ke CodeCommit repositori. Sebagai bagian dari proses ini, Anda:

- Selesaikan pengaturan awal yang diperlukan untuk CodeCommit.
- Buat CodeCommit repositori.
- Tempatkan folder lokal di bawah kontrol versi Git dan dorong isi folder itu ke CodeCommit repositori.
- Lihat file di CodeCommit repositori.
- Bagikan CodeCommit repositori dengan tim Anda.



Topik

- [Langkah 0: Pengaturan diperlukan untuk akses ke CodeCommit](#)
- [Langkah 1: Buat CodeCommit repositori](#)
- [Langkah 2: Migrasikan konten lokal ke repositori CodeCommit](#)
- [Langkah 3: Lihat file di CodeCommit](#)
- [Langkah 4: Bagikan CodeCommit repositori](#)


Langkah 0: Pengaturan diperlukan untuk akses ke CodeCommit

Sebelum Anda dapat memigrasikan konten lokal ke CodeCommit, Anda harus membuat dan mengkonfigurasi pengguna IAM untuk CodeCommit dan mengkonfigurasi komputer lokal Anda untuk akses. Anda juga harus menginstal AWS CLI to manage CodeCommit. Meskipun Anda dapat melakukan sebagian besar CodeCommit tugas tanpa itu, AWS CLI menawarkan fleksibilitas saat bekerja dengan Git.

Jika Anda sudah siap CodeCommit, Anda dapat melompat ke depan [Langkah 1: Buat CodeCommit repositori](#).

Untuk membuat dan mengkonfigurasi pengguna IAM untuk mengakses CodeCommit

1. Buat akun Amazon Web Services dengan membuka <http://aws.amazon.com> dan memilih Mendaftar.
2. Membuat pengguna IAM, atau menggunakan yang sudah ada, di akun Amazon Web Services Anda. Pastikan Anda memiliki access key ID dan secret access key yang terkait dengan pengguna IAM. Untuk informasi selengkapnya, lihat [Membuat pengguna IAM di akun Amazon Web Services Anda](#).

 Note

CodeCommit membutuhkan AWS Key Management Service. Jika Anda menggunakan pengguna IAM yang sudah ada, pastikan tidak ada kebijakan yang dilampirkan pada pengguna yang secara tegas menolak AWS KMS tindakan yang diperlukan oleh CodeCommit Untuk informasi selengkapnya, lihat [AWS KMS dan enkripsi](#).

3. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
4. Di konsol IAM, di panel navigasi, pilih Pengguna, lalu pilih pengguna IAM yang ingin Anda konfigurasi untuk akses. CodeCommit
5. Pada tab Izin, pilih Tambahkan izin.
6. Dalam Berikan izin, pilih Lampirkan kebijakan yang ada secara langsung.
7. Dari daftar kebijakan, pilih AWSCodeCommitPowerUser atau kebijakan terkelola lainnya untuk CodeCommit akses. Untuk informasi selengkapnya, lihat [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#).

Setelah memilih kebijakan yang ingin dilampirkan, pilih Berikutnya: Tinjau untuk meninjau daftar kebijakan untuk dilampirkan ke pengguna IAM. Jika daftar sudah benar, pilih Tambahkan izin.


Untuk informasi selengkapnya tentang kebijakan CodeCommit terkelola dan berbagi akses ke repositori dengan grup dan pengguna lain, lihat [Bagikan sebuah repositori](#) dan [Kontrol autentikasi dan akses untuk AWS CodeCommit](#)

Untuk menginstal dan mengonfigurasi AWS CLI

1. Pada komputer lokal Anda, unduh dan instal AWS CLI. Ini adalah prasyarat untuk berinteraksi dengan CodeCommit dari baris perintah. Kami merekomendasikan agar Anda menginstal AWS

CLI versi 2. Ini adalah versi utama terbaru dari AWS CLI dan mendukung semua fitur terbaru. Ini adalah satu-satunya versi AWS CLI yang mendukung penggunaan akun root, akses federasi, atau kredensial sementara dengan git-remote-codecommit.

Untuk informasi selengkapnya, lihat [Menyiapkan Antarmuka Baris Perintah AWS](#).

 Note

CodeCommit hanya berfungsi dengan AWS CLI versi 1.7.38 dan yang lebih baru. Sebagai praktik terbaik, instal atau tingkatkan AWS CLI untuk versi terbaru yang tersedia. Untuk menentukan versi AWS CLI yang telah Anda instal, jalankan perintah `aws --version`.

Untuk meng-upgrade versi lama AWS CLI ke versi terbaru, lihat [Menginstal AWS Command Line Interface](#).

2. Jalankan perintah ini untuk memverifikasi bahwa CodeCommit perintah AWS CLI untuk diinstal.

```
aws codecommit help
```

Perintah ini mengembalikan daftar CodeCommit perintah.

3. Konfigurasi AWS CLI dengan profil dengan menggunakan perintah `configure`, sebagai berikut:

```
aws configure
```

Saat diminta, tentukan kunci AWS akses dan kunci akses AWS rahasia pengguna IAM untuk digunakan. CodeCommit Juga, pastikan untuk menentukan Wilayah AWS di mana repositori ada, seperti `us-east-2`. Saat diminta untuk format output default, tentukan `json`. Misalnya, jika Anda mengonfigurasi profil untuk pengguna IAM:

```
AWS Access Key ID [None]: Type your IAM user AWS access key ID here, and then press Enter
```

```
AWS Secret Access Key [None]: Type your IAM user AWS secret access key here, and then press Enter
```

```
Default region name [None]: Type a supported region for CodeCommit here, and then press Enter
```

```
Default output format [None]: Type json here, and then press Enter
```

Untuk informasi selengkapnya tentang membuat dan mengonfigurasi profil untuk digunakan dengan AWS CLI, lihat berikut ini:

- [Profil Bernama](#)
- [Menggunakan Peran IAM dalam AWS CLI](#)
- [Tetapkan perintah](#)
- [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#)

Untuk terkoneksi ke repositori atau sumber daya di Wilayah AWS lain, Anda harus mengonfigurasi ulang AWS CLI dengan nama wilayah default. Nama Wilayah default yang didukung untuk CodeCommit menyertakan:

- us-east-2
- us-east-1
- eu-west-1
- us-west-2
- ap-northeast-1
- ap-southeast-1
- ap-southeast-2
- ap-southeast-3
- eu-central-1
- eu-central-1
- ap-northeast-2
- sa-east-1
- us-west-1
- eu-west-2
- ap-south-1
- ap-south-1
- ca-central-1
- us-gov-west-1
- us-gov-east-1
- eu-north-1

- ap-east-1
- me-south-1
- cn-north-1
- cn-northwest-1
- eu-south-1
- ap-northeast-3
- af-south-1
- ll-sentral-1

Untuk informasi lebih lanjut tentang CodeCommit dan Wilayah AWS, lihat [Wilayah dan titik akhir koneksi Git](#). Untuk informasi selengkapnya tentang IAM, access key, dan kunci rahasia, lihat [Bagaimana Saya Mendapatkan Kredensial?](#) dan [Mengelola Access Key untuk Pengguna IAM](#). Untuk informasi selengkapnya tentang AWS CLI dan profil, lihat [Profil Bernama](#).

Selanjutnya, Anda harus menginstal Git.

- Untuk Linux, macOS, atau Unix:

Untuk bekerja dengan file, commit, dan informasi lainnya di CodeCommit repositori, Anda harus menginstal Git pada mesin lokal Anda. CodeCommit mendukung Git versi 1.7.9 dan yang lebih baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git.

Untuk menginstal Git, kami merekomendasikan situs web seperti [Pengunduhan Git](#).

Note

Git adalah platform yang berkembang dan diperbarui secara teratur. Terkadang, perubahan fitur dapat memengaruhi cara kerjanya CodeCommit. Jika Anda mengalami masalah dengan versi Git tertentu dan CodeCommit, tinjau informasinya di [Memecahkan masalah](#).

- Untuk Windows:

Untuk bekerja dengan file, commit, dan informasi lainnya di CodeCommit repositori, Anda harus menginstal Git pada mesin lokal Anda. CodeCommit mendukung Git versi 1.7.9 dan yang lebih

baru. Versi Git 2.28 mendukung konfigurasi nama cabang untuk commit awal. Kami menyarankan penggunaan versi terbaru Git.

Untuk menginstal Git, kami merekomendasikan situs web seperti [Git untuk Windows](#). Jika Anda menggunakan link ini untuk menginstal Git, Anda dapat menerima semua pengaturan default instalasi kecuali untuk berikut:

- Ketika diminta selama langkah Menyesuaikan lingkungan PATH Anda, pilih opsi untuk menggunakan Git dari baris perintah.
- (Opsional) Jika Anda bermaksud menggunakan HTTPS dengan pembantu kredensi yang disertakan dalam AWS CLI alih-alih mengonfigurasi kredensi Git CodeCommit untuk, pada halaman Mengonfigurasi opsi tambahan, pastikan opsi Aktifkan Pengelola Kredensial Git dihapus. Git Credential Manager hanya kompatibel dengan CodeCommit jika pengguna IAM mengonfigurasi kredensi Git. Untuk informasi selengkapnya, silakan lihat [Untuk pengguna HTTPS menggunakan kredensial Git](#) dan [Git untuk Windows: Saya menginstal Git untuk Windows, tapi akses saya ke repositori saya ditolak \(403\)](#).

Note

Git adalah platform yang berkembang dan diperbarui secara teratur. Terkadang, perubahan fitur dapat memengaruhi cara kerjanya CodeCommit. Jika Anda mengalami masalah dengan versi Git tertentu dan CodeCommit, tinjau informasinya di [Memecahkan masalah](#).

CodeCommit mendukung otentikasi HTTPS dan SSH. Untuk menyelesaikan penyiapan, Anda harus mengonfigurasi kredensi Git untuk CodeCommit (HTTPS, direkomendasikan untuk sebagian besar pengguna), key pair SSH (SSH) untuk digunakan saat mengakses CodeCommit, git-remote-codecommit (direkomendasikan untuk pengguna yang menggunakan akses federasi), atau pembantu kredensial yang disertakan dalam AWS CLI

- Untuk kredensial Git pada semua sistem operasi yang didukung, lihat [Langkah 3: Buat kredensi Git untuk koneksi HTTPS CodeCommit](#).
- Untuk SSH di Linux, macOS, atau Unix, lihat [SSH dan Linux, macOS, atau Unix: Siapkan kunci publik dan privat untuk Git dan CodeCommit](#).
- Untuk SSH di Windows, lihat [Langkah 3: Siapkan kunci publik dan privat untuk Git dan CodeCommit](#).

- Untuk git-remote-codecommit, lihat [Langkah-langkah penyiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit](#).
- Untuk pembantu kredensial di Linux, macOS, atau Unix, lihat [Mengatur Credential Helper \(Linux, macOS, atau Unix\)](#).
- Untuk pembantu kredensial di Windows, lihat [Mengatur Credential Helper \(Windows\)](#).

Langkah 1: Buat CodeCommit repositori

Di bagian ini, Anda menggunakan CodeCommit konsol untuk membuat CodeCommit repositori yang Anda gunakan untuk sisa tutorial ini. Untuk menggunakan AWS CLI untuk membuat repositori, lihat [Buatlah sebuah repositori \(AWS CLI\)](#).

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di pemilih wilayah, pilih Wilayah AWS tempat Anda ingin membuat repositori tersebut. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Pada halaman Repositori, pilih Buat repositori.
4. Pada halaman Buat repositori, di Nama repositori, masukkan nama untuk repositori tersebut.

Note

Nama repositori sensitif huruf besar dan kecil. Nama harus unik di Wilayah AWS untuk akun Amazon Web Services Anda.

5. (Opsional) Di Deskripsi, masukkan deskripsi untuk repositori tersebut. Ini dapat membantu Anda dan pengguna lain mengidentifikasi tujuan repositori tersebut.

Note

Bidang deskripsi menampilkan Markdown di konsol dan menerima semua karakter HTML dan karakter Unicode yang valid. Jika Anda adalah pengembang aplikasi yang menggunakan BatchGetRepositories API GetRepository atau dan Anda berencana untuk menampilkan bidang deskripsi repositori di browser web, lihat Referensi [CodeCommit API](#).

6. (Opsional) Pilih Tambahkan tag untuk menambahkan satu atau lebih tag repositori (label atribut khusus yang membantu Anda mengatur dan mengelola sumber daya AWS Anda) ke repositori Anda. Untuk informasi selengkapnya, lihat [Menandai repositori di AWS CodeCommit](#).

7. (Opsional) Perluas konfigurasi tambahan untuk menentukan apakah akan menggunakan default Kunci yang dikelola AWS atau kunci yang dikelola pelanggan Anda sendiri untuk mengenkripsi dan mendekripsi data dalam repositori ini. Jika Anda memilih untuk menggunakan kunci yang dikelola pelanggan Anda sendiri, Anda harus memastikan bahwa itu tersedia di Wilayah AWS tempat Anda membuat repositori, dan bahwa kuncinya aktif. Untuk informasi selengkapnya, lihat [AWS Key Management Service dan enkripsi untuk repositori AWS CodeCommit](#).
8. (Opsional) Pilih Aktifkan Amazon CodeGuru Reviewer untuk Java dan Python jika repositori ini berisi kode Java atau Python, dan Anda ingin Reviewer menganalisisnya. CodeGuru CodeGuru Reviewer menggunakan beberapa model pembelajaran mesin untuk menemukan cacat kode dan menyarankan perbaikan dan perbaikan dalam permintaan tarik. Untuk informasi selengkapnya, lihat [Panduan Pengguna Amazon CodeGuru Reviewer](#).
9. Pilih Buat.

Setelah dibuat, repositori muncul di daftar Repositori. Di kolom URL, pilih ikon salin, lalu pilih protokol (HTTPS atau SSH) yang akan digunakan untuk CodeCommit terhubung. Salin URL.

Misalnya, jika Anda menamai repositori Anda *MyFirstRepo* dan Anda menggunakan HTTPS, URL akan terlihat seperti berikut:

```
https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyFirstRepo
```

Anda membutuhkan URL ini nanti di [Langkah 2: Migrasikan konten lokal ke repositori CodeCommit](#).

Langkah 2: Migrasikan konten lokal ke repositori CodeCommit

Sekarang setelah Anda memiliki CodeCommit repositori, Anda dapat memilih direktori di komputer lokal Anda untuk dikonversi menjadi repositori Git lokal. Perintah git init dapat digunakan baik untuk mengkonversi konten yang ada, konten tanpa versi ke repositori Git atau, jika Anda belum memiliki file atau konten, untuk memulai repositori baru yang kosong.

1. Dari terminal atau baris perintah pada komputer lokal Anda, ubah direktori ke direktori yang ingin Anda gunakan sebagai sumber untuk repositori Anda.
2. Jalankan perintah berikut untuk mengkonfigurasi Git menggunakan cabang default bernama **main**:

```
git config --local init.defaultBranch main
```

Anda juga dapat menjalankan perintah ini untuk mengatur nama cabang default Anda ke **main** untuk semua repositori yang baru dibuat:

```
git config --global init.defaultBranch main
```

3. Jalankan perintah git init untuk memulai kontrol versi Git dalam direktori. Hal ini menciptakan subdirektori .git di akar direktori yang memungkinkan pelacakan kontrol versi. Folder .git juga berisi semua metadata yang diperlukan untuk repositori.

```
git init
```

4. Periksa status direktori yang dimulai dengan menjalankan perintah berikut:

```
git status
```

Tambahkan file yang ingin Anda tambahkan ke kontrol versi. Dalam tutorial ini, Anda menjalankan perintah git add dengan specifier . untuk menambahkan semua file dalam direktori ini. Untuk pilihan lain, konsultasikan dokumentasi Git Anda.

```
git add .
```

5. Buat commit untuk file yang ditambahkan dengan pesan commit.

```
git commit -m "Initial commit"
```

6. Jalankan git push perintah, tentukan URL dan nama CodeCommit repositori tujuan dan opsi. --all (Ini adalah URL yang Anda salin di [Langkah 1: Buat CodeCommit repositori.](#))

Misalnya, jika Anda menamai repositori Anda *MyFirstRepo* dan Anda diatur untuk menggunakan HTTPS, Anda akan menjalankan perintah berikut:

```
git push https://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyFirstRepo --all
```

Langkah 3: Lihat file di CodeCommit

Setelah Anda mendorong isi direktori Anda, Anda dapat menggunakan CodeCommit konsol untuk dengan cepat melihat semua file di repositori.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Di Repositori, pilih nama repositori (misalnya, *MyFirstRepository*) dari daftar.
3. Lihat file dalam repositori untuk cabang, klon, URL, pengaturan, dan banyak lagi.

Langkah 4: Bagikan CodeCommit repositori

Saat Anda membuat repositori di CodeCommit, dua titik akhir dihasilkan: satu untuk koneksi HTTPS dan satu untuk koneksi SSH. Keduanya menyediakan koneksi yang aman melalui sebuah jaringan. Para pengguna Anda dapat menggunakan salah satu protokol. Kedua titik akhir tetap aktif terlepas dari protokol mana yang Anda rekomendasikan untuk para pengguna Anda. Sebelum Anda dapat membagikan repositori dengan orang lain, Anda harus membuat kebijakan IAM yang memungkinkan pengguna lain mengakses repositori Anda. Berikan petunjuk akses tersebut kepada pengguna Anda.

Buatlah kebijakan terkelola pelanggan untuk repositori Anda

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Pada panel navigasi Dasbor, pilih Kebijakan, dan kemudian pilih Buat Kebijakan.
3. Pada halaman Buat Kebijakan, pilih Impor kebijakan terkelola.
4. Pada halaman Impor kebijakan terkelola, di Kebijakan filter, masukkan **AWSCodeCommitPowerUser**. Pilih tombol di samping nama kebijakan, lalu pilih Impor.
5. Di halaman Buat kebijakan, pilih JSON. Ganti bagian "*" dari Resource baris untuk CodeCommit tindakan dengan Nama Sumber Daya Amazon (ARN) dari CodeCommit repositori, seperti yang ditunjukkan di sini:

```
"Resource": [  
  "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo"  
]
```

Tip

Untuk menemukan ARN untuk CodeCommit repositori, buka CodeCommit konsol, pilih nama repositori dari daftar, lalu pilih Pengaturan. Untuk informasi selengkapnya, lihat [Tampilkan detail repositori](#).

Jika Anda ingin kebijakan ini diterapkan ke lebih dari satu repositori, tambahkan setiap repositori sebagai sumber daya dengan menentukan ARN-nya. Sertakan koma antara setiap pernyataan sumber daya, seperti yang ditunjukkan di sini:


```
"Resource": [  
  "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",  
  "arn:aws:codecommit:us-east-2:111111111111:MyOtherDemoRepo"  
]
```

Setelah Anda selesai mengedit, pilih Tinjau kebijakan.

6. Pada halaman Kebijakan Tinjauan, di Nama, masukkan nama baru untuk kebijakan (misalnya, *AWSCodeCommitPowerUser- MyDemoRepo*). Secara opsional berikan deskripsi untuk kebijakan ini.
7. Pilih Buat Kebijakan.

Untuk mengelola akses ke repositori Anda, buatlah sebuah grup IAM untuk para penggunanya, tambahkan pengguna IAM ke grup tersebut, lalu lampirkan kebijakan terkelola pelanggan yang Anda buat pada langkah sebelumnya. Melampirkan kebijakan lain yang diperlukan untuk akses, seperti `IAMSelfManageServiceSpecificCredentials` atau `IAMUserSSHKeys`.

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Dalam area navigasi Dasbor, pilih Grup, dan kemudian pilih Buat Grup Baru.
3. Pada halaman Setel Nama Grup, di Nama Grup, masukkan nama untuk grup (misalnya, *MyDemoRepoGroup*), lalu pilih Langkah Berikutnya. Pertimbangkan memasukkan nama repositori sebagai bagian dari nama grup.

 Note

Nama ini harus unik di seluruh akun Amazon Web Services.

4. Pilih kotak di sebelah kebijakan terkelola pelanggan yang Anda buat di bagian sebelumnya (misalnya, *AWSCodeCommitPowerUser- MyDemoRepo*).

5. Pada halaman Tinjauan, pilih Buat Grup. IAM membuat grup ini dengan kebijakan tertentu yang sudah terlampir. Grup tersebut muncul dalam daftar grup yang terkait dengan akun Amazon Web Services Anda.
6. Pilih nama grup Anda dari daftar.
7. Di halaman ringkasan grup, pilih tab Pengguna, dan kemudian pilih Tambahkan Pengguna ke Grup. Pada daftar yang menampilkan semua pengguna yang terkait dengan akun Amazon Web Services Anda, pilih kotak di sebelah pengguna yang ingin Anda izinkan akses ke CodeCommit repositori, lalu pilih Tambah Pengguna.

 Tip

Anda dapat menggunakan kotak Pencarian untuk menemukan pengguna dengan cepat berdasarkan nama.

8. Bila Anda telah menambahkan pengguna Anda, tutup konsol IAM tersebut.

Setelah Anda membuat pengguna IAM yang akan digunakan untuk mengakses CodeCommit menggunakan grup kebijakan dan kebijakan yang Anda konfigurasi, kirimkan informasi yang diperlukan kepada pengguna tersebut untuk terhubung ke repositori.

1. Buka CodeCommit konsol di <https://console.aws.amazon.com/codesuite/codecommit/home>.
2. Dalam pemilih wilayah, pilih Wilayah AWS di mana repositori dibuat. Repositori adalah khusus untuk sebuah Wilayah AWS. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).
3. Pada halaman Repositori, pilih repositori yang Anda ingin bagikan.
4. Dalam URL Klon, pilih protokol yang Anda inginkan agar digunakan oleh para pengguna Anda. Ini menyalin URL klon untuk protokol koneksi.
5. Kirimkan URL klon kepada para pengguna Anda beserta petunjuk lainnya, seperti menginstal AWS CLI, mengkonfigurasi profil, atau menginstal Git. Pastikan untuk menyertakan informasi konfigurasi untuk protokol koneksi (misalnya, HTTPS).

Memigrasi repositori secara bertahap

Ketika bermigrasi ke AWS CodeCommit, pertimbangkan untuk mengirim repositori Anda secara bertahap atau dalam beberapa potongan untuk mengurangi kemungkinan masalah jaringan

intermiten atau kinerja jaringan yang memburuk yang mengakibatkan seluruh pengiriman gagal. Dengan menggunakan pengiriman tambahan dengan skrip seperti yang disertakan di sini, Anda dapat me-restart migrasi dan mengirim hanya commit yang tidak berhasil pada upaya sebelumnya.

Prosedur dalam topik ini menunjukkan kepada Anda cara membuat dan menjalankan skrip yang memigrasi repositori Anda secara bertahap dan hanya mengirim kembali tahapan yang tidak berhasil sampai migrasi selesai.

Instruksi ini ditulis dengan asumsi bahwa Anda telah menyelesaikan langkah-langkah di [Pengaturan](#) dan [Buatlah sebuah repositori](#).

Topik

- [Langkah 0: Menentukan apakah akan bermigrasi secara bertahap](#)
- [Langkah 1: Instal prasyarat dan tambahkan repositori sebagai CodeCommit remote](#)
- [Langkah 2: Buat skrip untuk digunakan untuk migrasi secara bertahap](#)
- [Langkah 3: Jalankan skrip dan migrasi secara bertahap ke CodeCommit](#)
- [Lampiran: Skrip contoh incremental-repo-migration.py](#)

Langkah 0: Menentukan apakah akan bermigrasi secara bertahap

Ada beberapa faktor yang perlu dipertimbangkan untuk menentukan ukuran keseluruhan repositori Anda dan apakah akan dimigrasi secara bertahap. Yang paling jelas adalah ukuran keseluruhan artefak dalam repositori. Faktor-faktor seperti akumulasi riwayat repositori juga dapat berkontribusi terhadap ukuran. Sebuah repositori dengan riwayat tahunan dan banyak cabang bisa memiliki ukuran yang sangat besar, meskipun aset individunya tidak. Ada sejumlah strategi Anda dapat jalankan untuk membuat migrasi repositori ini lebih sederhana dan lebih efisien. Misalnya, Anda dapat menggunakan strategi klon dangkal saat mengkloning repositori dengan riwayat perkembangan yang panjang, atau Anda dapat mematikan kompresi delta untuk file biner besar. Anda dapat meneliti pilihan dengan mengkonsultasikan dokumentasi Git Anda, atau Anda dapat memilih untuk mengatur dan mengkonfigurasi pengiriman tambahan untuk migrasi repositori Anda menggunakan contoh skrip yang disertakan dalam topik ini, `incremental-repo-migration.py`.

Anda mungkin ingin mengkonfigurasi pengiriman tambahan jika satu atau lebih kondisi berikut ini benar:

- Repositori yang ingin Anda migrasikan memiliki riwayat lebih dari lima tahun.

- Koneksi internet Anda tergantung pada gangguan intermiten, paket yang terputus, respons lambat, atau gangguan lain dalam layanan.
- Ukuran keseluruhan repositori lebih besar dari 2 GB dan Anda berniat untuk memigrasikan seluruh repositori.
- Repositori berisi artefak besar atau binari yang tidak memampatkan dengan baik, seperti file gambar besar dengan lebih dari lima versi yang dilacak.
- Anda sebelumnya telah mencoba migrasi ke CodeCommit dan menerima pesan “Kesalahan Layanan Internal”.

Bahkan jika tidak ada kondisi di atas yang benar, Anda masih dapat memilih untuk mengirim secara bertahap.

Langkah 1: Instal prasyarat dan tambahkan repositori sebagai CodeCommit remote

Anda dapat membuat skrip khusus Anda sendiri, yang memiliki prasyarat tersendiri. Jika Anda menggunakan sampel yang disertakan dalam topik ini, Anda harus:

- Memasang prasyaratnya.
- Mengkloning repositori ke komputer lokal Anda.
- Tambahkan CodeCommit repositori sebagai remote untuk repositori yang ingin Anda migrasikan.

Siapkan untuk menjalankan `incremental-repo-migration .py`

1. Pada komputer lokal Anda, instal Python 2.6 atau versi yang lebih baru. Untuk informasi selengkapnya dan versi terbaru, lihat [situs web Python](#).
2. Pada komputer yang sama, instal GitPython, yang merupakan pustaka Python yang digunakan untuk berinteraksi dengan repositori Git. Untuk informasi selengkapnya, lihat [dokumentasi GitPython](#).
3. Gunakan perintah `git clone --mirror` untuk mengkloning repositori yang ingin Anda migrasikan ke komputer lokal Anda. Dari terminal (Linux, macOS, atau Unix) atau command prompt (Windows), gunakan perintah `git clone --mirror` untuk membuat repo lokal untuk repositori, termasuk direktori di mana Anda ingin membuat repo lokal. *Misalnya, untuk mengkloning repositori Git bernama MyMigrationRepo dengan URL `https://example.com/my-repo/` ke direktori bernama `my-repo`:*

```
git clone --mirror https://example.com/my-repo/MyMigrationRepo.git my-repo
```

Anda akan melihat output yang mirip dengan berikut ini, yang menunjukkan repositori telah dikloning ke repo lokal telanjang bernama my-repo:

```
Cloning into bare repository 'my-repo'...
remote: Counting objects: 20, done.
remote: Compressing objects: 100% (17/17), done.
remote: Total 20 (delta 5), reused 15 (delta 3)
Unpacking objects: 100% (20/20), done.
Checking connectivity... done.
```

- Ubah direktori ke repo lokal untuk repositori yang baru saja Anda kloning (misalnya, *my-repo*). Dari direktori itu, gunakan git remote add **DefaultRemoteName RemoteRepositoryURL** perintah untuk menambahkan repositori sebagai CodeCommit repositori jarak jauh untuk repo lokal.

Note

Saat mengirim repositori besar, pertimbangkan untuk menggunakan SSH dan bukan HTTPS. Ketika Anda mengirim perubahan besar, sejumlah besar perubahan, atau repositori besar, koneksi HTTPS yang berjalan lama sering dihentikan sebelum waktunya karena masalah jaringan atau pengaturan firewall. Untuk informasi selengkapnya tentang pengaturan CodeCommit SSH, lihat [Untuk koneksi SSH di Linux, macOS, atau Unix](#) atau [Untuk koneksi SSH pada Windows](#).

Misalnya, gunakan perintah berikut untuk menambahkan titik akhir SSH untuk repositori bernama MyDestinationRepo sebagai CodeCommit repositori jarak jauh untuk remote bernama: codecommit

```
git remote add codecommit ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDestinationRepo
```

i Tip

Karena ini adalah klon, nama remote default (`origin`) sudah digunakan. Anda harus menggunakan nama remote lain. Meskipun contoh menggunakan `codecommit`, Anda dapat menggunakan nama yang Anda inginkan. Gunakan perintah `git remote show` untuk meninjau daftar remote yang ditetapkan untuk repo lokal Anda.

5. Gunakan perintah `git remote -v` untuk menampilkan pengaturan pengambilan dan pengiriman untuk repo lokal Anda dan pastikan pengaturannya sudah benar. Misalnya:

```
codecommit ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDestinationRepo
(fetch)
codecommit ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDestinationRepo
(push)
```

i Tip

Jika Anda masih melihat entri pengambilan dan pengiriman untuk repositori remote yang berbeda (misalnya, entri untuk asal), gunakan perintah `git remote set-url --delete` untuk menghapusnya.

Langkah 2: Buat skrip untuk digunakan untuk migrasi secara bertahap

Langkah-langkah ini ditulis dengan asumsi bahwa Anda menggunakan contoh skrip `incremental-repo-migration.py`.

1. Buka editor teks dan tempel konten [contoh skrip](#) ke dalam dokumen kosong.
2. Simpan dokumen dalam direktori dokumen (bukan direktori kerja dari repo lokal Anda) dan namai `incremental-repo-migration.py`. Pastikan direktori yang Anda pilih adalah salah satu yang dikonfigurasi di lingkungan lokal atau variabel jalur, sehingga Anda dapat menjalankan skrip Python dari baris perintah atau terminal.

Langkah 3: Jalankan skrip dan migrasi secara bertahap ke CodeCommit

Sekarang setelah Anda membuat `incremental-repo-migration.py` skrip, Anda dapat menggunakannya untuk memigrasikan repo lokal secara bertahap ke repositori. CodeCommit Secara default, skrip mengirim commit dalam batch 1.000 commit dan mencoba untuk menggunakan pengaturan Git untuk direktori dari yang dijalankan sebagai pengaturan untuk repo lokal dan repositori remote. Anda dapat menggunakan opsi yang disertakan dalam `incremental-repo-migration.py` untuk mengkonfigurasi pengaturan lainnya, jika perlu.

1. Dari terminal atau command prompt, ubah direktori ke repo lokal Anda yang ingin Anda migrasikan.
2. Dari direktori tersebut, jalankan perintah berikut:

```
python incremental-repo-migration.py
```

3. Skrip berjalan dan menunjukkan kemajuan pada terminal atau command prompt. Beberapa repositori besar lambat menunjukkan kemajuan. Skrip berhenti sebuah pengiriman tunggal gagal tiga kali. Selanjutnya Anda dapat menjalankan kembali skrip, dan mulai dari batch yang gagal. Anda dapat menjalankan kembali skrip sampai semua pengiriman berhasil dan migrasi selesai.

Tip

Anda dapat menjalankan `incremental-repo-migration.py` dari direktori apapun selama Anda menggunakan pilihan `-l` dan `-r` untuk menentukan pengaturan lokal dan remote yang akan digunakan. Misalnya, untuk menggunakan skrip dari direktori apapun untuk memigrasi repo lokal yang terletak di `/tmp/my-repo` ke remote yang dijuluki *codecommit*:

```
python incremental-repo-migration.py -l "/tmp/my-repo" -r "codecommit"
```

Anda juga mungkin ingin menggunakan pilihan `-b` untuk mengubah ukuran batch default yang digunakan ketika melakukan pengiriman secara bertahap. Sebagai contoh, jika Anda secara teratur mengirim repositori dengan file biner yang sangat besar yang sering berubah dan bekerja dari lokasi yang memiliki bandwidth jaringan terbatas, Anda mungkin ingin menggunakan pilihan `-b` untuk mengubah ukuran batch ke 500 bukan 1.000. Misalnya:

```
python incremental-repo-migration.py -b 500
```

Hal ini mengirim repo lokal secara bertahap dalam batch 500 commit. Jika Anda memutuskan untuk mengubah ukuran batch lagi ketika Anda memigrasi repositori (misalnya, jika Anda memutuskan untuk mengurangi ukuran batch setelah gagal melakukan upaya pengiriman), ingat untuk menggunakan pilihan `-c` untuk menghapus tanda batch sebelum mengatur ulang ukuran batch dengan `-b`:

```
python incremental-repo-migration.py -c
python incremental-repo-migration.py -b 250
```

Important

Jangan gunakan pilihan `-c` jika Anda ingin menjalankan ulang skrip setelah gagal. Pilihan `-c` menghapus tanda yang digunakan untuk batch commit. Gunakan pilihan `-c` hanya jika Anda ingin mengubah ukuran batch dan mulai lagi, atau jika Anda memutuskan tidak lagi ingin menggunakan skrip.

Lampiran: Skrip contoh **incremental-repo-migration.py**

Untuk kenyamanan Anda, kami telah mengembangkan contoh skrip Python, `incremental-repo-migration.py`, untuk mengirim repositori secara bertahap. Skrip ini adalah contoh kode sumber terbuka dan disediakan apa adanya.

```
# Copyright 2015 Amazon.com, Inc. or its affiliates. All Rights Reserved. Licensed
  under the Amazon Software License (the "License").
# You may not use this file except in compliance with the License. A copy of the
  License is located at
#   http://aws.amazon.com/asl/
# This file is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
  KIND, express or implied. See the License for
# the specific language governing permissions and limitations under the License.

#!/usr/bin/env python

import os
import sys
from optparse import OptionParser
from git import Repo, TagReference, RemoteProgress, GitCommandError
```



```
class PushProgressPrinter(RemoteProgress):
    def update(self, op_code, cur_count, max_count=None, message=""):
        op_id = op_code & self.OP_MASK
        stage_id = op_code & self.STAGE_MASK
        if op_id == self.WRITING and stage_id == self.BEGIN:
            print("\tObjects: %d" % max_count)

class RepositoryMigration:
    MAX_COMMITS_TOLERANCE_PERCENT = 0.05
    PUSH_RETRY_LIMIT = 3
    MIGRATION_TAG_PREFIX = "codecommit_migration_"

    def migrate_repository_in_parts(
        self, repo_dir, remote_name, commit_batch_size, clean
    ):
        self.next_tag_number = 0
        self.migration_tags = []
        self.walked_commits = set()
        self.local_repo = Repo(repo_dir)
        self.remote_name = remote_name
        self.max_commits_per_push = commit_batch_size
        self.max_commits_tolerance = (
            self.max_commits_per_push * self.MAX_COMMITS_TOLERANCE_PERCENT
        )

        try:
            self.remote_repo = self.local_repo.remote(remote_name)
            self.get_remote_migration_tags()
        except (ValueError, GitCommandError):
            print(
                "Could not contact the remote repository. The most common reasons for
                this error are that the name of the remote repository is incorrect, or that you do not
                have permissions to interact with that remote repository."
            )
            sys.exit(1)

        if clean:
            self.clean_up(clean_up_remote=True)
            return

        self.clean_up()
```

```
print("Analyzing repository")
head_commit = self.local_repo.head.commit
sys.setrecursionlimit(max(sys.getrecursionlimit(), head_commit.count()))

# tag commits on default branch
leftover_commits = self.migrate_commit(head_commit)
self.tag_commits([commit for (commit, commit_count) in leftover_commits])

# tag commits on each branch
for branch in self.local_repo.heads:
    leftover_commits = self.migrate_commit(branch.commit)
    self.tag_commits([commit for (commit, commit_count) in leftover_commits])

# push the tags
self.push_migration_tags()

# push all branch references
for branch in self.local_repo.heads:
    print("Pushing branch %s" % branch.name)
    self.do_push_with_retries(ref=branch.name)

# push all tags
print("Pushing tags")
self.do_push_with_retries(push_tags=True)

self.get_remote_migration_tags()
self.clean_up(clean_up_remote=True)

print("Migration to CodeCommit was successful")

def migrate_commit(self, commit):
    if commit in self.walked_commits:
        return []

    pending_ancestor_pushes = []
    commit_count = 1

    if len(commit.parents) > 1:
        # This is a merge commit
        # Ensure that all parents are pushed first
        for parent_commit in commit.parents:
            pending_ancestor_pushes.extend(self.migrate_commit(parent_commit))
    elif len(commit.parents) == 1:
```

```
        # Split linear history into individual pushes
        next_ancestor, commits_to_next_ancestor = self.find_next_ancestor_for_push(
            commit.parents[0]
        )
        commit_count += commits_to_next_ancestor
        pending_ancestor_pushes.extend(self.migrate_commit(next_ancestor))

    self.walked_commits.add(commit)

    return self.stage_push(commit, commit_count, pending_ancestor_pushes)

def find_next_ancestor_for_push(self, commit):
    commit_count = 0

    # Traverse linear history until we reach our commit limit, a merge commit, or
    # an initial commit
    while (
        len(commit.parents) == 1
        and commit_count < self.max_commits_per_push
        and commit not in self.walked_commits
    ):
        commit_count += 1
        self.walked_commits.add(commit)
        commit = commit.parents[0]

    return commit, commit_count

def stage_push(self, commit, commit_count, pending_ancestor_pushes):
    # Determine whether we can roll up pending ancestor pushes into this push
    combined_commit_count = commit_count + sum(
        ancestor_commit_count
        for (ancestor, ancestor_commit_count) in pending_ancestor_pushes
    )

    if combined_commit_count < self.max_commits_per_push:
        # don't push anything, roll up all pending ancestor pushes into this
        # pending push
        return [(commit, combined_commit_count)]

    if combined_commit_count <= (
        self.max_commits_per_push + self.max_commits_tolerance
    ):
        # roll up everything into this commit and push
        self.tag_commits([commit])
```

```
        return []

    if commit_count >= self.max_commits_per_push:
        # need to push each pending ancestor and this commit
        self.tag_commits(
            [
                ancestor
                for (ancestor, ancestor_commit_count) in pending_ancestor_pushes
            ]
        )
        self.tag_commits([commit])
        return []

    # push each pending ancestor, but roll up this commit
    self.tag_commits(
        [ancestor for (ancestor, ancestor_commit_count) in pending_ancestor_pushes]
    )
    return [(commit, commit_count)]

def tag_commits(self, commits):
    for commit in commits:
        self.next_tag_number += 1
        tag_name = self.MIGRATION_TAG_PREFIX + str(self.next_tag_number)

        if tag_name not in self.remote_migration_tags:
            tag = self.local_repo.create_tag(tag_name, ref=commit)
            self.migration_tags.append(tag)
        elif self.remote_migration_tags[tag_name] != str(commit):
            print(
                "Migration tags on the remote do not match the local tags. Most
likely your batch size has changed since the last time you ran this script. Please run
this script with the --clean option, and try again."
            )
            sys.exit(1)

def push_migration_tags(self):
    print("Will attempt to push %d tags" % len(self.migration_tags))
    self.migration_tags.sort(
        key=lambda tag: int(tag.name.replace(self.MIGRATION_TAG_PREFIX, ""))
    )
    for tag in self.migration_tags:
        print(
            "Pushing tag %s (out of %d tags), commit %s"
            % (tag.name, self.next_tag_number, str(tag.commit))
        )
```

```
)
    self.do_push_with_retries(ref=tag.name)

def do_push_with_retries(self, ref=None, push_tags=False):
    for i in range(0, self.PUSH_RETRY_LIMIT):
        if i == 0:
            progress_printer = PushProgressPrinter()
        else:
            progress_printer = None

        try:
            if push_tags:
                infos = self.remote_repo.push(tags=True, progress=progress_printer)
            elif ref is not None:
                infos = self.remote_repo.push(
                    refspec=ref, progress=progress_printer
                )
            else:
                infos = self.remote_repo.push(progress=progress_printer)

            success = True
            if len(infos) == 0:
                success = False
            else:
                for info in infos:
                    if (
                        info.flags & info.UP_TO_DATE
                        or info.flags & info.NEW_TAG
                        or info.flags & info.NEW_HEAD
                    ):
                        continue
                    success = False
                    print(info.summary)

            if success:
                return
        except GitCommandError as err:
            print(err)

    if push_tags:
        print("Pushing all tags failed after %d attempts" %
              (self.PUSH_RETRY_LIMIT))
    elif ref is not None:
        print("Pushing %s failed after %d attempts" % (ref, self.PUSH_RETRY_LIMIT))
```

```
        print(
            "For more information about the cause of this error, run the following
command from the local repo: 'git push %s %s'"
            % (self.remote_name, ref)
        )
    else:
        print(
            "Pushing all branches failed after %d attempts"
            % (self.PUSH_RETRY_LIMIT)
        )
    sys.exit(1)

def get_remote_migration_tags(self):
    remote_tags_output = self.local_repo.git.ls_remote(
        self.remote_name, tags=True
    ).split("\n")
    self.remote_migration_tags = dict(
        (tag.split()[1].replace("refs/tags/", ""), tag.split()[0])
        for tag in remote_tags_output
        if self.MIGRATION_TAG_PREFIX in tag
    )

def clean_up(self, clean_up_remote=False):
    tags = [
        tag
        for tag in self.local_repo.tags
        if tag.name.startswith(self.MIGRATION_TAG_PREFIX)
    ]

    # delete the local tags
    TagReference.delete(self.local_repo, *tags)

    # delete the remote tags
    if clean_up_remote:
        tags_to_delete = [":" + tag_name for tag_name in
self.remote_migration_tags]
        self.remote_repo.push(refspec=tags_to_delete)

parser = OptionParser()
parser.add_option(
    "-l",
    "--local",
    action="store",
```

```
    dest="localrepo",
    default=os.getcwd(),
    help="The path to the local repo. If this option is not specified, the script will
attempt to use current directory by default. If it is not a local git repo, the script
will fail.",
)
parser.add_option(
    "-r",
    "--remote",
    action="store",
    dest="remoterepo",
    default="codecommit",
    help="The name of the remote repository to be used as the push or migration
destination. The remote must already be set in the local repo ('git remote add ...').
If this option is not specified, the script will use 'codecommit' by default.",
)
parser.add_option(
    "-b",
    "--batch",
    action="store",
    dest="batchsize",
    default="1000",
    help="Specifies the commit batch size for pushes. If not explicitly set, the
default is 1,000 commits.",
)
parser.add_option(
    "-c",
    "--clean",
    action="store_true",
    dest="clean",
    default=False,
    help="Remove the temporary tags created by migration from both the local repo
and the remote repository. This option will not do any migration work, just cleanup.
Cleanup is done automatically at the end of a successful migration, but not after a
failure so that when you re-run the script, the tags from the prior run can be used to
identify commit batches that were not pushed successfully.",
)

(options, args) = parser.parse_args()

migration = RepositoryMigration()
migration.migrate_repository_in_parts(
    options.localrepo, options.remoterepo, int(options.batchsize), options.clean
)
```


Keamanan di AWS CodeCommit

Keamanan cloud di AWS merupakan prioritas tertinggi. Sebagai pelanggan AWS, Anda mendapatkan manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara AWS dan Anda. [Model tanggung jawab bersama](#) menggambarkan ini sebagai keamanan dari cloud dan keamanan di dalam cloud:

- Keamanan cloud – AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan layanan AWS di Cloud AWS. AWS juga menyediakan layanan yang dapat Anda gunakan dengan aman. Auditor pihak ketiga menguji dan memverifikasi efektivitas keamanan kami secara berkala sebagai bagian dari [Program Kepatuhan AWS](#). Untuk mempelajari tentang program kepatuhan yang berlaku untuk AWS CodeCommit, lihat [AWS Layanan dalam Lingkup oleh Program Kepatuhan](#).
- Keamanan di cloud – Tanggung jawab Anda ditentukan menurut layanan AWS yang Anda gunakan. Anda juga bertanggung jawab atas faktor lain termasuk sensitivitas data Anda, persyaratan perusahaan Anda, serta hukum dan peraturan yang berlaku.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan CodeCommit. Topik berikut menunjukkan kepada Anda cara mengkonfigurasi CodeCommit untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan layanan AWS lain yang membantu Anda memantau dan mengamankan sumber daya CodeCommit Anda.

Topik

- [Perlindungan data di AWS CodeCommit](#)
- [Identity and Access Management untuk AWS CodeCommit](#)
- [Ketahanan di AWS CodeCommit](#)
- [Keamanan infrastruktur dalam AWS CodeCommit](#)

Perlindungan data di AWS CodeCommit

Sebagai layanan terkelola, dilindungi oleh keamanan jaringan AWS global. Untuk informasi tentang layanan keamanan AWS dan cara AWS melindungi infrastruktur, lihat [Keamanan Cloud AWS](#). Guna

mendesain lingkungan AWS Anda menggunakan praktik terbaik untuk keamanan infrastruktur, lihat [Perlindungan Infrastruktur](#) dalam Kerangka Kerja AWS Well-Architected Pilar Keamanan.

Anda menggunakan panggilan API yang AWS dipublikasikan untuk mengakses melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Pengangkutan (TLS) Kami mensyaratkan TLS 1.2 dan merekomendasikan TLS 1.3.
- Suite cipher dengan perfect forward secrecy (PFS) seperti DHE (Ephemeral Diffie-Hellman) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Sebagian besar sistem modern seperti Java 7 dan sistem yang lebih baru mendukung mode ini.

Selain itu, permintaan harus ditandatangani menggunakan ID kunci akses dan kunci akses rahasia yang dikaitkan dengan pengguna utama IAM. Atau Anda bisa menggunakan [AWS Security Token Service](#) (AWS STS) untuk membuat kredensial keamanan sementara guna menandatangani permintaan.

CodeCommit repositori secara otomatis dienkripsi saat istirahat. Tidak ada tindakan pelanggan yang diperlukan. CodeCommit juga mengenkripsi data repositori dalam perjalanan. Anda dapat menggunakan protokol HTTPS, protokol SSH, atau keduanya dengan CodeCommit repositori. Untuk informasi selengkapnya, lihat [Pengaturan untuk AWS CodeCommit](#). Anda juga dapat mengonfigurasi [akses lintas akun ke](#) CodeCommit repositori.

Topik

- [AWS Key Management Service dan enkripsi untuk repositori AWS CodeCommit](#)
- [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#)

AWS Key Management Service dan enkripsi untuk repositori AWS CodeCommit

Data dalam CodeCommit repositori dienkripsi dalam perjalanan dan saat istirahat. Ketika data didorong ke dalam CodeCommit repositori (misalnya, dengan memanggil push), CodeCommit mengenkripsi data yang diterima saat disimpan dalam repositori. Ketika data ditarik dari CodeCommit repositori (misalnya, dengan memanggil pull), CodeCommit mendekripsi data dan kemudian mengirimkannya ke pemanggil. Ini mengasumsikan pengguna IAM yang terkait dengan permintaan push atau pull telah dikonfirmasi oleh AWS. Data yang dikirim atau diterima ditransmisikan menggunakan protokol jaringan terenkripsi HTTPS atau SSH.

Anda dapat menggunakan salah satu Kunci yang dikelola AWS atau kunci yang dikelola pelanggan untuk mengenkripsi dan mendekripsi data di repositori Anda. Untuk informasi selengkapnya tentang perbedaan antara kunci yang dikelola pelanggan dan Kunci yang dikelola AWS, lihat [Kunci yang dikelola pelanggan dan Kunci yang dikelola AWS](#). Jika Anda tidak menentukan kunci yang dikelola pelanggan, CodeCommit akan menggunakan Kunci yang dikelola AWS untuk mengenkripsi dan mendekripsi data di repositori Anda. Kunci yang dikelola AWS ini dibuat secara otomatis untuk Anda di Akun AWS. Pertama kali Anda membuat CodeCommit repositori di akun Amazon Web Services baru Wilayah AWS, jika Anda tidak menentukan kunci yang dikelola pelanggan, akan CodeCommit membuat Kunci yang dikelola AWS (`aws/codecommit` kunci) yang sama Wilayah AWS di AWS Key Management Service (AWS KMS). `aws/codecommit` Kunci ini hanya digunakan oleh CodeCommit. Ini disimpan dalam akun Amazon Web Services Anda. Bergantung pada apa yang Anda tentukan, CodeCommit baik menggunakan kunci yang dikelola pelanggan atau Kunci yang dikelola AWS untuk mengenkripsi dan mendekripsi data dalam repositori.

Important

CodeCommit melakukan AWS KMS tindakan berikut terhadap AWS KMS kunci yang digunakan untuk mengenkripsi dan mendekripsi data dalam repositori. Jika Anda menggunakan Kunci yang dikelola AWS, pengguna tidak memerlukan izin eksplisit untuk tindakan ini, tetapi pengguna tidak boleh memiliki kebijakan terlampir yang menolak tindakan ini untuk kunci tersebut `aws/codecommit`. Jika Anda menggunakan kunci terkelola pelanggan yang memiliki Akun AWS ID yang ditetapkan sebagai prinsip kebijakan untuk kunci tersebut, izin ini harus disetel secara eksplisit. `allow` Khususnya, ketika Anda membuat repositori pertama Anda, dan jika Anda memperbarui kunci untuk repositori Anda, Anda tidak boleh memiliki izin berikut yang disetel ke `deny` jika Anda menggunakan Kunci yang dikelola AWS, dan harus disetel ke `allow` jika Anda menggunakan kunci yang dikelola pelanggan dengan prinsip kebijakan:

- `"kms:Encrypt"`
- `"kms:Decrypt"`
- `"kms:ReEncrypt"` (tergantung pada konteksnya, ini bisa membutuhkan `kms:ReEncryptFrom`,

`kms:ReEncryptTo`, atau `kms:ReEncrypt*` tidak diatur untuk menyangkal)
- `"kms:GenerateDataKey"`
- `"kms:GenerateDataKeyWithoutPlaintext"`

- "kms:DescribeKey"

Jika Anda ingin menggunakan kunci yang dikelola pelanggan Anda sendiri, kunci harus tersedia di Wilayah AWS tempat repositori ada. CodeCommit mendukung penggunaan kunci terkelola pelanggan tunggal dan Multi-wilayah. Meskipun semua jenis asal material utama didukung, sebaiknya gunakan opsi KMS default. Pelanggan yang menggunakan opsi toko kunci Eksternal mungkin mengalami penundaan dari penyedia toko mereka. Selain itu, CodeCommit memiliki persyaratan berikut untuk kunci yang dikelola pelanggan:

- CodeCommit hanya mendukung menggunakan tombol simetris.
- Jenis penggunaan kunci harus diatur ke Enkripsi dan dekripsi.

Untuk informasi selengkapnya tentang membuat kunci terkelola pelanggan, lihat [Konsep](#) dan [Membuat kunci](#).

Untuk melihat informasi tentang yang Kunci yang dikelola AWS dihasilkan oleh CodeCommit, lakukan hal berikut:

1. Masuk ke AWS Management Console dan buka konsol AWS Key Management Service (AWS KMS) di <https://console.aws.amazon.com/kms>.
2. Untuk mengubah Wilayah AWS, gunakan pemilih Wilayah di sudut kanan atas halaman.
3. Di panel navigasi, pilih Kunci yang dikelola AWS. Pastikan Anda masuk ke Wilayah AWS di mana Anda ingin meninjau kunci.
4. Dalam daftar kunci enkripsi, pilih Kunci yang dikelola AWS dengan alias aws/codecommit. Informasi dasar tentang Kunci milik AWS akan ditampilkan.

Anda tidak dapat mengubah atau menghapus Kunci yang dikelola AWS ini.

Bagaimana algoritma enkripsi digunakan untuk mengenkripsi data repositori

CodeCommit menggunakan dua pendekatan berbeda untuk mengenkripsi data. Objek Git individu di bawah 6 MB dienkrpsi menggunakan AES-GCM-256, yang menyediakan validasi integritas data. Objek antara 6 MB dan maksimum 2 GB untuk satu gumpalan dienkrpsi menggunakan AES-CBC-256. CodeCommit selalu memvalidasi konteks enkripsi.

Konteks enkripsi

Setiap layanan terintegrasi dengan AWS KMS menentukan konteks enkripsi untuk operasi enkripsi dan dekripsi. Konteks enkripsi tersebut adalah informasi diautentikasi tambahan yang digunakan AWS KMS untuk memeriksa integritas data. Bila ditentukan untuk operasi enkripsi, maka harus juga ditentukan dalam operasi dekripsi. Jika tidak, dekripsi gagal. CodeCommit menggunakan ID CodeCommit repositori untuk konteks enkripsi. Anda dapat menggunakan `get-repository` perintah atau CodeCommit konsol untuk menemukan ID repositori. Cari ID CodeCommit repositori di AWS CloudTrail log untuk memahami operasi enkripsi mana yang diambil untuk mengenkripsi atau AWS KMS mendekripsi data dalam repositori. CodeCommit

Untuk informasi lebih lanjut tentang AWS KMS, lihat [AWS Key Management Service Panduan Developer](#).

Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar

Anda dapat memberi pengguna akses ke repositori AWS CodeCommit Anda tanpa mengonfigurasi pengguna IAM untuknya atau menggunakan access key dan secret key. Untuk menetapkan izin ke identitas gabungan, Anda membuat peran dan menentukan izin untuk peran tersebut. Ketika identitas gabungan mengautentikasi, identitas tersebut dikaitkan dengan peran dan diberikan izin yang didefinisikan oleh peran. Untuk informasi tentang peran untuk federasi, lihat [Membuat peran untuk Penyedia Identitas pihak ketiga](#) dalam Panduan Pengguna IAM. Jika Anda menggunakan IAM Identity Center, Anda mengkonfigurasi set izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah mengautentikasi, IAM Identity Center menghubungkan izin yang ditetapkan ke peran dalam IAM. Untuk informasi tentang set izin, lihat [Kumpulan izin](#) di Panduan AWS IAM Identity Center Pengguna. Anda juga dapat mengkonfigurasi akses berbasis peran untuk pengguna IAM untuk mengakses CodeCommit repositori di akun Amazon Web Services yang terpisah (teknik yang dikenal sebagai akses lintas akun). Untuk panduan konfigurasi akses lintas akun ke repositori, lihat [Konfigurasi akses lintas akun ke AWS CodeCommit repositori menggunakan peran](#).

Anda dapat mengkonfigurasi akses untuk pengguna yang ingin atau harus mengotentikasi melalui metode seperti:

- Security Assertion Markup Language (SAML)
- Autentikasi multi-faktor (MFA)
- Federation

- Login with Amazon
- Amazon Cognito
- Facebook
- Google
- Penyedia identitas yang kompatibel OpenID Connect (OIDC)

Note

Informasi berikut hanya berlaku untuk pengguna `git-remote-codecommit` atau pembantu `AWS CLI` kredensial untuk terhubung ke `CodeCommit` repositori. Karena pendekatan yang direkomendasikan untuk akses sementara atau federasi untuk `CodeCommit` diatur `git-remote-codecommit`, topik ini memberikan contoh penggunaan utilitas tersebut. Untuk informasi selengkapnya, lihat [Langkah-langkah penyiapan untuk koneksi HTTPS ke `AWS CodeCommit` bersama `git-remote-codecommit`](#).
Anda tidak dapat menggunakan kredensial SSH atau Git dan HTTPS untuk terhubung ke `CodeCommit` repositori dengan kredensial akses berputar atau sementara.

Anda tidak perlu menyelesaikan langkah-langkah ini jika semua persyaratan berikut ini benar:

- Anda masuk ke instans Amazon EC2.
- Anda menggunakan Git dan HTTPS dengan pembantu `AWS CLI` kredensial untuk menghubungkan dari instans Amazon EC2 ke `CodeCommit` repositori.
- Instans Amazon EC2 memiliki profil instans IAM terlampir yang berisi izin akses yang dijelaskan di [Untuk koneksi HTTPS di Linux, macOS, atau Unix dengan pembantu kredensial `AWS CLI`](#) atau [Untuk sambungan HTTPS pada Windows dengan pembantu kredensial `AWS CLI`](#).
- Anda telah menginstal dan mengkonfigurasi Git pembantu kredensial pada instans Amazon EC2, seperti yang dijelaskan dalam [Untuk koneksi HTTPS di Linux, macOS, atau Unix dengan pembantu kredensial `AWS CLI`](#) atau [Untuk sambungan HTTPS pada Windows dengan pembantu kredensial `AWS CLI`](#).

Instans Amazon EC2 yang memenuhi persyaratan sebelumnya sudah diatur untuk mengkomunikasikan kredensial akses sementara untuk `CodeCommit` atas nama Anda.

Note

Anda dapat mengonfigurasi dan menggunakan git-remote-codecommit pada instans Amazon EC2.

Untuk memberi pengguna akses sementara keCodeCommit repositori Anda, selesaikan langkah berikut.

Langkah 1: Selesaikan prasyarat

Selesaikan langkah-langkah pengaturan untuk memberi pengguna akses keCodeCommit repositori Anda menggunakan kredensial berputar:

- Untuk akses lintas akun, lihat [Panduan: Mendelegasikan Akses Akun Amazon Web Services menggunakan IAM role](#) dan [Konfigurasi akses lintas akun ke AWS CodeCommit repositori menggunakan peran](#).
- Untuk SAML dan federation, lihat [Menggunakan Sistem Autentikasi Organisasi Anda Untuk Memberikan Akses ke Sumber Daya AWS](#) and [Tentang Federation Berbasis AWS STS SAML 2.0](#).
- Untuk MFA, lihat [Menggunakan Perangkat Multi-Factor Authentication \(MFA\) dengan AWS dan Membuat Kredensial Keamanan Sementara Untuk Mengaktifkan Akses Untuk Pengguna IAM](#).
- Untuk Login with Amazon, Amazon Cognito, Facebook, Google, atau penyedia identitas mana pun yang kompatibel dengan OIDC, lihat [Tentang Web Identity Federation AWS STS](#).

Gunakan informasi [Kontrol autentikasi dan akses untuk AWS CodeCommit](#) untuk menentukanCodeCommit izin yang ingin Anda berikan kepada pengguna.

Langkah 2: Dapatkan nama peran atau kredensial akses

Jika Anda ingin pengguna Anda mengakses repositori dengan memiliki suatu peran, berikan pengguna Anda Amazon Resource Name (ARN) dari peran itu. Jika tidak, tergantung pada cara Anda mengatur akses, pengguna Anda bisa mendapatkan kredensial berputar dengan salah satu cara berikut:

- Untuk akses lintas akun, panggil perintahAWS CLI [assume-role](#) atau panggilAWS STS [AssumeRoleAPI](#).

- Untuk SALL, panggil AWS CLI [assume-role-with-saml](#) perintah atau AWS STS [AssumeRoleWithSALL](#) API.
- Untuk federasi, panggil AWS CLI [assume-role](#) atau [get-federation-token](#) perintah atau AWS STS [AssumeRole](#) atau [GetFederationToken](#) API.
- Untuk MFA, panggil AWS CLI [get-session-token](#) perintah atau AWS STS [GetSessionToken](#) API.
- Untuk Login with Amazon, Amazon Cognito, Facebook, Google, atau penyedia identitas mana pun yang kompatibel dengan OIDC, panggil perintah AWS CLI [assume-role-with-web-identity](#) atau AWS STS [AssumeRoleWithWebIdentity](#) API.

Langkah 3: Instal `git-remote-codecommit` dan konfigurasi AWS CLI

Anda harus mengkonfigurasi komputer lokal Anda untuk menggunakan kredensial akses dengan menginstal [git-remote-codecommit](#) dan mengkonfigurasi profil di AWS CLI.

1. Ikuti instruksi di [Pengaturan](#) untuk mengatur AWS CLI. Gunakan perintah `aws configure` untuk mengkonfigurasi satu atau lebih profil. Pertimbangkan untuk membuat profil bernama untuk digunakan saat Anda terhubung ke CodeCommit repositori menggunakan kredensial berputar.
2. Anda dapat mengasosiasikan kredensialnya dengan profil bernama AWS CLI dari pengguna dengan salah satu cara berikut.
 - Jika Anda memiliki peran untuk mengakses CodeCommit, konfigurasi profil bernama dengan informasi yang diperlukan untuk mengasumsikan peran itu. Misalnya, jika Anda ingin memiliki peran bernama `CodeCommitAccess` di akun Amazon Web Services 111111111111, Anda dapat mengkonfigurasi profil default untuk digunakan saat bekerja dengan AWS sumber daya lainnya dan profil bernama untuk digunakan ketika melaksanakan peran itu. Perintah berikut membuat profil bernama `CodeAccess` yang memiliki peran bernama `CodeCommitAccess`. Nama pengguna `Maria_Garcia` dikaitkan dengan sesi dan profil default diatur sebagai sumber AWS kredensialnya:

```
aws configure set role_arn arn:aws:iam::111111111111:role/CodeCommitAccess --  
profile CodeAccess  
aws configure set source_profile default --profile CodeAccess  
aws configure set role_session_name "Maria_Garcia" --profile CodeAccess
```

Jika Anda ingin memverifikasi perubahan, secara manual melihat atau mengedit file `~/.aws/config` (untuk Linux) atau file `%UserProfile%.aws\config` (untuk Windows) dan

meninjau informasi di bawah profil bernama. Misalnya, file Anda mungkin terlihat seperti berikut:

```
[default]
region = us-east-1
output = json

[profile CodeAccess]
source_profile = default
role_session_name = Maria_Garcia
role_arn = arn:aws:iam::111111111111:role/CodeCommitAccess
```

Setelah Anda mengkonfigurasi profil bernama Anda, Anda kemudian dapat mengkloning CodeCommit repositori dengan `git clone` menggunakan profil bernama. Misalnya, untuk mengkloning repositori bernama *MyDemoRepo*:

```
git clone codecommit://CodeAccess@MyDemoRepo
```

- Jika Anda menggunakan federasi identitas web dan OpenID Connect (OIDC), konfigurasi profil bernama yang membuat panggilan AWS Security Token Service (AWS STS) `AssumeRoleWithWebIdentity` API atas nama Anda untuk merefresh kredensial sementara. Gunakan perintah `aws configure set` atau edit secara manual file `~/.aws/credentials` (untuk Linux) atau file `%UserProfile%.aws\credentials` (untuk Windows) untuk menambahkan profil yang dinamai AWS CLI dengan nilai pengaturan yang diperlukan. Misalnya, untuk membuat profil yang mengasumsikan *CodeCommitAccess* peran dan menggunakan file token identitas web `~/my-credentials/my-token-file`:

```
[CodeCommitWebIdentity]
role_arn = arn:aws:iam::111111111111:role/CodeCommitAccess
web_identity_token_file=~/my-credentials/my-token-file
role_session_name = Maria_Garcia
```

Untuk informasi selengkapnya, lihat [Mengonfigurasi AWS Command Line Interface](#) dan [Menggunakan IAM role di AWS CLI](#) di Panduan Pengguna AWS Command Line Interface.

Langkah 4: Mengakses CodeCommit repositori

Dengan asumsi pengguna Anda telah mengikuti petunjuk [Connect ke sebuah repositori](#) untuk terhubung ke CodeCommit repositori, pengguna kemudian menggunakan fungsionalitas diperpanjang yang disediakan oleh `git-remote-codecommit` dan `git pull` untuk memanggil `git clone`, `git push`, dan untuk mengkloning, mendorong ke, dan menarik dari CodeCommit repositori yang dia akses. Misalnya, untuk mengkloning suatu repositori:

```
git clone codecommit://CodeAccess@MyDemoRepo
```

Perintah `git commit`, `git push`, dan `git pull` menggunakan sintaks Git biasa.

Ketika pengguna menggunakan AWS CLI dan menentukan profil yang dinamai AWS CLI yang terkait dengan kredensial akses berputar, hasil yang dinilai ke profil tersebut dikembalikan.

Identity and Access Management untuk AWS CodeCommit

(IAM) AWS Identity and Access Management adalah Layanan AWS yang membantu seorang administrator dalam mengendalikan akses ke sumber daya AWS secara aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya. CodeCommit IAM adalah sebuah layanan Layanan AWS yang dapat Anda gunakan tanpa dikenakan biaya tambahan.

Topik

- [Audiens](#)
- [Mengautentikasi dengan identitas](#)
- [Mengelola kebijakan menggunakan akses](#)
- [Kontrol autentikasi dan akses untuk AWS CodeCommit](#)
- [Cara kerja AWS CodeCommit dengan IAM](#)
- [CodeCommit Kebijakan berbasis sumber daya](#)
- [Otorisasi berdasarkan tag CodeCommit](#)
- [CodeCommit Peran IAM](#)
- [Contoh kebijakan berbasis identitas AWS CodeCommit](#)
- [Pemecahan masalah identitas dan akses AWS CodeCommit](#)

Audiens

Cara menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan di CodeCommit.

Pengguna layanan — Jika Anda menggunakan CodeCommit layanan untuk melakukan pekerjaan Anda, administrator Anda memberi Anda kredensi dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak CodeCommit fitur untuk melakukan pekerjaan Anda, Anda mungkin memerlukan izin tambahan. Memahami cara akses dikelola dapat membantu Anda meminta izin yang tepat dari administrator Anda. Jika Anda tidak dapat mengakses fitur di CodeCommit, lihat [Pemecahan masalah identitas dan akses AWS CodeCommit](#).

Administrator layanan — Jika Anda bertanggung jawab atas CodeCommit sumber daya di perusahaan Anda, Anda mungkin memiliki akses penuh ke CodeCommit. Tugas Anda adalah menentukan CodeCommit fitur dan sumber daya mana yang harus diakses pengguna layanan Anda. Kemudian, Anda harus mengirimkan permintaan kepada administrator IAM Anda untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep dasar IAM. Untuk mempelajari lebih lanjut tentang bagaimana perusahaan Anda dapat menggunakan IAM CodeCommit, lihat [Cara kerja AWS CodeCommit dengan IAM](#).

Administrator IAM – Jika Anda adalah administrator IAM, Anda mungkin ingin belajar dengan lebih detail tentang cara Anda menulis kebijakan untuk mengelola akses ke CodeCommit. Untuk melihat contoh kebijakan CodeCommit berbasis identitas yang dapat Anda gunakan di IAM, lihat. [Contoh kebijakan berbasis identitas AWS CodeCommit](#)

Mengautentikasi dengan identitas

Autentikasi merupakan cara Anda untuk masuk ke AWS dengan menggunakan kredensial identitas Anda. Anda harus terautentikasi (masuk keAWS) sebagai Pengguna root akun AWS, sebagai pengguna IAM, atau dengan mengambil peran IAM.

Anda dapat masuk ke AWS sebagai identitas terfederasi dengan menggunakan kredensial yang disediakan melalui sumber identitas. AWS IAM Identity Center Untuk pengguna (Pusat Identitas IAM), autentikasi sign-on tunggal perusahaan Anda, dan kredensial Google atau Facebook Anda merupakan contoh identitas terfederasi. Saat Anda masuk sebagai identitas terfederasi, administrator Anda sebelumnya menyiapkan federasi identitas dengan menggunakan peran IAM. Ketika Anda mengakses AWS dengan menggunakan federasi, Anda secara tidak langsung mengambil suatu peran.

Tergantung pada jenis pengguna Anda, Anda dapat masuk ke AWS Management Console atau portal akses AWS. Untuk informasi selengkapnya tentang masuk ke AWS, silakan lihat [Cara masuk ke Akun AWS Anda](#) di Panduan Pengguna AWS Sign-In.

Jika Anda mengakses AWS secara terprogram, AWS sediakan kit pengembangan perangkat lunak (SDK) dan antarmuka baris perintah (CLI) untuk menandatangani permintaan Anda secara kriptografis dengan menggunakan kredensial Anda. Jika Anda tidak menggunakan peralatan AWS, maka Anda harus menandatangani sendiri permintaan tersebut. Untuk informasi selengkapnya tentang penggunaan metode yang disarankan untuk menandatangani permintaan sendiri, silakan lihat [Menandatangani permintaan API AWS](#) di Panduan Pengguna IAM.

Terlepas dari metode autentikasi yang Anda gunakan, Anda mungkin juga diminta untuk menyediakan informasi keamanan tambahan. Sebagai contoh, AWS menyarankan supaya Anda menggunakan autentikasi multi-faktor (MFA) untuk meningkatkan keamanan akun Anda. Untuk mempelajari selengkapnya, silakan lihat [Autentikasi multi-faktor](#) di Panduan Pengguna AWS IAM Identity Center dan [Menggunakan autentikasi multi-faktor \(MFA\) di AWS](#) di Panduan Pengguna IAM.

Pengguna root Akun AWS

Ketika Anda membuat Akun AWS, Anda memulai dengan satu identitas masuk yang memiliki akses ke semua Layanan AWS dan sumber daya di akun tersebut. Identitas ini disebut pengguna root Akun AWS dan diakses dengan cara masuk ke alamat email dan kata sandi yang Anda gunakan untuk membuat akun. Kami sangat menyarankan Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar lengkap tugas yang mengharuskan Anda masuk sebagai pengguna root, silakan lihat [Tugas yang memerlukan kredensial pengguna root](#) di Panduan Pengguna IAM.

Pengguna dan Grup IAM

[Pengguna IAM](#) adalah identitas dalam Akun AWS Anda yang memiliki izin khusus untuk satu orang atau aplikasi. Apabila memungkinkan, kami menyarankan untuk mengandalkan pada kredensial temporer alih-alih membuat pengguna IAM yang memiliki kredensial jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan khusus yang memerlukan kredensial jangka panjang dengan pengguna IAM, kami menyarankan Anda memutar kunci akses. Untuk informasi selengkapnya, silakan lihat [Memutar kunci akses secara teratur untuk kasus penggunaan yang memerlukan kredensial jangka panjang](#) di Panduan Pengguna IAM.

[Grup IAM](#) adalah identitas yang menerangkan secara spesifik kumpulan pengguna IAM. Anda tidak dapat masuk sebagai kelompok. Anda dapat menggunakan grup untuk menerangkan secara spesifik izin untuk beberapa pengguna sekaligus. Grup membuat izin lebih mudah dikelola untuk sekelompok besar pengguna. Sebagai contoh, Anda dapat memiliki grup yang diberi nama AdminIAM dan memberikan izin kepada grup tersebut untuk mengelola sumber daya IAM.

Pengguna berbeda dari peran. Pengguna secara unik terkait dengan satu orang atau aplikasi, tetapi peran tersebut dimaksudkan untuk dapat digunakan oleh siapa pun yang membutuhkannya. Pengguna memiliki kredensial jangka panjang permanen, tetapi peran memberikan kredensial temporer. Untuk mempelajari selengkapnya, silakan lihat [Kapan harus membuat pengguna IAM \(alih-alih peran\)](#) di Panduan Pengguna IAM.

Peran IAM

[Peran IAM](#) merupakan identitas dalam Akun AWS Anda yang memiliki izin khusus. Peran ini mirip dengan pengguna IAM, tetapi tidak terkait dengan orang tertentu. Anda dapat menggunakan peran IAM untuk sementara dalam AWS Management Console dengan [berganti peran](#). Anda dapat mengambil peran dengan cara memanggil operasi API AWS CLI atau AWS atau menggunakan URL kustom. Untuk informasi selengkapnya tentang cara menggunakan peran, silakan lihat [menggunakan peran IAM](#) di Panduan Pengguna IAM.

IAM role dengan kredensial temporer berguna dalam situasi berikut:

- Akses pengguna gabungan – Untuk menetapkan izin ke sebuah identitas terfederasi, Anda harus membuat sebuah peran dan menentukan izin untuk peran tersebut. Ketika identitas gabungan terfederasi mengautentikasi, identitas tersebut terhubung dengan peran dan diberikan izin yang ditentukan oleh peran. Untuk informasi tentang peran-peran untuk federasi, silakan lihat [Membuat sebuah peran untuk Penyedia Identitas pihak ketiga](#) di Panduan Pengguna IAM. Jika Anda menggunakan Pusat Identitas IAM, Anda mengonfigurasi serangkaian izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah identitas tersebut diautentikasi, Pusat Identitas IAM mengkorelasikan izin yang diatur ke peran dalam IAM. Untuk informasi tentang rangkaian izin, silakan lihat [Rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center.
- Izin pengguna IAM sementara – Pengguna atau peran IAM dapat mengambil peran IAM untuk sementara mengambil izin berbeda untuk tugas tertentu.
- Akses lintas akun – Anda dapat menggunakan peran IAM untuk mengizinkan seseorang (pengguna utama tepercaya) di akun berbeda untuk mengakses sumber daya yang ada di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, pada beberapa Layanan AWS, Anda dapat melampirkan kebijakan secara langsung ke sumber daya (alih-alih

menggunakan suatu peran sebagai proksi). Untuk mempelajari perbedaan antara kebijakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, silakan lihat [Bagaimana peran IAM role berbeda dari kebijakan berbasis sumber daya](#) dalam Panduan Pengguna IAM.

- Akses lintas layanan – Sebagian Layanan AWS menggunakan fitur di Layanan AWS lainnya. Sebagai contoh, ketika Anda melakukan panggilan dalam suatu layanan, lazim pada layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Suatu layanan mungkin melakukan hal tersebut menggunakan izin pengguna utama panggilan, menggunakan peran layanan, atau peran tertaut layanan.
- Sesi akses maju (FAS) – Ketika Anda menggunakan pengguna IAM atau peran IAM untuk melakukan tindakan-tindakan di AWS, Anda akan dianggap sebagai seorang pengguna utama. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian dilanjutkan oleh tindakan lain pada layanan yang berbeda. FAS menggunakan izin dari pengguna utama untuk memanggil Layanan AWS, yang dikombinasikan dengan Layanan AWS yang diminta untuk membuat pengajuan ke layanan hilir. Permintaan FAS hanya diajukan ketika sebuah layanan menerima pengajuan yang memerlukan interaksi dengan Layanan AWS lain atau sumber daya lain untuk diselesaikan. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan ketika mengajukan permintaan FAS, silakan lihat [Meneruskan sesi akses](#).
- Peran layanan – Sebuah peran layanan adalah sebuah [peran IAM](#) yang dijalankan oleh suatu layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, memodifikasi, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, silakan lihat [Membuat sebuah peran untuk mendelegasikan izin ke Layanan AWS](#) dalam Panduan pengguna IAM.
- Peran tertaut layanan – Peran tertaut layanan adalah tipe peran layanan yang tertaut dengan Layanan AWS. Layanan tersebut dapat menjalankan peran untuk melakukan sebuah tindakan atas nama Anda. Peran tertaut layanan akan muncul di Akun AWS Anda dan dimiliki oleh layanan tersebut. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran tertaut layanan.
- Aplikasi yang berjalan di Amazon EC2 – Anda dapat menggunakan peran IAM untuk mengelola kredensial temporer untuk aplikasi yang berjalan di instans EC2 dan mengajukan permintaan AWS CLI atau API AWS. Cara ini lebih baik daripada menyimpan kunci akses dalam instans EC2. Untuk memberikan peran AWS ke instans EC2 dan menjadikannya terdapat di semua aplikasinya, Anda dapat membuat profil instans yang dilampirkan ke instans tersebut. Profil instans berisi peran dan memungkinkan program yang berjalan di instans EC2 untuk mendapatkan kredensial temporer.

Untuk informasi selengkapnya, silakan lihat [Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan di instans Amazon EC2](#) di Panduan Pengguna IAM.

Untuk mempelajari apakah kita harus menggunakan peran IAM atau pengguna IAM, silakan lihat [Kapan harus membuat peran IAM \(alih-alih pengguna\)](#) di Panduan Pengguna IAM.

Mengelola kebijakan menggunakan akses

Anda mengendalikan akses di AWS dengan membuat kebijakan dan melampirkannya ke identitas atau sumber daya AWS. Kebijakan adalah objek di AWS yang, ketika terkait dengan identitas atau sumber daya, akan menentukan izinnya. AWS mengevaluasi kebijakan-kebijakan tersebut ketika seorang pengguna utama (pengguna, root user, atau sesi peran) mengajukan permintaan. Izin dalam kebijakan menentukan apakah permintaan diberikan atau ditolak. Sebagian besar kebijakan disimpan di AWS sebagai dokumen JSON. Untuk informasi selengkapnya tentang struktur dan isi dokumen kebijakan JSON, silakan lihat [Gambaran Umum kebijakan JSON](#) di Panduan Pengguna IAM.

Administrator dapat menggunakan kebijakan JSON AWS untuk menentukan secara spesifik siapa yang memiliki akses pada apa. Yaitu, pengguna utama manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan syarat apa.

Secara default, para pengguna dan peran tidak memiliki izin. Untuk mengabulkan izin kepada pengguna untuk melakukan tindakan di sumber daya yang mereka perlukan, administrator IAM dapat membuat kebijakan IAM. Administrator kemudian akan dapat menambahkan kebijakan IAM ke peran, dan para pengguna dapat mengambil peran.

Kebijakan IAM mendefinisikan izin untuk suatu tindakan terlepas dari metode yang Anda gunakan untuk pengoperasiannya. Sebagai contoh, anggap saja Anda memiliki kebijakan yang mengizinkan tindakan `iam:GetRole`. Pengguna dengan kebijakan tersebut dapat memperoleh informasi peran dari AWS Management Console, AWS CLI, atau APIAWS.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan izin JSON yang dapat Anda lampirkan ke sebuah identitas, misalnya pengguna IAM, grup pengguna IAM, atau peran IAM. Kebijakan ini mengontrol apa yang pengguna tindakan dan peran dapat kerjakan, pada sumber daya mana, dan dalam keadaan apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, silakan lihat [Membuat kebijakan IAM](#) dalam Panduan Pengguna IAM.

Kebijakan berbasis identitas dapat dikategorikan lebih lanjut sebagai kebijakan inline atau kebijakan terkelola. Kebijakan inline ditanam secara langsung ke pengguna tunggal, grup, atau peran. Kebijakan terkelola adalah kebijakan yang berdiri sendiri yang dapat Anda lampirkan ke beberapa pengguna, grup, dan peran di Akun AWS Anda. Kebijakan terkelola mencakup kebijakan terkelola AWS dan kebijakan terkelola pelanggan. Untuk mempelajari cara memilih antara kebijakan terkelola atau kebijakan selaras, silakan lihat [Memilih antara kebijakan terkelola dan kebijakan selaras](#) di Panduan Pengguna IAM.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen kebijakan JSON yang Anda lampirkan ke sumber daya. Contoh kebijakan-kebijakan berbasis sumber daya adalah kebijakan terpercaya peran IAM dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya yang dilampiri kebijakan, kebijakan tersebut menentukan tindakan apa yang dapat dilakukan oleh pengguna utama yang ditentukan di sumber daya tersebut dan dalam kondisi apa. Anda harus [menentukan pengguna utama](#) dalam kebijakan berbasis sumber daya. Pengguna utama dapat mencakup akun, pengguna, peran, pengguna gabungan, atau Layanan AWS.

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan terkelola AWS dari IAM dalam kebijakan berbasis sumber daya.

Daftar kontrol akses (ACL)

Daftar kontrol akses (ACL) mengendalikan pengguna utama mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACL serupa dengan kebijakan berbasis sumber daya, meskipun kebijakan-kebijakan tersebut tidak menggunakan format dokumen kebijakan JSON.

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh-contoh layanan yang mendukung ACL. Untuk mempelajari ACL selengkapnya, silakan lihat [Gambaran umum daftar kontrol akses \(ACL\)](#) di Panduan Pengembang Layanan Penyimpanan Ringkas Amazon.

Tipe-tipe kebijakan lain

AWS mendukung tipe kebijakan tambahan, yang kurang umum. Tipe-tipe kebijakan ini dapat mengatur izin maksimum yang diberikan kepada Anda oleh tipe kebijakan yang lebih umum.

- Batasan izin – Batasan izin adalah fitur lanjutan tempat Anda mengatur izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas IAM (pengguna IAM atau peran IAM).

Anda dapat menetapkan batasan izin untuk suatu entitas. Izin yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas milik entitas dan batas izinnya. Kebijakan berbasis sumber daya yang menentukan pengguna atau peran dalam bidang `Principal` tidak dibatasi oleh batasan izin. Penolakan eksplisit dalam salah satu kebijakan ini menindahi izin. Untuk informasi selengkapnya tentang batasan izin, silakan lihat [Batasan izin untuk entitas IAM](#) di Panduan Pengguna IAM.

- Kebijakan kontrol layanan (SCP) – SCP adalah kebijakan JSON yang menentukan izin maksimum untuk sebuah organisasi atau unit organisasi (OU) di AWS Organizations. AWS Organizations adalah layanan untuk mengelompokkan dan secara terpusat mengelola beberapa Akun AWS yang dimiliki bisnis Anda. Jika Anda mengaktifkan semua fitur di suatu organisasi, maka Anda dapat menerapkan kebijakan kontrol layanan (SCP) ke salah satu atau ke semua akun Anda. SCP membatasi izin untuk entitas dalam akun anggota, termasuk setiap Pengguna root akun AWS. Untuk informasi selengkapnya tentang Organisasi dan SCP, silakan lihat [Cara kerja SCP](#) di Panduan Pengguna AWS Organizations.
- Kebijakan sesi – Kebijakan sesi adalah kebijakan lanjutan yang Anda berikan sebagai parameter ketika Anda membuat sesi sementara secara terprogram untuk peran atau pengguna gabungan. Izin sesi yang dihasilkan adalah perpotongan kebijakan berbasis identitas pengguna atau peran dan kebijakan sesi. Izin juga dapat berasal dari kebijakan berbasis sumber daya. Penolakan eksplisit dalam salah satu kebijakan ini menindahi izin. Untuk informasi selengkapnya, silakan lihat [Kebijakan sesi](#) di Panduan Pengguna IAM.

Berbagai tipe kebijakan

Ketika beberapa tipe kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan untuk mengizinkan permintaan ketika beberapa tipe kebijakan dilibatkan, lihat [Logika evaluasi kebijakan](#) dalam Panduan Pengguna IAM.

Kontrol autentikasi dan akses untuk AWS CodeCommit

Akses ke AWS CodeCommit memerlukan kredensial. Kredensi tersebut harus memiliki izin untuk mengakses AWS sumber daya, seperti CodeCommit repositori, dan pengguna IAM Anda, yang Anda gunakan untuk mengelola kredensi Git Anda atau kunci publik SSH yang Anda gunakan untuk membuat koneksi Git. Bagian berikut memberikan rincian tentang bagaimana Anda dapat menggunakan [AWS Identity and Access Management\(IAM\)](#) dan CodeCommit untuk membantu mengamankan akses ke sumber daya Anda:

- [Autentikasi](#)
- [Kontrol akses](#)

Autentikasi

Karena CodeCommit repositori berbasis Git dan mendukung fungsionalitas dasar Git, termasuk kredensi Git, kami menyarankan Anda menggunakan pengguna IAM saat bekerja dengannya. CodeCommit Anda dapat mengakses CodeCommit dengan tipe identitas lain, tetapi jenis identitas lainnya tunduk pada batasan, seperti yang dijelaskan di bawah ini.

Jenis identitas:

- Pengguna IAM — [Pengguna IAM](#) adalah identitas dalam akun Amazon Web Services Anda yang memiliki izin khusus tertentu. Misalnya, pengguna IAM dapat memiliki izin untuk membuat dan mengelola kredensi Git untuk mengakses repositori. CodeCommit Ini adalah tipe pengguna yang disarankan untuk bekerja dengan CodeCommit. Anda dapat menggunakan nama pengguna dan kata sandi IAM untuk masuk untuk mengamankan halaman web AWS seperti [AWS Management Console](#), [Forum Diskusi AWS](#), atau [Pusat AWS Support](#).

Anda dapat menghasilkan kredensial Git atau mengaitkan kunci publik SSH dengan pengguna IAM Anda, atau Anda dapat menginstal dan mengkonfigurasi git-remote-codecommit. Ini adalah cara termudah untuk mengatur Git agar berfungsi dengan CodeCommit repositori Anda. Dengan [Kredensial Git](#), Anda menghasilkan nama pengguna statis dan kata sandi di IAM. Anda kemudian menggunakan kredensial ini untuk koneksi HTTPS dengan Git dan alat pihak ketiga yang mendukung otentikasi nama pengguna dan kata sandi Git. Dengan koneksi SSH, Anda membuat file kunci publik dan pribadi di mesin lokal Anda yang Git dan CodeCommit gunakan untuk otentikasi SSH. Anda mengaitkan kunci publik dengan pengguna IAM Anda, dan Anda menyimpan kunci privat di mesin lokal Anda. [git-remote-codecommit](#) meluaskan Git itu sendiri, dan tidak memerlukan menyiapkan kredensial Git untuk pengguna.

Selain itu, Anda juga dapat membuat [access key](#) untuk setiap pengguna. Gunakan access keys ketika Anda mengakses layanan AWS secara terprogram, baik melalui [salah satu SDK AWS](#) atau dengan menggunakan [AWS Command Line Interface \(AWS CLI\)](#). Alat SDK dan CLI menggunakan access key untuk menandatangani permintaan Anda secara kriptografis. Jika Anda tidak menggunakan alat AWS, Anda harus menandatangani permintaan tersebut sendiri. CodeCommit mendukung Signature Version 4, protokol untuk mengautentikasi permintaan API masuk. Untuk informasi selengkapnya tentang melakukan autentikasi permintaan, lihat [Proses Penandatanganan Tanda Tangan Versi 4](#) dalam Referensi Umum AWS.

- Pengguna root akun Amazon Web Services – Saat mendaftar ke AWS, Anda memberikan alamat dan kata sandi yang dikaitkan dengan akun Amazon Web Services Anda. Ini adalah kredensial root Anda, dan memberikan akses penuh ke semua sumber daya AWS Anda. Beberapa CodeCommit

fitur tidak tersedia untuk pengguna akun root. Selain itu, satu-satunya cara untuk menggunakan Git dengan akun root Anda adalah menginstal dan mengkonfigurasi git-remote-codecommit (disarankan) atau untuk mengkonfigurasi pembantu kredensial AWS, yang disertakan dengan AWS CLI. Anda tidak dapat menggunakan kredensial Git atau SSH pasangan kunci publik-privat dengan pengguna akun root Anda. Untuk alasan ini, kami tidak menyarankan menggunakan pengguna akun root Anda saat berinteraksi dengan CodeCommit.

Important

Demi alasan keamanan, kami menyarankan Anda menggunakan kredensial root hanya untuk membuat pengguna administrator, yang merupakan pengguna IAM dengan izin penuh ke akun AWS Anda. Kemudian, Anda dapat menggunakan administrator ini untuk membuat pengguna IAM lainnya dan peran dengan izin terbatas. Untuk informasi selengkapnya, lihat [Praktik Terbaik IAM](#) dan [Membuat Pengguna Admin dan Grup](#) dalam Panduan Pengguna IAM.

- IAM Identity Center dan pengguna di IAM Identity Center — AWS IAM Identity Center memperluas kemampuan AWS Identity and Access Management untuk menyediakan tempat sentral yang menyatukan administrasi pengguna dan akses mereka ke Akun AWS dan aplikasi cloud. Meskipun direkomendasikan sebagai praktik terbaik bagi sebagian besar pengguna yang bekerja dengan AWS, IAM Identity Center saat ini tidak menyediakan mekanisme untuk kredensi Git atau pasangan kunci SSH. Pengguna ini dapat menginstal dan mengonfigurasi git-remote-codecommit untuk mengkloning CodeCommit repositori secara lokal, tetapi tidak semua lingkungan pengembangan terintegrasi (IDE) mendukung kloning, mendorong, atau menarik. git-remote-codecommit

Praktik terbaiknya berupa, mewajibkan pengguna manusia, termasuk pengguna yang memerlukan akses administrator, untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS dengan menggunakan kredensial temporer.

Identitas terfederasi adalah pengguna dari direktori pengguna perusahaan Anda, penyedia identitas web, dikenal sebagai AWS Directory Service, direktori Pusat Identitas, atau pengguna mana pun yang mengakses Layanan AWS dengan menggunakan kredensial yang disediakan melalui sumber identitas. Ketika identitas terfederasi mengakses Akun AWS, identitas tersebut mengambil peran, dan peran memberikan kredensial temporer.

Untuk pengelolaan akses terpusat, kami sarankan Anda menggunakan AWS IAM Identity Center. Anda dapat membuat pengguna dan grup di Pusat Identitas IAM, atau Anda dapat

menghubungkan dan menyinkronkan ke sekumpulan pengguna dan grup di sumber identitas Anda sendiri untuk digunakan di semua Akun AWS Anda dan aplikasi Anda. Untuk informasi tentang Pusat Identitas IAM, silakan lihat [Apakah Pusat Identitas IAM itu?](#) di User Guide AWS IAM Identity Center.

- IAM role — Seperti pengguna IAM, sebuah [IAM role](#) adalah identitas IAM yang dapat Anda buat di akun Anda untuk memberikan izin spesifik.

[Peran IAM](#) merupakan identitas dalam Akun AWS Anda yang memiliki izin khusus. Peran ini mirip dengan pengguna IAM, tetapi tidak terkait dengan orang tertentu. Anda dapat menggunakan peran IAM untuk sementara dalam AWS Management Console dengan [berganti peran](#). Anda dapat mengambil peran dengan cara memanggil operasi API AWS CLI atau AWS atau menggunakan URL kustom. Untuk informasi selengkapnya tentang cara menggunakan peran, silakan lihat [menggunakan peran IAM](#) di Panduan Pengguna IAM.

IAM role dengan kredensial temporer berguna dalam situasi berikut:

- Akses pengguna gabungan – Untuk menetapkan izin ke sebuah identitas terfederasi, Anda harus membuat sebuah peran dan menentukan izin untuk peran tersebut. Ketika identitas gabungan terfederasi mengautentikasi, identitas tersebut terhubung dengan peran dan diberikan izin yang ditentukan oleh peran. Untuk informasi tentang peran-peran untuk federasi, silakan lihat [Membuat sebuah peran untuk Penyedia Identitas pihak ketiga](#) di Panduan Pengguna IAM. Jika Anda menggunakan Pusat Identitas IAM, Anda mengonfigurasi serangkaian izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah identitas tersebut diautentikasi, Pusat Identitas IAM mengkorelasikan izin yang diatur ke peran dalam IAM. Untuk informasi tentang rangkaian izin, silakan lihat [Rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center.
- Izin pengguna IAM sementara – Pengguna atau peran IAM dapat mengambil peran IAM untuk sementara mengambil izin berbeda untuk tugas tertentu.
- Akses lintas akun – Anda dapat menggunakan peran IAM untuk mengizinkan seseorang (pengguna utama tepercaya) di akun berbeda untuk mengakses sumber daya yang ada di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, pada beberapa Layanan AWS, Anda dapat melampirkan kebijakan secara langsung ke sumber daya (alih-alih menggunakan suatu peran sebagai proksi). Untuk mempelajari perbedaan antara kebijakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, silakan lihat [Bagaimana peran IAM role berbeda dari kebijakan berbasis sumber daya](#) dalam Panduan Pengguna IAM.
- Akses lintas layanan – Sebagian Layanan AWS menggunakan fitur di Layanan AWS lainnya. Sebagai contoh, ketika Anda melakukan panggilan dalam suatu layanan, lazim pada layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Suatu

layanan mungkin melakukan hal tersebut menggunakan izin pengguna utama panggilan, menggunakan peran layanan, atau peran tertaut layanan.

- Sesi akses maju (FAS) – Ketika Anda menggunakan pengguna IAM atau peran IAM untuk melakukan tindakan-tindakan di AWS, Anda akan dianggap sebagai seorang pengguna utama. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian dilanjutkan oleh tindakan lain pada layanan yang berbeda. FAS menggunakan izin dari pengguna utama untuk memanggil Layanan AWS, yang dikombinasikan dengan Layanan AWS yang diminta untuk membuat pengajuan ke layanan hilir. Permintaan FAS hanya diajukan ketika sebuah layanan menerima pengajuan yang memerlukan interaksi dengan Layanan AWS lain atau sumber daya lain untuk diselesaikan. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan ketika mengajukan permintaan FAS, silakan lihat [Meneruskan sesi akses](#).
- Peran layanan – Sebuah peran layanan adalah sebuah [peran IAM](#) yang dijalankan oleh suatu layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, memodifikasi, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, silakan lihat [Membuat sebuah peran untuk mendelegasikan izin ke Layanan AWS](#) dalam Panduan pengguna IAM.
- Peran tertaut layanan – Peran tertaut layanan adalah tipe peran layanan yang tertaut dengan Layanan AWS. Layanan tersebut dapat menjalankan peran untuk melakukan sebuah tindakan atas nama Anda. Peran tertaut layanan akan muncul di Akun AWS Anda dan dimiliki oleh layanan tersebut. Administrator IAM dapat melihat, tetapi tidak dapat mengedit izin untuk peran tertaut layanan.
- Aplikasi yang berjalan di Amazon EC2 – Anda dapat menggunakan peran IAM untuk mengelola kredensial temporer untuk aplikasi yang berjalan di instans EC2 dan mengajukan permintaan AWS CLI atau API AWS. Cara ini lebih baik daripada menyimpan kunci akses dalam instans EC2. Untuk memberikan peran AWS ke instans EC2 dan menjadikannya terdapat di semua aplikasinya, Anda dapat membuat profil instans yang dilampirkan ke instans tersebut. Profil instans berisi peran dan memungkinkan program yang berjalan di instans EC2 untuk mendapatkan kredensial temporer. Untuk informasi selengkapnya, silakan lihat [Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan di instans Amazon EC2](#) di Panduan Pengguna IAM.

Untuk mempelajari apakah kita harus menggunakan peran IAM atau pengguna IAM, silakan lihat [Kapan harus membuat peran IAM \(alih-alih pengguna\)](#) di Panduan Pengguna IAM.

Note

Anda tidak dapat menggunakan kredensial Git atau SSH pasangan kunci publik-privat dengan pengguna gabungan. Selain itu, preferensi pengguna tidak tersedia untuk pengguna gabungan. Untuk informasi tentang cara menyiapkan koneksi menggunakan akses gabungan, lihat [Langkah-langkah persiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit](#).

Kontrol akses

Anda dapat memiliki kredensi yang valid untuk mengautentikasi permintaan Anda, tetapi kecuali Anda memiliki izin, Anda tidak dapat membuat atau mengakses sumber daya. CodeCommit Misalnya, Anda harus memiliki izin untuk melihat repositori, kode dorongan, membuat dan mengelola kredensial Git, dan sebagainya.

Bagian berikut menjelaskan cara mengelola izin untuk CodeCommit. Kami menyarankan agar Anda membaca gambaran umum terlebih dahulu.

- [Ikhtisar mengelola izin akses ke sumber daya Anda CodeCommit](#)
- [Menggunakan kebijakan berbasis identitas \(Kebijakan IAM\) untuk CodeCommit](#)
- [Referensi izin CodeCommit](#)

Ikhtisar mengelola izin akses ke sumber daya Anda CodeCommit

Setiap sumber daya AWS dimiliki oleh akun Amazon Web Services. Izin untuk membuat, dan mengakses sumber daya diatur oleh kebijakan izin. Administrator akun dapat melampirkan kebijakan izin pada identitas IAM (yaitu pengguna, grup, dan peran). Layanan lain, seperti AWS Lambda, juga support melampirkan kebijakan izin untuk sumber daya.

Note

Seorang administrator akun (atau pengguna administrator) adalah pengguna dengan hak istimewa administrator. Untuk informasi lebih lanjut, lihat [Praktik Terbaik IAM](#) di Panduan Pengguna IAM.

Ketika memberikan izin, Anda memutuskan orang yang mendapatkan izin, sumber daya yang mereka dapatkan izinnnya, dan tindakan khusus yang ingin Anda izinkan di sumber daya tersebut.

Topik

- [CodeCommit sumber daya dan operasi](#)
- [Memahami kepemilikan sumber daya](#)
- [Mengelola akses ke sumber daya](#)
- [Pelingkupan sumber daya di CodeCommit](#)
- [Menentukan elemen kebijakan: sumber daya, tindakan, efek, dan prinsipal](#)
- [Menetapkan ketentuan dalam kebijakan](#)

CodeCommit sumber daya dan operasi

Di CodeCommit, sumber daya utama adalah repositori. Setiap sumber daya ini memiliki Amazon Resource Name (ARN) yang unik yang dikaitkan dengannya. Dalam kebijakan, Anda menggunakan Amazon Resource Name (ARN) untuk mengidentifikasi sumber daya yang mengikuti kebijakan tersebut. Untuk informasi selengkapnya tentang ARN, lihat [Nama Sumber Daya Amazon \(ARN\) AWS dan Ruang Nama Layanan](#) di. Referensi Umum Amazon Web Services CodeCommit saat ini tidak mendukung jenis sumber daya lain, yang disebut sebagai subsumber daya.

Tabel berikut menjelaskan cara menentukan CodeCommit sumber daya.

Jenis Sumber Daya	Format ARN
Repositori	<code>arn:aws:codecommit:<i>wilayah</i>:<i>account-id</i> :<i>Repositori nama</i></code>
Semua CodeCommit repositori	<code>arn:aws:codecommit:*</code>
Semua CodeCommit repositori yang dimiliki oleh akun yang ditentukan dalam yang ditentukan Wilayah AWS	<code>arn:aws:codecommit:<i>wilayah</i>:<i>account-id</i> :*</code>

Note

Sebagian besar layanan AWS memperlakukan tanda titik dua (:) atau garis miring (/) sebagai karakter yang sama di ARN. Namun, CodeCommit membutuhkan kecocokan yang tepat dalam pola dan aturan sumber daya. Saat membuat pola peristiwa, pastikan untuk menggunakan karakter ARN yang benar sehingga cocok dengan ARN sintaks di sumber daya.

Misalnya, Anda dapat menunjukkan repositori tertentu (*MyDemoRepo*) dalam pernyataan Anda menggunakan ARN sebagai berikut:

```
"Resource": "arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo"
```

Untuk menentukan semua repositori milik akun tertentu, gunakan karakter wildcard (*) sebagai berikut:

```
"Resource": "arn:aws:codecommit:us-west-2:111111111111:*"
```

Untuk menentukan semua sumber daya, atau tindakan API tertentu yang tidak mendukung ARN, gunakan karakter wildcard (*) dalam elemen Resource seperti berikut ini:

```
"Resource": "*"
```

Anda juga dapat menggunakan karakter wildcard (*) untuk menentukan semua sumber daya yang cocok dengan bagian dari nama repositori. Misalnya, ARN berikut menentukan CodeCommit repositori apa pun yang dimulai dengan nama MyDemo dan yang terdaftar ke akun Amazon Web Services di: 111111111111 us-east-2 Wilayah AWS

```
arn:aws:codecommit:us-east-2:111111111111:MyDemo*
```

Untuk daftar operasi yang tersedia yang bekerja dengan CodeCommit sumber daya, lihat [Referensi izin CodeCommit](#).

Memahami kepemilikan sumber daya

Akun Amazon Web Services memiliki sumber daya yang dibuat dalam akun, terlepas dari orang yang membuat sumber daya tersebut. Secara khusus, pemilik sumber daya adalah akun Amazon Web

Services dari [entitas utama](#) (yaitu, akun root, pengguna IAM, atau IAM role) yang mengautentikasi permintaan pembuatan sumber daya. Contoh berikut menggambarkan cara kerjanya:

- Jika Anda membuat pengguna IAM di akun Amazon Web Services dan memberikan izin untuk membuat CodeCommit sumber daya kepada pengguna tersebut, pengguna dapat membuat CodeCommit sumber daya. Namun, akun Amazon Web Services Anda, yang menjadi milik pengguna, memiliki CodeCommit sumber daya.
- Jika Anda menggunakan kredensi akun root dari akun Amazon Web Services Anda untuk membuat aturan, akun Amazon Web Services Anda adalah pemilik sumber daya. CodeCommit
- Jika Anda membuat peran IAM di akun Amazon Web Services dengan izin untuk membuat CodeCommit sumber daya, siapa pun yang dapat mengambil peran tersebut dapat membuat CodeCommit sumber daya. Akun Amazon Web Services Anda, tempat peran tersebut berada, memiliki CodeCommit sumber daya.

Mengelola akses ke sumber daya

Untuk mengelola akses ke sumber daya AWS, Anda menggunakan kebijakan izin. Kebijakan izin menjelaskan siapa yang memiliki akses ke objek. Bagian berikut menjelaskan pilihan yang tersedia untuk membuat kebijakan izin.

Note

Bagian ini membahas penggunaan IAM dalam konteks. CodeCommit Bagian ini tidak memberikan informasi detail tentang layanan IAM. Untuk informasi selengkapnya tentang IAM, lihat [Apa itu IAM?](#) dalam Panduan Pengguna IAM. Untuk informasi tentang sintaksis dan penjelasan kebijakan IAM, lihat [Referensi Kebijakan IAM](#) di Panduan Pengguna IAM.

Kebijakan yang terlampir pada identitas IAM disebut sebagai kebijakan berbasis identitas (kebijakan IAM). Kebijakan izin yang terlampir pada sumber daya disebut sebagai kebijakan berbasis sumber daya. Saat ini, hanya CodeCommit mendukung kebijakan berbasis identitas (kebijakan IAM).

Topik

- [Kebijakan berbasis identitas \(kebijakan IAM\)](#)
- [Kebijakan berbasis sumber daya](#)

Kebijakan berbasis identitas (kebijakan IAM)

Untuk mengelola akses ke sumber daya AWS, Anda melampirkan kebijakan izin ke identitas IAM. Di CodeCommit, Anda menggunakan kebijakan berbasis identitas untuk mengontrol akses ke repositori. Misalnya, Anda dapat melakukan hal berikut:

- Lampirkan kebijakan izin ke pengguna atau grup di akun Anda — Untuk memberikan izin pengguna untuk melihat CodeCommit sumber daya di CodeCommit konsol, lampirkan kebijakan izin berbasis identitas ke pengguna atau grup tempat pengguna tersebut berada.
- Lampirkan kebijakan izin untuk peran (untuk memberikan izin lintas akun) - Delegasi, seperti ketika Anda ingin memberikan akses lintas akun, melibatkan pengaturan kepercayaan antara akun yang memiliki sumber daya (rekening terpercaya), dan akun yang berisi pengguna yang perlu mengakses sumber daya (rekening terpercaya). Kebijakan izin memberikan izin yang diperlukan pengguna suatu peran untuk melaksanakan tugas yang diinginkan pada sumber daya. Kebijakan kepercayaan menentukan akun terpercaya mana yang diizinkan untuk memberikan izin kepada pengguna untuk menjalankan peran tersebut. Untuk informasi selengkapnya, lihat [Syarat dan Konsep IAM](#).

Untuk memberikan izin lintas-akun, lampirkan kebijakan izin berbasis identitas ke IAM role.

Misalnya, administrator di Akun A dapat membuat peran untuk memberikan izin lintas akun ke akun Amazon Web Services lain (misalnya, Akun B) atau layanan AWS sebagai berikut:

1. Administrator akun A membuat IAM role dan melampirkan kebijakan izin ke peran yang memberikan izin pada sumber daya di akun A.
2. Administrator akun A melampirkan kebijakan kepercayaan peran yang mengidentifikasi Akun B sebagai principal yang dapat mengambil peran tersebut.
3. Administrator Akun B kemudian dapat mendelegasikan izin untuk mengambil peran tersebut kepada pengguna mana pun di Akun B. Dengan melakukan ini, pengguna di Akun B dapat membuat atau mengakses sumber daya di Akun A. Jika Anda ingin memberikan izin layanan AWS untuk mengambil peran tersebut, maka prinsipal di kebijakan kepercayaan juga bisa menjadi prinsipal layanan AWS. Untuk informasi selengkapnya, lihat Delegasi dalam [Syarat dan Konsep IAM](#).

Untuk informasi selengkapnya tentang menggunakan IAM untuk mendelegasikan izin, lihat [Manajemen Akses](#) dalam Panduan Pengguna IAM.

Contoh kebijakan berikut memungkinkan pengguna untuk membuat cabang dalam repositori bernama: *MyDemoRepo*

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
      "Action" : [
        "codecommit:CreateBranch"
      ],
      "Resource" : "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo"
    }
  ]
}
```

Untuk membatasi panggilan dan sumber daya yang dapat diakses oleh pengguna di akun Anda, buat kebijakan IAM tertentu, lalu lampirkan kebijakan tersebut ke pengguna IAM. Untuk informasi selengkapnya tentang cara membuat peran IAM dan untuk menjelajahi contoh pernyataan kebijakan IAM CodeCommit, lihat [Contoh kebijakan identitas yang dikelola pelanggan](#)

Kebijakan berbasis sumber daya

Layanan lain, seperti Amazon S3, juga men-support kebijakan izin berbasis sumber daya. Misalnya, Anda dapat melampirkan kebijakan berbasis sumber daya ke bucket S3 untuk mengelola izin akses ke bucket tersebut. CodeCommit tidak mendukung kebijakan berbasis sumber daya, tetapi Anda dapat menggunakan tag untuk mengidentifikasi sumber daya, yang kemudian dapat Anda gunakan dalam kebijakan IAM. Untuk contoh kebijakan berbasis tag, lihat [Kebijakan berbasis identitas \(kebijakan IAM\)](#).

Pelingkupan sumber daya di CodeCommit

Di CodeCommit, Anda dapat membuat cakupan kebijakan dan izin berbasis identitas ke sumber daya, seperti yang dijelaskan dalam [CodeCommit sumber daya dan operasi](#). Namun, Anda tidak dapat membuat cakupan izin `ListRepositories` ke sumber daya. Alih-alih, Anda harus membuat cakupan untuk semua sumber daya (menggunakan `*` wildcard). Jika tidak, tindakan tersebut akan gagal.

Semua CodeCommit izin lainnya dapat dicakup ke sumber daya.

Menentukan elemen kebijakan: sumber daya, tindakan, efek, dan prinsipal

Anda dapat membuat kebijakan untuk mengizinkan atau menolak akses pengguna ke sumber daya, atau mengizinkan atau menolak pengguna untuk mengambil tindakan spesifik pada sumber daya tersebut. CodeCommit mendefinisikan satu set operasi API publik yang menentukan cara pengguna bekerja dengan layanan, apakah itu melalui CodeCommit konsol, SDKAWS CLI, atau dengan langsung memanggil API tersebut. Untuk memberikan izin untuk operasi API ini, CodeCommit tentukan serangkaian tindakan yang dapat Anda tentukan dalam kebijakan.

Sebagian operasi API dapat mensyaratkan izin untuk lebih dari satu tindakan. Untuk informasi selengkapnya tentang sumber daya dan operasi API, lihat [CodeCommit sumber daya dan operasi](#) dan [Referensi izin CodeCommit](#).

Berikut ini adalah elemen kebijakan dasar:

- Sumber daya – Untuk mengidentifikasi sumber daya yang menerapkan kebijakan, Anda menggunakan Amazon Resource Name (ARN). Untuk informasi selengkapnya, lihat [CodeCommit sumber daya dan operasi](#).
- Tindakan – Untuk mengidentifikasi operasi sumber daya yang ingin Anda izinkan atau tolak, Anda menggunakan kata kunci tindakan. Misalnya, tergantung pada yang ditentukan `Effect`, `codecommit:GetBranch` izin mengizinkan atau menolak pengguna untuk melakukan `GetBranch` operasi, yang mendapatkan detail tentang cabang di CodeCommit repositori.
- Pengaruh – Anda menetapkan pengaruh, baik mengizinkan atau menolak, ketika pengguna meminta tindakan tertentu. Jika Anda tidak secara eksplisit memberikan akses ke (mengizinkan) sumber daya, akses akan ditolak secara implisit. Anda juga dapat secara eksplisit menolak akses ke sumber daya untuk memastikan bahwa pengguna tidak dapat mengaksesnya, meskipun kebijakan yang berbeda memberikan akses.
- Principal — Dalam kebijakan berbasis identitas (kebijakan IAM), satu-satunya jenis kebijakan yang CodeCommit mendukung, pengguna yang melekat pada kebijakan tersebut adalah prinsipal implisit.

Untuk mempelajari selengkapnya tentang sintaks dan penjelasan kebijakan IAM, lihat [Referensi Kebijakan IAM](#) dalam Panduan Pengguna IAM.

Untuk tabel yang menunjukkan semua tindakan CodeCommit API dan sumber daya yang diterapkan, lihat [Referensi izin CodeCommit](#).

Menetapkan ketentuan dalam kebijakan

Saat Anda memberikan izin, Anda menggunakan bahasa kebijakan akses untuk IAM untuk menentukan syarat di mana kebijakan harus diterapkan. Misalnya, Anda mungkin ingin kebijakan diterapkan hanya setelah tanggal tertentu. Untuk informasi selengkapnya tentang menentukan syarat dalam bahasa kebijakan, lihat [Syarat](#) dan [Tata Bahasa Kebijakan](#) di Panduan Pengguna IAM.

Untuk menyatakan syarat, Anda menggunakan kunci kondisi yang telah ditentukan sebelumnya. Tidak ada kunci syarat khusus untuk CodeCommit. Namun, ada kunci syarat seluruh AWS yang dapat Anda gunakan sesuai kebutuhan. Untuk daftar lengkap kunci di seluruh AWS, lihat [Kunci yang Tersedia untuk Syarat](#) dalam Panduan Pengguna IAM.

Menggunakan kebijakan berbasis identitas (Kebijakan IAM) untuk CodeCommit

Contoh kebijakan berbasis identitas berikut menunjukkan bagaimana administrator akun dapat melampirkan kebijakan izin ke identitas IAM (pengguna, grup, dan peran) untuk memberikan izin untuk melakukan operasi pada sumber daya. CodeCommit

Important

Kami menyarankan Anda terlebih dahulu meninjau topik pengantar yang menjelaskan konsep dasar dan opsi yang tersedia untuk mengelola akses ke CodeCommit sumber daya Anda. Untuk informasi selengkapnya, lihat [Ikhtisar mengelola izin akses ke sumber daya Anda CodeCommit](#).

Topik

- [Izin yang diperlukan untuk menggunakan konsol CodeCommit](#)
- [Menampilkan sumber daya di konsol](#)
- [Kebijakan yang dikelola oleh AWS untuk CodeCommit](#)
- [Contoh-contoh kebijakan yang dikelola pelanggan](#)

Berikut ini adalah contoh kebijakan izin berbasis identitas:

```
{
  "Version": "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Allow",
```

```
"Action" : [  
  "codecommit:BatchGetRepositories"  
],  
"Resource" : [  
  "arn:aws:codecommit:us-east-2:111111111111:MyDestinationRepo",  
  "arn:aws:codecommit:us-east-2:111111111111:MyDemo*"  
]  
}  
]  
}
```

Kebijakan ini memiliki satu pernyataan yang memungkinkan pengguna untuk mendapatkan informasi tentang CodeCommit repositori bernama `MyDestinationRepo` dan semua CodeCommit repositori yang dimulai dengan nama `MyDemo` di Wilayah. **us-east-2**

Izin yang diperlukan untuk menggunakan konsol CodeCommit

Untuk melihat izin yang diperlukan untuk setiap operasi CodeCommit API, dan untuk informasi selengkapnya tentang CodeCommit operasi, lihat [Referensi izin CodeCommit](#).

Untuk memungkinkan pengguna menggunakan CodeCommit konsol, administrator harus memberi mereka izin untuk CodeCommit tindakan. Misalnya, Anda dapat melampirkan kebijakan [AWSCodeCommitPowerUser](#) terkelola atau yang setara dengan pengguna atau grup.

Selain izin yang diberikan kepada pengguna berdasarkan kebijakan berbasis identitas, CodeCommit memerlukan izin untuk tindakan (). AWS Key Management Service AWS KMS Pengguna IAM tidak perlu izin `Allow` eksplisit untuk tindakan ini, tetapi pengguna tidak boleh memiliki kebijakan terlampir yang menetapkan izin berikut untuk `Deny`:

```
"kms:Encrypt",  
"kms:Decrypt",  
"kms:ReEncrypt",  
"kms:GenerateDataKey",  
"kms:GenerateDataKeyWithoutPlaintext",  
"kms:DescribeKey"
```

Untuk informasi selengkapnya tentang enkripsi dan CodeCommit, lihat [AWS KMS dan enkripsi](#).

Menampilkan sumber daya di konsol

CodeCommit Konsol memerlukan `ListRepositories` izin untuk menampilkan daftar repositori untuk akun Amazon Web Services Anda di Wilayah AWS tempat Anda masuk. Konsol juga termasuk

fungsi Pergi ke sumber daya untuk secara cepat melakukan pencarian sensitif huruf besar/kecil untuk sumber daya. Pencarian ini dilakukan di akun Amazon Web Services Anda di Wilayah AWS tempat Anda masuk. Sumber daya berikut ditampilkan di seluruh layanan berikut:

- AWS CodeBuild: Bangun proyek
- AWS CodeCommit: Repositori
- AWS CodeDeploy: Aplikasi
- AWS CodePipeline: Alur

Untuk melakukan pencarian ini di sumber daya di semua layanan, Anda harus memiliki izin berikut:

- CodeBuild: `ListProjects`
- CodeCommit: `ListRepositories`
- CodeDeploy: `ListApplications`
- CodePipeline: `ListPipelines`

Hasil tidak dikembalikan untuk sumber daya layanan jika Anda tidak memiliki izin untuk layanan tersebut. Bahkan jika Anda memiliki izin untuk melihat sumber daya, sumber daya tertentu tidak akan dikembalikan jika ada Deny eksplisit untuk melihat sumber daya tersebut.

Kebijakan yang dikelola oleh AWS untuk CodeCommit

Menambahkan izin ke para pengguna, grup, dan peran lebih mudah dilakukan dengan menggunakan kebijakan terkelola AWS dibandingkan dengan menulis kebijakan sendiri. Dibutuhkan waktu dan keahlian untuk [membuat kebijakan terkelola pelanggan IAM](#) yang hanya menyediakan izin sesuai kebutuhan tim Anda. Untuk mulai dengan cepat, Anda dapat menggunakan kebijakan-kebijakan terkelola AWS kami. Kebijakan ini mencakup kasus penggunaan umum dan tersedia di akun Akun AWS Anda. Untuk informasi lebih lanjut tentang kebijakan terkelola AWS, lihat [kebijakan terkelola AWS](#) di Panduan Pengguna IAM.

Layanan AWS mempertahankan dan memperbarui kebijakan-kebijakan terkelola AWS. Anda tidak dapat mengubah izin yang ada dalam kebijakan-kebijakan yang dikelola AWS. Layanan terkadang menambahkan izin tambahan ke kebijakan yang dikelola AWS untuk mendukung fitur-fitur baru. Jenis pembaruan ini akan memengaruhi semua identitas (pengguna, grup, dan peran) di mana kebijakan tersebut dilampirkan. Layanan kemungkinan besar akan memperbarui kebijakan yang dikelola AWS saat ada fitur baru yang diluncurkan atau saat ada operasi baru yang tersedia. Layanan tidak

menghapus izin yang ada di kebijakan yang dikelola AWS, sehingga pembaruan-pembaruan yang terjadi pada kebijakan tidak akan membuat izin yang ada rusak.

Selain itu, AWS mendukung kebijakan-kebijakan terkelola untuk fungsi tugas yang mencakup beberapa layanan. Misalnya, kebijakan `ReadOnlyAccessAWSTerkelola` menyediakan akses hanya-baca ke semua AWS layanan dan sumber daya. Saat layanan meluncurkan fitur baru, AWS menambahkan izin hanya-baca untuk operasi dan sumber daya yang baru. Untuk melihat daftar dan deskripsi dari kebijakan-kebijakan fungsi tugas, lihat [kebijakan terkelola AWS untuk fungsi tugas](#) di Panduan Pengguna IAM.

AWS menangani banyak kasus penggunaan umum dengan menyediakan kebijakan IAM mandiri yang dibuat dan dikelola oleh AWS. Kebijakan terkelola AWS ini memberikan izin yang diperlukan untuk kasus penggunaan umum. Kebijakan terkelola CodeCommit juga memberikan izin untuk melakukan operasi di layanan lain, seperti IAM, Amazon SNS, dan CloudWatch Amazon Events, sebagaimana diperlukan untuk tanggung jawab bagi pengguna yang telah diberikan kebijakan tersebut. Misalnya, `AWSCodeCommitFullAccess` kebijakan tersebut adalah kebijakan pengguna tingkat administratif yang memungkinkan pengguna dengan kebijakan ini membuat dan mengelola aturan CloudWatch Acara untuk repositori (aturan yang namanya diawali) dan topik `codecommit` Amazon SNS untuk pemberitahuan tentang peristiwa terkait repositori (topik yang namanya diawali), serta mengelola repositori di `codecommit` CodeCommit

Kebijakan AWS terkelola berikut, yang dapat Anda lampirkan ke pengguna di akun Anda, khusus untuk CodeCommit.

Topik

- [AWSkebijakan terkelola: `AWSCodeCommitFullAccess`](#)
- [AWSkebijakan terkelola: `AWSCodeCommitPowerUser`](#)
- [AWSkebijakan terkelola: `AWSCodeCommitReadOnly`](#)
- [CodeCommit kebijakan dan pemberitahuan terkelola](#)
- [AWS CodeCommitkebijakan terkelola dan Amazon CodeGuru Reviewer](#)
- [CodeCommit pembaruan kebijakan AWS terkelola](#)

AWSkebijakan terkelola: `AWSCodeCommitFullAccess`

Anda dapat melampirkan kebijakan `AWSCodeCommitFullAccess` ke identitas-identitas IAM Anda. Kebijakan ini memberikan akses penuh ke CodeCommit. Terapkan kebijakan ini hanya untuk pengguna tingkat administratif yang ingin Anda berikan kontrol penuh atas CodeCommit

repositori dan sumber daya terkait di akun Amazon Web Services Anda, termasuk kemampuan untuk menghapus repositori.

AWSCodeCommitFullAccess Kebijakan tersebut berisi pernyataan kebijakan berikut:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "codecommit:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "CloudWatchEventsCodeCommitRulesAccess",
      "Effect": "Allow",
      "Action": [
        "events:DeleteRule",
        "events:DescribeRule",
        "events:DisableRule",
        "events:EnableRule",
        "events:PutRule",
        "events:PutTargets",
        "events:RemoveTargets",
        "events:ListTargetsByRule"
      ],
      "Resource": "arn:aws:events:*:*:rule/codecommit*"
    },
    {
      "Sid": "SNSTopicAndSubscriptionAccess",
      "Effect": "Allow",
      "Action": [
        "sns:CreateTopic",
        "sns>DeleteTopic",
        "sns:Subscribe",
        "sns:Unsubscribe",
        "sns:SetTopicAttributes"
      ],
      "Resource": "arn:aws:sns:*:*:codecommit*"
    },
    {
      "Sid": "SNSTopicAndSubscriptionReadAccess",
```

```
"Effect": "Allow",
"Action": [
  "sns:ListTopics",
  "sns:ListSubscriptionsByTopic",
  "sns:GetTopicAttributes"
],
"Resource": "*"
},
{
  "Sid": "LambdaReadOnlyListAccess",
  "Effect": "Allow",
  "Action": [
    "lambda:ListFunctions"
  ],
  "Resource": "*"
},
{
  "Sid": "IAMReadOnlyListAccess",
  "Effect": "Allow",
  "Action": [
    "iam:ListUsers"
  ],
  "Resource": "*"
},
{
  "Sid": "IAMReadOnlyConsoleAccess",
  "Effect": "Allow",
  "Action": [
    "iam:ListAccessKeys",
    "iam:ListSSHPublicKeys",
    "iam:ListServiceSpecificCredentials"
  ],
  "Resource": "arn:aws:iam::*:user/${aws:username}"
},
{
  "Sid": "IAMUserSSHKeys",
  "Effect": "Allow",
  "Action": [
    "iam:DeleteSSHPublicKey",
    "iam:GetSSHPublicKey",
    "iam:ListSSHPublicKeys",
    "iam:UpdateSSHPublicKey",
    "iam:UploadSSHPublicKey"
  ],
}
```

```
    "Resource": "arn:aws:iam::*:user/${aws:username}"
  },
  {
    "Sid": "IAMSelfManageServiceSpecificCredentials",
    "Effect": "Allow",
    "Action": [
      "iam:CreateServiceSpecificCredential",
      "iam:UpdateServiceSpecificCredential",
      "iam>DeleteServiceSpecificCredential",
      "iam:ResetServiceSpecificCredential"
    ],
    "Resource": "arn:aws:iam::*:user/${aws:username}"
  },
  {
    "Sid": "CodeStarNotificationsReadWriteAccess",
    "Effect": "Allow",
    "Action": [
      "codestar-notifications:CreateNotificationRule",
      "codestar-notifications:DescribeNotificationRule",
      "codestar-notifications:UpdateNotificationRule",
      "codestar-notifications>DeleteNotificationRule",
      "codestar-notifications:Subscribe",
      "codestar-notifications:Unsubscribe"
    ],
    "Resource": "*",
    "Condition": {
      "StringLike": {
        "codestar-notifications:NotificationsForResource": "arn:aws:codecommit:*"
      }
    }
  },
  {
    "Sid": "CodeStarNotificationsListAccess",
    "Effect": "Allow",
    "Action": [
      "codestar-notifications:ListNotificationRules",
      "codestar-notifications:ListTargets",
      "codestar-notifications:ListTagsForResource",
      "codestar-notifications:ListEventTypes"
    ],
    "Resource": "*"
  },
  {
    "Sid": "CodeStarNotificationsSNSTopicCreateAccess",
```

```

    "Effect": "Allow",
    "Action": [
      "sns:CreateTopic",
      "sns:SetTopicAttributes"
    ],
    "Resource": "arn:aws:sns:*:*:codestar-notifications*"
  },
  {
    "Sid": "AmazonCodeGuruReviewerFullAccess",
    "Effect": "Allow",
    "Action": [
      "codeguru-reviewer:AssociateRepository",
      "codeguru-reviewer:DescribeRepositoryAssociation",
      "codeguru-reviewer:ListRepositoryAssociations",
      "codeguru-reviewer:DisassociateRepository",
      "codeguru-reviewer:DescribeCodeReview",
      "codeguru-reviewer:ListCodeReviews"
    ],
    "Resource": "*"
  },
  {
    "Sid": "AmazonCodeGuruReviewerSLRCreation",
    "Action": "iam:CreateServiceLinkedRole",
    "Effect": "Allow",
    "Resource": "arn:aws:iam:*:*:role/aws-service-role/codeguru-
reviewer.amazonaws.com/AWSServiceRoleForAmazonCodeGuruReviewer",
    "Condition": {
      "StringLike": {
        "iam:AWSServiceName": "codeguru-reviewer.amazonaws.com"
      }
    }
  },
  {
    "Sid": "CloudWatchEventsManagedRules",
    "Effect": "Allow",
    "Action": [
      "events:PutRule",
      "events:PutTargets",
      "events>DeleteRule",
      "events:RemoveTargets"
    ],
    "Resource": "*",
    "Condition": {
      "StringEquals": {

```

```

        "events:ManagedBy": "codeguru-reviewer.amazonaws.com"
    }
}
},
{
    "Sid": "CodeStarNotificationsChatbotAccess",
    "Effect": "Allow",
    "Action": [
        "chatbot:DescribeSlackChannelConfigurations",
        "chatbot:ListMicrosoftTeamsChannelConfigurations"
    ],
    "Resource": "*"
},
{
    "Sid": "CodeStarConnectionsReadOnlyAccess",
    "Effect": "Allow",
    "Action": [
        "codestar-connections:ListConnections",
        "codestar-connections:GetConnection"
    ],
    "Resource": "arn:aws:codestar-connections:*:*:connection/*"
}
]
}

```

AWSkebijakan terkelola: AWSCodeCommitPowerUser

Anda dapat melampirkan kebijakan `AWSCodeCommitPowerUser` ke identitas-identitas IAM Anda. Kebijakan ini memungkinkan pengguna mengakses semua fungsi CodeCommit dan sumber daya terkait repositori, kecuali tidak memungkinkan mereka untuk menghapus repositori atau membuat atau menghapus sumber daya terkait CodeCommit repositori di layanan lain, seperti Amazon Events. AWS CloudWatch Kami merekomendasikan bahwa Anda menerapkan kebijakan ini untuk sebagian besar pengguna.

`AWSCodeCommitPowerUser` Kebijakan tersebut berisi pernyataan kebijakan berikut:

```

{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "codecommit:AssociateApprovalRuleTemplateWithRepository",

```

```

        "codecommit:BatchAssociateApprovalRuleTemplateWithRepositories",
        "codecommit:BatchDisassociateApprovalRuleTemplateFromRepositories",
        "codecommit:BatchGet*",
        "codecommit:BatchDescribe*",
        "codecommit:Create*",
        "codecommit>DeleteBranch",
        "codecommit>DeleteFile",
        "codecommit:Describe*",
        "codecommit:DisassociateApprovalRuleTemplateFromRepository",
        "codecommit:EvaluatePullRequestApprovalRules",
        "codecommit:Get*",
        "codecommit:List*",
        "codecommit:Merge*",
        "codecommit:OverridePullRequestApprovalRules",
        "codecommit:Put*",
        "codecommit:Post*",
        "codecommit:TagResource",
        "codecommit:Test*",
        "codecommit:UntagResource",
        "codecommit:Update*",
        "codecommit:GitPull",
        "codecommit:GitPush"
    ],
    "Resource": "*"
},
{
    "Sid": "CloudWatchEventsCodeCommitRulesAccess",
    "Effect": "Allow",
    "Action": [
        "events:DeleteRule",
        "events:DescribeRule",
        "events:DisableRule",
        "events:EnableRule",
        "events:PutRule",
        "events:PutTargets",
        "events:RemoveTargets",
        "events:ListTargetsByRule"
    ],
    "Resource": "arn:aws:events:*:*:rule/codecommit*"
},
{
    "Sid": "SNSTopicAndSubscriptionAccess",
    "Effect": "Allow",
    "Action": [

```

```
        "sns:Subscribe",
        "sns:Unsubscribe"
    ],
    "Resource": "arn:aws:sns:*:*:codecommit*"
},
{
    "Sid": "SNSTopicAndSubscriptionReadAccess",
    "Effect": "Allow",
    "Action": [
        "sns:ListTopics",
        "sns:ListSubscriptionsByTopic",
        "sns:GetTopicAttributes"
    ],
    "Resource": "*"
},
{
    "Sid": "LambdaReadOnlyListAccess",
    "Effect": "Allow",
    "Action": [
        "lambda:ListFunctions"
    ],
    "Resource": "*"
},
{
    "Sid": "IAMReadOnlyListAccess",
    "Effect": "Allow",
    "Action": [
        "iam:ListUsers"
    ],
    "Resource": "*"
},
{
    "Sid": "IAMReadOnlyConsoleAccess",
    "Effect": "Allow",
    "Action": [
        "iam:ListAccessKeys",
        "iam:ListSSHPublicKeys",
        "iam:ListServiceSpecificCredentials"
    ],
    "Resource": "arn:aws:iam:*:*:user/${aws:username}"
},
{
    "Sid": "IAMUserSSHKeys",
    "Effect": "Allow",
```

```

    "Action": [
      "iam:DeleteSSHPublicKey",
      "iam:GetSSHPublicKey",
      "iam:ListSSHPublicKeys",
      "iam:UpdateSSHPublicKey",
      "iam:UploadSSHPublicKey"
    ],
    "Resource": "arn:aws:iam::*:user/${aws:username}"
  },
  {
    "Sid": "IAMSelfManageServiceSpecificCredentials",
    "Effect": "Allow",
    "Action": [
      "iam:CreateServiceSpecificCredential",
      "iam:UpdateServiceSpecificCredential",
      "iam>DeleteServiceSpecificCredential",
      "iam:ResetServiceSpecificCredential"
    ],
    "Resource": "arn:aws:iam::*:user/${aws:username}"
  },
  {
    "Sid": "CodeStarNotificationsReadWriteAccess",
    "Effect": "Allow",
    "Action": [
      "codestar-notifications:CreateNotificationRule",
      "codestar-notifications:DescribeNotificationRule",
      "codestar-notifications:UpdateNotificationRule",
      "codestar-notifications:Subscribe",
      "codestar-notifications:Unsubscribe"
    ],
    "Resource": "*",
    "Condition": {
      "StringLike": {
        "codestar-notifications:NotificationsForResource": "arn:aws:codecommit:*"
      }
    }
  },
  {
    "Sid": "CodeStarNotificationsListAccess",
    "Effect": "Allow",
    "Action": [
      "codestar-notifications:ListNotificationRules",
      "codestar-notifications:ListTargets",
      "codestar-notifications:ListTagsForResource",

```



```
    "codestar-notifications:ListEventTypes"
  ],
  "Resource": "*"
},
{
  "Sid": "AmazonCodeGuruReviewerFullAccess",
  "Effect": "Allow",
  "Action": [
    "codeguru-reviewer:AssociateRepository",
    "codeguru-reviewer:DescribeRepositoryAssociation",
    "codeguru-reviewer:ListRepositoryAssociations",
    "codeguru-reviewer:DisassociateRepository",
    "codeguru-reviewer:DescribeCodeReview",
    "codeguru-reviewer:ListCodeReviews"
  ],
  "Resource": "*"
},
{
  "Sid": "AmazonCodeGuruReviewerSLRCreation",
  "Action": "iam:CreateServiceLinkedRole",
  "Effect": "Allow",
  "Resource": "arn:aws:iam::*:role/aws-service-role/codeguru-
reviewer.amazonaws.com/AWSServiceRoleForAmazonCodeGuruReviewer",
  "Condition": {
    "StringLike": {
      "iam:AWSServiceName": "codeguru-reviewer.amazonaws.com"
    }
  }
},
{
  "Sid": "CloudWatchEventsManagedRules",
  "Effect": "Allow",
  "Action": [
    "events:PutRule",
    "events:PutTargets",
    "events>DeleteRule",
    "events:RemoveTargets"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "events:ManagedBy": "codeguru-reviewer.amazonaws.com"
    }
  }
}
```

```
    },
    {
      "Sid": "CodeStarNotificationsChatbotAccess",
      "Effect": "Allow",
      "Action": [
        "chatbot:DescribeSlackChannelConfigurations",
        "chatbot:ListMicrosoftTeamsChannelConfigurations"
      ],
      "Resource": "*"
    },
    {
      "Sid": "CodeStarConnectionsReadOnlyAccess",
      "Effect": "Allow",
      "Action": [
        "codestar-connections:ListConnections",
        "codestar-connections:GetConnection"
      ],
      "Resource": "arn:aws:codestar-connections:*:*:connection/*"
    }
  ]
}
```

AWSkebijakan terkelola: AWSCodeCommitReadOnly

Anda dapat melampirkan kebijakan `AWSCodeCommitReadOnly` ke identitas-identitas IAM Anda. Kebijakan ini memberikan akses hanya-baca ke CodeCommit dan sumber daya terkait repositori di AWS layanan lain, serta kemampuan untuk membuat dan mengelola sumber daya terkait mereka sendiri (seperti kredensi Git dan kunci SSH untuk digunakan pengguna CodeCommit IAM mereka saat mengakses repositori). Terapkan kebijakan ini kepada pengguna yang ingin Anda beri kemampuan untuk membaca konten repositori, tetapi tidak membuat perubahan apa pun pada kontennya.

`AWSCodeCommitReadOnly` Kebijakan tersebut berisi pernyataan kebijakan berikut:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "codecommit:BatchGet*",
        "codecommit:BatchDescribe*",

```

```
        "codecommit:Describe*",
        "codecommit:EvaluatePullRequestApprovalRules",
        "codecommit:Get*",
        "codecommit:List*",
        "codecommit:GitPull"
    ],
    "Resource": "*"
},
{
    "Sid": "CloudWatchEventsCodeCommitRulesReadOnlyAccess",
    "Effect": "Allow",
    "Action": [
        "events:DescribeRule",
        "events:ListTargetsByRule"
    ],
    "Resource": "arn:aws:events:*:*:rule/codecommit*"
},
{
    "Sid": "SNSSubscriptionAccess",
    "Effect": "Allow",
    "Action": [
        "sns:ListTopics",
        "sns:ListSubscriptionsByTopic",
        "sns:GetTopicAttributes"
    ],
    "Resource": "*"
},
{
    "Sid": "LambdaReadOnlyListAccess",
    "Effect": "Allow",
    "Action": [
        "lambda:ListFunctions"
    ],
    "Resource": "*"
},
{
    "Sid": "IAMReadOnlyListAccess",
    "Effect": "Allow",
    "Action": [
        "iam:ListUsers"
    ],
    "Resource": "*"
},
{
```

```

    "Sid": "IAMReadOnlyConsoleAccess",
    "Effect": "Allow",
    "Action": [
        "iam:ListAccessKeys",
        "iam:ListSSHPublicKeys",
        "iam:ListServiceSpecificCredentials",
        "iam:GetSSHPublicKey"
    ],
    "Resource": "arn:aws:iam::*:user/${aws:username}"
},
{
    "Sid": "CodeStarNotificationsReadOnlyAccess",
    "Effect": "Allow",
    "Action": [
        "codestar-notifications:DescribeNotificationRule"
    ],
    "Resource": "*",
    "Condition": {
        "StringLike": {
            "codestar-
notifications:NotificationsForResource": "arn:aws:codecommit:*"
        }
    }
},
{
    "Sid": "CodeStarNotificationsListAccess",
    "Effect": "Allow",
    "Action": [
        "codestar-notifications:ListNotificationRules",
        "codestar-notifications:ListEventTypes",
        "codestar-notifications:ListTargets"
    ],
    "Resource": "*"
},
{
    "Sid": "AmazonCodeGuruReviewerReadOnlyAccess",
    "Effect": "Allow",
    "Action": [
        "codeguru-reviewer:DescribeRepositoryAssociation",
        "codeguru-reviewer:ListRepositoryAssociations",
        "codeguru-reviewer:DescribeCodeReview",
        "codeguru-reviewer:ListCodeReviews"
    ],
    "Resource": "*"
}

```

```

    },
    {
      "Sid": "CodeStarConnectionsReadOnlyAccess",
      "Effect": "Allow",
      "Action": [
        "codestar-connections:ListConnections",
        "codestar-connections:GetConnection"
      ],
      "Resource": "arn:aws:codestar-connections:*:*:connection/*"
    }
  ]
}

```

CodeCommit kebijakan dan pemberitahuan terkelola

AWS CodeCommit men-support notifikasi, yang dapat memberi tahu pengguna tentang perubahan penting ke repositori. Kebijakan terkelola untuk CodeCommit menyertakan pernyataan kebijakan untuk fungsionalitas notifikasi. Untuk informasi selengkapnya, lihat [Apa itu notifikasi?](#)

Izin yang terkait dengan notifikasi dalam kebijakan terkelola akses penuh

Kebijakan terkelola `AWSCodeCommitFullAccess` mencakup pernyataan berikut untuk mengizinkan akses penuh ke notifikasi. Pengguna dengan kebijakan terkelola ini juga dapat membuat dan mengelola topik Amazon SNS untuk notifikasi, berlangganan dan berhenti berlangganan topik, membuat daftar topik untuk dipilih sebagai target untuk aturan notifikasi, dan membuat daftar klien AWS Chatbot yang dikonfigurasi untuk Slack.

```

{
  "Sid": "CodeStarNotificationsReadWriteAccess",
  "Effect": "Allow",
  "Action": [
    "codestar-notifications:CreateNotificationRule",
    "codestar-notifications:DescribeNotificationRule",
    "codestar-notifications:UpdateNotificationRule",
    "codestar-notifications>DeleteNotificationRule",
    "codestar-notifications:Subscribe",
    "codestar-notifications:Unsubscribe"
  ],
  "Resource": "*",
  "Condition" : {
    "StringLike" : {"codestar-notifications:NotificationsForResource" :
"arn:aws:codecommit:*"}
  }
}

```

```

    }
  },
  {
    "Sid": "CodeStarNotificationsListAccess",
    "Effect": "Allow",
    "Action": [
      "codestar-notifications:ListNotificationRules",
      "codestar-notifications:ListTargets",
      "codestar-notifications:ListTagsForResource",
      "codestar-notifications:ListEventTypes"
    ],
    "Resource": "*"
  },
  {
    "Sid": "CodeStarNotificationsSNSTopicCreateAccess",
    "Effect": "Allow",
    "Action": [
      "sns:CreateTopic",
      "sns:SetTopicAttributes"
    ],
    "Resource": "arn:aws:sns:*:*:codestar-notifications*"
  },
  {
    "Sid": "CodeStarNotificationsChatbotAccess",
    "Effect": "Allow",
    "Action": [
      "chatbot:DescribeSlackChannelConfigurations",
      "chatbot:ListMicrosoftTeamsChannelConfigurations"
    ],
    "Resource": "*"
  }
}

```

Izin yang terkait dengan notifikasi dalam kebijakan terkelola hanya-baca

Kebijakan terkelola `AWSCodeCommitReadOnlyAccess` mencakup pernyataan berikut untuk mengizinkan akses penuh ke notifikasi. Pengguna yang menerapkan kebijakan terkelola ini dapat melihat pemberitahuan untuk sumber daya, tetapi tidak dapat membuat, mengelola, atau berlangganan ke sumber daya tersebut.

```

{
  "Sid": "CodeStarNotificationsPowerUserAccess",
  "Effect": "Allow",
  "Action": [

```

```

        "codestar-notifications:DescribeNotificationRule"
    ],
    "Resource": "*",
    "Condition" : {
        "StringLike" : {"codestar-notifications:NotificationsForResource" :
"arn:aws:codecommit:*"}
    }
},
{
    "Sid": "CodeStarNotificationsListAccess",
    "Effect": "Allow",
    "Action": [
        "codestar-notifications:ListNotificationRules",
        "codestar-notifications:ListEventTypes",
        "codestar-notifications:ListTargets"
    ],
    "Resource": "*"
}

```

Izin terkait notifikasi dalam kebijakan terkelola lainnya

Kebijakan terkelola `AWSCodeCommitPowerUser` mencakup pernyataan berikut untuk mengizinkan pengguna membuat, mengedit, dan berlangganan notifikasi. Mereka tidak dapat menghapus aturan notifikasi atau mengelola tag untuk sumber daya.

```

{
    "Sid": "CodeStarNotificationsReadWriteAccess",
    "Effect": "Allow",
    "Action": [
        "codestar-notifications:CreateNotificationRule",
        "codestar-notifications:DescribeNotificationRule",
        "codestar-notifications:UpdateNotificationRule",
        "codestar-notifications>DeleteNotificationRule",
        "codestar-notifications:Subscribe",
        "codestar-notifications:Unsubscribe"
    ],
    "Resource": "*",
    "Condition" : {
        "StringLike" : {"codestar-notifications:NotificationsForResource" :
"arn:aws:codecommit*"}
    }
},
{

```

```

    "Sid": "CodeStarNotificationsListAccess",
    "Effect": "Allow",
    "Action": [
        "codestar-notifications:ListNotificationRules",
        "codestar-notifications:ListTargets",
        "codestar-notifications:ListTagsForResource",
        "codestar-notifications:ListEventTypes"
    ],
    "Resource": "*"
},
{
    "Sid": "SNSTopicListAccess",
    "Effect": "Allow",
    "Action": [
        "sns:ListTopics"
    ],
    "Resource": "*"
},
{
    "Sid": "CodeStarNotificationsChatbotAccess",
    "Effect": "Allow",
    "Action": [
        "chatbot:DescribeSlackChannelConfigurations",
        "chatbot:ListMicrosoftTeamsChannelConfigurations"
    ],
    "Resource": "*"
}

```

Untuk informasi selengkapnya tentang IAM dan notifikasi, lihat [Identity and Access Management for AWS CodeStar Notifications](#).

AWS CodeCommit kebijakan terkelola dan Amazon CodeGuru Reviewer

CodeCommit mendukung Amazon CodeGuru Reviewer, layanan peninjauan kode otomatis yang menggunakan analisis program dan pembelajaran mesin untuk mendeteksi masalah umum dan merekomendasikan perbaikan dalam kode Java atau Python Anda. Kebijakan terkelola untuk CodeCommit menyertakan pernyataan kebijakan untuk fungsionalitas CodeGuru Peninjau. Untuk informasi selengkapnya, lihat [Apa Itu Amazon CodeGuru Reviewer](#).

Izin yang terkait dengan CodeGuru Reviewer di AWSCodeCommitFullAccess

Kebijakan AWSCodeCommitFullAccess terkelola mencakup pernyataan berikut untuk memungkinkan CodeGuru Reviewer dikaitkan dan dipisahkan dengan CodeCommit repositori.

Pengguna dengan kebijakan terkelola ini diterapkan juga dapat melihat status asosiasi antara CodeCommit repositori dan CodeGuru Peninjau dan melihat status pekerjaan peninjauan untuk permintaan tarik.

```
{
  "Sid": "AmazonCodeGuruReviewerFullAccess",
  "Effect": "Allow",
  "Action": [
    "codeguru-reviewer:AssociateRepository",
    "codeguru-reviewer:DescribeRepositoryAssociation",
    "codeguru-reviewer:ListRepositoryAssociations",
    "codeguru-reviewer:DisassociateRepository",
    "codeguru-reviewer:DescribeCodeReview",
    "codeguru-reviewer:ListCodeReviews"
  ],
  "Resource": "*"
},
{
  "Sid": "AmazonCodeGuruReviewerSLRCreation",
  "Action": "iam:CreateServiceLinkedRole",
  "Effect": "Allow",
  "Resource": "arn:aws:iam::*:role/aws-service-role/codeguru-reviewer.amazonaws.com/AWSServiceRoleForAmazonCodeGuruReviewer",
  "Condition": {
    "StringLike": {
      "iam:AWSServiceName": "codeguru-reviewer.amazonaws.com"
    }
  }
},
{
  "Sid": "CloudWatchEventsManagedRules",
  "Effect": "Allow",
  "Action": [
    "events:PutRule",
    "events:PutTargets",
    "events>DeleteRule",
    "events:RemoveTargets"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "events:ManagedBy": "codeguru-reviewer.amazonaws.com"
    }
  }
}
```

```

    }
  }

```

Izin yang terkait dengan CodeGuru Reviewer di AWSCodeCommitPowerUser

Kebijakan AWSCodeCommitPowerUser terkelola mencakup pernyataan berikut untuk memungkinkan pengguna mengaitkan dan memisahkan repositori dengan CodeGuru Peninjau, melihat status asosiasi, dan melihat status pekerjaan peninjauan untuk permintaan tarik.

```

{
  "Sid": "AmazonCodeGuruReviewerFullAccess",
  "Effect": "Allow",
  "Action": [
    "codeguru-reviewer:AssociateRepository",
    "codeguru-reviewer:DescribeRepositoryAssociation",
    "codeguru-reviewer:ListRepositoryAssociations",
    "codeguru-reviewer:DisassociateRepository",
    "codeguru-reviewer:DescribeCodeReview",
    "codeguru-reviewer:ListCodeReviews"
  ],
  "Resource": "*"
},
{
  "Sid": "AmazonCodeGuruReviewerSLRCreation",
  "Action": "iam:CreateServiceLinkedRole",
  "Effect": "Allow",
  "Resource": "arn:aws:iam::*:role/aws-service-role/codeguru-
reviewer.amazonaws.com/AWSServiceRoleForAmazonCodeGuruReviewer",
  "Condition": {
    "StringLike": {
      "iam:AWSServiceName": "codeguru-reviewer.amazonaws.com"
    }
  }
},
{
  "Sid": "CloudWatchEventsManagedRules",
  "Effect": "Allow",
  "Action": [
    "events:PutRule",
    "events:PutTargets",
    "events>DeleteRule",
    "events:RemoveTargets"
  ],

```

```
"Resource": "*",
"Condition": {
  "StringEquals": {
    "events:ManagedBy": "codeguru-reviewer.amazonaws.com"
  }
}
}
```

Izin yang terkait dengan CodeGuru Reviewer di AWSCodeCommitReadOnly

Kebijakan AWSCodeCommitReadOnlyAccess terkelola mencakup pernyataan berikut untuk mengizinkan akses hanya-baca ke status asosiasi CodeGuru Peninjau dan melihat status pekerjaan peninjauan untuk permintaan tarik. Pengguna yang menerapkan kebijakan terkelola ini tidak dapat mengaitkan atau memisahkan repositori.

```
{
  "Sid": "AmazonCodeGuruReviewerReadOnlyAccess",
  "Effect": "Allow",
  "Action": [
    "codeguru-reviewer:DescribeRepositoryAssociation",
    "codeguru-reviewer:ListRepositoryAssociations",
    "codeguru-reviewer:DescribeCodeReview",
    "codeguru-reviewer:ListCodeReviews"
  ],
  "Resource": "*"
}
```

Peran terkait layanan Amazon CodeGuru Reviewer

Saat Anda mengaitkan repositori dengan CodeGuru Reviewer, peran terkait layanan dibuat sehingga CodeGuru Reviewer dapat mendeteksi masalah dan merekomendasikan perbaikan untuk kode Java atau Python dalam permintaan tarik. Peran terkait layanan diberi nama `AWSServiceRoleForAmazonCodeGuruReviewer` Untuk informasi selengkapnya, lihat [Menggunakan Peran Tertaut Layanan untuk Amazon CodeGuru Reviewer](#).

Untuk informasi selengkapnya, lihat [Kebijakan Terkelola AWS](#) dalam Panduan Pengguna IAM.

CodeCommit pembaruan kebijakan AWS terkelola

Lihat detail tentang pembaruan kebijakan AWS terkelola CodeCommit sejak layanan ini mulai melacak perubahan ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan RSS feed on. [Riwayat dokumen Panduan Pengguna AWS CodeCommit](#)

Perubahan	Deskripsi	Tanggal
AWSkebijakan terkelola : AWSCodeCommitFullAccess dan AWSkebijakan terkelola: AWSCodeCommitPowerUser — Perbarui ke kebijakan yang ada	CodeCommit menambahkan izin ke kebijakan ini untuk mendukung jenis notifikasi tambahan yang digunakan AWS Chatbot. AWSCodeCommitFullAccess Kebijakan AWSCodeCommitPowerUser dan telah diubah untuk menambahkan izin, chatbot:ListMicrosoftTeamsChannelConfigurations .	16 Mei 2023
AWSkebijakan terkelola: AWSCodeCommitReadOnly – Perbarui ke kebijakan yang sudah ada	CodeCommit menghapus izin duplikat dari kebijakan. AWSCodeCommitReadOnly Telah diubah untuk menghapus izin duplikat, "iam:ListAccessKeys" .	Agustus 18, 2021
CodeCommit mulai melacak perubahan	CodeCommit mulai melacak perubahan untuk kebijakan AWS terkelolanya.	Agustus 18, 2021

Contoh-contoh kebijakan yang dikelola pelanggan

Anda dapat membuat kebijakan IAM kustom Anda sendiri untuk mengizinkan izin CodeCommit tindakan dan sumber daya. Anda dapat melampirkan kebijakan-kebijakan kustom ini ke pengguna

IAM atau grup yang memerlukan izin-izin tersebut. Anda juga dapat membuat kebijakan IAM kustom Anda sendiri untuk integrasi antara CodeCommit dan AWS layanan lainnya.

Topik

- [Contoh kebijakan identitas yang dikelola pelanggan](#)
- [Contoh kebijakan integrasi yang dikelola pelanggan](#)

Contoh kebijakan identitas yang dikelola pelanggan

Contoh berikut kebijakan IAM memberikan izin untuk berbagai CodeCommit tindakan. Gunakan mereka untuk membatasi CodeCommit akses bagi pengguna dan peran IAM Anda. Kebijakan ini mengontrol kemampuan untuk melakukan tindakan dengan CodeCommit konsol, API, AWS SDK, atau perangkat. AWS CLI

Note

Semua contoh menggunakan wilayah US West (Oregon) (us-west-2) dan berisi ID akun fiktif.

Contoh

- [Contoh 1: Izinkan pengguna untuk melakukan CodeCommit operasi dalam satu Wilayah AWS](#)
- [Contoh 2: Izinkan pengguna menggunakan Git untuk repositori tunggal](#)
- [Contoh 3: Izinkan pengguna terhubung dari rentang alamat IP tertentu untuk mengakses repositori](#)
- [Contoh 4: Tolak atau izinkan tindakan di cabang](#)
- [Contoh 5: Tolak atau izinkan tindakan pada repositori dengan tag](#)

Contoh 1: Izinkan pengguna untuk melakukan CodeCommit operasi dalam satu Wilayah AWS

Kebijakan izin berikut menggunakan karakter wildcard ("codecommit:*") untuk memungkinkan pengguna melakukan semua CodeCommit tindakan di Wilayah us-east-2 dan bukan dari yang lain. Wilayah AWS

```
{
  "Version": "2012-10-17",
```

```

    "Statement": [
      {
        "Effect": "Allow",
        "Action": "codecommit:*",
        "Resource": "arn:aws:codecommit:us-east-2:111111111111:*",
        "Condition": {
          "StringEquals": {
            "aws:RequestedRegion": "us-east-2"
          }
        }
      },
      {
        "Effect": "Allow",
        "Action": "codecommit:ListRepositories",
        "Resource": "*",
        "Condition": {
          "StringEquals": {
            "aws:RequestedRegion": "us-east-2"
          }
        }
      }
    ]
  }

```

Contoh 2: Izinkan pengguna menggunakan Git untuk repositori tunggal

Dalam CodeCommit, izin kebijakan `GitPull` IAM berlaku untuk setiap perintah klien Git di mana data diambil dari CodeCommit, termasuk, `git fetch`, `git clone`, dan sebagainya. Demikian pula, izin kebijakan `GitPush` IAM berlaku untuk setiap perintah klien Git di mana data dikirim ke CodeCommit. Misalnya, jika izin kebijakan IAM `GitPush` diatur ke `Allow`, pengguna dapat mendorong penghapusan cabang menggunakan protokol Git. Dorongan itu tidak terpengaruh oleh izin yang diterapkan ke operasi `DeleteBranch` untuk pengguna IAM tersebut. Izin `DeleteBranch` berlaku untuk tindakan yang dilakukan dengan konsol, AWS CLI, SDK, dan API, tetapi bukan protokol Git.

Contoh berikut memungkinkan pengguna tertentu untuk menarik dari, dan mendorong ke, CodeCommit repositori bernama: `MyDemoRepo`

```

{
  "Version": "2012-10-17",
  "Statement" : [
    {

```

```
    "Effect" : "Allow",
    "Action" : [
      "codecommit:GitPull",
      "codecommit:GitPush"
    ],
    "Resource" : "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo"
  }
]
```

Contoh 3: Izinkan pengguna terhubung dari rentang alamat IP tertentu untuk mengakses repositori

Anda dapat membuat kebijakan yang hanya memungkinkan pengguna untuk terhubung ke CodeCommit repositori jika alamat IP mereka berada dalam rentang alamat IP tertentu. Ada dua pendekatan yang sama-sama valid untuk ini. Anda dapat membuat Deny kebijakan yang melarang CodeCommit operasi jika alamat IP untuk pengguna tidak berada dalam blok tertentu, atau Anda dapat membuat Allow kebijakan yang memungkinkan CodeCommit operasi jika alamat IP untuk pengguna berada dalam blok tertentu.

Anda dapat membuat kebijakan Deny yang menolak akses ke semua pengguna yang tidak berada dalam kisaran IP tertentu. Misalnya, Anda dapat melampirkan kebijakan AWSCodeCommitPowerUser terkelola dan kebijakan yang dikelola pelanggan ke semua pengguna yang memerlukan akses ke repositori Anda. Contoh kebijakan berikut menolak semua CodeCommit izin untuk pengguna yang alamat IP-nya tidak berada dalam blok alamat IP yang ditentukan 203.0.113.0/16:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "codecommit:*"
      ],
      "Resource": "*",
      "Condition": {
        "NotIpAddress": {
          "aws:SourceIp": [
            "203.0.113.0/16"
          ]
        }
      }
    }
  ]
}
```

```

    }
  }
]
}

```

Contoh kebijakan berikut memungkinkan pengguna tertentu untuk mengakses CodeCommit repositori bernama MyDemoRepo dengan izin setara dari kebijakan AWSCodeCommitPowerUser terkelola hanya jika alamat IP mereka berada dalam blok alamat yang ditentukan 203.0.113.0/16:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "codecommit:BatchGetRepositories",
        "codecommit:CreateBranch",
        "codecommit:CreateRepository",
        "codecommit:Get*",
        "codecommit:GitPull",
        "codecommit:GitPush",
        "codecommit:List*",
        "codecommit:Put*",
        "codecommit:Post*",
        "codecommit:Merge*",
        "codecommit:TagResource",
        "codecommit:Test*",
        "codecommit:UntagResource",
        "codecommit:Update*"
      ],
      "Resource": "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",
      "Condition": {
        "IpAddress": {
          "aws:SourceIp": [
            "203.0.113.0/16"
          ]
        }
      }
    }
  ]
}

```


Contoh 4: Tolak atau izinkan tindakan di cabang

Anda dapat membuat kebijakan yang menolak izin pengguna untuk tindakan yang Anda tentukan di satu cabang atau lebih. Atau, Anda dapat membuat kebijakan yang mengizinkan tindakan pada satu atau beberapa cabang yang mungkin tidak mereka miliki di cabang repositori lainnya. Anda dapat menggunakan kebijakan ini dengan kebijakan terkelola (yang telah ditetapkan) yang sesuai. Untuk informasi selengkapnya, lihat [Batasi dorongan dan penggabungan ke cabang di AWS CodeCommit](#).

Misalnya, Anda dapat membuat Deny kebijakan yang menolak kemampuan pengguna untuk membuat perubahan pada cabang bernama main, termasuk menghapus cabang tersebut, dalam repositori bernama *MyDemoRepo*. Anda dapat menggunakan kebijakan ini dengan kebijakan `AWSCodeCommitPowerUserterkelola`. Pengguna dengan dua kebijakan ini diterapkan akan dapat membuat dan menghapus cabang, membuat permintaan tarik, dan semua tindakan lain sebagaimana diizinkan oleh `AWSCodeCommitPowerUser`, tetapi mereka tidak akan dapat mendorong perubahan ke cabang bernama main, menambah atau mengedit file di cabang utama di CodeCommit konsol, atau menggabungkan cabang atau permintaan tarik ke cabang utama. Karena Deny diterapkan untuk `GitPush`, Anda harus menyertakan pernyataan `Null` dalam kebijakan, untuk mengizinkan panggilan `GitPush` awal untuk dianalisis untuk validitas ketika pengguna membuat dorongan dari repos lokal mereka.

Tip

Jika Anda ingin membuat kebijakan yang berlaku untuk semua cabang yang bernama utama di semua repositori di akun Amazon Web Services Anda, untuk `Resource`, tentukan tanda bintang (*) bukan repositori ARN.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "codecommit:GitPush",
        "codecommit>DeleteBranch",
        "codecommit:PutFile",
        "codecommit:Merge*"
      ],
      "Resource": "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",
```

```

        "Condition": {
            "StringEqualsIfExists": {
                "codecommit:References": [
                    "refs/heads/main"
                ]
            },
            "Null": {
                "codecommit:References": "false"
            }
        }
    ]
}

```

Contoh kebijakan berikut mengizinkan pengguna untuk membuat perubahan pada cabang bernama main di semua repositori di akun Amazon Web Services. Ini tidak mengizinkan perubahan pada cabang lainnya. Anda dapat menggunakan kebijakan ini dengan kebijakan `AWSCodeCommitReadOnly` terkelola untuk mengizinkan push otomatis ke repositori di cabang utama. Karena `EfeknyaAllow`, kebijakan contoh ini tidak akan bekerja dengan kebijakan terkelola seperti `AWSCodeCommitPowerUser`.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "codecommit:GitPush",
        "codecommit:Merge*"
      ],
      "Resource": "*",
      "Condition": {
        "StringEqualsIfExists": {
          "codecommit:References": [
            "refs/heads/main"
          ]
        }
      }
    }
  ]
}

```

Contoh 5: Tolak atau izinkan tindakan pada repositori dengan tag

Anda dapat membuat kebijakan yang mengizinkan atau menolak tindakan pada repositori berdasarkan tag AWS yang terkait dengan repositori tersebut, lalu menerapkan kebijakan tersebut ke grup IAM yang Anda konfigurasi untuk mengelola pengguna IAM. *Misalnya, Anda dapat membuat kebijakan yang menolak semua CodeCommit tindakan pada repositori apa pun dengan Status kunci AWS tag dan nilai kunci Rahasia, lalu menerapkan kebijakan tersebut ke grup IAM yang Anda buat untuk pengembang umum (Pengembang).* Anda kemudian perlu memastikan bahwa pengembang yang bekerja pada repositori yang diberi tag tersebut bukan anggota grup *Pengembang* umum tersebut, melainkan milik grup IAM berbeda yang tidak memiliki kebijakan restriktif yang diterapkan (). SecretDevelopers

Contoh berikut menyangkal semua CodeCommit tindakan pada repositori yang ditandai dengan Status kunci dan nilai kunci Rahasia:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": [
        "codecommit:Associate*",
        "codecommit:Batch*",
        "codecommit:CancelUploadArchive",
        "codecommit:CreateBranch",
        "codecommit:CreateCommit",
        "codecommit:CreatePullRequest*",
        "codecommit:CreateRepository",
        "codecommit:CreateUnreferencedMergeCommit",
        "codecommit>DeleteBranch",
        "codecommit>DeleteCommentContent",
        "codecommit>DeleteFile",
        "codecommit>DeletePullRequest*",
        "codecommit>DeleteRepository",
        "codecommit:Describe*",
        "codecommit:DisassociateApprovalRuleTemplateFromRepository",
        "codecommit:EvaluatePullRequestApprovalRules",
        "codecommit:GetBlob",
        "codecommit:GetBranch",
        "codecommit:GetComment*",
        "codecommit:GetCommit",
```

```

    "codecommit:GetDifferences*",
    "codecommit:GetFile",
    "codecommit:GetFolder",
    "codecommit:GetMerge*",
    "codecommit:GetObjectIdentifier",
    "codecommit:GetPullRequest*",
    "codecommit:GetReferences",
    "codecommit:GetRepository*",
    "codecommit:GetTree",
    "codecommit:GetUploadArchiveStatus",
    "codecommit:Git*",
    "codecommit:ListAssociatedApprovalRuleTemplatesForRepository",
    "codecommit:ListBranches",
    "codecommit:ListPullRequests",
    "codecommit:ListTagsForResource",
    "codecommit:Merge*",
    "codecommit:OverridePullRequestApprovalRules",
    "codecommit:Post*",
    "codecommit:Put*",
    "codecommit:TagResource",
    "codecommit:TestRepositoryTriggers",
    "codecommit:UntagResource",
    "codecommit:UpdateComment",
    "codecommit:UpdateDefaultBranch",
    "codecommit:UpdatePullRequest*",
    "codecommit:UpdateRepository*",
    "codecommit:UploadArchive"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "aws:ResourceTag/Status": "Secret"
    }
  }
}
]
}

```

Anda dapat lebih menyempurnakan strategi ini dengan menentukan repositori tertentu, bukan semua repositori, sebagai sumber daya. Anda juga dapat membuat kebijakan yang memungkinkan CodeCommit tindakan pada semua repositori yang tidak ditandai dengan tag tertentu. Misalnya, kebijakan berikut mengizinkan AWSCodeCommitPowerUserizin yang setara untuk CodeCommit

tindakan, kecuali hanya mengizinkan CodeCommit tindakan pada repositori yang tidak ditandai dengan tag yang ditentukan:

Note

Contoh kebijakan ini hanya mencakup tindakan untuk CodeCommit. Ini tidak termasuk tindakan untuk AWS layanan lain yang termasuk dalam kebijakan yang `AWSCodeCommitPowerUser` dikelola. Untuk informasi selengkapnya, lihat [AWSkebijakan terkelola: AWSCodeCommitPowerUser](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "codecommit:Associate*",
        "codecommit:Batch*",
        "codecommit:CancelUploadArchive",
        "codecommit:CreateBranch",
        "codecommit:CreateCommit",
        "codecommit:CreatePullRequest*",
        "codecommit:CreateRepository",
        "codecommit:CreateUnreferencedMergeCommit",
        "codecommit>DeleteBranch",
        "codecommit>DeleteCommentContent",
        "codecommit>DeleteFile",
        "codecommit>DeletePullRequest*",
        "codecommit:Describe*",
        "codecommit:DisassociateApprovalRuleTemplateFromRepository",
        "codecommit:EvaluatePullRequestApprovalRules",
        "codecommit:GetBlob",
        "codecommit:GetBranch",
        "codecommit:GetComment*",
        "codecommit:GetCommit",
        "codecommit:GetDifferences*",
        "codecommit:GetFile",
        "codecommit:GetFolder",
        "codecommit:GetMerge*",
        "codecommit:GetObjectIdentifier",
        "codecommit:GetPullRequest*",

```

```

    "codecommit:GetReferences",
    "codecommit:GetRepository*",
    "codecommit:GetTree",
    "codecommit:GetUploadArchiveStatus",
    "codecommit:Git*",
    "codecommit:ListAssociatedApprovalRuleTemplatesForRepository",
    "codecommit:ListBranches",
    "codecommit:ListPullRequests",
    "codecommit:ListTagsForResource",
    "codecommit:Merge*",
    "codecommit:OverridePullRequestApprovalRules",
    "codecommit:Post*",
    "codecommit:Put*",
    "codecommit:TagResource",
    "codecommit:TestRepositoryTriggers",
    "codecommit:UntagResource",
    "codecommit:UpdateComment",
    "codecommit:UpdateDefaultBranch",
    "codecommit:UpdatePullRequest*",
    "codecommit:UpdateRepository*",
    "codecommit:UploadArchive"
  ],
  "Resource": "*",
  "Condition": {
    "StringNotEquals": {
      "aws:ResourceTag/Status": "Secret",
      "aws:ResourceTag/Team": "Saanvi"
    }
  }
},
{
  "Effect": "Allow",
  "Action": [
    "codecommit:CreateApprovalRuleTemplate",
    "codecommit:GetApprovalRuleTemplate",
    "codecommit:ListApprovalRuleTemplates",
    "codecommit:ListRepositories",
    "codecommit:ListRepositoriesForApprovalRuleTemplate",
    "codecommit:UpdateApprovalRuleTemplateContent",
    "codecommit:UpdateApprovalRuleTemplateDescription",
    "codecommit:UpdateApprovalRuleTemplateName"
  ],
  "Resource": "*"
}

```

```
]
}
```

Contoh kebijakan integrasi yang dikelola pelanggan

Bagian ini memberikan contoh kebijakan pengguna yang dikelola pelanggan yang memberikan izin untuk integrasi antara CodeCommit dan layanan lainnya. AWS Untuk contoh spesifik kebijakan yang memungkinkan akses lintas akun ke CodeCommit repositori, lihat. [Konfigurasi akses lintas akun ke AWS CodeCommit repositori menggunakan peran](#)

Note

Semua contoh menggunakan Wilayah US West (Oregon) (us-west-2) ketika Wilayah AWS diperlukan, dan berisi ID akun fiktif.

Contoh

- [Contoh 1: Buat kebijakan yang memungkinkan akses lintas-akun ke topik Amazon SNS](#)
- [Contoh 2: Buat kebijakan topik Amazon Simple Notification Service \(Amazon SNS\) untuk mengizinkan CloudWatch Amazon Events CodeCommit mempublikasikan acara ke topik](#)
- [Contoh 3: Membuat kebijakan untuk AWS Lambda integrasi dengan CodeCommit pemicu](#)

Contoh 1: Buat kebijakan yang memungkinkan akses lintas-akun ke topik Amazon SNS

Anda dapat mengonfigurasi CodeCommit repositori sehingga mendorong kode atau peristiwa lain memicu tindakan, seperti mengirim pemberitahuan dari Amazon Simple Notification Service (Amazon SNS). Jika Anda membuat topik Amazon SNS dengan akun yang sama yang digunakan untuk membuat CodeCommit repositori, Anda tidak perlu mengonfigurasi kebijakan atau izin IAM tambahan. Anda dapat membuat topik, dan kemudian membuat pemicu untuk repositori. Untuk informasi selengkapnya, lihat [Buat pemicu untuk topik Amazon SNS](#).

Namun, jika Anda ingin mengonfigurasi pemicu untuk menggunakan topik Amazon SNS di akun Amazon Web Services lainnya, Anda harus terlebih dahulu mengonfigurasi topik tersebut dengan kebijakan yang memungkinkan CodeCommit untuk mempublikasikan ke topik tersebut. Dari akun lain, buka konsol Amazon SNS, pilih topik dari daftar, dan untuk Tindakan topik lainnya, pilih Edit kebijakan topik. Pada tab Advanced, ubah kebijakan untuk topik yang memungkinkan CodeCommit untuk dipublikasikan ke topik tersebut. Sebagai contoh, jika kebijakan tersebut adalah kebijakan

default, Anda akan mengubah kebijakan sebagai berikut, mengubah item dalam *teks huruf miring merah* untuk mencocokkan nilai-nilai untuk repositori Anda, topik Amazon SNS, dan akun:

```
{
  "Version": "2008-10-17",
  "Id": "__default_policy_ID",
  "Statement": [
    {
      "Sid": "__default_statement_ID",
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "sns:Subscribe",
        "sns:ListSubscriptionsByTopic",
        "sns>DeleteTopic",
        "sns:GetTopicAttributes",
        "sns:Publish",
        "sns:RemovePermission",
        "sns:AddPermission",          "sns:SetTopicAttributes"
      ],
      "Resource": "arn:aws:sns:us-east-2:111111111111:NotMySNSTopic",
      "Condition": {
        "StringEquals": {
          "AWS:SourceOwner": "111111111111"
        }
      }
    },
    {
      "Sid": "CodeCommit-Policy_ID",
      "Effect": "Allow",
      "Principal": {
        "Service": "codecommit.amazonaws.com"
      },
      "Action": "sns:Publish",
      "Resource": "arn:aws:sns:us-east-2:111111111111:NotMySNSTopic",
      "Condition": {
        "StringEquals": {
          "AWS:SourceArn": "arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",
          "AWS:SourceAccount": "111111111111"
        }
      }
    }
  ]
}
```



```
    }  
  ]  
}
```

Contoh 2: Buat kebijakan topik Amazon Simple Notification Service (Amazon SNS) untuk mengizinkan CloudWatch Amazon Events CodeCommit mempublikasikan acara ke topik

Anda dapat mengonfigurasi CloudWatch Acara untuk dipublikasikan ke topik Amazon SNS saat peristiwa terjadi, termasuk CodeCommit peristiwa. Untuk melakukannya, Anda harus memastikan bahwa CloudWatch Acara memiliki izin untuk mempublikasikan peristiwa ke topik Amazon SNS Anda dengan membuat kebijakan untuk topik tersebut atau memodifikasi kebijakan yang ada untuk topik yang serupa dengan berikut ini:

```
{  
  "Version": "2008-10-17",  
  "Id": "__default_policy_ID",  
  "Statement": [  
    {  
      "Sid": "__default_statement_ID",  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": "*"  
      },  
      "Action": "sns:Publish",  
      "Resource": "arn:aws:sns:us-east-2:123456789012:MyTopic",  
      "Condition": {  
        "StringEquals": {  
          "AWS:SourceOwner": "123456789012"  
        }  
      }  
    },  
    {  
      "Sid": "Allow_Publish_Events",  
      "Effect": "Allow",  
      "Principal": {  
        "Service": "events.amazonaws.com"  
      },  
      "Action": "sns:Publish",  
      "Resource": "arn:aws:sns:us-east-2:123456789012:MyTopic"  
    }  
  ]  
}
```

Untuk informasi selengkapnya tentang CodeCommit dan CloudWatch Acara, lihat [Contoh CloudWatch Acara Acara Dari Layanan yang Didukung](#). Untuk informasi selengkapnya tentang IAM dan kebijakan bahasa, lihat [Tata bahasa kebijakan JSON IAM](#).

Contoh 3: Membuat kebijakan untuk AWS Lambda integrasi dengan CodeCommit pemicu

Anda dapat mengonfigurasi CodeCommit repositori sehingga kode mendorong atau peristiwa lain memicu tindakan, seperti memanggil fungsi. AWS Lambda Untuk informasi selengkapnya, lihat [Buatlah sebuah pemicu untuk sebuah fungsi Lambda](#). Informasi ini khusus untuk pemicu, dan bukan CloudWatch Peristiwa.

Jika Anda ingin pemicu menjalankan fungsi Lambda secara langsung (alih-alih menggunakan topik Amazon SNS untuk menjalankan fungsi Lambda), dan Anda tidak mengonfigurasi pemicu di konsol Lambda, Anda harus menyertakan pernyataan yang mirip dengan yang berikut ini dalam kebijakan berbasis sumber daya fungsi:

```
{
  "Statement":{
    "StatementId":"Id-1",
    "Action":"lambda:InvokeFunction",
    "Principal":"codecommit.amazonaws.com",
    "SourceArn":"arn:aws:codecommit:us-east-2:111111111111:MyDemoRepo",
    "SourceAccount":"111111111111"
  }
}
```

Saat mengonfigurasi CodeCommit pemicu yang memanggil fungsi Lambda secara manual, Anda juga harus menggunakan [AddPermission](#) perintah Lambda untuk memberikan izin untuk menjalankan fungsi tersebut. CodeCommit Sebagai contoh, lihat bagian [Untuk CodeCommit memungkinkan menjalankan fungsi Lambda](#) dari [Buatlah sebuah pemicu untuk sebuah fungsi Lambda yang sudah ada](#).

Untuk informasi selengkapnya tentang kebijakan sumber daya untuk fungsi Lambda, lihat [AddPermission](#) dan [Model Peristiwa Tarik/Dorong](#) di Panduan Pengembang. AWS Lambda

Referensi izin CodeCommit

Tabel berikut mencantumkan setiap operasi CodeCommit API, tindakan terkait yang dapat Anda berikan izin, dan format ARN sumber daya yang akan digunakan untuk memberikan izin. CodeCommit API dikelompokkan ke dalam tabel berdasarkan cakupan tindakan yang diizinkan oleh

API tersebut. Lihat saat menyiapkan [Kontrol akses](#) dan menulis kebijakan izin yang dapat Anda lampirkan ke identitas IAM (kebijakan berbasis identitas).

Saat Anda membuat kebijakan izin, Anda menentukan tindakan di bagian kebijakan bidang Action. Anda menentukan nilai sumber daya di bidang Resource sebagai ARN, dengan atau tanpa karakter wildcard (*).

Untuk menyatakan kondisi dalam CodeCommit kebijakan Anda, gunakan tombol kondisi AWS - wide. Untuk daftar lengkap kunci di seluruh AWS, lihat [Kunci yang Tersedia](#) di Panduan Pengguna IAM. Untuk informasi lengkap tentang tindakan, sumber daya, dan kunci kondisi CodeCommit dalam kebijakan IAM, lihat [Kunci tindakan, sumber daya, dan kondisi untuk AWS CodeCommit](#).

Note

Untuk menentukan tindakan, gunakan awalan `codecommit:` yang diikuti dengan nama operasi API (misalnya, `codecommit:GetRepository` atau `codecommit>CreateRepository`).

Menggunakan Wildcard

Untuk menentukan beberapa tindakan atau sumber daya, gunakan karakter wildcard (*) di ARN Anda. Misalnya, `codecommit:*` menentukan semua CodeCommit tindakan dan `codecommit:Get*` menentukan semua CodeCommit tindakan yang dimulai dengan kata. `Get` Contoh berikut memberikan akses ke semua repositori dengan nama yang dimulai dengan `MyDemo`.

```
arn:aws:codecommit:us-west-2:111111111111:MyDemo*
```

Anda dapat menggunakan wildcard hanya dengan sumber daya *Nama repositori* yang tercantum dalam tabel berikut. Anda tidak dapat menggunakan wildcard dengan sumber daya *wilayah* atau *account-id*. Untuk informasi lebih lanjut tentang wildcard, lihat [Pengidentifikasi IAM](#) dalam Panduan Pengguna IAM.

Topik

- [Izin yang diperlukan untuk perintah klien Git](#)
- [Izin untuk tindakan di cabang](#)
- [Izin untuk tindakan pada penggabungan](#)

- [Izin untuk tindakan pada permintaan tarik](#)
- [Izin untuk tindakan pada templat aturan persetujuan](#)
- [Izin yang diperlukan untuk tindakan pada file](#)
- [Izin untuk tindakan pada komentar](#)
- [Izin untuk tindakan pada kode yang dilakukan commit](#)
- [Izin untuk tindakan pada repositori](#)
- [Izin yang diperlukan untuk tindakan pada tag](#)
- [Izin yang diperlukan untuk tindakan pada pemicu](#)
- [Izin untuk tindakan pada integrasi CodePipeline](#)

Izin yang diperlukan untuk perintah klien Git

Dalam CodeCommit, izin kebijakan `GitPull` IAM berlaku untuk setiap perintah klien Git di mana data diambil dari CodeCommit, termasuk, `git fetch`, `git clone`, dan sebagainya. Demikian pula, izin kebijakan `GitPush` IAM berlaku untuk setiap perintah klien Git di mana data dikirim ke CodeCommit. Misalnya, jika izin kebijakan IAM `GitPush` diatur ke `Allow`, pengguna dapat mendorong penghapusan cabang menggunakan protokol Git. Dorongan itu tidak terpengaruh oleh izin yang diterapkan ke operasi `DeleteBranch` untuk pengguna IAM tersebut. Izin `DeleteBranch` berlaku untuk tindakan yang dilakukan dengan konsol, AWS CLI, SDK, dan API, tetapi bukan protokol Git.

`GitPull` dan `GitPush` adalah izin kebijakan IAM. Mereka bukan tindakan API.

CodeCommit Izin yang Diperlukan untuk Tindakan untuk Perintah Klien Git

GitPull

Tindakan: `codecommit:GitPull`

Diperlukan untuk menarik informasi dari CodeCommit repositori ke repo lokal. Ini adalah izin kebijakan IAM saja, bukan tindakan API.

Resource: `arn:aws:codecommit:region:account-id:repository-name`

GitPush

Tindakan: `codecommit:Git Push`

Diperlukan untuk mendorong informasi dari repo lokal ke CodeCommit repositori. Ini adalah izin kebijakan IAM saja, bukan tindakan API.

Resource: `arn:aws:codecommit:region:account-id:repository-name`

Izin untuk tindakan di cabang

Izin berikut mengizinkan atau menolak tindakan pada cabang di CodeCommit repositori. Izin ini hanya berkaitan dengan tindakan yang dilakukan di CodeCommit konsol dan dengan CodeCommit API, dan perintah yang dilakukan menggunakan AWS CLI Mereka tidak berkaitan dengan tindakan serupa yang dapat dilakukan dengan menggunakan protokol Git. Misalnya, perintah `git show-branch -r` menampilkan daftar cabang remote untuk repositori dan `commit` menggunakan protokol Git. Itu tidak terpengaruh oleh izin apa pun untuk CodeCommit `ListBranches` operasi.

Untuk informasi selengkapnya tentang kebijakan untuk cabang, lihat [Batasi dorongan dan penggabungan ke cabang di AWS CodeCommit](#) dan [Contoh-contoh kebijakan yang dikelola pelanggan](#).

CodeCommit Operasi API dan Izin yang Diperlukan untuk Tindakan di Cabang

[CreateBranch](#)

Tindakan: `codecommit:CreateBranch`

Diperlukan untuk membuat cabang di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[DeleteBranch](#)

Tindakan: `codecommit>DeleteBranch`

Diperlukan untuk menghapus cabang dari CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetBranch](#)

Tindakan: `codecommit:GetBranch`

Diperlukan untuk mendapatkan detail tentang cabang di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[ListBranches](#)

Tindakan: `codecommit:ListBranches`

Diperlukan untuk mendapatkan daftar cabang di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[MergeBranchesByFastForward](#)

Tindakan: `codecommit:MergeBranchesByFastForward`

Diperlukan untuk menggabungkan dua cabang menggunakan strategi penggabungan maju cepat dalam repositori. CodeCommit

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[MergeBranchesBySquash](#)

Tindakan: `codecommit:ListBranches`

Diperlukan untuk menggabungkan dua cabang menggunakan strategi penggabungan squash dalam repositori. CodeCommit

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[MergeBranchesByThreeWay](#)

Tindakan: `codecommit:ListBranches`

Diperlukan untuk menggabungkan dua cabang menggunakan strategi penggabungan tiga arah dalam repositori. CodeCommit

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[UpdateDefaultBranch](#)

Tindakan: `codecommit:UpdateDefaultBranch`

Diperlukan untuk mengubah cabang default di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

Izin untuk tindakan pada penggabungan

Izin berikut mengizinkan atau menolak tindakan pada penggabungan di CodeCommit repositori. Izin ini berkaitan dengan tindakan yang dilakukan dengan CodeCommit konsol dan CodeCommit API, dan perintah yang dilakukan menggunakan AWS CLI Mereka tidak berkaitan dengan tindakan serupa yang dapat dilakukan dengan menggunakan protokol Git. Untuk izin terkait pada cabang, lihat [Izin untuk tindakan di cabang](#). Untuk izin terkait pada permintaan tarik, lihat [Izin untuk tindakan pada permintaan tarik](#).

CodeCommit Operasi API dan Izin yang Diperlukan untuk Tindakan untuk Perintah Gabungan

[BatchDescribeMergeConflicts](#)

Tindakan: `codecommit:BatchDescribeMergeConflicts`

Diperlukan untuk mengembalikan informasi tentang konflik dalam penggabungan antara komit dalam repositori. CodeCommit

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[CreateUnreferencedMergeCommit](#)

Tindakan: `codecommit>CreateUnreferencedMergeCommit`

Diperlukan untuk membuat komit yang tidak direferensikan antara dua cabang atau komit dalam CodeCommit repositori untuk tujuan membandingkannya dan mengidentifikasi potensi konflik.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[DescribeMergeConflicts](#)

Tindakan: `codecommit:DescribeMergeConflicts`

Diperlukan untuk mengembalikan informasi tentang konflik gabungan antara versi dasar, sumber, dan tujuan file dalam kemungkinan penggabungan dalam repositori. CodeCommit

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetMergeCommit](#)

Tindakan: `codecommit:GetMergeCommit`

Diperlukan untuk mengembalikan informasi tentang penggabungan antara komit sumber dan tujuan dalam CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetMergeOptions](#)

Tindakan: `codecommit:GetMergeOptions`

Diperlukan untuk mengembalikan informasi tentang opsi penggabungan yang tersedia antara dua cabang atau penentu komit dalam repositori. CodeCommit

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

Izin untuk tindakan pada permintaan tarik

Izin berikut mengizinkan atau menolak tindakan pada permintaan tarik di CodeCommit repositori. Izin ini berkaitan dengan tindakan yang dilakukan dengan CodeCommit konsol dan CodeCommit API, dan perintah yang dilakukan menggunakan AWS CLI Mereka tidak berkaitan dengan tindakan serupa yang dapat dilakukan dengan menggunakan protokol Git. Untuk izin terkait pada komentar, lihat [Izin untuk tindakan pada komentar](#).

CodeCommit Operasi API dan Izin yang Diperlukan untuk Tindakan pada Permintaan Tarik

BatchGetPullRequests

Tindakan: `codecommit:BatchGetPullRequests`

Diperlukan untuk mengembalikan informasi tentang satu atau beberapa permintaan tarik dalam CodeCommit repositori. Ini adalah izin kebijakan IAM saja, bukan tindakan API yang dapat Anda panggil.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[CreatePullRequest](#)

Tindakan: `codecommit>CreatePullRequest`

Diperlukan untuk membuat permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[CreatePullRequestApprovalRule](#)

Tindakan: `codecommit>CreatePullRequestApprovalRule`

Diperlukan untuk membuat aturan persetujuan untuk permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

DeletePullRequestApprovalRule

Tindakan: `codecommit>DeletePullRequestApprovalRule`

Diperlukan untuk menghapus aturan persetujuan untuk permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

DescribePullRequestEvents

Tindakan: `codecommit:DescribePullRequestEvents`

Diperlukan untuk mengembalikan informasi tentang satu atau beberapa peristiwa permintaan tarik dalam CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

EvaluatePullRequestApprovalRules

Tindakan: `codecommit:EvaluatePullRequestApprovalRules`

Diperlukan untuk mengevaluasi apakah permintaan tarik telah memenuhi semua kondisi yang ditentukan dalam aturan persetujuan terkait dalam CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetCommentsForPullRequest

Tindakan: `codecommit:GetCommentsForPullRequest`

Diperlukan untuk mengembalikan komentar yang dibuat atas permintaan tarik.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetCommitsFromMergeBase

Tindakan: `codecommit:GetCommitsFromMergeBase`

Diperlukan untuk mengembalikan informasi tentang perbedaan antara commit dalam konteks penggabungan potensial. Ini adalah izin kebijakan IAM saja, bukan tindakan API yang dapat Anda panggil.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetMergeConflicts](#)

Tindakan: `codecommit:GetMergeConflicts`

Diperlukan untuk mengembalikan informasi tentang konflik gabungan antara sumber dan cabang tujuan dalam permintaan tarik.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetPullRequest](#)

Tindakan: `codecommit:GetPullRequest`

Diperlukan untuk mengembalikan informasi tentang permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetPullRequestApprovalStates](#)

Tindakan: `codecommit:GetPullRequestApprovalStates`

Diperlukan untuk mengembalikan informasi tentang status persetujuan untuk permintaan tarik yang ditentukan.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetPullRequestOverrideState](#)

Tindakan: `codecommit:GetPullRequestOverrideState`

Diperlukan untuk mengembalikan informasi tentang apakah aturan persetujuan telah disisihkan (ditimpa) untuk permintaan tarik, dan jika demikian, Amazon Resource Name (ARN) pengguna atau identitas yang mengesampingkan aturan dan persyaratannya untuk permintaan tarik.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[ListPullRequests](#)

Tindakan: `codecommit:ListPullRequests`

Diperlukan untuk membuat daftar permintaan tarik dalam repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[MergePullRequestByFastForward](#)

Tindakan: `codecommit:MergePullRequestByFastForward`

Diperlukan untuk menutup permintaan tarik dan mencoba untuk menggabungkan cabang sumber ke cabang tujuan permintaan tarik menggunakan strategi penggabungan maju-cepat.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[MergePullRequestBySquash](#)

Tindakan: `codecommit:MergePullRequestBySquash`

Diperlukan untuk menutup permintaan tarik dan mencoba menggabungkan cabang sumber ke cabang tujuan permintaan tarik menggunakan strategi penggabungan squash.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[MergePullRequestByThreeWay](#)

Tindakan: `codecommit:MergePullRequestByThreeWay`

Diperlukan untuk menutup permintaan tarik dan mencoba untuk menggabungkan cabang sumber ke cabang tujuan permintaan tarik menggunakan strategi gabungan tiga arah.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[OverridePullRequestApprovalRules](#)

Tindakan: `codecommit:OverridePullRequestApprovalRules`

Diperlukan untuk menyisihkan semua persyaratan aturan persetujuan untuk permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[PostCommentForPullRequest](#)

Tindakan: `codecommit:PostCommentForPullRequest`

Diperlukan untuk mengirim komentar pada permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

UpdatePullRequestApprovalRuleContent

Tindakan: `codecommit:UpdatePullRequestApprovalRuleContent`

Diperlukan untuk mengubah struktur aturan persetujuan untuk permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

UpdatePullRequestApprovalState

Tindakan: `codecommit:UpdatePullRequestApprovalState`

Diperlukan untuk memperbarui status persetujuan pada permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

UpdatePullRequestDescription

Tindakan: `codecommit:UpdatePullRequestDescription`

Diperlukan untuk mengubah deskripsi permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

UpdatePullRequestStatus

Tindakan: `codecommit:UpdatePullRequestStatus`

Diperlukan untuk mengubah status permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

UpdatePullRequestTitle

Tindakan: `codecommit:UpdatePullRequestTitle`

Diperlukan untuk mengubah judul permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

Izin untuk tindakan pada templat aturan persetujuan

Izin berikut mengizinkan atau menolak tindakan pada templat aturan persetujuan di CodeCommit repositori. Izin ini hanya berkaitan dengan tindakan yang dilakukan di CodeCommit konsol,

CodeCommit API, dan perintah yang dilakukan menggunakan. AWS CLI Mereka tidak berkaitan dengan tindakan serupa yang dapat dilakukan dengan menggunakan protokol Git. Untuk izin terkait pada permintaan tarik, lihat [Izin untuk tindakan pada permintaan tarik](#).

CodeCommit Operasi API dan Izin yang Diperlukan untuk Tindakan pada Template Aturan Persetujuan

[AssociateApprovalRuleTemplateWithRepository](#)

Tindakan: `codecommit:AssociateApprovalRuleTemplateWithRepository`

Diperlukan untuk mengaitkan templat dengan repositori tertentu dalam akun Amazon Web Services. Setelah dikaitkan, ini secara otomatis membuat aturan persetujuan yang cocok dengan kondisi templat pada setiap permintaan tarik yang dibuat dalam repositori yang ditentukan.

Sumber daya: *

[BatchAssociateApprovalRuleTemplateWithRepositories](#)

Tindakan: `codecommit:BatchAssociateApprovalRuleTemplateWithRepositories`

Diperlukan untuk mengaitkan suatu templat dengan satu atau lebih repositori tertentu dalam akun Amazon Web Services.

Sumber daya: *

[BatchDisassociateApprovalRuleTemplateFromRepositories](#)

Tindakan: `codecommit:BatchDisassociateApprovalRuleTemplateFromRepositories`

Diperlukan untuk memisahkan kaitan suatu templat dari satu atau lebih repositori tertentu dalam akun Amazon Web Services.

Sumber daya: *

[CreateApprovalRuleTemplate](#)

Tindakan: `codecommit>CreateApprovalRuleTemplate`

Diperlukan untuk membuat templat untuk aturan persetujuan yang kemudian dapat dikaitkan dengan satu atau lebih repositori di akun Amazon Web Services Anda.

Sumber daya: *

[DeleteApprovalRuleTemplate](#)

Tindakan: `codecommit:DeleteApprovalRuleTemplate`

Diperlukan untuk menghapus templat aturan persetujuan dari akun AWS.

Sumber daya: *

[DisassociateApprovalRuleTemplateFromRepository](#)

Tindakan: `codecommit:DisassociateApprovalRuleTemplateFromRepository`

Diperlukan untuk memisahkan kaitan templat yang ditentukan dari repositori di akun Amazon Web Services. Ini tidak menghapus aturan persetujuan pada permintaan tarik yang telah dibuat dengan templat.

Sumber daya: *

[GetApprovalRuleTemplate](#)

Tindakan: `codecommit:GetApprovalRuleTemplate`

Diperlukan untuk mengembalikan informasi tentang templat aturan persetujuan di akun Amazon Web Services.

Sumber daya: *

[ListApprovalRuleTemplates](#)

Tindakan: `codecommit:ListApprovalRuleTemplates`

Diperlukan untuk daftar persetujuan aturan templat dalam sebuah akun Amazon Web Services.

Sumber daya: *

[ListAssociatedApprovalRuleTemplatesForRepository](#)

Tindakan: `codecommit:ListAssociatedApprovalRuleTemplatesForRepository`

Diperlukan untuk daftar semua templat aturan persetujuan yang dikaitkan dengan repositori tertentu dalam akun Amazon Web Services.

Sumber daya: *

[ListRepositoriesForApprovalRuleTemplate](#)

Tindakan: `codecommit:ListRepositoriesForApprovalRuleTemplate`

Diperlukan untuk daftar semua repositori yang dikaitkan dengan templat aturan persetujuan tertentu di akun Amazon Web Services.

Sumber daya: *

[UpdateApprovalRuleTemplateContent](#)

Tindakan: `codecommit:UpdateApprovalRuleTemplateContent`

Diperlukan untuk memperbarui isi templat aturan persetujuan di akun Amazon Web Services.

Sumber daya: *

[UpdateApprovalRuleTemplateDescription](#)

Tindakan: `codecommit:UpdateApprovalRuleTemplateDescription`

Diperlukan untuk memperbarui deskripsi templat aturan persetujuan di akun Amazon Web Services.

Sumber daya: *

[UpdateApprovalRuleTemplateName](#)

Tindakan: `codecommit:UpdateApprovalRuleTemplateName`

Diperlukan untuk memperbarui nama templat aturan persetujuan dalam akun AWS.

Sumber daya: *

Izin yang diperlukan untuk tindakan pada file

Izin berikut memungkinkan atau menolak tindakan pada file individual di CodeCommit repositori. Izin ini hanya berkaitan dengan tindakan yang dilakukan di CodeCommit konsol, CodeCommit API, dan perintah yang dilakukan menggunakan AWS CLI Mereka tidak berkaitan dengan tindakan serupa yang dapat dilakukan dengan menggunakan protokol Git. Misalnya, `git push` perintah mendorong file baru dan diubah ke CodeCommit repositori dengan menggunakan protokol Git. Itu tidak terpengaruh oleh izin apa pun untuk CodeCommit `PutFile` operasi.

CodeCommit Operasi API dan Izin yang Diperlukan untuk Tindakan pada File Individu

[DeleteFile](#)

Tindakan: `codecommit>DeleteFile`

Diperlukan untuk menghapus file tertentu dari cabang tertentu dalam CodeCommit repositori dari konsol. CodeCommit

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetBlob

Tindakan: `codecommit:GetBlob`

Diperlukan untuk melihat konten yang dikodekan dari file individual dalam CodeCommit repositori dari konsol. CodeCommit

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetFile

Tindakan: `codecommit:GetFile`

Diperlukan untuk melihat konten yang dikodekan dari file tertentu dan metadatanya dalam CodeCommit repositori dari konsol. CodeCommit

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetFolder

Tindakan: `codecommit:GetFolder`

Diperlukan untuk melihat isi folder tertentu dalam CodeCommit repositori dari konsol. CodeCommit

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

PutFile

Tindakan: `codecommit:PutFile`

Diperlukan untuk menambahkan file baru atau yang dimodifikasi ke CodeCommit repositori dari CodeCommit konsol, CodeCommit API, atau file. AWS CLI

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

Izin untuk tindakan pada komentar

Izin berikut mengizinkan atau menolak tindakan pada komentar di CodeCommit repositori. Izin ini berkaitan dengan tindakan yang dilakukan dengan CodeCommit konsol dan CodeCommit API, dan

perintah yang dilakukan menggunakan AWS CLI Untuk izin terkait pada komentar dalam permintaan tarik, lihat [Izin untuk tindakan pada permintaan tarik](#).

CodeCommit Operasi API dan Izin yang Diperlukan untuk Tindakan di Repositori

[DeleteCommentContent](#)

Tindakan: `codecommit:DeleteCommentContent`

Diperlukan untuk menghapus isi dari komentar yang dibuat pada perubahan, file, atau commit dalam repositori. Komentar tidak dapat dihapus, tetapi isi komentar dapat dihapus jika pengguna memiliki izin ini.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetComment](#)

Tindakan: `codecommit:GetComment`

Diperlukan untuk mengembalikan informasi tentang komentar yang dibuat pada perubahan, file, atau komit dalam CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetCommentReactions](#)

Tindakan: `codecommit:GetCommentReactions`

Diperlukan untuk mengembalikan informasi tentang reaksi emoji ke komentar yang dibuat pada perubahan, file, atau komit di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetCommentsForComparedCommit](#)

Tindakan: `codecommit:GetCommentsForComparedCommit`

Diperlukan untuk mengembalikan informasi tentang komentar yang dibuat pada perbandingan antara dua komit dalam CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[PostCommentForComparedCommit](#)

Tindakan: `codecommit:PostCommentForComparedCommit`

Diperlukan untuk mengomentari perbandingan antara dua komit dalam CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

PostCommentReply

Tindakan: `codecommit:PostCommentReply`

Diperlukan untuk membuat balasan ke komentar tentang perbandingan antara komit atau permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

PutCommentReaction

Tindakan: `codecommit:PutCommentReaction`

Diperlukan untuk membalas komentar dengan emoji pada komit atau permintaan tarik di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

UpdateComment

Tindakan: `codecommit:UpdateComment`

Diperlukan untuk mengedit komentar pada perbandingan antara commit atau permintaan tarik. Komentar hanya dapat diedit oleh penulis komentar.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

Izin untuk tindakan pada kode yang dilakukan commit

Izin berikut mengizinkan atau menolak tindakan pada kode yang dilakukan ke CodeCommit repositori. Izin ini berkaitan dengan tindakan yang dilakukan dengan CodeCommit konsol dan CodeCommit API, dan perintah yang dilakukan menggunakan AWS CLI Mereka tidak berkaitan dengan tindakan serupa yang dapat dilakukan dengan menggunakan protokol Git. Misalnya, perintah `git commit` membuat sebuah commit untuk cabang dalam repositori menggunakan protokol Git. Itu tidak terpengaruh oleh izin apa pun untuk CodeCommit `CreateCommit` operasi.

Secara eksplisit menolak beberapa izin ini dapat mengakibatkan konsekuensi yang tidak terduga di konsol. CodeCommit Misalnya, mengatur `GetTree` ke `Deny` mencegah pengguna menavigasi

isi repositori di konsol, tetapi tidak memblokir pengguna untuk melihat isi file dalam repositori (jika mereka dikirim link ke file dalam email, misalnya). Mengatur `GetBlob` ke `Deny` mencegah pengguna melihat isi file, tetapi tidak memblokir pengguna dari browsing struktur repositori. Mengatur `GetCommit` ke `Deny` mencegah pengguna mengambil rincian tentang commit. Mengatur `GetObjectIdentifier` ke `Deny` memblokir sebagian besar fungsi penjelajahan kode. Jika Anda menyetel ketiga tindakan ini ke `Deny` dalam kebijakan, pengguna dengan kebijakan tersebut tidak dapat menelusuri kode di CodeCommit konsol.

CodeCommit Operasi API dan Izin yang Diperlukan untuk Tindakan pada Kode yang Berkomitmen

BatchGetCommits

Tindakan: `codecommit:BatchGetCommits`

Diperlukan untuk mengembalikan informasi tentang satu atau lebih commit dalam CodeCommit repositori. Ini adalah izin kebijakan IAM saja, bukan tindakan API yang dapat Anda panggil.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[CreateCommit](#)

Tindakan: `codecommit:CreateCommit`

Diperlukan untuk membuat sebuah commit.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetCommit](#)

Tindakan: `codecommit:GetCommit`

Diperlukan untuk mengembalikan informasi tentang commit.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetCommitHistory

Tindakan: `codecommit:GetCommitHistory`

Diperlukan untuk mengembalikan informasi tentang riwayat komit dalam repositori. Ini adalah izin kebijakan IAM saja, bukan tindakan API yang dapat Anda panggil.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetDifferences

Tindakan: `codecommit:GetDifferences`

Diperlukan untuk mengembalikan informasi tentang perbedaan dalam penentu komit (seperti cabang, tag, HEAD, ID commit, atau referensi yang sepenuhnya memenuhi syarat lainnya).

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetObjectIdentifier

Tindakan: `codecommit:GetObjectIdentifier`

Diperlukan untuk menyelesaikan gumpalan, pohon, dan commit untuk pengenalan mereka. Ini adalah izin kebijakan IAM saja, bukan tindakan API yang dapat Anda panggil.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetReferences

Tindakan: `codecommit:GetReferences`

Diperlukan untuk mengembalikan semua referensi, seperti cabang dan tag. Ini adalah izin kebijakan IAM saja, bukan tindakan API yang dapat Anda panggil.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetTree

Tindakan: `codecommit:GetTree`

Diperlukan untuk melihat isi pohon tertentu dalam CodeCommit repositori dari konsol. CodeCommit Ini adalah izin kebijakan IAM saja, bukan tindakan API yang dapat Anda panggil.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

Izin untuk tindakan pada repositori

Izin berikut mengizinkan atau menolak tindakan pada CodeCommit repositori. Izin ini berkaitan dengan tindakan yang dilakukan dengan CodeCommit konsol dan CodeCommit API, dan perintah yang dilakukan menggunakan. AWS CLI Mereka tidak berkaitan dengan tindakan serupa yang dapat dilakukan dengan menggunakan protokol Git.

CodeCommit Operasi API dan Izin yang Diperlukan untuk Tindakan di Repositori

[BatchGetRepositories](#)

Tindakan: `codecommit:BatchGetRepositories`

Diperlukan untuk mendapatkan informasi tentang beberapa CodeCommit repositori yang ada di akun Amazon Web Services. DiResource, Anda harus menentukan nama semua CodeCommit repositori tempat pengguna diizinkan (atau ditolak) informasinya.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[CreateRepository](#)

Tindakan: `codecommit>CreateRepository`

Diperlukan untuk membuat CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[DeleteRepository](#)

Tindakan: `codecommit>DeleteRepository`

Diperlukan untuk menghapus CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[GetRepository](#)

Tindakan: `codecommit:GetRepository`

Diperlukan untuk mendapatkan informasi tentang satu CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[ListRepositories](#)

Tindakan: `codecommit:ListRepositories`

Diperlukan untuk mendapatkan daftar nama dan ID sistem dari beberapa CodeCommit repositori untuk akun Amazon Web Services. Satu-satunya nilai yang diizinkan untuk Resource untuk tindakan ini adalah semua repositori (*).

Sumber Daya: *

[UpdateRepositoryDescription](#)

Tindakan: `codecommit:UpdateRepositoryDescription`

Diperlukan untuk mengubah deskripsi CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[UpdateRepositoryName](#)

Tindakan: `codecommit:UpdateRepositoryName`

Diperlukan untuk mengubah nama CodeCommit repositori. DiResource, Anda harus menentukan CodeCommit repositori yang diizinkan untuk diubah dan nama repositori baru.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

Izin yang diperlukan untuk tindakan pada tag

Izin berikut mengizinkan atau menolak tindakan pada AWS tag untuk CodeCommit sumber daya.

CodeCommit Operasi API dan Izin yang Diperlukan untuk Tindakan pada Tag

[ListTagsForResource](#)

Tindakan: `codecommit:ListTagsForResource`

Diperlukan untuk mengembalikan informasi tentang AWS tag yang dikonfigurasi pada sumber daya di CodeCommit.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[TagResource](#)

Tindakan: `codecommit:TagResource`

Diperlukan untuk menambah atau mengedit tag AWS untuk repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

[UntagResource](#)

Tindakan: `codecommit:UntagResource`

Diperlukan untuk menghapus AWS tag dari sumber daya di CodeCommit.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

Izin yang diperlukan untuk tindakan pada pemicu

Izin berikut mengizinkan atau menolak tindakan pada pemicu untuk CodeCommit repositori.

CodeCommit Operasi API dan Izin yang Diperlukan untuk Tindakan pada Pemicu

GetRepositoryTriggers

Tindakan: `codecommit:GetRepositoryTriggers`

Diperlukan untuk mengembalikan informasi tentang pemicu yang dikonfigurasi untuk repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

PutRepositoryTriggers

Tindakan: `codecommit:PutRepositoryTriggers`

Diperlukan untuk membuat, mengedit, atau menghapus pemicu untuk repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

TestRepositoryTriggers

Tindakan: `codecommit:TestRepositoryTriggers`

Diperlukan untuk menguji fungsi pemicu repositori dengan mengirimkan data ke topik atau fungsi yang dikonfigurasi untuk pemicu.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

Izin untuk tindakan pada integrasi CodePipeline

CodePipeline Agar dapat menggunakan CodeCommit repositori dalam tindakan sumber untuk pipeline, Anda harus memberikan semua izin yang tercantum dalam tabel berikut ke peran layanan. CodePipeline Jika izin ini tidak ditetapkan dalam peran layanan atau diatur ke **Deny**, alur tidak berjalan secara otomatis ketika perubahan dibuat ke repositori, dan perubahan tidak dapat dirilis secara manual.

CodeCommit Operasi API dan Izin yang Diperlukan untuk Tindakan pada CodePipeline Integrasi

GetBranch

Tindakan: `codecommit:GetBranch`

Diperlukan untuk mendapatkan detail tentang cabang di CodeCommit repositori.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetCommit

Tindakan: `codecommit:GetCommit`

Diperlukan untuk mengembalikan informasi tentang commit.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

UploadArchive

Tindakan: `codecommit:UploadArchive`

Diperlukan untuk mengizinkan peran layanan CodePipeline untuk mengunggah perubahan repositori ke dalam pipeline. Ini adalah izin kebijakan IAM saja, bukan tindakan API yang dapat Anda panggil.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

GetUploadArchiveStatus

Tindakan: `codecommit:GetUploadArchiveStatus`

Diperlukan untuk menentukan status unggahan arsip: apakah sedang berlangsung, selesai, dibatalkan, atau jika terjadi kesalahan. Ini adalah izin kebijakan IAM saja, bukan tindakan API yang dapat Anda panggil.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

CancelUploadArchive

Tindakan: `codecommit:CancelUploadArchive`

Diperlukan untuk membatalkan pengunggahan arsip ke alur. Ini adalah izin kebijakan IAM saja, bukan tindakan API yang dapat Anda panggil.

Sumber Daya: `arn:aws:codecommit:region:account-id:repository-name`

Cara kerja AWS CodeCommit dengan IAM

Sebelum Anda menggunakan IAM untuk mengelola akses CodeCommit, Anda harus memahami fitur IAM apa yang tersedia untuk digunakan. CodeCommit Untuk mendapatkan tampilan tingkat tinggi tentang cara CodeCommit dan layanan AWS lainnya bekerja dengan IAM, lihat [Layanan AWS yang bekerja dengan IAM](#) dalam Panduan Pengguna IAM.

Topik

- [Kunci syarat](#)
- [Contoh](#)

Kunci syarat

Administrator dapat menggunakan kebijakan JSON AWS untuk menentukan siapa yang memiliki akses ke hal apa. Yaitu, pengguna utama manakah yang dapat melakukan tindakan pada sumber daya apa, dan dengan syarat apa.

Elemen `Condition` (atau blok `Condition`) akan memungkinkan Anda menentukan syarat yang menjadi dasar suatu pernyataan berlaku. Elemen `Condition` bersifat opsional. Anda dapat membuat ekspresi bersyarat yang menggunakan [operator syarat](#), misalnya sama dengan atau kurang dari, untuk mencocokkan syarat dalam kebijakan dengan nilai-nilai yang diminta.

Jika Anda menentukan beberapa elemen `Condition` dalam sebuah pernyataan, atau beberapa kunci dalam elemen `Condition` tunggal, maka AWS akan mengevaluasinya dengan menggunakan operasi AND yang logis. Jika Anda menentukan beberapa nilai untuk satu kunci persyaratan, maka AWS akan mengevaluasi syarat tersebut menggunakan operasi OR yang logis. Semua persyaratan harus dipenuhi sebelum izin pernyataan diberikan.

Anda juga dapat menggunakan variabel placeholder saat menentukan syarat. Sebagai contoh, Anda dapat memberikan izin kepada pengguna IAM untuk mengakses sumber daya hanya jika izin tersebut mempunyai tag yang sesuai dengan nama pengguna IAM mereka. Untuk informasi selengkapnya, silakan lihat [Elemen kebijakan IAM: variabel dan tag](#) di Panduan Pengguna IAM.

AWS mendukung kunci-kunci syarat global dan kunci-kunci syarat spesifik layanan. Untuk melihat semua kunci persyaratan global AWS, silakan lihat [kunci konteks syarat global AWS](#) di Panduan Pengguna IAM.

CodeCommit mendefinisikan kumpulan kunci kondisinya sendiri dan juga mendukung penggunaan beberapa kunci kondisi global. Untuk melihat semua kunci syarat global AWS, lihat [Kunci Konteks Syarat Global AWS](#) dalam Panduan Pengguna IAM.

Beberapa CodeCommit tindakan mendukung kunci `codecommit:References` kondisi. Untuk contoh kebijakan yang menggunakan kunci ini, lihat [Contoh 4: Tolak atau izinkan tindakan di cabang](#).

Untuk melihat daftar kunci CodeCommit kondisi, lihat [Condition Keys untuk AWS CodeCommit](#) di Panduan Pengguna IAM. Untuk mempelajari tindakan dan sumber daya mana yang dapat Anda gunakan kunci syarat, lihat [Tindakan yang Ditentukan oleh AWS CodeCommit](#).

Contoh

Untuk melihat contoh kebijakan CodeCommit berbasis identitas, lihat. [Contoh kebijakan berbasis identitas AWS CodeCommit](#)

CodeCommit Kebijakan berbasis sumber daya

CodeCommit tidak mendukung kebijakan berbasis sumber daya.

Otorisasi berdasarkan tag CodeCommit

Anda dapat melampirkan tag ke CodeCommit sumber daya atau meneruskan tag dalam permintaan CodeCommit. Untuk mengendalikan akses berdasarkan tag, berikan informasi tentang tag di [elemen syarat](#) dari sebuah kebijakan dengan menggunakan kunci-kunci persyaratan `codecommit:ResourceTag/key-name`, `aws:RequestTag/key-name`, atau `aws:TagKeys`. Untuk informasi selengkapnya tentang menandai CodeCommit sumber daya, lihat [Contoh 5: Tolak atau izinkan tindakan pada repositori dengan tag](#). Untuk informasi selengkapnya tentang strategi penandaan, lihat [Menandai Sumber Daya AWS](#).

CodeCommit juga mendukung kebijakan berdasarkan tag sesi. Untuk informasi lebih lanjut, lihat [Tanda Sesi](#).

Menggunakan tag untuk memberikan informasi identitas di CodeCommit

CodeCommit mendukung penggunaan tag sesi, yang merupakan atribut pasangan nilai kunci yang Anda lewati saat Anda mengambil peran IAM, menggunakan kredensial sementara, atau menyatukan pengguna di (). AWS Security Token Service AWS STS Anda juga dapat mengaitkan tag dengan pengguna IAM. Anda dapat menggunakan informasi yang disediakan dalam tag ini untuk

mempermudah mengidentifikasi siapa yang membuat perubahan atau menyebabkan suatu peristiwa. CodeCommit termasuk nilai untuk tag dengan nama kunci berikut dalam CodeCommit acara:

Nama kunci	Nilai
<code>displayName</code>	Nama yang dapat dibaca manusia untuk ditampilkan dan dikaitkan dengan pengguna (misalnya, Mary Major atau Saanvi Sarkar).
<code>emailAddress</code>	Alamat email yang ingin Anda tampilkan dan kaitkan dengan pengguna (misalnya, <code>mary_major@example.com</code> atau <code>saanvi_sarkar@example.com</code>).

Jika informasi ini diberikan, CodeCommit sertakan dalam acara yang dikirim ke Amazon EventBridge dan Amazon CloudWatch Events. Untuk informasi selengkapnya, lihat [Pemantauan CodeCommit peristiwa di Amazon EventBridge dan Amazon CloudWatch Peristiwa](#).

Untuk menggunakan penandaan sesi, peran harus memiliki kebijakan yang mencakup izin `sts:TagSession` yang diatur ke `Allow`. Jika Anda menggunakan akses gabungan, Anda dapat mengkonfigurasi nama tampilan dan informasi tag email sebagai bagian dari pengaturan Anda. Misalnya, jika Anda menggunakan Azure Active Directory, Anda dapat memberikan informasi klaim berikut:

Nama klaim	Nilai
<code>https://aws.amazon.com/SAML/Attributes/PrincipalTag:displayName</code>	<code>user.displayName</code>
<code>https://aws.amazon.com/SAML/Attributes/PrincipalTag:emailAddress</code>	<code>user.mail</code>

Anda dapat menggunakan AWS CLI untuk meneruskan tag sesi untuk `displayName` dan `emailAddress` menggunakan `AssumeRole`. Misalnya, pengguna yang ingin mengambil peran

bernama *Developer* yang ingin mengaitkan namanya *Mary Major* mungkin menggunakan `assume-role` perintah yang mirip dengan berikut ini:

```
aws sts assume-role \  
--role-arn arn:aws:iam::123456789012:role/Developer \  
--role-session-name Mary-Major \  
--tags Key=displayName,Value="Mary Major"  
      Key=emailAddress,Value="mary_major@example.com" \  
--external-id Example987
```

Untuk informasi lebih lanjut, lihat [AssumeRole](#).

Anda dapat menggunakan operasi `AssumeRoleWithSAML` untuk mengembalikan satu set kredensial sementara yang mencakup tanda `displayName` dan `emailAddress`. Anda dapat menggunakan tag ini saat mengakses CodeCommit repositori. Ini mengharuskan perusahaan atau grup Anda telah mengintegrasikan solusi SAML pihak ketiga Anda dengan AWS. Jika demikian, Anda dapat meneruskan atribut SAML sebagai tag sesi. Misalnya, jika Anda ingin meneruskan atribut identitas untuk nama tampilan dan alamat email untuk pengguna yang bernama *Sarkar Saanvi* sebagai tag sesi:

```
<Attribute Name="https://aws.amazon.com/SAML/Attributes/PrincipalTag:displayName">  
  <AttributeValue>Saanvi Sarkar</AttributeValue>  
</Attribute>  
<Attribute Name="https://aws.amazon.com/SAML/Attributes/PrincipalTag:emailAddress">  
  <AttributeValue>saanvi_sarkar@example.com</AttributeValue>  
</Attribute>
```

Untuk informasi selengkapnya, lihat [Melewati Tag Sesi menggunakan AssumeRoleWith SALL](#).

Anda dapat menggunakan operasi `AssumeRoleWithIdentity` untuk mengembalikan satu set kredensial sementara yang mencakup tanda `displayName` dan `emailAddress`. Anda dapat menggunakan tag ini saat mengakses CodeCommit repositori. Untuk meneruskan tag sesi dari OpenID Connect (OIDC), Anda harus menyertakan tag sesi di JSON Web Token (JWT). Misalnya, token JWP yang terdekodkan digunakan untuk memanggil `AssumeRoleWithWebIdentity` yang mencakup tag sesi `displayName` dan `emailAddress` untuk pengguna yang bernama *Li Juan*:

```
{  
  "sub": "lijuan",  
  "aud": "ac_oic_client",  
  "jti": "ZYUCeREXAMPLE",
```

```
"iss": "https://xyz.com",
"iat": 1566583294,
"exp": 1566583354,
"auth_time": 1566583292,
"https://aws.amazon.com/tags": {
  "principal_tags": {
    "displayName": ["Li Juan"],
    "emailAddress": ["li_juan@example.com"],
  },
  "transitive_tag_keys": [
    "displayName",
    "emailAddress"
  ]
}
}
```

Untuk informasi selengkapnya, lihat [Melewati Tag Sesi menggunakan AssumeRoleWithWebIdentity](#).

Anda dapat menggunakan operasi `GetFederationToken` untuk mengembalikan satu set kredensial sementara yang mencakup tanda `displayName` dan `emailAddress`. Anda dapat menggunakan tag ini saat mengakses CodeCommit repositori. Misalnya, untuk menggunakan AWS CLI untuk mendapatkan token gabungan yang mencakup tanda `displayName` dan `emailAddress`:

```
aws sts get-federation-token \
--name my-federated-user \
--tags key=displayName,value="Nikhil Jayashankar"
key=emailAddress,value=nikhil_jayashankar@example.com
```

Untuk informasi selengkapnya, lihat [Melewati Tag Sesi menggunakan GetFederationToken](#).

CodeCommit Peran IAM

[IAM role](#) adalah entitas di dalam akun Amazon Web Services Anda yang memiliki izin khusus.

Menggunakan kredensi sementara dengan CodeCommit

Anda dapat menggunakan kredensial sementara untuk masuk dengan gabungan, menjalankan IAM role, atau menjalankan peran lintas akun. Anda memperoleh kredensial keamanan sementara dengan memanggil operasi AWS STS API seperti [AssumeRole](#) atau [GetFederationToken](#).

CodeCommit mendukung menggunakan kredensial sementara. Untuk informasi selengkapnya, lihat [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#).

Peran terkait layanan

[Peran terkait layanan](#) memungkinkan layanan AWS untuk mengakses sumber daya di layanan lain untuk menyelesaikan tindakan atas nama Anda. Peran terkait layanan muncul di akun IAM Anda dan dimiliki oleh layanan tersebut. Administrator IAM dapat melihat tetapi tidak dapat mengedit izin untuk peran terkait layanan.

CodeCommit tidak menggunakan peran terkait layanan.

Peran layanan

Fitur ini memungkinkan layanan untuk menerima [peran layanan](#) atas nama Anda. Peran ini memungkinkan layanan untuk mengakses sumber daya di layanan lain untuk menyelesaikan tindakan atas nama Anda. Peran layanan muncul di akun IAM Anda dan dimiliki oleh akun tersebut. Ini berarti administrator IAM dapat mengubah izin untuk peran ini. Namun, melakukan hal itu dapat merusak fungsionalitas layanan.

CodeCommit tidak menggunakan peran layanan.

Contoh kebijakan berbasis identitas AWS CodeCommit

Secara default, pengguna dan peran IAM tidak memiliki izin untuk membuat atau memodifikasi sumber daya CodeCommit. Mereka juga tidak dapat melakukan tugas menggunakan API AWS Management Console, AWS CLI, or AWS. Administrator IAM harus membuat kebijakan IAM yang memberikan izin kepada pengguna dan peran untuk melakukan operasi API tertentu pada sumber daya yang diperlukan. Administrator kemudian harus melampirkan kebijakan tersebut ke pengguna IAM atau grup yang memerlukan izin tersebut.

Untuk contoh kebijakan, lihat berikut:

- [Contoh 1: Izinkan pengguna untuk melakukan CodeCommit operasi dalam satu Wilayah AWS](#)
- [Contoh 2: Izinkan pengguna menggunakan Git untuk repositori tunggal](#)
- [Contoh 3: Izinkan pengguna terhubung dari rentang alamat IP tertentu untuk mengakses repositori](#)
- [Contoh 4: Tolak atau izinkan tindakan di cabang](#)
- [Contoh 5: Tolak atau izinkan tindakan pada repositori dengan tag](#)
- [Konfigurasi akses lintas akun ke AWS CodeCommit repositori menggunakan peran](#)

Untuk mempelajari cara membuat kebijakan berbasis identitas IAM menggunakan contoh dokumen kebijakan JSON ini, lihat [Membuat Kebijakan pada Tab JSON](#) dalam Panduan Pengguna IAM.

Topik

- [Praktik terbaik kebijakan](#)
- [Menggunakan konsol CodeCommit](#)
- [Izinkan para pengguna untuk melihat izin mereka sendiri](#)
- [Melihat CodeCommit repositori berdasarkan tag](#)

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus CodeCommit sumber daya di akun Anda. Tindakan ini mengenakan biaya kepada Anda Akun AWS. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Mulailah dengan kebijakan terkelola AWS dan beralih ke izin dengan hak akses paling rendah – Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan terkelola AWS yang memberikan izin untuk banyak kasus penggunaan umum. Kebijakan terdapat di Akun AWS Anda. Kami menyarankan Anda untuk mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola pelanggan AWS yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, silakan lihat [kebijakan-kebijakan terkelola AWS](#) atau [kebijakan-kebijakan terkelola AWS untuk fungsi tugas](#) di Panduan Pengguna IAM.
- Menerapkan izin dengan hak akses paling rendah – Ketika Anda menetapkan izin dengan kebijakan IAM, hanya berikan izin yang diperlukan untuk melakukan tugas. Anda melakukan ini dengan mendefinisikan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, juga dikenal sebagai izin dengan hak akses paling rendah. Untuk informasi selengkapnya tentang cara menggunakan pengguna IAM untuk mengajukan izin, silakan lihat [Kebijakan dan izin di IAM](#) di Panduan Pengguna IAM.
- Gunakan syarat dalam kebijakan IAM untuk membatasi akses lebih lanjut – Anda dapat menambahkan suatu syarat ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Sebagai contoh, Anda dapat menulis syarat kebijakan untuk menentukan bahwa semua pengajuan harus dikirim menggunakan SSL. Anda juga dapat menggunakan syarat untuk memberi akses ke tindakan layanan jika digunakan melalui Layanan AWS yang spesifik, seperti AWS CloudFormation. Untuk informasi selengkapnya, silakan lihat [Elemen kebijakan JSON IAM: Syarat](#) di Panduan Pengguna IAM.
- Gunakan Analizer Akses IAM untuk memvalidasi kebijakan IAM Anda untuk memastikan izin yang aman dan fungsional – Analizer Akses IAM memvalidasi kebijakan baru dan yang sudah ada

sehingga kebijakan tersebut mematuhi bahasa kebijakan IAM (JSON) dan praktik terbaik IAM. Analizer Akses IAM menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, silakan lihat [validasi kebijakan Analizer Akses IAM](#) di Panduan Pengguna IAM.

- Memerlukan autentikasi multi-faktor (MFA) – Jika Anda memiliki skenario yang mengharuskan pengguna IAM atau pengguna root di Akun AWS Anda, aktifkan MFA untuk keamanan tambahan. Untuk meminta MFA ketika operasi API dipanggil, tambahkan syarat MFA pada kebijakan Anda. Untuk informasi selengkapnya, silakan lihat [Mengonfigurasi akses API yang diproteksi MFA](#) di Panduan Pengguna IAM.

Untuk informasi selengkapnya tentang praktik terbaik dalam IAM, silakan lihat [Praktik terbaik keamanan di IAM](#) di Panduan Pengguna IAM.

Menggunakan konsol CodeCommit

Untuk mengakses konsol AWS CodeCommit tersebut, Anda harus memiliki rangkaian izin minimum. Izin ini harus memungkinkan Anda untuk membuat daftar dan melihat detail tentang CodeCommit sumber daya di akun Amazon Web Services Anda. Jika Anda membuat kebijakan berbasis identitas yang lebih ketat daripada izin minimum yang diperlukan, konsol tersebut tidak akan berfungsi sebagaimana mestinya untuk entitas (pengguna IAM atau peran) dengan kebijakan tersebut.

Untuk memastikan bahwa entitas tersebut masih dapat menggunakan CodeCommit konsol, lampirkan juga kebijakan AWS terkelola berikut ke entitas. Untuk informasi selengkapnya, lihat [Menambahkan Izin ke Pengguna](#) dalam Panduan Pengguna IAM:

Untuk informasi selengkapnya, lihat [Menggunakan kebijakan berbasis identitas \(Kebijakan IAM\) untuk CodeCommit](#).

Anda tidak perlu mengizinkan konsol minimum untuk pengguna yang melakukan panggilan hanya ke API AWS CLI atau AWS. Alih-alih, izinkan akses hanya ke tindakan yang sesuai dengan operasi API yang Anda coba lakukan.

Izinkan para pengguna untuk melihat izin mereka sendiri

Contoh ini menunjukkan cara Anda dapat membuat kebijakan yang mengizinkan para pengguna IAM untuk melihat kebijakan inline dan terkelola yang dilampirkan ke identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan pada konsol atau secara terprogram menggunakan API AWS CLI atau AWS.


```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

Melihat CodeCommit *repositori* berdasarkan tag

Anda dapat menggunakan syarat dalam kebijakan berbasis identitas Anda untuk mengontrol akses ke sumber daya CodeCommit berdasarkan tanda. Untuk contoh kebijakan yang menunjukkan cara melakukan hal ini, lihat [Contoh 5: Tolak atau izinkan tindakan pada repositori dengan tag](#).

Untuk informasi selengkapnya, lihat [Elemen kebijakan IAM JSON: Syarat](#) dalam Panduan Pengguna IAM.

Pemecahan masalah identitas dan akses AWS CodeCommit

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan CodeCommit dan IAM.

Topik

- [Saya tidak berwenang untuk melakukan tindakan di CodeCommit](#)
- [Saya tidak berwenang untuk melakukan iam: PassRole](#)
- [Saya ingin melihat access key saya](#)
- [Saya seorang administrator dan ingin mengizinkan orang lain mengakses CodeCommit](#)
- [Saya ingin mengizinkan orang di luar akun Amazon Web Services saya untuk mengakses CodeCommit sumber daya saya](#)

Saya tidak berwenang untuk melakukan tindakan di CodeCommit

Jika AWS Management Console memberi tahu bahwa Anda tidak diotorisasi untuk melakukan tindakan, Anda harus menghubungi administrator untuk mendapatkan bantuan. Administrator Anda adalah orang yang memberi Anda kredensi masuk Anda.

Untuk informasi selengkapnya, silakan lihat [Izin yang diperlukan untuk menggunakan konsol CodeCommit](#)

Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan yang tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran CodeCommit.

Sebagian Layanan AWS mengizinkan Anda untuk memberikan peran yang sudah ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran tertaut-layanan. Untuk melakukan tindakan tersebut, Anda harus memiliki izin untuk memberikan peran pada layanan tersebut.

Contoh kesalahan berikut terjadi ketika pengguna IAM bernama `marymajor` mencoba menggunakan konsol tersebut untuk melakukan tindakan di CodeCommit. Namun, tindakan tersebut memerlukan layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda membutuhkan bantuan, hubungi administrator AWS Anda. Administrator Anda adalah orang yang memberikan kredensial masuk Anda.

Saya ingin melihat access key saya

Setelah membuat access key pengguna IAM, Anda dapat melihat access key ID Anda setiap saat. Namun, Anda tidak dapat melihat secret access key Anda lagi. Jika Anda kehilangan secret key, Anda harus membuat pasangan access key baru.

Access key terdiri dari dua bagian: access key ID (misalnya, AKIAIOSFODNN7EXAMPLE) dan secret access key (misalnya, wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY). Seperti nama pengguna dan kata sandi, Anda harus menggunakan access key ID dan secret access key sekaligus untuk mengautentikasi permintaan Anda. Kelola access key Anda seaman nama pengguna dan kata sandi Anda.

Important

Jangan memberikan access key Anda kepada pihak ke tiga, bahkan untuk membantu [menemukan ID pengguna kanonis Anda](#). Dengan melakukan ini, Anda mungkin memberi seseorang akses permanen ke AndaAkun AWS.

Saat Anda membuat pasangan access key, Anda diminta menyimpan access key ID dan secret access key di lokasi yang aman. secret access key hanya tersedia saat Anda membuatnya. Jika Anda kehilangan secret access key Anda, Anda harus menambahkan access key baru ke pengguna IAM Anda. Anda dapat memiliki maksimum dua access key. Jika Anda sudah memiliki dua, Anda harus menghapus satu pasangan kunci sebelum membuat pasangan baru. Untuk melihat instruksi, lihat [Mengelola access keys](#) di Panduan Pengguna IAM.

Saya seorang administrator dan ingin mengizinkan orang lain mengakses CodeCommit

Untuk memungkinkan orang lain mengakses CodeCommit, Anda harus membuat entitas IAM (pengguna atau peran) untuk orang atau aplikasi yang membutuhkan akses. Mereka akan menggunakan kredensial untuk entitas tersebut untuk mengakses AWS. Anda kemudian harus melampirkan kebijakan yang memberi mereka izin yang tepat di CodeCommit.

Untuk segera mulai, lihat [Membuat pengguna dan grup khusus IAM pertama Anda](#) di Panduan Pengguna IAM.

Saya ingin mengizinkan orang di luar akun Amazon Web Services saya untuk mengakses CodeCommit sumber daya saya

Untuk informasi selengkapnya, lihat [Konfigurasi akses lintas akun ke AWS CodeCommit repositori menggunakan peran](#).

Ketahanan di AWS CodeCommit

Infrastruktur global AWS dibangun di sekitar Wilayah AWS dan Availability Zone. Wilayah AWS menyediakan beberapa Availability Zone yang terpisah secara fisik dan terisolasi yang terhubung dengan jaringan latensi rendah, throughput tinggi, dan jaringan yang sangat berlebihan. Dengan Availability Zone, Anda dapat merancang serta mengoperasikan aplikasi dan basis data yang secara otomatis mengalami fail over antar zona tanpa gangguan. Zona Ketersediaan memiliki ketersediaan dan toleransi kesalahan yang lebih baik, dan dapat diskalakan dibandingkan infrastruktur pusat data tunggal atau multi tradisional.

SEBUAH CodeCommit repositori atau CodeCommit templat aturan persetujuan ada di templat aturan persetujuan Wilayah AWS di mana ia diciptakan. Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git untuk AWS CodeCommit](#). Untuk ketahanan dalam repositori, Anda dapat mengkonfigurasi klien Git Anda untuk mendorong ke dua repositori sekaligus. Untuk informasi selengkapnya, lihat [Dorong commit ke repositori Git tambahan](#).

Untuk informasi selengkapnya tentang Wilayah AWS dan Availability Zone, lihat [Infrastruktur Global AWS](#).

Keamanan infrastruktur dalam AWS CodeCommit

Sebagai layanan terkelola, AWS CodeCommit dilindungi oleh AWS prosedur keamanan jaringan global yang dijelaskan dalam [Amazon Web Services: Whitepaper Ikhtisar Proses Keamanan](#).

Anda menggunakan panggilan API yang dipublikasikan AWS untuk mengakses CodeCommit melalui jaringan. Klien harus mendukung Keamanan Lapisan Pengangkutan (TLS) 1.0 atau versi yang lebih baru. Kami merekomendasikan TLS 1.2 atau versi yang lebih baru. Klien juga harus mendukung suite cipher dengan perfect forward secrecy (PFS) seperti Ephemeral Diffie-Hellman (DHE) atau Elliptic

Curve Ephemeral Diffie-Hellman (ECDHE). Sebagian besar sistem modern seperti Java 7 dan yang lebih baru mendukung mode ini.

Permintaan harus ditandatangani menggunakan access key ID dan secret access key yang terkait dengan prinsipal IAM. Atau Anda dapat menggunakan [AWS Security Token Service](#) (AWS STS) untuk membuat kredensial keamanan sementara untuk menandatangani permintaan.

Anda dapat memanggil operasi API ini dari lokasi jaringan mana pun, tetapi CodeCommit mendukung pembatasan berdasarkan alamat IP sumber. Anda juga dapat menggunakan kebijakan CodeCommit untuk mengontrol akses dari titik akhir Amazon Virtual Private Cloud (Amazon VPC) tertentu atau VPC tertentu. Secara efektif, ini mengisolasi akses jaringan ke sumber daya CodeCommit yang diberikan hanya dari VPC tertentu dalam jaringan AWS.

Untuk informasi selengkapnya, lihat yang berikut:

- [Contoh 1: Izinkan pengguna untuk melakukan CodeCommit operasi dalam satu Wilayah AWS](#)
- [Contoh 3: Izinkan pengguna terhubung dari rentang alamat IP tertentu untuk mengakses repositori](#)
- [Menggunakan AWS CodeCommit dengan antarmuka VPC endpoint](#)

Memantau AWS CodeCommit

Pemantauan adalah bagian penting dari pemeliharaan keandalan, ketersediaan, dan kinerja CodeCommit dan yang lainAWSsolusi.AWSmenyediakan alat pemantauan berikut untuk diawasi CodeCommit, melaporkan saat terjadi kesalahan, dan mengambil tindakan otomatis jika diperlukan:

- Amazon EventBridge dapat digunakan untuk mengotomatisasiAWSlayanan dan merespons peristiwa sistem secara otomatis seperti masalah ketersediaan aplikasi atau perubahan sumber daya. Peristiwa dariAWSlayanan dikirimkan ke EventBridge dalam waktu dekat real time. Anda dapat menuliskan aturan sederhana untuk menunjukkan peristiwa mana yang sesuai kepentingan Anda, dan tindakan otomatis mana yang diambil ketika suatu peristiwa sesuai dengan suatu aturan. Untuk informasi selengkapnya, lihat[Amazon EventBridge Panduan Penggunaan](#)dan[Pemantauan CodeCommit peristiwa di Amazon EventBridge dan Amazon CloudWatch Peristiwa](#).
- Amazon CloudWatch Peristiwa memberikan aliran sistem secara hampir waktu-nyata yang menjelaskan perubahanAWSsumber daya. CloudWatch Peristiwa memungkinkan komputasi berbasis peristiwa otomatis, karena Anda dapat menulis aturan yang mengawasi peristiwa tertentu, dan memicu tindakan otomatis dalamAWSlayanan ketika peristiwa ini terjadi. Untuk informasi selengkapnya, lihat[Amazon CloudWatch Panduan Penggunaan](#)dan[Pemantauan CodeCommit peristiwa di Amazon EventBridge dan Amazon CloudWatch Peristiwa](#).
- Amazon CloudWatch Log dapat digunakan untuk memantau, menyimpan, dan mengakses berkas log CloudTrail dan sumber lainnya. CloudWatch Log dapat memantau informasi dalam berkas log dan memberi tahu Anda saat ambang tertentu terpenuhi. Anda juga dapat mengarsipkan data log Anda dalam penyimpanan yang sangat tahan lama. Untuk informasi selengkapnya, lihat[Amazon CloudWatch Panduan Pengguna](#).
- AWS CloudTrail merekam panggilan API dan peristiwa terkait yang dilakukan oleh atau atas nama akun Amazon Web Services Anda dan mengirimkan berkas log ke bucket Amazon S3 yang Anda tentukan. Anda dapat mengidentifikasi pengguna dan akun yang memanggil AWS, alamat IP asal panggilan dilakukan, dan waktu panggilan terjadi. Untuk informasi selengkapnya, lihat[Panduan Pengguna AWS CloudTrail](#) dan [Mencatat panggilan API AWS CodeCommit dengan AWS CloudTrail](#).

Pemantauan CodeCommit peristiwa di Amazon EventBridge dan Amazon CloudWatch Peristiwa

Anda dapat memonitor AWS CodeCommit peristiwa EventBridge, yang memberikan aliran data real-time dari aplikasi Anda sendiri, software-as-a-service (SaaS) aplikasi, dan AWS jasa. EventBridge merutekan data ke target seperti AWS Lambda dan Amazon Simple Notification Service. Peristiwa ini sama dengan yang muncul di Amazon CloudWatch Peristiwa, yang memberikan aliran sistem secara hampir waktu-nyata yang menjelaskan perubahan dalam AWS sumber daya.

Contoh berikut ini menunjukkan peristiwa CodeCommit.

Note

CodeCommit mendukung penyedia `displayName` dan `emailAddress` informasi yang disertakan dalam tag sesi, jika informasi tersedia. Untuk informasi lebih lanjut, lihat [Tanda Sesi](#) dan [Menggunakan tag untuk memberikan informasi identitas di CodeCommit](#).

Topik

- [Peristiwa `referenceCreated`](#)
- [Peristiwa `referenceUpdated`](#)
- [Peristiwa `referenceDeleted`](#)
- [unreferencedMergeCommitPeristiwa](#)
- [commentOnCommitPeristiwa](#)
- [commentOnCommitPeristiwa](#)
- [commentOnPullRequestCreated peristiwa](#)
- [commentOnPullRequestUpdated peristiwa](#)
- [pullRequestCreated peristiwa](#)
- [pullRequestSourceBranchUpdated peristiwa](#)
- [pullRequestStatusBerubah acara](#)
- [pullRequestMergeStatusUpdated peristiwa](#)
- [approvalRuleTemplatePeristiwa](#)
- [approvalRuleTemplatePeristiwa](#)

- [approvalRuleTemplatePeristiwa](#)
- [approvalRuleTemplateAssociatedWithRepository peristiwa](#)
- [approvalRuleTemplateDisassociatedWithRepository peristiwa](#)
- [approvalRuleTemplateBatchAssociatedWithRepositories peristiwa](#)
- [approvalRuleTemplateBatchDisassociatedFromRepositories peristiwa](#)
- [pullRequestApprovalRuleCreated peristiwa](#)
- [pullRequestApprovalRuleDeleted peristiwa](#)
- [pullRequestApprovalRuleOverriden peristiwa](#)
- [pullRequestApprovalStateChanged peristiwa](#)
- [pullRequestApprovalRuleUpdated peristiwa](#)
- [Peristiwa reactionCreated](#)
- [Peristiwa reactionUpdated](#)

Peristiwa referenceCreated

Dalam contoh peristiwa ini, sebuah cabang bernama myBranch telah dibuat dalam sebuah repositori bernama MyDemoRepo.

```
{
  "version": "0",
  "id": "01234567-EXAMPLE",
  "detail-type": "CodeCommit Repository State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-06-12T10:23:43Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "event": "referenceCreated",
    "repositoryName": "MyDemoRepo",
    "repositoryId": "12345678-1234-5678-abcd-12345678abcd",
    "referenceType": "branch",
    "referenceName": "myBranch",
    "referenceFullName": "refs/heads/myBranch",
```



```
    "commitId": "3e5983DESTINATION"
  }
}
```

Peristiwa referenceUpdated

Dalam contoh peristiwa ini, sebuah cabang bernama myBranch telah diperbarui oleh suatu penggabungan dalam sebuah repositori bernama MyDemoRepo.

```
{
  "version": "0",
  "id": "01234567-EXAMPLE",
  "detail-type": "CodeCommit Repository State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-06-12T10:23:43Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "event": "referenceUpdated",
    "repositoryName": "MyDemoRepo",
    "repositoryId": "12345678-1234-5678-abcd-12345678abcd",
    "referenceType": "branch",
    "referenceName": "myBranch",
    "referenceFullName": "refs/heads/myBranch",
    "commitId": "7f0103fMERGE",
    "oldCommitId": "3e5983DESTINATION",
    "baseCommitId": "3e5a9bf1BASE",
    "sourceCommitId": "26a8f2SOURCE",
    "destinationCommitId": "3e5983DESTINATION",
    "mergeOption": "THREE_WAY_MERGE",
    "conflictDetailsLevel": "LINE_LEVEL",
    "conflictResolutionStrategy": "AUTOMERGE"
  }
}
```

Peristiwa referenceDeleted

Dalam contoh peristiwa ini, sebuah cabang bernama myBranch telah dihapus dalam sebuah repositori bernama MyDemoRepo.

```
{
  "version": "0",
  "id": "01234567-EXAMPLE",
  "detail-type": "CodeCommit Repository State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-06-12T10:23:43Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "event": "referenceDeleted",
    "repositoryName": "MyDemoRepo",
    "repositoryId": "12345678-1234-5678-abcd-12345678abcd",
    "referenceType": "branch",
    "referenceName": "myBranch",
    "referenceFullName": "refs/heads/myBranch",
    "oldCommitId": "26a8f2EXAMPLE"
  }
}
```

unreferencedMergeCommitPeristiwa

Dalam contoh peristiwa ini, sebuah commit penggabungan yang tidak bernama telah dibuat dalam sebuah repositori bernama MyDemoRepo.

```
{
  "version": "0",
  "id": "01234567-EXAMPLE",
  "detail-type": "CodeCommit Repository State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-06-12T10:23:43Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "event": "unreferencedMergeCommitCreated",
    "repositoryName": "MyDemoRepo",
    "repositoryId": "12345678-1234-5678-abcd-12345678abcd",
  }
}
```

```

    "commitId": "7f0103fMERGE",
    "baseCommitId": "3e5a9bf1BASE",
    "sourceCommitId": "26a8f2SOURCE",
    "destinationCommitId": "3e5983DESTINATION",
    "mergeOption": "SQUASH_MERGE",
    "conflictDetailsLevel": "LINE_LEVEL",
    "conflictResolutionStrategy": "AUTOMERGE"
  }
}

```

commentOnCommitPeristiwa

Dalam contoh peristiwa ini, pengguna gabungan bernama Mary_Major mengomentari suatu commit. Dalam contoh ini, penyedia identitas gabungannya mengkonfigurasi tanda sesi untuk `displayName` dan `emailAddress`. Informasi tersebut termasuk dalam peristiwa tersebut.

```

{
  "version": "0",
  "id": "e9dce2e9-EXAMPLE",
  "detail-type": "CodeCommit Comment on Commit",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-09-29T20:20:39Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "beforeCommitId": "3c5dEXAMPLE",
    "repositoryId": "7dd1EXAMPLE...",
    "inReplyTo": "695bEXAMPLE...",
    "notificationBody": "A comment event occurred in the following repository: MyDemoRepo. The display name for the user is Mary Major. The email address for the user is mary_major@example.com. The user arn:aws:sts::123456789012:federated-user/Mary_Major made a comment. The comment was made on the following comment ID: 463bEXAMPLE.... For more information, go to the AWS CodeCommit console at https://us-east-2.console.aws.amazon.com/codecommit/home?region=us-east-2#/repository/MyDemoRepo/compare/3c5dEXAMPLE...f4d5EXAMPLE#463bEXAMPLE....",
    "commentId": "463bEXAMPLE...",
    "afterCommitId": "f4d5EXAMPLE",
    "event": "commentOnCommitCreated",
    "repositoryName": "MyDemoRepo",
    "callerUserArn": "arn:aws:sts::123456789012:federated-user/Mary_Major",

```

```
"displayName": "Mary Major",
"emailAddress": "mary_major@example.com"
}
}
```

commentOnCommitPeristiwa

Dalam contoh peristiwa ini, pengguna yang mengambil peran bernama Admin dengan nama sesi `Mary_Major` mengedit komentar pada suatu commit. Dalam contoh ini, peran termasuk tanda sesi yang dikonfigurasi untuk `displayName` dan `emailAddress`. Informasi tersebut termasuk dalam peristiwa tersebut.

```
{
  "version": "0",
  "id": "98377d67-EXAMPLE",
  "detail-type": "CodeCommit Comment on Commit",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-02-09T07:15:16Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "afterCommitId": "53812581",
    "beforeCommitId": "03314446",
    "callerUserArn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
    "commentId": "a7e5471e-EXAMPLE",
    "event": "commentOnCommitUpdated",
    "inReplyTo": "bdb07d47-EXAMPLE",
    "notificationBody": "A comment event occurred in the following AWS
CodeCommit repository: MyDemoRepo. The display name for the user is Mary
Major. The email address for the user is mary_major@example.com. The user
arn:aws:sts::123456789012:federated-user/Mary_Major updated a comment or
replied to a comment. The comment was made on the following comment ID:
bdb07d47-6fe9-47b0-a839-b93cc743b2ac:468cd1cb-2dfb-4f68-9636-8de52431d1d6.
For more information, go to the AWS CodeCommit console https://us-
east-2.console.aws.amazon.com/codesuite/codecommit/repositories/MyDemoRepo/
compare/0331444646178429589969823096709582251768/.../5381258150293783361471680277136017291382?
region\u003dus-east-2",
    "repositoryId": "12345678-1234-1234-1234-123456789012",
    "repositoryName": "MyDemoRepo",
    "displayName": "Mary Major",
```

```

    "emailAddress": "mary_major@example.com"
  }
}

```

commentOnPullRequestCreated peristiwa

Dalam contoh ini peristiwa, pengguna gabungan bernama Saanvi_Sarkar mengomentari permintaan pull. Dalam contoh ini, penyedia identitas gabungannya mengkonfigurasi tanda sesi untuk `displayName` dan `emailAddress`. Informasi tersebut termasuk dalam peristiwa tersebut.

```

{
  "version": "0",
  "id": "98377d67-EXAMPLE",
  "detail-type": "CodeCommit Comment on Pull Request",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-02-09T07:15:16Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "beforeCommitId": "3c5dEXAMPLE",
    "repositoryId": "7dd1EXAMPLE...",
    "inReplyTo": "695bEXAMPLE...",
    "notificationBody": "A comment event occurred in the following AWS
CodeCommit repository: MyDemoRepo. The display name for the user is Saanvi
Sarkar. The email address for the user is saanvi_sarkar@example.com. The user
arn:aws:sts::123456789012:federated-user/Saanvi_Sarkar made a comment. The comment
was made on the following Pull Request: 201. For more information, go to the AWS
CodeCommit console https://us-east-2.console.aws.amazon.com/codecommit/home?region=us-
east-2#/repository/MyDemoRepo/pull-request/201/activity#3276EXAMPLE...",
    "commentId": "463bEXAMPLE...",
    "afterCommitId": "f4d5EXAMPLE",
    "event": "commentOnPullRequestCreated",
    "repositoryName": "MyDemoRepo",
    "callerUserArn": "arn:aws:sts::123456789012:federated-user/Saanvi_Sarkar",
    "pullRequestId": "201",
    "displayName": "Saanvi Sarkar",
    "emailAddress": "saanvi_sarkar@example.com"
  }
}

```

commentOnPullRequestUpdated peristiwa

Dalam contoh peristiwa ini, pengguna gabungan bernama Saanvi_Sarkar mengedit komentar pada suatu permintaan pull. Dalam contoh ini, penyedia identitas gabungannya mengkonfigurasi tanda sesi untuk `displayName` dan `emailAddress`. Informasi tersebut termasuk dalam peristiwa tersebut.

```
{
  "version": "0",
  "id": "98377d67-EXAMPLE",
  "detail-type": "CodeCommit Comment on Pull Request",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-02-09T07:15:16Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "afterCommitId": "96814774EXAMPLE",
    "beforeCommitId": "6031971EXAMPLE",
    "callerUserArn": "arn:aws:sts::123456789012:federated-user/Saanvi_Sarkar",
    "commentId": "40cb52f0-EXAMPLE",
    "event": "commentOnPullRequestUpdated",
    "inReplyTo": "1285e713-EXAMPLE",
    "notificationBody": "A comment event occurred in the following AWS
CodeCommit repository: MyDemoRepo. The display name for the user is Saanvi
Sarkar. The email address for the user is saanvi_sarkar@example.com. The user
arn:aws:sts::123456789012:federated-user/Saanvi_Sarkar updated a comment or
replied to a comment. The comment was made on the following Pull Request:
1. For more information, go to the AWS CodeCommit console https://us-
east-2.console.aws.amazon.com/codesuite/codecommit/repositories/MyDemoRepo/pull-
requests/1/activity#40cb52f0-aac7-4c43-b771-601eff02EXAMPLE",
    "pullRequestId": "1",
    "repositoryId": "12345678-1234-1234-1234-123456789012",
    "repositoryName": "MyDemoRepo"
  }
}
```

pullRequestCreated peristiwa

Dalam contoh peristiwa ini, pull request dibuat dalam repositori bernama MyDemoRepo oleh pengguna yang mengambil peran bernama Admin dengan nama sesi Mary_Major. Tidak ada informasi tanda sesi yang disediakan, sehingga informasi tidak termasuk dalam peristiwa.

```
{
  "version": "0",
  "id": "98377d67-EXAMPLE",
  "detail-type": "CodeCommit Pull Request State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-02-09T07:15:16Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "author": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
    "callerUserArn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
    "creationDate": "Tue Feb 9 2019 10:18:42 PDT ",
    "description": "An example description.",
    "destinationCommit": "12241970EXAMPLE",
    "destinationReference": "refs/heads/main",
    "event": "pullRequestCreated",
    "isMerged": "False",
    "lastModifiedDate": "Tue Feb 9 2019 10:18:42 PDT",
    "notificationBody": "A pull request event occurred in the following AWS CodeCommit repository: MyDemoRepo. User: arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major. Event: Created. The pull request was created with the following information: Pull Request ID as 1 and title as My Example Pull Request. For more information, go to the AWS CodeCommit console https://us-east-2.console.aws.amazon.com/codesuite/codecommit/repositories/MyDemoRepo/pull-requests/1",
    "pullRequestId": "1",
    "pullRequestStatus": "Open",
    "repositoryNames": ["MyDemoRepo"],
    "revisionId": "bdc0cb9bEXAMPLE",
    "sourceCommit": "2774290EXAMPLE",
    "sourceReference": "refs/heads/test-branch",
    "title": "My Example Pull Request"
  }
}
```

pullRequestSourceBranchUpdated peristiwa

Dalam contoh peristiwa ini, pengguna yang mengambil peran bernama Admin dengan nama sesi Mary_Major memperbarui cabang sumber bernama test-branch untuk permintaan pull dengan ID 1.

```
{
  "version": "0",
  "id": "98377d67-EXAMPLE",
  "detail-type": "CodeCommit Pull Request State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-02-09T07:15:16Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "author": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
    "callerUserArn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
    "creationDate": "Tue Feb 9 2019 10:18:42 PDT",
    "description": "An example description.",
    "destinationCommit": "7644990EXAMPLE",
    "destinationReference": "refs/heads/main",
    "event": "pullRequestSourceBranchUpdated",
    "isMerged": "False",
    "lastModifiedDate": "Tue Feb 9 2019 10:18:42 PDT",
    "notificationBody": "A pull request event occurred in the following AWS
CodeCommit repository: MyDemoRepo. User: arn:aws:sts::123456789012:assumed-role/
Admin/Mary_Major. Event: Updated. The user updated the following pull request:
1. The pull request was updated with one or more commits to the source branch:
test-branch. For more information, go to the AWS CodeCommit console https://us-
east-2.console.aws.amazon.com/codesuite/codecommit/repositories/MyDemoRepo/pull-
requests/1?region\u003dus-east-2",
    "pullRequestId": "1",
    "pullRequestStatus": "Open",
    "repositoryNames": ["MyDemoRepo"],
    "revisionId": "bdc0cb9b4EXAMPLE",
    "sourceCommit": "64875001EXAMPLE",
    "sourceReference": "refs/heads/test-branch",
    "title": "My Example Pull Request"
  }
}
```



```
}
```

pullRequestStatusBerubah acara

Dalam contoh peristiwa ini, pengguna yang mengambil peran bernama Admin dengan nama sesi Mary_Major menutup permintaan pull dengan ID 1. Permintaan pull tidak digabung.

```
{
  "version": "0",
  "id": "98377d67-EXAMPLE",
  "detail-type": "CodeCommit Pull Request State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-02-09T07:15:16Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "author": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
    "callerUserArn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
    "creationDate": "Tue Jun 18 10:34:20 PDT 2019",
    "description": "An example description.",
    "destinationCommit": "95149731EXAMPLE",
    "destinationReference": "refs/heads/main",
    "event": "pullRequestStatusChanged",
    "isMerged": "False",
    "lastModifiedDate": "Tue Jun 18 10:34:20 PDT 2019",
    "notificationBody": "A pull request event occurred in the following AWS CodeCommit repository: MyDemoRepo. arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major updated the following PullRequest 1. The pull request status has been updated. The status is closed. For more information, go to the AWS CodeCommit console https://us-east-2.console.aws.amazon.com/codesuite/codecommit/repositories/MyDemoRepo/pull-requests/1?region\u003dus-east-2",
    "pullRequestId": "1",
    "pullRequestStatus": "Closed",
    "repositoryNames": ["MyDemoRepo"],
    "revisionId": "bdc0cb9bEXAMPLE",
    "sourceCommit": "4409936EXAMPLE",
    "sourceReference": "refs/heads/test-branch",
    "title": "My Example Pull Request"
  }
}
```

```
}
```

pullRequestMergeStatusUpdated peristiwa

Dalam contoh peristiwa ini, pengguna yang mengambil peran bernama Admin dengan nama sesi Mary_Major menggabungkan permintaan pull dengan ID 1.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-012345678901",
  "detail-type": "CodeCommit Pull Request State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-06-12T10:23:43Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "author": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
    "callerUserArn": "arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major",
    "creationDate": "Mon Mar 11 14:42:31 PDT 2019",
    "description": "An example description.",
    "destinationCommit": "4376719EXAMPLE",
    "destinationReference": "refs/heads/main",
    "event": "pullRequestMergeStatusUpdated",
    "isMerged": "True",
    "lastModifiedDate": "Mon Mar 11 14:42:31 PDT 2019",
    "mergeOption": "FAST_FORWARD_MERGE",
    "notificationBody": "A pull request event occurred in the following AWS CodeCommit repository: MyDemoRepo. arn:aws:sts::123456789012:assumed-role/Admin/Mary_Major updated the following PullRequest 1. The pull request merge status has been updated. The status is merged. For more information, go to the AWS CodeCommit console https://us-east-2.console.aws.amazon.com/codesuite/codecommit/repositories/MyDemoRepo/pull-requests/1?region\u003dus-east-2",
    "pullRequestId": "1",
    "pullRequestStatus": "Closed",
    "repositoryNames": ["MyDemoRepo"],
    "revisionId": "bdc0cb9beEXAMPLE",
    "sourceCommit": "0701696EXAMPLE",
    "sourceReference": "refs/heads/test-branch",
    "title": "My Example Pull Request"
  }
}
```

```
}
```

approvalRuleTemplatePeristiwa

Dalam contoh peristiwa ini, pengguna dengan nama pengguna IAM `Mary_Major` membuat sebuah templat aturan persetujuan bernama `2-approvers-required-for-main`.

```
{
  "version": "0",
  "id": "f7702227-EXAMPLE",
  "detail-type": "CodeCommit Approval Rule Template Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-11-06T19:02:27Z",
  "region": "us-east-2",
  "resources": [],
  "detail": {
    "approvalRuleTemplateContentSha256": "f742eebbEXAMPLE",
    "approvalRuleTemplateId": "d7385967-EXAMPLE",
    "approvalRuleTemplateName": "2-approvers-required-for-main",
    "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
    "creationDate": "Wed Nov 06 19:02:14 UTC 2019",
    "event": "approvalRuleTemplateCreated",
    "lastModifiedDate": "Wed Nov 06 19:02:14 UTC 2019",
    "notificationBody": "A approval rule template event occurred in the following
AWS CodeCommit account: 123456789012. User: arn:aws:iam::123456789012:user/Mary_Major.
Additional information: An approval rule template with the following name has been
created: 2-approvers-required-for-main. The ID of the created template is: d7385967-
EXAMPLE. For more information, go to the AWS CodeCommit console.",
    "repositories": {}
  }
}
```

approvalRuleTemplatePeristiwa

Dalam contoh peristiwa ini, pengguna dengan nama pengguna IAM `Mary_Major` mengedit sebuah templat aturan persetujuan bernama `2-approvers-required-for-main`. Templat aturan persetujuan tidak terkait dengan repositori mana pun.

```
{
  "version": "0",
  "id": "66403118-EXAMPLE",
```

```

"detail-type": "CodeCommit Approval Rule Template Change",
"source": "aws.codecommit",
"account": "123456789012",
"time": "2019-11-12T23:03:30Z",
"region": "us-east-2",
"resources": [

],
"detail": {
  "approvalRuleTemplateContentSha256": "f742eebbEXAMPLE",
  "approvalRuleTemplateId": "c9d2b844-EXAMPLE",
  "approvalRuleTemplateName": "2-approvers-required-for-main",
  "callerUserArn": "arn:aws:iam::123456789012:user\Mary_Major",
  "creationDate": "Tue Nov 12 23:03:06 UTC 2019",
  "event": "approvalRuleTemplateDeleted",
  "lastModifiedDate": "Tue Nov 12 23:03:20 UTC 2019",
  "notificationBody": "A approval rule template event occurred in the following AWS
CodeCommit account: 123456789012. User: arn:aws:iam::123456789012:user\Mary_Major.
Additional information: An approval rule template with the following name has been
deleted: 2-approvers-required-for-main. The ID of the updated template is: c9d2b844-
EXAMPLE. For more information, go to the AWS CodeCommit console.",
  "repositories": {}
}
}

```

approvalRuleTemplatePeristiwa

Dalam contoh peristiwa ini, pengguna dengan nama pengguna IAM `Mary_Major` menghapus sebuah templat aturan persetujuan bernama `2-approvers-required-for-main`. Templat aturan persetujuan tidak terkait dengan repositori mana pun.

```

{
  "version": "0",
  "id": "66403118-EXAMPLE",
  "detail-type": "CodeCommit Approval Rule Template Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-11-12T23:03:30Z",
  "region": "us-east-2",
  "resources": [],
  "detail": {
    "approvalRuleTemplateContentSha256": "4f3de6632EXAMPLE",
    "approvalRuleTemplateId": "c9d2b844-EXAMPLE",

```

```

    "approvalRuleTemplateName": "2-approvers-required-for-main",
    "callerUserArn": "arn:aws:iam::123456789012:user\Mary_Major",
    "creationDate": "Tue Nov 12 23:03:06 UTC 2019",
    "event": "approvalRuleTemplateUpdated",
    "lastModifiedDate": "Tue Nov 12 23:03:20 UTC 2019",
    "notificationBody": "A approval rule template event occurred in the following AWS
CodeCommit account: 123456789012. User: arn:aws:iam::123456789012:user\Mary_Major.
Additional information: An approval rule template with the following name has
been updated: 2-approvers-required-for-main. The ID of the updated template is:
c9d2b844-EXAMPLE. The after rule template content SHA256 is 4f3de663EXAMPLE. For more
information, go to the AWS CodeCommit console.",
    "repositories": {}
  }
}

```

approvalRuleTemplateAssociatedWithRepository peristiwa

Dalam contoh peristiwa ini, pengguna dengan nama pengguna IAM `Mary_Major` mengaitkan sebuah templat aturan persetujuan bernama `2-approvers-required-for-main` dengan repositori bernama `MyDemoRepo`.

```

{
  "version": "0",
  "id": "ea1c6d73-EXAMPLE",
  "detail-type": "CodeCommit Approval Rule Template Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-11-06T19:02:27Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "approvalRuleTemplateContentSha256": "f742eebbEXAMPLE",
    "approvalRuleTemplateId": "d7385967-EXAMPLE",
    "approvalRuleTemplateName": "2-approvers-required-for-main",
    "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
    "creationDate": "Wed Nov 06 19:02:14 UTC 2019",
    "event": "approvalRuleTemplateAssociatedWithRepository",
    "lastModifiedDate": "Wed Nov 06 19:02:14 UTC 2019",
    "notificationBody": "A approval rule template event occurred in the following
AWS CodeCommit account: 123456789012. User: arn:aws:iam::123456789012:user/Mary_Major.
Additional information: An approval rule template has been associated with the

```

```

following repository: [MyDemoRepo]. For more information, go to the AWS CodeCommit
console.",
  "repositories": {
    "MyDemoRepo": "92ca7bf2-d878-49ed-a994-336a6cc7c574"
  }
}
}

```

approvalRuleTemplateDisassociatedWithRepository peristiwa

Dalam contoh peristiwa ini, pengguna dengan nama pengguna IAM Mary_Major melepaskan sebuah templat aturan persetujuan bernama 2-approvers-required-for-main dari repositori bernama MyDemoRepo.

```

{
  "version": "0",
  "id": "ea1c6d73-EXAMPLE",
  "detail-type": "CodeCommit Approval Rule Template Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-11-06T19:02:27Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "approvalRuleTemplateContentSha256": "f742eebbEXAMPLE",
    "approvalRuleTemplateId": "d7385967-EXAMPLE",
    "approvalRuleTemplateName": "2-approvers-required-for-main",
    "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
    "creationDate": "Wed Nov 06 19:02:14 UTC 2019",
    "event": "approvalRuleTemplateDisassociatedFromRepository",
    "lastModifiedDate": "Wed Nov 06 19:02:14 UTC 2019",
    "notificationBody": "A approval rule template event occurred in the following
AWS CodeCommit account: 123456789012. User: arn:aws:iam::123456789012:user/Mary_Major.
Additional information: An approval rule template has been disassociated from the
following repository: [MyDemoRepo]. For more information, go to the AWS CodeCommit
console.",
    "repositories": {
      "MyDemoRepo": "92ca7bf2-d878-49ed-a994-336a6cc7c574"
    }
  }
}

```

```
}
```

approvalRuleTemplateBatchAssociatedWithRepositories peristiwa

Dalam contoh peristiwa ini, pengguna dengan nama pengguna IAM batch `Mary_Major` mengaitkan sebuah templat aturan persetujuan bernama `2-approvers-required-for-main` dengan repositori bernama `MyDemoRepo` dan repositori bernama `MyTestRepo`.

```
{
  "version": "0",
  "id": "0f861e5b-EXAMPLE",
  "detail-type": "CodeCommit Approval Rule Template Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-11-12T23:39:09Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "approvalRuleTemplateContentSha256": "f742eebbEXAMPLE",
    "approvalRuleTemplateId": "c71c1fe0-EXAMPLE",
    "approvalRuleTemplateName": "2-approvers-required-for-main",
    "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
    "creationDate": "Tue Nov 12 23:38:57 UTC 2019",
    "event": "batchAssociateApprovalRuleTemplateWithRepositories",
    "lastModifiedDate": "Tue Nov 12 23:38:57 UTC 2019",
    "notificationBody": "A approval rule template event occurred in the following
AWS CodeCommit account: 123456789012. User: arn:aws:iam::123456789012:user\Mary_Major.
Additional information: An approval rule template has been batch associated with the
following repository names: [MyDemoRepo, MyTestRepo]. For more information, go to the
AWS CodeCommit console.",
    "repositories": {
      "MyDemoRepo": "MyTestRepo"
    }
  }
}
```

approvalRuleTemplateBatchDisassociatedFromRepositories peristiwa

Dalam contoh peristiwa ini, pengguna dengan nama pengguna IAM batch `Mary_Major` melepaskan sebuah templat aturan persetujuan bernama `2-approvers-required-for-main` dari repositori bernama `MyDemoRepo` dan repositori bernama `MyTestRepo`.

```
{
  "version": "0",
  "id": "e08fc996-EXAMPLE",
  "detail-type": "CodeCommit Approval Rule Template Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-11-12T23:39:09Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "approvalRuleTemplateContentSha256": "f742eebbEXAMPLE",
    "approvalRuleTemplateId": "c71c1fe0-ff91-4db4-9a45-a86a7b6c474f",
    "approvalRuleTemplateName": "2-approvers-required-for-main",
    "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
    "creationDate": "Tue Nov 12 23:38:57 UTC 2019",
    "event": "batchDisassociateApprovalRuleTemplateFromRepositories",
    "lastModifiedDate": "Tue Nov 12 23:38:57 UTC 2019",
    "notificationBody": "A approval rule template event occurred in the following
AWS CodeCommit account: 123456789012. User: arn:aws:iam::123456789012:user/Mary_Major.
Additional information: An approval rule template has been batch disassociated from
the following repository names: [MyDemoRepo, MyTestRepo]. For more information, go to
the AWS CodeCommit console.",
    "repositories": {
      "MyDemoRepo": "MyTestRepo"
    }
  }
}
```

pullRequestApprovalRuleCreated peristiwa

Dalam contoh peristiwa ini, pengguna dengan nama pengguna IAM `Mary_Major` membuat sebuah templat aturan persetujuan bernama `1-approver-needed` untuk permintaan pull dengan ID 227.

```
{
```



```

"version": "0",
"id": "ad860f12-EXAMPLE",
"detail-type": "CodeCommit Pull Request State Change",
"source": "aws.codecommit",
"account": "123456789012",
"time": "2019-11-06T19:12:19Z",
"region": "us-east-2",
"resources": [
  "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
],
"detail": {
  "approvalRuleContentSha256": "f742eebbEXAMPLE",
  "approvalRuleId": "0a9b5dfc-EXAMPLE",
  "approvalRuleName": "1-approver-needed",
  "author": "arn:aws:iam::123456789012:user/Mary_Major",
  "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
  "creationDate": "Wed Nov 06 19:10:58 UTC 2019",
  "description": "An An example description.",
  "destinationCommit": "194fdf00EXAMPLE",
  "destinationReference": "refs/heads/main",
  "event": "pullRequestApprovalRuleCreated",
  "isMerged": "False",
  "lastModifiedDate": "Wed Nov 06 19:10:58 UTC 2019",
  "notificationBody": "A pull request event occurred in the following AWS
CodeCommit repository: MyDemoRepo. User: arn:aws:iam::123456789012:user/Mary_Major.
Event: Updated. Pull request: 227. Additional information: An approval rule has been
created with the following name: 1-approver-needed. For more information, go to the
AWS CodeCommit console https://us-east-2.console.aws.amazon.com/codesuite/codecommit/
repositories/MyDemoRepo/pull-requests/227?region=us-east-2",
  "pullRequestId": "227",
  "pullRequestStatus": "Open",
  "repositoryNames": [
    "MyDemoRepo"
  ],
  "revisionId": "3b8cecab3EXAMPLE",
  "sourceCommit": "29964a17EXAMPLE",
  "sourceReference": "refs/heads/test-branch",
  "title": "My example pull request"
}
}

```

pullRequestApprovalRuleDeleted peristiwa

Dalam contoh peristiwa ini, pengguna dengan nama pengguna IAM Mary_Major menghapus sebuah templat aturan persetujuan bernama 1-approver-needed untuk permintaan pull dengan ID 227. Pengguna IAM dengan nama Saanvi_Sarkar awalnya menulis aturan persetujuan.

```
{
  "version": "0",
  "id": "c1c3509d-EXAMPLE",
  "detail-type": "CodeCommit Pull Request State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-11-06T19:12:19Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "approvalRuleContentSha256": "f742eebbEXAMPLE",
    "approvalRuleId": "0a9b5dfc-EXAMPLE",
    "approvalRuleName": "1-approver-needed",
    "author": "arn:aws:iam::123456789012:user/Saanvi_Sarkar",
    "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
    "creationDate": "Wed Nov 06 19:10:58 UTC 2019",
    "description": "An An example description.",
    "destinationCommit": "194fdf00EXAMPLE",
    "destinationReference": "refs/heads/main",
    "event": "pullRequestApprovalRuleDeleted",
    "isMerged": "False",
    "lastModifiedDate": "Wed Nov 06 19:10:58 UTC 2019",
    "notificationBody": "A pull request event occurred in the following AWS
CodeCommit repository: MyDemoRepo. User: arn:aws:iam::123456789012:user/Mary_Major.
Event: Created. Pull request: 227. Additional information: An approval rule has been
deleted: 1-approver-needed was deleted. For more information, go to the AWS CodeCommit
console https://us-east-2.console.aws.amazon.com/codesuite/codecommit/repositories/
MyDemoRepo/pull-requests/227?region=us-east-2",
    "pullRequestId": "227",
    "pullRequestStatus": "Open",
    "repositoryNames": [
      "MyDemoRepo"
    ],
    "revisionId": "3b8cecabEXAMPLE",
    "sourceCommit": "29964a17EXAMPLE",
```

```

    "sourceReference": "refs/heads/test-branch",
    "title": "My example pull request"
  }
}

```

pullRequestApprovalRuleOverridden peristiwa

Dalam contoh peristiwa ini, persyaratan aturan persetujuan untuk permintaan pull telah disisihkan (OVERRIDE) oleh pengguna dengan nama pengguna IAM dari `Mary_Major`. Permintaan pull ditulis oleh pengguna dengan nama pengguna IAM dari `Li_Juan`.

```

{
  "version": "0",
  "id": "52d2cb73-EXAMPLE",
  "detail-type": "CodeCommit Pull Request State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-11-06T19:12:19Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "author": "arn:aws:iam::123456789012:user/Li_Juan",
    "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
    "creationDate": "Wed Nov 06 19:10:58 UTC 2019",
    "description": "An An example description.",
    "destinationCommit": "194fdf00EXAMPLE",
    "destinationReference": "refs/heads/main",
    "event": "pullRequestApprovalRuleOverridden",
    "isMerged": "False",
    "lastModifiedDate": "Wed Nov 06 19:10:58 UTC 2019",
    "notificationBody": "A pull request event occurred in the following AWS
CodeCommit repository: MyDemoRepo. User: arn:aws:iam::123456789012:user/Mary_Major.
Event: Updated. Pull request name: 227. Additional information: An override
event has occurred for the approval rules for this pull request. Override status:
OVERRIDE. For more information, go to the AWS CodeCommit console https://us-
east-2.console.aws.amazon.com/codesuite/codecommit/repositories/MyDemoRepo/pull-
requests/227?region=us-east-2",
    "overrideStatus": "OVERRIDE",
    "pullRequestId": "227",
    "pullRequestStatus": "Open",
    "repositoryNames": [

```

```

        "MyDemoRepo"
    ],
    "revisionId": "3b8cecabEXAMPLE",
    "sourceCommit": "29964a17EXAMPLE",
    "sourceReference": "refs/heads/test-branch",
    "title": "My example pull request"
}
}

```

Dalam peristiwa contoh ini, persyaratan aturan persetujuan untuk pull request telah dipulihkan (REVOKE).

```

{
  "version": "0",
  "id": "2895482d-13eb-b783-270d-76588e6029fa",
  "detail-type": "CodeCommit Pull Request State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-11-06T19:12:19Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "author": "arn:aws:iam::123456789012:user/Li_Juan",
    "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
    "creationDate": "Wed Nov 06 19:10:58 UTC 2019",
    "description": "An An example description.",
    "destinationCommit": "194fdf00EXAMPLE",
    "destinationReference": "refs/heads/main",
    "event": "pullRequestApprovalRuleOverridden",
    "isMerged": "False",
    "lastModifiedDate": "Wed Nov 06 19:10:58 UTC 2019",
    "notificationBody": "A pull request event occurred in the following
AWS CodeCommit repository: MyDemoRepo. User: arn:aws:iam::123456789012:user/
Mary_Major. Event: Updated. Pull request name: 227. Additional information: An
override event has occurred for the approval rules for this pull request. Override
status: REVOKE. For more information, go to the AWS CodeCommit console https://
us-east-2.console.aws.amazon.com/codesuite/codecommit/repositories/MyDemoRepo/pull-
requests/227?region=us-east-2",
    "overrideStatus": "REVOKE",
    "pullRequestId": "227",
    "pullRequestStatus": "Open",

```

```
    "repositoryNames": [
      "MyDemoRepo"
    ],
    "revisionId": "3b8cecabEXAMPLE",
    "sourceCommit": "29964a17EXAMPLE",
    "sourceReference": "refs/heads/test-branch",
    "title": "My example pull request"
  }
}
```

pullRequestApprovalStateChanged peristiwa

Dalam contoh peristiwa ini, permintaan pull telah disetujui oleh pengguna dengan nama pengguna IAM dari Mary_Major.

```
{
  "version": "0",
  "id": "53e5d7e9-986c-1ebf-9d8b-ebef5596da0e",
  "detail-type": "CodeCommit Pull Request State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-11-06T19:12:19Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "approvalStatus": "APPROVE",
    "author": "arn:aws:iam::123456789012:user/Li_Juan",
    "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
    "creationDate": "Wed Nov 06 19:10:58 UTC 2019",
    "description": "An An example description.",
    "destinationCommit": "194fdf00EXAMPLE",
    "destinationReference": "refs/heads/main",
    "event": "pullRequestApprovalStateChanged",
    "isMerged": "False",
    "lastModifiedDate": "Wed Nov 06 19:10:58 UTC 2019",
    "notificationBody": "A pull request event occurred in the following
AWS CodeCommit repository: MyDemoRepo. User: arn:aws:iam::123456789012:user/
Mary_Major. Event: Updated. Pull request name: 227. Additional information:
A user has changed their approval state for the pull request. State change:
APPROVE. For more information, go to the AWS CodeCommit console https://us-
```

```

east-2.console.aws.amazon.com/codesuite/codecommit/repositories/MyDemoRepo/pull-
requests/227?region=us-east-2",
  "pullRequestId": "227",
  "pullRequestStatus": "Open",
  "repositoryNames": [
    "MyDemoRepo"
  ],
  "revisionId": "3b8cecabEXAMPLE",
  "sourceCommit": "29964a17EXAMPLE",
  "sourceReference": "refs/heads/test-branch",
  "title": "My example pull request"
}
}

```

Dalam contoh peristiwa ini, persetujuan untuk permintaan pull telah dicabut oleh pengguna dengan nama pengguna IAM dari Mary_Major.

```

{
  "version": "0",
  "id": "25e183d7-d01a-4e07-2bd9-b2d56bebcc81",
  "detail-type": "CodeCommit Pull Request State Change",
  "source": "aws.codecommit",
  "account": "123456789012",
  "time": "2019-11-06T19:12:19Z",
  "region": "us-east-2",
  "resources": [
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail": {
    "approvalStatus": "REVOKE",
    "author": "arn:aws:iam::123456789012:user/Li_Juan",
    "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
    "creationDate": "Wed Nov 06 19:10:58 UTC 2019",
    "description": "An An example description.",
    "destinationCommit": "194fdf00EXAMPLE",
    "destinationReference": "refs/heads/main",
    "event": "pullRequestApprovalStateChanged",
    "isMerged": "False",
    "lastModifiedDate": "Wed Nov 06 19:10:58 UTC 2019",
    "notificationBody": "A pull request event occurred in the following AWS
CodeCommit repository: MyDemoRepo. User: arn:aws:iam::123456789012:user/Mary_Major.
Event: Updated. Pull request name: 227. Additional information: A user has changed
their approval state for the pull request. State change: REVOKE. For more information,

```

```

go to the AWS CodeCommit console https://us-east-2.console.aws.amazon.com/codesuite/codecommit/repositories/MyDemoRepo/pull-requests/227?region=us-east-2",
    "pullRequestId": "227",
    "pullRequestStatus": "Open",
    "repositoryNames": [
        "MyDemoRepo"
    ],
    "revisionId": "3b8cecabEXAMPLE",
    "sourceCommit": "29964a17EXAMPLE",
    "sourceReference": "refs/heads/test-branch",
    "title": "My example pull request"
}
}

```

pullRequestApprovalRuleUpdated peristiwa

Dalam peristiwa contoh ini, aturan persetujuan untuk permintaan pull telah diedit oleh pengguna dengan nama pengguna IAM dari `Mary_Major`. Dia juga pengguna yang menulis permintaan pull.

```

{
    "version": "0",
    "id": "21b1c819-2889-3528-1cb8-3861aacf9d42",
    "detail-type": "CodeCommit Pull Request State Change",
    "source": "aws.codecommit",
    "account": "123456789012",
    "time": "2019-11-06T19:12:19Z",
    "region": "us-east-2",
    "resources": [
        "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
    ],
    "detail": {
        "approvalRuleContentSha256": "f742eebbEXAMPLE",
        "approvalRuleId": "0a9b5dfc-EXAMPLE",
        "approvalRuleName": "1-approver-needed",
        "author": "arn:aws:iam::123456789012:user/Mary_Major",
        "callerUserArn": "arn:aws:iam::123456789012:user/Mary_Major",
        "creationDate": "Wed Nov 06 19:10:58 UTC 2019",
        "description": "An example description.",
        "destinationCommit": "194fdf00EXAMPLE",
        "destinationReference": "refs/heads/main",
        "event": "pullRequestApprovalRuleUpdated",
        "isMerged": "False",
        "lastModifiedDate": "Wed Nov 06 19:10:58 UTC 2019",
    }
}

```

```

    "notificationBody": "A pull request event occurred in the following
AWS CodeCommit repository: MyDemoRepo. User: arn:aws:iam::123456789012:user/
Mary_Major. Event: Updated. Pull request name: 227. The content of an approval
rule has been updated for the pull request. The name of the updated rule is: 1-
approver-needed. For more information, go to the AWS CodeCommit console https://
us-east-2.console.aws.amazon.com/codesuite/codecommit/repositories/MyDemoRepo/pull-
requests/227?region=us-east-2",
    "pullRequestId": "227",
    "pullRequestStatus": "Open",
    "repositoryNames": [
        "MyDemoRepo"
    ],
    "revisionId": "3b8cecab3EXAMPLE",
    "sourceCommit": "29964a17EXAMPLE",
    "sourceReference": "refs/heads/test-branch",
    "title": "My example pull request"
}
}

```

Peristiwa reactionCreated

Dalam contoh peristiwa ini, reaksi terhadap komentar telah ditambahkan oleh pengguna dengan nama pengguna IAM dari Mary_Major.

```

{
  "version":"0",
  "id":"59fccccd8-217a-32ce-2b05-561ed68a1c42",
  "detail-type":"CodeCommit Comment Reaction Change",
  "source":"aws.codecommit",
  "account":"123456789012",
  "time":"2020-04-14T00:49:03Z",
  "region":"us-east-2",
  "resources":[
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail":{
    "callerUserArn":"arn:aws:iam::123456789012:user/Mary_Major",
    "commentId":"28930161-EXAMPLE",
    "event":"commentReactionCreated",
    "notificationBody":"A comment reaction event occurred in the following AWS
CodeCommit Repository: MyDemoRepo. The user: arn:aws:iam::123456789012:user/Mary_Major
made a comment reaction # to the comment with comment ID: 28930161-EXAMPLE",
    "reactionEmojis":["#"],

```



```

    "reactionShortcodes":["thumbsdown:"],
    "reactionUnicode":["U+1F44E"],
    "repositoryId":"12345678-1234-5678-abcd-12345678abcd",
    "repositoryName":"MyDemoRepo"
  }
}

```

Peristiwa reactionUpdated

Dalam contoh peristiwa ini, reaksi terhadap komentar telah diperbarui oleh pengguna dengan nama pengguna IAM dari Mary_Major. Pengguna hanya dapat memperbarui reaksi mereka sendiri.

```

{
  "version":"0",
  "id":"0844ed99-a53f-3bdb-6048-4de315516889",
  "detail-type":"CodeCommit Comment Reaction Change",
  "source":"aws.codecommit",
  "account":"123456789012",
  "time":"2020-04-22T23:19:42Z",
  "region":"us-east-2",
  "resources":[
    "arn:aws:codecommit:us-east-2:123456789012:MyDemoRepo"
  ],
  "detail":{
    "callerUserArn":"arn:aws:iam::123456789012:user/Mary_Major",
    "commentId":"28930161-EXAMPLE",
    "event":"commentReactionUpdated",
    "notificationBody":"A comment reaction event occurred in the following AWS CodeCommit Repository: MyDemoRepo. The user: arn:aws:iam::123456789012:user/Mary_Major updated a reaction :smile: to the comment with comment ID: 28930161-EXAMPLE",
    "reactionEmojis":[
      "#"
    ],
    "reactionShortcodes":[
      ":smile:"
    ],
    "reactionUnicode":["U+1F604"]
  ],
  "repositoryId":"12345678-1234-5678-abcd-12345678abcd",
  "repositoryName":"MyDemoRepo"
}

```

```
}
```

Mencatat panggilan API AWS CodeCommit dengan AWS CloudTrail

CodeCommit terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di CodeCommit. CloudTrail menangkap semua panggilan API untuk CodeCommit peristiwa, termasuk panggilan dari CodeCommit konsol, klien Git Anda, dan panggilan kode ke CodeCommit API. Jika membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail peristiwa berkelanjutan ke bucket Amazon S3, termasuk peristiwa untuk CodeCommit. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol di Riwayat peristiwa. Menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat CodeCommit, alamat IP asal permintaan tersebut dibuat, siapa yang membuat permintaan, kapan permintaan dibuat, dan detail lainnya.

Untuk mempelajari lebih lanjut CloudTrail, lihat [Panduan AWS CloudTrail Pengguna](#).

CodeCommit informasi dalam CloudTrail

CloudTrail diaktifkan pada akun Amazon Web Services Anda pada saat membuat akun. Saat aktivitas terjadi di CodeCommit, aktivitas tersebut dicatat dalam CloudTrail peristiwa bersama peristiwa AWS layanan lainnya di Riwayat peristiwa. Anda dapat melihat, mencari, dan mengunduh kejadian terbaru di akun Amazon Web Services Anda. Untuk informasi selengkapnya, lihat [Melihat Kejadian dengan Riwayat CloudTrail Kejadian](#).

Untuk catatan peristiwa yang sedang berlangsung di akun Amazon Web Services Anda, termasuk peristiwa untuk CodeCommit, buatlah jejak. Jejak CloudTrail memungkinkan pengiriman file log ke bucket Amazon S3. Secara default, saat Anda membuat lintasan di konsol, lintasan tersebut berlaku untuk semua wilayah. Jejak mencatat kejadian dari semua wilayah dalam partisi AWS dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lainnya untuk menganalisis lebih lanjut dan bertindak berdasarkan data peristiwa yang dikumpulkan di CloudTrail log. Untuk informasi selengkapnya, lihat :

- [Ikhtisar untuk Membuat Jejak](#)
- [CloudTrail Layanan dan Integrasi yang Didukung](#)
- [Mengonfigurasi Notifikasi Amazon SNS untuk CloudTrail](#)

- [Menerima Berkas CloudTrail Log dari Beberapa Wilayah](#) dan [Menerima Berkas CloudTrail Log dari Beberapa Akun](#)

Saat CloudTrail pencatatan diaktifkan di akun Amazon Web Services Anda, panggilan API dibuat untuk CodeCommit tindakan dilacak dalam file CloudTrail log, di mana mereka ditulis dengan catatan AWS layanan lainnya. CloudTrail menentukan kapan membuat dan tulis ke file baru berdasarkan periode waktu dan ukuran file.

Semua CodeCommit tindakan dicatat oleh CloudTrail, termasuk beberapa (seperti `GetObjectIdentifier`) yang saat ini tidak didokumentasikan dalam [Referensi AWS CodeCommit API](#) tetapi sebaliknya direferensikan sebagai izin akses dan didokumentasikan di [Referensi izin CodeCommit](#). Misalnya, panggilan ke `ListRepositories` (dalam AWS CLI, `aws codecommit list-repositories`), `CreateRepository` (`aws codecommit create-repository`) dan `PutRepositoryTriggers` (`aws codecommit put-repository-triggers`) tindakan menghasilkan entri dalam file CloudTrail log, serta panggilan klien Git ke `GitPull` dan `GitPush`. Selain itu, jika Anda memiliki CodeCommit repositori yang dikonfigurasi sebagai sumber untuk pipeline CodePipeline, Anda akan melihat panggilan untuk CodeCommit mengakses tindakan izin seperti `UploadArchive` dari CodePipeline. Karena CodeCommit menggunakan AWS Key Management Service untuk mengenkripsi dan mendekripsi repositori, Anda juga akan melihat panggilan dari CodeCommit ke `Encrypt` dan `Decrypt` tindakan dari AWS KMS dalam CloudTrail log.

Setiap entri log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas pengguna dalam entri log membantu Anda menentukan hal berikut:

- Jika permintaan tersebut dibuat dengan kredensial pengguna root atau IAM
- Jika permintaan tersebut dibuat dengan kredensial keamanan sementara untuk peran atau pengguna gabungan, atau dibuat oleh peran yang dimiliki
- Jika permintaan tersebut dibuat oleh layanan AWS lainnya

Untuk informasi lain, lihat [Elemen userIdentity CloudTrail](#).

Anda dapat menyimpan berkas log dalam bucket Amazon S3 selama yang diinginkan, tetapi Anda juga dapat menentukan aturan siklus hidup Amazon S3 untuk mengarsipkan atau menghapus berkas log secara otomatis. Secara default, berkas log Anda dienkripsi dengan menggunakan enkripsi sisi server (SSE) Amazon S3.

Memahami entri berkas CodeCommit log

CloudTrail file log berisi satu atau beberapa entri log. Setiap entri berisi beberapa peristiwa yang diformat JSON. Sebuah log acara mewakili satu permintaan dari sumber apa pun dan mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, parameter permintaan, dan sebagainya. Entri log bukan jejak tumpukan yang dipesan dari panggilan API publik, sehingga tidak muncul dalam urutan tertentu.

Note

Contoh ini telah diformat untuk meningkatkan keterbacaan. Dalam file CloudTrail log, semua entri dan peristiwa digabungkan menjadi satu baris. Contoh ini juga terbatas pada CodeCommit entri tunggal. Dalam file CloudTrail log nyata, Anda melihat entri dan peristiwa dari beberapa AWS layanan.

Daftar Isi

- [Contoh: Entri log untuk daftar CodeCommit repositori](#)
- [Contoh: Entri log untuk membuat CodeCommit repositori](#)
- [Contoh: Entri log untuk Git menarik panggilan ke CodeCommit repositori](#)
- [Contoh: Entri log untuk keberhasilan pengiriman ke CodeCommit repositori](#)

Contoh: Entri log untuk daftar CodeCommit repositori

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `ListRepositories` tindakan.

Note

Meskipun `ListRepositories` mengembalikan daftar repositori, tanggapan yang tidak dapat dibisukan tidak dicatat dalam CloudTrail log, sehingga `responseElements` ditampilkan seperti `null` dalam file log.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
```

```
"principalId":"AIDACKCEVSQ6C2EXAMPLE",
"arn":"arn:aws:iam::444455556666:user/Mary_Major",
"accountId":"444455556666",
"accessKeyId":"AKIAIOSFODNN7EXAMPLE",
"userName":"Mary_Major"
},
"eventTime":"2016-12-14T17:57:36Z",
"eventSource":"codecommit.amazonaws.com",
"eventName":"ListRepositories",
"awsRegion":"us-east-1",
"sourceIPAddress":"203.0.113.12",
"userAgent":"aws-cli/1.10.53 Python/2.7.9 Windows/8 boto/1.4.43",
"requestParameters":null,
"responseElements":null,
"requestID":"cb8c167e-EXAMPLE",
"eventID":"e3c6f4ce-EXAMPLE",
"readOnly":true,
"eventType":"AwsApiCall",
"apiVersion":"2015-04-13",
"recipientAccountId":"444455556666"
}
```

Contoh: Entri log untuk membuat CodeCommit repositori

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `CreateRepository` tindakan di Wilayah US East (Ohio).

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::444455556666:user/Mary_Major",
    "accountId": "444455556666",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "Mary_Major"
  },
  "eventTime": "2016-12-14T18:19:15Z",
  "eventSource": "codecommit.amazonaws.com",
  "eventName": "CreateRepository",
  "awsRegion": "us-east-2",
  "sourceIPAddress": "203.0.113.12",
  "userAgent": "aws-cli/1.10.53 Python/2.7.9 Windows/8 boto/1.4.43",
```

```

"requestParameters": {
  "repositoryDescription": "Creating a demonstration repository.",
  "repositoryName": "MyDemoRepo"
},
"responseElements": {
  "repositoryMetadata": {
    "arn": "arn:aws:codecommit:us-east-2:111122223333:MyDemoRepo",
    "creationDate": "Dec 14, 2016 6:19:14 PM",
    "repositoryId": "8afe792d-EXAMPLE",
    "cloneUrlSsh": "ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/
MyDemoRepo",
    "repositoryName": "MyDemoRepo",
    "accountId": "111122223333",
    "cloneUrlHttp": "https://git-codecommit.us-east-2.amazonaws.com/v1/repos/
MyDemoRepo",
    "repositoryDescription": "Creating a demonstration repository.",
    "lastModifiedDate": "Dec 14, 2016 6:19:14 PM"
  }
},
"requestID": "d148de46-EXAMPLE",
"eventID": "740f179d-EXAMPLE",
"readOnly": false,
"resources": [
  {
    "ARN": "arn:aws:codecommit:us-east-2:111122223333:MyDemoRepo",
    "accountId": "111122223333",
    "type": "AWS::CodeCommit::Repository"
  }
],
"eventType": "AwsApiCall",
"apiVersion": "2015-04-13",
"recipientAccountId": "111122223333"
}

```

Contoh: Entri log untuk Git menarik panggilan ke CodeCommit repositori

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `GitPull` tindakan di mana repo lokal sudah up-to-date.

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",

```

```

    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::444455556666:user/Mary_Major",
    "accountId": "444455556666",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "Mary_Major"
  },
  "eventTime": "2016-12-14T18:19:15Z",
  "eventSource": "codecommit.amazonaws.com",
  "eventName": "GitPull",
  "awsRegion": "us-east-2",
  "sourceIPAddress": "203.0.113.12",
  "userAgent": "git/2.11.0.windows.1",
  "requestParameters": null,
  "responseElements": null,
  "additionalEventData": {
    "protocol": "HTTP",
    "dataTransferred": false,
    "repositoryName": "MyDemoRepo",
    "repositoryId": "8afe792d-EXAMPLE",
  },
  "requestID": "d148de46-EXAMPLE",
  "eventID": "740f179d-EXAMPLE",
  "readOnly": true,
  "resources": [
    {
      "ARN": "arn:aws:codecommit:us-east-2:111122223333:MyDemoRepo",
      "accountId": "111122223333",
      "type": "AWS::CodeCommit::Repository"
    }
  ],
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
}

```

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `GitPull` tindakan di mana repo lokal tidak up-to-date dan data ditransfer dari CodeCommit repositori ke repo lokal.

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::444455556666:user/Mary_Major",

```

```
"accountId": "444455556666",
"accessKeyId": "AKIAIOSFODNN7EXAMPLE",
"userName": "Mary_Major"
},
"eventTime": "2016-12-14T18:19:15Z",
"eventSource": "codecommit.amazonaws.com",
"eventName": "GitPull",
"awsRegion": "us-east-2",
"sourceIPAddress": "203.0.113.12",
"userAgent": "git/2.10.1",
"requestParameters": null,
"responseElements": null,
"additionalEventData": {
  "protocol": "HTTP",
  "capabilities": [
    "multi_ack_detailed",
    "side-band-64k",
    "thin-pack"
  ],
  "dataTransferred": true,
  "repositoryName": "MyDemoRepo",
  "repositoryId": "8afe792d-EXAMPLE",
  "shallow": false
},
"requestID": "d148de46-EXAMPLE",
"eventID": "740f179d-EXAMPLE",
"readOnly": true,
"resources": [
  {
    "ARN": "arn:aws:codecommit:us-east-2:111122223333:MyDemoRepo",
    "accountId": "111122223333",
    "type": "AWS::CodeCommit::Repository"
  }
],
"eventType": "AwsApiCall",
"recipientAccountId": "111122223333"
}
```

Contoh: Entri log untuk keberhasilan pengiriman ke CodeCommit repositori

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan `GitPush` tindakan yang berhasil. Tindakan `GitPush` muncul dua kali dalam entri log untuk keberhasilan pengiriman.


```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::444455556666:user/Mary_Major",
    "accountId": "444455556666",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "Mary_Major"
  },
  "eventTime": "2016-12-14T18:19:15Z",
  "eventSource": "codecommit.amazonaws.com",
  "eventName": "GitPush",
  "awsRegion": "us-east-2",
  "sourceIPAddress": "203.0.113.12",
  "userAgent": "git/2.10.1",
  "requestParameters": null,
  "responseElements": null,
  "additionalEventData": {
    "protocol": "HTTP",
    "dataTransferred": false,
    "repositoryName": "MyDemoRepo",
    "repositoryId": "8afe792d-EXAMPLE",
  },
  "requestID": "d148de46-EXAMPLE",
  "eventID": "740f179d-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "ARN": "arn:aws:codecommit:us-east-2:111122223333:MyDemoRepo",
      "accountId": "111122223333",
      "type": "AWS::CodeCommit::Repository"
    }
  ],
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
},
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::444455556666:user/Mary_Major",
```

```
    "accountId": "444455556666",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "Mary_Major"
  },
  "eventTime": "2016-12-14T18:19:15Z",
  "eventSource": "codecommit.amazonaws.com",
  "eventName": "GitPush",
  "awsRegion": "us-east-2",
  "sourceIPAddress": "203.0.113.12",
  "userAgent": "git/2.10.1",
  "requestParameters": {
    "references": [
      {
        "commit": "100644EXAMPLE",
        "ref": "refs/heads/main"
      }
    ]
  },
  "responseElements": null,
  "additionalEventData": {
    "protocol": "HTTP",
    "capabilities": [
      "report-status",
      "side-band-64k"
    ],
    "dataTransferred": true,
    "repositoryName": "MyDemoRepo",
    "repositoryId": "8afe792d-EXAMPLE",
  },
  "requestID": "d148de46-EXAMPLE",
  "eventID": "740f179d-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "ARN": "arn:aws:codecommit:us-east-2:111122223333:MyDemoRepo",
      "accountId": "111122223333",
      "type": "AWS::CodeCommit::Repository"
    }
  ],
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
}
```

Membuat sumber daya CodeCommit dengan AWS CloudFormation

AWS CodeCommit terintegrasi dengan AWS CloudFormation, yaitu layanan yang membantu Anda membuat model dan mengatur sumber daya AWS agar Anda dapat menghemat waktu untuk membuat dan mengelola sumber daya dan infrastruktur Anda. Anda membuat templat yang menggambarkan semua sumber daya AWS yang Anda inginkan (seperti repositori), dan AWS CloudFormation memasok persediaan dan mengonfigurasi sumber daya tersebut untuk Anda.

Saat menggunakan AWS CloudFormation, Anda dapat menggunakan kembali templat Anda untuk menyiapkan sumber daya CodeCommit secara konsisten dan berulang kali. Jelaskan sumber daya Anda satu kali, lalu sediakan sumber daya yang sama berulang kali dalam beberapa Akun AWS dan Wilayah.

Templat CodeCommit dan AWS CloudFormation

Untuk menyediakan dan mengonfigurasi sumber daya untuk CodeCommit dan layanan terkait, Anda harus memahami [templat AWS CloudFormation](#). Templat adalah file teks dengan format JSON atau YAML. Templat ini menjelaskan sumber daya yang ingin Anda sediakan di tumpukan AWS CloudFormation Anda. Jika Anda tidak terbiasa dengan JSON atau YAML, Anda dapat menggunakan AWS CloudFormation Designer untuk membantu Anda memulai dengan templat AWS CloudFormation. Untuk informasi selengkapnya, lihat [Apa yang dimaksud dengan AWS CloudFormation Designer?](#) dalam Panduan Pengguna AWS CloudFormation.

CodeCommit support pembuatan repositori di AWS CloudFormation Tidak seperti membuat repositori dari konsol atau baris perintah, Anda dapat menggunakan AWS CloudFormation untuk membuat repositori dan secara otomatis commit kode ke repositori yang baru dibuat dari file .zip yang ditentukan dalam bucket Amazon S3. Untuk informasi lebih lanjut, termasuk contoh templat JSON dan YAL untuk repositori, lihat [AWS::CodeCommit::Repository](#).

Saat Anda membuat CodeCommit repositori menggunakan AWS CloudFormation, Anda memiliki opsi untuk melakukan kode ke repositori tersebut sebagai bagian dari proses pembuatan selama arsip kurang dari 20 MB dengan mengonfigurasi properti di [AWS::CodeCommit::Repository Code](#). Anda dapat menentukan bucket Amazon S3 di mana kode disimpan, dan memilih menggunakan [BranchName properti](#) untuk menentukan nama cabang default yang akan dibuat di commit awal kode itu. Properti ini hanya digunakan dalam pembuatan repositori awal, dan diabaikan pada

pembaruan tumpukan. Anda tidak dapat menggunakan properti ini untuk membuat commit tambahan ke repositori, atau untuk mengubah nama cabang default setelah commit awal dibuat.

Note

Pada tanggal 19 Januari 2021, AWS mengubah nama cabang default di CodeCommit dari master ke utama. Perubahan nama ini mempengaruhi perilaku default CodeCommit saat membuat commit awal untuk repositori menggunakan CodeCommit konsol, CodeCommit API, AWS SDK, dan AWS CLI. Repositori dibuat dengan AWS CloudFormation atau AWS CDK dengan commit awal kode sebagai bagian dari penciptaan yang selaras dengan perubahan ini mulai 4 Maret 2021. Perubahan ini tidak mempengaruhi repositori atau cabang yang ada. Pelanggan yang menggunakan klien Git lokal untuk membuat commit awal mereka memiliki nama cabang default yang mengikuti konfigurasi klien Git tersebut. Untuk informasi selengkapnya, lihat [Bekerja dengan cabang](#), [Buat commit](#), dan [Ubah pengaturan cabang](#).

Anda juga dapat membuat templat yang membuat sumber terkait, seperti [aturan pemberitahuan](#) untuk repositori, [Membangun proyek AWS CodeBuild](#), [aplikasi AWS CodeDeploy](#), dan [alur AWS CodePipeline](#).

Contoh templat

Contoh berikut membuat CodeCommit repositori bernama *MyDemoRepo*. Repositori yang baru dibuat diisi dengan kode yang disimpan dalam bucket Amazon S3 bernama *MySourceCodeBucket* dan ditempatkan di cabang bernama *pengembangan*, yang merupakan cabang default untuk repositori.

Note

Nama bucket Amazon S3 yang berisi file ZIP dengan konten yang akan di-commit-kan untuk repositori baru dapat ditentukan menggunakan ARN atau nama bucket di akun Amazon Web Services. Kunci objek Amazon S3 adalah sebagaimana dijabarkan dalam [Panduan Developer Amazon S3](#).

JSON:

```
{
```

```

    "MyRepo": {
      "Type": "AWS::CodeCommit::Repository",
      "Properties": {
        "RepositoryName": "MyDemoRepo",
        "RepositoryDescription": "This is a repository for my project with code
from MySourceCodeBucket.",
        "Code": {
          "BranchName": "development",
          "S3": {
            "Bucket": "MySourceCodeBucket",
            "Key": "MyKey",
            "ObjectVersion": "1"
          }
        }
      }
    }
  }
}

```

YAML:

```

MyRepo:
  Type: AWS::CodeCommit::Repository
  Properties:
    RepositoryName: MyDemoRepo
    RepositoryDescription: This is a repository for my project with code from
MySourceCodeBucket.
  Code:
    BranchName: development
    S3:
      Bucket: MySourceCodeBucket,
      Key: MyKey,
      ObjectVersion: 1

```

Untuk contoh lainnya, lihat [AWS::CodeCommit::Repository](#).

AWS CloudFormation, CodeCommit, dan AWS Cloud Development Kit (AWS CDK)

Repositori yang dibuat menggunakan AWS CDK menggunakan fungsionalitas AWS CloudFormation dalam pembuatannya. Memahami bagaimana AWS CloudFormation templat bekerja dengan CodeCommit sumber daya dapat membantu Anda membuat dan mengelola AWS CDK kode

Anda. Untuk informasi selengkapnya tentang AWS CDK, lihat [Panduan Developer AWS Cloud Development Kit \(AWS CDK\)](#) dan [Referensi API AWS CDK](#).

Contoh AWS CDK naskah berikut menciptakan CodeCommit repositori bernama *MyDemoRepo*. Repositori yang baru dibuat diisi dengan kode yang disimpan dalam bucket Amazon S3 bernama *MySourceCodeBucket* dan ditempatkan di cabang bernama *pengembangan*, yang merupakan cabang default untuk repositori.

```
import * as cdk from '@aws-cdk/core';
import codecommit = require('@aws-cdk/aws-codecommit');
export class CdkCodecommitStack extends cdk.Stack {
  constructor(scope: cdk.Construct, id: string, props?: cdk.StackProps) {
    super(scope, id, props);
    // The code creates a CodeCommit repository with a default branch name development
    new codecommit.CfnRepository(this, 'MyRepoResource', {
      repositoryName: "MyDemoRepo",
      code: {
        "branchName": "development",
        "s3": {
          "bucket": "MySourceCodeBucket",
          "key": "MyKey"
        }
      },
    });
  }
}
```

Pelajari selengkapnya tentang AWS CloudFormation

Untuk mempelajari selengkapnya tentang AWS CloudFormation, lihat sumber daya berikut:

- [AWS CloudFormation](#)
- [AWS CloudFormation Panduan Pengguna](#)
- [AWS CloudFormation Panduan Pengguna Baris Perintah](#)

Pemecahan Masalah AWS CodeCommit

Informasi berikut dapat membantu Anda memecahkan masalah umum di AWS CodeCommit.

Topik

- [Mengatasi masalah kredensial Git dan koneksi HTTPS ke AWS CodeCommit](#)
- [Mengatasi masalah git-remote-codecommit dan AWS CodeCommit](#)
- [Pemecahan masalah koneksi SSH ke AWS CodeCommit](#)
- [Pemecahan masalah pembantu kredensial dan koneksi HTTPS ke AWS CodeCommit](#)
- [Mengatasi masalah klien Git dan AWS CodeCommit](#)
- [Mengatasi masalah kesalahan akses dan AWS CodeCommit](#)
- [Mengatasi masalah kesalahan konfigurasi dan AWS CodeCommit](#)
- [Mengatasi masalah kesalahan konsol dan AWS CodeCommit](#)
- [Pemecahan masalah pemicu dan AWS CodeCommit](#)
- [Mengaktifkan debugging](#)

Mengatasi masalah kredensial Git dan koneksi HTTPS ke AWS CodeCommit

Informasi berikut dapat membantu Anda memecahkan masalah umum yang terjadi saat menggunakan kredensial Git dan HTTPS untuk terhubung ke repositori AWS CodeCommit.

Topik

- [Kredensial Git untuk AWS CodeCommit: Saya terus melihat permintaan kredensial ketika saya terhubung ke repositori CodeCommit saya di terminal atau baris perintah](#)
- [Kredensial Git untuk AWS CodeCommit: Saya menyiapkan kredensial Git, tetapi sistem saya tidak menggunakannya](#)

Kredensial Git untukAWS CodeCommit: Saya terus melihat permintaan kredensial ketika saya terhubung ke repositori CodeCommit saya di terminal atau baris perintah

Masalah: Ketika Anda mencoba untuk melakukan push, pull, atau interaksi lainnya dengan repositori CodeCommit dari terminal atau baris perintah, Anda akan diminta untuk memberikan nama pengguna dan kata sandi, dan Anda harus menyediakan kredensial Git untuk pengguna IAM Anda.

Perbaikan yang dimungkinkan: Penyebab paling umum untuk kesalahan ini adalah bahwa komputer lokal Anda menjalankan sistem operasi yang tidak mendukung pengelolaan kredensial, atau tidak menginstal utilitas pengelolaan kredensial, atau kredensial Git untuk pengguna IAM Anda belum disimpan ke salah satu sistem pengelolaan kredensial ini. Tergantung pada sistem operasi dan lingkungan lokal Anda, Anda mungkin perlu menginstal pengelola kredensial, mengkonfigurasi pengelola kredensial yang disertakan dalam sistem operasi Anda, atau menyesuaikan lingkungan lokal Anda untuk menggunakan penyimpanan kredensial. Misalnya, jika komputer menjalankan macOS, maka Anda dapat menggunakan utilitas Keychain Access untuk menyimpan kredensial Anda. Jika komputer Anda menjalankan Windows, Anda dapat menggunakan Pengelola Kredensial Git yang diinstal dengan Git untuk Windows. Untuk informasi selengkapnya, lihat [Untuk pengguna HTTPS menggunakan kredensial Git](#) dan [Penyimpanan Kredensial](#) dalam dokumentasi Git.

Kredensial Git untukAWS CodeCommit: Saya menyiapkan kredensial Git, tetapi sistem saya tidak menggunakannya

Masalah: Ketika Anda mencoba untuk menggunakan CodeCommit dengan klien Git, klien tidak muncul untuk menggunakan kredensial Git untuk pengguna IAM Anda.

Perbaikan yang dimungkinkan: Penyebab paling umum untuk kesalahan ini adalah bahwa Anda sebelumnya menyiapkan komputer Anda untuk menggunakan penolong kredensial yang disertakan denganAWS CLI. Periksa file `.gitconfig` Anda untuk bagian konfigurasi yang mirip dengan berikut ini, dan hapus:

```
[credential "https://git-codecommit.*.amazonaws.com"]
  helper = !aws codecommit credential-helper $@
  UseHttpPath = true
```

Simpan file, dan kemudian buka baris perintah baru atau sesi terminal sebelum Anda mencoba untuk terhubung lagi.

Anda mungkin juga memiliki beberapa penolong atau pengelola kredensial yang disiapkan pada komputer Anda, dan sistem Anda mungkin secara default menggunakan konfigurasi yang lain. Untuk menyetel ulang penolong kredensial mana yang digunakan sebagai default, Anda dapat menggunakan pilihan `--system`, bukan `--global` atau `--local`, ketika menjalankan perintah `git config`.

Untuk informasi selengkapnya, lihat [Untuk pengguna HTTPS menggunakan kredensial Git](#) dan [Penyimpanan Kredensial](#) dalam dokumentasi Git.

Mengatasi masalah git-remote-codecommit dan AWS CodeCommit

Informasi berikut dapat membantu Anda memecahkan masalah git-remote-codecommit saat terhubung dengan repositori AWS CodeCommit.

Topik

- [Saya melihat kesalahan: git: 'remote-codecommit' bukan perintah git](#)
- [Saya melihat kesalahan: fatal: Tidak dapat menemukan penolong jarak jauh untuk 'codecommit'](#)
- [Kesalahan kloning: Saya tidak dapat mengkloning repositori CodeCommit dari IDE](#)
- [Push atau pull error: Saya tidak dapat melakukan push atau pull commit dari IDE ke repositori CodeCommit](#)

Saya melihat kesalahan: git: 'remote-codecommit' bukan perintah git

Masalah: Ketika Anda mencoba untuk menggunakan git-remote-codecommit, Anda melihat kesalahan yang menyatakan bahwa git-remote-codecommit bukan perintah git. Lihat 'git `—help`'.

Perbaiki yang dimungkinkan: Alasan paling umum untuk kesalahan ini adalah bahwa Anda belum menambahkan git-remote-codecommit executable ke PATH Anda, atau bahwa string berisikesalahan sintaks. Hal ini dapat terjadi bila tidak ada tanda hubung antara git dan remote-codecommit, atau ketika git tambahan ditempatkan sebelum git-remote-codecommit.

Untuk informasi lebih lanjut tentang pengaturan dan penggunaan git-remote-codecommit, lihat [Langkah-langkah penyiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit](#).

Saya melihat kesalahan: fatal: Tidak dapat menemukan penolong jarak jauh untuk 'codecommit'

Masalah: Ketika Anda mencoba untuk menggunakan git-remote-codecommit, Anda melihat kesalahan yang menyatakan “fatal: Tidak dapat menemukan penolong jarak jauh untuk 'codecommit'”.

Perbaiki yang dimungkinkan: Alasan paling umum untuk kesalahan ini adalah:

- Pengaturan ini tidak lengkap untuk git-remote-codecommit
- Anda telah menginstal git-remote-codecommit di lokasi yang tidak di path Anda atau tidak dikonfigurasi sebagai bagian dari variabel lingkungan Path
- Python tidak ada di path Anda atau tidak dikonfigurasi sebagai bagian dari variabel lingkungan Path
- Anda menggunakan jendela terminal atau baris perintah yang belum dimulai ulang sejak penginstalan git-remote-codecommit selesai

Untuk informasi lebih lanjut tentang pengaturan dan penggunaan git-remote-codecommit, lihat [Langkah-langkah penyiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit](#).

Kesalahan kloning: Saya tidak dapat mengkloning repositori CodeCommit dari IDE

Masalah: Ketika Anda mencoba untuk mengkloning repositori CodeCommit di IDE, Anda melihat kesalahan yang mengatakan titik akhir atau URL tidak valid.

Perbaiki yang dimungkinkan: Tidak semua IDE mendukung URL yang digunakan olehgit-remote-codecommit selama kloning. Lakukan klon repositori secara lokal dari terminal atau baris perintah, dan kemudian tambahkan repo lokal tersebut pada IDE Anda. Untuk informasi selengkapnya, lihat [Langkah 3: Hubungkan keCodeCommitkonsol dan kloning repositori](#).

Push atau pull error: Saya tidak dapat melakukan push atau pull commit dari IDE ke repositori CodeCommit

Masalah: Ketika Anda mencoba untuk melakukan pull atau push kode dari IDE, Anda melihat kesalahan koneksi.

Perbaikan yang dimungkinkan: Alasan paling umum untuk kesalahan ini adalah bahwa IDE tidak kompatibel dengan penolong jarak jauh Git seperti `git-remote-codecommit`. Alih-alih menggunakan fungsi IDE untuk melakukan commit, push, dan pull kode, perbarui repo lokal secara manual dari baris perintah atau terminal menggunakan perintah Git.

Untuk informasi lebih lanjut tentang penolong jarak jauh dan Git, lihat [dokumentasi Git](#).

Pemecahan masalah koneksi SSH ke AWS CodeCommit

Informasi berikut dapat membantu Anda memecahkan masalah umum saat menggunakan SSH untuk terhubung ke CodeCommit repositori.

Topik

- [Kesalahan akses: Kunci publik berhasil diunggah ke IAM namun koneksi gagal pada sistem Linux, macOS, atau Unix](#)
- [Kesalahan akses: Kunci publik berhasil diunggah ke IAM dan SSH berhasil diuji tetapi koneksi gagal pada sistem Windows](#)
- [Tantangan otentikasi: Keaslian host tidak dapat dibuat saat menghubungkan ke CodeCommit repositori](#)
- [Galat IAM: 'Format tidak valid' ketika mencoba untuk menambahkan kunci publik ke IAM](#)
- [Saya perlu mengakses CodeCommit repositori di beberapa akun Amazon Web Services dengan kredensi SSH](#)
- [Git pada Windows: Bash emulator atau baris perintah membeku ketika mencoba untuk terhubung menggunakan SSH](#)
- [Format kunci publik membutuhkan spesifikasi dalam beberapa distribusi Linux](#)
- [Kesalahan akses: Kunci publik SSH ditolak saat menghubungkan ke CodeCommit repositori](#)

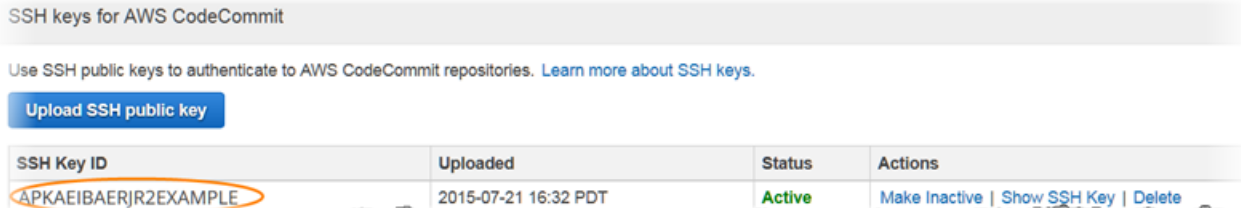
Kesalahan akses: Kunci publik berhasil diunggah ke IAM namun koneksi gagal pada sistem Linux, macOS, atau Unix

Masalah: Ketika Anda mencoba untuk terhubung ke endpoint SSH untuk berkomunikasi dengan CodeCommit repositori, baik saat menguji koneksi atau mengkloning repositori, koneksi gagal atau ditolak.

Perbaikan yang dimungkinkan: ID kunci SSH yang ditetapkan ke kunci publik Anda di IAM mungkin tidak terkait dengan upaya koneksi Anda. [Anda mungkin belum mengonfigurasi file konfigurasi](#), Anda

mungkin tidak memiliki akses ke file konfigurasi, pengaturan lain mungkin mencegah keberhasilan membaca file konfigurasi, Anda mungkin telah memberikan ID kunci yang salah, atau Anda mungkin telah memberikan ID dari pengguna IAM bukan ID kunci.

ID kunci SSH dapat ditemukan di konsol IAM di profil untuk pengguna IAM Anda:



SSH Key ID	Uploaded	Status	Actions
APKAEIBAERJR2EXAMPLE	2015-07-21 16:32 PDT	Active	Make Inactive Show SSH Key Delete

Note

Jika Anda memiliki lebih dari satu ID kunci SSH yang diunggah, kunci tercantum menurut abjad berdasarkan ID kunci, bukan tanggal pengunggahan. Pastikan bahwa Anda telah menyalin ID kunci yang terkait dengan tanggal pengunggahan yang benar.

Coba uji koneksi dengan perintah berikut:

```
ssh Your-SSH-Key-ID@git-codecommit.us-east-2.amazonaws.com
```

Jika Anda melihat pesan sukses setelah mengonfirmasi koneksi, ID kunci SSH Anda valid. Edit file konfigurasi Anda untuk mengaitkan upaya koneksi Anda dengan kunci publik Anda di IAM. Jika Anda tidak ingin mengedit file konfigurasi Anda, Anda dapat mendahulukan semua upaya koneksi ke repositori Anda dengan ID kunci SSH Anda. Misalnya, jika Anda ingin mengkloning repositori bernama *MyDemoRepo* tanpa memodifikasi file konfigurasi Anda untuk mengaitkan upaya koneksi Anda, Anda akan menjalankan perintah berikut:

```
git clone ssh://Your-SSH-Key-ID@git-codecommit.us-east-2.amazonaws.com/v1/  
repos/MyDemoRepo my-demo-repo
```

Untuk informasi selengkapnya, lihat [Untuk koneksi SSH di Linux, macOS, atau Unix](#).

Kesalahan akses: Kunci publik berhasil diunggah ke IAM dan SSH berhasil diuji tetapi koneksi gagal pada sistem Windows

Masalah:Ketika Anda mencoba menggunakan endpoint SSH untuk mengkloning atau berkomunikasi denganCodeCommitrepositori, pesan kesalahan muncul yang berisi frasaNo supported authentication methods available.

Perbaikan yang dimungkinkan: Alasan paling umum untuk kesalahan ini adalah bahwa Anda memiliki pengaturan variabel lingkungan sistem Windows yang mengarahkan Windows untuk menggunakan program lain ketika Anda mencoba untuk menggunakan SSH. Misalnya, Anda mungkin telah menetapkan variabel GIT_SSH untuk menunjuk ke salah satu set PuTTY alat (plink.exe). Ini mungkin konfigurasi lama, atau mungkin diperlukan untuk satu atau lebih program lain yang diinstal pada komputer Anda. Jika Anda yakin bahwa variabel lingkungan ini tidak diperlukan, Anda dapat menghapusnya dengan membuka properti sistem Anda.

Untuk mengatasi masalah ini, buka Bash emulator dan kemudian mencoba koneksi SSH Anda lagi, tetapi termasuk GIT_SSH_COMMAND="SSH" sebagai awalan. Misalnya, untuk mengkloning suatu repositori menggunakan SSH:

```
GIT_SSH_COMMAND="ssh" git clone ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Masalah serupa mungkin terjadi jika versi Windows mengharuskan Anda menyertakan ID kunci SSH sebagai bagian dari rangkaian sambungan saat menghubungkan melalui SSH pada baris perintah Windows. Coba koneksi Anda lagi, kali ini termasuk ID kunci SSH disalin dari IAM sebagai bagian dari perintah. Misalnya:

```
git clone ssh://Your-SSH-Key-ID@git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Tantangan otentikasi: Keaslian host tidak dapat dibuat saat menghubungkan keCodeCommitrepositori

Masalah:Ketika Anda mencoba menggunakan endpoint SSH untuk berkomunikasi denganCodeCommitrepositori, pesan peringatan muncul berisi frasaThe authenticity of host '*host-name*' can't be established.

Perbaikan yang dimungkinkan: Kredensial Anda mungkin tidak diatur dengan benar. Ikuti instruksi di [Untuk koneksi SSH di Linux, macOS, atau Unix](#) atau [Untuk koneksi SSH pada Windows](#).

Jika Anda telah mengikuti langkah-langkah itu dan masalah tetap ada, seseorang mungkin mencobaman-in-the-middle menyerang. Ketika Anda melihat pesan berikut, ketik no, dan tekan Enter.

```
Are you sure you want to continue connecting (yes/no)?
```

Pastikan sidik jari dan kunci publik untuk CodeCommit koneksi cocok dengan yang didokumentasikan dalam topik penyiapan SSH sebelum Anda melanjutkan koneksi.

Sidik jari publik untuk CodeCommit

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.us-east-2.amazonaws.com	MD5	a9:6d:03:ed:08:42: 21:be:06:e1:e0:2a: d1:75:31:5e
git-codecommit.us-east-2.amazonaws.com	SHA256	3lB1W2g5xn/NA2Ck6d yeJIrQ0Wvn7n8UEs56 fG6ZIZQ
git-codecommit.us-east-1.amazonaws.com	MD5	a6:9c:7d:bc:35:f5: d4:5f:8b:ba:6f:c8: bc:d4:83:84
git-codecommit.us-east-1.amazonaws.com	SHA256	eLMY1j0DKA4uvDZc1/ KgtIayZANwX6t8+8is PtotBoY
git-codecommit.us-west-2.amazonaws.com	MD5	a8:68:53:e3:99:ac: 6e:d7:04:7e:f7:92: 95:77:a9:77
git-codecommit.us-west-2.amazonaws.com	SHA256	0pJx9SQpkbPUAHwy58 UVIq0IHcyo1fwCp00u VgcAWPo

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.eu-west-1.amazonaws.com	MD5	93:42:36:ea:22:1f: f1:0f:20:02:4a:79: ff:ea:12:1d
git-codecommit.eu-west-1.amazonaws.com	SHA256	tKjRk0L8dmJyTmSbeS dN1S8F/f0iql3R1vqg TOP1UyQ
git-codecommit.ap-northeast-1.amazonaws.com	MD5	8e:a3:f0:80:98:48: 1c:5c:6f:59:db:a7: 8f:6e:c6:cb
git-codecommit.ap-northeast-1.amazonaws.com	SHA256	Xk/WeYD/K/bnBybzhi uu4dWpBJtXPf7E30jH U7se40w
git-codecommit.ap-southeast-1.amazonaws.com	MD5	65:e5:27:c3:09:68: 0d:8e:b7:6d:94:25: 80:3e:93:cf
git-codecommit.ap-southeast-1.amazonaws.com	SHA256	ZIsVa70VzxrTIf+Rk4 UbhPv6Es22mSB3uTBo jfPXIno
git-codecommit.ap-southeast-2.amazonaws.com	MD5	7b:d2:c1:24:e6:91: a5:7b:fa:c1:0c:35: 95:87:da:a0
git-codecommit.ap-southeast-2.amazonaws.com	SHA256	nYp+gHas80HY3DqbP4 yanCDFhqDVjseefVbH EXqH2Ec
git-codecommit.ap-southeast-3.amazonaws.com	MD5	64:d9:e0:53:19:4f: a8:91:9a:c3:53:22: a6:a8:ed:a6

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.ap-southeast-3.amazonaws.com	SHA256	ATdkGSFhpqIu7RqUVT/1RZo6MLxxxUW9NoDV MbAc/6g
git-codecommit.me-central-1.amazonaws.com	MD5	bd:fa:e2:f9:05:84:d6:39:6f:bc:d6:8d:fe:de:61:76
git-codecommit.me-central-1.amazonaws.com	SHA256	grceUDWubo4MzG1NoaKZKUfrgPvfN3ijli0nQr11TZA
git-codecommit.eu-central-1.amazonaws.com	MD5	74:5a:e8:02:fc:b2:9c:06:10:b4:78:84:65:94:22:2d
git-codecommit.eu-central-1.amazonaws.com	SHA256	MwGrkiEki8QkkBt1AgXbYt0hoZYBnZF62VY5RzGJEUy
git-codecommit.ap-northeast-2.amazonaws.com	MD5	9f:68:48:9b:5f:fc:96:69:39:45:58:87:95:b3:69:ed
git-codecommit.ap-northeast-2.amazonaws.com	SHA256	eegAPQrWY9YsYo9ZHIK0mxfXBHzAZd8Eya53Qcwko
git-codecommit.sa-east-1.amazonaws.com	MD5	74:99:9d:ff:2b:ef:63:c6:4b:b4:6a:7f:62:c5:4b:51
git-codecommit.sa-east-1.amazonaws.com	SHA256	kW+VKB0jpRaG/ZbXkgbtMQbKgEDK7JnISV3SVoyCmzU

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.us-west-1.amazonaws.com	MD5	3b:76:18:83:13:2c: f8:eb:e9:a3:d0:51: 10:32:e7:d1
git-codecommit.us-west-1.amazonaws.com	SHA256	gzauWTWXDK2u5KuMMi 5vbKTmfyerdIwgSbzY BODLpzg
git-codecommit.eu-west-2.amazonaws.com	MD5	a5:65:a6:b1:84:02: b1:95:43:f9:0e:de: dd:ed:61:d3
git-codecommit.eu-west-2.amazonaws.com	SHA256	r0Rwz5k/IHp/QyrRnf iM9j02D5UEqMbtFNTu DG2hNbs
git-codecommit.ap-south-1.amazonaws.com	MD5	da:41:1e:07:3b:9e: 76:a0:c5:1e:64:88: 03:69:86:21
git-codecommit.ap-south-1.amazonaws.com	SHA256	hUKwnTj7+Xpx4Kddb6 p45j4RazIJ4IhAMD8k 29it0fE
git-codecommit.ap-south-2.amazonaws.com	MD5	bc:cc:9f:15:f8:f3: 58:a2:68:65:21:e2: 23:71:8d:ce
git-codecommit.ap-south-2.amazonaws.com	SHA256	Xe0CyZE0vgR5Xa2YUG qf+jn8/Ut717nX/Cms 1SFNEig
git-codecommit.ca-central-1.amazonaws.com	MD5	9f:7c:a2:2f:8c:b5: 74:fd:ab:b7:e1:fd: af:46:ed:23

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.ca-central-1.amazonaws.com	SHA256	Qz5pua fQdANVprL1j6 r0Qyh4lCNsF6ob61dG cPtFS7w
git-codecommit.eu-west-3.amazonaws.com	MD5	1b:7f:97:dd:d7:76: 8a:32:2c:bd:2c:7b: 33:74:6a:76
git-codecommit.eu-west-3.amazonaws.com	SHA256	uw7c2FL564jVoFgtc+ ikzILnKBsZz7t9+CFd SJjKbLI
git-codecommit.us-gov-west-1.amazonaws.com	MD5	9f:6c:19:3b:88:cd: e8:88:1b:9c:98:6a: 95:31:8a:69
git-codecommit.us-gov-west-1.amazonaws.com	SHA256	djXQoSIFcg8vHe0KVH 1xW/g0F9X37tWTqu4H kng75x4
git-codecommit.us-gov-east-1.amazonaws.com	MD5	00:8d:b5:55:6f:05: 78:05:ed:ea:cb:3f: e6:f0:62:f2
git-codecommit.us-gov-east-1.amazonaws.com	SHA256	fVb+R0z7qW7minenW+ rUpAABRCRBTCzmETAJ EQrg98
git-codecommit.eu-north-1.amazonaws.com	MD5	8e:53:d8:59:35:88: 82:fd:73:4b:60:8a: 50:70:38:f4
git-codecommit.eu-north-1.amazonaws.com	SHA256	b6KSK7xKq+V8j17iuA cjqXsG7zkqoUZZmmhY YFBq1wQ

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.me-south-1.amazonaws.com	MD5	0e:39:28:56:d5:41: e6:8d:fa:81:45:37: fb:f3:cd:f7
git-codecommit.me-south-1.amazonaws.com	SHA256	0+NToCGgj rHekiBu01 0ad7R0GEsz+DBLX0d/ c9wc0JU
git-codecommit.ap-east-1.amazonaws.com	MD5	a8:00:3d:24:52:9d: 61:0e:f6:e3:88:c8: 96:01:1c:fe
git-codecommit.ap-east-1.amazonaws.com	SHA256	LafadYwUYW8h0NoTRp objjNs9IRnbEwHtezD 3aAIBX0
git-codecommit.cn-north-1.amazonaws.com.cn	MD5	11:7e:2d:74:9e:3b: 94:a2:69:14:75:6f: 5e:22:3b:b3
git-codecommit.cn-north-1.amazonaws.com.cn	SHA256	IYUXxH20pTDsyYMLIp +JY8CTLS4UX+ZC5JVZ XPRaxc8
git-codecommit.cn-northwest-1.amazonaws.com.cn	MD5	2e:a7:fb:4c:33:ac: 6c:f9:aa:f2:bc:fb: 0a:7b:1e:b6
git-codecommit.cn-northwest-1.amazonaws.com.cn	SHA256	wqjd6eHd0+m0Bx+dCN uL0omUoCNjaDtZiEpW j5TmCfQ
git-codecommit.eu-south-1.amazonaws.com	MD5	b9:f6:5d:e2:48:92: 3f:a9:37:1e:c4:d0: 32:0e:fb:11

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.eu-south-1.amazonaws.com	SHA256	1yXrWbCg3uQmJr11XxB/ASR7ugW1Ysf5yzY0JbudHsI
git-codecommit.ap-northeast-3.amazonaws.com	MD5	25:17:40:da:b9:d4:18:c3:b6:b3:fb:ed:1c:20:fe:29
git-codecommit.ap-northeast-3.amazonaws.com	SHA256	2B815B9F0AvwLnRxSVxUz4kDYmtEQUGGdQYP8OQLXhA
git-codecommit.af-south-1.amazonaws.com	MD5	21:a0:ba:d7:c1:d1:b5:39:98:8d:4d:7c:96:f5:ca:29
git-codecommit.af-south-1.amazonaws.com	SHA256	C34ji3x/cnsDZjUpyNGXdE5pjHYimqJrQZ31eTgqJHM
git-codecommit.il-central-1.amazonaws.com	MD5	04:74:89:16:98:7a:61:b1:69:46:42:3c:d1:b4:ac:a9
git-codecommit.il-central-1.amazonaws.com	SHA256	uFxhp51kUWh1eTLeYbxQVYm4RnNLNZ5Dbdm1cgdS1/8

Galat IAM: 'Format tidak valid' ketika mencoba untuk menambahkan kunci publik ke IAM

Masalah: Di IAM, saat mencoba mengatur untuk menggunakan SSH dengan CodeCommit, pesan kesalahan muncul yang berisi frasa `Invalid format` ketika Anda mencoba untuk menambahkan kunci publik Anda.

Kemungkinan perbaikan:IAM mensyaratkan bahwa kunci publik harus dikodekan dalam format ssh-rsa atau format PEM. Ini menerima kunci publik dalam format OpenSSH saja dan memiliki persyaratan tambahan seperti yang ditentukan dalam[Gunakan SSH Keys denganCodeCommit](#)diPanduan Pengguna IAM. Jika Anda memberikan kunci publik Anda dalam format lain, atau jika kunci tidak berisi jumlah bit yang diperlukan, Anda akan melihat kesalahan ini.

- Ketika Anda menyalin kunci publik SSH, sistem operasi Anda mungkin telah memperkenalkan jeda baris. Pastikan bahwa tidak ada jeda baris dalam kunci publik yang Anda tambahkan ke IAM.
- Beberapa sistem operasi Windows tidak menggunakan format OpenSSH. Untuk menghasilkan pasangan kunci dan menyalin format OpenSSH yang dibutuhkan oleh IAM, lihat [the section called “Langkah 3: Siapkan kunci publik dan privat untuk Git dan CodeCommit”](#).

Untuk informasi selengkapnya tentang persyaratan kunci SSH di IAM, lihat[Gunakan SSH Keys denganCodeCommit](#)diPanduan Pengguna IAM.

Saya perlu mengaksesCodeCommitrepositori di beberapa akun Amazon Web Services dengan kredensi SSH

Masalah:Saya ingin mengatur akses SSH keCodeCommitrepositori di lebih dari satu akun Amazon Web Services.

Perbaikan yang dimungkinkan: Anda dapat membuat pasangan kunci publik/privat SSH unik untuk setiap akun Amazon Web Services dan mengkonfigurasi IAM dengan setiap tombol. Anda kemudian dapat mengkonfigurasi file `~/.ssh/config` Anda dengan informasi tentang setiap ID Pengguna IAM yang terkait dengan kunci publik. Misalnya:

```
Host codecommit-1
  Hostname git-codecommit.us-east-1.amazonaws.com
  User SSH-KEY-ID-1 # This is the SSH Key ID you copied from IAM in Amazon Web
  Services account 1 (for example, APKAEIBAERJR2EXAMPLE1).
  IdentityFile ~/.ssh/codecommit_rsa # This is the path to the associated public key
  file, such as id_rsa. We advise creating CodeCommit specific _rsa files.

Host codecommit-2
  Hostname git-codecommit.us-east-1.amazonaws.com
  User SSH-KEY-ID-2 # This is the SSH Key ID you copied from IAM in Amazon Web
  Services account 2 (for example, APKAEIBAERJR2EXAMPLE2).
  IdentityFile ~/.ssh/codecommit_2_rsa # This is the path to the other associated
  public key file. We advise creating CodeCommit specific _rsa files.
```

Dalam konfigurasi ini, Anda akan dapat mengganti 'git-codecommit.us-east-1.amazonaws.com' dengan 'codecommit-2'. Misalnya, untuk mengkloning repositori di akun Amazon Web Services kedua Anda:

```
git clone ssh://codecommit-2/v1/repos/YourRepositoryName
```

Untuk mengatur remote untuk repositori Anda, jalankan git remote add. Misalnya:

```
git remote add origin ssh://codecommit-2/v1/repos/YourRepositoryName
```

Untuk contoh lainnya, lihat [posting forum ini](#) dan [kontribusi ini pada GitHub](#).

Git pada Windows: Bash emulator atau baris perintah membeku ketika mencoba untuk terhubung menggunakan SSH

Masalah: Setelah Anda mengkonfigurasi akses SSH untuk Windows dan mengkonfirmasi konektivitas pada baris perintah atau terminal, Anda melihat pesan bahwa server host kunci tidak disimpan dalam registri, dan prompt untuk menyimpan kunci dalam cache beku (tidak menerima y/n/return input) ketika Anda mencoba untuk menggunakan perintah seperti git pull, git push, atau git clone pada prompt perintah atau Bash emulator.

Perbaiki yang dimungkinkan: Penyebab paling umum untuk kesalahan ini adalah bahwa lingkungan Git Anda dikonfigurasi untuk menggunakan sesuatu selain OpenSSH untuk autentikasi (mungkin PuTTY). Hal ini diketahui menyebabkan masalah dengan cache kunci dalam beberapa konfigurasi. Untuk memperbaiki masalah ini, cobalah lakukan salah satu hal berikut:

- Buka emulator Bash dan tambahkan parameter `GIT_SSH_COMMAND="ssh"` sebelum perintah Git. Sebagai contoh, jika Anda mencoba untuk mengirim ke repositori, jangan mengetik git push, tetapi ketik:

```
GIT_SSH_COMMAND="ssh" git push
```

- Jika Anda telah menginstal PuTTY, buka PuTTY, dan diNama Host (atau alamat IP), masukkanCodeCommittitik akhir yang ingin Anda jangkau (misalnya, git-codecommit.us-east-2.amazonaws.com). Pilih Buka . Saat diminta oleh peringatan keamanan PuTTY, pilih Ya untuk menyimpan kunci secara permanen.

- Ubah nama atau hapus variabel lingkungan GIT_SSH jika Anda tidak lagi menggunakannya. Kemudian buka command prompt baru atau sesi emulator Bash, dan coba jalankan perintah Anda lagi.

Untuk solusi lainnya, lihat [Git clone/pull terus membeku di kunci Penyimpanan dalam cache](#) pada Stack Overflow.

Format kunci publik membutuhkan spesifikasi dalam beberapa distribusi Linux

Masalah: Ketika Anda mencoba mengkonfigurasi pasangan kunci publik-privat, Anda akan menerima kesalahan.

Perbaiki yang dimungkinkan: Beberapa distribusi Linux memerlukan baris konfigurasi tambahan di file `~/.ssh/config` yang menentukan jenis kunci publik yang diterima. Untuk informasi lebih lanjut, lihat dokumentasi untuk Anda bagian tentang `PubkeyAcceptedKeyTypes`.

Kesalahan akses: Kunci publik SSH ditolak saat menghubungkan keCodeCommitrepositori

Masalah:Ketika Anda mencoba menggunakan endpoint SSH untuk berkomunikasi denganCodeCommitrepositori, pesan kesalahan muncul yang berisi frasa`Error: public key denied`.

Perbaiki yang mungkin dilakukan: Alasan paling umum untuk kesalahan ini adalah bahwa Anda belum menyelesaikan setup untuk koneksi SSH. Mengkonfigurasi pasangan kunci SSH publik dan privat, dan kemudian mengaitkan kunci publik dengan pengguna IAM Anda. Untuk informasi lebih lanjut tentang mengkonfigurasi SSH, lihat [Untuk koneksi SSH di Linux, macOS, atau Unix](#) dan [Untuk koneksi SSH pada Windows](#).

Pemecahan masalah pembantu kredensial dan koneksi HTTPS ke AWS CodeCommit

Informasi berikut dapat membantu Anda memecahkan masalah umum yang terjadi saat menggunakan pembantu kredensial yang termasuk dengan AWS CLI dan HTTPS untuk terhubung ke repositori CodeCommit.

Note

Meskipun pembantu kredensial adalah metode yang didukung untuk menghubungkan ke CodeCommit menggunakan akses gabungan, penyedia identitas, atau kredensial sementara, metode yang disarankan adalah menginstal dan menggunakan utilitas `git-remote-codecommit`. Untuk informasi selengkapnya, lihat [Langkah-langkah penyiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit](#).

Topik

- [Saya menerima galat saat menjalankan perintah `git config` untuk mengkonfigurasi pembantu kredensial](#)
- [Saya mendapatkan perintah tidak ditemukan kesalahan di Windows saat menggunakan pembantu kredensial](#)
- [Saya diminta untuk memasukkan nama pengguna ketika saya terhubung ke sebuah repositori CodeCommit](#)
- [Git untuk macOS: Saya mengkonfigurasi pembantu kredensial dengan sukses, tapi sekarang saya menolak akses ke repositori saya \(403\)](#)
- [Git untuk Windows: Saya menginstal Git untuk Windows, tapi akses saya ke repositori saya ditolak \(403\)](#)

Saya menerima galat saat menjalankan perintah **git config** untuk mengkonfigurasi pembantu kredensial

Masalah: Ketika Anda mencoba untuk menjalankan perintah `git config` untuk mengkonfigurasi pembantu kredensial untuk berkomunikasi dengan repositori CodeCommit, Anda melihat kesalahan bahwa terlalu sedikit argumen, atau prompt penggunaan menyarankan perintah dan sintaks Git config.

Perbaikan yang dimungkinkan: Alasan paling umum untuk kesalahan ini adalah bahwa tanda kutip tunggal digunakan untuk perintah pada sistem operasi Windows, atau tanda kutip ganda digunakan untuk perintah dalam sistem operasi Linux, macOS, atau Unix. Sintaks yang benar adalah sebagai berikut:

- Windows: `git config --global credential.helper "!aws codecommit credential-helper %@"`

- Linux, macOS, atau Unix: `git config --global credential.helper '!aws codecommit credential-helper $@'`

Saya mendapatkan perintah tidak ditemukan kesalahan di Windows saat menggunakan pembantu kredensial

Masalah: Setelah memperbarui AWS CLI, koneksi pembantu kredensial ke repositori CodeCommit gagal dengan `aws codecommit credential-helper $@ get: aws: command not found`.

Penyebab: Alasan paling umum untuk kesalahan ini adalah bahwa kesalahan Anda AWS CLI telah diperbarui ke versi yang menggunakan Python 3. Ada masalah yang diketahui dengan paket MSI. Untuk memverifikasi apakah Anda memiliki salah satu versi yang terpengaruh, buka baris perintah dan jalankan perintah berikut: `aws --version`

Jika versi Python output dimulai dengan 3, Anda memiliki versi yang terpengaruh. Misalnya:

```
aws-cli/1.16.62 Python/3.6.2 Darwin/16.7.0 botocore/1.12.52
```

Perbaiki yang dimungkinkan: Anda dapat mengatasi masalah ini dengan melakukan salah satu hal berikut:

- Instal dan konfigurasi AWS CLI pada Windows menggunakan Python dan pip bukannya MSI. Untuk informasi selengkapnya, lihat [Instal Python, pip, dan AWS CLI di Windows](#).
- Mengedit secara manual file `.gitconfig` Anda untuk mengubah bagian `[credential]` untuk secara eksplisit menunjuk ke `aws.cmd` di komputer lokal Anda. Misalnya:

```
[credential]
  helper = !"C:\\Program Files\\Amazon\\AWSCLI\\bin\\aws.cmd\" codecommit
  credential-helper $@
  UseHttpPath = true
```

- Jalankan perintah `git config` untuk memperbarui file `.gitconfig` Anda untuk secara eksplisit merujuk ke `aws.cmd`, dan secara manual memperbarui variabel lingkungan `PATH` Anda untuk menyertakan path ke perintah yang diperlukan. Misalnya:

```
git config --global credential.helper "!aws.cmd codecommit credential-helper $@"
git config --global credential.UseHttpPath true
```

Saya diminta untuk memasukkan nama pengguna ketika saya terhubung ke sebuah repositori CodeCommit

Masalah: Ketika Anda mencoba untuk menggunakan pembantu kredenal untuk berkomunikasi dengan repositori CodeCommit, muncul pesan yang meminta Anda untuk memasukkan nama pengguna Anda.

Perbaikan yang dimungkinkan: Mengonfigurasi AWS profil atau pastikan profil yang Anda gunakan adalah profil yang Anda konfigurasi untuk bekerja dengan CodeCommit. Untuk informasi lebih lanjut tentang pengaturan, lihat [Langkah-langkah pengaturan untuk koneksi HTTPS ke AWS CodeCommit di Linux, macOS, atau Unix dengan pembantu kredensial AWS CLI](#), atau [Langkah pengaturan untuk koneksi HTTPS ke repositori AWS CodeCommit pada Windows dengan pembantu kredensial AWS CLI](#). Untuk informasi selengkapnya tentang IAM, access key, dan secret key, lihat [Mengelola Access Key](#) dan [Bagaimana Saya Mendapatkan Kredensial?](#)

Git untuk macOS: Saya mengkonfigurasi pembantu kredenal dengan sukses, tapi sekarang saya menolak akses ke repositori saya (403)

Masalah: Di macOS, pembantu kredensial sepertinya tidak mengakses atau menggunakan kredensial Anda seperti yang diharapkan. Hal ini dapat disebabkan oleh dua masalah yang berbeda:

- AWS CLI dikonfigurasi untuk Wilayah AWS berbeda dari tempat repositori berada.
- Keychain Access telah menyimpan kredensial yang telah kedaluwarsa.

Perbaikan yang dimungkinkan: Untuk memverifikasi apakah AWS CLI dikonfigurasi untuk wilayah yang benar, jalankan `aws configure` perintah, dan meninjau informasi yang ditampilkan. Jika repositori CodeCommit berada dalam Wilayah AWS berbeda dari yang ditunjukkan untuk AWS CLI, Anda harus menjalankan perintah `aws configure` dan mengubah nilai yang sesuai untuk Wilayah tersebut. Untuk informasi selengkapnya, lihat [Langkah 1: Konfigurasi awal untuk CodeCommit](#).

Versi default Git yang dirilis di OS X dan macOS menggunakan utilitas Keychain Access untuk menyimpan kredensial yang dihasilkan. Untuk alasan keamanan, kata sandi yang dihasilkan untuk akses ke repositori CodeCommit Anda bersifat sementara, sehingga kredensial yang tersimpan di keychain berhenti bekerja setelah sekitar 15 menit. Jika Anda hanya mengakses Git dengan CodeCommit, coba hal berikut:

1. Di Terminal, jalankan perintah git config untuk menemukan file konfigurasi Git (gitconfig) dimana utilitas Keychain Access didefinisikan. Tergantung pada sistem dan preferensi lokal Anda, Anda mungkin memiliki lebih dari satu file gitconfig.

```
git config -l --show-origin | grep credential
```

Dalam output dari perintah ini, mencari hasil yang mirip dengan:

```
file:./path/to/gitconfig credential.helper=osxkeychain
```

File yang tercantum di awal baris ini adalah file konfigurasi Git yang harus Anda edit.

2. Untuk mengedit file konfigurasi Git, gunakan editor teks biasa atau jalankan perintah berikut:

```
nano /usr/local/git/etc/gitconfig
```

3. Modifikasi konfigurasi menggunakan salah satu strategi berikut:

- Komentari atau hapus bagian kredensial yang berisi helper = osxkeychain. Misalnya:

```
# helper = osxkeychain
```

- Perbarui bagian pembantu kredensial aws credential helper dan osxkeychain untuk memiliki konteks. Misalnya, jika osxkeychain digunakan untuk mengotentikasi ke GitHub:

```
[credential "https://git-codecommit.us-east-1.amazonaws.com"]
  helper = !aws --profile CodeCommitProfile codecommit credential-helper $@
  UseHttpPath = true
[credential "https://github.com"]
  helper = osxkeychain
```

Dalam konfigurasi ini, Git akan menggunakan pembantu osxkeychain ketika host remote cocok dengan "https://github.com" dan pembantu kredensial saat host remote cocok dengan "https://git-codecommit.us-east-1.amazonaws.com".

- Sertakan pembantu string kosong sebelum pembantu kredensial. Misalnya:

```
[credential]
  helper =
  helper = !aws --profile CodeCommitProfile codecommit credential-helper $@
```

```
UseHttpPath = true
```

Atau, jika Anda ingin terus menggunakan utilitas Keychain Access untuk cache kredensial untuk repositori Git lainnya, modifikasi header bukannya mengomentari keluar garis. Misalnya, untuk mengizinkan kredensial cache untuk GitHub, Anda dapat memodifikasi header sebagai berikut:

```
[credential "https://github.com"]  
  helper = osxkeychain
```

Jika Anda mengakses repositori lain dengan Git, Anda dapat mengkonfigurasi utilitas Keychain Access sehingga tidak menyediakan kredensial untuk repositori CodeCommit Anda. Untuk mengkonfigurasi utilitas Keychain Access:

1. Buka utilitas Keychain Access. (Anda dapat menggunakan Finder untuk menemukannya.)
2. Cari `git-codecommit.us-east-2.amazonaws.com` dan ganti `us-east-2` dengan Wilayah AWS dimana repositori berada. Sorot baris, buka menu konteks (klik kanan), lalu pilih Dapatkan Info.
3. Pilih tab Kontrol akses.
4. Dalam Konfirmasi sebelum mengizinkan akses, pilih `git-credential-osxkeychain`, lalu pilih tanda minus untuk menghapusnya dari daftar.

Note

Setelah menghapus `git-credential-osxkeychain` dari daftar, Anda melihat kotak dialog setiap kali Anda menjalankan perintah Git. Pilih Tolak untuk melanjutkan. Jika menurut Anda pop-up terlalu mengganggu, berikut adalah beberapa alternatif:

- Connect ke CodeCommit menggunakan SSH atau kredensial Git bukan pembantu kredensial dengan HTTPS. Untuk informasi selengkapnya, lihat [Untuk koneksi SSH di Linux, macOS, atau Unix](#) dan [Pengaturan untuk pengguna HTTPS menggunakan kredensial Git](#).
- Dalam utilitas Keychain Access, pada tab Kontrol Akses untuk `git-codecommit.us-east-2.amazonaws.com`, pilih opsi Izinkan semua aplikasi mengakses item ini (akses ke item ini tidak dibatasi). Hal ini mencegah pop-up, tetapi kredensialnya akhirnya berakhir (rata-rata, ini memakan waktu sekitar 15 menit) dan

Anda kemudian melihat pesan kesalahan 403. Jika ini terjadi, Anda harus menghapus item keychain untuk memulihkan fungsionalitas.

- Menginstal versi Git yang tidak menggunakan keychain secara default.
- Pertimbangkan solusi penulisan untuk menghapus item keychain. Untuk melihat sampel solusi naskah yang dibuat oleh komunitas, lihat [Script Mac OS X untuk menghapus kredensial dalam cache secara berkala di Penyimpanan Sertifikat OS X di Integrasi produk dan layanan](#).

Jika Anda ingin menghentikan Git dari menggunakan utilitas Keychain Access sepenuhnya, Anda dapat mengkonfigurasi Git untuk berhenti menggunakan osxkeychain sebagai pembantu kredensial. Misalnya, jika Anda membuka terminal dan menjalankan perintah `git config --system credential.helper`, dan menjawab `osxkeychain`, Git diatur untuk menggunakan utilitas Keychain Access. Anda dapat mengganti ini dengan menjalankan perintah berikut:

```
git config --system --unset credential.helper
```

Ketahui bahwa dengan menjalankan perintah ini dengan opsi `--system` mengubah perilaku Git seluruh sistem untuk semua pengguna, dan ini mungkin memiliki konsekuensi yang tidak diinginkan bagi pengguna lain, atau untuk repositori lain jika Anda menggunakan layanan repositori lain selain CodeCommit. Juga diketahui bahwa pendekatan ini mungkin memerlukan penggunaan `sudo`, dan akun Anda mungkin tidak memiliki izin sistem yang memadai untuk menerapkan perubahan ini. Pastikan untuk memverifikasi bahwa perintah diterapkan berhasil dengan menjalankan perintah `git config --system credential.helper` lagi. Untuk informasi selengkapnya, lihat [Kustomisasi Git - Konfigurasi Git](#) dan [artikel ini di Stack Overflow](#).

Git untuk Windows: Saya menginstal Git untuk Windows, tapi akses saya ke repositori saya ditolak (403)

Masalah: Pada Windows, pembantu kredensial tampaknya tidak mengakses atau menggunakan kredensial Anda seperti yang diharapkan. Hal ini dapat disebabkan oleh masalah yang berbeda:

- AWS CLI dikonfigurasi untuk Wilayah AWS berbeda dari tempat repositori berada.
- Secara default, Git untuk Windows menginstal utilitas Git Credential Manager yang tidak kompatibel dengan koneksi CodeCommit yang menggunakan pembantu kredensial AWS. Ketika diinstal, ini menyebabkan koneksi ke repositori gagal meskipun pembantu kredensial telah diinstal dengan AWS CLI dan dikonfigurasi untuk koneksi ke CodeCommit.

- Beberapa versi Git untuk Windows mungkin tidak sepenuhnya sesuai dengan [RFC 2617](#) dan [RFC 4559](#), yang berpotensi menyebabkan masalah dengan kredensial Git dan pembantu kredensialnya disertakan dengan AWS CLI. Untuk informasi lebih lanjut, lihat [Versi 2.11.0\(3\) tidak meminta nama pengguna/kata sandi](#).


Perbaiki yang dimungkinkan:

- Jika Anda mencoba untuk menggunakan pembantu kredensial yang disertakan dengan AWS CLI, pertimbangkan menghubungkan dengan kredensial Git melalui HTTPS bukan menggunakan pembantu kredensial. Git credentials yang dikonfigurasi untuk pengguna IAM Anda kompatibel dengan Git Credential Manager untuk Windows, tidak seperti pembantu kredensial untuk AWS CodeCommit. Untuk informasi selengkapnya, lihat [Untuk pengguna HTTPS menggunakan credentials Git](#).

Jika Anda ingin menggunakan pembantu kredensial, untuk memverifikasi apakah AWS CLI dikonfigurasi untuk Wilayah AWS yang benar, jalankan perintah `aws configure`, dan tinjau informasi yang ditampilkan. Jika repositori CodeCommit berada dalam Wilayah AWS berbeda dari yang ditunjukkan untuk AWS CLI, Anda harus menjalankan perintah `aws configure` dan mengubah nilai yang sesuai untuk Wilayah tersebut. Untuk informasi selengkapnya, lihat [Langkah 1: Konfigurasi awal untuk CodeCommit](#).

- Jika memungkinkan, uninstall dan instal ulang Git untuk Windows. Ketika Anda menginstal Git untuk Windows, kosongkan kotak centang untuk opsi untuk menginstal utilitas Git Credential Manager. Credential manager ini tidak kompatibel dengan pembantu kredensial untuk AWS CodeCommit. Jika Anda menginstal Git Credential Manager atau utilitas manajemen kredensial lain dan Anda tidak ingin meng-uninstal-nya, Anda dapat memodifikasi file `.gitconfig` Anda dan menambahkan manajemen kredensial untuk CodeCommit:

1. Buka Panel Kontrol, pilih Credential Manager, dan hapus kredensial yang disimpan untuk CodeCommit.
2. Buka file `.gitconfig` Anda dalam editor teks biasa, seperti Notepad.

 Note

Jika Anda bekerja dengan beberapa profil Git, Anda mungkin memiliki file `.gitconfig` lokal dan global. Pastikan untuk mengedit file yang sesuai.

3. Tambahkan bagian berikut ke file `.gitconfig` Anda:

```
[credential "https://git-codecommit.*.amazonaws.com"]  
  helper = !aws codecommit credential-helper $@  
  UseHttpPath = true
```

4. Simpan file, dan kemudian buka sesi baris perintah baru sebelum Anda mencoba untuk terhubung lagi.

Anda juga dapat menggunakan pendekatan ini jika Anda ingin menggunakan pembantu kredensial untuk AWS CodeCommit saat Anda terhubung ke repositori CodeCommit dan sistem manajemen kredensial lainnya saat Anda terhubung ke repositori lain yang dihosting, seperti repositori GitHub.

Untuk me-reset pembantu kredensial mana yang digunakan sebagai default, Anda dapat menggunakan opsi `--system` bukan `--global` atau `--local` ketika Anda menjalankan perintah `git config`.

- Jika Anda menggunakan kredensial Git pada komputer Windows, Anda dapat mencoba memecahkan setiap masalah ketidaksesuaian RFC dengan memasukkan nama pengguna kredensial Git Anda sebagai bagian dari string sambungan. Misalnya, untuk mengatasi masalah dan mengkloning repositori bernama *MyDemoRepo* di US East (Ohio):

```
git clone https://Your-Git-Credential-Username@git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

Note

Pendekatan ini tidak bekerja jika Anda memiliki karakter `@` dalam nama pengguna kredensial Git Anda. Anda harus mengkodekan URL (juga dikenal sebagai pelolosan URL atau [Pengkodean persen](#)) karakter.

Mengatasi masalah klien Git dan AWS CodeCommit

Informasi berikut dapat membantu Anda memecahkan masalah umum yang terjadi saat menggunakan Git dengan repositori AWS CodeCommit. Untuk mengatasi masalah yang terkait dengan klien Git saat menggunakan HTTPS atau SSH, lihat juga [Mengatasi masalah kredensial Git \(HTTPS\)](#), [Pemecahan masalah koneksi SSH](#), dan [Pemecahan masalah pembantu kredensial \(HTTPS\)](#).

Topik

- [Kesalahan Git: Kesalahan: RPC gagal; hasil = 56, kode HTTP = 200 fatal: Ujung terpencil menutup telepon secara tak terduga](#)
- [Kesalahan Git: Terlalu banyak perintah pembaruan referensi](#)
- [Kesalahan Git: Dorong melalui HTTPS rusak di beberapa versi Git](#)
- [Kesalahan Git: 'gnutls_handshake \(\) gagal'](#)
- [Kesalahan Git: Git tidak dapat menemukan repositori CodeCommit atau tidak memiliki izin untuk mengakses repositori](#)
- [Git pada Windows: Tidak tersedia metode autentikasi yang didukung \(publickey\)](#)

Kesalahan Git: Kesalahan: RPC gagal; hasil = 56, kode HTTP = 200 fatal: Ujung terpencil menutup telepon secara tak terduga

Masalah: Ketika melakukan push perubahan besar, sejumlah besar perubahan, atau repositori besar, koneksi HTTPS yang berjalan lama sering dihentikan sebelum waktunya karena masalah jaringan atau pengaturan firewall.

Perbaikan yang dimungkinkan: Dorong dengan SSH sebagai gantinya, atau saat Anda memigrasi repositori besar, ikuti langkah-langkah di [Memigrasi repositori secara bertahap](#). Selain itu, pastikan juga Anda tidak melebihi batas ukuran untuk masing-masing file. Untuk informasi selengkapnya, lihat [Kuota](#).

Kesalahan Git: Terlalu banyak perintah pembaruan referensi

Masalah: Jumlah maksimum referensi pembaruan per push adalah 4.000. Kesalahan ini muncul ketika push berisi lebih dari 4.000 pembaruan referensi.

Perbaikan yang dimungkinkan: Cobalah mendorong cabang dan tag secara individual dengan `git push --all` dan `git push --tags`. Jika Anda memiliki terlalu banyak tag, pisahkan tag menjadi beberapa push. Untuk informasi selengkapnya, lihat [Kuota](#).

Kesalahan Git: Dorong melalui HTTPS rusak di beberapa versi Git

Masalah: Masalah dengan pembaruan curl ke 7.41.0 menyebabkan autentikasi berbasis SSPI gagal. Versi Git terpengaruh yang dikenal termasuk 1.9.5.msysgit.1. Beberapa versi Git untuk Windows mungkin tidak sepenuhnya sesuai dengan [RFC 2617](#) dan [RFC 4559](#), yang berpotensi

menyebabkan masalah dengan koneksi HTTPS yang menggunakan kredensial Git atau yang penolong kredensialnya disertakan dengan AWS CLI.

Perbaikan yang dimungkinkan: Periksa versi Git Anda untuk masalah yang diketahui atau gunakan versi sebelumnya atau yang lebih baru. Untuk informasi selengkapnya tentang `mysysgit`, lihat [Push ke HTTPS Rusak](#) di forum GitHub. Untuk informasi lebih lanjut tentang masalah versi Git untuk Windows, lihat [Versi 2.11.0\(3\) tidak meminta nama pengguna/kata sandi](#).

Kesalahan Git: 'gnutls_handshake () gagal'

Masalah: Di Linux, ketika Anda mencoba menggunakan Git untuk berkomunikasi dengan repositori CodeCommit, akan muncul pesan kesalahan yang berisi kalimat `error: gnutls_handshake() failed`.

Perbaikan yang dimungkinkan: Kompilasi Git terhadap OpenSSL. Untuk satu pendekatan, lihat ["Kesalahan: gnutls_handshake\(\) gagal" Saat Terhubung ke Server HTTPS](#) di forum Ask Ubuntu.

Atau, gunakan SSH, bukan HTTPS, untuk berkomunikasi dengan repositori CodeCommit.

Kesalahan Git: Git tidak dapat menemukan repositori CodeCommit atau tidak memiliki izin untuk mengakses repositori

Masalah: Sebuah garis miring pada string koneksi dapat menyebabkan upaya koneksi gagal.

Perbaikan yang dimungkinkan: Pastikan bahwa Anda telah memberikan nama dan string koneksi yang benar untuk repositori, dan bahwa tidak ada garis miring di belakang. Untuk informasi selengkapnya, lihat [Connect ke sebuah repositori](#).

Git pada Windows: Tidak tersedia metode autentikasi yang didukung (publickey)

Masalah: Setelah Anda mengkonfigurasi akses SSH untuk Windows, Anda melihat kesalahan akses ditolak ketika Anda mencoba untuk menggunakan perintah seperti `git pull`, `git push`, atau `git clone`.

Perbaikan yang dimungkinkan: Penyebab paling umum untuk kesalahan ini adalah bahwa variabel lingkungan `GIT_SSH` ada di komputer Anda dan dikonfigurasi untuk mendukung utilitas koneksi lain, PuTTY misalnya. Untuk memperbaiki masalah ini, cobalah lakukan salah satu hal berikut:

- Buka emulator Bash dan tambahkan parameter `GIT_SSH_COMMAND="ssh"` sebelum perintah Git. Sebagai contoh, jika Anda mencoba untuk melakukan kloning repositori, jangan menjalankan `git`

clone `ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo`, tetapi jalankan:

```
GIT_SSH_COMMAND="ssh" git clone ssh://git-codecommit.us-east-2.amazonaws.com/v1/repos/MyDemoRepo my-demo-repo
```

- Ubah nama atau hapus `GIT_SSH` jika Anda tidak lagi menggunakannya. Kemudian buka command prompt baru atau sesi emulator Bash, dan coba jalankan perintah Anda lagi.

Untuk informasi selengkapnya tentang pemecahan masalah Git pada Windows saat menggunakan SSH, lihat [Pemecahan masalah koneksi SSH](#).

Mengatasi masalah kesalahan akses dan AWS CodeCommit

Informasi berikut dapat membantu Anda memecahkan masalah kesalahan akses saat terhubung dengan repositori AWS CodeCommit.

Topik

- [Kesalahan akses: Saya diminta untuk memasukkan nama pengguna dan sandi ketika saya terhubung ke sebuah repositori CodeCommit dari Windows](#)
- [Kesalahan akses: Kunci publik ditolak saat menghubungkan ke sebuah repositori CodeCommit](#)
- [Kesalahan akses: Pesan “Tingkat Terlampaui” atau “429” saat menghubungkan ke sebuah repositori CodeCommit](#)

Kesalahan akses: Saya diminta untuk memasukkan nama pengguna dan sandi ketika saya terhubung ke sebuah repositori CodeCommit dari Windows

Masalah: Ketika Anda mencoba menggunakan Git untuk berkomunikasi dengan sebuah repositori CodeCommit, Anda melihat kotak dialog yang meminta nama pengguna dan sandi Anda.

Perbaiki yang dimungkinkan: Hal ini mungkin sistem manajemen kredensial bawaan untuk Windows. Tergantung konfigurasi Anda, lakukan salah satu hal berikut ini:

- Jika Anda menggunakan HTTPS dengan kredensial Git, berarti kredensial Git Anda belum disimpan dalam sistem. Berikan kredensial Git dan lanjutkan. Anda tidak akan diminta lagi. Untuk informasi selengkapnya, lihat [Untuk pengguna HTTPS menggunakan kredensial Git](#).

- Jika Anda menggunakan HTTPS dengan pembantu kredensial untuk AWS CodeCommit, ia tidak kompatibel dengan sistem manajemen kredensial Windows. Pilih Batalkan.

Hal ini mungkin juga merupakan indikasi bahwa Anda menginstal Pengelola Kredensial Git ketika Anda menginstal Git untuk Windows. Pengelola Kredensial Git ini tidak kompatibel dengan penolong kredensial untuk CodeCommit yang disertakan dalam AWS CLI. Pertimbangkan untuk menghapus Pengelola Kredensial Git. Anda juga dapat menginstal dan mengkonfigurasi `git-remote-codecommit` sebagai alternatif untuk menggunakan penolong kredensial untuk CodeCommit.

Untuk informasi lebih lanjut, lihat [Langkah-langkah penyiapan untuk koneksi HTTPS keAWS CodeCommitbersamagit-remote-codecommit](#), [Untuk sambungan HTTPS pada Windows dengan pembantu kredensial AWS CLI](#), dan [Git untuk Windows: Saya menginstal Git untuk Windows, tapi akses saya ke repositori saya ditolak \(403\)](#).

Kesalahan akses: Kunci publik ditolak saat menghubungkan ke sebuah repositori CodeCommit

Masalah: Ketika Anda mencoba menggunakan titik akhir SSH untuk berkomunikasi dengan sebuah repositori CodeCommit, akan muncul pesan kesalahan yang berisi frasa `Error: public key denied`.

Perbaiki yang dimungkinkan: Alasan paling umum untuk kesalahan ini adalah bahwa Anda belum menyelesaikan setup untuk koneksi SSH. Mengkonfigurasi pasangan kunci SSH publik dan privat, dan kemudian mengaitkan kunci publik dengan pengguna IAM Anda. Untuk informasi lebih lanjut tentang mengkonfigurasi SSH, lihat [Untuk koneksi SSH di Linux, macOS, atau Unix](#) dan [Untuk koneksi SSH pada Windows](#).

Kesalahan akses: Pesan “Tingkat Terlampaui” atau “429” saat menghubungkan ke sebuah repositori CodeCommit

Masalah: Ketika Anda mencoba untuk berkomunikasi dengan sebuah repositori CodeCommit, muncul pesan yang mengatakan “Tingkat Terlampaui” atau dengan kode kesalahan “429”. Komunikasi menjadi lambat secara signifikan atau gagal.

Penyebab: Semua panggilan ke CodeCommit, apakah dari aplikasi, AWS CLI, klien Git, atau AWS Management Console, tunduk pada jumlah maksimum permintaan per detik dan permintaan aktif

secara keseluruhan. Anda tidak dapat melebihi tingkat permintaan maksimum yang diizinkan untuk akun Amazon Web Services di Wilayah AWS. Jika permintaan melebihi tingkat maksimum, maka Anda akan menerima kesalahan dan panggilan lebih lanjut sementara akan diperlambat untuk akun Amazon Web Services Anda. Selama periode throttling, koneksi Anda ke CodeCommit akan diperlambat dan mungkin gagal.

Perbaikan yang dimungkinkan: Ambil langkah-langkah untuk mengurangi jumlah koneksi atau panggilan ke CodeCommit atau untuk menyebarkan permintaan. Beberapa pendekatan yang bisa dipertimbangkan:

- Menerapkan jitter dalam permintaan, khususnya dalam permintaan pengumpulan secara berkala

Jika Anda memiliki aplikasi yang melakukan pengumpulan CodeCommit secara berkala dan aplikasi ini berjalan pada beberapa instans Amazon EC2, perkenalkan jitter (jumlah acak penundaan) sehingga instans Amazon EC2 yang berbeda tidak melakukan pengumpulan pada detik yang sama. Kami merekomendasikan nomor acak dari 0 sampai 59 detik untuk secara merata mendistribusikan mekanisme pengumpulan dalam jangka waktu satu menit.

- Gunakan arsitektur berbasis peristiwa, bukan pengumpulan

Daripada pengumpulan, gunakan arsitektur berbasis peristiwa sehingga panggilan hanya dibuat ketika suatu peristiwa terjadi. Pertimbangkan untuk menggunakan notifikasi CloudWatch Events untuk [peristiwa AWS CodeCommit](#) untuk memicu alur kerja Anda.

- Menerapkan percobaan ulang kesalahan dan backoff eksponensial untuk API dan tindakan Git otomatis

Percobaan ulang kesalahan dan backoff eksponensial dapat membantu membatasi laju panggilan. Masing-masing SDK AWS menerapkan logika coba ulang otomatis dan algoritme backoff eksponensial. Untuk push Git otomatis dan pull Git, Anda mungkin perlu menerapkan logika percobaan ulang Anda sendiri. Untuk informasi selengkapnya, lihat [Percobaan Ulang Kesalahan dan Backoff Eksponensial di AWS](#).

- Meminta peningkatan kuota layanan CodeCommitAWS Pusat Dukungan

Untuk menerima peningkatan kuota layanan, Anda harus mengonfirmasi bahwa Anda telah mengikuti saran yang diberikan di sini, termasuk penerapan pengulangan kesalahan atau metode backoff eksponensial. Dalam permintaan Anda, Anda juga harus menyediakan Wilayah AWS, akun Amazon Web Services, dan jangka waktu yang dipengaruhi oleh masalah throttling.

Mengatasi masalah kesalahan konfigurasi dan AWS CodeCommit

Informasi berikut dapat membantu Anda memecahkan masalah kesalahan konfigurasi yang mungkin Anda alami saat terhubung dengan repositori AWS CodeCommit.

Topik

- [Kesalahan konfigurasi: Tidak dapat mengonfigurasi AWS CLI kredensi pada macOS](#)

Kesalahan konfigurasi: Tidak dapat mengonfigurasi AWS CLI kredensi pada macOS

Masalah: Ketika Anda menjalankan `aws configure` untuk mengonfigurasi AWS CLI, Anda melihat `ConfigParseError` pesan.

Perbaikan yang dimungkinkan: Penyebab paling umum untuk kesalahan ini adalah bahwa file kredensi sudah ada. Jelajahi `~/.aws` dan cari file bernama `credentials`. Ubah nama atau hapus file tersebut, dan kemudian jalankan `aws configure` sekali lagi.

Mengatasi masalah kesalahan konsol dan AWS CodeCommit

Informasi berikut dapat membantu Anda memecahkan masalah kesalahan konsol saat menggunakan repositori AWS CodeCommit.

Topik

- [Kesalahan akses: Akses kunci enkripsi ditolak untuk CodeCommit repositori dari konsol atau AWS CLI](#)
- [Kesalahan enkripsi: Repositori tidak dapat didekripsi](#)
- [Kesalahan konsol: Tidak dapat menelusuri kode di CodeCommit repositori dari konsol](#)
- [Kesalahan tampilan: Tidak dapat melihat file atau perbandingan antar file](#)

Kesalahan akses: Akses kunci enkripsi ditolak untuk CodeCommit repositori dari konsol atau AWS CLI

Masalah: Ketika Anda mencoba mengakses CodeCommit dari konsol atau AWS CLI, pesan kesalahan muncul berisi frasa `EncryptionKeyAccessDeniedException` atau `User is not`

authorized for the KMS default key for CodeCommit 'aws/codecommit' in your account.

Kemungkinan perbaikan: Penyebab paling umum untuk kesalahan ini adalah akun Amazon Web Services Anda tidak berlangganan AWS Key Management Service, yang diperlukan untuk CodeCommit. Buka konsol AWS KMS, pilih kunci terkelola AWS, lalu pilih Memulai Sekarang. Jika Anda melihat pesan bahwa Anda saat ini tidak berlangganan AWS Key Management Service, ikuti petunjuk di halaman tersebut untuk berlangganan. Untuk informasi lebih lanjut tentang CodeCommit dan AWS Key Management Service, lihat [AWS KMS dan enkripsi](#).

Kesalahan enkripsi: Repositori tidak dapat didekripsi

Masalah: Ketika Anda mencoba mengakses CodeCommit repositori dari konsol atau AWS CLI, pesan kesalahan muncul berisi frasa. Repository can't be decrypted

Kemungkinan perbaikan: Penyebab paling umum untuk kesalahan ini adalah bahwa AWS KMS kunci yang digunakan untuk mengenkripsi dan mendekripsi data untuk repositori ini tidak aktif atau tertunda penghapusan. Kunci aktif Kunci yang dikelola AWS atau terkelola pelanggan AWS Key Management Service diperlukan untuk CodeCommit. Buka AWS KMS konsol, pilih Kunci yang dikelola AWS atau Kunci yang dikelola Pelanggan, dan pastikan bahwa kunci yang digunakan untuk repositori ada di Wilayah AWS tempat repositori ada dan statusnya Aktif. Untuk informasi lebih lanjut tentang CodeCommit dan AWS Key Management Service, lihat [AWS KMS dan enkripsi](#).

Important

Jika kunci yang digunakan untuk mengenkripsi dan mendekripsi data untuk repositori telah dihapus secara permanen atau tidak dapat diakses, data dalam repositori yang dienkripsi dengan kunci itu tidak dapat diakses.

Kesalahan konsol: Tidak dapat menelusuri kode di CodeCommit repositori dari konsol

Masalah: Ketika Anda mencoba untuk menelusuri isi repositori dari konsol, ada pesan kesalahan muncul yang menyatakan akses ditolak.

Kemungkinan perbaikan: Penyebab paling umum untuk kesalahan ini adalah kebijakan IAM yang diterapkan ke akun Amazon Web Services Anda menolak satu atau beberapa izin yang diperlukan

untuk kode penelusuran dari konsol. CodeCommit Untuk informasi selengkapnya tentang izin CodeCommit akses dan penelusuran, lihat [Kontrol autentikasi dan akses untuk AWS CodeCommit](#).

Kesalahan tampilan: Tidak dapat melihat file atau perbandingan antar file

Masalah: Saat Anda mencoba melihat file atau perbandingan antara dua versi file di CodeCommit konsol, muncul kesalahan yang menyatakan bahwa file atau perbedaan terlalu besar untuk ditampilkan.

Kemungkinan perbaikan: Penyebab paling umum untuk kesalahan ini adalah bahwa file terlalu besar untuk ditampilkan, berisi satu atau lebih baris yang melebihi batas karakter untuk satu baris dalam file, atau perbedaan antara dua versi file melebihi batas baris. Untuk informasi selengkapnya, lihat [Kuota](#). Untuk melihat file atau perbedaan antara versi file, Anda dapat membuka file secara lokal di IDE pilihan Anda, menggunakan alat diff Git, atau menjalankan perintah. `git diff`

Pemecahan masalah pemicu dan AWS CodeCommit

Informasi berikut dapat membantu Anda memecahkan masalah dengan pemicu di AWS CodeCommit.

Topik

- [Kesalahan pemicu: Pemicu repositori tidak berjalan ketika diharapkan](#)

Kesalahan pemicu: Pemicu repositori tidak berjalan ketika diharapkan

Masalah: Satu atau beberapa pemicu yang dikonfigurasi untuk sebuah repositori tampaknya tidak berjalan atau tidak berjalan seperti yang diharapkan.

Perbaikan yang dimungkinkan: Jika target pemicu adalah AWS Lambda fungsi, pastikan Anda telah mengkonfigurasi kebijakan sumber daya fungsi untuk akses oleh CodeCommit. Untuk informasi selengkapnya, lihat [Contoh 3: Membuat kebijakan untuk AWS Lambda integrasi dengan CodeCommit pemicu](#).

Atau, edit pemicu dan pastikan peristiwa yang untuknya Anda ingin memicu tindakan telah dipilih dan bahwa cabang untuk pemicu termasuk cabang di mana Anda ingin melihat tanggapan terhadap tindakan. Coba ubah pengaturan untuk pemicu ke Semua peristiwa repositori dan Semua cabang dan kemudian uji pemicu. Untuk informasi selengkapnya, lihat [Edit pemicu untuk sebuah repositori](#).

Mengaktifkan debugging

Masalah: Saya ingin mengaktifkan debugging untuk mendapatkan informasi lebih lanjut tentang repositori saya dan bagaimana Git mengeksekusi perintah.

Perbaiki yang mungkin: Cobalah cara berikut ini:

1. Pada terminal atau command prompt, jalankan perintah berikut pada mesin lokal Anda sebelum menjalankan perintah Git:

Di Linux, macOS, atau Unix:

```
export GIT_TRACE_PACKET=1
export GIT_TRACE=1
export GIT_CURL_VERBOSE=1
```

Di Windows:

```
set GIT_TRACE_PACKET=1
set GIT_TRACE=1
set GIT_CURL_VERBOSE=1
```

Note

Pengaturan `GIT_CURL_VERBOSE` berguna untuk koneksi HTTPS saja. SSH tidak menggunakan perpustakaan `libcurl`.

2. [Untuk mendapatkan informasi lebih lanjut tentang repositori Git Anda, kami sarankan menginstal versi terbaru dari git-sizer.](#) Ikuti instruksi untuk memasukkan utilitas yang sesuai dengan sistem operasi dan lingkungan Anda. Setelah diinstal, di baris perintah atau terminal, ubah direktori ke repositori lokal Anda dan kemudian jalankan perintah berikut:

```
git-sizer --verbose
```


 Tip

Pertimbangkan untuk menyimpan output perintah ke file sehingga Anda dapat dengan mudah membagikannya dengan orang lain saat memecahkan masalah, terutama seiring waktu.

AWS CodeCommit referensi

Topik referensi berikut dapat membantu Anda lebih memahami CodeCommit, Git, Wilayah AWS, batas layanan, dan banyak lagi.

Topik

- [Wilayah dan titik akhir koneksi Git untuk AWS CodeCommit](#)
- [Menggunakan AWS CodeCommit dengan antarmuka VPC endpoint](#)
- [Kuota di AWS CodeCommit](#)
- [AWS CodeCommit referensi baris perintah](#)
- [Perintah Git Basic](#)

Wilayah dan titik akhir koneksi Git untuk AWS CodeCommit

Setiap CodeCommit repositori dikaitkan dengan file. Wilayah AWS CodeCommit menawarkan titik akhir regional untuk mengajukan permintaan Anda ke layanan. Selain itu, CodeCommit menyediakan titik akhir koneksi Git untuk protokol SSH dan HTTPS di setiap Wilayah yang tersedia. CodeCommit

Semua contoh dalam panduan ini menggunakan URL titik akhir yang sama untuk Git di US East (Ohio): `git-codecommit.us-east-2.amazonaws.com`. Namun, ketika Anda menggunakan Git dan mengonfigurasi koneksi Anda, pastikan Anda memilih titik akhir koneksi Git yang cocok dengan Wilayah AWS yang menghosting CodeCommit repositori Anda. Misalnya, jika Anda ingin membuat koneksi ke repositori di US East (N. Virginia), gunakan URL titik akhir `git-codecommit.us-east-1.amazonaws.com`. Ini juga berlaku untuk panggilan API. Ketika Anda membuat koneksi ke CodeCommit repositori dengan AWS CLI atau SDK, pastikan Anda menggunakan endpoint regional yang benar untuk repositori.

Topik

- [Didukung Wilayah AWS untuk CodeCommit](#)
- [Titik akhir koneksi Git](#)
- [Sidik jari server untuk CodeCommit](#)

Didukung Wilayah AWS untuk CodeCommit

Anda dapat membuat dan menggunakan CodeCommit repositori sebagai berikut: Wilayah AWS

- AS Timur (Ohio)
- AS Timur (Virginia Utara)
- AS Barat (California Utara)
- US West (Oregon)
- Europe (Ireland)
- Eropa (London)
- Europe (Paris)
- Europe (Frankfurt)
- Eropa (Stockholm)
- Europe (Milan)
- Afrika (Cape Town)
- Israel (Tel Aviv)
- Asia Pasifik (Tokyo)
- Asia Pasifik (Singapura)
- Asia Pasifik (Sydney)
- Asia Pasifik (Jakarta)
- Timur Tengah (UEA)
- Asia Pasifik (Seoul)
- Asia Pasifik (Osaka)
- Asia Pasifik (Mumbai)
- Asia Pasifik (Hyderabad)
- Asia Pasifik (Hong Kong)
- South America (Sao Paulo)
- Middle East (Bahrain)
- Canada (Central)
- Tiongkok (Beijing)
- China (Ningxia)
- AWS GovCloud (AS-Barat)
- AWS GovCloud (AS-Timur)

CodeCommit telah menambahkan dukungan untuk standar pemerintah Federal Information Processing Standard (FIPS) Publikasi 140-2 di beberapa daerah. Untuk informasi lebih lanjut tentang titik akhir FIPS yang tersedia, lihat [Gambaran umum Standar Pengolahan Informasi Federal \(FIPS\) 140-2](#). Untuk titik akhir koneksi Git yang mendukung FIPS, lihat [Titik akhir koneksi Git](#).

Untuk informasi selengkapnya tentang titik akhir regional untuk AWS CLI, layanan, dan panggilan API CodeCommit, lihat [AWS CodeCommit Titik Akhir dan Kuota](#).

Titik akhir koneksi Git

Gunakan URL berikut saat Anda mengonfigurasi koneksi Git ke CodeCommit repositori:

Titik akhir koneksi Git untuk AWS CodeCommit

Nama Wilayah	Wilayah	URL Titik Akhir	Protokol
US East (Ohio)	us-east-2	<code>https://git-codecommit.us-east-2.amazonaws.com</code>	HTTPS
US East (Ohio)	us-east-2	<code>ssh://git-codecommit.us-east-2.amazonaws.com</code>	SSH
AS Timur (Ohio)	us-east-2	<code>https://git-codecommit-fips.us-east-2.amazonaws.com</code>	HTTPS
US East (N. Virginia)	us-east-1	<code>https://git-codecommit.us-east-1.amazonaws.com</code>	HTTPS
US East (N. Virginia)	us-east-1	<code>ssh://git-codecommit.us-east-1.amazonaws.com</code>	SSH
US East (Northern Virginia)	us-east-1	<code>https://git-codecommit-fips.us-east-1.amazonaws.com</code>	HTTPS

Nama Wilayah	Wilayah	URL Titik Akhir	Protokol
US West (Oregon)	us-west-2	https://git-codecommit.us-west-2.amazonaws.com	HTTPS
US West (Oregon)	us-west-2	ssh://git-codecommit.us-west-2.amazonaws.com	SSH
AS Barat (Oregon)	us-west-2	https://git-codecommit-fips.us-west-2.amazonaws.com	HTTPS
US West (N. California)	us-west-1	https://git-codecommit.us-west-1.amazonaws.com	HTTPS
US West (N. California)	us-west-1	ssh://git-codecommit.us-west-1.amazonaws.com	SSH
US West (Northern California)	us-west-1	https://git-codecommit-fips.us-west-1.amazonaws.com	HTTPS
Europe (Ireland)	eu-west-1	https://git-codecommit.eu-west-1.amazonaws.com	HTTPS
Europe (Ireland)	eu-west-1	ssh://git-codecommit.eu-west-1.amazonaws.com	SSH
Asia Pasifik (Tokyo)	ap-northeast-1	https://git-codecommit.ap-northeast-1.amazonaws.com	HTTPS

Nama Wilayah	Wilayah	URL Titik Akhir	Protokol
Asia Pacific (Tokyo)	ap-northeast-1	ssh://git-codecommit.ap-northeast-1.amazonaws.com	SSH
Asia Pacific (Singapore)	ap-southeast-1	https://git-codecommit.ap-southeast-1.amazonaws.com	HTTPS
Asia Pacific (Singapore)	ap-southeast-1	https://git-codecommit.ap-southeast-1.amazonaws.com	SSH
Asia Pasifik (Sydney)	ap-southeast-2	https://git-codecommit.ap-southeast-2.amazonaws.com	HTTPS
Asia Pacific (Sydney)	ap-southeast-2	ssh://git-codecommit.ap-southeast-2.amazonaws.com	SSH
Asia Pasifik (Jakarta)	ap-southeast-3	https://git-codecommit.ap-southeast-3.amazonaws.com	HTTPS
Asia Pasifik (Jakarta)	ap-southeast-3	ssh://git-codecommit.ap-southeast-3.amazonaws.com	SSH
Timur Tengah (UEA)	me-central-1	https://git-codecommit.me-central-1.amazonaws.com	HTTPS
Timur Tengah (UEA)	me-central-1	ssh://git-codecommit.me-central-1.amazonaws.com	SSH

Nama Wilayah	Wilayah	URL Titik Akhir	Protokol
Europa (Frankfurt)	eu-central-1	<code>https://git-codecommit.eu-central-1.amazonaws.com</code>	HTTPS
Europe (Frankfurt)	eu-central-1	<code>ssh://git-codecommit.eu-central-1.amazonaws.com</code>	SSH
Asia Pasifik (Seoul)	ap-northeast-2	<code>https://git-codecommit.ap-northeast-2.amazonaws.com</code>	HTTPS
Asia Pacific (Seoul)	ap-northeast-2	<code>ssh://git-codecommit.ap-northeast-2.amazonaws.com</code>	SSH
Amerika Selatan (Sao Paulo)	sa-east-1	<code>https://git-codecommit.sa-east-1.amazonaws.com</code>	HTTPS
South America (São Paulo)	sa-east-1	<code>ssh://git-codecommit.sa-east-1.amazonaws.com</code>	SSH
Europa (London)	eu-west-2	<code>https://git-codecommit.eu-west-2.amazonaws.com</code>	HTTPS
Europe (London)	eu-west-2	<code>ssh://git-codecommit.eu-west-2.amazonaws.com</code>	SSH
Asia Pasifik (Mumbai)	ap-south-1	<code>https://git-codecommit.ap-south-1.amazonaws.com</code>	HTTPS

Nama Wilayah	Wilayah	URL Titik Akhir	Protokol
Asia Pacific (Mumbai)	ap-south-1	ssh://git-codecommit.ap-south-1.amazonaws.com	SSH
Asia Pasifik (Hyderabad)	ap-south-2	https://git-codecommit.ap-south-2.amazonaws.com	HTTPS
Asia Pasifik (Hyderabad)	ap-south-2	ssh://git-codecommit.ap-south-2.amazonaws.com	SSH
Kanada (Pusat)	ca-central-1	https://git-codecommit.ca-central-1.amazonaws.com	HTTPS
Canada (Central)	ca-central-1	ssh://git-codecommit.ca-central-1.amazonaws.com	SSH
Kanada (Pusat)	ca-central-1	https://git-codecommit-fips.ca-central-1.amazonaws.com	HTTPS
Europe (Paris)	eu-west-3	https://git-codecommit.eu-west-3.amazonaws.com	HTTPS
Europe (Paris)	eu-west-3	ssh://git-codecommit.eu-west-3.amazonaws.com	SSH
AWS GovCloud (AS- Barat)	us-gov-west-1	https://git-codecommit.us-gov-west-1.amazonaws.com	HTTPS

Nama Wilayah	Wilayah	URL Titik Akhir	Protokol
AWS GovCloud (AS-Barat)	us-gov-west-1	ssh://git-codecommit.us-gov-west-1.amazonaws.com	SSH
AWS GovCloud (AS-Barat)	us-gov-west-1	https://git-codecommit-fips.us-gov-west-1.amazonaws.com	HTTPS
AWS GovCloud (AS-Timur)	us-gov-east-1	https://git-codecommit.us-gov-east-1.amazonaws.com	HTTPS
AWS GovCloud (AS-Timur)	us-gov-east-1	ssh://git-codecommit.us-gov-east-1.amazonaws.com	SSH
AWS GovCloud (AS-Timur)	us-gov-east-1	https://git-codecommit-fips.us-gov-east-1.amazonaws.com	HTTPS
Europe (Stockholm)	eu-north-1	https://git-codecommit.eu-north-1.amazonaws.com	HTTPS
Europe (Stockholm)	eu-north-1	ssh://git-codecommit.eu-north-1.amazonaws.com	SSH
Timur Tengah (Bahrain)	me-south-1	https://git-codecommit.me-south-1.amazonaws.com	HTTPS
Middle East (Bahrain)	me-south-1	ssh://git-codecommit.me-south-1.amazonaws.com	SSH

Nama Wilayah	Wilayah	URL Titik Akhir	Protokol
Asia Pasifik (Hong Kong)	ap-east-1	<code>https://git-codecommit.ap-east-1.amazonaws.com</code>	HTTPS
Asia Pacific (Hong Kong)	ap-east-1	<code>ssh://git-codecommit.ap-east-1.amazonaws.com</code>	SSH
Tiongkok (Beijing)	cn-north-1	<code>https://git-codecommit.cn-north-1.amazonaws.com.cn</code>	HTTPS
China (Beijing)	cn-north-1	<code>ssh://git-codecommit.cn-north-1.amazonaws.com.cn</code>	SSH
Tiongkok (Ningxia)	cn-northwest-1	<code>https://git-codecommit.cn-northwest-1.amazonaws.com.cn</code>	HTTPS
China (Ningxia)	cn-northwest-1	<code>ssh://git-codecommit.cn-northwest-1.amazonaws.com.cn</code>	SSH
Eropa (Milan)	eu-south-1	<code>https://git-codecommit.eu-south-1.amazonaws.com</code>	HTTPS
Europe (Milan)	eu-south-1	<code>ssh://git-codecommit.eu-south-1.amazonaws.com</code>	SSH
Asia Pacific (Osaka)	ap-northeast-3	<code>https://git-codecommit.ap-northeast-3.amazonaws.com</code>	HTTPS

Nama Wilayah	Wilayah	URL Titik Akhir	Protokol
Asia Pacific (Osaka)	ap-northeast-3	ssh://git-codecommit.ap-northeast-3.amazonaws.com	SSH
Afrika (Cape Town)	af-south-1	https://git-codecommit.af-south-1.amazonaws.com	HTTPS
Africa (Cape Town)	af-south-1	ssh://git-codecommit.af-south-1.amazonaws.com	SSH
Israel (Tel Aviv)	il-central-1	https://git-codecommit.il-central-1.amazonaws.com	HTTPS
Israel (Tel Aviv)	il-central-1	ssh://git-codecommit.il-central-1.amazonaws.com	SSH

Sidik jari server untuk CodeCommit

Tabel berikut mencantumkan sidik jari publik untuk titik akhir koneksi Git di CodeCommit. Sidik jari server ini ditampilkan sebagai bagian dari proses verifikasi untuk menambahkan titik akhir ke file host Anda yang dikenal.

Sidik jari publik untuk CodeCommit

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.us-east-2.amazonaws.com	MD5*	a9:6d:03:ed:08:42: 21:be:06:e1:e0:2a: d1:75:31:5e

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.us-east-2.amazonaws.com	SHA256	31B1W2g5xn/NA2Ck6dyeJIrQ0Wvn7n8UEs56fG6ZIZQ
git-codecommit.us-east-1.amazonaws.com	MD5*	a6:9c:7d:bc:35:f5:d4:5f:8b:ba:6f:c8:bc:d4:83:84
git-codecommit.us-east-1.amazonaws.com	SHA256	eLMY1j0DKA4uvDZc1/KgtIayZANwX6t8+8isPtotBoY
git-codecommit.us-west-2.amazonaws.com	MD5*	a8:68:53:e3:99:ac:6e:d7:04:7e:f7:92:95:77:a9:77
git-codecommit.us-west-2.amazonaws.com	SHA256	0pJx9SQpkbPUAHwy58UVIq0IHcyo1fwCp00uVgcAWPo
git-codecommit.eu-west-1.amazonaws.com	MD5*	93:42:36:ea:22:1f:f1:0f:20:02:4a:79:ff:ea:12:1d
git-codecommit.eu-west-1.amazonaws.com	SHA256	tKjRk0L8dmJyTmSbeSdN1S8F/f0iq13R1vqgTOP1UyQ
git-codecommit.ap-northeast-1.amazonaws.com	MD5*	8e:a3:f0:80:98:48:1c:5c:6f:59:db:a7:8f:6e:c6:cb
git-codecommit.ap-northeast-1.amazonaws.com	SHA256	Xk/WeYD/K/bnBybzhiuu4dWpBJtXPf7E30jHU7se40w

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.ap-southeast-1.amazonaws.com	MD5*	65:e5:27:c3:09:68: 0d:8e:b7:6d:94:25: 80:3e:93:cf
git-codecommit.ap-southeast-1.amazonaws.com	SHA256	ZISVa70VzxrTIf+Rk4 UbhPv6Es22mSB3uTBo jfPXIno
git-codecommit.ap-southeast-2.amazonaws.com	MD5*	7b:d2:c1:24:e6:91: a5:7b:fa:c1:0c:35: 95:87:da:a0
git-codecommit.ap-southeast-2.amazonaws.com	SHA256	nYp+gHas80HY3DqbP4 yanCDFhqDVjseefVbH EXqH2Ec
git-codecommit.ap-southeast-3.amazonaws.com	MD5*	64:d9:e0:53:19:4f: a8:91:9a:c3:53:22: a6:a8:ed:a6
git-codecommit.ap-southeast-3.amazonaws.com	SHA256	ATdkGSFhpqIu7RqUVT /1RZo6MLxxxUW9NoDV MbAc/6g
git-codecommit.me-central-1.amazonaws.com	MD5*	bd:fa:e2:f9:05:84: d6:39:6f:bc:d6:8d: fe:de:61:76
git-codecommit.me-central-1.amazonaws.com	SHA256	grceUDWubo4MzG1Noa KZKUfrgPvfN3ijli0n Qr11TZA
git-codecommit.eu-central-1.amazonaws.com	MD5*	74:5a:e8:02:fc:b2: 9c:06:10:b4:78:84: 65:94:22:2d

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.eu-central-1.amazonaws.com	SHA256	MwGrkiEki8QkkBt1Ag XbYt0hoZYBnZF62VY5 RzGJEUY
git-codecommit.ap-northeast-2.amazonaws.com	MD5*	9f:68:48:9b:5f:fc: 96:69:39:45:58:87: 95:b3:69:ed
git-codecommit.ap-northeast-2.amazonaws.com	SHA256	eegAPQrWY9YsYo9ZHI K0mxeTfXBHzAZd8Eya 53Qcwko
git-codecommit.sa-east-1.amazonaws.com	MD5*	74:99:9d:ff:2b:ef: 63:c6:4b:b4:6a:7f: 62:c5:4b:51
git-codecommit.sa-east-1.amazonaws.com	SHA256	kW+VKB0jpRaG/ZbXkg btMQbKgEDK7JnISV3S VoyCmzU
git-codecommit.us-west-1.amazonaws.com	MD5*	3b:76:18:83:13:2c: f8:eb:e9:a3:d0:51: 10:32:e7:d1
git-codecommit.us-west-1.amazonaws.com	SHA256	gzauWTWXDK2u5KuMMi 5vbKTmfyerdIwgSbzY B0DLpzg
git-codecommit.eu-west-2.amazonaws.com	MD5*	a5:65:a6:b1:84:02: b1:95:43:f9:0e:de: dd:ed:61:d3
git-codecommit.eu-west-2.amazonaws.com	SHA256	r0Rwz5k/IHp/QyrRnf iM9j02D5UEqMbtFNTu DG2hNbs

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.ap-south-1.amazonaws.com	MD5*	da:41:1e:07:3b:9e: 76:a0:c5:1e:64:88: 03:69:86:21
git-codecommit.ap-south-1.amazonaws.com	SHA256	hUKwnTj7+Xpx4Kddb6 p45j4RazIJ4IhAMD8k 29it0fE
git-codecommit.ap-south-2.amazonaws.com	MD5*	bc:cc:9f:15:f8:f3: 58:a2:68:65:21:e2: 23:71:8d:ce
git-codecommit.ap-south-2.amazonaws.com	SHA256	Xe0CyZE0vgR5Xa2YUG qf+jn8/Ut7l7nX/Cms lSFNEig
git-codecommit.ca-central-1.amazonaws.com	MD5*	9f:7c:a2:2f:8c:b5: 74:fd:ab:b7:e1:fd: af:46:ed:23
git-codecommit.ca-central-1.amazonaws.com	SHA256	Qz5puafQdANVprLlj6 r0Qyh4lCNsF6ob61dG cPtFS7w
git-codecommit.eu-west-3.amazonaws.com	MD5*	1b:7f:97:dd:d7:76: 8a:32:2c:bd:2c:7b: 33:74:6a:76
git-codecommit.eu-west-3.amazonaws.com	SHA256	uw7c2FL564jVoFgtc+ ikzILnKBsZz7t9+CFd SJjKbLI
git-codecommit.us-gov-west-1.amazonaws.com	MD5*	9f:6c:19:3b:88:cd: e8:88:1b:9c:98:6a: 95:31:8a:69

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.us-gov-west-1.amazonaws.com	SHA256	djXQoSIFcg8vHe0KVH 1xW/g0F9X37tWTqu4H kng75x4
git-codecommit.us-gov-east-1.amazonaws.com	MD5*	00:8d:b5:55:6f:05: 78:05:ed:ea:cb:3f: e6:f0:62:f2
git-codecommit.us-gov-east-1.amazonaws.com	SHA256	fVb+R0z7qW7minenW+ rUpAABRCRBTczmETAJ EQrg98
git-codecommit.eu-north-1.amazonaws.com	MD5*	8e:53:d8:59:35:88: 82:fd:73:4b:60:8a: 50:70:38:f4
git-codecommit.eu-north-1.amazonaws.com	SHA256	b6KSK7xKq+V8jl7iuA cjqXsG7zkqoUZZmmhY YFBq1wQ
git-codecommit.me-south-1.amazonaws.com	MD5*	0e:39:28:56:d5:41: e6:8d:fa:81:45:37: fb:f3:cd:f7
git-codecommit.me-south-1.amazonaws.com	SHA256	0+NTToCGgjHekiBu01 0ad7R0GEsz+DBLX0d/ c9wc0JU
git-codecommit.ap-east-1.amazonaws.com	MD5*	a8:00:3d:24:52:9d: 61:0e:f6:e3:88:c8: 96:01:1c:fe
git-codecommit.ap-east-1.amazonaws.com	SHA256	LafadYwUYW8h0NoTRp objbjNs9IRnbEwHtezD 3aAIBX0

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.cn-north-1.amazonaws.com.cn	MD5*	11:7e:2d:74:9e:3b: 94:a2:69:14:75:6f: 5e:22:3b:b3
git-codecommit.cn-north-1.amazonaws.com.cn	SHA256	IYUXxH20pTDsyYMLIp +JY8CTLS4UX+ZC5JVZ XPRaxc8
git-codecommit.cn-northwest-1.amazonaws.com.cn	MD5*	2e:a7:fb:4c:33:ac: 6c:f9:aa:f2:bc:fb: 0a:7b:1e:b6
git-codecommit.cn-northwest-1.amazonaws.com.cn	SHA256	wqjd6eHd0+m0Bx+dCN uL0omUoCNjaDtZiEpW j5TmCfQ
git-codecommit.eu-south-1.amazonaws.com	MD5*	b9:f6:5d:e2:48:92: 3f:a9:37:1e:c4:d0: 32:0e:fb:11
git-codecommit.eu-south-1.amazonaws.com	SHA256	1yXrWbCg3uQmJr11Xx B/ASR7ugW1Ysf5yzY0 JbudHsI
git-codecommit.ap-northeast-3.amazonaws.com	MD5*	25:17:40:da:b9:d4: 18:c3:b6:b3:fb:ed: 1c:20:fe:29
git-codecommit.ap-northeast-3.amazonaws.com	SHA256	2B815B9F0AvwLnRxSV xUz4kDYmtEQUGGdQYP 80QLXhA
git-codecommit.af-south-1.amazonaws.com	MD5*	21:a0:ba:d7:c1:d1: b5:39:98:8d:4d:7c: 96:f5:ca:29

Server	Tipe hash kriptografi	Sidik jari
git-codecommit.af-south-1.amazonaws.com	SHA256	C34ji3x/cnsDZjUpyN GXdE5pjHYimqJrQZ31 eTgqJHM
git-codecommit.il-central-1.amazonaws.com	MD5*	04:74:89:16:98:7a: 61:b1:69:46:42:3c: d1:b4:ac:a9
git-codecommit.il-central-1.amazonaws.com	SHA256	uFxhp51kUWh1eTLeyb xQVYm4RnNLNZ5Dbdm1 cgdS1/8

Menggunakan AWS CodeCommit dengan antarmuka VPC endpoint

Jika Anda menggunakan Amazon Virtual Private Cloud (Amazon VPC) untuk meng-host AWS sumber daya Anda, Anda dapat membuat koneksi pribadi antara VPC dan VPC. CodeCommit Anda dapat menggunakan koneksi ini untuk memungkinkan CodeCommit untuk berkomunikasi dengan sumber daya Anda di VPC Anda tanpa melalui internet publik.

Amazon VPC adalah AWS layanan yang dapat Anda gunakan untuk meluncurkan AWS sumber daya di jaringan virtual yang Anda tentukan. Dengan VPC, Anda memiliki kendali terhadap pengaturan jaringan, seperti rentang alamat IP, subnet, tabel rute, dan gateway jaringan. Dengan titik akhir VPC, perutean antara VPC dan AWS layanan ditangani oleh AWS jaringan, dan Anda dapat menggunakan kebijakan IAM untuk mengontrol akses ke sumber daya layanan.

Untuk menghubungkan VPC Anda CodeCommit, Anda menentukan titik akhir VPC antarmuka untuk. CodeCommit Endpoint antarmuka adalah elastic network interface dengan alamat IP pribadi yang berfungsi sebagai titik masuk untuk lalu lintas yang ditujukan ke layanan yang didukung AWS . Endpoint menyediakan konektivitas yang andal dan dapat diskalakan CodeCommit tanpa memerlukan gateway internet, instance terjemahan alamat jaringan (NAT), atau koneksi VPN. Untuk informasi selengkapnya, silakan lihat [Apa itu Amazon VPC](#) dalam Panduan Pengguna Amazon VPC.

Note

AWS Layanan lain yang menyediakan dukungan dan integrasi VPC CodeCommit, seperti AWS CodePipeline, mungkin tidak mendukung penggunaan titik akhir VPC Amazon untuk integrasi tersebut. Misalnya, lalu lintas antara CodePipeline dan CodeCommit tidak dapat dibatasi pada rentang subnet VPC. Layanan yang mendukung integrasi, seperti [AWS Cloud9](#), mungkin memerlukan layanan tambahan seperti AWS Systems Manager.

Endpoint VPC antarmuka didukung oleh AWS PrivateLink, sebuah AWS teknologi yang memungkinkan komunikasi pribadi antara AWS layanan menggunakan antarmuka jaringan elastis dengan alamat IP pribadi. Untuk informasi lebih lanjut, lihat [AWS PrivateLink](#).

Langkah-langkah berikut ditujukan untuk para pengguna Amazon VPC. Untuk informasi selengkapnya, silakan lihat [Getting Started](#) di Panduan Pengguna Amazon VPC.

Ketersediaan

CodeCommit saat ini mendukung titik akhir VPC sebagai berikut: Wilayah AWS

- AS Timur (Ohio)
- AS Timur (Virginia Utara)
- AS Barat (California Utara)
- US West (Oregon)
- Europe (Ireland)
- Eropa (London)
- Europe (Paris)
- Europe (Frankfurt)
- Eropa (Stockholm)
- Europe (Milan)
- Afrika (Cape Town)
- Israel (Tel Aviv)
- Asia Pasifik (Tokyo)
- Asia Pasifik (Singapura)
- Asia Pasifik (Sydney)

- Asia Pasifik (Jakarta)
- Timur Tengah (UEA)
- Asia Pasifik (Seoul)
- Asia Pasifik (Osaka)
- Asia Pasifik (Mumbai)
- Asia Pasifik (Hyderabad)
- Asia Pasifik (Hong Kong)
- South America (Sao Paulo)
- Middle East (Bahrain)
- Canada (Central)
- Tiongkok (Beijing)
- China (Ningxia)
- AWS GovCloud (AS-Barat)
- AWS GovCloud (AS-Timur)

Buat titik akhir VPC untuk CodeCommit

Untuk mulai menggunakan CodeCommit dengan VPC Anda, buat antarmuka VPC endpoint untuk CodeCommit. CodeCommit memerlukan titik akhir terpisah untuk operasi Git dan untuk operasi CodeCommit API. Bergantung pada kebutuhan bisnis Anda, Anda mungkin perlu membuat lebih dari satu VPC endpoint. Saat Anda membuat titik akhir VPC untuk CodeCommit, pilih AWS Layanan, dan di Nama Layanan, pilih dari opsi berikut:

- `com.amazonaws.region.git-codecommit`: Pilih opsi ini jika Anda ingin membuat titik akhir VPC untuk operasi Git dengan repositori. CodeCommit Misalnya, pilih opsi ini jika pengguna Anda menggunakan klien Git dan perintah seperti `git pull`, `git commit`, dan `git push` ketika mereka berinteraksi dengan CodeCommit repositori.
- `com.amazonaws.wilayah.git-codecommit-fips`: Pilih opsi ini jika Anda ingin membuat titik akhir VPC untuk operasi Git dengan CodeCommit repositori yang sesuai dengan Standar Pemrosesan Informasi Federal (FIPS) Publikasi 140-2 standar pemerintah AS.

Note

Titik akhir FIPS untuk Git tidak tersedia di semua AWS Wilayah. Untuk informasi selengkapnya, lihat [Titik akhir koneksi Git](#).

- `com.amazonaws.region.codecommit`: Pilih opsi ini jika Anda ingin membuat titik akhir VPC untuk operasi API. CodeCommit Misalnya, pilih opsi ini jika pengguna Anda menggunakan AWS CLI, CodeCommit API, atau AWS SDK untuk berinteraksi dengan CodeCommit operasi seperti `CreateRepository`, `ListRepositories`, dan `PutFile`.
- `com.amazonaws.region.codecommit-fips`: Pilih opsi ini jika Anda ingin membuat titik akhir VPC untuk operasi CodeCommit API yang sesuai dengan Standar Pemrosesan Informasi Federal (FIPS) Publikasi 140-2 standar pemerintah AS.

Note

Titik akhir FIPS tidak tersedia di semua AWS Wilayah. Untuk informasi lebih lanjut, lihat entri untuk AWS CodeCommit di [Federal Information Processing Standard \(FIPS\) 140-2](#) Ikhtisar.

Membuat kebijakan titik akhir VPC untuk CodeCommit

Anda dapat membuat kebijakan untuk titik akhir VPC Amazon yang dapat Anda tentukan: CodeCommit

- Prinsipal yang dapat melakukan tindakan.
- Tindakan yang dapat dilakukan.
- Sumber daya yang dapat memiliki tindakan yang dilakukan pada mereka.

Misalnya, sebuah perusahaan mungkin ingin membatasi akses ke repositori ke kisaran alamat jaringan untuk VPC. Anda dapat menampilkan contoh jenis kebijakan ini di sini: [Contoh 3: Izinkan pengguna terhubung dari rentang alamat IP tertentu untuk mengakses repositori](#). Perusahaan mengonfigurasi dua VPC endpoint Git untuk wilayah US East (Ohio): `com.amazonaws.us-east-2.codecommit` dan `com.amazonaws.us-east-2.git-codecommit-fips`. Mereka ingin mengizinkan dorongan kode ke CodeCommit repositori yang *MyDemoRepo*hanya diberi nama pada titik akhir yang sesuai dengan FIPS saja. Untuk menegakkan ini, mereka akan mengkonfigurasi

kebijakan yang mirip dengan berikut pada titik akhir `com.amazonaws.us-east-2.codecommit` yang secara khusus menolak tindakan dorongan Git:

```
{
  "Statement": [
    {
      "Action": "*",
      "Effect": "Allow",
      "Resource": "*",
      "Principal": "*"
    },
    {
      "Action": "codecommit:GitPush",
      "Effect": "Deny",
      "Resource": "arn:aws:codecommit:us-west-2:123456789012:MyDemoRepo",
      "Principal": "*"
    }
  ]
}
```

Untuk informasi selengkapnya, silakan lihat [Membuat sebuah Titik Akhir Antarmuka](#) dalam Panduan Pengguna Amazon VPC.

Kuota di AWS CodeCommit

Tabel berikut menjelaskan kuota di CodeCommit. Untuk informasi tentang kuota yang dapat diubah, lihat [Titik akhir dan Kuota AWS CodeCommit](#). Untuk informasi tentang meminta peningkatan kuota layanan, lihat Service [AWS Quotas](#). Untuk informasi tentang versi Git yang diperlukan dan perangkat lunak lainnya, lihat [Kompatibilitas untuk CodeCommit, Git, dan komponen lainnya](#).

Aturan persetujuan dan nama templat aturan persetujuan	Kombinasi huruf, angka, titik, spasi, garis bawah, dan tanda hubung dengan panjang antara 1 dan 100 karakter. Nilai peka huruf besar/kecil. Nama tidak dapat diakhiri dengan <code>.git</code> dan tidak dapat berisi salah satu karakter berikut: <code>! ? @ # \$ % ^ & * () + = { } [] \ / > < ~ ` ' " ; :</code>
Panjang isi aturan persetujuan	3000 karakter

Panjang deskripsi templat aturan persetujuan	1000 karakter
Referensi tujuan templat aturan persetujuan	100
Templat aturan persetujuan	1000 dalam sebuah Wilayah AWS
Aturan persetujuan pada permintaan tarik	Maksimum hingga 30. Sampai dengan 25 ini dapat dari templat aturan persetujuan.
Aturan persetujuan pada permintaan tarik yang dibuat dari templat aturan persetujuan	25
Persetujuan pada permintaan tarik	200
Pemberi persetujuan dalam sebuah kolom persetujuan	50
Nama cabang	<p>Kombinasi karakter yang diizinkan antara 1 dan 256 karakter panjangnya, kecuali bahwa nama cabang persis 40 karakter heksadesimal tidak diperbolehkan. Nama cabang tidak boleh:</p> <ul style="list-style-type: none"> • dimulai atau diakhiri dengan garis miring (/) atau tanda titik (.) • terdiri dari karakter tunggal @ • mengandung dua atau lebih tanda titik (.) berturut-turut, garis miring ke depan (//), atau kombinasi dari karakter berikut: @{ • berisi spasi atau salah satu karakter berikut: ? ^ * [\ ~ : <p>Nama cabang adalah referensi. Banyak pembatasan pada nama cabang didasarkan pada standar referensi Git. Untuk informasi selengkapnya, lihat Git Internals dan git-check-ref-format.</p>
Panjang komentar	Maksimal 10.240 karakter.

Data khusus untuk pemicu	Ini adalah bidang string yang dibatasi hingga 1.000 karakter. Ini tidak dapat digunakan untuk melewati parameter dinamis.
Tampilan di konsol	File atau perbandingan antar file mungkin tidak dapat dilihat di konsol jika: <ul style="list-style-type: none">• File lebih besar dari 2 MB• File berisi lebih dari 25.000 karakter dalam satu baris• Perbandingan berisi lebih dari 6.500 baris perbedaan
Alamat email dalam commit yang dibuat di konsol	Setiap kombinasi karakter diperbolehkan dengan panjang antara 1 dan 256 karakter. Alamat email tidak divalidasi.

Jalur file

Setiap kombinasi karakter yang diperbolehkan dengan panjang antara 1 dan 4.096 karakter. Jalur file harus berupa nama yang tidak ambigu yang menentukan file dan lokasi yang tepat dari file tersebut. Kedalaman jalur file tidak boleh melebihi 20 direktori. Selain itu, jalur file tidak boleh:

- berisi string kosong
- berupa jalur file relatif
- menyertakan salah satu kombinasi karakter berikut:

`./`

`../`



`//`

- diakhiri dengan garis miring atau garis miring terbalik

Nama file dan jalur harus memenuhi syarat. Nama dan jalur ke sebuah file pada komputer lokal Anda harus mengikuti standar untuk sistem operasi tersebut. Saat menentukan jalur ke file dalam CodeCommit repositori, gunakan standar untuk Amazon Linux.

Ukuran file

Maksimum 6 MB untuk setiap file individual saat menggunakan CodeCommit konsol, API, atau file AWS CLI.

Ukuran gumpalan Git	Maksimal 2 GB. <div> Note Tidak ada batas pada jumlah atau ukuran total semua file dalam satu commit, selama metadata tidak melebihi 6 MB dan gumpalan tunggal tidak melebihi 2 GB.</div>
Tampilan grafik dari cabang-cabang dalam Pemvisualisasi Commit	35 per halaman. Jika ada lebih dari 35 cabang pada satu halaman, grafik tersebut tidak ditampilkan.
Metadata untuk commit	Maksimum 20 MB untuk metadata gabungan untuk komit (misalnya, kombinasi informasi penulis, tanggal, daftar komit induk, dan pesan komit) saat menggunakan CodeCommit konsol, API, atau file. AWS CLI <div> Note Tidak ada batasan jumlah atau ukuran total semua file dalam satu komit, selama metadata tidak melebihi 6 MB, file individual tidak melebihi 6 MB, dan satu gumpalan tidak melebihi 2 GB.</div>
Jumlah file dalam komit	Maksimal 100.
Jumlah permintaan tarik terbuka	Maksimum 1.000.
Jumlah referensi dalam satu kali push	Maksimum 4.000, termasuk membuat, menghapus, dan memperbarui. Tidak ada batas jumlah keseluruhan referensi dalam repositori.

Jumlah repositori	Maksimal 5.000 per akun Amazon Web Services. Batas ini tidak dapat diubah. Untuk informasi selengkapnya, lihat AWS CodeCommit Titik Akhir dan Kuota dan Service AWS Quotas .
Jumlah pemacu dalam sebuah repositori	Maksimum 10.

Wilayah

CodeCommit tersedia sebagai berikut Wilayah AWS:

- AS Timur (Ohio)
- AS Timur (Virginia Utara)
- AS Barat (California Utara)
- US West (Oregon)
- Europe (Ireland)
- Eropa (London)
- Europe (Paris)
- Europe (Frankfurt)
- Eropa (Stockholm)
- Europe (Milan)
- Afrika (Cape Town)
- Israel (Tel Aviv)
- Asia Pasifik (Tokyo)
- Asia Pasifik (Singapura)
- Asia Pasifik (Sydney)
- Asia Pasifik (Jakarta)
- Timur Tengah (UEA)
- Asia Pasifik (Seoul)
- Asia Pasifik (Osaka)
- Asia Pasifik (Mumbai)
- Asia Pasifik (Hyderabad)
- Asia Pasifik (Hong Kong)
- South America (Sao Paulo)
- Middle East (Bahrain)
- Canada (Central)
- Tiongkok (Beijing)
- China (Ningxia)
- AWS GovCloud (AS-Barat)

- AWS GovCloud (AS-Timur)

Untuk informasi selengkapnya, lihat [Wilayah dan titik akhir koneksi Git](#).

Deskripsi repositori	Kombinasi karakter apa pun dengan panjang antara 0 dan 1.000 karakter. Deskripsi repositori adalah opsional.
Nama repositori	Kombinasi huruf, angka, titik, spasi, garis bawah, dan tanda hubung dengan panjang antara 1 dan 100 karakter. Nilai peka huruf besar/kecil. Nama tidak dapat diakhiri dengan.git dan tidak boleh berisi salah satu karakter berikut: ! ? @ # \$ % ^ & * () + = { } [] \ / > < ~ ` ' " ; :
Nama kunci tag repositori	<p>Setiap kombinasi huruf Unicode, angka, spasi, dan karakter diperbolehkan dalam UTF-8 dengan panjang antara 1 dan 128 karakter. Karakter yang diizinkan adalah + - = . _ : / @</p> <p>Nama kunci tag harus unik, dan setiap kunci hanya dapat memiliki satu nilai. Sebuah tanda tidak boleh:</p> <ul style="list-style-type: none"> • dimulai dengan aws : • hanya terdiri dari spasi • diakhiri dengan spasi • berisi spasi atau setiap karakter berikut: ? ^ * [\ ~ ! # \$ % & * () > < " ' ` [] { } ;

<p>Nilai tag repositori</p>	<p>Setiap kombinasi huruf Unicode, angka, spasi, dan karakter diizinkan dalam UTF-8 dengan panjang antara 1 dan 256 karakter. Karakter yang diizinkan adalah + - = . _ : / @</p> <p>Kunci hanya dapat memiliki satu nilai, tetapi banyak kunci dapat memiliki nilai yang sama. Sebuah tanda tidak boleh:</p> <ul style="list-style-type: none"> • hanya terdiri dari spasi • diakhiri dengan spasi • berisi emoji atau salah satu dari karakter berikut: ? ^ * [\ ~ ! # \$ % & * () > < " ' ` [] { } ;
<p>Tanda repositori</p>	<p>Tag peka terhadap huruf besar dan kecil. Maksimum 50 per sumber daya. Nama tag persis 40 karakter heksadesimal tidak diperbolehkan.</p>
<p>Nama pemicu</p>	<p>Kombinasi huruf, angka, titik, spasi, garis bawah, dan tanda hubung dengan panjang antara 1 dan 100 karakter. Nama pemicu tidak boleh berisi spasi atau koma.</p>
<p>Nama pengguna dalam commit yang dibuat di konsol</p>	<p>Setiap kombinasi karakter yang diizinkan dengan panjang antara 1 dan 1.024 karakter.</p>

AWS CodeCommit referensi baris perintah

Referensi ini membantu Anda mempelajari cara menggunakan AWS CLI.

Untuk menginstal dan mengkonfigurasi AWS CLI

1. Di mesin lokal Anda, unduh dan instal file AWS CLI. Ini adalah prasyarat untuk berinteraksi dengan CodeCommit dari baris perintah. Kami merekomendasikan agar Anda menginstal AWS CLI versi 2. Ini adalah versi utama terbaru dari AWS CLI dan mendukung semua fitur terbaru. Ini

adalah satu-satunya versi AWS CLI yang mendukung penggunaan akun root, akses federasi, atau kredensi sementara dengan `git-remote-codecommit`

Untuk informasi selengkapnya, lihat [Menyiapkan dengan Antarmuka Baris AWS Perintah](#).

Note

CodeCommit hanya berfungsi dengan AWS CLI versi 1.7.38 dan yang lebih baru. Sebagai praktik terbaik, instal atau tingkatkan AWS CLI ke versi terbaru yang tersedia. Untuk menentukan versi mana yang telah AWS CLI Anda instal, jalankan `aws --version` perintah.

Untuk memutakhirkan versi yang lebih lama AWS CLI ke versi terbaru, lihat [Menginstal AWS Command Line Interface](#).

2. Jalankan perintah ini untuk memverifikasi bahwa CodeCommit perintah AWS CLI untuk diinstal.

```
aws codecommit help
```

Perintah ini mengembalikan daftar CodeCommit perintah.

3. Konfigurasi AWS CLI dengan profil dengan menggunakan `configure` perintah, sebagai berikut:

```
aws configure
```

Saat diminta, tentukan kunci AWS akses dan kunci akses AWS rahasia pengguna IAM untuk digunakan. CodeCommit Juga, pastikan untuk menentukan di Wilayah AWS mana repositori ada, seperti `us-east-2` Saat diminta untuk format output default, tentukan `json`. Misalnya, jika Anda mengonfigurasi profil untuk pengguna IAM:

```
AWS Access Key ID [None]: Type your IAM user AWS access key ID here, and then press Enter
```

```
AWS Secret Access Key [None]: Type your IAM user AWS secret access key here, and then press Enter
```

```
Default region name [None]: Type a supported region for CodeCommit here, and then press Enter
```

```
Default output format [None]: Type json here, and then press Enter
```

Untuk informasi selengkapnya tentang membuat dan mengonfigurasi profil yang akan digunakan AWS CLI, lihat berikut ini:

- [Profil Bernama](#)
- [Menggunakan Peran IAM dalam AWS CLI](#)
- [Tetapkan perintah](#)
- [Menghubungkan ke repositori AWS CodeCommit dengan kredensial berputar](#)

Untuk terhubung ke repositori atau sumber daya di tempat lain Wilayah AWS, Anda harus mengkonfigurasi ulang AWS CLI dengan nama Wilayah default. Nama Wilayah default yang didukung untuk CodeCommit menyertakan:

- us-east-2
- us-east-1
- eu-west-1
- us-west-2
- ap-northeast-1
- ap-southeast-1
- ap-southeast-2
- ap-southeast-3
- me-central-1
- eu-central-1
- ap-northeast-2
- sa-east-1
- us-west-1
- eu-west-2
- ap-south-1
- ap-south-1
- ca-central-1
- us-gov-west-1
- us-gov-east-1

- [ap-east-1](#)
- [me-south-1](#)
- [cn-north-1](#)
- [cn-northwest-1](#)
- [eu-south-1](#)
- [ap-northeast-3](#)
- [af-south-1](#)
- [il-central-1](#)

Untuk informasi lebih lanjut tentang CodeCommit dan Wilayah AWS, lihat [Wilayah dan titik akhir koneksi Git](#). Untuk informasi selengkapnya tentang IAM, access key, dan kunci rahasia, lihat [Bagaimana Saya Mendapatkan Kredensial?](#) dan [Mengelola Access Key untuk Pengguna IAM](#). Untuk informasi selengkapnya tentang profil AWS CLI dan profil, lihat [Profil Bernama](#).

Untuk melihat daftar semua CodeCommit perintah yang tersedia, jalankan perintah berikut:

```
aws codecommit help
```

Untuk melihat informasi tentang CodeCommit perintah, jalankan perintah berikut, di mana *command-name* adalah nama perintah (misalnya, create-repository):

```
aws codecommit command-name help
```

Lihat berikut ini untuk deskripsi dan contoh penggunaan perintah di AWS CLI:

- [associate-approval-rule-template-dengan-repositori](#)
- [batch-associate-approval-rule-template-with-repositories](#)
- [batch-disassociate-approval-rule-template-from-repositories](#)
- [batch-describe-merge-conflicts](#)
- [batch-get-commits](#)
- [batch-get-repositories](#)
- [create-approval-rule-template](#)
- [buat-cabang](#)

- [buat-komit](#)
- [create-pull-request](#)
- [create-pull-request-approval-aturan](#)
- [buat-repositori](#)
- [create-unreferenced-merge-commit](#)
- [delete-approval-rule-template](#)
- [hapus-cabang](#)
- [delete-comment-content](#)
- [menghapus-file](#)
- [hapus-repositori](#)
- [describe-merge-conflicts](#)
- [delete-pull-request-approval-aturan](#)
- [describe-pull-request-events](#)
- [disassociate-pull-request-approval-rule-template-from-repository](#)
- [evaluate-pull-request-approval-aturan](#)
- [get-approval-rule-template](#)
- [mendapatkan-gumpalan](#)
- [dapatkan-cabang](#)
- [dapatkan-komentar](#)
- [get-comment-reactions](#)
- [get-comments-for-compared-berkomitmen](#)
- [get-comments-for-pull-permintaan](#)
- [dapatkan-komit](#)
- [dapatkan-perbedaan](#)
- [get-merge-commit](#)
- [get-merge-conflicts](#)
- [get-merge-options](#)
- [get-pull-request](#)
- [get-pull-request-approval-negara](#)

- [get-pull-request-override-negara](#)
- [dapatkan-repositori](#)
- [get-repository-triggers](#)
- [list-approval-rule-templates](#)
- [list-associated-approval-rule-templates-for-repository](#)
- [daftar-cabang](#)
- [list-pull-requests](#)
- [daftar-repositori](#)
- [list-repositories-for-approval-aturan-templat](#)
- [list-tags-for-resource](#)
- [merge-branches-by-fast-maju](#)
- [merge-branches-by-squash](#)
- [merge-branches-by-three-cara](#)
- [merge-pull-request-by-maju cepat](#)
- [merge-pull-request-by-labu](#)
- [merge-pull-request-by-tiga arah](#)
- [override-pull-request-approval-aturan](#)
- [post-comment-for-compared-berkomitmen](#)
- [post-comment-for-pull-permintaan](#)
- [post-comment-reply](#)
- [put-comment-reaction](#)
- [taruh-file](#)
- [put-repository-triggers](#)
- [tag-sumber daya](#)
- [test-repository-triggers](#)
- [untag-sumber daya](#)
- [update-approval-rule-template-konten](#)
- [update-approval-rule-template-deskripsi](#)
- [update-approval-rule-template-nama](#)
- [pembaruan-komentar](#)

- [update-default-branch](#)
- [update-pull-request-approval-aturan-konten](#)
- [update-pull-request-approval-negara](#)
- [update-pull-request-description](#)
- [update-pull-request-status](#)
- [update-pull-request-title](#)
- [update-repository-description](#)
- [update-repository-name](#)

Perintah Git Basic

Anda dapat menggunakan Git untuk bekerja dengan repo lokal dan CodeCommit repositori tempat Anda menghubungkan repo lokal.

Berikut ini adalah beberapa contoh dasar dari perintah Git yang sering digunakan.

Untuk opsi lainnya, lihat dokumentasi Git Anda.

Topik

- [Variabel konfigurasi](#)
- [Repositori jarak jauh](#)
- [Berkomitmen](#)
- [Cabang](#)
- [Tanda](#)

Variabel konfigurasi

Daftar semua variabel konfigurasi.	<code>git config --list</code>
Hanya daftar variabel konfigurasi lokal.	<code>git config --local -l</code>
Hanya daftar variabel konfigurasi sistem.	<code>git config --system -l</code>
Hanya daftar variabel konfigurasi global.	<code>git config --global -l</code>

Mengatur variabel konfigurasi dalam file konfigurasi yang ditentukan.

```
git config [--local | --global | --system] variable-name variable-value
```

Mengatur nama cabang default menjadi utama untuk semua repositori lokal ketika commit awal dibuat ke repositori yang belum memiliki cabang default

```
git config --global init.defaultBranch main
```

Mengedit file konfigurasi secara langsung. Dapat juga digunakan untuk menemukan lokasi file konfigurasi tertentu. Untuk keluar dari mode edit, biasanya Anda mengetik :q (untuk keluar tanpa menyimpan perubahan) atau :wq (untuk menyimpan perubahan dan kemudian keluar), dan kemudian tekan Enter.

```
git config [--local | --global | --system] --edit
```

Repositori jarak jauh

Menginisialisasi repo lokal sebagai persiapan untuk menghubungkannya ke repositori. CodeCommit

```
git init
```

Dapat digunakan untuk mengatur koneksi antara repo lokal dan repositori jarak jauh (seperti repositori) menggunakan nama panggilan tertentu yang dimiliki repo lokal untuk CodeCommit repositori dan URL yang ditentukan ke CodeCommit repositori. CodeCommit

```
git remote add remote-name remote-url
```

Membuat repo lokal dengan membuat salinan CodeCommit repositori di URL yang ditentukan, di subfolder tertentu dari folder saat ini di mesin lokal. Perintah ini juga membuat cabang pelacakan jarak jauh untuk setiap cabang di

```
git clone remote-url local-subfolder-name
```

CodeCommit repositori kloning dan membuat dan memeriksa cabang awal yang bercabang dari cabang default saat ini di repositori kloning. CodeCommit	
Menampilkan nama panggilan yang digunakan repo lokal untuk repositori. CodeCommit	<code>git remote</code>
Menampilkan nama panggilan dan URL yang digunakan repo lokal untuk mengambil dan mendorong ke repositori. CodeCommit	<code>git remote -v</code>
Mendorong komit yang diselesaikan dari repo lokal ke CodeCommit repositori, menggunakan nama panggilan yang ditentukan yang dimiliki repo lokal untuk repositori dan cabang yang ditentukan. CodeCommit Juga mengatur informasi pelacakan hulu untuk repo lokal selama dorongan.	<code>git push -u <i>remote-name</i> <i>branch-name</i></code>
Mendorong komit yang diselesaikan dari repo lokal ke CodeCommit repositori setelah informasi pelacakan hulu disetel.	<code>git push</code>
Menarik komit yang telah diselesaikan ke repo lokal dari CodeCommit repositori, menggunakan nama panggilan yang ditentukan yang dimiliki repo lokal untuk repositori dan cabang yang ditentukan CodeCommit	<code>git pull <i>remote-name</i> <i>branch-name</i></code>
Menarik komit yang telah diselesaikan ke repo lokal dari CodeCommit repositori setelah informasi pelacakan hulu disetel.	<code>git pull</code>
Putuskan sambungan repo lokal dari CodeCommit repositori, menggunakan nama panggilan yang ditentukan yang dimiliki repo lokal untuk repositori. CodeCommit	<code>git remote rm <i>remote-name</i></code>

Berkomitmen

Menunjukkan apa yang telah atau belum ditambahkan ke commit yang tertunda di repo lokal.	<code>git status</code>
Menunjukkan apa yang telah atau belum ditambahkan ke commit yang tertunda di repo lokal dalam format ringkas. (M = dimodifikasi, A = ditambahkan, D = dihapus, dan sebagainya)	<code>git status -sb</code>
Menunjukkan perubahan antara commit yang tertunda dan commit terbaru di repo lokal.	<code>git diff HEAD</code>
Menambahkan file tertentu untuk commit yang tertunda di repo lokal.	<code>git add [file-name-1 file-name-2 file-name-N file-pattern]</code>
Menambahkan semua file baru, yang diubah, dan yang dihapus ke commit yang tertunda di repo lokal.	<code>git add</code>
Mulai menyelesaikan commit yang tertunda di repo lokal, yang menampilkan editor untuk memberikan pesan commit. Setelah pesan dimasukkan, commit yang tertunda diselesaikan.	<code>git commit</code>
Menyelesaikan commit yang tertunda di repo lokal, termasuk menentukan pesan commit pada saat yang sama.	<code>git commit -m "Some meaningful commit comment"</code>
Mendaftar commit terbaru di repo lokal.	<code>git log</code>
Mendaftar commit terbaru di repo lokal dalam format grafik.	<code>git log --graph</code>

Mendaftar commit terbaru di repo lokal dalam format kental yang telah ditetapkan.	<code>git log --pretty=oneline</code>
Mendaftar commit terbaru di repo lokal dalam format kental yang telah ditetapkan, dengan grafik.	<code>git log --graph --pretty=oneline</code>
Mendaftar commit terbaru di repo lokal dalam format kustom, dengan grafik. (Untuk opsi lainnya, lihat Dasar-dasar Git - Menampilkan Riwayat Commit)	<code>git log --graph --pretty=format:"%H (%h) : %cn : %ar : %s"</code>

Cabang

Mendaftar semua cabang di repo lokal dengan tanda bintang (*) yang ditampilkan di sebelah cabang Anda saat ini.	<code>git branch</code>
Menarik informasi tentang semua cabang yang ada di CodeCommit repositori ke repo lokal.	<code>git fetch</code>
Mendaftar semua cabang di repo lokal dan cabang pelacakan jarak jauh di repo lokal.	<code>git branch -a</code>
Mendaftar hanya cabang pelacakan jarak jauh di repo lokal.	<code>git branch -r</code>
Membuat sebuah cabang baru di repo lokal menggunakan nama cabang yang ditentukan.	<code>git branch <i>new-branch-name</i></code>
Beralih ke cabang lain di repo lokal dengan menggunakan nama cabang yang ditentukan.	<code>git checkout <i>other-branch-name</i></code>
Membuat sebuah cabang baru di repo lokal dengan menggunakan nama cabang yang ditentukan, dan kemudian beralih ke sana.	<code>git checkout -b <i>new-branch-name</i></code>

Mendorong cabang baru dari repo lokal ke CodeCommit repositori menggunakan nama panggilan yang ditentukan yang dimiliki repo lokal untuk CodeCommit repositori dan nama cabang yang ditentukan. Juga mengatur informasi pelacakan hulu untuk cabang di repo lokal selama dorongan.

```
git push -u remote-name new-branch-name
```

Membuat sebuah cabang baru di repo lokal menggunakan nama cabang yang ditentukan. Kemudian hubungkan cabang baru di repo lokal ke cabang yang ada di CodeCommit repositori, menggunakan nama panggilan yang ditentukan yang dimiliki repo lokal untuk CodeCommit repositori dan nama cabang yang ditentukan.

```
git branch --track new-branch-name remote-name /remote-branch-name
```

Menggabungkan perubahan dari cabang lain di repo lokal untuk cabang saat ini di repo lokal.

```
git merge from-other-branch-name
```

Menghapus cabang di repo lokal kecuali cabang tersebut berisi pekerjaan yang belum digabung.

```
git branch -d branch-name
```

Menghapus cabang di CodeCommit repositori menggunakan nama panggilan yang ditentukan yang dimiliki repo lokal untuk CodeCommit repositori dan nama cabang yang ditentukan. (Perhatikan penggunaan titik dua (:).)

```
git push remote-name :branch-name
```

Tanda

Mendaftar semua tag di repo lokal.

```
git tag
```

Menarik semua tag dari CodeCommit repositori ke repo lokal.

```
git fetch --tags
```

Menunjukkan informasi tentang tag tertentu di repo lokal.	<code>git show <i>tag-name</i></code>
Membuat sebuah tanda "ringan" di repo lokal.	<code>git tag <i>tag-name</i> <i>commit-id-to-point-tag-at</i></code>
Mendorong tag tertentu dari repo lokal ke CodeCommit repositori menggunakan nama panggilan yang ditentukan yang dimiliki repo lokal untuk CodeCommit repositori dan nama tag yang ditentukan.	<code>git push <i>remote-name</i> <i>tag-name</i></code>
Mendorong semua tag dari repo lokal ke CodeCommit repositori menggunakan nama panggilan yang ditentukan yang dimiliki repo lokal untuk repositori. CodeCommit	<code>git push <i>remote-name</i> --tags</code>
Menghapus tag di repo lokal.	<code>git tag -d <i>tag-name</i></code>
Menghapus tag di CodeCommit repositori menggunakan nama panggilan yang ditentukan yang dimiliki repo lokal untuk CodeCommit repositori dan nama tag yang ditentukan. (Perhatikan penggunaan titik dua (:).)	<code>git push <i>remote-name</i> :<i>tag-name</i></code>

Riwayat dokumen Panduan Pengguna AWS CodeCommit

Tabel berikut menjelaskan perubahan penting pada dokumentasi untuk CodeCommit. Untuk notifikasi tentang pembaruan dokumentasi ini, Anda dapat berlangganan RSS feed.

- Versi API: 2015-04-13

Perubahan	Deskripsi	Tanggal
CodeCommit sekarang mendukung penggunaan kunci yang dikelola pelanggan	Anda sekarang dapat menggunakan kunci yang dikelola pelanggan atau Kunci yang dikelola AWS untuk mengenkripsi dan mendekripsi data dalam repositori. Untuk informasi selengkapnya, lihat AWS KMS dan enkripsi , Buat repositori , dan Ubah pengaturan repositori .	Desember 21, 2023
CodeCommit sekarang tersedia di Israel (Tel Aviv) ;	CodeCommit sekarang tersedia di Israel (Tel Aviv). Untuk informasi lebih lanjut, lihat Wilayah dan Titik Akhir Koneksi Git .	Agustus 28, 2023
Perubahan kebijakan terkelola untuk CodeCommit	AWSCodeCommitFullAccess Kebijakan AWSCodeCommitPowerUser dan kebijakan telah diperbarui dengan izin tambahan. Untuk informasi selengkapnya, lihat CodeCommit pembaruan kebijakan AWS terkelola .	16 Mei 2023

[CodeCommit sekarang tersedia dalam tiga tambahan Wilayah AWS](#)

CodeCommit Sekarang tersedia dalam tiga tambahan Wilayah AWS: Asia Pasifik (Jakarta), Timur Tengah (UEA), dan Asia Pasifik (Hyderabad). Untuk informasi lebih lanjut, lihat [Wilayah dan Titik Akhir Koneksi Git](#).

28 Februari 2023

[CodeCommit sekarang tersedia di Afrika \(Cape Town\)](#)

CodeCommit sekarang tersedia dalam tambahan Wilayah AWS: Afrika (Cape Town). Untuk informasi lebih lanjut, lihat [Wilayah dan Titik Akhir Koneksi Git](#).

15 September 2021

[Perubahan kebijakan terkelola untuk CodeCommit](#)

Detail tentang pembaruan kebijakan AWS terkelola untuk sekarang CodeCommit tersedia. Untuk informasi selengkapnya, lihat [CodeCommit pembaruan kebijakan AWS terkelola](#).

Agustus 18, 2021

[CodeCommit sekarang tersedia di Asia Pasifik \(Osaka\)](#)

CodeCommit sekarang tersedia dalam tambahan Wilayah AWS: Asia Pasifik (Osaka). Untuk informasi lebih lanjut, lihat [Wilayah dan Titik Akhir Koneksi Git](#).

14 April 2021

[AWS CloudFormation dan AWS Cloud Development Kit \(AWS CDK\) mengubah perilaku penamaan untuk cabang default di CodeCommit](#)

Repositori yang dibuat menggunakan AWS CloudFormation atau AWS CDK dengan commit awal dari kode sekarang menggunakan an nama cabang default utama. Perubahan ini tidak mempengaruhi repositori atau cabang yang ada. Pelanggan yang menggunakan klien Git lokal untuk membuat commit awal mereka memiliki nama cabang default yang mengikuti konfigurasi klien Git tersebut. Untuk informasi selengkapnya, lihat [Membuat CodeCommit sumber daya dengan AWS CloudFormation](#).

4 Maret 2021

[CodeCommit mengubah perilaku penamaan untuk cabang default](#)

Mulai 19 Januari 2021, nama cabang default yang dibuat oleh komit awal ke CodeCommit repositori adalah main. Perubahan ini tidak mempengaruhi repositori atau cabang yang ada. Pelanggan yang menggunakan klien Git lokal untuk membuat commit awal mereka memiliki nama cabang default yang mengikuti konfigurasi klien Git tersebut. Untuk informasi selengkapnya, lihat [Bekerja dengan cabang](#), [Buat commit](#), dan [Ubah pengaturan cabang](#).

19 Januari 2021

[CodeCommit sekarang tersedia di Eropa \(Milan\)](#)

CodeCommit sekarang tersedia dalam tambahan Wilayah AWS: Eropa (Milan). Untuk informasi lebih lanjut, lihat [Wilayah dan Titik Akhir Koneksi Git](#).

16 September 2020

[CodeCommit menambahkan dukungan untuk reaksi emoji ke komentar](#)

CodeCommit sekarang mendukung reaksi terhadap komentar dari pengguna lain dengan emoji. Untuk informasi selengkapnya, lihat [Mengomentari commit](#) dan [Tinjau Permintaan Pull](#).

24 Juni 2020

[CodeCommit sekarang tersedia di China \(Beijing\) dan China \(Ningxia\)](#)

CodeCommit sekarang tersedia dalam dua tambahan Wilayah AWS: China (Beijing) dan China (Ningxia). Untuk informasi lebih lanjut, lihat [Wilayah dan Titik Akhir Koneksi Git](#).

23 April 2020

[CodeCommit menambahkan dukungan untuk git-remote-codecommit](#)

CodeCommit mendukung koneksi ke CodeCommit repositori melalui HTTPS dengan git-remote-codecommit, utilitas yang memodifikasi Git. Ini adalah pendekatan yang direkomendasikan untuk koneksi akses federasi atau sementara ke CodeCommit repositori. Anda juga dapat menggunakan git-remote-codecommit dengan pengguna IAM. git-remote-codecommit tidak mengharuskan Anda untuk mengatur kredensial Git untuk pengguna. Untuk informasi selengkapnya, lihat [Langkah Pengaturan untuk Koneksi HTTPS ke AWS CodeCommit dengan git-remote-codecommit](#).

4 Maret 2020

[CodeCommit mendukung tag sesi](#)

CodeCommit mendukung penggunaan tag sesi, yang merupakan atribut pasangan nilai kunci yang Anda lewati saat Anda mengambil peran IAM, menggunakan kredensial sementara, atau menyatukan pengguna di (). AWS Security Token Service AWS STS Anda dapat menggunakan informasi yang diberikan dalam tag ini untuk mempermudah mengidentifikasi siapa yang membuat perubahan atau menyebabkan suatu peristiwa. Untuk informasi selengkapnya, lihat [Memantau CodeCommit](#) dan [Menggunakan Tag untuk Memberikan Informasi Identitas di CodeCommit](#).

19 Desember 2019

[CodeCommit tersedia di Asia Pasifik \(Hong Kong\)](#)

Anda sekarang dapat menggunakan CodeCommit di Asia Pasifik (Hong Kong). Untuk informasi lebih lanjut, termasuk titik akhir koneksi Git, lihat [Wilayah](#).

11 Desember 2019

[CodeCommit mendukung Amazon CodeGuru Reviewer](#)

CodeCommit mendukung Amazon CodeGuru Reviewer, layanan peninjauan kode otomatis yang menggunakan analisis program dan pembelajaran mesin untuk mendeteksi masalah umum dan merekomendasikan perbaikan dalam kode Java atau Python Anda. Untuk informasi selengkapnya, lihat [Mengaitkan atau Memutus Repositori dengan Amazon CodeGuru Reviewer dan Bekerja dengan Permintaan Tarik](#).

3 Desember 2019

[CodeCommit mendukung aturan persetujuan](#)

Sekarang Anda dapat menggunakan aturan persetujuan membantu Anda menyesuaikan alur kerja pengembangan Anda di seluruh repositori sehingga cabang yang berbeda memiliki tingkat persetujuan dan kontrol yang sesuai. Untuk informasi selengkapnya, lihat [Bekerja Dengan Template Aturan Persetujuan](#) dan [Bekerja Dengan Permintaan Pull](#).

20 November 2019

[CodeCommit mendukung aturan pemberitahuan](#)

Anda sekarang dapat menggunakan aturan notifikasi untuk menginformasikan pengguna perubahan penting dalam repositori. Fungsionalitas dalam fitur ini menggantikan notifikasi yang dibuat sebelum 5 November 2019. Untuk informasi selengkapnya, lihat [Buat Aturan Notifikasi](#).

5 November 2019

[CodeCommit tersedia di Timur Tengah \(Bahrain\)](#)

Anda sekarang dapat menggunakan CodeCommit di Timur Tengah (Bahrain). Untuk informasi lebih lanjut, termasuk titik akhir koneksi Git, lihat [Wilayah](#).

30 Oktober 2019

[CodeCommit menambahkan dukungan untuk mengambil informasi tentang beberapa komit](#)

Anda bisa mendapatkan informasi tentang beberapa komit dengan menggunakan `batch-get-commits` perintah di AWS CLI. Untuk informasi selengkapnya, lihat [Tampilkan Detail commit](#).

15 Agustus 2019

[CodeCommit tersedia di Eropa \(Stockholm\)](#)

Anda sekarang dapat menggunakan CodeCommit di Eropa (Stockholm). Untuk informasi lebih lanjut, termasuk titik akhir koneksi Git, lihat [Wilayah](#).

31 Juli 2019

[CodeCommit menambahkan dukungan untuk menandai repositori di konsol CodeCommit](#)

Sekarang Anda dapat menambahkan, mengelola, dan menghapus tag untuk repositori untuk membantu Anda mengelola AWS sumber daya dari konsol. CodeCommit Untuk informasi selengkapnya, lihat [Menandai Repositori](#).

2 Juli 2019

[CodeCommit menambahkan dukungan untuk strategi penggabungan Git tambahan](#)

Anda sekarang dapat memilih antara strategi penggabungan Git saat menggabungkan permintaan tarik. CodeCommit Anda juga dapat menyelesaikan konflik gabungan di CodeCommit konsol. Untuk informasi selengkapnya, lihat [Bekerja dengan Permintaan Pull](#).

10 Juni 2019

[CodeCommit tersedia di AWS GovCloud \(AS-Timur\)](#)

Anda sekarang dapat menggunakan CodeCommit di AWS GovCloud (AS-Timur). Untuk informasi lebih lanjut, termasuk titik akhir koneksi Git, lihat [Wilayah](#).

31 Mei 2019

[CodeCommit menambahkan dukungan untuk menandai repositori](#)

Sekarang Anda dapat menambahkan, mengelola, dan menghapus tanda untuk repositori untuk membantu Anda mengelola sumber daya AWS. Untuk informasi selengkapnya, lihat [Menandai repositori](#).

30 Mei 2019

Temukan sumber daya di konsol	Sekarang Anda dapat dengan cepat mencari sumber daya Anda, seperti repositori, membangun proyek, aplikasi deployment, dan alur. Pilih Pergi ke sumber daya atau tekan tombol /, dan kemudian ketik nama sumber daya. Untuk informasi lebih lanjut, lihat CodeCommit Tutorial .	14 Mei 2019
CodeCommit tersedia di AWS GovCloud (AS-Barat)	Anda sekarang dapat menggunakan CodeCommit di AWS GovCloud (AS-Barat). Untuk informasi lebih lanjut, termasuk titik akhir koneksi Git, lihat Wilayah .	18 April 2019
CodeCommit menambahkan dukungan untuk titik akhir Amazon VPC	Anda sekarang dapat membuat koneksi pribadi antara VPC Anda dan CodeCommit Untuk informasi selengkapnya, lihat Menggunakan CodeCommit dengan Titik Akhir VPC Antarmuka .	7 Maret 2019
CodeCommit menambahkan API baru	CodeCommit telah menambahkan API untuk membuat komit. Untuk informasi selengkapnya, lihat Buat commit .	20 Februari 2019
Pembaruan konten	Isi dalam panduan ini telah diperbarui dengan perbaikan kecil dan panduan pemecahan masalah tambahan.	2 Januari 2019

Pembaruan konten	Konten dalam panduan ini telah diperbarui untuk mendukung pengalaman CodeCommit konsol baru.	30 Oktober 2018
CodeCommit dan standar pemrosesan informasi federal (FIPS)	CodeCommit telah menambahkan dukungan untuk standar pemerintah Federal Information Processing Standard (FIPS) Publikasi 140-2 di beberapa daerah. Untuk informasi lebih lanjut tentang FIPS dan titik akhir FIPS yang tersedia, lihat Federal Information Processing Standard (FIPS) 140-2 . Untuk informasi selengkapnya tentang titik akhir koneksi Git, lihat Wilayah .	25 Oktober 2018
CodeCommit menambahkan tiga API	CodeCommit telah menambahkan tiga API untuk mendukung bekerja dengan file. Untuk informasi selengkapnya tentang titik akhir koneksi Git, lihat Izin untuk Tindakan pada File Individu dan Referensi API AWS CodeCommit .	27 September 2018
CodeCommit pemberitahuan riwayat dokumentasi tersedia melalui umpan RSS	Anda sekarang dapat menerima pemberitahuan tentang pembaruan CodeCommit dokumentasi dengan berlangganan umpan RSS.	29 Juni 2018

Pembaruan lebih awal

Tabel berikut menjelaskan perubahan penting pada dokumentasi sebelum tanggal 29 Juni 2018.

Perubahan	Deskripsi	Tanggal diubah
Topik baru	Topik Batasi dorongan dan penggabungan ke cabang telah ditambahkan. Topik Referensi izin CodeCommit telah diperbarui.	16 Mei 2018
Bagian baru	Bagian Bekerja dengan file repositori AWS CodeCommit telah ditambahkan. Topik Referensi izin CodeCommit dan Memulai dengan AWS CodeCommit telah diperbarui.	21 Februari 2018
Topik baru	Topik Konfigurasi akses lintas akun ke AWS CodeCommit repositori menggunakan peran telah ditambahkan.	21 Februari 2018
Topik baru	Topik Integrasi AWS Cloud9 dengan AWS CodeCommit telah ditambahkan. Topik Integrasi produk dan layanan telah diperbarui dengan informasi tentang AWS Cloud9.	1 Desember 2017
Bagian baru	Bagian Bekerja dengan permintaan pull di repositori AWS CodeCommit telah ditambahkan. Bagian Kontrol autentikasi dan akses untuk AWS CodeCommit telah diperbarui dengan informasi tentang izin untuk permintaan pull dan komentar. Ini juga mencakup laporan kebijakan terkelola yang diperbarui.	20 November 2017
Topik yang diperbarui	Integrasi produk dan layanan Topik telah diperbarui untuk menyertakan tautan bagi pelanggan yang ingin memperbaiki saluran pipa mereka yang ada untuk menggunakan Amazon CloudWatch Events untuk memulai pipeline sebagai respons terhadap perubahan dalam repositori. CodeCommit	11 Oktober 2017

Perubahan	Deskripsi	Tanggal diubah
Topik baru	Bagian Kontrol autentikasi dan akses untuk AWS CodeCommit telah ditambahkan. Ini menggantikan topik Referensi Izin Akses.	11 September 2017
Topik yang diperbarui	Bagian Kelola pemicu untuk repositori telah diperbarui untuk mencerminkan perubahan dalam konfigurasi pemicu. Topik dan gambar telah diperbarui di seluruh panduan untuk mencerminkan perubahan di bilah navigasi.	29 Agustus 2017
Topik baru	Topik Bekerja dengan preferensi pengguna telah ditambahkan. Topik Tampilkan detail tag telah diperbarui. Integrasi produk dan layanan Topik telah diperbarui dengan informasi tentang integrasi dengan Amazon CloudWatch Events.	3 Agustus 2017
Topik baru	Topik Mengintegrasikan Eclipse dengan AWS CodeCommit dan Mengintegrasikan Visual Studio dengan AWS CodeCommit telah ditambahkan.	29 Juni 2017
Topik yang diperbarui	CodeCommit Sekarang tersedia di dua wilayah tambahan: Asia Pasifik (Mumbai), dan Kanada (Tengah). Topik Wilayah dan titik akhir koneksi Git telah diperbarui.	29 Juni 2017
Topik yang diperbarui	CodeCommit Sekarang tersedia di empat wilayah tambahan: Asia Pasifik (Seoul), Amerika Selatan (São Paulo), AS Barat (California N.), dan Eropa (London). Topik Wilayah dan titik akhir koneksi Git telah diperbarui.	6 Juni 2017
Topik yang diperbarui	CodeCommit Sekarang tersedia di empat wilayah tambahan: Asia Pasifik (Tokyo), Asia Pasifik (Singapura), Asia Pasifik (Sydney), dan Eropa (Frankfurt). Wilayah dan titik akhir koneksi Git Topik telah diperbarui untuk memberikan informasi tentang titik akhir koneksi Git dan wilayah yang didukung untuk CodeCommit.	25 Mei 2017

Perubahan	Deskripsi	Tanggal diubah
Topik baru	Topik Bandingkan dan gabungkan cabang telah ditambahkan. Isi Bekerja dengan cabang bagian telah diperbarui dengan informasi tentang penggunaan CodeCommit konsol untuk bekerja dengan cabang di repositori.	18 Mei 2017
Topik baru	Topik Bandingkan commit telah ditambahkan dengan informasi tentang membandingkan commit. Struktur panduan pengguna telah diperbarui untuk bekerja dengan repositori , commit , dan cabang .	28 Maret 2017
Topik yang diperbarui	Topik Tampilkan detail commit telah diperbarui dengan informasi tentang melihat perbedaan antara commit dan induknya di konsol, dan menggunakan perintah <code>get-diff</code> untuk menampilkan perbedaan antara commit dengan menggunakan AWS CLI.	24 Januari 2017
Topik baru	Mencatat panggilan API AWS CodeCommit dengan AWS CloudTrail Topik telah ditambahkan dengan informasi tentang log koneksi untuk CodeCommit menggunakan AWS CloudFormation.	11 Januari 2017
Topik baru	Untuk pengguna HTTPS menggunakan kredensial Git Topik telah ditambahkan dengan informasi tentang pengaturan koneksi untuk CodeCommit menggunakan kredensi Git melalui HTTPS.	22 Desember 2016
Topik yang diperbarui	Topik Integrasi produk dan layanan telah diperbarui untuk menyertakan informasi tentang integrasi dengan AWS CodeBuild.	5 Desember 2016
Topik yang diperbarui	CodeCommit sekarang tersedia di wilayah lain, Eropa (Irlandia). Wilayah dan titik akhir koneksi Git Topik telah diperbarui untuk memberikan informasi tentang titik akhir koneksi Git dan wilayah yang didukung untuk CodeCommit.	16 Nopember 2016

Perubahan	Deskripsi	Tanggal diubah
Topik yang diperbarui	CodeCommit sekarang tersedia di wilayah lain, US West (Oregon). Wilayah dan titik akhir koneksi Git Topik telah diperbarui untuk memberikan informasi tentang titik akhir koneksi Git dan wilayah yang didukung untuk CodeCommit.	14 November 2016
Topik baru	Buatlah sebuah pemicu untuk sebuah fungsi Lambda Topik telah diperbarui untuk mencerminkan kemampuan untuk membuat CodeCommit pemicu sebagai bagian dari pembuatan fungsi Lambda. Proses yang disederhanakan ini menyederhanakan pembuatan pemicu dan secara otomatis mengonfigurasi pemicu dengan izin yang diperlukan CodeCommit untuk menjalankan fungsi Lambda. Buatlah sebuah pemicu untuk sebuah fungsi Lambda yang sudah ada Topik telah ditambahkan untuk menyertakan informasi tentang membuat pemicu untuk fungsi Lambda yang ada di CodeCommit konsol.	19 Oktober 2016
Topik baru	CodeCommit sekarang tersedia di wilayah lain, AS Timur (Ohio). Wilayah dan titik akhir koneksi Git Topik telah ditambahkan untuk memberikan informasi tentang titik akhir koneksi Git dan wilayah yang didukung untuk CodeCommit.	17 Oktober 2016
Pembaruan topik	Topik Integrasi produk dan layanan telah diperbarui untuk menyertakan informasi tentang integrasi dengan AWS Elastic Beanstalk.	13 Oktober 2016
Pembaruan topik	Topik Integrasi produk dan layanan telah diperbarui untuk menyertakan informasi tentang integrasi dengan AWS CloudFormation.	6 Oktober 2016
Pembaruan topik	Topik Untuk koneksi SSH pada Windows telah direvisi untuk memberikan panduan untuk menggunakan emulator Bash untuk koneksi SSH pada Windows sebagai ganti seperangkat alat PuTTY.	29 September 2016

Perubahan	Deskripsi	Tanggal diubah
Pembaruan topik	Memulai dengan CodeCommit Topik Tampilkan detail commit dan telah diperbarui untuk menyertakan informasi tentang Visualizer Komit di konsol. CodeCommit Topik Kuota telah diperbarui dengan peningkatan hingga jumlah referensi yang diperbolehkan dalam satu push.	14 September 2016
Pembaruan topik	Memulai dengan CodeCommit Topik Tampilkan detail commit dan telah diperbarui untuk menyertakan informasi tentang melihat riwayat komit di CodeCommit konsol.	28 Juli 2016
Topik baru	Topik Memigrasikan repositori Git ke AWS CodeCommit dan Memigrasi konten lokal atau tanpa versi ke AWS CodeCommit telah ditambahkan.	29 Juni 2016
Pembaruan topik	Pembaruan kecil telah dibuat untuk topik Memecahkan masalah dan Untuk sambungan HTTPS pada Windows dengan pembantu kredensial AWS CLI .	22 Juni 2016
Pembaruan topik	Topik Referensi Integrasi produk dan layanan dan Izin Akses telah diperbarui untuk menyertakan informasi tentang integrasi dengan CodePipeline.	18 April 2016
Topik baru	Bagian Kelola pemicu untuk repositori telah ditambahkan. Topik baru mencakup contoh, termasuk contoh kebijakan dan kode, tentang cara membuat, mengedit, dan menghapus pemicu.	7 Maret 2016
Topik baru	Topik Integrasi produk dan layanan telah ditambahkan. Pembaruan kecil telah dibuat untuk Memecahkan masalah .	7 Maret 2016
Pembaruan topik	Selain sidik jari server MD5, sidik jari server SHA256 untuk CodeCommit telah ditambahkan ke Untuk koneksi SSH di Linux, macOS, atau Unix dan Untuk koneksi SSH pada Windows	9 Desember 2015

Perubahan	Deskripsi	Tanggal diubah
Topik baru	Topik Menelusuri file dalam sebuah repositori telah ditambahkan. Masalah baru telah ditambahkan ke Memecahkan masalah . Peningkatan dan perbaikan kecil telah dilakukan di seluruh panduan pengguna.	5 Oktober 2015
Topik baru	Topik Untuk pengguna SSH yang tidak menggunakan AWS CLI telah ditambahkan. Topik di bagian Pengaturan telah disederhanakan. Panduan untuk membantu pengguna menentukan langkah-langkah yang harus diikuti untuk sistem operasi mereka dan protokol pilihan telah disediakan.	5 Agustus 2015
Pembaruan topik	Klarifikasi dan contoh telah ditambahkan ke langkah-langkah ID kunci SSH di SSH dan Linux, macOS, atau Unix: Siapkan kunci publik dan privat untuk Git dan CodeCommit dan Langkah 3: Siapkan kunci publik dan privat untuk Git dan CodeCommit .	24 Juli 2015
Pembaruan topik	Langkah-langkah dalam Langkah 3: Siapkan kunci publik dan privat untuk Git dan CodeCommit telah diperbarui untuk mengatasi masalah dengan IAM dan menyimpan file kunci publik.	22 Juli 2015
Pembaruan topik	Memecahkan masalah telah diperbarui dengan alat bantu navigasi. Informasi pemecahan masalah lebih lanjut untuk masalah rantai kunci kredensial telah ditambahkan.	20 Juli 2015
Pembaruan topik	Informasi lebih lanjut tentang izin AWS Key Management Service telah ditambahkan ke AWS KMS dan enkripsi dan topik Referensi Izin Akses.	17 Juli 2015
Pembaruan topik	Bagian lain telah ditambahkan ke Memecahkan masalah dengan informasi tentang pemecahan masalah dengan AWS Key Management Service.	10 Juli 2015

Perubahan	Deskripsi	Tanggal diubah
Rilis awal	Ini adalah rilis awal Panduan Pengguna CodeCommit .	9 Juli 2015

AWSGlosarium

Untuk AWS terminologi terbaru, lihat [AWSglosarium di Referensi](#). Glosarium AWS

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.