



사용 설명서

Amazon Virtual Private Cloud



Amazon Virtual Private Cloud: 사용 설명서

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

Amazon의 상표 및 브랜드 디자인은 Amazon 외 제품 또는 서비스와 함께, Amazon 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계와 관계없이 해당 소유자의 자산입니다.

Table of Contents

Amazon VPC란 무엇인가?	1
특성	1
Amazon VPC 시작하기	2
Amazon VPC 작업	3
Amazon VPC 요금	3
Amazon VPC 작동 방식	5
VPC 및 서브넷	6
기본 VPC와 기본이 아닌 VPC	6
라우팅 테이블	7
인터넷 액세스	7
회사 또는 홈 네트워크에 액세스	8
VPC 및 네트워크 연결	8
AWS 프라이빗 글로벌 네트워크	9
시작	10
AWS 계정 가입	10
권한 확인	11
IP 주소 범위 설명	11
가용 영역 선택	11
인터넷 연결 계획	11
VPC 생성	12
애플리케이션 배포	12
IP 주소 지정	14
IPv4 및 IPv6 비교	15
프라이빗 IPv4 주소	16
퍼블릭 IPv4 주소	16
IPv6 주소	17
고유 IP 주소 사용	19
Amazon VPC IP 주소 관리자 사용	19
VPC CIDR 블록	19
IPv4 VPC CIDR 블록	19
VPC에 IPv4 CIDR 블록 관리	20
IPv4 CIDR 블록 연결 제한	23
IPv6 VPC CIDR 블록	24
서브넷 CIDR 블록	25

IPv4에 대한 서브넷 크기 조정	26
IPv6에 대한 서브넷 크기 조정	26
관리형 접두사 목록	27
접두사 목록 개념 및 규칙	28
접두사 목록에 대한 Identity and Access Management	29
고객 관리형 접두사 목록	30
AWS 관리형 접두사 목록	35
공유 접두사 목록	36
AWS 리소스의 참조 접두사 목록	40
AWS IP 주소 범위	42
다운로드	43
구문	43
범위 중복	46
JSON 파일 필터링	46
송신 제어 구현	50
AWS IP 주소 범위 알림	50
릴리스 정보	52
자세히 알아보기	54
VPC에 IPv6 지원 추가	54
예: 퍼블릭 및 프라이빗 서브넷이 있는 VPC에서 IPv6 사용	55
1단계: IPv6 CIDR 블록을 VPC 및 서브넷에 연결	58
2단계: 라우팅 테이블 업데이트	59
3단계: 보안 그룹 규칙 업데이트	60
4단계: 인스턴스에 IPv6 주소 할당	61
AWS에서 IPv6 지원	61
IPv6를 지원하는 서비스	62
추가 IPv6 지원	68
자세히 알아보기	69
Virtual Private Cloud	70
VPC 기초	70
VPC IP 주소 범위	70
VPC 다이어그램	71
VPC 리소스	71
기본 VPC	72
기본 VPC 구성 요소	72
기본 서브넷	75

기본 VPC와 기본 서브넷 보기	75
기본 VPC 생성	76
기본 서브넷 만들기	77
기본 서브넷과 기본 VPC 삭제	79
VPC 생성	79
VPC 구성 옵션	79
VPC 및 기타 VPC 리소스 생성	81
VPC만 생성	82
AWS CLI를 사용하여 VPC 생성	84
VPC 구성	89
VPC에 대한 세부 정보 보기	89
VPC의 리소스 시각화	90
IPv4 CIDR 블록 추가	91
IPv6 CIDR 블록 추가	92
IPv4 CIDR 블록 제거	93
IPv6 CIDR 블록 제거	94
DHCP 옵션 세트	94
DHCP란 무엇인가요?	95
DHCP 옵션 세트 개념	96
DHCP 옵션 세트로 작업	99
DNS 속성	104
Amazon DNS 서버	104
DNS 호스트 이름	105
VPC의 DNS 속성	106
DNS 할당량	108
EC2 인스턴스의 DNS 호스트 이름 보기	108
VPC에 대한 DNS 속성 보기 및 업데이트	109
프라이빗 호스팅 영역	110
네트워크 주소 사용량	111
NAU를 계산하는 방법	112
NAU 예시	113
VPC 공유	114
공유 VPC 사전 조건	115
서브넷 공유	115
공유 서브넷 공유 해제	116
공유 서브넷의 소유자 식별	117

VPC 리소스 관리	117
소유자 및 참가자에 대한 책임 및 권한	118
AWS 리소스 및 공유 VPC 서브넷	120
VPC 공유 할당량	121
서브넷 공유의 예	121
다른 영역으로 VPC 확장	123
AWS Local Zones의 서브넷	123
AWS Wavelength의 서브넷	128
AWS Outposts의 서브넷	131
VPC 삭제	132
콘솔을 사용하여 삭제	133
CLI를 사용하여 삭제	133
서브넷	136
서브넷 기본 사항	136
서브넷 IP 주소 범위	136
서브넷 유형	137
서브넷 다이어그램	137
서브넷 라우팅	138
서브넷 설정	138
서브넷 보안	139
서브넷 생성	139
서브넷 구성	141
서브넷 보기	141
서브넷에 IPv6 CIDR 블록 추가	142
서브넷에서 IPv6 CIDR 블록 제거	142
서브넷의 퍼블릭 IPv4 주소 지정 속성 수정	143
서브넷의 IPv6 주소 지정 속성 수정	143
서브넷 CIDR 예약	144
콘솔을 사용한 서브넷 CIDR 예약 작업	145
AWS CLI를 사용한 서브넷 CIDR 예약 작업	145
라우팅 테이블	146
라우팅 테이블 개념	147
서브넷 라우팅 테이블	148
게이트웨이 라우팅 테이블	155
라우팅 우선순위	158
라우팅 테이블 할당량	160

연결 문제 해결	160
라우팅 옵션 예	160
라우팅 테이블 작업	175
미들박스 라우팅 마법사	184
서브넷 삭제	197
VPC 연결	199
인터넷 게이트웨이	200
인터넷 액세스 구성	200
인터넷 게이트웨이 작업	203
API 및 명령 개요	205
요금	206
외부 전용 인터넷 게이트웨이	206
외부 전용 인터넷 게이트웨이 기본 사항	207
송신 전용 인터넷 게이트웨이 작업	208
API 및 CLI 개요	210
요금	211
NAT 디바이스	211
NAT 게이트웨이	212
NAT 인스턴스	256
NAT 디바이스 비교	267
탄력적 IP 주소	269
탄력적 IP 주소 개념 및 규칙	270
탄력적 IP 주소 작업	271
요금	280
AWS Transit Gateway	280
AWS Virtual Private Network	281
VPC 피어링 연결	282
모니터링	283
VPC 흐름 로그	284
흐름 로그 기본 사항	285
흐름 로그 레코드	288
흐름 로그 레코드의 예	298
흐름 로그 제한	306
요금	308
흐름 로그 작업	308
CloudWatch Logs에 게시	312

Amazon S3에 게시	320
Amazon Data Firehose에 게시	328
Athena를 사용하여 쿼리	334
문제 해결	339
CloudWatch 지표	342
NAU 지표 및 차원	342
NAU 모니터링 활성화 또는 비활성화	345
NAU CloudWatch 경보 예시	346
보안	347
데이터 보호	348
인터넷워크 트래픽 개인 정보	348
ID 및 액세스 관리	349
고객	350
ID로 인증	350
정책을 사용하여 액세스 관리	353
Amazon VPC가 IAM과 작동하는 방식	355
정책 예	359
문제 해결	369
AWS 관리형 정책	371
인프라 보안	373
네트워크 격리	374
네트워크 트래픽 제어	374
보안 그룹 및 네트워크 ACL 비교	375
보안 그룹	376
보안 그룹 기본 사항	377
보안 그룹 예시	379
보안 그룹 규칙	380
기본 보안 그룹	390
보안 그룹 작업	392
네트워크 ACL	396
네트워크 ACL 기본 사항	397
네트워크 ACL 규칙	399
기본 네트워크 ACL	399
사용자 지정 네트워크 ACL	401
사용자 지정 네트워크 ACL 및 기타 AWS 서비스	408
취발성 포트	408

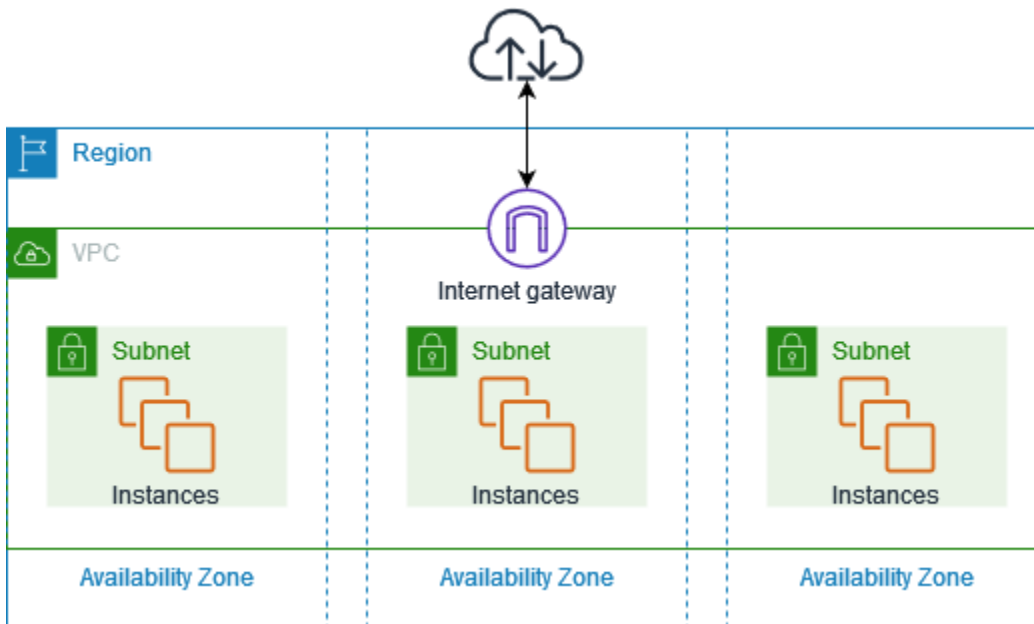
경로 MTU 검색	409
네트워크 ACL 작업	409
예: 서브넷의 인스턴스에 대한 액세스 제어	415
연결 문제 해결	418
복원력	419
규정 준수 확인	419
모범 사례	420
다른 서비스와 함께 사용	422
AWS PrivateLink	422
AWS Network Firewall	423
Route 53 Resolver DNS Firewall	424
Reachability Analyzer	425
예제	427
테스트 환경	427
개요	428
VPC 생성	430
애플리케이션 배포	431
구성 테스트	431
정리	431
웹 및 데이터베이스 서버	431
개요	432
VPC 생성	435
애플리케이션 배포	437
구성 테스트	437
정리	437
프라이빗 서버	437
개요	438
VPC 생성	441
애플리케이션 배포	442
구성 테스트	442
정리	442
할당량	443
VPC 및 서브넷	443
DNS	443
탄력적 IP 주소	444
게이트웨이	444

고객 관리형 접두사 목록	445
네트워크 ACL	446
네트워크 인터페이스	446
라우팅 테이블	447
보안 그룹	447
VPC 공유	448
네트워크 주소 사용량	449
Amazon EC2 API 조절	449
추가 할당량 리소스	449
문서 기록	451

Amazon VPC란 무엇인가?

Amazon Virtual Private Cloud(Amazon VPC)를 사용하면 정의한 논리적으로 격리된 가상 네트워크에서 AWS 리소스를 시작할 수 있습니다. 이 가상 네트워크는 AWS의 확장 가능한 인프라를 사용한다는 이점과 함께 고객의 자체 데이터 센터에서 운영하는 기존 네트워크와 매우 유사합니다.

다음 다이어그램에서는 VPC 예를 보여 줍니다. VPC에는 리전의 각 가용성 영역에 하나의 서브넷이 있고, 각 서브넷에 EC2 인스턴스가 있고, VPC의 리소스와 인터넷 간의 통신을 허용하는 인터넷 게이트웨이가 있습니다.



자세한 내용은 [Amazon Virtual Private Cloud\(Amazon VPC\)](#)를 참조하십시오.

특성

다음 기능은 애플리케이션에 필요한 연결을 제공하도록 VPC 구성하는 데 도움이 됩니다.

Virtual Private Cloud(VPC)

[VPC](#)는 자체 데이터 센터에서 운영하는 기존 네트워크와 아주 유사한 가상 네트워크입니다. VPC를 생성한 후 서브넷을 추가할 수 있습니다.

서브넷

[서브넷](#)은 VPC의 IP 주소 범위입니다. 서브넷은 단일 가용 영역에 상주해야 합니다. 서브넷을 추가한 후에는 VPC에 AWS 리소스 배포할 수 있습니다.

IP 주소 지정

VPC와 서브넷에 [IP 주소](#)를 IPv4와 IPv6 모두 할당할 수 있습니다. 또한 퍼블릭 IPv4 주소 및 IPv6 GUA 주소를 AWS로 가져오고 VPC의 리소스(예: EC2 인스턴스, NAT 게이트웨이, Network Load Balancer)에 할당할 수 있습니다.

라우팅

[라우팅 테이블](#)을 사용하여 서브넷 또는 게이트웨이의 네트워크 트래픽이 전달되는 위치를 결정합니다.

게이트웨이 및 엔드포인트

[게이트웨이](#)는 VPC를 다른 네트워크에 연결합니다. 예를 들면, [인터넷 게이트웨이](#)를 사용하여 VPC를 인터넷에 연결합니다. [VPC 엔드포인트](#)를 사용하여 인터넷 게이트웨이 또는 NAT 장치를 사용하지 않고 AWS 서비스에 비공개로 연결합니다.

피어링 연결

[VPC 피어링 연결](#)을 사용하여 두 VPC의 리소스 간 트래픽을 라우팅합니다.

트래픽 미러링

네트워크 인터페이스에서 [네트워크 트래픽을 복사](#)하고 심층 패킷 검사를 위해 보안 및 모니터링 어플라이언스로 전송합니다.

Transit Gateway

중앙 허브 역할을 하는 [전송 게이트웨이](#)를 사용하여 VPC, VPN 연결 및 AWS Direct Connect 연결 간에 트래픽을 라우팅합니다.

VPC 흐름 로그

[흐름 로그](#)는 VPC의 네트워크 인터페이스로 들어오고 나가는 IP 트래픽에 대한 정보를 캡처합니다.

VPN 연결

[AWS Virtual Private Network\(AWS VPN\)](#)을 사용하여 온프레미스 네트워크에 VPC를 연결합니다.

Amazon VPC 시작하기

AWS 계정의 각 [에](#)는 기본 VPCAWS 리전가 있습니다. 기본 VPC는 EC2 인스턴스 시작 및 연결을 즉시 시작할 수 있도록 구성되어 있습니다. 자세한 정보는 [시작](#)를 참조하십시오.

필요한 서브넷, IP 주소, 게이트웨이 및 라우팅으로 추가 VPC를 생성하도록 선택할 수 있습니다. 자세한 내용은 [the section called “VPC 생성”](#) 단원을 참조하십시오.

Amazon VPC 작업

다음 인터페이스 중 하나를 사용하여 VPC를 생성하고 관리할 수 있습니다.

- AWS Management Console — VPC에 액세스할 때 사용할 수 있는 웹 인터페이스를 제공합니다.
- AWS Command Line Interface(AWS CLI) - Amazon VPC를 포함한 다양한 AWS 서비스에서 사용되는 명령을 제공하며 Windows, macOS 및 Linux에서 지원됩니다. 자세한 내용은 [AWS Command Line Interface](#)를 참조하세요.
- AWS SDK — 언어별 API를 제공하고, 서명 계산, 요청 재시도 처리 및 오류 처리와 같은 많은 연결 세부 정보를 관리합니다. 자세한 정보는 [AWS SDK](#)를 참조하세요.
- 쿼리 API — HTTPS 요청을 사용하여 호출하는 하위 수준의 API 작업을 제공합니다. 쿼리 API 사용이 Amazon VPC에 액세스하는 가장 직접적인 방법이지만, 애플리케이션에서 요청에 서명할 해시 생성 및 오류 처리와 같은 하위 수준의 세부 정보를 처리해야 합니다. 자세한 내용은 Amazon EC2 API Reference(Amazon EC2 API 참조)의 [Amazon VPC actions](#)(Amazon VPC 작업)를 참조하세요.

Amazon VPC 요금

VPC 사용에 따르는 추가 요금은 없습니다. 단 NAT 게이트웨이, IP 주소 관리자, 트래픽 미러링, Reachability Analyzer, Network Access Analyzer와 같은 일부 VPC 구성 요소에 대해 요금이 부과됩니다. 자세한 내용은 [Amazon VPC 요금](#)을 참조하세요.

Virtual Private Cloud(VPC)에서 시작하는 거의 모든 리소스는 연결을 위한 IP 주소와 함께 제공됩니다. VPC에 있는 대부분의 리소스는 프라이빗 IPv4 주소를 사용합니다. 하지만 IPv4를 통해 인터넷에 직접 액세스해야 하는 리소스는 퍼블릭 IPv4 주소를 사용합니다.

퍼블릭 IPv4 주소 요금

퍼블릭 IPv4 주소는 인터넷에서 라우팅할 수 있는 IPv4 주소입니다. 퍼블릭 IPv4 주소는 인터넷에서 IPv4를 통해 리소스에 직접 연결하는 데 필요합니다.

기존 또는 신규 [AWS 프리 티어](#) 고객은 750시간 동안 퍼블릭 IPv4 주소를 무료로 사용할 수 있습니다. AWS 프리 티어를 사용하지 않는 경우에는 퍼블릭 IPv4 주소에 요금이 부과됩니다. 구체적인 요금 정보는 [Amazon VPC 요금](#)의 퍼블릭 IPv4 주소 탭을 참조하세요.

프라이빗 IPv4 주소([RFC 1918](#))에는 요금이 부과되지 않습니다. 공유 VPC의 퍼블릭 IPv4 주소에 요금이 부과되는 방식에 대한 자세한 내용은 [소유자 및 참여자에 대한 청구 및 측정](#)을 참조하세요.

퍼블릭 IPv4 주소에는 다음 유형이 있습니다.

- 탄력적 IP 주소(EIP): EC2 인스턴스, 탄력적 네트워크 인터페이스 또는 AWS 리소스와 연결할 수 있는 Amazon에서 제공하는 정적 퍼블릭 IPv4 주소입니다.
- EC2 퍼블릭 IPv4 주소: Amazon에서 EC2 인스턴스에 할당한 퍼블릭 IPv4 주소입니다(EC2 인스턴스가 기본 서브넷에서 시작되거나 인스턴스가 퍼블릭 IPv4 주소를 자동으로 할당하도록 구성된 서브넷에서 시작되는 경우).
- BYOIPv4 주소: [고유 IP 주소 가져오기\(BYOIP\)](#)를 사용하여 AWS로 가져온 IPv4 주소 범위의 퍼블릭 IPv4 주소입니다.
- 서비스 관리형 IPv4 주소: 퍼블릭 IPv4 주소는 AWS 리소스에 자동으로 프로비저닝되고 AWS 서비스에 의해 관리됩니다. 예를 들어 Amazon ECS, Amazon RDS 또는 Amazon WorkSpaces의 퍼블릭 IPv4 주소입니다.

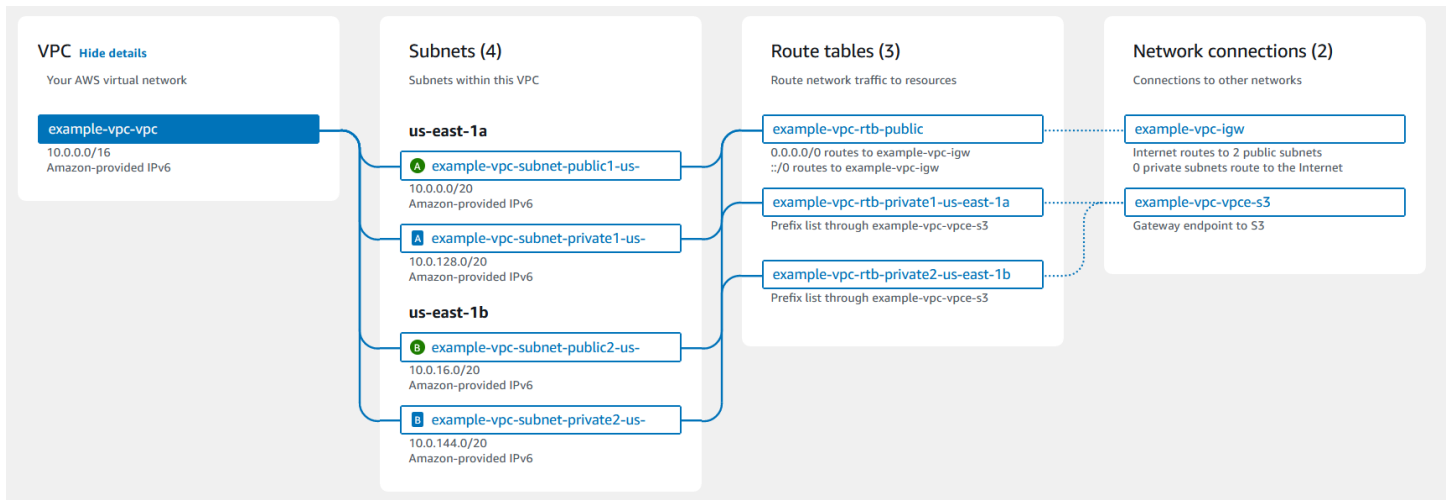
다음 목록은 퍼블릭 IPv4 주소를 사용할 수 있는 가장 일반적인 AWS 서비스를 보여줍니다.

- Amazon AppStream 2.0
- [AWS Client VPN](#)
- AWS Database Migration Service
- Amazon EC2
- Amazon Elastic Container Service
- Amazon EKS
- Amazon EMR
- Amazon GameLift
- AWS Global Accelerator
- AWS Mainframe Modernization
- Amazon Managed Streaming for Apache Kafka
- Amazon MQ
- Amazon RDS
- Amazon Redshift
- AWS Site-to-Site VPN
- Amazon VPC NAT 게이트웨이
- Amazon WorkSpaces
- Elastic Load Balancing

Amazon VPC 작동 방식

Amazon Virtual Private Cloud(Amazon VPC)를 사용하면 정의한 논리적으로 격리된 가상 네트워크에서 AWS 리소스를 시작할 수 있습니다. 이 가상 네트워크는 AWS의 확장 가능한 인프라를 사용한다는 이점과 함께 고객의 자체 데이터 센터에서 운영하는 기존 네트워크와 매우 유사합니다.

다음은 AWS Management Console를 사용하여 VPC를 생성할 때 표시되는 미리 보기 창에서 VPC와 그 리소스를 시각적으로 표현한 것입니다. 기존 VPC의 경우 [리소스 맵](#) 탭에서 이 시각화에 액세스할 수 있습니다. 이 예제는 VPC 및 기타 네트워킹 리소스 생성을 선택할 때 VPC 생성 페이지에서 처음 선택되는 리소스를 보여 줍니다. 이 VPC는 IPv4 CIDR 및 Amazon에서 제공하는 IPv6 CIDR, 2개의 가용 영역에 있는 서브넷, 3개의 라우팅 테이블, 인터넷 게이트웨이 및 게이트웨이 엔드포인트로 구성됩니다. 인터넷 게이트웨이를 선택했기 때문에 해당 라우팅 테이블이 트래픽을 인터넷 게이트웨이로 전송하므로 퍼블릭 서브넷의 트래픽이 인터넷으로 라우팅되는 것이 시각화에 표시됩니다.



개념

- [VPC 및 서브넷](#)
- [기본 VPC와 기본이 아닌 VPC](#)
- [라우팅 테이블](#)
- [인터넷 액세스](#)
- [회사 또는 홈 네트워크에 액세스](#)
- [VPC 및 네트워크 연결](#)
- [AWS 프라이빗 글로벌 네트워크](#)

VPC 및 서브넷

Virtual Private Cloud(VPC)는 사용자의 AWS 계정 전용 가상 네트워크입니다. VPC는 AWS 클라우드에서 다른 가상 네트워크와 논리적으로 분리되어 있습니다. VPC의 IP 주소 범위를 지정하고 서브넷과 게이트웨이를 추가하고 보안 그룹을 연결합니다.

서브넷은 VPC의 IP 주소 범위입니다. Amazon EC2 인스턴스와 같은 AWS 리소스를 서브넷으로 실행할 수 있습니다. 서브넷을 인터넷, 다른 VPC 및 자체 데이터 센터에 연결하고 라우팅 테이블을 사용하여 서브넷으로/서브넷에서 트래픽을 라우팅할 수 있습니다.

자세히 알아보기

- [IP 주소 지정](#)
- [Virtual Private Cloud](#)
- [서브넷](#)

기본 VPC와 기본이 아닌 VPC

계정이 2013년 12월 4일 이후에 생성된 경우 각 리전에 기본 VPC가 함께 제공됩니다. 기본 VPC가 구성되어 사용할 준비가 되었습니다. 예를 들어, 리전의 각 가용 영역에 기본 서브넷, 연결된 인터넷 게이트웨이, 모든 트래픽을 인터넷 게이트웨이로 보내는 기본 라우팅 테이블의 경로, 퍼블릭 IP 주소가 있는 인스턴스에 퍼블릭 DNS 호스트 이름을 자동으로 할당하고 Amazon 제공 DNS 서버를 통해 DNS 확인을 활성화하는 DNS 설정이 있습니다([VPC의 DNS 속성](#) 섹션 참조). 따라서 기본 서브넷에서 시작된 EC2 인스턴스는 자동으로 인터넷에 액세스할 수 있습니다. 리전에 기본 VPC가 있고 해당 리전에서 EC2 인스턴스를 시작할 때 서브넷을 지정하지 않으면 기본 서브넷 중 하나가 선택되고 해당 서브넷에서 인스턴스가 시작됩니다.

자체 VPC를 생성하고 필요에 따라 구성할 수도 있습니다. 이를 기본이 아닌 VPC라고 합니다. 기본이 아닌 VPC에 만든 서브넷과 기본 VPC에 만든 추가 서브넷은 기본이 아닌 서브넷이라고 합니다.

자세히 알아보기

- [the section called “기본 VPC”](#)
- [the section called “VPC 생성”](#)

라우팅 테이블

라우팅 테이블에는 VPC의 네트워크 트래픽을 전달할 위치를 결정하는 데 사용되는 라우팅이라는 규칙 집합이 포함되어 있습니다. 서브넷을 특정 라우팅 테이블과 명시적으로 연결할 수 있습니다. 그렇지 않으면 서브넷이 기본 라우팅 테이블과 암시적으로 연결됩니다.

라우팅 테이블의 각 라우팅은 트래픽을 전달할 IP 주소 범위(대상 주소)와 트래픽을 전송할 게이트웨이, 네트워크 인터페이스 또는 연결(대상)을 지정합니다.

자세히 알아보기

- [라우팅 테이블 구성](#)

인터넷 액세스

VPC에서 시작한 인스턴스가 VPC 외부의 리소스를 어떻게 액세스할지를 제어할 수 있습니다.

기본 VPC에는 인터넷 게이트웨이가 포함되며, 각각의 기본 서브넷은 퍼블릭 서브넷입니다. 기본 서브넷에서 시작한 각 인스턴스에는 프라이빗 IPv4 주소와 퍼블릭 IPv4 주소가 있습니다. 이러한 인스턴스는 인터넷 게이트웨이를 통해 인터넷과 통신할 수 있습니다. 인터넷 게이트웨이를 통해 인스턴스는 Amazon EC2 네트워크 엣지를 통해 인터넷에 연결할 수 있습니다.

기본적으로 기본이 아닌 서브넷에서 시작한 각 인스턴스에는 프라이빗 IPv4 주소가 있으며, 시작 시 특별히 지정하거나 서브넷의 퍼블릭 IP 주소 속성을 수정하지 않는 한 퍼블릭 IPv4 주소는 없습니다. 이러한 인스턴스는 서로 통신할 수는 있지만 인터넷에 액세스할 수는 없습니다.

기본이 아닌 서브넷에서 시작한 인스턴스에 대해 해당 VPC에 인터넷 게이트웨이를 추가하고(해당 VPC가 기본 VPC가 아닐 경우) 인스턴스에 탄력적 IP 주소를 연결하여 인터넷 액세스를 가능하게 할 수 있습니다.

또는 VPC의 인스턴스가 인터넷으로 아웃바운드 연결을 시작할 수 있도록 하지만 인터넷으로부터의 원치 않는 인바운드 연결은 차단하려면 네트워크 주소 변환(NAT) 디바이스를 사용하면 됩니다. NAT는 여러 개의 프라이빗 IPv4 주소를 하나의 퍼블릭 IPv4 주소에 매핑합니다. 탄력적 IP 주소로 NAT 디바이스를 구성하고 인터넷 게이트웨이를 통해 인터넷에 연결할 수 있습니다. 프라이빗 서브넷의 인스턴스를 NAT 디바이스를 통해 인터넷에 연결할 수 있으며, 이렇게 하면 인스턴스의 트래픽이 인터넷 게이트웨이로 라우팅되고, 모든 응답은 인스턴스로 라우팅됩니다.

IPv6 CIDR 블록을 VPC와 연결하고 인스턴스에 IPv6 주소를 할당하면 인스턴스가 IPv6로 인터넷 게이트웨이를 통해 인터넷에 연결할 수 있습니다. 또는 인스턴스는 외부 전용 인터넷 게이트웨이를 사용

하여 IPv6를 통해 인터넷에 대한 아웃바운드 연결을 시작할 수 있습니다. IPv6 트래픽은 IPv4 트래픽에서 분리되어 있으므로, 라우팅 테이블에는 IPv6 트래픽에 대한 별도의 경로가 포함되어야 합니다.

자세히 알아보기

- [인터넷 게이트웨이를 사용하여 인터넷에 연결](#)
- [송신 전용 인터넷 게이트웨이를 사용하여 아웃바운드 IPv6 트래픽 활성화](#)
- [NAT 디바이스를 사용하여 인터넷 또는 다른 네트워크에 연결](#)

회사 또는 홈 네트워크에 액세스

원할 경우 IPsec AWS Site-to-Site VPN 연결을 사용하여 VPC를 회사의 데이터 센터에 연결함으로써 회사 데이터 센터를 AWS 클라우드로 확장할 수 있습니다.

Site-to-Site VPN 연결은 AWS 측의 가상 프라이빗 게이트웨이 또는 전송 게이트웨이와 데이터 센터의 고객 게이트웨이 디바이스 간 두 개의 VPN 터널로 구성됩니다. 고객 게이트웨이는 Site-to-Site VPN 연결에서 고객 측이 구성하는 물리적 디바이스 또는 소프트웨어 애플리케이션입니다.

자세히 알아보기

- [AWS Site-to-Site VPN 사용 설명서](#)
- [Amazon VPC Transit Gateway](#)

VPC 및 네트워크 연결

두 VPC 간에 VPC 피어링 연결을 생성하여 비공개적으로 두 VPC 간에 트래픽을 라우팅할 수 있습니다. 동일한 네트워크에 속하는 경우와 같이 VPC의 인스턴스가 서로 통신할 수 있습니다.

또한 전송 게이트웨이를 생성하고 사용해 VPC와 온프레미스 네트워크를 상호 연결할 수 있습니다. 전송 게이트웨이는 VPC, VPN 연결, AWS Direct Connect 게이트웨이, 전송 게이트웨이 피어링 연결 등 연결 간에 이동하는 트래픽에 대해 리전 가상 라우터 역할을 합니다.

자세히 알아보기

- [Amazon VPC Peering Guide](#)
- [Amazon VPC Transit Gateway](#)

AWS 프라이빗 글로벌 네트워크

AWS는 고객의 네트워킹 요구 사항을 지원하는 안전한 클라우드 컴퓨팅 환경을 제공하기 위해 고성능, 낮은 대기 시간의 프라이빗 글로벌 네트워크를 운영합니다. AWS 리전은 여러 인터넷 서비스 제공업체 (ISP)와 연결되는 것은 물론 프라이빗 글로벌 네트워크 백본과도 연결되어 고객으로부터 전송되는 교차 리전 트래픽을 향상된 네트워크 성능으로 처리합니다.

다음 사항을 고려하십시오.

- 모든 리전에서 특정 가용 영역 내부 또는 가용 영역 간 트래픽은 AWS 프라이빗 글로벌 네트워크를 통해 라우팅됩니다.
- 리전 간 트래픽은 중국 리전을 제외하면 항상 AWS 프라이빗 글로벌 네트워크를 통해 라우팅됩니다.

네트워크 패킷 손실은 네트워크 흐름 충돌, 낮은 수준(계층 2) 오류 및 기타 네트워크 오류를 비롯한 여러 요인으로 인해 발생할 수 있습니다. 패킷 손실이 최소화되도록 네트워크가 엔지니어링되고 운영됩니다. 또한 AWS 리전을 연결하는 글로벌 백본에서 PLR(패킷 손실률)을 측정합니다. 백본 네트워크를 운영하여 0.0001% 미만의 시간당 PLR 중 p99를 목표로 합니다.

Amazon VPC 시작하기

VPC 생성 및 연결을 준비하려면 다음 작업을 완료합니다. 작업을 마치면 AWS에서 애플리케이션을 배포할 준비가 됩니다.

작업

- [AWS 계정 가입](#)
- [권한 확인](#)
- [IP 주소 범위 설명](#)
- [가용 영역 선택](#)
- [인터넷 연결 계획](#)
- [VPC 생성](#)
- [애플리케이션 배포](#)

AWS 계정 가입

AWS 계정이 없는 경우 다음 절차에 따라 계정을 생성합니다.

AWS 계정에 등록하려면

1. <https://portal.aws.amazon.com/billing/signup>을 여세요.
2. 온라인 지시 사항을 따르세요.

등록 절차 중에는 전화를 받고 키패드로 인증 코드를 입력하는 과정이 있습니다.

AWS 계정에 가입하면 AWS 계정 루트 사용자들이 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스 액세스 권한이 있습니다. 보안 모범 사례는 사용자에게 관리 액세스 권한을 할당하고, 루트 사용자만 사용하여 [루트 사용자 액세스 권한이 필요한 작업](#)을 수행하는 것입니다.

AWS는 가입 절차 완료된 후 사용자에게 확인 이메일을 전송합니다. 언제든지 <https://aws.amazon.com/>으로 가서 내 계정(My Account)을 선택하여 현재 계정 활동을 보고 계정을 관리할 수 있습니다.

권한 확인

Amazon VPC 사용하려면 먼저 필요한 권한이 있어야 합니다. 자세한 내용은 [Amazon VPC용 Identity and Access Management](#) 및 [Amazon VPC 정책 예](#) 단원을 참조하세요.

IP 주소 범위 설명

VPC의 리소스는 IP 주소를 사용하여 인터넷을 통해 서로 통신하고 다른 리소스와 통신합니다. VPC와 서브넷을 생성할 때 IP 주소 범위를 선택할 수 있습니다. 서브넷에 EC2 인스턴스와 같은 리소스를 배포하면 서브넷의 IP 주소 범위에서 IP 주소를 받습니다. 자세한 내용은 [IP 주소 지정](#) 단원을 참조하십시오.

VPC의 크기를 선택할 때는 AWS 계정 및 VPC 간에 필요한 IP 주소 수를 고려하세요. VPC의 IP 주소 범위가 자체 네트워크의 IP 주소 범위와 겹치지 않도록 하세요. 여러 VPC 간에 연결이 필요한 경우 겹치는 IP 주소가 없는지 확인해야 합니다.

IP 주소 관리자(IPAM)를 사용하면 애플리케이션의 IP 주소를 보다 쉽게 계획, 추적, 모니터링할 수 있습니다. 자세한 내용은 [IP 주소 관리자 설명서](#)를 참조하세요.

가용 영역 선택

AWS 지역은 가용 영역이라고 하는 데이터 센터를 클러스터링하는 물리적 위치입니다. 각 가용 영역은 중복 전원, 네트워킹 및 연결과 함께 독립된 전원, 냉각 및 물리적 보안이 있습니다. 리전의 가용 영역은 유의미한 거리만큼 물리적으로 분리되어 있으며 지연 시간이 짧은 높은 대역폭 네트워킹을 통해 상호 연결됩니다. 여러 가용 영역에서 실행되도록 애플리케이션을 설계하여 내결함성을 더욱 높일 수 있습니다.

프로덕션 환경

프로덕션 환경의 경우 두 개 이상의 가용 영역을 선택하고 각 활성 가용 영역에서 AWS 리소스를 균등하게 배포하는 것이 좋습니다.

개발 또는 테스트 환경

개발 또는 테스트 환경의 경우 하나의 가용 영역에만 리소스를 배포하여 비용을 절감할 수 있습니다.

인터넷 연결 계획

연결 요구 사항에 따라 각 VPC를 서브넷으로 나눌 계획을 세웁니다. 예:

- 인터넷에 있는 클라이언트로부터 트래픽을 받는 웹 서버가 있는 경우 각 가용 영역에서 해당 서버에 대한 서브넷을 생성합니다.
- 또한 VPC에 있는 다른 서버로부터 트래픽을 받는 서버가 있는 경우 각 가용 영역에서 해당 서버에 대한 별개의 서브넷을 생성합니다.
- 네트워크에 대한 VPN 연결을 통해서만 트래픽을 받는 서버가 있는 경우 각 가용 영역에서 해당 서버에 대한 별개의 서브넷을 생성합니다.

애플리케이션이 인터넷에서 트래픽을 수신하려면 VPC에 인터넷 게이트웨이가 있어야 합니다. VPC에 인터넷 게이트웨이를 연결한다고 해서 인터넷에서 인스턴스에 자동으로 액세스할 수 있는 것은 아닙니다. 인터넷 게이트웨이를 연결하는 것 외에도 서브넷 라우팅 테이블을 인터넷 게이트웨이에 대한 경로로 업데이트해야 합니다. 또한 인스턴스에 퍼블릭 IP 주소와 애플리케이션에 필요한 특정 포트 및 프로토콜을 통한 인터넷의 트래픽을 허용하는 연결된 보안 그룹이 있는지 확인해야 합니다.

또는 인터넷이 연결된 로드 밸런서를 사용하여 인스턴스를 등록합니다. 로드 밸런서는 클라이언트에서 트래픽을 수신하고 해당 트래픽을 하나 이상의 가용 영역에 등록된 인스턴스로 분산합니다. 자세한 내용은 [Elastic Load Balancing](#)을 참조하세요. 인터넷에서 요청하지 않은 인바운드 연결을 허용하지 않고 프라이빗 서브넷의 인스턴스가 인터넷에 액세스(예: 업데이트 다운로드)할 수 있도록 하려면 각 활성 가용 영역에 퍼블릭 NAT 게이트웨이를 추가하고 라우팅 테이블을 업데이트하여 인터넷 트래픽을 NAT 게이트웨이로 보냅니다. 자세한 내용은 [the section called “프라이빗 서브넷에서 인터넷 액세스”](#) 단원을 참조하십시오.

VPC 생성

필요한 VPC와 서브넷 수, VPC와 서브넷에 할당할 CIDR 블록, VPC를 인터넷에 연결하는 방법을 결정했으면 이제 VPC를 생성할 준비가 되었습니다. AWS Management Console를 사용하여 VPC를 생성하고 구성에 퍼블릭 서브넷을 포함하면 서브넷에 대한 라우팅 테이블이 생성되고 인터넷에 직접 액세스하는 데 필요한 경로가 추가됩니다. 자세한 내용은 [the section called “VPC 생성”](#) 단원을 참조하십시오.

애플리케이션 배포

VPC를 생성한 후에는 애플리케이션을 배포할 수 있습니다.

프로덕션 환경

프로덕션 환경의 경우 다음 서비스 중 하나를 사용하여 여러 가용 영역에 서버를 배포하고, 애플리케이션에 필요한 최소 서버 수를 유지하도록 조정을 구성하고, 로드 밸런서에 서버를 등록하여 트래픽을 서버 간에 균등하게 분산할 수 있습니다.

- [Amazon EC2 Auto Scaling](#)
- [EC2 플릿](#)
- [Amazon Elastic Container Service\(Amazon ECS\)](#)

개발 또는 테스트 환경

개발 또는 테스트 환경의 경우 단일 EC2 인스턴스를 시작하도록 선택할 수 있습니다. 자세한 내용은 Amazon EC2 사용 설명서의 [Amazon EC2 시작하기](#)를 참조하세요.

VPC 및 서브넷의 IP 주소 지정

IP 주소가 있으면 VPC 내의 리소스가 서로 통신하고, 인터넷을 통해 다른 리소스와 통신할 수 있습니다.

Classless Inter-Domain Routing(CIDR) 표기법은 IP 주소 및 네트워크 마스크를 나타내는 방법입니다. 이러한 주소의 형식은 다음과 같습니다.

- 개별 IPv4 주소는 32비트로, 최대 3개의 십진수로 구성된 그룹 4개를 포함합니다. 예를 들어 10.0.1.0입니다.
- IPv4 CIDR 블록에는 마침표로 구분된 0~255의 십진수가 최대 3개인 4개 그룹이 있으며, 그 뒤에 슬래시와 0~32의 숫자가 표시됩니다. 예: 10.0.0.0/16.
- 개별 IPv6 주소는 128비트로, 4개의 16진수로 구성된 그룹 8개를 포함합니다. 예: 2001:0db8:85a3:0000:0000:8a2e:0370:7334.
- IPv6 CIDR 블록에는 콜론으로 구분된 최대 4개의 16진수로 구성된 4개 그룹이 있으며, 그 뒤에 이중 콜론, 슬래시, 1~128의 숫자가 차례로 따라옵니다. 예: 2001:db8:1234:1a00::/56.

자세한 내용은 [CIDR이란 무엇입니까?](#)를 참조하세요.

내용

- [IPv4 및 IPv6 비교](#)
- [프라이빗 IPv4 주소](#)
- [퍼블릭 IPv4 주소](#)
- [IPv6 주소](#)
- [고유 IP 주소 사용](#)
- [Amazon VPC IP 주소 관리자 사용](#)
- [VPC CIDR 블록](#)
- [서브넷 CIDR 블록](#)
- [관리형 접두사 목록을 사용하여 CIDR 블록 그룹화](#)
- [AWS IP 주소 범위](#)
- [VPC에 IPv6 지원 추가](#)
- [IPv6를 지원하는 AWS 서비스](#)

IPv4 및 IPv6 비교

다음 표에는 Amazon EC2와 Amazon VPC의 IPv4 및 IPv6 간 차이점이 요약되어 있습니다. 듀얼 스택 구성(IPv4 및 IPv6)과 IPv6 전용 구성을 지원하는 AWS 서비스 목록은 [IPv6를 지원하는 서비스](#) 섹션을 참조하세요

기능	IPv4	IPv6
VPC 크기	/16부터 /28까지 최대 5개의 CIDR. 이 할당량 은 조정할 수 있습니다.	/44부터 /60까지(/4씩 증가) 최대 5개 CIDR. 이 할당량 은 조정할 수 있습니다.
서브넷 크기	/16~/28	/44에서 /64까지(/4씩 증가).
주소 선택	VPC용 IPv4 CIDR 블록을 선택하거나 Amazon VPC IP 주소 관리자(IPAM)에서 CIDR 블록을 할당할 수 있습니다. 자세한 내용은 Amazon VPC IPAM 사용 설명서의 IPAM이란 무엇인가? 를 참조하세요.	자체 IPv6 CIDR 블록을 VPC용으로 AWS로 가져오거나, Amazon 제공 IPv6 CIDR 블록을 선택하거나, Amazon VPC IP 주소 관리자(IPAM)에서 CIDR 블록을 할당할 수 있습니다. 자세한 내용은 Amazon VPC IPAM 사용 설명서의 IPAM이란 무엇인가? 를 참조하세요.
인터넷 액세스	인터넷 게이트웨이 가 필요합니다.	인터넷 게이트웨이가 필요합니다. 송신 전용 인터넷 게이트웨이 를 사용하여 아웃바운드 전용 통신을 지원합니다.
탄력적 IP 주소	지원 EC2 인스턴스에 영구 정적 퍼블릭 IPv4 주소를 제공합니다.	지원하지 않음. EIP는 인스턴스 재시작 시 인스턴스의 퍼블릭 IPv4 주소를 정적으로 유지합니다. IPv6 주소는 기본적으로 정적입니다.
NAT 게이트웨이	지원 프라이빗 서브넷의 인스턴스는 퍼블릭 NAT 게이트웨이를 사용하여 인터넷에 연결하거나 프라이빗 NAT 게이트웨이를 사용하여 다른 VPC의 리소스에 연결할 수 있습니다.	지원 NAT 게이트웨이를 NAT64와 함께 사용하면 IPv6 전용 서브넷의 인스턴스가 VPC 내, VPC 간, 온프레미스 네트워크 또는 인터넷을 통해 IPv4 전용 리소스와 통신할 수 있습니다.

기능	IPv4	IPv6
DNS 이름	인스턴스는 Amazon에서 제공한 IPBN 또는 RBN 기반 DNS 이름을 받습니다. DNS 이름은 인스턴스에 대해 선택한 DNS 레코드로 확인됩니다.	인스턴스는 Amazon에서 제공한 IPBN 또는 RBN 기반 DNS 이름을 받습니다. DNS 이름은 인스턴스에 대해 선택한 DNS 레코드로 확인됩니다.

프라이빗 IPv4 주소

프라이빗 IPv4 주소(이 단원에서는 프라이빗 IP 주소로도 표시)는 인터넷을 통해 액세스할 수 없고, VPC의 인스턴스 간 통신에 사용할 수 있습니다. VPC에서 인스턴스를 시작할 경우, 서브넷의 IPv4 주소 범위에 속한 주 프라이빗 IP 주소는 인스턴스의 주 네트워크 인터페이스(eth0)에 할당됩니다. 또한 각 인스턴스에는 인스턴스의 프라이빗 IP 주소를 확인하는 프라이빗(내부) DNS 호스트 이름이 할당됩니다. 호스트 이름은 리소스 기반 또는 IP 기반의 두 가지 유형이 될 수 있습니다. 자세한 내용은 [EC2 인스턴스 이름 지정](#)을 참조하세요. 주 프라이빗 IP 주소를 지정하지 않으면 서브넷 범위에서 사용할 수 있는 IP 주소가 선택됩니다. 네트워크 인터페이스에 대한 자세한 내용은 Amazon EC2 사용 설명서의 [탄력적 네트워크 인터페이스](#)를 참조하세요.

보조 프라이빗 IP 주소인 추가 프라이빗 IP 주소를 VPC에서 실행 중인 인스턴스에 할당할 수 있습니다. 주 프라이빗 IP 주소와 달리, 보조 프라이빗 IP 주소는 한 네트워크 인스턴스에서 다른 네트워크 인스턴스로 재할당할 수 있습니다. 인스턴스가 중지 및 재시작될 때 프라이빗 IP 주소는 네트워크 인터페이스와 계속해서 연동되고 인스턴스가 종료되면 연동이 해제됩니다. 주 IP 주소와 보조 IP 주소에 대한 자세한 내용은 Amazon EC2 사용 설명서의 [여러 IP 주소](#)를 참조하세요.

프라이빗 IP 주소란 VPC의 IPv4 CIDR 범위 내 IP 주소를 말합니다. 대부분의 VPC IP 주소 범위는 RFC 1918에서 지정된 프라이빗(비공개적으로 라우팅 가능) IP 주소 범위 내에 들어가지만 VPC에 대해 공개적으로 라우팅이 가능한 CIDR 블록을 사용할 수 있습니다. VPC의 IP 주소 범위에 상관없이 공개적으로 라우팅 가능한 CIDR 블록을 포함해 VPC의 CIDR 블록에서 인터넷으로 직접 액세스하는 것은 지원하지 않습니다. 인터넷 게이트웨이, 가상 프라이빗 게이트웨이, AWS Site-to-Site VPN 연결 또는 AWS Direct Connect와 같은 게이트웨이를 통한 인터넷 액세스를 설정해야 합니다.

AWS는 서브넷의 IPv4 주소 범위를 절대 인터넷이 알리지 않습니다.

퍼블릭 IPv4 주소

모든 서브넷은 해당 서브넷에서 생성된 네트워크 인터페이스가 퍼블릭 IPv4 주소(이 단원에서는 퍼블릭 IP 주소로도 표시함)를 받을 것인지 여부를 결정하는 속성을 갖습니다. 따라서 이 속성이 활성화된

서브넷에서 인스턴스를 시작할 경우, 퍼블릭 IP 주소는 인스턴스에 대해 생성된 주 네트워크 인터페이스(eth0)에 할당됩니다. 퍼블릭 IP 주소는 NAT(Network Address Translation)를 통해 주 프라이빗 IP 주소로 매핑됩니다.

Note

AWS에서는 탄력적 IP 주소 및 실행 중인 인스턴스에 연결된 퍼블릭 IPv4 주소를 포함하여 모든 퍼블릭 IPv4 주소에 요금을 부과합니다. 자세한 내용은 [Amazon VPC 요금 페이지](#)의 퍼블릭 IPv4 주소 탭을 참조하세요.

다음을 수행하여 인스턴스가 퍼블릭 IP 주소를 수신할지 여부를 제어할 수 있습니다.

- 서브넷의 퍼블릭 IP 주소 지정 속성 수정. 자세한 내용은 [서브넷의 퍼블릭 IPv4 주소 지정 속성 수정](#) 단원을 참조하십시오.
- 인스턴스를 시작하는 동안 퍼블릭 IP 주소 지정 기능을 활성화하거나 비활성화하면 서브넷의 퍼블릭 IP 주소 지정 속성을 재정의합니다.
- 네트워크 인터페이스와 연결된 IP 주소를 관리하여 시작 후 인스턴스에서 퍼블릭 IP 주소 할당을 취소할 수 있습니다. 자세한 내용은 Amazon EC2 사용 설명서의 [IP 주소 관리](#)를 참조하세요.

퍼블릭 IP 주소는 Amazon의 퍼블릭 IP 주소 풀로부터 할당되며 계정과는 관련이 없습니다. 인스턴스에서 퍼블릭 IP 주소의 연결이 해제되면 이 주소는 풀로 돌아가지만 더 이상 사용할 수 없습니다. 다음과 같은 특정 경우에 인스턴스에서 퍼블릭 IP 주소를 해제하거나 새 인스턴스에 할당합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [퍼블릭 IP 주소](#)를 참조하세요.

필요에 따라 인스턴스에 할당하거나 인스턴스에서 제거가 가능한, 계정에 할당된 영구 퍼블릭 IP 주소가 필요한 경우, 그 대신에 탄력적 IP 주소를 사용하십시오. 자세한 내용은 [탄력적 IP 주소를 VPC의 리소스와 연결](#) 단원을 참조하십시오.

VPC에서 DNS 호스트 이름을 지원하는 경우, 퍼블릭 IP 주소 또는 탄력적 IP 주소를 받는 각 인스턴스에도 퍼블릭 DNS 호스트 이름이 할당됩니다. Amazon은 퍼블릭 DNS 호스트 이름을 인스턴스 네트워크 외부에서는 인스턴스의 퍼블릭 IP 주소로 변환하고 인스턴스 네트워크 내부에서는 인스턴스의 프라이빗 IP 주소로 변환합니다. 자세한 내용은 [VPC의 DNS 속성](#) 단원을 참조하십시오.

IPv6 주소

IPv6 CIDR 블록과 VPC를 연결하고 IPv6 CIDR 블록과 서브넷을 연결할 수도 있습니다. 자세한 정보는 다음 주제를 참조하세요.

- [VPC에 IPv6 CIDR 블록 추가](#)
- [서브넷에 IPv6 CIDR 블록 추가](#)

IPv6 주소는 전역적으로 고유하며 프라이빗으로 유지되거나 인터넷으로 접속하도록 구성할 수 있습니다. IPv6 CIDR 블록이 VPC와 서브넷에 연결되어 있고 다음 중 하나가 true이면 인스턴스는 IPv6 주소를 받습니다.

- 서브넷은 시작 중인 인스턴스에 IPv6 주소를 자동으로 할당하도록 구성됩니다. 자세한 내용은 [서브넷의 IPv6 주소 지정 속성 수정](#) 단원을 참조하십시오.
- 시작하는 동안 인스턴스에 IPv6 주소를 할당합니다.
- 시작 후 인스턴스의 기본 네트워크 인터페이스에 IPv6 주소를 할당합니다.
- 동일 서브넷에서 네트워크 인터페이스에 IPv6 주소를 할당하고 시작을 완료한 후에 인스턴스에 네트워크 인터페이스를 연결합니다.

시작하는 과정에서 인스턴스가 IPv6 주소를 받는 경우, 해당 주소는 인스턴스의 주 네트워크 인터페이스(eth0)와 연결됩니다. 다음 방법으로 인스턴스 기본 네트워크 인터페이스(eth0)에 대한 IPv6 주소를 관리할 수 있습니다.

- 네트워크 인터페이스에서 IPv6 주소를 할당 및 할당 해제합니다. 네트워크 인터페이스에 할당할 수 있는 IPv6 주소의 개수, 그리고 인스턴스에 연결할 수 있는 네트워크 인터페이스의 개수는 인스턴스 유형에 따라 달라집니다. 자세한 내용은 Amazon EC2 사용 설명서의 [IP addresses per network interface per instance type](#)를 참조하세요.
- 기본 IPv6 주소를 활성화합니다. 기본 IPv6 주소를 사용하면 인스턴스나 ENI에 대한 트래픽 중단을 방지할 수 있습니다. 자세한 내용은 Amazon EC2 사용 설명서의 [네트워크 인터페이스 생성 및 IP 주소 관리](#)를 참조하세요.

인스턴스를 중지했다가 시작할 때 또는 최대 절전 모드로 전환했다가 시작할 때에는 IPv6 주소가 지속되다가 인스턴스를 종료하면 릴리스됩니다. IPv6 주소는 다른 네트워크 인터페이스에 할당되는 동안에는 재할당할 수 없으므로 먼저 할당을 해제해야 합니다.

서브넷에 대한 라우팅을 제어하거나 보안 그룹 및 네트워크 ACL 규칙을 사용함으로써 인스턴스가 IPv6 주소를 통해 접속이 가능하도록 할지 여부를 제어할 수 있습니다. 자세한 내용은 [Amazon VPC의 인터넷네트워크 트래픽 프라이버시](#) 단원을 참조하십시오.

예약된 IPv6 주소 범위에 대한 자세한 정보는 [IANA IPv6 Special-Purpose Address Registry](#) 및 [RFC4291](#)을 참조하십시오.

고유 IP 주소 사용

AWS 계정으로 고유 퍼블릭 IPv4 주소 범위 또는 IPv6 주소 범위의 일부 또는 전체를 가져올 수 있습니다. 주소 범위를 계속해서 소유할 수 있지만 AWS에서는 기본적으로 인터넷에 이러한 주소 범위를 알립니다. 주소 범위를 AWS(으)로 가져오고 나면 이러한 주소가 계정에 주소 풀로 나타납니다. IPv4 주소 풀에서 탄력적 IP 주소를 생성할 수 있으며, IPv6 주소 풀의 IPv6 CIDR 블록을 VPC와 연결할 수 있습니다.

자세한 내용을 알아보려면 Amazon EC2 사용 설명서의 [고유 IP 주소 가져오기\(BYOIP\)](#)를 참조하세요.

Amazon VPC IP 주소 관리자 사용

Amazon VPC IP 주소 관리자(IPAM)는 AWS 워크로드의 IP 주소를 보다 쉽게 계획, 추적 및 모니터링할 수 있게 해주는 VPC 기능입니다. IPAM을 통해 특정 비즈니스 규칙을 사용하여 VPC에 IP 주소 CIDR을 할당할 수 있습니다.

자세한 내용은 Amazon VPC IPAM 사용 설명서의 [IPAM이란 무엇인가?](#)를 참조하세요.

VPC CIDR 블록

Virtual Private Cloud(VPC)의 IP 주소는 Classless Inter-Domain Routing(CIDR) 표기법을 사용하여 표시됩니다. VPC에는 연결된 IPv4 CIDR 블록이 있어야 합니다. 선택에 따라 추가 IPv4 CIDR 블록과 하나 이상의 IPv6 CIDR 블록을 연결할 수 있습니다. 자세한 내용은 [VPC 및 서브넷의 IP 주소 지정](#) 단원을 참조하십시오.

목차

- [IPv4 VPC CIDR 블록](#)
- [VPC에 IPv4 CIDR 블록 관리](#)
- [IPv4 CIDR 블록 연결 제한](#)
- [IPv6 VPC CIDR 블록](#)

IPv4 VPC CIDR 블록

VPC를 만들 때 VPC의 IPv4 CIDR 블록을 지정해야 합니다. 허용된 블록 크기는 /16 넷마스크 (IP 주소 65,536개)~ /28 넷마스크 (IP 주소 16개)입니다. VPC 생성을 마쳤으면 추가 IPv4 CIDR 블록을 VPC에 연결할 수 있습니다. 자세한 내용은 [VPC에 IPv4 CIDR 블록 추가](#) 단원을 참조하십시오.

VPC를 생성하는 경우, 다음과 같이 [RFC 1918](#) 규격에 따라 프라이빗 IPv4 주소 범위에 속하는 CIDR 블록을 지정하는 것이 좋습니다.

RFC 1918 범위	CIDR 블록의 예
10.0.0.0 - 10.255.255.255 (10/8 접두사)	10.0.0.0/16
172.16.0.0 - 172.31.255.255 (172.16/12 접두사)	172.31.0.0/16
192.168.0.0 - 192.168.255.255 (192.168/16 접두사)	192.168.0.0/20

Important

일부 AWS 서비스는 172.17.0.0/16 CIDR 범위를 사용합니다. 향후 충돌을 방지하려면 VPC를 생성할 때 이 범위를 사용하지 마세요. 예를 들어 네트워크 어디에서 172.17.0.0/16 IP 주소 범위를 이미 사용 중인 경우 AWS Cloud9 또는 Amazon SageMaker와 같은 서비스에서 IP 주소 충돌이 발생할 수 있습니다. 자세한 내용은 AWS Cloud9 사용 설명서의 [Docker에서 VPC의 IP 주소를 사용하므로 EC2 환경에 연결할 수 없음](#)을 참조하세요.

RFC 1918에 지정된 프라이빗 IPv4 주소 범위에 속하지 않는 공개적으로 라우팅 가능한 CIDR 블록을 사용하여 VPC를 생성할 수 있습니다. 하지만 이 설명서에서는 프라이빗 IP 주소는 VPC의 CIDR 범위 내에 있는 프라이빗 IPv4 주소를 말합니다.

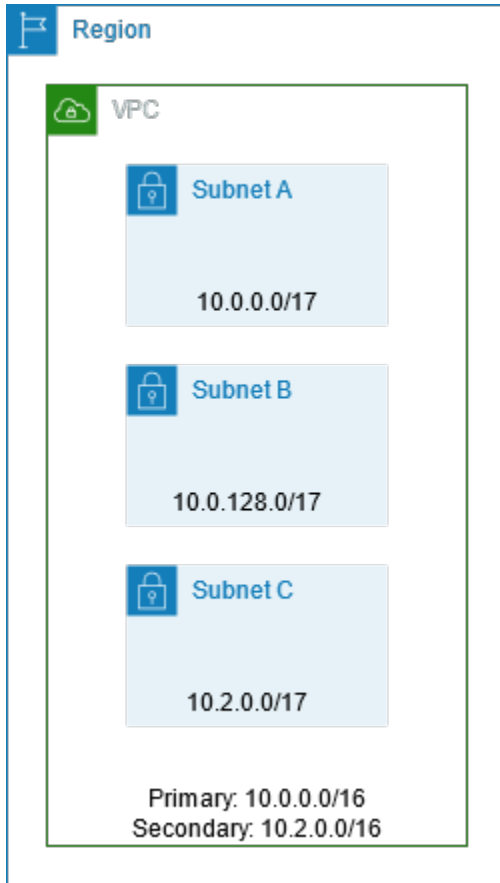
AWS 서비스와 함께 사용할 VPC를 생성하려면 해당 서비스 설명서를 참조하여 해당 구성에 대한 특정 요구 사항이 있는지 확인하십시오.

명령줄 도구 또는 Amazon EC2 API를 사용하여 VPC를 생성하면 CIDR 블록이 표준 형식으로 자동 수정됩니다. 예를 들어 CIDR 블록에 100.68.0.18/18을 지정하면 100.68.0.0/18의 CIDR 블록이 생성됩니다.

VPC에 IPv4 CIDR 블록 관리

보조 IPv4 CIDR 블록을 VPC와 연결할 수 있습니다. CIDR 블록을 VPC에 연결하면 VPC 라우팅 테이블에 경로가 자동으로 추가되면서 VPC 내에서 라우팅이 가능하게 됩니다(대상 주소는 CIDR 블록이고 대상은 local).

다음 예에서 VPC에는 기본 및 보조 CIDR 블록이 모두 있습니다. 서브넷 A와 서브넷 B의 CIDR 블록은 기본 VPC CIDR 블록에서 가져옵니다. 서브넷 C의 CIDR 블록은 보조 VPC CIDR 블록에서 가져옵니다.



다음 라우팅 테이블은 VPC의 로컬 경로를 보여줍니다.

대상 주소	대상
10.0.0.0/16	로컬
10.2.0.0/16	로컬

VPC에 CIDR 블록을 추가할 경우 다음 규칙이 적용됩니다.

- 허용된 블록 크기는 /28 넷마스크~/16 넷마스크입니다.
- CIDR 블록은 VPC에 연결된 기존 CIDR 블록과 겹치지 않습니다.
- 사용 가능한 IPv4 주소 범위에 제한이 있습니다. 자세한 내용은 [IPv4 CIDR 블록 연결 제한](#) 단원을 참조하십시오.

- 기존 CIDR 블록의 크기를 늘리거나 줄일 수 없습니다.
- VPC에 연결할 수 있는 CIDR 블록의 수와 라우팅 테이블에 추가할 수 있는 경로의 수에는 할당량이 있습니다. 할당량을 초과하는 경우에는 CIDR 블록을 연결할 수 없습니다. 자세한 내용은 [Amazon VPC 할당량](#) 단원을 참조하십시오.
- CIDR 블록은 모든 VPC 라우팅 테이블에서 경로의 대상 CIDR 범위보다 작아야 합니다. 예를 들어, 기본 CIDR 블록이 10.2.0.0/16인 VPC에서 가상 프라이빗 게이트웨이에 대한 대상이 10.0.0.0/24인 라우팅 테이블에 기존 라우팅이 있습니다. 10.0.0.0/16 범위의 보조 CIDR 블록을 연결하려고 합니다. 기존 경로 때문에 10.0.0.0/24 이상의 CIDR 블록을 연결할 수 없습니다. 그러나 10.0.0.0/25 이하의 보조 CIDR 블록은 연결할 수 있습니다.
- VPC 피어링 연결에 포함된 VPC에 IPv4 CIDR 블록을 추가할 때 다음 규칙이 적용됩니다.
 - VPC 피어링 연결이 active인 경우, 피어 VPC의 CIDR 블록과 겹치지 않으면 VPC에 CIDR 블록을 추가할 수 있습니다.
 - VPC 피어링 연결이 pending-acceptance인 경우, 수락자 VPC의 CIDR 블록과 겹치는지 여부에 관계 없이 요청자 VPC의 소유자가 VPC에 CIDR 블록을 추가할 수 없습니다. 수락자 VPC의 소유자가 피어링 연결을 수락하거나, 요청자 VPC의 소유자가 VPC 피어링 연결 요청을 삭제하고 CIDR 블록을 추가한 다음 VPC 피어링 연결을 새로 요청해야 합니다.
 - VPC 피어링 연결이 pending-acceptance인 경우, 수락자 VPC의 소유자는 VPC에 CIDR 블록을 추가할 수 있습니다. 보조 CIDR 블록이 요청자 VPC의 CIDR 블록과 겹치는 경우에는 VPC 피어링 연결 요청이 실패하고 요청을 수락할 수 없게 됩니다.
- AWS Direct Connect를 사용하여 Direct Connect 게이트웨이를 통해 여러 VPC에 연결하는 경우 Direct Connect 게이트웨이에 연결된 VPC에는 중첩되는 CIDR 블록이 있으면 안 됩니다. Direct Connect 게이트웨이와 연결된 VPC 중 하나에 CIDR 블록을 추가한 경우 새로운 CIDR 블록이 다른 연결된 VPC에 있는 기존 CIDR 블록과 중첩되어서는 안 됩니다. 자세한 내용은 AWS Direct Connect 사용 설명서의 [Direct Connect 게이트웨이](#)를 참조하세요.
- CIDR 블록을 추가하거나 제거하는 경우, associating | associated | disassociating | disassociated | failing | failed와 같은 다양한 상태를 통과할 수 있습니다. 사용자가 CIDR 블록을 사용할 수 있는 상태가 되면 associated 상태가 됩니다.

VPC에 연결한 CIDR 블록은 연결 해제가 가능하지만, 원래 VPC(기본 CIDR 블록)를 생성한 CIDR 블록은 연결을 해제할 수 없습니다. Amazon VPC 콘솔에서 VPC의 기본 CIDR을 보려면 Your VPCs(사용자 VPC)를 선택하고 자신의 VPC의 확인란을 선택한 다음 CIDRs 탭을 선택하십시오. AWS CLI를 사용하여 기본 CIDR을 보려면 다음과 같이 [describe-vpcs](#) 명령을 사용하십시오. 기본 CIDR은 최상위 CidrBlock element로 반환됩니다.


```
aws ec2 describe-vpcs --vpc-id vpc-1a2b3c4d --query Vpcs[*].CidrBlock --output text
```

출력의 예제는 다음과 같습니다.

```
10.0.0.0/16
```

IPv4 CIDR 블록 연결 제한

다음 표에서는 허용 및 제한된 VPC CIDR 블록 연결에 대한 개요를 제공합니다. 제한 이유는 일부 AWS 서비스가 AWS 서비스 측에서 충돌하지 않는 CIDR 블록이 필요한 교차 VPC 및 교차 계정 기능을 사용하기 때문입니다.

IP 주소 범위	제한된 연결	허용된 연결
10.0.0.0/8	<p>다른 RFC 1918* 범위(172.16.0.0/12 및 192.168.0.0/16)의 CIDR 블록입니다.</p> <p>VPC에 연결된 CIDR 블록 중 하나가 10.0.0.0/15 범위(10.0.0.0~10.1.255.255)에 해당되면 10.0.0.0/16 범위(10.0.0.0~10.0.255.255)의 CIDR 블록을 추가할 수 없습니다.</p> <p>198.19.0.0/16 범위의 CIDR 블록입니다.</p>	<p>제한되지 않는 10.0.0.0/8 범위의 기타 모든 /16 넷마스크~/28 넷마스크 CIDR 블록입니다.</p> <p>100.64.0.0/10 범위의 공개적으로 라우팅할 수 있는 모든 /16 넷마스크~/28 넷마스크 IPv4 CIDR 블록(비 RFC 1918) 또는 /16 넷마스크~/28 넷마스크 CIDR 블록입니다.</p>
169.254.0.0/16	'link local' 블록의 CIDR 블록은 RFC 5735 에 설명된 대로 예약되어 있으며 VPC에 할당할 수 없습니다.	
172.16.0.0/12	<p>다른 RFC 1918* 범위(10.0.0.0/8 및 192.168.0.0/16)의 CIDR 블록입니다.</p> <p>172.31.0.0/16 범위의 CIDR 블록입니다.</p>	<p>제한되지 않는 172.16.0.0/12 범위의 기타 모든 /16 넷마스크~/28 넷마스크 CIDR 블록입니다.</p> <p>100.64.0.0/10 범위의 공개적으로 라우팅할 수 있는 모든 /16 넷마스크~/28 넷마스크 IPv4 CIDR 블록(비</p>

IP 주소 범위	제한된 연결	허용된 연결
	198.19.0.0/16 범위의 CIDR 블록입니다.	RFC 1918) 또는 /16 넷마스크~/28 넷마스크 CIDR 블록입니다.
192.168.0.0/16	다른 RFC 1918* 범위(10.0.0.0/8 및 172.16.0.0/12)의 CIDR 블록입니다. 198.19.0.0/16 범위의 CIDR 블록입니다.	192.168.0.0/16 범위의 기타 모든 /16 넷마스크~/28 넷마스크 CIDR 블록입니다. 공개적으로 라우팅할 수 있는 모든 /16 넷마스크~/28 넷마스크 IPv4 CIDR 블록(비 RFC 1918) 또는 100.64.0.0/10 범위의 /16 넷마스크~/28 넷마스크 CIDR 블록입니다.
198.19.0.0/16	RFC 1918* 범위의 CIDR 블록입니다.	공개적으로 라우팅할 수 있는 모든 /16 넷마스크~/28 넷마스크 IPv4 CIDR 블록(비 RFC 1918) 또는 100.64.0.0/10 범위의 /16 넷마스크~/28 넷마스크 CIDR 블록입니다.
공개적으로 라우팅이 가능한 CIDR 블록(비-RFC 1918) 또는 100.64.0.0/10 범위의 CIDR 블록	RFC 1918* 범위의 CIDR 블록입니다. 198.19.0.0/16 범위의 CIDR 블록입니다.	공개적으로 라우팅할 수 있는 기타 모든 /16 넷마스크~/28 넷마스크 IPv4 CIDR 블록(비 RFC 1918) 또는 100.64.0.0/10 범위의 /16 넷마스크~/28 넷마스크 CIDR 블록입니다.

* RFC 1918 범위는 [RFC 1918](#)에 지정된 프라이빗 IPv4 주소 범위입니다.

IPv6 VPC CIDR 블록

새 VPC를 생성할 때 단일 IPv6 CIDR 블록을 연결하거나 /44에서 /60까지 /4씩 증가하면서 최대 5개의 IPv6 CIDR 블록을 연결할 수 있습니다. Amazon의 IPv6 주소 풀에서 IPv6 CIDR 블록을 요청할 수 있습니다. 자세한 내용은 [VPC에 IPv6 CIDR 블록 추가](#) 단원을 참조하십시오.

IPv6 CIDR 블록을 VPC와 연결한 경우, IPv6 CIDR 블록을 VPC의 기존 서브넷 또는 새로 생성한 서브넷과 연결할 수 있습니다. 자세한 내용은 [the section called “IPv6에 대한 서브넷 크기 조정”](#) 단원을 참조하십시오.

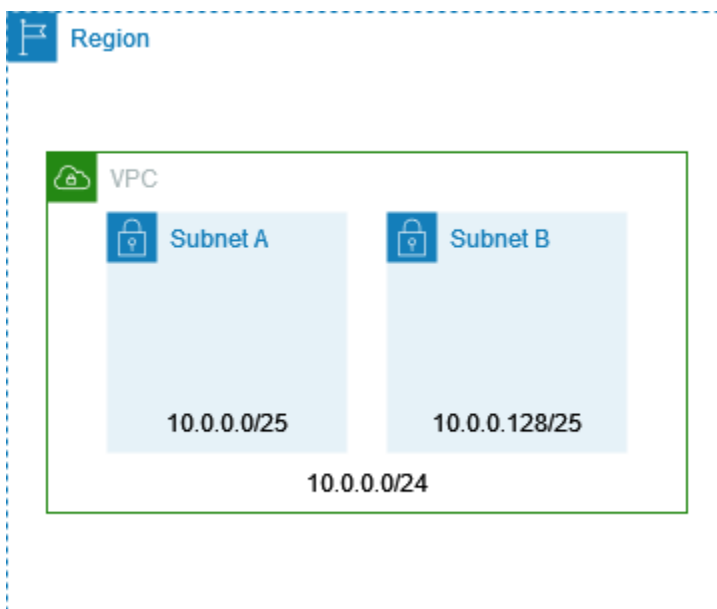
예를 들어, VPC를 생성하고 이 VPC에 Amazon이 제공하는 IPv6 CIDR 블록을 연결하도록 지정합니다. Amazon은 VPC에 IPv6 CIDR 블록 `2001:db8:1234:1a00::/56`을 할당합니다. IP 주소 범위는 직접 선택할 수 없습니다. 서브넷을 생성하고 이 범위에 속하는 IPv6 CIDR 블록을 연결할 수 있습니다 (예: `2001:db8:1234:1a00::/64`).

VPC에서 IPv6 CIDR 블록을 연결 해제할 수 있습니다. VPC에 연결된 IPv6 CIDR 블록을 분리한 후 나중에 다시 VPC에 IPv6 CIDR 블록을 연결하는 경우, 동일한 CIDR을 받을 것으로 기대할 수는 없습니다.

서브넷 CIDR 블록

서브넷의 IP 주소는 Classless Inter-Domain Routing(CIDR) 표기법을 사용하여 표시됩니다. 서브넷의 CIDR 블록은 VPC에 대한 CIDR 블록(VPC의 단일 서브넷 생성용) 또는 VPC에 대한 CIDR 블록의 하위 세트(VPC에 여러 서브넷 생성용)와 동일할 수 있습니다. VPC에 두 개 이상의 서브넷을 만들 경우, 서브넷의 CIDR 블록이 겹치지 않아야 합니다.

예를 들어 CIDR 블록이 `10.0.0.0/24`인 VPC를 만들 경우 256개의 IP 주소를 지원합니다. 이 CIDR 블록을 각각 128개의 IP 주소를 지원하는 2개의 서브넷으로 나눌 수 있습니다. 한 서브넷은 `10.0.0.0/25` CIDR 블록(`10.0.0.0~10.0.0.127` 사이의 주소)을, 다른 서브넷은 `10.0.0.128/25` CIDR 블록(`10.0.0.128~10.0.0.255` 사이의 주소)을 사용합니다.



인터넷에서 IPv4 및 IPv6 서브넷 CIDR 블록을 계산하고 생성하는 데 도움이 되는 도구가 있습니다. '서브넷 계산기' 또는 'CIDR 계산기'와 같은 용어를 검색하여 필요에 맞는 도구를 찾을 수 있습니다. 또한 네트워크 엔지니어링 그룹의 도움을 받아 서브넷에 지정할 IPv4 및 IPv6 CIDR 블록을 정할 수도 있습니다.

IPv4에 대한 서브넷 크기 조정

서브넷에 허용되는 IPv4 CIDR 블록 크기는 /28 넷마스크~/16 넷마스크입니다. 각 서브넷 CIDR 블록에서 첫 4개의 IP 주소와 마지막 IP 주소는 사용자가 사용할 수 없으므로 EC2 인스턴스 등의 리소스에 할당할 수 없습니다. 예를 들어 10.0.0.0/24 CIDR 블록의 서브넷에서는 다음 5개 IP 주소가 예약되어 있습니다.

- 10.0.0.0: 네트워크 주소.
- 10.0.0.1: AWS에서 VPC 라우터용으로 예약한 주소.
- 10.0.0.2: AWS에서 예약한 주소. DNS 서버의 IP 주소는 기본 VPC 네트워크 범위에 2를 더한 주소입니다. CIDR 블록이 여러 개인 VPC의 경우, DNS 서버의 IP 주소가 기본 CIDR에 위치합니다. 또한 각 서브넷 범위의 기본에 2를 더한 주소를 VPC의 모든 CIDR 블록에 대해 예약합니다. 자세한 내용은 [Amazon DNS 서버](#) 섹션을 참조하세요.
- 10.0.0.3: AWS에서 앞으로 사용하려고 예약한 주소.
- 10.0.0.255: 네트워크 브로드캐스트 주소. VPC에서는 브로드캐스트를 지원하지 않으므로, 이 주소를 예약합니다.

명령줄 도구 또는 Amazon EC2 API를 사용하여 서브넷을 추가하면 CIDR 블록이 표준 형식으로 자동 수정됩니다. 예를 들어 CIDR 블록에 100.68.0.18/18을 지정하면 100.68.0.0/18의 CIDR 블록이 생성됩니다.

[BYOIP](#)를 사용하여 AWS로 IPv4 주소 범위를 가져오면 첫 번째 주소(네트워크 주소)와 마지막 주소(브로드캐스트 주소)를 포함하여 범위 내의 IP 주소를 모두 사용할 수 있습니다.

IPv6에 대한 서브넷 크기 조정

IPv6 CIDR 블록을 VPC와 연결한 경우, IPv6 CIDR 블록을 VPC의 기존 서브넷 또는 새로 생성한 서브넷과 연결할 수 있습니다. 가능한 IPv6 넷마스크 길이는 /44~/64(/4씩 증가)입니다.

인터넷에서 IPv6 서브넷 CIDR 블록을 계산하고 생성하는 데 도움이 되는 도구가 있습니다. 'IPv6 서브넷 계산기' 또는 'IPv6 CIDR 계산기'와 같은 용어를 검색하여 필요에 맞는 도구를 찾을 수 있습니다. 또한 네트워크 엔지니어링 그룹의 도움을 받아 서브넷에 지정할 IPv6 CIDR 블록을 정할 수도 있습니다.

각 서브넷 CIDR 블록에서 첫 4개의 IPv6 주소와 마지막 IPv6 주소는 사용자가 사용할 수 없으므로 EC2 인스턴스에 할당할 수 없습니다. 예를 들어 2001:db8:1234:1a00/64 CIDR 블록의 서브넷에서는 다음 5개 IP 주소가 예약되어 있습니다.

- 2001:db8:1234:1a00::
- 2001:db8:1234:1a00::1: AWS에서 VPC 라우터용으로 예약한 주소.
- 2001:db8:1234:1a00::2
- 2001:db8:1234:1a00::3
- 2001:db8:1234:1a00:ffff:ffff:ffff:ffff

위 예제에서 VPC 라우터용으로 AWS에서 예약한 IP 주소 외에도 다음 IPv6 주소가 기본 VPC 라우터용으로 예약되어 있습니다.

- EUI-64를 사용하여 생성된 FE80::/10 범위의 링크-로컬 IPv6 주소입니다. 링크-로컬 주소에 대한 자세한 내용은 [링크-로컬 주소](#)를 참조하세요.
- 링크-로컬 IPv6 주소 FE80:ec2::1입니다.

IPv6를 통해 VPC 라우터와 통신해야 하는 경우 필요에 가장 적합한 주소와 통신하도록 애플리케이션을 구성할 수 있습니다.

관리형 접두사 목록을 사용하여 CIDR 블록 그룹화

관리형 접두사 목록은 하나 이상의 CIDR 블록 세트입니다. 접두사 목록을 사용하면 보안 그룹과 라우팅 테이블을 보다 쉽게 구성하고 유지 관리할 수 있습니다. 자주 사용하는 IP 주소에서 접두사 목록을 만들고, 이를 개별적으로 참조하지 않고 보안 그룹 규칙 및 경로의 집합으로 참조할 수 있습니다. 예를 들어, 서로 다른 CIDR 블록은 있지만 포트와 프로토콜은 동일한 보안 그룹 규칙을 접두사 목록을 사용하는 단일 규칙으로 통합할 수 있습니다. 네트워크를 확장하고 다른 CIDR 블록의 트래픽을 허용해야 하는 경우, 관련 접두사 목록을 업데이트할 수 있으며 그러면 접두사 목록을 사용하는 모든 보안 그룹이 업데이트됩니다. Resource Access Manager(RAM)를 사용하여 다른 AWS 계정과 함께 관리형 접두사 목록을 사용할 수도 있습니다.

접두사 목록에는 두 가지 유형이 있습니다.

- 고객 관리형 접두사 목록 — 사용자가 정의하고 관리하는 IP 주소 범위 세트입니다. 접두사 목록을 다른 AWS 계정과 공유하여 해당 계정이 자체 리소스의 접두사 목록을 참조하도록 할 수 있습니다.

- AWS 관리형 접두사 목록 - AWS 서비스의 IP 주소 범위 세트입니다. AWS 관리형 접두사 목록은 생성, 수정, 공유 또는 삭제할 수 없습니다.

목차

- [접두사 목록 개념 및 규칙](#)
- [접두사 목록에 대한 Identity and Access Management](#)
- [고객 관리형 접두사 목록 작업](#)
- [AWS 관리형 접두사 목록 작업](#)
- [공유 접두사 목록 작업](#)
- [AWS 리소스의 참조 접두사 목록](#)

접두사 목록 개념 및 규칙

접두사 목록은 항목으로 구성됩니다. 각 항목은 CIDR 블록과 CIDR 블록에 대한 설명(선택 사항)으로 구성됩니다.

고객 관리형 접두사 목록

고객 관리형 접두사 목록에는 다음 규칙이 적용됩니다.

- 접두사 목록은 단일 IP 주소 지정 유형(IPv4 또는 IPv6)만 지원합니다. IPv4 및 IPv6 CIDR 블록을 하나의 접두사 목록에 결합할 수 없습니다.
- 접두사 목록은 해당 목록을 생성한 리전에만 적용됩니다.
- 접두사 목록을 생성할 때 접두사 목록에서 지원할 수 있는 최대 항목 수를 지정해야 합니다.
- 리소스의 접두사 목록을 참조할 때 접두사 목록의 최대 항목 수는 리소스의 항목 수에 대한 할당량에 따라 계산됩니다. 예를 들어, 최대 항목이 20개인 접두사 목록을 만들고 보안 그룹 규칙에서 해당 접두사 목록을 참조하는 경우, 20개의 보안 그룹 규칙이 있는 것으로 계산됩니다.
- 라우팅 테이블의 접두사 목록을 참조할 때 라우팅 우선 순위 규칙이 적용됩니다. 자세한 내용은 [라우팅 우선 순위 및 접두사 목록](#) 단원을 참조하세요.
- 접두사 목록을 수정할 수 있습니다. 항목을 추가하거나 제거하면 접두사 목록의 새 버전이 생성됩니다. 접두사를 참조하는 리소스는 항상 현재(최신) 버전을 사용합니다. 이전 버전의 접두사 목록에 있는 항목을 복원할 수 있으며 이것도 새 버전을 생성합니다.
- 접두사 목록과 관련된 할당량이 있습니다. 자세한 내용은 [고객 관리형 접두사 목록](#) 단원을 참조하십시오.

- 고객 관리형 접두사 목록은 모든 상용 [AWS 리전](#)(GovCloud(미국) 및 중국 리전 포함)에서 사용할 수 있습니다.

AWS 관리형 접두사 목록

AWS 관리형 접두사 목록에는 다음 규칙이 적용됩니다.

- AWS 관리형 접두사 목록은 생성, 수정, 공유 또는 삭제할 수 없습니다.
- 서로 다른 AWS 관리형 접두사 목록은 사용할 때 다른 가중치를 갖습니다. 자세한 내용은 [AWS 관리형 접두사 목록 가중치](#) 단원을 참조하십시오.
- AWS 관리형 접두사 목록의 버전 번호는 볼 수 없습니다.

접두사 목록에 대한 Identity and Access Management

기본적으로 사용자는 접두사 목록을 생성, 보기, 수정 또는 삭제할 수 있는 권한이 없습니다. IAM 정책을 생성하여 이를 사용자가 접두사 목록 작업을 할 수 있는 역할에 연결할 수 있습니다.

IAM 정책에서 사용할 수 있는 Amazon VPC 작업 목록과 리소스 및 조건 키를 보려면 IAM 사용 설명서의 [Amazon EC2에서 사용하는 작업, 리소스 및 조건 키](#)를 참조하십시오.

다음 예제 정책은 사용자가 접두사 목록 p1-123456abcde123456만 보고 작업할 수 있도록 허용합니다. 사용자는 접두사 목록을 생성하거나 삭제할 수 없습니다.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "ec2:GetManagedPrefixListAssociations",
      "ec2:GetManagedPrefixListEntries",
      "ec2:ModifyManagedPrefixList",
      "ec2:RestoreManagedPrefixListVersion"
    ],
    "Resource": "arn:aws:ec2:region:account:prefix-list/p1-123456abcde123456"
  },
  {
    "Effect": "Allow",
    "Action": "ec2:DescribeManagedPrefixLists",
    "Resource": "*"
  }
}
```

```
]
}
```

Amazon VPC에서 IAM을 이용하는 자세한 방법은 [Amazon VPC용 Identity and Access Management](#)을 참조하십시오.

고객 관리형 접두사 목록 작업

고객 관리형 접두사 목록을 생성하고 관리할 수 있습니다. AWS 관리형 접두사 목록을 볼 수 있습니다.

Tasks

- [접두사 목록 생성](#)
- [접두사 목록 보기](#)
- [접두사 목록 항목 보기](#)
- [접두사 목록에 대한 연결\(참조\) 보기](#)
- [접두사 목록 수정](#)
- [접두사 목록 크기 조정](#)
- [이전 버전의 접두사 목록 복원](#)
- [접두사 목록 삭제](#)

접두사 목록 생성

접두사 목록을 생성할 때 접두사 목록에서 지원할 수 있는 최대 항목 수를 지정해야 합니다.

제한 사항

규칙 수에 접두사 목록의 최대 항목을 더한 값이 계정의 보안 그룹당 규칙의 할당량을 초과하는 경우, 보안 그룹 규칙에 접두사 목록을 추가할 수 없습니다.

콘솔을 사용하여 접두사 목록을 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 관리형 접두사 목록을 선택합니다.
3. 접두사 목록 생성을 선택합니다.
4. 접두사 목록 이름에 접두사 목록 이름을 입력합니다.

5. 최대 항목에 접두사 목록의 최대 항목 수를 입력합니다.
6. 주소 패밀리에 접두사 목록에서 IPv4 항목을 지원하는지 IPv6 항목을 지원하는지 여부를 선택합니다.
7. 접두사 목록 항목에서 새 항목 추가를 선택하고 CIDR 블록과 항목에 대한 설명을 입력합니다. 각 항목에 대해 이 단계를 반복합니다.
8. (선택 사항) 나중에 식별할 수 있도록 태그에서 접두사 목록에 대한 태그를 추가합니다.
9. 접두사 목록 생성을 선택합니다.

AWS CLI를 사용하여 접두사 목록을 생성하려면

[create-managed-prefix-list](#) 명령을 사용합니다.

접두사 목록 보기

접두사 목록, 공유된 접두사 목록 및 AWS 관리형 접두사 목록을 볼 수 있습니다.

콘솔을 사용하여 접두사 목록을 보려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 관리형 접두사 목록을 선택합니다.
3. 소유자 ID 옆에 접두사 목록 소유자의 AWS 계정 ID가 표시됩니다. AWS 관리형 접두사 목록의 경우 소유자 ID는 AWS입니다.

AWS CLI를 사용하여 접두사 목록을 보려면

[describe-managed-prefix-lists](#) 명령을 사용합니다.

접두사 목록 항목 보기

접두사 목록, 공유된 접두사 목록 및 AWS 관리형 접두사 목록의 항목을 볼 수 있습니다.

콘솔을 사용하여 접두사 목록의 항목을 보려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 관리형 접두사 목록을 선택합니다.
3. 접두사 목록의 확인란을 선택합니다.

4. 아래쪽 창에서 항목을 선택하여 접두사 목록의 항목을 봅니다.

AWS CLI를 사용하여 접두사 목록의 항목을 보려면

[get-managed-prefix-list-entries](#) 명령을 사용합니다.

접두사 목록에 대한 연결(참조) 보기

접두사 목록과 연결된 리소스의 ID 및 소유자를 볼 수 있습니다. 연결된 리소스는 해당 항목 또는 규칙에서 접두사 목록을 참조하는 리소스입니다.

제한 사항

AWS 관리형 접두사 목록에 대한 관련 리소스는 볼 수 없습니다.

콘솔을 사용하여 접두사 목록 연결을 보려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 관리형 접두사 목록을 선택합니다.
3. 접두사 목록의 확인란을 선택합니다.
4. 아래쪽 창에서 연결을 선택하여 접두사 목록을 참조하는 리소스를 봅니다.

AWS CLI를 사용하여 접두사 목록 연결을 보려면

[get-managed-prefix-list-associations](#) 명령을 사용합니다.

접두사 목록 수정

접두사 목록의 이름을 수정하고 항목을 추가 또는 제거할 수 있습니다. 최대 항목 수를 수정하려면 [접두사 목록 크기 조정](#) 섹션을 참조하세요.

접두사 목록의 항목을 업데이트하면 접두사 목록의 새 버전이 생성됩니다. 접두사 목록의 이름이나 최대 항목 수를 업데이트해도 접두사 목록의 새 버전이 생성되지 않습니다.

고려 사항

- AWS 관리형 접두사 목록은 수정할 수 없습니다.
- 접두사 목록에서 최대 항목 수를 늘리면 접두사 목록을 참조하는 리소스의 항목 할당량에 증가된 최대 크기가 적용됩니다. 이러한 리소스 중 증가된 최대 크기를 지원할 수 없는 경우 수정 작업이 실패하고 이전 최대 크기가 복원됩니다.

콘솔을 사용하여 접두사 목록을 수정하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 관리형 접두사 목록을 선택합니다.
3. 접두사 목록의 확인란을 선택하고 작업(Actions), 접두사 목록 수정(Modify prefix list)을 선택합니다.
4. 접두사 목록 이름에 접두사 목록의 새 이름을 입력합니다.
5. 접두사 목록 항목에서 제거를 선택하여 기존 항목을 제거합니다. 새 항목을 추가하려면 새 항목 추가를 선택하고 CIDR 블록과 항목에 대한 설명을 입력합니다.
6. 접두사 목록 저장을 선택합니다.

AWS CLI를 사용하여 접두사 목록을 수정하려면

[modify-managed-prefix-list](#) 명령을 사용합니다.

접두사 목록 크기 조정

접두사 목록의 크기를 조정하고 접두사 목록의 최대 항목 수를 최대 1000개까지 수정할 수 있습니다. 고객 관리형 접두사 키에 대한 자세한 내용은 [고객 관리형 접두사 목록](#) 섹션을 참조하세요.

콘솔을 사용하여 접두사 목록 크기 조정

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 관리형 접두사 목록(Managed Prefix Lists)을 선택합니다.
3. 접두사 목록의 확인란을 선택하고 작업(Actions), 접두사 목록 크기 조정(Resize prefix list)을 선택합니다.
4. 새 최대 항목(New max entries)에 값을 입력합니다.
5. 크기 조정(Resize)을 선택합니다.

AWS CLI를 사용하여 접두사 목록 크기 조정

[modify-managed-prefix-list](#) 명령을 사용합니다.

이전 버전의 접두사 목록 복원

이전 버전의 접두사 목록에서 항목을 복원할 수 있습니다. 그러면 새로운 버전의 접두사 목록이 생성됩니다.

접두사 목록의 크기를 줄인 경우 접두사 목록이 이전 버전의 항목을 포함할 수 있을 만큼 충분히 커야 합니다.

콘솔을 사용하여 이전 버전의 접두사 목록을 복원하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 관리형 접두사 목록을 선택합니다.
3. 접두사 목록의 확인란을 선택하고 작업(Actions), 접두사 목록 복원(Restore prefix list)을 선택합니다.
4. 접두사 목록 버전 선택(Select prefix list version)에서 이전 버전을 선택합니다. 선택한 버전에 대한 항목은 접두사 목록 항목(Prefix list entries)에 표시됩니다.
5. 접두사 목록 복원(Restore prefix list)을 선택합니다.

AWS CLI를 사용하여 이전 버전의 접두사 목록을 복원하려면

[restore-managed-prefix-list-version](#) 명령을 사용합니다.

접두사 목록 삭제

접두사 목록을 삭제하려면 먼저 라우팅 테이블과 같은 리소스에서 접두사 목록에 대한 참조를 제거해야 합니다. AWS RAM을 사용하여 접두사 목록을 공유한 경우 먼저 소비자가 소유한 리소스에서 참조를 제거해야 합니다.

제한 사항

AWS 관리형 접두사 목록은 삭제할 수 없습니다.

콘솔을 사용하여 접두사 목록을 삭제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 관리형 접두사 목록을 선택합니다.
3. 접두사 목록을 선택하고 작업, 접두사 목록 삭제를 선택합니다.
4. 확인 대화 상자에 delete를 입력한 다음 삭제를 선택합니다.

AWS CLI를 사용하여 접두사 목록을 삭제하려면

[delete-managed-prefix-list](#) 명령을 사용합니다.

AWS 관리형 접두사 목록 작업

AWS 관리형 접두사 목록은 AWS 서비스에 대한 IP 주소 범위 세트입니다.

내용

- [AWS 관리형 접두사 목록 사용](#)
- [AWS 관리형 접두사 목록 가중치](#)
- [사용 가능한 AWS 관리형 접두사 목록](#)

AWS 관리형 접두사 목록 사용

AWS 관리형 접두사 목록은 AWS에서 생성하고 유지관리할 수 있고 AWS 계정이 있는 모든 사용자가 사용할 수 있습니다. AWS 관리형 접두사 목록은 생성, 수정, 공유 또는 삭제할 수 없습니다.

고객 관리형 접두사 목록과 마찬가지로 AWS 관리형 접두사 목록을 보안 그룹 및 라우팅 테이블과 같은 AWS 리소스와 함께 사용할 수 있습니다. 자세한 내용은 [AWS 리소스의 참조 접두사 목록](#) 단원을 참조하십시오.

AWS 관리형 접두사 목록 가중치

AWS 관리형 접두사 목록의 가중치는 리소스에서 차지하는 항목 수를 나타냅니다.

예를 들어 Amazon CloudFront 관리형 접두사 목록의 가중치는 55입니다. Amazon VPC 할당량에 미치는 영향은 다음과 같습니다.

- 보안 그룹 – [기본 할당량](#)이 규칙 60개이므로 보안 그룹에 추가 규칙의 여지가 5개만 남습니다. 이 할당량에 대해 [할당량 증가 요청](#)을 할 수 있습니다.
- 라우팅 테이블 – [기본 할당량](#)이 경로 50개이므로 라우팅 테이블에 접두사 목록을 추가하기 전에 [할당량 증가를 요청](#)해야 합니다.

사용 가능한 AWS 관리형 접두사 목록

다음과 같은 서비스에서 AWS 관리형 접두사 목록이 제공됩니다.

AWS 서비스	접두사 목록 이름	가중치
Amazon CloudFront	com.amazonaws.global.cloudfront.origin-facing	55

AWS 서비스	접두사 목록 이름	가중치
Amazon DynamoDB	com.amazonaws. <i>region</i> .dynamodb	1
AWS Ground Station	com.amazonaws.global.groundstation	5
Amazon Route 53	com.amazonaws. <i>region</i> .ipv6.route53-healthchecks	25
	com.amazonaws. <i>region</i> .route53-healthchecks	25
Amazon S3	com.amazonaws. <i>region</i> .s3	1
Amazon S3 Express One Zone	com.amazonaws. <i>region</i> .s3express	6
Amazon VPC Lattice	com.amazonaws. <i>region</i> .vpc-lattice	10
	com.amazonaws. <i>region</i> .ipv6.vpc-lattice	10

콘솔을 사용하여 AWS 관리형 접두사 목록을 보는 방법

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 관리형 접두사 목록을 선택합니다.
3. 검색 필드에서 Owner ID: AWS 필터를 추가합니다.

AWS CLI를 사용하여 AWS 관리형 접두사 목록을 보는 방법

다음과 같이 [describe-managed-prefix-lists](#) 명령을 사용합니다.

```
aws ec2 describe-managed-prefix-lists --filters Name=owner-id,Values=AWS
```

공유 접두사 목록 작업

AWS Resource Access Manager(AWS RAM)을 사용하여 접두사 목록의 소유자는 접두사 목록을 다음과 공유할 수 있습니다.

- AWS 조직 내부 또는 외부의 특정 AWS Organizations 계정

- AWS Organizations에서 조직 내부의 조직 단위
- AWS Organizations의 전체 조직

접두사 목록을 공유하는 소비자는 접두사 목록과 해당 항목을 볼 수 있으며 AWS 리소스에서 접두사 목록을 참조할 수 있습니다.

AWS RAM에 대한 자세한 내용은 [AWS RAM 사용 설명서](#)를 참조하세요.

내용

- [접두사 목록 공유를 위한 사전 조건](#)
- [접두사 목록 공유](#)
- [공유 접두사 목록 식별](#)
- [공유 접두사 목록에 대한 참조 식별](#)
- [공유 접두사 목록 공유 해제](#)
- [공유 접두사 목록 사용 권한](#)
- [결제 및 측정](#)
- [AWS RAM에 대한 할당량](#)

접두사 목록 공유를 위한 사전 조건

- 접두사 목록을 공유하려면 계정에서 해당 접두사 목록을 소유해야 합니다. 다른 사용자가 자신과 공유한 접두사 목록은 공유할 수 없습니다. AWS 관리형 접두사 목록은 공유할 수 없습니다.
- AWS Organizations의 조직 또는 조직 단위와 접두사 목록을 공유하려면 AWS Organizations와의 공유를 활성화해야 합니다. 자세한 내용은 AWS RAM 사용 설명서에서 [AWS Organizations를 사용하여 공유 사용](#)을 참조하세요.

접두사 목록 공유

접두사 목록을 공유하려면 리소스 공유에 추가해야 합니다. 리소스 공유가 없는 경우 먼저 [AWS RAM 콘솔](#)을 사용하여 리소스 공유를 생성해야 합니다.

AWS Organizations의 조직에 속해 있고 조직 내의 공유가 활성화되어 있으면 조직의 소비자에게 공유 접두사 목록에 대한 액세스 권한이 자동으로 부여됩니다. 그렇지 않은 경우 리소스 공유에 가입하라는 초대를 받은 소비자가 초대를 수락하면 공유 접두사 목록에 대한 액세스 권한이 부여됩니다.

리소스 공유를 생성하고 AWS RAM 콘솔 또는 AWS CLI를 사용하여 소유한 접두사 목록을 공유할 수 있습니다.

AWS RAM 콘솔을 사용하여 리소스 공유를 생성하고 접두사 목록을 공유하려면

AWS RAM 사용 설명서의 [리소스 공유 생성](#) 단계를 따릅니다. 리소스 유형 선택에서 접두사 목록을 선택한 다음 접두사 목록에 대한 확인란을 선택합니다.

AWS RAM 콘솔을 사용하여 기존 리소스 공유에 접두사 목록을 추가하려면

자신이 소유한 관리형 접두사를 기존 리소스 공유에 추가하려면 AWS RAM 사용 설명서의 [리소스 공유 업데이트](#) 단계를 따릅니다. 리소스 유형 선택에서 접두사 목록을 선택한 다음 접두사 목록에 대한 확인란을 선택합니다.

AWS CLI를 사용하여 자신이 소유한 접두사 목록을 공유하려면

다음 명령을 사용하여 리소스 공유를 생성하고 업데이트합니다.

- [create-resource-share](#)
- [associate-resource-share](#)
- [update-resource-share](#)

공유 접두사 목록 식별

소유자와 소비자는 Amazon VPC 콘솔 및 AWS CLI를 사용하여 공유 접두사 목록을 식별할 수 있습니다.

Amazon VPC 콘솔을 사용하여 공유 접두사 목록을 식별하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 관리형 접두사 목록을 선택합니다.
3. 이 페이지에는 자신이 소유한 접두사 목록과 다른 사용자가 자신과 공유한 접두사 목록이 표시됩니다. 소유자 ID 옆에 접두사 목록 소유자의 AWS 계정 ID가 표시됩니다.
4. 접두사 목록에 대한 리소스 공유 정보를 보려면 접두사 목록을 선택하고 아래쪽 창에서 공유를 선택합니다.

AWS CLI를 사용하여 공유 접두사 목록을 식별하려면

[describe-managed-prefix-lists](#) 명령을 사용합니다. 이 명령은 자신이 소유한 접두사 목록과 다른 사용자가 자신과 공유한 접두사 목록을 반환합니다. OwnerId에는 접두사 목록 소유자의 AWS 계정 ID가 표시됩니다.

공유 접두사 목록에 대한 참조 식별

소유자는 공유 접두사 목록을 참조하는 소비자 소유 리소스를 식별할 수 있습니다.

Amazon VPC 콘솔을 사용하여 공유 접두사 목록에 대한 참조를 식별하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 관리형 접두사 목록을 선택합니다.
3. 접두사 목록을 선택하고 아래쪽 창에서 연결을 선택합니다.
4. 접두사 목록을 참조하는 리소스의 ID가 리소스 ID 옆에 나열됩니다. 리소스 소유자가 리소스 소유자 옆에 나열됩니다.

AWS CLI를 사용하여 공유 접두사 목록에 대한 참조를 식별하려면

[get-managed-prefix-list-associations](#) 명령을 사용합니다.

공유 접두사 목록 공유 해제

접두사 목록의 공유를 해제하면 소비자가 더 이상 해당 계정에서 접두사 목록 또는 해당 항목을 볼 수 없으며 리소스에서 접두사 목록을 참조할 수 없습니다. 접두사 목록이 이미 소비자의 리소스에서 참조된 경우 이러한 참조는 정상적으로 계속 작동하며 [이러한 참조를 계속 볼](#) 수 있습니다. 접두사 목록을 새 버전으로 업데이트하는 경우 참조는 최신 버전을 사용합니다.

자신이 소유한 공유 접두사 목록을 공유 해제하려면 AWS RAM을 사용해 리소스 공유에서 제거해야 합니다.

AWS RAM 콘솔을 사용하여 자신이 소유한 공유 접두사 목록을 공유 해제하려면

AWS RAM 사용 설명서에서 [리소스 공유 업데이트](#)를 참조하세요.

AWS CLI를 사용하여 자신이 소유한 공유 접두사 목록을 공유 해제하려면

[disassociate-resource-share](#) 명령을 사용합니다.

공유 접두사 목록 사용 권한

소유자에 대한 권한

소유자는 공유 접두사 목록 및 해당 항목을 관리할 책임이 있습니다. 소유자는 접두사 목록을 참조하는 AWS 리소스의 ID를 볼 수 있습니다. 그러나 소비자가 소유한 AWS 리소스의 접두사 목록에 대한 참조를 추가하거나 제거할 수는 없습니다.

접두사 목록이 소비자가 소유한 리소스에서 참조되는 경우 소유자는 접두사 목록을 삭제할 수 없습니다.

소비자에 대한 권한

소비자는 공유 접두사 목록의 항목을 볼 수 있으며 AWS 리소스에서 공유 접두사 목록을 참조할 수 있습니다. 그러나 소비자는 공유 접두사 목록을 수정, 복원 또는 삭제할 수 없습니다.

결제 및 측정

접두사 목록 공유에 대한 추가 비용은 없습니다.

AWS RAM에 대한 할당량

자세한 내용은 [Service Quotas](#)을 참조하세요.

AWS 리소스의 참조 접두사 목록

다음 AWS 리소스에서 접두사 목록을 참조할 수 있습니다.

리소스

- [VPC 보안 그룹](#)
- [서브넷 라우팅 테이블](#)
- [전송 게이트웨이 라우팅 테이블](#)
- [AWS Network Firewall 규칙 그룹](#)
- [Amazon Managed Grafana 네트워크 액세스 제어](#)
- [AWS Outposts 랙 로컬 게이트웨이](#)

VPC 보안 그룹

접두사 목록을 인바운드 규칙의 소스로 지정하거나 아웃바운드 규칙의 대상으로 지정할 수 있습니다. 자세한 내용은 [보안 그룹](#) 단원을 참조하십시오.

콘솔을 사용하여 보안 그룹 규칙에서 접두사 목록을 참조하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Security Groups]를 선택합니다.
3. 업데이트할 보안 그룹을 선택합니다.
4. 작업, 인바운드 규칙 편집 또는 작업, 아웃바운드 규칙 편집을 선택합니다.
5. [Add another rule]을 선택합니다. 유형에서 트래픽 유형을 선택합니다. 소스(인바운드 규칙) 또는 대상(아웃바운드 규칙)에서 접두사 목록의 ID를 선택합니다.
6. 규칙 저장을 선택합니다.

AWS CLI를 사용하여 보안 그룹 규칙에서 접두사 목록을 참조하려면

[authorize-security-group-ingress](#) 및 [authorize-security-group-egress](#) 명령을 사용합니다. `--ip-permissions` 파라미터의 경우 `PrefixListIds`를 사용하여 접두사 목록의 ID를 지정합니다.

서브넷 라우팅 테이블

라우팅 테이블 항목의 대상으로 접두사 목록을 지정할 수 있습니다. 게이트웨이 라우팅 테이블에서는 접두사 목록을 참조할 수 없습니다. 라우팅 테이블에 대한 자세한 내용은 [라우팅 테이블 구성](#) 단원을 참조하십시오.

콘솔을 사용하여 라우팅 테이블에서 접두사 목록을 참조하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블을 선택한 후 라우팅 테이블을 선택합니다.
3. 작업(Actions), 라우팅 편집(Edit routes)을 선택합니다.
4. 라우팅을 추가하려면 라우팅 추가를 선택합니다.
5. 대상에 접두사 목록의 ID를 입력합니다.
6. 대상에서 대상을 선택합니다.
7. [Save changes]를 선택합니다.

AWS CLI를 사용하여 라우팅 테이블에서 접두사 목록을 참조하려면

[create-route](#)(AWS CLI) 명령을 사용합니다. `--destination-prefix-list-id` 파라미터를 사용하여 접두사 목록의 ID를 지정합니다.

전송 게이트웨이 라우팅 테이블

라우팅의 대상으로 접두사 목록을 지정할 수 있습니다. 자세한 내용은 Amazon VPC Transit Gateways의 [접두사 목록 참조](#)를 참조하십시오.

AWS Network Firewall 규칙 그룹

AWS Network Firewall 그룹은 네트워크 트래픽을 검사하고 처리하기 위한 재사용 가능한 기준 세트입니다. AWS Network Firewall에서 Suricata와 호환되는 상태 저장 규칙 그룹을 생성하는 경우 규칙 그룹에서 접두사 목록을 참조할 수 있습니다. 자세한 내용은 AWS Network Firewall 개발자 안내서의 [Amazon VPC 접두사 목록 참조](#) 및 [상태 저장 규칙 그룹 생성](#)을 참조하십시오.

Amazon Managed Grafana 네트워크 액세스 제어

하나 이상의 접두사 목록을 Amazon Managed Grafana 작업 영역에 대한 요청에 대한 인바운드 규칙으로 지정할 수 있습니다. 접두사 목록을 참조하는 방법을 포함하여 Grafana 작업 영역 네트워크 액세스 제어에 대한 자세한 내용은 Amazon Managed Grafana 사용 설명서의 [네트워크 액세스 관리](#)를 참조하세요.

AWS Outposts 랙 로컬 게이트웨이

각 AWS Outposts 랙에서는 Outpost 리소스를 온프레미스 네트워크와 연결할 수 있는 로컬 게이트웨이가 제공됩니다. 접두사 목록에서 자주 사용하는 CIDR를 그룹화하고 이 목록을 로컬 게이트웨이 라우팅 테이블의 라우팅 대상으로 참조할 수 있습니다. 자세한 내용은 랙용 AWS Outposts 사용 설명서의 [로컬 게이트웨이 라우팅 테이블 경로 관리](#)를 참조하세요.

AWS IP 주소 범위

AWS는 현재 IP 주소 범위를 JSON 형식으로 게시합니다. 이 정보를 사용하여 AWS에서 트래픽을 식별할 수 있습니다. 또한, 일부 AWS 서비스로 전송하거나 그로부터 수신되는 트래픽을 허용 또는 거부할 수도 있습니다.

Note

- 오직 일부 AWS 서비스 IP 주소 범위만 ip-ranges.json에 게시됩니다. 고객이 흔히 [송신 필터링](#) 작업을 수행하려고 하는 대상 서비스의 IP 주소 범위를 게시합니다.
- 서비스에서 IP 주소 범위를 사용하여 다른 서비스와 통신하거나 서비스에서 IP 범위를 사용하여 고객 네트워크와 통신할 수 있습니다.

현재 범위를 보려면 .json 파일을 다운로드합니다. 기록을 유지하려면 연속 버전의 .json 파일을 시스템에 저장합니다. 파일을 마지막으로 저장한 이후에 변경 사항이 있는지 확인하려면 현재 파일의 게시 시간을 마지막으로 저장한 파일의 게시 시간과 비교합니다.

고유 IP 주소 가져오기(BYOIP)를 통해 AWS로 가져온 IP 주소 범위는 .json 파일에 포함되지 않습니다.

또는 일부 서비스는 AWS 관리형 접두사 목록을 사용하여 주소 범위를 게시합니다. 자세한 내용은 [the section called “사용 가능한 AWS 관리형 접두사 목록”](#) 단원을 참조하십시오.

목차

- [다운로드](#)
- [구문](#)
- [범위 중복](#)
- [JSON 파일 필터링](#)
- [송신 제어 구현](#)
- [AWS IP 주소 범위 알림](#)
- [릴리스 정보](#)
- [자세히 알아보기](#)

다운로드

[ip-ranges.json](#)을 다운로드합니다.

이 파일에 프로그래밍 방식으로 액세스할 경우 애플리케이션에서 서버에 제공된 TLS 인증서를 확인한 이후에 파일을 다운로드하는지 직접 확인해야 합니다.

구문

ip-ranges.json 구문은 다음과 같습니다.

```
{
  "syncToken": "0123456789",
  "createDate": "yyyy-mm-dd-hh-mm-ss",
  "prefixes": [
    {
      "ip_prefix": "cidr",
      "region": "region",
```

```

    "network_border_group": "network_border_group",
    "service": "subset"
  }
],
"ipv6_prefixes": [
  {
    "ipv6_prefix": "cidr",
    "region": "region",
    "network_border_group": "network_border_group",
    "service": "subset"
  }
]
}

```

syncToken

게시 시간(Unix epoch 시간 형식)

타입: 문자열

예제: "syncToken": "1416435608"

createDate

발행 날짜 및 시간(UTC YY-MM-DD-hh-mm-ss 형식)

타입: 문자열

예제: "createDate": "2014-11-19-23-29-02"

prefixes

IPv4 주소 범위에 대한 IP 접두사

형식: 배열

ipv6_prefixes

IPv6 주소 범위에 대한 IP 접두사

형식: 배열

ip_prefix

퍼블릭 IPv4 주소 범위(CIDR 표기법)입니다. AWS에서는 구체적인 범위로 접두사를 알릴 수 있습니다. 예를 들어, 파일의 96.127.0.0/17 접두사를 96.127.0.0/21, 96.127.8.0/21, 96.127.32.0/19 및 96.127.64.0/18로 알릴 수 있습니다.

타입: 문자열

예제: "ip_prefix": "198.51.100.2/24"

ipv6_prefix

퍼블릭 IPv6 주소 범위(CIDR 표기법)입니다. AWS에서는 구체적인 범위로 접두사를 알릴 수 있습니다.

타입: 문자열

예제: "ipv6_prefix": "2001:db8:1234::/64"

network_border_group

AWS가 IP 주소 또는 GLOBAL을 알리는 가용 영역 또는 로컬 영역의 고유한 집합인 네트워크 경계 그룹의 이름입니다. GLOBAL 서비스의 트래픽이 AWS가 IP 주소를 알리는 여러 또는 모든 가용 영역이나 로컬 영역으로 유입되거나 이들 영역에서 시작될 수 있습니다.

타입: 문자열

예제: "network_border_group": "us-west-2-lax-1"

region

AWS 리전 또는 GLOBAL입니다. GLOBAL 서비스의 트래픽이 여러 또는 모든 AWS 리전으로 유입되거나 이들 리전에서 시작될 수 있습니다.

타입: 문자열

유효한 값: af-south-1 | ap-east-1 | ap-northeast-1 | ap-northeast-2 | ap-northeast-3 | ap-south-1 | ap-south-2 | ap-southeast-1 | ap-southeast-2 | ap-southeast-3 | ap-southeast-4 | ca-central-1 | cn-north-1 | cn-northwest-1 | eu-central-1 | eu-central-2 | eu-north-1 | eu-south-1 | eu-south-2 | eu-west-1 | eu-west-2 | eu-west-3 | me-central-1 | me-south-1 | sa-east-1 | us-east-1 | us-east-2 | us-gov-east-1 | us-gov-west-1 | us-west-1 | us-west-2 | GLOBAL

예제: "region": "us-east-1"

서비스

IP 주소 범위의 일부입니다. API_GATEWAY에 대해 나열된 주소는 송신 전용입니다. 모든 IP 주소 범위를 가져오도록 AMAZON을 지정합니다. 이렇게 하면 모든 서브셋이 AMAZON 서브셋에도 있습니다. 하지만 일부 IP 주소 범위는 AMAZON 서브셋에만 있습니다. 즉, 다른 서브셋에서 사용할 수 없습니다.

타입: 문자열

유효한 값: AMAZON | AMAZON_APPFLOW | AMAZON_CONNECT | API_GATEWAY | CHIME_MEETINGS | CHIME_VOICECONNECTOR | CLOUD9 | CLOUDFRONT | CLOUDFRONT_ORIGIN_FACING | CODEBUILD | DYNAMODB | EBS | EC2 | EC2_INSTANCE_CONNECT | GLOBALACCELERATOR | IVS_REALTIME | KINESIS_VIDEO_STREAMS | MEDIA_PACKAGE_V2 | ROUTE53 | ROUTE53_HEALTHCHECKS | ROUTE53_HEALTHCHECKS_PUBLISHING | ROUTE53_RESOLVER | S3 | WORKSPACES_GATEWAYS

예제: "service": "AMAZON"

범위 중복

모든 서비스 코드에서 반환되는 IP 주소 범위는 AMAZON 서비스 코드에서도 반환됩니다. 예를 들어 S3 서비스 코드에서 반환되는 모든 IP 주소 범위는 AMAZON 서비스 코드에서도 반환됩니다.

서비스 A가 서비스 B의 리소스를 사용하는 경우 서비스 A와 서비스 B 모두에 대해 서비스 코드에서 반환되는 IP 주소 범위가 있습니다. 그러나 이러한 IP 주소 범위는 서비스 A에서만 사용되며 서비스 B에서는 사용할 수 없습니다. 예를 들어 Amazon S3는 Amazon EC2의 리소스를 사용하므로 S3 및 EC2 서비스 코드 모두에서 반환되는 IP 주소 범위가 있습니다. 그러나 이러한 IP 주소 범위는 Amazon S3에서만 사용됩니다. 따라서 S3 서비스 코드는 Amazon S3에서만 사용하는 모든 IP 주소 범위를 반환합니다. Amazon EC2에서만 사용되는 IP 주소 범위를 식별하려면 EC2 서비스 코드에서 반환되지만 S3 서비스 코드에서는 반환되지 않는 IP 주소 범위를 찾으세요.

JSON 파일 필터링

명령줄 도구를 다운로드하여 원하는 정보만 표시하도록 필터링할 수 있습니다.

Windows

[AWS Tools for Windows PowerShell](#)에는 이 JSON 파일을 구문 분석하는 cmdlet인 `Get-AWSPublicIpAddressRange`가 포함되어 있습니다. 다음 예는 이 cmdlet의 사용법을 보여줍니다. 자세한 내용은 [Querying the Public IP Address Ranges for AWS](#) 및 [Get-AWSPublicIpAddressRange](#) 단원을 참조하십시오.

Example 1. 생성 날짜 가져오기

```
PS C:\> Get-AWSPublicIpAddressRange -OutputPublicationDate
```

```
Wednesday, August 22, 2018 9:22:35 PM
```


Example 2. 특정 리전에 대한 정보 가져오기

```
PS C:\> Get-AWSPublicIpAddressRange -Region us-east-1
```

IpPrefix	Region	NetworkBorderGroup	Service
-----	-----	-----	-----
23.20.0.0/14	us-east-1	us-east-1	AMAZON
50.16.0.0/15	us-east-1	us-east-1	AMAZON
50.19.0.0/16	us-east-1	us-east-1	AMAZON
...			

Example 3. 모든 IP 주소 가져오기

```
PS C:\> (Get-AWSPublicIpAddressRange).IpPrefix
23.20.0.0/14
27.0.0.0/22
43.250.192.0/24
...
2406:da00:ff00::/64
2600:1fff:6000::/40
2a01:578:3::/64
2600:9000::/28
```

Example 4. 모든 IPv4 주소 가져오기

```
PS C:\> Get-AWSPublicIpAddressRange | where {$_.IpAddressFormat -eq "Ipv4"} | select
  IpPrefix

IpPrefix
-----
23.20.0.0/14
27.0.0.0/22
43.250.192.0/24
...
```

Example 5. 모든 IPv6 주소 가져오기

```
PS C:\> Get-AWSPublicIpAddressRange | where {$_.IpAddressFormat -eq "Ipv6"} | select
  IpPrefix

IpPrefix
-----
```

```
2a05:d07c:2000::/40
2a05:d000:8000::/40
2406:dafe:2000::/40
...
```

Example 6. 특정 서비스에 대한 모든 IP 주소를 가져옵니다.

```
PS C:\> Get-AWSPublicIpAddressRange -ServiceKey CODEBUILD | select IpPrefix

IpPrefix
-----
52.47.73.72/29
13.55.255.216/29
52.15.247.208/29
...
```

Linux

다음 예제 명령에서는 [jq 도구](#)를 사용하여 JSON 파일의 로컬 사본을 구문 분석합니다.

Example 1. 생성 날짜 가져오기

```
$ jq .createDate < ip-ranges.json

"2016-02-18-17-22-15"
```

Example 2. 특정 리전에 대한 정보 가져오기

```
$ jq '.prefixes[] | select(.region=="us-east-1")' < ip-ranges.json

{
  "ip_prefix": "23.20.0.0/14",
  "region": "us-east-1",
  "network_border_group": "us-east-1",
  "service": "AMAZON"
},
{
  "ip_prefix": "50.16.0.0/15",
  "region": "us-east-1",
  "network_border_group": "us-east-1",
  "service": "AMAZON"
},
```

```
{
  "ip_prefix": "50.19.0.0/16",
  "region": "us-east-1",
  "network_border_group": "us-east-1",
  "service": "AMAZON"
},
...
```

Example 3. 모든 IPv4 주소 가져오기

```
$ jq -r '.prefixes | [].ip_prefix' < ip-ranges.json

23.20.0.0/14
27.0.0.0/22
43.250.192.0/24
...
```

Example 4. 모든 IPv6 주소 가져오기

```
$ jq -r '.ipv6_prefixes | [].ipv6_prefix' < ip-ranges.json

2a05:d07c:2000::/40
2a05:d000:8000::/40
2406:dafe:2000::/40
...
```

Example 5. 특정 서비스에 대한 모든 IPv4 주소를 가져옵니다.

```
$ jq -r '.prefixes[] | select(.service=="CODEBUILD") | .ip_prefix' < ip-ranges.json

52.47.73.72/29
13.55.255.216/29
52.15.247.208/29
...
```

Example 6. 특정 리전의 특정 서비스에 대한 모든 IPv4 주소 가져오기

```
$ jq -r '.prefixes[] | select(.region=="us-east-1") | select(.service=="CODEBUILD")
| .ip_prefix' < ip-ranges.json

34.228.4.208/28
```

Example 7. 특정 네트워크 경계 그룹에 대한 정보 가져오기

```
$ jq -r '.prefixes[] | select(.region=="us-west-2") |
  select(.network_border_group=="us-west-2-lax-1") | .ip_prefix' < ip-ranges.json
70.224.192.0/18
52.95.230.0/24
15.253.0.0/16
...
```

송신 제어 구현

하나의 AWS 서비스를 이용하여 생성한 리소스가 다른 AWS 서비스에만 액세스하는 것을 허용하기 위해서는 ip-ranges.json 파일에 있는 IP 주소 범위 정보를 사용하여 [송신 필터링](#) 작업을 수행할 수 있습니다. 보안 그룹 규칙에서 AMAZON 목록의 CIDR 블록으로의 아웃바운드 트래픽이 허용되는지 확인하세요. [보안 그룹에 대한 할당량](#)이 있습니다. 각 리전의 IP 주소 범위 수에 따라 리전당 여러 보안 그룹이 필요할 수 있습니다.

Note

일부 AWS 서비스는 EC2를 기반으로 구축되었으며 EC2 IP 주소 공간을 사용합니다. EC2 IP 주소 공간으로의 트래픽을 차단할 경우 이러한 EC2 이외의 서비스에 대한 트래픽도 차단됩니다.

AWS IP 주소 범위 알림

AWS IP 주소 범위가 변경될 때마다 AmazonIpSpaceChanged 주제 구독자에게 알림을 보냅니다. 페이로드에는 다음 형식의 정보가 포함되어 있습니다.

```
{
  "create-time": "yyyy-mm-ddThh:mm:ss+00:00",
  "synctoken": "0123456789",
  "md5": "6a45316e8bc9463c9e926d5d37836d33",
  "url": "https://ip-ranges.amazonaws.com/ip-ranges.json"
}
```

create-time

생성 날짜 및 시간

알림이 제공되는 순서는 정해져 있지 않습니다. 따라서 타임스탬프를 통해 올바른 순서를 확인하는 것이 좋습니다.

synctoken

게시 시간(Unix epoch 시간 형식)

md5

ip-ranges.json 파일의 암호화 해시 값입니다. 이 값을 사용하여 다운로드한 파일이 손상되었는지 여부를 확인할 수 있습니다.

url

ip-ranges.json 파일의 위치

AWS IP 주소 범위가 변경될 때마다 알림을 받으려면 Amazon SNS를 사용하여 알림을 수신하도록 다음과 같이 구독할 수 있습니다.

AWS IP 주소 범위 알림을 구독하려면

1. <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 필요한 경우 탐색 모음에서 리전을 미국 동부(버지니아 북부)로 변경합니다. 구독하는 SNS 알림이 이 리전에 생성되었기 때문에 이 리전을 선택해야 합니다.
3. 탐색 창에서 구독을 선택합니다.
4. Create subscription을 선택합니다.
5. 구독 생성 대화 상자에서 다음 작업을 수행합니다.

- a. [Topic ARN]의 경우, 다음 Amazon 리소스 이름(ARN)을 복사합니다.

```
arn:aws:sns:us-east-1:806199016981:AmazonIpSpaceChanged
```

- b. [Protocol]의 경우, 사용할 프로토콜을 선택합니다(예: Email).
 - c. [Endpoint]의 경우, 알림을 받을 엔드포인트를 입력합니다(예: 이메일 주소).
 - d. 구독 생성을 선택합니다.
6. 지정한 엔드포인트에서 연락이 오고 구독을 확인하라는 메시지가 표시됩니다. 예를 들어 이메일 주소를 지정한 경우, 제목이 AWS Notification - Subscription Confirmation인 이메일 메시지를 받게 됩니다. 지시에 따라 구독을 확인합니다.

알림은 엔드포인트의 가용성을 따릅니다. 따라서 JSON 파일을 주기적으로 확인하여 최신 범위를 가져왔는지 확인할 수 있습니다. Amazon SNS 안정성에 대한 자세한 내용은 <https://aws.amazon.com/sns/faqs/#Reliability>를 참조하세요.

이런 알림을 더 이상 받지 않기를 원하는 경우, 다음 절차를 수행해서 구독을 해제하세요.

AWS IP 주소 범위 알림을 구독 해제하려면

1. <https://console.aws.amazon.com/sns/v3/home>에서 Amazon SNS 콘솔을 엽니다.
2. 탐색 창에서 구독을 선택합니다.
3. 구독의 확인란을 선택합니다.
4. [Actions], [Delete subscriptions]를 차례로 선택합니다.
5. 확인 메시지가 나타나면 삭제를 선택합니다.

Amazon SNS에 대한 자세한 내용은 [Amazon Simple Notification Service 개발자 안내서](#)를 참조하세요.

릴리스 정보

다음 표에서는 ip-ranges.json 구문에 대한 업데이트를 설명합니다. 또한 각 리전 출시와 함께 새로운 리전 코드도 추가됩니다.

설명	릴리스 날짜
IVS_REALTIME 서비스 코드를 추가했습니다.	2024년 6월 11일
MEDIA_PACKAGE_V2 서비스 코드를 추가했습니다.	2023년 5월 9일
CLOUDFRONT_ORIGIN_FACING 서비스 코드를 추가했습니다.	2021년 10월 12일
ROUTE53_RESOLVER 서비스 코드를 추가했습니다.	2021년 6월 24일
EBS 서비스 코드를 추가했습니다.	2021년 5월 12일

설명	릴리스 날짜
KINESIS_VIDEO_STREAMS 서비스 코드를 추가했습니다.	2020년 11월 19일
CHIME_MEETINGS 및 CHIME_VOI CECONNECTOR 서비스 코드를 추가했습니다.	2020년 6월 19일
AMAZON_APPFLOW 서비스 코드를 추가했습니다.	2020년 6월 9일
네트워크 경계 그룹에 대한 지원을 추가합니다.	2020년 4월 7일
WORKSPACES_GATEWAYS 서비스 코드를 추가했습니다.	2020년 3월 30일
ROUTE53_HEALTHCHECK_PUBLISHING 서비스 코드를 추가했습니다.	2020년 1월 30일
API_GATEWAY 서비스 코드를 추가했습니다.	2019년 9월 26일
EC2_INSTANCE_CONNECT 서비스 코드를 추가했습니다.	2019년 26월 6일
DYNAMODB 서비스 코드를 추가했습니다.	2019년 4월 25일
GLOBALACCELERATOR 서비스 코드를 추가했습니다.	2018년 12월 20일
AMAZON_CONNECT 서비스 코드를 추가했습니다.	2018년 6월 20일
CLOUD9 서비스 코드를 추가했습니다.	2018년 6월 20일
CODEBUILD 서비스 코드를 추가했습니다.	2018년 4월 19일
S3 서비스 코드를 추가했습니다.	2017년 2월 28일
IPv6 주소 범위에 대한 지원을 추가했습니다.	2016년 8월 22일
최초 릴리스	2014년 11월 19일

자세히 알아보기

- AMAZON_APPFLOW – [IP 주소 범위](#)
- AMAZON_CONNECT – [네트워크 설정](#)
- CHIME_MEETINGS – [미디어 및 신호 전송 구성](#)
- CLOUDFRONT – [CloudFront 엣지 서버의 위치 및 IP 주소 범위](#)
- DYNAMODB – [IP 주소 범위](#)
- EC2 – [퍼블릭 IPv4 주소](#)
- EC2_INSTANCE_CONNECT – [EC2 인스턴스 연결 전제 조건](#)
- GLOBALACCELERATOR – [Global Accelerator 엣지 서버의 위치 및 IP 주소 범위](#)
- ROUTE53 – [Amazon Route 53 서버의 IP 주소 범위](#)
- ROUTE53_HEALTHCHECKS – [Amazon Route 53 서버의 IP 주소 범위](#)
- ROUTE53_HEALTHCHECKS_PUBLISHING – [Amazon Route 53 서버의 IP 주소 범위](#)
- WORKSPACES_GATEWAYS – [PCoIP 게이트웨이 서버](#)

VPC에 IPv6 지원 추가

IPv4만을 지원하는 기존 VPC와 서브넷에서 IPv4만을 사용하도록 구성된 리소스가 있으면 VPC 및 리소스에 대한 IPv6 지원을 추가할 수 있습니다. VPC는 듀얼 스택 모드로 작동할 수 있으므로, 리소스는 IPv4나 IPv6 또는 둘 다를 통해 통신할 수 있습니다. IPv4 및 IPv6 통신 프로토콜은 상호 독립적입니다.

VPC 및 서브넷에 대한 IPv4 지원은 Amazon VPC 및 Amazon EC2의 기본 IP 주소 지정 시스템이므로 비활성화할 수 없습니다.

고려 사항

- IPv4 전용 서브넷에서 IPv6 전용 서브넷으로의 마이그레이션 경로가 없습니다.
- 이 예시에서는 퍼블릭 및 프라이빗 서브넷을 포함하는 VPC를 보유하고 있다고 가정합니다. IPv6에서 사용할 새 VPC 생성에 대한 자세한 내용은 [the section called “VPC 생성”](#) 섹션을 참조하세요.
- IPv6를 사용하여 시작하기 전에 Amazon VPC에 대한 IPv6 주소 지정 기능을 읽어 보시길 바랍니다 ([IPv4 및 IPv6 비교](#)).

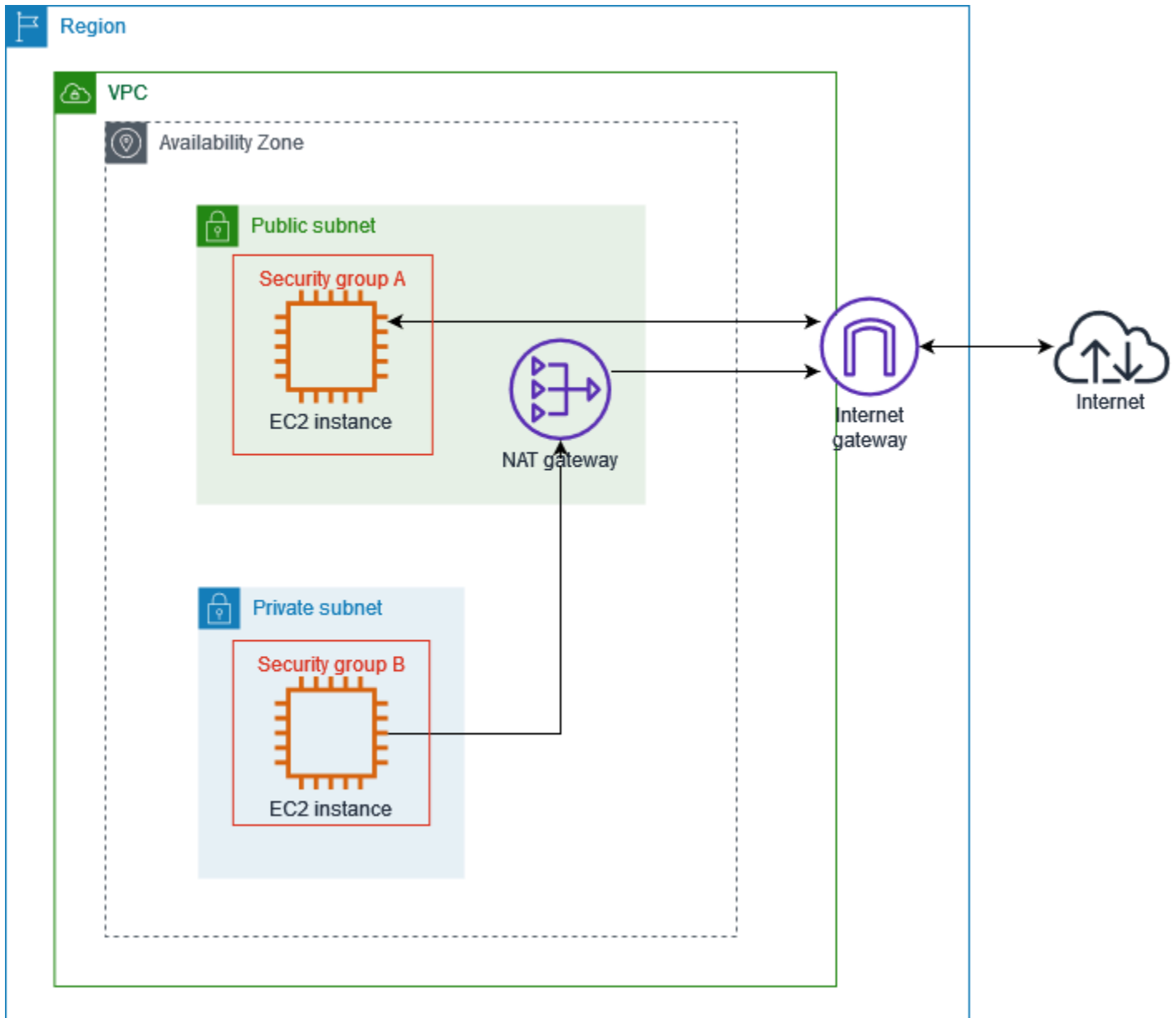
프로세스

다음 표는 VPC에 대해 IPv6를 활성화하는 프로세스의 개요를 간략히 설명합니다.

단계	참고
1단계: IPv6 CIDR 블록을 VPC 및 서브넷에 연결	Amazon 제공 또는 BYOIP IPv6 CIDR 블록을 VPC 및 서브넷과 연결합니다.
2단계: 라우팅 테이블 업데이트	라우팅 테이블을 업데이트하여 IPv6 트래픽을 라우팅합니다. 퍼블릭 서브넷의 경우, 서브넷의 모든 IPv6 트래픽을 인터넷 게이트웨이로 라우팅하는 경로를 생성합니다. 프라이빗 서브넷의 경우, 서브넷의 인터넷 바인딩된 모든 IPv6 트래픽을 외부 전용 인터넷 게이트웨이로 라우팅하는 경로를 생성합니다.
3단계: 보안 그룹 규칙 업데이트	보안 그룹 규칙을 업데이트하여 IPv6 주소용 규칙을 포함합니다. 이렇게 하면 IPv6 트래픽이 인스턴스로 그리고 인스턴스로부터 흐르도록 할 수 있습니다. 서브넷으로 가는, 그리고 서브넷에서 나오는 트래픽의 흐름을 제어하기 위해 사용자 지정 네트워크 ACL 규칙을 생성한 경우, IPv6 트래픽에 대한 규칙을 포함시켜야 합니다.
4단계: 인스턴스에 IPv6 주소 할당	서브넷의 IPv6 주소 범위에서 인스턴스에 IPv6 주소를 할당합니다.

예: 퍼블릭 및 프라이빗 서브넷이 있는 VPC에서 IPv6 사용

이 예에서는 VPC에 퍼블릭 및 프라이빗 서브넷이 있습니다. VPC의 NAT 게이트웨이를 통해 인터넷과 아웃바운드 통신을 하는 프라이빗 서브넷에 데이터베이스 인스턴스가 있습니다. 인터넷 게이트웨이를 통해 인터넷에 액세스할 수 있는 퍼블릭 서브넷에 퍼블릭 웹 서버가 있습니다. 다음 다이어그램은 VPC의 아키텍처를 보여줍니다.



웹 서버 보안 그룹(예: 보안 그룹 ID sg-11aa22bb11aa22bb1)에는 다음과 같은 인바운드 규칙이 포함되어 있습니다.

Type	프로토콜	포트 범위	소스	Comment
모든 트래픽	모두	모두	sg-33cc44 dd33cc44dd3	sg-33cc44 dd33cc44dd3(데이터베이스 인스턴스)과 연결된 인스턴스에서 나

Type	프로토콜	포트 범위	소스	Comment
				오는 모든 트래픽에 대한 인바운드 액세스 허용
HTTP	TCP	80	0.0.0.0/0	HTTP를 통해 인터넷에서 오는 인바운드 트래픽을 허용합니다.
HTTPS	TCP	443	0.0.0.0/0	HTTPS를 통해 인터넷에서 오는 인바운드 트래픽을 허용합니다.
SSH	TCP	22	203.0.113.123/32	로컬 컴퓨터에서 접근하는 인바운드 SSH 액세스를 허용. 예를 들면, 관리 작업을 수행하기 위해 인스턴스에 연결해야 하는 경우.

데이터베이스 인스턴스 보안 그룹(예: 보안 그룹 ID sg-33cc44dd33cc44dd3)에는 다음과 같은 인바운드 규칙이 포함되어 있습니다.

Type	프로토콜	포트 범위	소스	Comment
MySQL	TCP	3306	sg-11aa22 bb11aa22bb1	sg-11aa22 bb11aa22bb1(웹 서버 인스턴스)과 연결된 인스턴스에서 나오는 MySQL 트래픽에

Type	프로토콜	포트 범위	소스	Comment
				대한 인바운드 액세스 허용

두 보안 그룹에는 모든 아웃바운드 IPv4 트래픽을 허용하는 기본 아웃바운드 규칙이 있고, 다른 아웃바운드 규칙은 없습니다.

웹 서버는 t2.medium 인스턴스 유형입니다. 데이터베이스 서버는 m3.large입니다.

VPC 및 리소스에서 IPv6를 사용하고 VPC 및 리소스를 듀얼 스택 모드로 작동하고 싶은 경우가 있습니다. 바꿔 말하면 인터넷을 통해 VPC 및 리소스의 리소스들 간에 IPv6 및 IPv4 주소 지정을 모두 사용하는 것입니다.

1단계: IPv6 CIDR 블록을 VPC 및 서브넷에 연결

IPv6 CIDR 블록을 VPC와 연결한 다음, 그 범위의 /64 CIDR 블록을 각 서브넷에 연결할 수 있습니다.

IPv6 CIDR 블록을 VPC와 연결하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 사용자 VPC(Your VPCs)를 선택합니다.
3. 해당 VPC를 선택합니다.
4. 작업, CIDR 편집을 선택한 후 새 IPv6 CIDR 추가를 선택합니다.
5. 다음 옵션 중 하나를 선택한 다음 CIDR 선택을 선택합니다.
 - Amazon에서 제공한 IPv6 CIDR 블록 - Amazon의 IPv6 주소 풀에서 IPv6 CIDR 블록을 사용합니다. 네트워크 경계 그룹에서 AWS가 IP 주소를 알리는 그룹을 선택합니다.
 - IPAM에 할당된 IPv6 CIDR 블록 - [IPAM 풀](#)의 IPv6 CIDR 블록을 사용합니다. IPAM 풀과 IPv6 CIDR 블록을 선택합니다.
 - 내가 소유한 IPv6 CIDR - IPv6 주소 풀에서 IPv6 CIDR 블록을 사용합니다([BYOIP](#)). IPv6 주소 풀과 IPv6 CIDR 블록을 선택합니다.
6. 달기를 선택하세요.

IPv6 CIDR 블록을 서브넷에 연결하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.

2. 탐색 창에서 서브넷(Subnets)을 선택합니다.
3. 서브넷을 선택합니다.
4. 작업, IPv6 CIDR 편집을 선택한 후 IPv6 CIDR 추가를 선택합니다.
5. 필요한 경우 CIDR 블록을 편집합니다(예: 00 교체).
6. Save(저장)를 선택합니다.
7. 이 절차를 VPC의 다른 서브넷에 반복합니다.

자세한 내용은 [IPv6 VPC CIDR 블록](#) 단원을 참조하십시오.

2단계: 라우팅 테이블 업데이트

IPv6 CIDR 블록을 VPC에 연결하면 로컬 경로가 각 라우팅 테이블에 자동으로 추가되어 VPC 내부에서 IPv6 트래픽이 허용됩니다.

인스턴스(예: 웹 서버)가 IPv6 트래픽용 인터넷 게이트웨이를 사용할 수 있도록 퍼블릭 서브넷의 라우팅 테이블을 업데이트해야 합니다. 또한 NAT 게이트웨이에서 IPv6이 지원되지 않기 때문에 인스턴스(예: 데이터베이스 인스턴스)가 IPv6 트래픽용 송신 전용 인터넷 게이트웨이를 사용할 수 있도록 프라이빗 서브넷의 라우팅 테이블을 업데이트해야 합니다.

퍼블릭 서브넷의 라우팅 테이블을 업데이트하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 서브넷(Subnets)을 선택합니다. 퍼블릭 서브넷을 선택합니다. 라우팅 테이블 탭에서 라우팅 테이블 ID를 선택하여 라우팅 테이블에 대한 세부 정보 페이지를 엽니다.
3. 라우팅 테이블을 선택합니다. 라우팅 탭에서 라우팅 편집을 선택합니다.
4. 라우팅 추가를 선택합니다. 대상에 대해 `::/0`을 선택합니다. 대상에 대한 인터넷 게이트웨이의 ID를 선택합니다.
5. Save changes(변경 사항 저장)를 선택합니다.

프라이빗 서브넷의 라우팅 테이블을 업데이트하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 송신 전용 인터넷 게이트웨이를 선택합니다. 송신 전용 인터넷 게이트웨이 생성을 선택합니다. VPC에서 VPC를 선택한 후 송신 전용 인터넷 게이트웨이 만들기를 선택합니다.

자세한 내용은 [송신 전용 인터넷 게이트웨이를 사용하여 아웃바운드 IPv6 트래픽 활성화](#) 단원을 참조하십시오.

3. 탐색 창에서 서브넷을 선택합니다. 프라이빗 서브넷을 선택합니다. 라우팅 테이블 탭에서 라우팅 테이블 ID를 선택하여 라우팅 테이블에 대한 세부 정보 페이지를 엽니다.
4. 라우팅 테이블을 선택합니다. 라우팅 탭에서 라우팅 편집을 선택합니다.
5. 라우팅 추가를 선택합니다. 대상에 대해 `::/0`을 선택합니다. 대상에 대한 송신 전용 인터넷 게이트웨이의 ID를 선택합니다.
6. Save changes(변경 사항 저장)를 선택합니다.

자세한 내용은 [라우팅 옵션](#) 에 단원을 참조하십시오.

3단계: 보안 그룹 규칙 업데이트

인스턴스가 IPv6을 통해 트래픽을 보내고 받을 수 있도록 하려면 보안 그룹 규칙을 업데이트하여 IPv6 주소용 규칙을 포함해야 합니다. 예를 들어 위의 예에서, 웹 서버 보안 그룹 (sg-11aa22bb11aa22bb1)을 업데이트하여 IPv6 주소로부터 인바운드 HTTP, HTTPS, SSH 액세스를 허용하는 규칙을 추가할 수 있습니다. 데이터베이스 보안 그룹에 대한 인바운드 규칙을 변경할 필요는 없습니다. sg-11aa22bb11aa22bb1의 모든 통신을 허용하는 규칙에는 IPv6 통신이 포함되어 있습니다.

인바운드 보안 그룹 규칙을 업데이트하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택하고 웹 서버 보안 그룹을 선택합니다.
3. 인바운드 규칙 탭에서 인바운드 규칙 편집을 선택합니다.
4. IPv4 트래픽이 허용되는 각 규칙에 대해 규칙 추가를 선택하고 해당 IPv6 트래픽이 허용되도록 규칙을 구성합니다. 예를 들어 IPv6를 통한 모든 HTTP 트래픽을 허용하는 규칙을 추가하려면 유형에서 HTTP를 선택하고 소스에서 `::/0`을 선택합니다.
5. 규칙 추가가 완료되면 규칙 저장을 선택합니다.

아웃바운드 보안 그룹 규칙 업데이트

IPv6 CIDR 블록을 VPC와 연결할 경우 모든 IPv6 트래픽을 허용하는 VPC의 보안 그룹에 아웃바운드 규칙이 자동으로 추가됩니다. 그러나 보안 그룹에 대한 원본 아웃바운드 규칙을 수정한 경우, 이 규칙

은 자동으로 추가되지 않으므로 IPv6 트래픽에 대한 동등한 수준의 아웃바운드 규칙을 추가해야 합니다.

네트워크 ACL 규칙 업데이트

IPv6 CIDR 블록을 VPC와 연결하면 규칙이 기본 네트워크 ACL에 자동으로 추가되어 IPv6 트래픽이 허용됩니다. 그러나 기본 네트워크 ACL을 수정했거나 사용자 지정 네트워크 ACL을 생성한 경우 IPv6 트래픽에 대한 규칙을 수동으로 추가해야 합니다. 자세한 내용은 [네트워크 ACL 작업](#) 단원을 참조하십시오.

4단계: 인스턴스에 IPv6 주소 할당

현재 세대의 모든 인스턴스 유형은 IPv6를 지원합니다. 인스턴스 유형이 IPv6를 지원하지 않는 경우 IPv6 주소를 할당하기 전에 지원되는 인스턴스 유형에 맞게 인스턴스 크기를 조정해야 합니다. 사용하는 프로세스는 선택한 새 인스턴스 유형과 현재 인스턴스 유형과의 호환 가능 여부에 의해 결정됩니다. 자세한 내용을 알아보려면 Amazon EC2 사용 설명서의 [인스턴스 유형 변경](#)을 참조하세요. IPv6를 지원하기 위해 새 AMI에서 인스턴스를 실행해야 하는 경우 시작 중에 인스턴스에 IPv6 주소를 할당할 수 있습니다.

인스턴스 유형이 IPv6를 지원한다는 것을 확인했으면 Amazon EC2 콘솔을 사용하여 인스턴스에 IPv6 주소를 할당할 수 있습니다. IPv6 주소는 인스턴스에 대한 주 네트워크 인터페이스(eth0)에 할당됩니다. 자세한 내용은 Amazon EC2 사용 설명서의 [인스턴스에 IPv6 주소 할당](#)을 참조하세요.

IPv6 주소를 사용하여 인스턴스에 연결할 수 있습니다. 자세한 내용은 Amazon EC2 사용 설명서의 [SSH 클라이언트를 사용하여 Linux 인스턴스에 연결](#) 또는 Amazon EC2 사용 설명서의 [IPv6 주소를 사용하여 Windows 인스턴스에 연결](#)을 참조하세요.

운영 체제의 현재 버전에 맞는 AMI를 사용하여 인스턴스를 실행한 경우에는 인스턴스가 IPv6에 대해 구성됩니다. 인스턴스에서 IPv6 주소를 ping할 수 없는 경우에는 운영 체제에 대한 설명서를 참조하여 IPv6를 구성하세요.

IPv6를 지원하는 AWS 서비스

컴퓨터와 스마트 장치는 IP 주소를 사용하여 인터넷 및 기타 네트워크를 통해 서로 통신합니다. 인터넷이 계속 성장함에 따라 IP 주소에 대한 필요성도 증가하고 있습니다. IP 주소의 가장 일반적인 형식은 IPv4입니다. IP 주소의 새로운 형식은 IPv4보다 더 큰 주소 공간을 제공하는 IPv6입니다.

IPv6에 대한 AWS 서비스 지원에는 듀얼 스택 구성(IPv4 및 IPv6) 또는 IPv6 전용 구성에 대한 지원이 포함됩니다. 예를 들어 Virtual Private Cloud(VPC)는 AWS 리소스를 시작할 수 있는 AWS 클라우드의

논리적으로 격리된 영역입니다. VPC 내에서 IPv4 전용, 듀얼 스택 또는 IPv6 전용 서브넷을 생성할 수 있습니다.

AWS 서비스는 퍼블릭 엔드포인트를 통한 액세스를 지원합니다. 일부 AWS 서비스는 AWS PrivateLink에서 제공하는 프라이빗 엔드포인트를 사용하는 액세스도 지원합니다. AWS 서비스는 퍼블릭 엔드포인트를 통해 IPv6를 지원하지 않더라도 프라이빗 엔드포인트를 통해 IPv6를 지원할 수 있습니다. IPv6를 지원하는 엔드포인트는 AAAA 레코드를 사용하여 DNS 쿼리에 응답할 수 있습니다.

IPv6를 지원하는 서비스

다음 표에는 듀얼 스택 지원, IPv6 전용 지원 및 IPv6를 지원하는 엔드포인트를 제공하는 AWS 서비스가 나와 있습니다. IPv6에 대한 추가 지원이 출시되면 이 표를 업데이트할 예정입니다. 서비스에서 IPv6이 지원되는 방법에 대한 구체적인 사항을 확인하려면 서비스에 대한 설명서를 참조하세요.




서비스 이름	듀얼 스택 지원	IPv6 전용 지원	IPv6를 지원하는 퍼블릭 엔드포인트	IPv6를 지원하는 프라이빗 엔드포인트 ¹
AWS App Mesh	 예	 예	 예	 아니요
Amazon AppStream 2.0	 예	 아니요	 아니요	 아니요
Amazon Athena	 예	 아니요	 예	 예
Amazon Aurora	 예	 아니요	 예	 아니요







서비스 이름	듀얼 스택 지원	IPv6 전용 지원	IPv6를 지원하는 퍼블릭 엔드포인트	IPv6를 지원하는 프라이빗 엔드포인트 ¹
AWS Cloud9	 <u>예</u>	 아니요	 예	
Amazon CloudFront	 <u>예</u>	 아니요	 아니요	
Amazon CloudWatch Logs		 <u>아</u> 아니요	 예	 아니요
AWS Cloud Map	 <u>예</u>	 예	 예	 예
AWS 클라우드 WAN	 예	 아니요	 예	 아니요
Amazon Cognito	 예	 아니요	 예	

서비스 이름	듀얼 스택 지원	IPv6 전용 지원	IPv6를 지원하는 퍼블릭 엔드포인트	IPv6를 지원하는 프라이빗 엔드포인트 ¹
AWS Database Migration Service	 <u>예</u>	 아니요	 아니요	 아니요
AWS Direct Connect	 예	 예	 아니요	
Amazon EC2	 <u>예</u>	 예	 <u>예</u>	 아니요
Amazon ECS	 <u>예</u>	 아니요	 아니요	 아니요
Amazon EKS	<u>노드: 예 / 포트: 아니요</u>	<u>포트: 예 / 노드: 아니요</u>	 아니요	 아니요
Elastic Load Balancing	로드 밸런서: 예 대상 그룹: 아니요	로드 밸런서: 아니요 대상 그룹: 예	 아니요	 아니요

서비스 이름	듀얼 스택 지원	IPv6 전용 지원	IPv6를 지원하는 퍼블릭 엔드포인트	IPv6를 지원하는 프라이빗 엔드포인트 ¹
Amazon ElastiCache	 <u>예</u>	 예	 아니요	 아니요
AWS Fargate	 <u>예</u>	 아니요	 아니요	 아니요
AWS Global Accelerator	 <u>예</u>	 아니요	 아니요	
AWS Glue	 아니요	 아니요	 아니요	 예
AWS IoT	 예	 아니요	 <u>예</u>	 아니요
AWS Lake Formation	 아니요	 아니요	 아니요	 예

서비스 이름	듀얼 스택 지원	IPv6 전용 지원	IPv6를 지원하는 퍼블릭 엔드포인트	IPv6를 지원하는 프라이빗 엔드포인트 ¹
AWS Lambda	 예	 아니요	 예	 아니요
Amazon Lightsail	 예	 예	 아니요	
AWS Network Firewall	 예	 예	 아니요	
Amazon OpenSearch Service	 예	 아니요	 예	 아니요
AWS PrivateLink	 예	 예	 예	
Amazon RDS	 예	 아니요	 예	 아니요

서비스 이름	듀얼 스택 지원	IPv6 전용 지원	IPv6를 지원하는 퍼블릭 엔드포인트	IPv6를 지원하는 프라이빗 엔드포인트 ¹
Amazon Route 53	 예	 예	 아니요	
Amazon S3	 예	 아니요	 예	 아니요
AWS Secrets Manager	 예	 아니요	 예	 아니요
AWS Shield	 예	 예	 아니요	
AWS Site-to-Site VPN	 예	 아니요	 예	 아니요
AWS Transit Gateway	 예	 아니요	 예	 아니요

서비스 이름	듀얼 스택 지원	IPv6 전용 지원	IPv6를 지원하는 퍼블릭 엔드포인트	IPv6를 지원하는 프라이빗 엔드포인트 ¹
Amazon VPC	 <u>예</u>	 예	 <u>예</u>	 아니요
AWS WAF	 <u>예</u>	 예	 아니요	
Amazon WorkSpaces	 <u>예</u>	 아니요	 아니요	 아니요

¹ 빈 셀은 해당 서비스가 [AWS PrivateLink와 통합](#)되지 않음을 나타냅니다.

추가 IPv6 지원

컴퓨팅

- Amazon EC2는 Nitro System 기반 인스턴스를 IPv6 전용 서브넷에서 시작할 수 있도록 지원합니다.
- Amazon EC2는 인스턴스 메타데이터 서비스(IMDS) 및 Amazon Time Sync Service에 대한 IPv6 엔드포인트를 제공합니다.

네트워킹 및 콘텐츠 전송

- Amazon VPC는 IPv6 전용 서브넷 생성을 지원합니다.
- Amazon VPC는 서브넷에서 DNS64를 지원하고 NAT 게이트웨이에서 NAT64를 지원하여 IPv6 AWS 리소스가 IPv4 리소스와 통신할 수 있도록 돕습니다.

보안, 자격 증명 및 규정 준수

- AWS Identity and Access Management(IAM)는 IAM 정책에서 IPv6 주소를 지원합니다.
- Amazon Macie는 개인 식별 정보 (PII)에서 IPv6 주소를 지원합니다.

관리 및 거버넌스

- AWS CloudTrail 레코드에는 소스 IPv6 정보가 포함됩니다.
- AWS CLI v2는 IPv6 전용 클라이언트에 대해 IPv6 연결을 통한 다운로드를 지원합니다.

자세히 알아보기

- [AWS에서의 IPv6](#)
- [듀얼 스택 및 IPv6 전용 Amazon VPC 참조 아키텍처\(PDF\)](#)

Virtual Private Cloud(VPC)

Virtual Private Cloud(VPC)는 사용자의 AWS 계정 전용 가상 네트워크입니다. VPC는 AWS 클라우드에서 다른 가상 네트워크와 논리적으로 분리되어 있습니다. AWS 리소스(예: Amazon EC2 인스턴스)를 VPC에서 시작할 수 있습니다.

계정에는 각 AWS 리전의 기본 VPC가 포함되어 있습니다. 추가 VPC를 생성할 수도 있습니다.

내용

- [VPC 기초](#)
- [기본 VPC](#)
- [VPC 생성](#)
- [VPC 구성](#)
- [Amazon VPC의 DHCP 옵션 세트](#)
- [VPC의 DNS 속성](#)
- [VPC의 네트워크 주소 사용량](#)
- [다른 계정과 VPC 공유](#)
- [로컬 영역, Wavelength 영역 또는 Outpost로 VPC 확장](#)
- [VPC 삭제](#)

VPC 기초

VPC는 리전의 모든 가용 영역에 적용됩니다. VPC를 생성한 후 각 가용 영역에 하나 이상의 서브넷을 추가할 수 있습니다. 자세한 내용은 [서브넷](#) 단원을 참조하십시오.

목차

- [VPC IP 주소 범위](#)
- [VPC 다이어그램](#)
- [VPC 리소스](#)

VPC IP 주소 범위

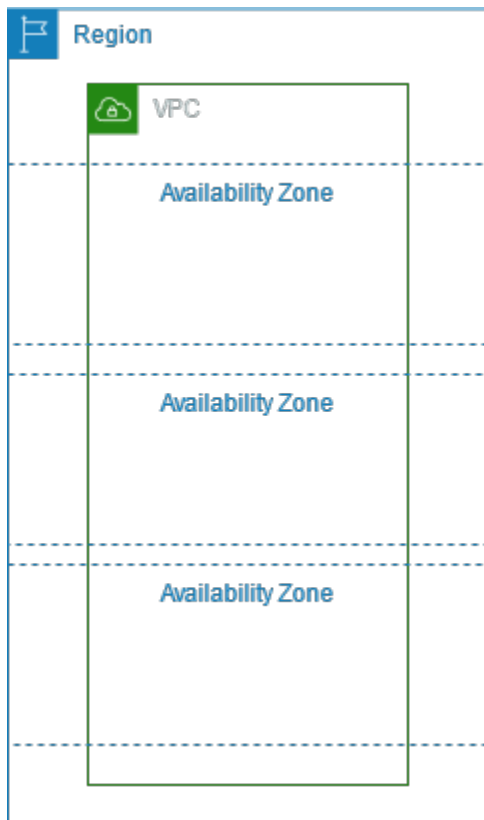
VPC를 생성할 때 다음과 같이 IP 주소를 지정합니다.

- IPv4 전용 - VPC에 IPv4 CIDR 블록은 있지만 IPv6 CIDR 블록은 없습니다.
- 듀얼 스택 - VPC에 IPv4 CIDR 블록 및 IPv6 CIDR 블록이 둘 다 있습니다.

자세한 내용은 [VPC 및 서브넷의 IP 주소 지정](#) 단원을 참조하십시오.

VPC 다이어그램

다음 다이어그램은 추가 VPC 리소스가 없는 VPC를 보여 줍니다. VPC 구성에 대한 예시는 [예제](#) 섹션을 참조하십시오.



VPC 리소스

각 VPC는 다음 리소스와 함께 제공됩니다.

- [기본 DHCP 옵션 세트](#)
- [기본 네트워크 ACL](#)
- [기본 보안 그룹](#)
- [기본 라우팅 테이블](#)

VPC에 대해 다음 리소스를 생성할 수 있습니다.

- [네트워크 ACL](#)
- [사용자 지정 라우팅 테이블](#)
- [보안 그룹](#)
- [인터넷 게이트웨이](#)
- [NAT 게이트웨이](#)

기본 VPC

Amazon VPC를 사용하기 시작하는 경우 각 AWS 리전에 기본 VPC가 있습니다. 기본 VPC는 각 가용 영역의 퍼블릭 서브넷, 인터넷 게이트웨이 및 DNS 확인 활성화 설정과 함께 제공됩니다. 따라서 기본 VPC로 Amazon EC2 인스턴스를 즉시 시작할 수 있습니다. 기본 VPC에서 Elastic Load Balancing, Amazon RDS, Amazon EMR 같은 서비스를 사용할 수도 있습니다.

기본 VPC는 준비 과정 없이 빠르게 시작하여 블로그나 간단한 웹 사이트 같은 퍼블릭 인스턴스를 시작하는 데 적합합니다. 기본 VPC의 구성 요소를 필요에 따라 수정할 수 있습니다.

기본 VPC에 서브넷을 추가할 수 있습니다. 자세한 내용은 [the section called “서브넷 생성”](#) 단원을 참조하십시오.

목차

- [기본 VPC 구성 요소](#)
- [기본 서브넷](#)
- [기본 VPC와 기본 서브넷 보기](#)
- [기본 VPC 생성](#)
- [기본 서브넷 만들기](#)
- [기본 서브넷과 기본 VPC 삭제](#)

기본 VPC 구성 요소

기본 VPC는 다음과 같이 생성됩니다.

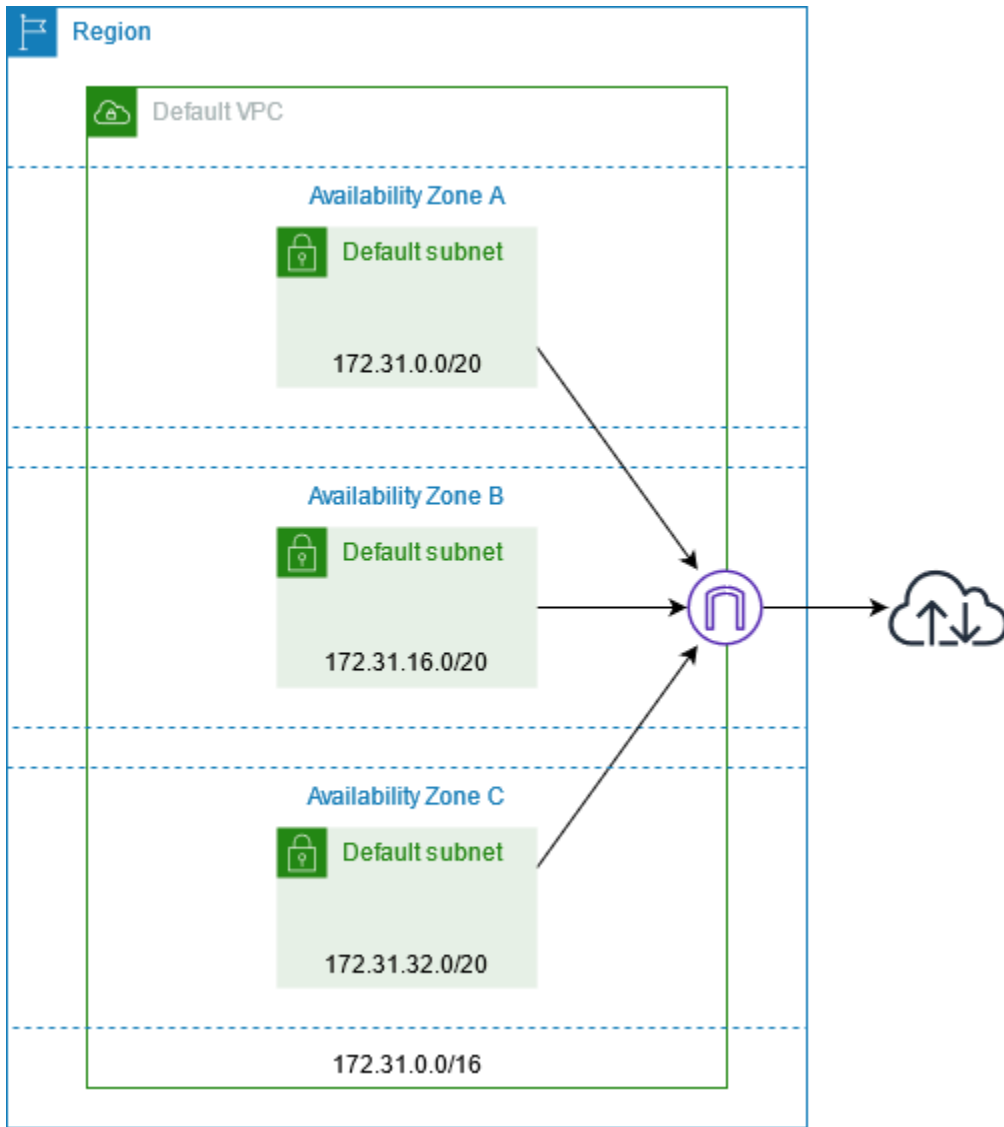
- IPv4 CIDR 블록의 크기가 /16인 VPC를 만듭니다 (172.31.0.0/16). 이는 최대 65,536개의 프라이빗 IPv4 주소를 제공합니다.

- 각 가용 영역에 크기 /20의 기본 서브넷을 생성합니다. 이렇게 하면 서브넷당 최대 4,096개의 주소가 제공되며, 그중 몇 개는 내부용으로 예약되어 있습니다.
- [인터넷 게이트웨이](#)를 만들어 기본 VPC에 연결합니다.
- 기본 라우팅 테이블에 모든 트래픽(0.0.0.0/0)이 인터넷 게이트웨이로 전달되는 경로를 추가합니다.
- 기본 보안 그룹을 만들어 기본 VPC와 연결합니다.
- 네트워크 ACL(액세스 제어 목록)을 생성하여 기본 VPC와 연결합니다.
- AWS 계정에서 설정된 기본 DHCP 옵션을 기본 VPC와 연결합니다.

Note

위의 리소스는 사용자 대신 Amazon에서 생성합니다. 사용자가 이러한 작업을 수행하는 것이 아니므로 여기에는 IAM 정책이 적용되지 않습니다. 예를 들어 CreateInternetGateway를 호출하는 기능을 거부하는 IAM 정책이 있고 CreateDefaultVpc를 호출하면 기본 VPC의 인터넷 게이트웨이가 여전히 생성됩니다.

다음 그림은 기본 VPC에 대해 설정되는 핵심 구성 요소를 보여 줍니다.



다음 표는 기본 VPC에 대한 기본 라우팅 테이블의 경로를 보여줍니다.

대상 주소	대상
172.31.0.0/16	로컬
0.0.0.0/0	<i>internet_gateway_id</i>

기본 VPC는 다른 일반 VPC와 동일한 방식으로 사용할 수 있습니다.

- 기본 서브넷이 아닌 서브넷을 추가합니다.
- 기본 라우팅 테이블을 수정합니다.

- 라우팅 테이블을 추가합니다.
- 추가 보안 그룹을 연결합니다.
- 기본 보안 그룹의 규칙을 업데이트합니다.
- AWS Site-to-Site VPN 연결을 추가합니다.
- 더 많은 IPv4 CIDR 블록을 추가합니다.
- Direct Connect 게이트웨이를 사용하여 원격 리전의 VPC에 액세스합니다. Direct Connect 게이트웨이 옵션에 대한 자세한 내용은 AWS Direct Connect 사용 설명서의 [Direct Connect 게이트웨이](#)를 참조하세요.

기본 서브넷도 다른 서브넷을 사용하듯이 사용할 수 있습니다. 즉, 사용자 지정 라우팅 테이블을 추가하고 네트워크 ACL을 설정할 수 있습니다. EC2 인스턴스를 시작할 때 특정 기본 서브넷을 지정할 수도 있습니다.

IPv6 CIDR 블록을 기본 VPC에 연결할 수도 있습니다.

기본 서브넷

기본 라우팅 테이블은 인터넷으로 대상 주소가 정해진 서브넷의 트래픽을 인터넷 게이트웨이로 전송하기 때문에 기본적으로 기본 서브넷은 퍼블릭 서브넷입니다. 대상 주소 0.0.0.0/0에서 인터넷 게이트웨이로 라우팅을 제거함으로써 기본 서브넷을 프라이빗 서브넷으로 만들 수 있습니다. 하지만 이렇게 하면 해당 서브넷에서 실행하는 EC2 인스턴스는 인터넷에 액세스할 수 없습니다.

기본 서브넷에서 시작한 인스턴스는 퍼블릭 IPv4 주소와 프라이빗 IPv4 주소, 퍼블릭 DNS 호스트 이름과 프라이빗 DNS 호스트 이름을 둘 다 받습니다. 기본 VPC의 기본이 아닌 서브넷에서 시작하는 인스턴스는 퍼블릭 IPv4 주소나 DNS 호스트 이름을 수신하지 않습니다. 서브넷의 퍼블릭 IP 주소 지정 동작은 변경할 수 있습니다. 자세한 내용은 [서브넷의 퍼블릭 IPv4 주소 지정 속성 수정](#) 단원을 참조하십시오.

때때로 AWS는 새로운 가용 영역을 리전에 추가할 수 있습니다. 대부분의 경우 며칠 내로 기본 VPC에 대한 이 가용 영역에 새로운 기본 서브넷이 자동으로 생성됩니다. 하지만 기본 VPC를 수정했을 경우에는 새로운 기본 서브넷이 추가되지 않습니다. 원한다면, 새로운 가용 영역에 대한 기본 서브넷을 직접 생성할 수 있습니다. 자세한 내용은 [기본 서브넷 만들기](#) 단원을 참조하십시오.

기본 VPC와 기본 서브넷 보기

Amazon VPC 콘솔이나 명령줄을 사용하여 기본 VPC와 서브넷을 볼 수 있습니다.

콘솔을 사용하여 기본 VPC와 서브넷을 보려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Your VPCs]를 선택합니다.
3. [Default VPC] 열에서 [Yes] 값을 확인합니다. 기본 VPC의 ID를 메모합니다.
4. 탐색 창에서 서브넷을 선택합니다.
5. 검색줄에 기본 VPC의 ID를 입력합니다. 검색 결과에 표시되는 서브넷이 기본 VPC의 서브넷입니다.
6. 어느 서브넷이 기본 서브넷인지 확인하려면 [Default Subnet] 열에서 [Yes] 값을 확인합니다.

명령줄을 사용하여 기본 VPC를 나타내려면

- [describe-vpcs](#)(AWS CLI)를 사용합니다.
- [Get-EC2Vpc](#)(AWS Tools for Windows PowerShell)를 사용합니다.

isDefault 필터를 포함하여 명령을 사용하고 필터 값을 true로 설정합니다.

명령줄을 사용하여 기본 서브넷을 나타내려면

- [describe-subnets](#)(AWS CLI)를 사용합니다.
- [Get-EC2Subnet](#)(AWS Tools for Windows PowerShell)를 사용합니다.

vpc-id 필터를 포함하여 명령을 사용하고 필터 값을 기본 VPC의 ID로 설정합니다. 기본 서브넷은 출력에서 DefaultForAz 필드가 true로 설정되어 있습니다.

기본 VPC 생성

기본 VPC를 삭제한 경우 새로 만들 수 있습니다. 기본 VPC를 삭제하면 복구할 수 없으며, 기본 VPC가 아닌 기존 VPC를 기본 VPC로 설정할 수도 없습니다.

기본 VPC를 만들면 각 가용 영역의 기본 서브넷을 비롯하여 기본 VPC의 표준 [구성 요소](#)와 함께 생성됩니다. 구성 요소를 직접 지정할 수 없습니다. 새로운 기본 VPC의 서브넷 CIDR 블록은 기존 기본 VPC와 동일한 가용 영역에 매핑되지 않을 수 있습니다. 예를 들어 기존 기본 VPC에서 CIDR 블록 172.31.0.0/20을 포함하는 서브넷이 us-east-2a에 생성된 경우, 새로운 기본 VPC에서는 us-east-2b에 생성될 수 있습니다.

리전에 기본 VPC가 이미 있으면 다른 기본 VPC를 만들 수 없습니다.

콘솔을 사용하여 기본 VPC를 만들려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Your VPCs]를 선택합니다.
3. [Actions], [Create Default VPC]를 선택합니다.
4. Create를 선택합니다. 확인 화면을 닫습니다.

명령줄을 사용하여 기본 VPC를 만들려면

[create-default-vpc](#) AWS CLI 명령을 사용할 수 있습니다. 이 명령에는 입력 파라미터가 없습니다.

```
aws ec2 create-default-vpc
```

다음은 예제 출력입니다.

```
{
  "Vpc": {
    "VpcId": "vpc-3f139646",
    "InstanceTenancy": "default",
    "Tags": [],
    "Ipv6CidrBlockAssociationSet": [],
    "State": "pending",
    "DhcpOptionsId": "dopt-61079b07",
    "CidrBlock": "172.31.0.0/16",
    "IsDefault": true
  }
}
```

또는 Windows PowerShell용 [New-EC2DefaultVpc](#) 도구나 [CreateDefaultVpc](#) Amazon EC2 API 작업을 사용해도 됩니다.

기본 서브넷 만들기

기본 서브넷이 없는 가용 영역에서 기본 서브넷을 생성할 수 있습니다. 예를 들어 기본 서브넷을 삭제한 경우 이를 생성하고자 할 수 있습니다. 또는 AWS가 새 가용 영역을 추가했지만 기본 VPC에서 해당 영역에 대한 기본 서브넷을 자동적으로 생성하지 않은 경우가 있습니다.

기본 서브넷을 생성할 때, 기본 VPC의 다음 사용 가능한 연속 공간에 IPv4 CIDR 블록의 크기가 /20인 서브넷이 생성됩니다. 다음 규칙이 적용됩니다.

- CIDR 블록을 직접 지정할 수 없습니다.
- 삭제한 이전 기본 서브넷을 복원할 수 없습니다.
- 가용 영역당 기본 서브넷은 한 개만 가질 수 있습니다.
- 기본 VPC가 아닌 VPC에는 기본 서브넷을 생성할 수 없습니다.

CIDR 블록 크기 /20을 생성할 충분한 주소 공간이 기본 VPC에 없는 경우 요청은 실패합니다. 더 많은 주소 공간이 필요한 경우 [VPC에 IPv4 CIDR 블록을 추가](#)할 수 있습니다.

기본 VPC에 IPv6 CIDR 블록을 연결한 경우 새로운 기본 서브넷이 자동으로 IPv6 CIDR 블록을 수신하지 않습니다. 대신 기본 서브넷을 생성한 다음 이에 IPv6 CIDR 블록을 연결할 수 있습니다. 자세한 내용은 [서브넷에 IPv6 CIDR 블록 추가](#) 단원을 참조하십시오.

AWS Management Console을 사용하여 기본 서브넷을 생성할 수 없습니다.

AWS CLI를 사용하여 기본 서브넷을 생성하려면

[create-default-subnet](#) AWS CLI 명령을 사용하고 서브넷을 생성할 가용 영역을 지정합니다.

```
aws ec2 create-default-subnet --availability-zone us-east-2a
```

다음은 예제 출력입니다.

```
{
  "Subnet": {
    "AvailabilityZone": "us-east-2a",
    "Tags": [],
    "AvailableIpAddressCount": 4091,
    "DefaultForAz": true,
    "Ipv6CidrBlockAssociationSet": [],
    "VpcId": "vpc-1a2b3c4d",
    "State": "available",
    "MapPublicIpOnLaunch": true,
    "SubnetId": "subnet-1122aabb",
    "CidrBlock": "172.31.32.0/20",
    "AssignIpv6AddressOnCreation": false
  }
}
```

AWS CLI 설정에 대한 자세한 내용은 [AWS Command Line Interface 사용 설명서](#)를 참조하세요.

또는 Windows PowerShell용 [New-EC2DefaultSubnet](#) 도구 명령이나 [CreateDefaultSubnet](#) Amazon EC2 API 작업을 사용할 수도 있습니다.

기본 서브넷과 기본 VPC 삭제

기본 서브넷이나 기본 VPC는 다른 서브넷 또는 VPC처럼 삭제할 수 있습니다. 그러나 기본 서브넷이나 기본 VPC를 삭제하면 인스턴스를 시작할 때 VPC 중 하나에 서브넷을 명시적으로 지정해야 합니다. 다른 VPC가 없으면 하나 이상의 가용 영역에 서브넷이 있는 VPC를 생성해야 합니다. 자세한 내용은 [VPC 생성](#) 단원을 참조하십시오.

기본 VPC를 삭제한 경우 새로 만들 수 있습니다. 자세한 내용은 [기본 VPC 생성](#) 단원을 참조하십시오.

기본 서브넷을 삭제한 경우 새로 만들 수 있습니다. 자세한 내용은 [기본 서브넷 만들기](#) 단원을 참조하십시오. 새로운 기본 서브넷이 정상적으로 동작하는지 확인하려면 서브넷 속성을 수정하여 해당 서브넷에서 시작되는 인스턴스에 퍼블릭 IP 주소를 할당하십시오. 자세한 내용은 [서브넷의 퍼블릭 IPv4 주소 지정 속성 수정](#) 단원을 참조하십시오. 가용 영역당 기본 서브넷은 한 개만 가질 수 있습니다. 기본 VPC가 아닌 VPC에는 기본 서브넷을 생성할 수 없습니다.

VPC 생성

다음 절차에 따라 Virtual Private Cloud(VPC)를 생성합니다. VPC에서 AWS 리소스를 생성하려면 먼저 VPC에 서브넷, 라우팅 테이블, 게이트웨이와 같은 추가 리소스가 있어야 합니다.

내용

- [VPC 구성 옵션](#)
- [VPC 및 기타 VPC 리소스 생성](#)
- [VPC만 생성](#)
- [AWS CLI를 사용하여 VPC 생성](#)

VPC 보기 또는 수정에 대한 자세한 내용은 [the section called “VPC 구성”](#) 섹션을 참조하세요.

VPC 구성 옵션

VPC를 생성할 때 다음 구성 옵션을 지정할 수 있습니다.

가용 영역

AWS 리전에 중복 전원, 네트워킹 및 연결이 있는 하나 이상의 개별 데이터 센터입니다. 여러 AZ를 사용하여 단일 데이터 센터에서 가능한 것보다 더 높은 가용성, 내결함성 및 확장성을 갖춘 프로덕

선 애플리케이션 및 데이터베이스를 운영할 수 있습니다. 여러 AZ의 서브넷에서 실행되는 애플리케이션을 분할하면 정전, 낙뢰, 토네이도, 지진과 같은 문제로부터 더 잘 격리되고 보호됩니다.

CIDR 블록

VPC와 서브넷의 IP 주소 범위를 지정해야 합니다. 자세한 내용은 [VPC 및 서브넷의 IP 주소 지정](#) 단원을 참조하십시오.

DNS 옵션

서브넷에서 시작된 EC2 인스턴스의 퍼블릭 IPv4 DNS 호스트 이름이 필요한 경우 두 DNS 옵션을 모두 활성화해야 합니다. 자세한 내용은 [VPC의 DNS 속성](#) 단원을 참조하십시오.

- DNS 호스트 이름 활성화: VPC에서 시작되는 EC2 인스턴스는 퍼블릭 IPv4 주소에 해당하는 퍼블릭 DNS 호스트 이름을 받습니다.
- DNS 확인 활성화: 프라이빗 DNS 호스트 이름에 대한 DNS 확인은 Route53 Resolver라고 하는 Amazon DNS 서버에서 VPC에 제공합니다.

인터넷 게이트웨이

인터넷에 VPC를 연결합니다. 퍼블릭 서브넷의 인스턴스는 서브넷 라우팅 테이블에 인터넷으로 향하는 트래픽을 인터넷 게이트웨이로 전송하는 경로가 포함되어 있으므로 인터넷에 액세스할 수 있습니다. 인터넷에서 서버에 직접 연결할 필요가 없는 경우 퍼블릭 서브넷에 배포하면 안 됩니다. 자세한 내용은 [인터넷 게이트웨이](#)를 참조하세요.

명칭

VPC 및 기타 VPC 리소스에 지정한 이름은 이름 태그를 생성하는 데 사용됩니다. 콘솔에서 이름 태그 자동 생성 기능을 사용하면 태그 값은 `name - resource` 형식을 갖습니다.

NAT 게이트웨이

프라이빗 서브넷의 인스턴스가 아웃바운드 트래픽을 전송할 수 있도록 하지만 인터넷의 리소스가 인스턴스에 연결하는 것을 방지합니다. 프로덕션 환경에서는 각 활성 AZ에 NAT 게이트웨이를 배포하는 것이 좋습니다. 자세한 내용은 [NAT 게이트웨이](#) 단원을 참조하세요.

라우팅 테이블

서브넷 또는 게이트웨이의 네트워크 트래픽이 전송되는 위치를 결정하는 라우팅이라는 규칙 집합을 포함합니다. 자세한 내용은 [라우팅 테이블](#)을 참조하세요.

서브넷

VPC의 IP 주소 범위입니다. EC2 인스턴스와 같은 AWS 리소스를 서브넷에서 실행할 수 있습니다. 각 서브넷은 단일 가용 영역 내에서만 존재합니다. 두 개 이상의 가용 영역에서 인스턴스를 시작하면 단일 가용 영역의 장애로부터 애플리케이션을 보호할 수 있습니다.

퍼블릭 서브넷에는 인터넷 게이트웨이로 직접 연결되는 경로가 있습니다. 퍼블릭 서브넷의 리소스는 퍼블릭 인터넷에 액세스할 수 있습니다. 프라이빗 서브넷에는 인터넷 게이트웨이로 직접 연결되는 경로가 없습니다. 프라이빗 서브넷의 리소스에는 퍼블릭 인터넷에 액세스하기 위해 NAT 장치와 같은 다른 구성 요소가 필요합니다.

자세한 내용은 [서브넷](#)을 참조하세요.

Tenancy

이 옵션은 VPC로 시작하는 EC2 인스턴스가 다른 AWS 계정과 공유되는 하드웨어에서 실행되는지 아니면 사용자 전용 하드웨어에서 실행되는지를 정의합니다. VPC의 테넌시를 Default로 선택하면 이 VPC로 시작된 EC2 인스턴스는 인스턴스를 시작할 때 지정된 테넌시 속성을 사용합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [정의된 파라미터를 사용하여 인스턴스 시작](#)을 참조하세요. VPC의 테넌시를 Dedicated로 선택하면 인스턴스는 항상 전용 하드웨어에서 [전용 인스턴스](#)로 실행됩니다. AWS Outposts를 사용하는 경우 Outpost에 프라이빗 연결이 필요합니다. Default 테넌시를 사용해야 합니다.

VPC 및 기타 VPC 리소스 생성

다음 절차에 따라 VPC 및 애플리케이션을 실행하는 데 필요한 추가 VPC 리소스(예: 서브넷, 라우팅 테이블, 인터넷 게이트웨이, NAT 게이트웨이)를 생성합니다. VPC 구성에 대한 예시는 [예제](#) 섹션을 참조하세요.

콘솔을 사용하여 VPC, 서브넷 및 기타 VPC 리소스를 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. VPC 대시보드에서 VPC 생성을 선택합니다.
3. 생성할 리소스에서 VPC 등을 선택합니다.
4. 이름 태그 자동 생성을 선택한 상태로 유지하여 VPC 리소스에 이름 태그를 생성하거나 선택을 취소하여 VPC 리소스에 고유한 이름 태그를 제공합니다.
5. IPv4 CIDR 블록에 VPC의 IPv4 주소 범위를 입력합니다. VPC는 IPv4 주소 범위를 가져야 합니다.
6. (선택 사항) IPv6 트래픽을 지원하려면 IPv6 CIDR 블록, Amazon에서 제공한 IPv6 CIDR 블록을 선택합니다.
7. 테넌시 옵션을 선택합니다. 이 옵션은 VPC로 시작하는 EC2 인스턴스가 다른 AWS 계정과 공유되는 하드웨어에서 실행되는지 아니면 사용자 전용 하드웨어에서 실행되는지를 정의합니다. VPC의 테넌시를 Default로 선택하면 이 VPC로 시작된 EC2 인스턴스에서는 인스턴스를 시작할 때 지정된 테넌시 속성을 사용합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [정의된 파라미터](#)

를 사용하여 [인스턴스 시작](#)을 참조하세요. VPC의 테넌시를 Dedicated로 선택하면 인스턴스는 항상 전용 하드웨어에서 [전용 인스턴스](#)로 실행됩니다. AWS Outposts를 사용하는 경우 Outpost에 프라이빗 연결이 필요합니다. Default 테넌시를 사용해야 합니다.

8. 가용 영역(AZ) 수에서 프로덕션 환경의 경우 2개 이상의 가용 영역에 서브넷을 프로비저닝하는 것이 좋습니다. 서브넷의 AZ를 선택하려면 AZ 사용자 지정을 확장합니다. 그렇지 않으면 AWS가 선택하도록 합니다.
9. 서브넷을 구성하려면 퍼블릭 서브넷 수 및 프라이빗 서브넷 수의 값을 선택합니다. 서브넷의 IP 주소 범위를 선택하려면 서브넷 CIDR 블록 사용자 지정을 확장합니다. 그렇지 않으면 AWS가 선택하도록 합니다.
10. (선택 사항) 프라이빗 서브넷의 리소스가 IPv4를 통해 퍼블릭 인터넷에 액세스해야 하는 경우 NAT 게이트웨이에서 NAT 게이트웨이를 생성할 AZ 수를 선택합니다. 프로덕션 환경에서는 퍼블릭 인터넷에 액세스해야 하는 리소스가 있는 각 AZ에 NAT 게이트웨이를 배포하는 것이 좋습니다. NAT 게이트웨이와 관련된 비용이 있습니다. 자세한 내용은 [요금](#) 단원을 참조하십시오.
11. (선택 사항) 프라이빗 서브넷의 리소스가 IPv6을 통해 퍼블릭 인터넷에 액세스해야 하는 경우 송신 전용 인터넷 게이트웨이에서 예를 선택합니다.
12. (선택 사항) VPC에서 직접 Amazon S3에 액세스해야 하는 경우 VPC 엔드포인트, S3 게이트웨이를 선택합니다. 이는 Amazon S3 게이트웨이 VPC 엔드포인트를 생성합니다. 자세한 내용은 AWS PrivateLink 사용 설명서의 [게이트웨이 VPC 엔드포인트](#)를 참조하세요.
13. (선택 사항) DNS 옵션의 경우 도메인 이름 확인을 위한 두 가지 옵션이 기본적으로 모두 활성화됩니다. 기본값이 요구 사항을 충족하지 않는 경우 해당 옵션을 비활성화할 수 있습니다.
14. (선택 사항) VPC에 태그를 추가하려면 추가 태그를 확장하고 새 태그 추가를 선택하여 태그 키와 태그 값을 입력합니다.
15. 미리 보기 창에서 구성한 VPC 리소스 간의 관계를 시각화할 수 있습니다. 실선은 리소스 간의 관계를 나타냅니다. 점선은 NAT 게이트웨이, 인터넷 게이트웨이 및 게이트웨이 엔드포인트에 대한 네트워크 트래픽을 나타냅니다. VPC를 생성한 후에는 리소스 맵 탭을 사용하여 언제든지 VPC의 리소스를 이 형식으로 시각화할 수 있습니다. 자세한 내용은 [VPC의 리소스 시각화](#) 단원을 참조하십시오.
16. VPC 구성을 마치면 VPC 생성을 선택합니다.

VPC만 생성

다음 절차에 따라 Amazon VPC 콘솔을 사용하여 추가 VPC 리소스 없이 VPC를 생성합니다.

콘솔을 사용하여 추가 VPC 리소스 없이 VPC를 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. VPC 대시보드에서 VPC 생성을 선택합니다.
3. 생성할 리소스에서 VPC 전용을 선택합니다.
4. (선택 사항) 이름 태그에 VPC의 이름을 입력합니다. 이렇게 하면 Name 키와 지정한 값으로 태그가 생성됩니다.
5. IPv4 CIDR 블록의 경우 다음 중 하나를 수행합니다.
 - IPv4 CIDR 수동 입력 을 선택하고 VPC에 대한 IPv4 주소 범위를 입력합니다.
 - IPAM에서 할당된 IPv4 CIDR 블록을 선택하고, Amazon VPC IP 주소 관리자(IPAM) IPv4 주소 풀과 넷마스크를 선택합니다. CIDR 블록의 크기는 IPAM 풀의 할당 규칙에 의해 제한됩니다. IPAM은 AWS 워크로드의 IP 주소를 보다 쉽게 계획, 추적 및 모니터링할 수 있게 해주는 VPC 기능입니다. 자세한 내용은 [Amazon VPC IPAM 사용 설명서](#)를 참조하세요.

IPAM을 사용하여 IP 주소를 관리하는 경우 이 옵션을 선택하는 것이 좋습니다. 그렇지 않으면 VPC에 지정한 CIDR 블록이 IPAM CIDR 할당과 겹칠 수 있습니다.
6. (선택 사항) 듀얼 스택 VPC를 생성하려면 VPC에 IPv6 주소 범위를 지정합니다. IPv6 CIDR 블록의 경우 다음 중 하나를 수행합니다.
 - Amazon VPC IP 주소 관리자를 사용 중이고 IPAM 풀에서 IPv6 CIDR을 프로비저닝하려는 경우 IPAM 할당 IPv6 CIDR 블록을 선택합니다. CIDR 블록에서 VPC에 IP 주소 범위를 프로비저닝하는 데는 두 가지 옵션이 있습니다.
 - 넷마스크 길이: CIDR의 넷마스크 길이를 선택하려면 이 옵션을 선택합니다. 다음 중 하나를 수행하십시오.
 - IPAM 풀에 대해 선택된 기본 넷마스크 길이가 있는 경우 IPAM 넷마스크 길이 기본값을 선택하여 IPAM 관리자가 IPAM 풀에 설정한 기본 넷마스크 길이를 사용할 수 있습니다. 선택적 기본 넷마스크 길이 할당 규칙에 대한 자세한 내용은 Amazon VPC IPAM 사용 설명서의 [리전 IPv6 풀 생성](#)을 참조하세요.
 - IPAM 풀에 대해 선택된 기본 넷마스크 길이가 없는 경우 IPAM 풀 CIDR의 넷마스크 길이보다 더 구체적인 넷마스크 길이를 선택합니다. 예를 들어 IPAM 풀 CIDR이 /50인 경우 VPC의 넷마스크 길이를 /52에서 /60 사이로 선택할 수 있습니다. 가능한 넷마스크 길이는 /44~/60(/4씩 증가)입니다.
 - CIDR 선택: IPv6 주소를 수동으로 입력하려면 이 옵션을 선택합니다. IPAM 풀 CIDR 넷마스크 길이보다 더 구체적인 넷마스크 길이만 선택할 수 있습니다. 예를 들어 IPAM 풀 CIDR

이 /50인 경우 VPC의 넷마스크 길이를 /52에서 /60 사이로 선택할 수 있습니다. 가능한 IPv6 넷마스크 길이는 /44~/60(/4씩 증가)입니다.

- Amazon의 IPv6 주소 풀에서 IPv6 CIDR 블록을 요청하려면 Amazon 제공 IPv6 CIDR 블록을 선택합니다. 네트워크 경계 그룹(Network Border Group)에서 AWS가 IP 주소를 알리는 그룹을 선택합니다. Amazon은 /56의 고정 IPv6 CIDR 블록 크기를 제공합니다.
 - 이미 AWS에 가져온 IPv6 CIDR을 프로비저닝하려면 내가 소유한 IPv6 CIDR 을 선택합니다. 고유 IP 주소 범위를 AWS로 가져오는 방법에 대한 자세한 내용은 Amazon EC2 사용 설명서의 [고유 IP 주소 가져오기\(BYOIP\)](#)를 참조하세요. CIDR 블록에 대한 다음과 같은 옵션을 사용하여 VPC의 IP 주소 범위를 프로비저닝할 수 있습니다.
 - 기본 설정 없음: /56의 넷마스크 길이를 사용하려면 이 옵션을 선택합니다.
 - CIDR 선택: IPv6 주소를 수동으로 입력하고 BYOIP CIDR 크기보다 더 구체적인 넷마스크 길이를 선택하려면 이 옵션을 선택합니다. 예를 들어 BYOIP 풀 CIDR이 /50인 경우 VPC의 넷마스크 길이를 /52에서 /60 사이로 선택할 수 있습니다. 가능한 IPv6 넷마스크 길이는 /44~/60(/4씩 증가)입니다.
7. (선택 사항) 테넌시 옵션을 선택합니다. 이 옵션은 VPC로 시작하는 EC2 인스턴스가 다른 AWS 계정과 공유되는 하드웨어에서 실행되는지 아니면 사용자 전용 하드웨어에서 실행되는지를 정의합니다. VPC의 테넌시를 Default로 선택하면 이 VPC로 시작된 EC2 인스턴스는 인스턴스를 시작할 때 지정된 테넌시 속성을 사용합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [정의된 파라미터를 사용하여 인스턴스 시작](#)을 참조하세요. VPC의 테넌시를 Dedicated로 선택하면 인스턴스는 항상 전용 하드웨어에서 [전용 인스턴스](#)로 실행됩니다. AWS Outposts를 사용하는 경우 Outpost에 프라이빗 연결이 필요합니다. Default 테넌시를 사용해야 합니다.
 8. (선택 사항) VPC에 태그를 추가하려면 새 태그 추가를 선택하고 태그 키와 태그 값을 입력합니다.
 9. VPC 생성을 선택합니다.
 10. VPC를 생성한 후 서브넷을 추가할 수 있습니다. 자세한 내용은 [서브넷 생성](#) 단원을 참조하십시오.

AWS CLI를 사용하여 VPC 생성

다음 절차에는 VPC를 만들기 위한 예제 AWS CLI 명령과 애플리케이션을 실행하는 데 필요한 추가 VPC 리소스가 포함되어 있습니다. 이 절차의 모든 명령을 실행하면 VPC, 퍼블릭 서브넷, 프라이빗 서브넷, 각 서브넷의 라우팅 테이블, 인터넷 게이트웨이, 송신 전용 인터넷 게이트웨이 및 퍼블릭 NAT 게이트웨이가 생성됩니다. 이러한 리소스가 모두 필요하지 않은 경우 필요한 예제 명령만 사용할 수 있습니다.

필수 조건

시작하기 전에 AWS CLI을(를) 설치하고 구성합니다. AWS CLI를 구성하면 AWS 보안 인증 정보를 입력하라는 메시지가 표시됩니다. 이 절차의 예제에서는 기본 리전도 구성했다고 가정합니다. 그렇지 않을 경우 각 명령에 `--region` 옵션을 적용합니다. 자세한 내용은 [AWS CLI 설치 또는 업데이트 및 AWS CLI 구성](#)을 참조하세요.

태그 지정

`create-tags` 명령을 사용하여 리소스를 생성한 후 리소스에 태그를 추가할 수 있습니다. 또는 다음과 같이 리소스 생성 명령에 `--tag-specification` 옵션을 추가할 수 있습니다.

```
--tag-specifications ResourceType=vpc,Tags=[{Key=Name,Value=my-project}]
```

AWS CLI를 사용하여 VPC 및 VPC 리소스 생성

1. 다음 [create-vpc](#) 명령을 사용하여 지정된 IPv4 CIDR 블록으로 VPC를 생성합니다.

```
aws ec2 create-vpc --cidr-block 10.0.0.0/24 --query Vpc.VpcId --output text
```

또는 듀얼 스택 VPC를 생성하려면 다음 예제와 같이 Amazon에서 제공하는 IPv6 CIDR 블록을 추가하는 `--amazon-provided-ipv6-cidr-block` 옵션을 추가합니다.

```
aws ec2 create-vpc --cidr-block 10.0.0.0/24 --amazon-provided-ipv6-cidr-block --query Vpc.VpcId --output text
```

해당 명령은 새 VPC의 ID를 반환합니다. 다음은 예입니다.

```
vpc-1a2b3c4d5e6f1a2b3
```

2. [듀얼 스택 VPC] 다음 [describe-vpcs](#) 명령을 사용하여 VPC에 연결된 IPv6 CIDR 블록을 가져옵니다.

```
aws ec2 describe-vpcs --vpc-id vpc-1a2b3c4d5e6f1a2b3 --query Vpcs[].Ipv6CidrBlockAssociationSet[].Ipv6CidrBlock --output text
```

출력의 예제는 다음과 같습니다.

```
2600:1f13:cfe:3600::/56
```

3. 사용 사례에 따라 하나 이상의 서브넷을 생성합니다. 프로덕션 환경에서는 2개 이상의 가용 영역에서 리소스를 시작하는 것이 좋습니다. 다음 명령 중 하나를 사용하여 각 서브넷을 생성합니다.
- IPv4 전용 서브넷 - 특정 IPv4 CIDR 블록으로 서브넷을 생성하려면 다음 [create-subnet](#) 명령을 사용합니다.

```
aws ec2 create-subnet --vpc-id vpc-1a2b3c4d5e6f1a2b3 --cidr-block 10.0.1.0/20
--availability-zone us-east-2a --query Subnet.SubnetId --output text
```

- 듀얼 스택 서브넷 - 듀얼 스택 VPC를 생성한 경우 다음 명령과 같이 `--ipv6-cidr-block` 옵션을 사용하여 듀얼 스택 서브넷을 생성할 수 있습니다.

```
aws ec2 create-subnet --vpc-id vpc-1a2b3c4d5e6f1a2b3 --cidr-block 10.0.1.0/20
--ipv6-cidr-block 2600:1f13:cfe:3600::/64 --availability-zone us-east-2a --
query Subnet.SubnetId --output text
```

- IPv6 전용 서브넷 - 듀얼 스택 VPC를 생성한 경우 다음 명령과 같이 `--ipv6-native` 옵션을 사용하여 IPv6 전용 서브넷을 생성할 수 있습니다.

```
aws ec2 create-subnet --vpc-id vpc-1a2b3c4d5e6f1a2b3 --ipv6-native --ipv6-
cidr-block 2600:1f13:cfe:3600::/64 --availability-zone us-east-2a --query
Subnet.SubnetId --output text
```

해당 명령은 새 서브넷의 ID를 반환합니다. 다음은 예입니다.

```
subnet-1a2b3c4d5e6f1a2b3
```

4. 웹 서버용 또는 NAT 게이트웨이용 퍼블릭 서브넷이 필요한 경우 다음을 수행합니다.
- 다음 [create-internet-gateway](#) 명령을 사용하여 인터넷 게이트웨이를 생성합니다. 이 명령은 새 인터넷 게이트웨이의 ID를 반환합니다.

```
aws ec2 create-internet-gateway --query InternetGateway.InternetGatewayId --
output text
```

- 다음 [attach-internet-gateway](#) 명령을 사용하여 VPC에 인터넷 게이트웨이를 연결합니다. 이전 단계에서 반환한 인터넷 게이트웨이 ID를 사용합니다.

```
aws ec2 attach-internet-gateway --vpc-id vpc-1a2b3c4d5e6f1a2b3 --internet-
gateway-id igw-id
```


- c. 다음 [create-route-table](#) 명령을 사용하여 퍼블릭 서브넷에 대한 사용자 지정 라우팅 테이블을 생성합니다. 이 명령은 새 라우팅 테이블의 ID를 반환합니다.

```
aws ec2 create-route-table --vpc-id vpc-1a2b3c4d5e6f1a2b3 --query
RouteTable.RouteTableId --output text
```

- d. 다음 [create-route](#) 명령을 사용하여 모든 IPv4 트래픽을 인터넷 게이트웨이로 보내는 라우팅 테이블의 경로를 생성합니다. 퍼블릭 서브넷의 라우팅 테이블 ID를 사용합니다.

```
aws ec2 create-route --route-table-id rtb-id-public --destination-cidr-block
0.0.0.0/0 --gateway-id igw-id
```

- e. 다음 [associate-route-table](#) 명령을 사용하여 라우팅 테이블을 퍼블릭 서브넷과 연결합니다. 퍼블릭 서브넷의 라우팅 테이블 ID와 퍼블릭 서브넷의 ID를 사용합니다.

```
aws ec2 associate-route-table --route-table-id rtb-id-public --subnet-
id subnet-id-public-subnet
```

5. [IPv6] 프라이빗 서브넷의 인스턴스가 IPv6를 통해 인터넷에 액세스할 수 있도록 송신 전용 인터넷 게이트웨이를 추가할 수 있지만(예: 소프트웨어 업데이트 받기) 인터넷의 호스트는 인스턴스에 액세스할 수 없습니다.

- a. 다음 [create-egress-only-internet-gateway](#) 명령을 사용하여 송신 전용 인터넷 게이트웨이를 생성합니다. 이 명령은 새 인터넷 게이트웨이의 ID를 반환합니다.

```
aws ec2 create-egress-only-internet-gateway --vpc-id vpc-1a2b3c4d5e6f1a2b3 --
query EgressOnlyInternetGateway.EgressOnlyInternetGatewayId --output text
```

- b. 다음 [create-route-table](#) 명령을 사용하여 프라이빗 서브넷에 대한 사용자 지정 라우팅 테이블을 생성합니다. 이 명령은 새 라우팅 테이블의 ID를 반환합니다.

```
aws ec2 create-route-table --vpc-id vpc-1a2b3c4d5e6f1a2b3 --query
RouteTable.RouteTableId --output text
```

- c. 다음 [create-route](#) 명령을 사용하여 모든 IPv6 트래픽을 송신 전용 인터넷 게이트웨이로 보내는 프라이빗 서브넷에 대한 라우팅 테이블의 경로를 생성합니다. 이전 단계에서 반환한 라우팅 테이블의 ID를 사용합니다.

```
aws ec2 create-route --route-table-id rtb-id-private --destination-cidr-
block ::/0 --egress-only-internet-gateway eigw-id
```

- d. 다음 [associate-route-table](#) 명령을 사용하여 라우팅 테이블을 프라이빗 서브넷과 연결합니다.

```
aws ec2 associate-route-table --route-table-id rtb-id-private --subnet-id subnet-id-private-subnet
```

6. 프라이빗 서브넷의 리소스에 대한 NAT 게이트웨이가 필요한 경우 다음을 수행합니다.

- a. 다음 [allocate-address](#) 명령을 사용하여 NAT 게이트웨이의 탄력적 IP 주소를 생성합니다.

```
aws ec2 allocate-address --domain vpc --query AllocationId --output text
```

- b. 다음 [create-nat-gateway](#) 명령을 사용하여 퍼블릭 서브넷의 NAT 게이트웨이를 생성합니다. 이전 단계에서 반환되는 할당 ID를 사용합니다.

```
aws ec2 create-nat-gateway --subnet-id subnet-id-public-subnet --allocation-id eipalloc-id
```

- c. (선택 사항) 5단계에서 프라이빗 서브넷에 대한 라우팅 테이블을 이미 생성한 경우 이 단계를 건너뛵니다. 그렇지 않으면 다음 [create-route-table](#) 명령을 사용하여 프라이빗 서브넷에 대한 라우팅 테이블을 생성합니다. 이 명령은 새 라우팅 테이블의 ID를 반환합니다.

```
aws ec2 create-route-table --vpc-id vpc-1a2b3c4d5e6f1a2b3 --query RouteTable.RouteTableId --output text
```

- d. 다음 [create-route](#) 명령을 사용하여 모든 IPv4 트래픽을 NAT 게이트웨이로 보내는 프라이빗 서브넷에 대한 라우팅 테이블의 경로를 생성합니다. 이 단계 또는 5단계에서 생성한 프라이빗 서브넷의 라우팅 테이블 ID를 사용합니다.

```
aws ec2 create-route --route-table-id rtb-id-private --destination-cidr-block 0.0.0.0/0 --gateway-id nat-id
```

- e. (선택 사항) 5단계에서 프라이빗 서브넷과 라우팅 테이블을 이미 연결한 경우 이 단계를 건너뛵니다. 그렇지 않으면 다음 [associate-route-table](#) 명령을 사용하여 라우팅 테이블을 프라이빗 서브넷과 연결합니다. 이 단계 또는 5단계에서 생성한 프라이빗 서브넷의 라우팅 테이블 ID를 사용합니다.

```
aws ec2 associate-route-table --route-table-id rtb-id-private --subnet-id subnet-id-private-subnet
```

VPC 구성

다음 절차에 따라 Virtual Private Cloud(VPC)를 보고 구성합니다.

Tasks

- [VPC에 대한 세부 정보 보기](#)
- [VPC의 리소스 시각화](#)
- [VPC에 IPv4 CIDR 블록 추가](#)
- [VPC에 IPv6 CIDR 블록 추가](#)
- [VPC에서 IPv4 CIDR 블록 제거](#)
- [VPC에서 IPv6 CIDR 블록 제거](#)

VPC 생성 또는 삭제에 대한 자세한 내용은 [the section called “VPC 생성”](#) 또는 [the section called “VPC 삭제”](#) 섹션을 참조하세요.

VPC에 대한 세부 정보 보기

다음 단계에 따라 VPC에 대한 세부 정보를 봅니다.

콘솔을 사용하여 VPC 세부 정보를 보려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 VPCs를 선택합니다.
3. VPC를 선택한 다음 세부 정보 보기를 선택하여 VPC의 구성 세부 정보를 봅니다.

AWS CLI를 사용하여 VPC를 설명하려면

[describe-vpcs](#) 명령을 사용합니다.

모든 리전의 VPC를 보려면

<https://console.aws.amazon.com/ec2globalview/home>에서 Amazon EC2 Global View 콘솔을 엽니다. 자세한 내용은 Amazon EC2 사용 설명서의 [Amazon EC2 Global View를 사용하여 리소스 나열 및 필터링](#)을 참조하세요.

VPC의 리소스 시각화

다음 단계에 따라 리소스 맵 탭을 사용하여 VPC 내 리소스의 시각적 표현을 봅니다. 리소스 맵에는 다음 리소스가 표시됩니다.

- VPC
- 서브넷
 - 가용 영역은 문자로 표시됩니다.
 - 퍼블릭 서브넷은 녹색입니다.
 - 프라이빗 서브넷은 파란색입니다.
- 라우팅 테이블
- 인터넷 게이트웨이
- 외부 전용 인터넷 게이트웨이
- NAT 게이트웨이
- 게이트웨이 엔드포인트(Amazon S3 및 Amazon DynamoDB)

리소스 맵은 VPC 내 리소스 간의 관계와 서브넷에서 NAT 게이트웨이, 인터넷 게이트웨이 및 게이트웨이 엔드포인트로 트래픽이 흐르는 방식을 보여줍니다.

리소스 맵을 사용하여 VPC의 아키텍처를 이해하고 VPC에 있는 서브넷 수, 어떤 서브넷이 어떤 라우팅 테이블과 연결되어 있는지, 어떤 라우팅 테이블에 NAT 게이트웨이, 인터넷 게이트웨이 및 게이트웨이 엔드포인트에 대한 경로가 있는지 볼 수 있습니다.

또한 리소스 맵을 사용하여 NAT 게이트웨이에서 연결 해제된 프라이빗 서브넷 또는 인터넷 게이트웨이로 직접 연결되는 경로가 있는 프라이빗 서브넷과 같이 바람직하지 않거나 잘못된 구성을 찾아낼 수 있습니다. 리소스 맵 내에서 라우팅 테이블과 같은 리소스를 선택하고 해당 리소스의 구성을 편집할 수 있습니다.

VPC의 리소스를 시각화하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 VPCs를 선택합니다.
3. VPC를 선택합니다.
4. 리소스 맵 탭을 선택하여 리소스 시각화를 표시합니다.

5. 세부 정보 표시를 선택하여 기본적으로 표시되는 리소스 ID 및 영역 외의 세부 정보를 봅니다.
 - VPC: VPC에 할당된 IPv4 및 IPv6 CIDR 범위입니다.
 - 서브넷: 각 서브넷에 할당된 IPv4 및 IPv6 CIDR 범위입니다.
 - 라우팅 테이블: 라우팅 테이블의 서브넷 연결 및 경로 수입니다.
 - 네트워크 연결: 각 연결 유형과 관련된 세부 정보입니다.
 - VPC에 퍼블릭 서브넷이 있는 경우 인터넷 게이트웨이를 사용하는 트래픽의 소스 및 대상 서브넷과 경로 수가 포함된 인터넷 게이트웨이 리소스가 있습니다.
 - 외부 전용 인터넷 게이트웨이가 있는 경우 외부 전용 인터넷 게이트웨이를 사용하는 트래픽의 소스 및 대상 서브넷과 경로 수가 포함된 외부 전용 인터넷 게이트웨이 리소스가 있습니다.
 - NAT 게이트웨이가 있는 경우 NAT 게이트웨이의 탄력적 IP 주소와 네트워크 인터페이스 수가 포함된 NAT 게이트웨이 리소스가 있습니다.
 - 게이트웨이 엔드포인트가 있는 경우 엔드포인트를 사용하여 연결할 수 있는 AWS 서비스 (Amazon S3 또는 Amazon DynamoDB)의 이름이 포함된 게이트웨이 엔드포인트 리소스가 있습니다.
6. 리소스 위로 마우스를 가져가면 리소스 간의 관계를 볼 수 있습니다. 실선은 리소스 간의 관계를 나타냅니다. 점선은 네트워크 연결에 대한 네트워크 트래픽을 나타냅니다.

VPC에 IPv4 CIDR 블록 추가

기본적으로 VPC는 IPv4 CIDR 블록을 최대 5개까지 가질 수 있지만 이 제한을 조정할 수 있습니다. 자세한 내용은 [Amazon VPC 할당량](#) 단원을 참조하십시오. VPC의 IPv4 CIDR 블록에 대한 제한 사항에 대한 자세한 내용은 [VPC CIDR 블록](#)을 참조하세요.

콘솔을 사용하여 VPC에 IPv4 CIDR 블록을 추가하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 사용자 VPC(Your VPCs)를 선택합니다.
3. VPC를 선택한 다음 작업과 CIDR 편집을 차례로 선택합니다.
4. 새 IPv4 CIDR 추가를 선택합니다.
5. IPv4 CIDR 블록의 경우 다음 중 하나를 수행합니다.
 - IPv4 CIDR 수동 입력을 선택하고 IPv4 CIDR 블록을 입력합니다.
 - IPAM에 할당된 IPv4 CIDR을 선택하고 IPv4 IPAM 풀에서 CIDR을 선택합니다.
6. 저장을 선택한 다음 닫기를 선택합니다.

7. VPC에 IPv4 CIDR 블록을 추가한 후에는 새 CIDR 블록을 사용하는 서브넷을 생성할 수 있습니다. 자세한 내용은 [서브넷 생성](#) 단원을 참조하십시오.

AWS CLI를 사용하여 VPC에 IPv4 CIDR 블록을 연결하려면

[associate-vpc-cidr-block](#) 명령을 사용합니다.

VPC에 IPv6 CIDR 블록 추가

기본적으로 VPC는 IPv6 CIDR 블록을 최대 5개까지 가질 수 있지만 이 제한을 조정할 수 있습니다. 자세한 내용은 [Amazon VPC 할당량](#) 단원을 참조하십시오. VPC의 IPv6 CIDR 블록에 대한 제한 사항에 대한 자세한 내용은 [VPC CIDR 블록](#)을 참조하세요.

콘솔을 사용하여 VPC에 IPv6 CIDR 블록을 추가하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 사용자 VPC(Your VPCs)를 선택합니다.
3. VPC를 선택한 다음 작업과 CIDR 편집을 차례로 선택합니다.
4. 새 IPv6 CIDR 추가를 선택합니다.
5. IPv6 CIDR 블록의 경우 다음 중 하나를 수행합니다.
 - Amazon VPC IP 주소 관리자를 사용 중이고 IPAM 풀에서 IPv6 CIDR을 프로비저닝하려는 경우 IPAM 할당 IPv6 CIDR 블록을 선택합니다. CIDR 블록에서 VPC에 IP 주소 범위를 프로비저닝하는 데는 두 가지 옵션이 있습니다.
 - 넷마스크 길이: CIDR의 넷마스크 길이를 선택하려면 이 옵션을 선택합니다. 다음 중 하나를 수행하십시오.
 - IPAM 풀에 대해 선택된 기본 넷마스크 길이가 있는 경우 IPAM 넷마스크 길이 기본값을 선택하여 IPAM 관리자가 IPAM 풀에 설정한 기본 넷마스크 길이를 사용할 수 있습니다. 선택적 기본 넷마스크 길이 할당 규칙에 대한 자세한 내용은 Amazon VPC IPAM 사용 설명서의 [리전 IPv6 풀 생성](#)을 참조하세요.
 - IPAM 풀에 대해 선택된 기본 넷마스크 길이가 없는 경우 IPAM 풀 CIDR의 넷마스크 길이보다 더 구체적인 넷마스크 길이를 선택합니다. 예를 들어 IPAM 풀 CIDR이 /50인 경우 VPC의 넷마스크 길이를 /52에서 /60 사이로 선택할 수 있습니다. 가능한 넷마스크 길이는 /44~/60(/4씩 증가)입니다.
 - CIDR 선택: IPv6 주소를 수동으로 입력하려면 이 옵션을 선택합니다. IPAM 풀 CIDR 넷마스크 길이보다 더 구체적인 넷마스크 길이만 선택할 수 있습니다. 예를 들어 IPAM 풀 CIDR

이 /50인 경우 VPC의 넷마스크 길이를 /52에서 /60 사이로 선택할 수 있습니다. 가능한 IPv6 넷마스크 길이는 /44~/60(/4씩 증가)입니다.

- Amazon의 IPv6 주소 풀에서 IPv6 CIDR 블록을 요청하려면 Amazon 제공 IPv6 CIDR 블록을 선택합니다. 네트워크 경계 그룹(Network Border Group)에서 AWS가 IP 주소를 알리는 그룹을 선택합니다. Amazon은 /56의 고정 IPv6 CIDR 블록 크기를 제공합니다.
 - 이미 AWS에 가져온 IPv6 CIDR을 프로비저닝하려면 내가 소유한 IPv6 CIDR 을 선택합니다. 고유 IP 주소 범위를 AWS로 가져오는 방법에 대한 자세한 내용은 Amazon EC2 사용 설명서의 [Amazon EC2에서 고유 IP 주소 가져오기\(BYOIP\)](#)를 참조하세요. CIDR 블록에서 VPC에 IP 주소 범위를 프로비저닝하는 데는 두 가지 옵션이 있습니다.
 - 기본 설정 없음: /56의 넷마스크 길이를 사용하려면 이 옵션을 선택합니다.
 - CIDR 선택: IPv6 주소를 수동으로 입력하고 BYOIP CIDR 크기보다 더 구체적인 넷마스크 길이를 선택하려면 이 옵션을 선택합니다. 예를 들어 BYOIP 풀 CIDR이 /50인 경우 VPC의 넷마스크 길이를 /52에서 /60 사이로 선택할 수 있습니다. 가능한 IPv6 넷마스크 길이는 /44~/60(/4씩 증가)입니다.
6. CIDR 선택 을 선택한 다음 달기를 선택합니다.
7. VPC에 IPv6 CIDR 블록을 추가한 후에는 새 CIDR 블록을 사용하는 서브넷을 생성할 수 있습니다. 자세한 내용은 [서브넷 생성](#) 단원을 참조하십시오.

AWS CLI를 사용하여 VPC에 IPv6 CIDR 블록을 연결하려면

[associate-vpc-cidr-block](#) 명령을 사용합니다.

VPC에서 IPv4 CIDR 블록 제거

VPC에 하나 이상의 IPv4 CIDR 블록이 연결되어 있는 경우 VPC에서 IPv4 CIDR 블록을 제거할 수 있습니다. 기본 IPv4 CIDR 블록을 제거할 수 없습니다. 전체 CIDR 블록을 제거해야 합니다. CIDR 블록의 하위 집합이나 병합된 CIDR 블록 범위를 제거할 수 없습니다. 먼저 CIDR 블록에서 모든 서브넷을 삭제해야 합니다.

콘솔을 사용하여 VPC에서 CIDR 블록을 제거하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Your VPCs]를 선택합니다.
3. VPC를 선택하고 [Actions]와 [Edit CIDRs]를 차례로 선택합니다.
4. VPC IPv4 CIDR에서 제거를 선택하여 CIDR을 제거합니다.
5. 달기를 선택하세요.

AWS CLI를 사용하여 VPC에서 IPv4 CIDR 블록을 연결 해제하려면

[disassociate-vpc-cidr-block](#) 명령을 사용합니다.

VPC에서 IPv6 CIDR 블록 제거

VPC에서 IPv6 지원이 더 이상 필요 없지만 IPv4 리소스 생성 및 IPv4 리소스와의 통신을 위해 VPC를 계속 사용하려는 경우 IPv6 CIDR 블록을 제거할 수 있습니다.

IPv6 CIDR 블록을 제거하려면 먼저 서브넷의 모든 인스턴스에 할당된 모든 IPv6 주소를 할당 해제해야 합니다.

IPv6 CIDR 블록을 제거해도 IPv6 네트워킹을 위해 구성된 보안 그룹 규칙, 네트워크 ACL 규칙 또는 라우팅 테이블 경로는 자동으로 삭제되지 않습니다. 이 규칙 또는 경로는 수동으로 수정하거나 삭제해야 합니다.

콘솔을 사용하여 VPC에서 IPv6 CIDR 블록을 제거하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Your VPCs]를 선택합니다.
3. VPC를 선택하고 Actions, Edit CIDRs를 선택합니다.
4. IPv6 CIDR에서 제거를 선택하여 IPv6 CIDR을 제거합니다.
5. 닫기를 선택하세요.

AWS CLI를 사용하여 VPC에서 IPv6 CIDR 블록을 연결 해제하려면

[disassociate-vpc-cidr-block](#) 명령을 사용합니다.

Amazon VPC의 DHCP 옵션 세트

VPC의 네트워크 디바이스는 동적 호스트 구성 프로토콜(DHCP)을 사용합니다. DHCP 옵션 세트를 사용하여 가상 네트워크에서 네트워크 구성의 다음 측면을 제어할 수 있습니다.

- VPC의 디바이스에서 사용하는 DNS 서버, 도메인 이름 또는 NTP(Network Time Protocol) 서버입니다.
- VPC에서 DNS 확인이 활성화되었는지 여부.

내용

- [DHCP란 무엇인가요?](#)
- [DHCP 옵션 세트 개념](#)
- [DHCP 옵션 세트로 작업](#)

DHCP란 무엇인가요?

TCP/IP 네트워크의 모든 디바이스에는 네트워크를 통해 통신하기 위해 IP 주소가 필요합니다. 이전에는 네트워크의 각 디바이스에 IP 주소가 수동으로 할당되었습니다. 오늘날 IP 주소는 동적 호스트 구성 프로토콜(DHCP)을 사용하여 DHCP 서버에서 동적으로 할당됩니다.

EC2 인스턴스에서 실행되는 애플리케이션은 필요에 따라 Amazon DHCP 서버와 통신하여 IP 주소 임대 또는 기타 네트워크 구성 정보(예: Amazon DNS 서버의 IP 주소 또는 VPC의 라우터 IP 주소)를 검색할 수 있습니다.

DHCP 옵션 세트를 사용하여 Amazon DHCP 서버에서 제공하는 네트워크 구성을 지정할 수 있습니다.

애플리케이션이 Amazon IPv6 DHCP 서버에 직접 요청해야 하는 VPC 구성이 있는 경우 다음 사항에 유의하세요.

- 이중 스택 서브넷의 EC2 인스턴스는 IPv6 DHCP 서버에서만 해당 IPv6 주소를 검색할 수 있습니다. DNS 서버 이름 또는 도메인 이름과 같은 IPv6 DHCP 서버에서 추가 네트워크 구성을 검색할 수 없습니다.
- IPv6 전용 서브넷의 EC2 인스턴스는 IPv6 DHCP 서버에서 IPv6 주소를 검색하고 DNS 서버 이름, 도메인 이름 등의 추가 네트워킹 구성 정보를 검색할 수 있습니다.
- IPv6 전용 서브넷에 있는 EC2 인스턴스의 경우 DHCP 옵션 세트에 'AmazonProvidedDNS'가 명시적으로 언급된 경우 IPv4 DHCP 서버에서는 이름 서버로 169.254.169.253가 반환됩니다. 옵션 세트에 'AmazonProvidedDNS'가 누락된 경우 옵션 세트에 다른 IPv4 이름 서버가 언급되어 있는지 여부와 관계없이 IPv4 DHCP 서버에서는 주소가 반환되지 않습니다.

또한 Amazon DHCP 서버는 접두사 위임을 사용하여 VPC의 네트워크 인터페이스에 전체 IPv4 또는 IPv6 접두사를 제공할 수도 있습니다(Amazon EC2 사용 설명서의 [Amazon EC2 네트워크 인터페이스에 접두사 할당](#) 참조). IPv4 접두사 위임은 DHCP 응답에서 제공되지 않습니다. 인터페이스에 할당된 IPv4 접두사는 IMDS를 사용하여 검색할 수 있습니다(Amazon EC2 사용 설명서의 [인스턴스 메타데이터 범주](#) 참조).

DHCP 옵션 세트 개념

DHCP 옵션 세트는 EC2 인스턴스와 같은 VPC의 리소스가 가상 네트워크를 통해 통신하는 데 사용하는 네트워크 설정 그룹입니다.

각 리전에 기본 DHCP 옵션 세트가 있습니다. 각 VPC는 해당 리전의 기본 DHCP 옵션 세트를 사용합니다. 단, 사용자 지정 DHCP 옵션 세트를 만들어 VPC에 연결하거나 DHCP 옵션 세트 없이 VPC를 구성한 경우는 예외입니다.

구성된 DHCP 옵션 세트가 VPC에 없는 경우:

- [Nitro 시스템에 구축된 EC2 인스턴스](#)의 경우 AWS에서는 169.254.169.253이 기본 도메인 이름 서버로 구성됩니다.
- [Xen에 구축된 EC2 인스턴스](#)의 경우 도메인 이름 서버가 구성되지 않으며, DNS 서버에 액세스하는 권한이 VPC의 인스턴스에 없기 때문에 해당 인스턴스에서 인터넷에 액세스할 수 없습니다.

여러 VPC와 DHCP 옵션 세트를 연결할 수 있지만 각 VPC에는 하나의 DHCP 옵션 세트만 연결되어 있어야 합니다.

VPC를 삭제하면 VPC와 연결된 DHCP 옵션 세트가 VPC에서 분리됩니다.

내용

- [기본 DHCP 옵션 세트](#)
- [사용자 정의 DHCP 옵션 세트](#)

기본 DHCP 옵션 세트

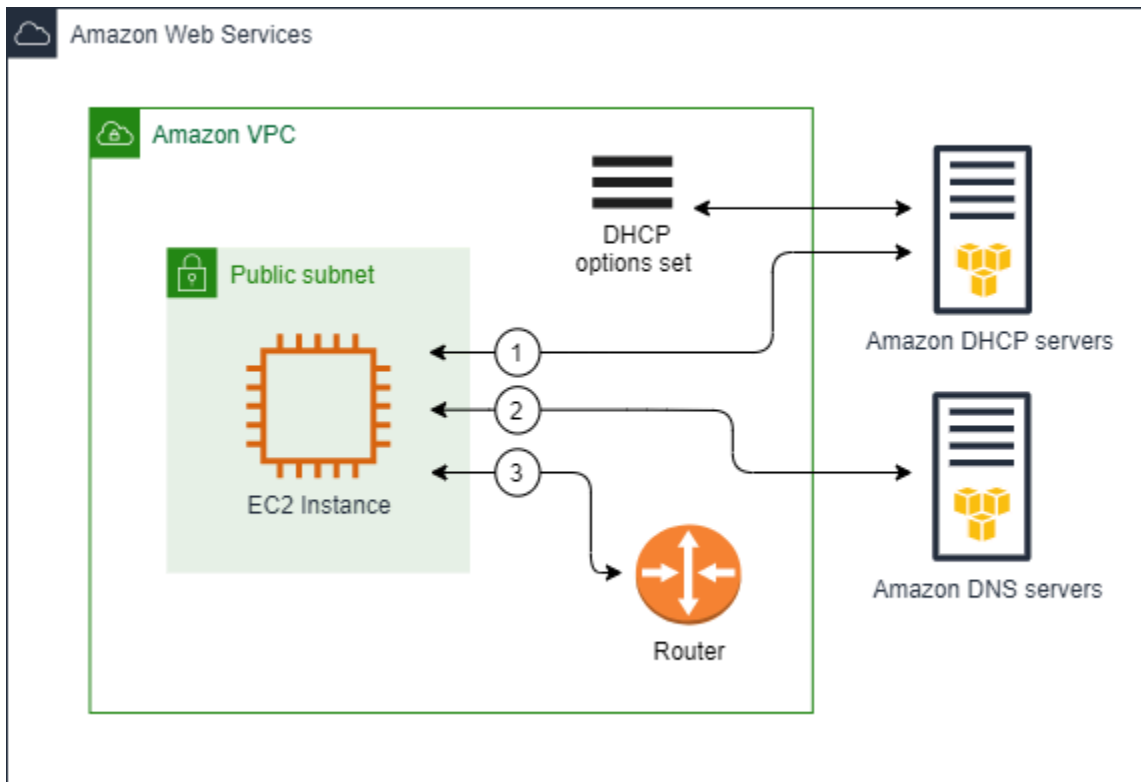
기본 DHCP 옵션 세트에는 다음과 같은 설정이 포함되어 있습니다.

- 도메인 이름 서버: 네트워크 인터페이스가 도메인 이름 확인에 사용하는 DNS 서버입니다. 기본 DHCP 옵션 세트의 경우 이 항목은 언제나 AmazonProvidedDNS입니다. 자세한 내용은 [Amazon DNS 서버](#) 단원을 참조하십시오.
- 도메인 이름: 도메인 이름 시스템(DNS)을 사용하여 호스트 이름을 확인하는 경우 클라이언트가 사용해야 하는 도메인 이름입니다. EC2 인스턴스에 사용되는 도메인 이름에 대한 자세한 내용은 [Amazon EC2 인스턴스 호스트 이름](#)을 참조하세요.
- IPv6 선호 임대 시간: IPv6가 할당되어 실행 중인 인스턴스의 DHCPv6 임대 갱신 빈도입니다. 기본 임대 시간은 140초입니다. 임대 갱신은 일반적으로 임대 기간의 절반이 경과했을 때 발생합니다.

기본 DHCP 옵션 세트를 사용하는 경우 다음의 설정은 사용되지 않습니다. 하지만 EC2 인스턴스에 대한 기본값이 있습니다.

- NTP 서버: 기본적으로 EC2 인스턴스는 시간을 검색하기 위해 [Amazon Time Sync Service](#)를 사용합니다.
- NetBIOS 이름 서버: Windows에서 실행되는 EC2 인스턴스의 경우 NetBIOS 컴퓨터 이름은 네트워크에서 인스턴스를 식별하기 위해 인스턴스에 할당된 친숙한 이름입니다. NetBIOS 이름 서버는 NetBIOS를 이름 지정 서비스로 사용하는 네트워크의 NetBIOS 컴퓨터 이름과 네트워크 주소 간의 매핑 목록을 유지 관리합니다.
- NetBIOS 노드 유형: Windows에서 실행되는 EC2 인스턴스의 경우 인스턴스가 NetBIOS 이름을 IP 주소로 확인하는 데 사용하는 방법입니다.

기본 옵션 세트를 사용하는 경우 Amazon DHCP 서버는 기본 옵션 세트의 네트워크 설정을 사용합니다. VPC에서 인스턴스를 실행하면 다이어그램에 표시된 대로 (1) DHCP 서버와 상호 작용하고, (2) Amazon DNS 서버와 상호 작용하며, (3) VPC용 라우터를 통해 네트워크의 다른 장치에 연결됩니다. 인스턴스는 언제든지 Amazon DHCP 서버와 상호 작용하여 IP 주소 임대 및 추가 네트워크 설정을 가져올 수 있습니다.

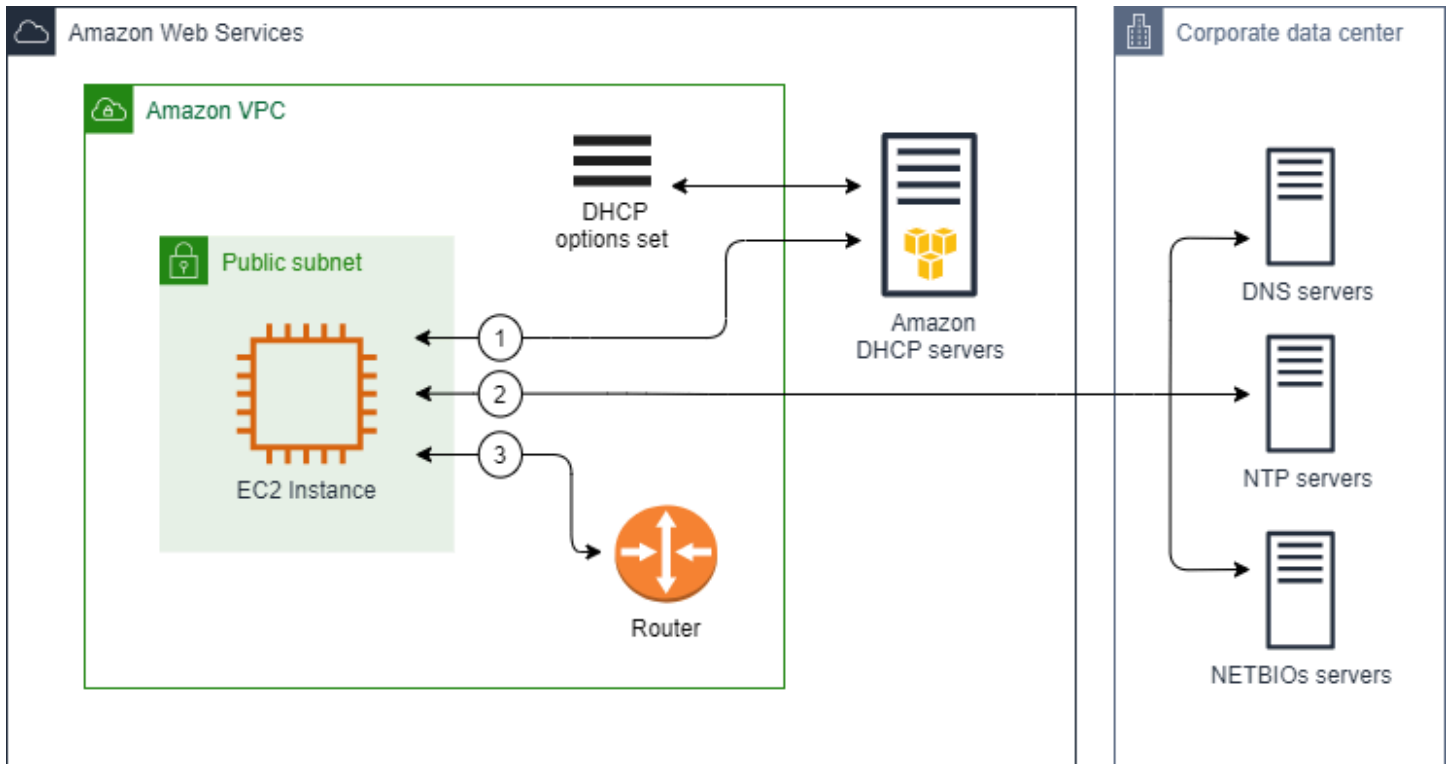


사용자 정의 DHCP 옵션 세트

설정이 다음과 같은 사용자 지정 DHCP 옵션 세트를 생성한 후 VPC와 연결할 수 있습니다.

- 도메인 이름 서버: 네트워크 인터페이스가 도메인 이름 확인에 사용하는 DNS 서버입니다.
- 도메인 이름: 도메인 이름 시스템(DNS)을 사용하여 호스트 이름을 확인하는 경우 클라이언트가 사용하는 도메인 이름입니다.
- NTP 서버: 인스턴스에 시간을 제공하는 NTP 서버입니다.
- NetBIOS 이름 서버: Windows에서 실행되는 EC2 인스턴스의 경우 NetBIOS 컴퓨터 이름은 네트워크에서 인스턴스를 식별하기 위해 인스턴스에 할당된 친숙한 이름입니다. NetBIOS 이름 서버는 NetBIOS를 이름 지정 서비스로 사용하는 네트워크의 NetBIOS 컴퓨터 이름과 네트워크 주소 간의 매핑 목록을 유지 관리합니다.
- NetBIOS 노드 유형: Windows에서 실행되는 EC2 인스턴스의 경우 인스턴스가 NetBIOS 이름을 IP 주소로 확인하는 데 사용하는 방법입니다.
- IPv6 선호 임대 시간(선택 사항): IPv6가 할당되어 실행 중인 인스턴스의 DHCPv6 임대 갱신 빈도의 값입니다(초, 분, 시간 또는 년). 허용되는 값은 140~4,294,967,295초(약 138년)입니다. 값을 입력하지 않는 경우 기본 임대 시간은 140초입니다. EC2 인스턴스에 장기 주소 지정을 사용하면 임대 시간을 늘리고 빈번한 임대 갱신 요청을 방지할 수 있습니다. 임대 갱신은 일반적으로 임대 기간의 절반이 경과했을 때 발생합니다.

사용자 지정 옵션 세트를 사용하면 다이어그램처럼 VPC에서 실행된 인스턴스가 (1) 사용자 지정 DHCP 옵션 세트의 네트워크 설정 사용, (2) 사용자 정의 DHCP 옵션 세트에 지정된 DNS, NTP 및 NetBIOS 서버와 상호 작용, (3) VPC용 라우터를 통해 네트워크의 다른 장치에 연결 작업을 수행합니다.



관련 작업

- [DHCP 옵션 세트 생성](#)
- [VPC와 연결된 옵션 세트 변경](#)

DHCP 옵션 세트로 작업

다음 절차를 따라 DHCP 옵션 세트를 확인하고 작업하세요. DHCP 옵션 세트의 작동 방식에 대한 자세한 내용은 [the section called “DHCP 옵션 세트 개념”](#) 섹션을 참조하세요.

Tasks

- [DHCP 옵션 세트 보기](#)
- [DHCP 옵션 세트 생성](#)
- [VPC와 연결된 옵션 세트 변경](#)
- [DHCP 옵션 세트 삭제](#)

DHCP 옵션 세트 보기

다음과 같이 DHCP 옵션 세트를 확인할 수 있습니다. 기본 DHCP 옵션 세트의 경우 값이 있는 유일한 설정은 도메인 이름과 도메인 이름 서버입니다.

콘솔을 사용하여 DHCP 옵션 세트를 보려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 DHCP option sets(DHCP 옵션 세트)를 선택합니다.
3. DHCP 옵션 세트의 ID를 선택하여 세부 정보 페이지를 엽니다.

명령줄을 사용하여 DHCP 옵션 세트를 보려면

명령줄 인터페이스에 대한 자세한 내용은 [Amazon VPC 작업](#) 단원을 참조하세요.

- [describe-dhcp-options](#)(AWS CLI)
- [Get-EC2DhcpOption](#)(AWS Tools for Windows PowerShell)

DHCP 옵션 세트 생성

사용자 지정 DHCP 옵션 세트를 사용하면 자체 DNS 서버, 도메인 이름 등으로 VPC를 사용자 지정할 수 있습니다. 원하는 만큼 DHCP 옵션 세트를 추가로 생성할 수 있습니다. 하지만 한 번에 한 DHCP 옵션 세트와 한 VPC만 연결할 수 있습니다.

Note

DHCP 옵션 세트를 생성한 후에는 이 옵션 세트를 수정할 수 없습니다. VPC의 DHCP 옵션을 업데이트하려면 새 DHCP 옵션 세트를 생성한 다음 이를 VPC와 연결해야 합니다.

콘솔을 사용하여 DHCP 옵션 세트를 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 DHCP option sets(DHCP 옵션 세트)를 선택합니다.
3. DHCP 옵션 세트 생성(Create DHCP options set)을 선택합니다.
4. 태그 설정(Tag settings)의 경우 선택적으로 DHCP 옵션 세트의 이름을 입력합니다. 값을 입력하면 DHCP 옵션 세트의 이름 태그가 자동으로 생성됩니다.

5. DHCP 옵션의 경우 필요한 구성 설정을 제공합니다.

- 도메인 이름(Domain name)(선택 사항): 도메인 이름 시스템을 통해 호스트 이름을 확인하는 경우 클라이언트가 사용해야 하는 도메인 이름을 입력합니다. AmazonProvidedDNS를 사용하지 않는 경우에는 사용자 지정 도메인 이름 서버가 적절하게 호스트 이름을 확인해야 합니다. Amazon Route 53 프라이빗 호스팅 영역을 사용하는 경우 AmazonProvidedDNS를 사용할 수 있습니다. 자세한 내용은 [VPC의 DNS 속성](#) 단원을 참조하십시오.

일부 Linux 운영 체제에서는 공백으로 구분된 여러 도메인 이름을 허용합니다. 하지만 Windows와 기타 Linux 운영 체제에서는 이 값을 단일 도메인으로 취급하므로 예기치 않은 동작이 발생합니다. DHCP 옵션 세트가 값을 단일 도메인으로 취급하는 운영 체제가 실행되는 인스턴스가 있는 VPC와 연결되어 있는 경우 도메인 이름을 하나만 지정합니다.

- 도메인 이름 서버(Domain name servers)(선택 사항): 호스트의 이름에서 호스트의 IP 주소를 확인하는 데 사용할 DNS 서버를 입력합니다.

AmazonProvidedDNS 또는 사용자 지정 도메인 이름 서버 중 하나를 입력할 수 있습니다. 둘 다 사용하면 예상치 못한 동작이 발생할 수 있습니다. 최대 4개의 IPv4 도메인 이름 서버(또는 최대 3개의 IPv4 도메인 이름 서버 및 **AmazonProvidedDNS**)와 4개의 IPv6 도메인 이름 서버의 IP 주소를 쉼표로 구분하여 입력할 수 있습니다. 최대 8개의 도메인 이름 서버를 지정할 수 있지만 일부 운영 체제에서는 더 낮은 제한이 적용될 수 있습니다. AmazonProvidedDNS 및 Amazon DNS 서버에 대한 자세한 내용은 [Amazon DNS 서버](#) 섹션을 참조하세요.

Important

VPC에 인터넷 게이트웨이가 있는 경우 도메인 이름 서버 값에 자체 DNS 서버 또는 Amazon DNS 서버(AmazonProvidedDNS)를 지정해야 합니다. 그렇지 않으면 VPC의 인스턴스가 DNS에 액세스할 수 없고 이로 인해 인터넷 액세스가 비활성화됩니다.

- NTP 서버(NTP servers)(선택 사항): 최대 8개의 NTP(Network Time Protocol) 서버의 IP 주소(4개의 IPv4 주소와 4개의 IPv6 주소)를 입력합니다.

NTP 서버는 네트워크에 시간을 제공합니다. IPv4 주소 169.254.169.123 또는 IPv6 주소 fd00:ec2:::123에 Amazon Time Sync Service를 지정할 수 있습니다. 인스턴스는 기본적으로 Amazon Time Sync Service와 통신합니다. IPv6 주소는 [Nitro 시스템에 구축된 EC2 인스턴스](#)에만 액세스할 수 있습니다.

NTP 서버 옵션에 대한 자세한 내용은 [RFC 2132](#)를 참조하세요. Amazon Time Sync Service에 대한 자세한 내용은 Amazon EC2 사용 설명서의 [인스턴스의 시간 설정](#)을 참조하세요.

- NetBIOS 이름 서버(NetBIOS name servers)(선택 사항): 최대 4개의 NetBIOS 이름 서버의 IP 주소를 입력합니다.

Windows OS에서 실행되는 EC2 인스턴스의 경우 NetBIOS 컴퓨터 이름은 네트워크에서 인스턴스를 식별하기 위해 인스턴스에 할당된 친숙한 이름입니다. NetBIOS 이름 서버는 NetBIOS를 이름 지정 서비스로 사용하는 네트워크의 NetBIOS 컴퓨터 이름과 네트워크 주소 간의 매핑 목록을 유지 관리합니다.

- NetBIOS 노드 유형(NetBIOS node type)(선택 사항): **1**, **2**, **4** 또는 **8**을 입력합니다. **2**(지점 간 또는 P-노드)를 지정하는 것이 좋습니다. 브로드캐스트 및 멀티캐스트는 현재 지원되지 않습니다. 이러한 노드 유형에 대한 자세한 내용은 [RFC 2132](#)의 단원 8.7 및 [RFC1001](#)의 단원 10을 참조하십시오.

Windows OS에서 실행되는 EC2 인스턴스의 경우 인스턴스가 NetBIOS 이름을 IP 주소로 확인하는 데 사용하는 방법입니다. 기본 옵션 세트에는 NetBIOS 노드 유형에 대한 값이 없습니다.

- IPv6 선호 임대 시간(선택 사항): IPv6가 할당되어 실행 중인 인스턴스의 DHCPv6 임대 갱신 빈도의 값입니다(초, 분, 시간 또는 년). 허용되는 값은 140~2,147,483,647초(약 68년)입니다. 값을 입력하지 않는 경우 기본 임대 시간은 140초입니다. EC2 인스턴스에 장기 주소 지정을 사용하면 임대 시간을 늘리고 빈번한 임대 갱신 요청을 방지할 수 있습니다. 임대 갱신은 일반적으로 임대 기간의 절반이 경과했을 때 발생합니다.
6. 태그(Tags)를 추가합니다.
 7. DHCP 옵션 세트 생성(Create DHCP options set)을 선택합니다. 새 DHCP 옵션 세트의 이름이나 ID를 기록해 둡니다.
 8. 새 옵션 세트를 사용하도록 VPC를 구성하려면 [VPC와 연결된 옵션 세트 변경](#) 섹션을 참조하세요.

명령줄을 사용하여 VPC에 대한 DHCP 옵션 세트를 생성하려면

명령줄 인터페이스에 대한 자세한 내용은 [Amazon VPC 작업](#) 단원을 참조하세요.

- [create-dhcp-options](#)(AWS CLI)
- [New-EC2DhcpOption](#)(AWS Tools for Windows PowerShell)

VPC와 연결된 옵션 세트 변경

DHCP 옵션 세트를 만들고 나면 해당 옵션 세트를 하나 이상의 VPC와 연결할 수 있습니다. 한 번에 하나의 DHCP 옵션 세트만 VPC와 연결할 수 있습니다. DHCP 옵션 세트를 VPC에 연결하지 않을 경우 VPC의 도메인 이름 확인이 비활성화됩니다.

새 DHCP 옵션 세트를 VPC와 연결하면 해당 VPC에서 시작하는 모든 새 인스턴스와 기존 인스턴스가 새 옵션을 사용합니다. 인스턴스를 다시 시작하거나 다시 실행할 필요가 없습니다. 인스턴스가 DHCP 임대 갱신을 빈도에 따라 몇 시간 안에 변경 내용이 자동으로 파악됩니다. 선호하는 경우 인스턴스에서 운영 체제를 사용하여 임대를 명시적으로 갱신할 수 있습니다.

콘솔을 사용하여 VPC와 연결된 DHCP 옵션 세트를 변경하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 사용자 VPC(Your VPCs)를 선택합니다.
3. VPC의 확인란을 선택한 다음에 Actions(작업), Edit VPC settings(VPC 설정 편집)를 선택합니다.
4. DHCP 옵션 세트(DHCP options set)에서 새 DHCP 옵션 세트를 선택합니다. 아니면 DHCP 옵션 세트 없음을 선택하여 VPC의 도메인 이름 확인을 비활성화합니다.
5. Save(저장)를 선택합니다.

명령줄을 사용하여 VPC와 연결된 DHCP 옵션 세트를 변경하려면

명령줄 인터페이스에 대한 자세한 내용은 [Amazon VPC 작업](#) 단원을 참조하세요.

- [associate-dhcp-options](#)(AWS CLI)
- [Register-EC2DhcpOption](#)(AWS Tools for Windows PowerShell)

DHCP 옵션 세트 삭제

더 이상 DHCP 옵션 세트가 필요하지 않으면 다음 절차에 따라 DHCP 옵션 세트를 삭제합니다. DHCP 옵션 세트가 사용 중인 경우 해당 옵션 세트는 삭제할 수 없습니다. 삭제하려는 DHCP 옵션 세트와 연결된 각 VPC의 경우, 다른 DHCP 옵션 세트를 VPC에 연결하거나 VPC가 DHCP 옵션 세트를 사용하지 않도록 구성해야 합니다. 자세한 내용은 [the section called “VPC와 연결된 옵션 세트 변경”](#) 단원을 참조하십시오.

콘솔을 사용하여 DHCP 옵션 세트를 삭제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 DHCP option sets(DHCP 옵션 세트)를 선택합니다.
3. DHCP 옵션 세트에 대해 라디오 버튼을 선택한 다음 작업, DHCP 옵션 세트 삭제를 선택합니다.
4. 확인 메시지가 나타나면 **delete**를 입력한 다음 DHCP 옵션 세트 삭제를 선택합니다.

명령줄을 사용하여 DHCP 옵션 세트를 삭제하려면

명령줄 인터페이스에 대한 자세한 내용은 [Amazon VPC 작업](#) 단원을 참조하세요.

- [delete-dhcp-options](#)(AWS CLI)
- [Remove-EC2DhcpOption](#)(AWS Tools for Windows PowerShell)

VPC의 DNS 속성

도메인 이름 시스템(DNS)은 인터넷에서 사용되는 이름을 해당 IP 주소로 확인할 때 기준이 됩니다. DNS 호스트 이름은 컴퓨터 이름을 고유하고 절대적으로 지정하는 이름으로서, 호스트 이름과 도메인 이름으로 구성됩니다. DNS 서버는 DNS 호스트 이름을 해당 IP 주소로 확인합니다.

퍼블릭 IPv4 주소를 사용하면 인터넷으로 통신할 수 있는 반면, 프라이빗 IPv4 주소를 사용하면 인스턴스의 네트워크 내에서 통신할 수 있습니다. 자세한 내용은 [VPC 및 서브넷의 IP 주소 지정](#) 단원을 참조하십시오.

Amazon은 VPC용 DNS 서버([Amazon Route 53 Resolver](#))를 제공합니다. 자체 DNS 서버를 대신 사용하려면 자신의 VPC에 대한 새로운 DHCP 옵션 세트를 생성하세요. 자세한 내용은 [Amazon VPC의 DHCP 옵션 세트](#) 섹션을 참조하세요.

목차

- [Amazon DNS 서버](#)
- [DNS 호스트 이름](#)
- [VPC의 DNS 속성](#)
- [DNS 할당량](#)
- [EC2 인스턴스의 DNS 호스트 이름 보기](#)
- [VPC에 대한 DNS 속성 보기 및 업데이트](#)
- [프라이빗 호스팅 영역](#)

Amazon DNS 서버

Route 53 Resolver('Amazon DNS 서버' 또는 'AmazonProvidedDNS'라고도 함)는 AWS 리전의 각 가용 영역에 내장된 DNS Resolver 서비스입니다. Route 53 Resolver는 169.254.169.253(IPv4), fd00::ec2::253(IPv6), VPC에 프로비저닝된 기본 프라이빗 IPV4 CIDR 범위에 2를 더한 범위에

있습니다. 예를 들어 IPv4 CIDR이 10.0.0.0/16이고 IPv6 CIDR이 fd00:ec2::253인 VPC가 있는 경우 169.254.169.253(IPv4), fd00:ec2::253(IPv6) 또는 10.0.0.2(IPv4)에서 Route 53 Resolver에 연결할 수 있습니다. VPC 내의 리소스에서는 [링크 로컬 주소](#)가 DNS 쿼리에 사용됩니다. 이러한 쿼리는 Route 53 Resolver로 비공개로 전송되며 네트워크에는 표시되지 않습니다. IPv6 전용 서브넷에서는 'AmazonProvidedDNS'가 DHCP 옵션 세트의 이름 서버인 한 IPv4 링크-로컬 주소 (169.254.169.253)에 계속 연결할 수 있습니다.

VPC에 인스턴스를 시작하는 경우 인스턴스에 프라이빗 DNS 호스트 이름을 제공합니다. 인스턴스가 퍼블릭 IPv4 주소로 구성되어 있고 VPC DNS 속성이 사용 설정된 경우 퍼블릭 DNS 호스트 이름도 제공합니다.

프라이빗 DNS 호스트 이름의 형식은 EC2 인스턴스를 시작할 때 구성하는 방법에 따라 다릅니다. 프라이빗 DNS 호스트 이름 유형에 대한 자세한 내용은 [EC2 인스턴스 이름 지정](#) 단원을 참조하십시오.

VPC에 있는 Amazon DNS 서버는 Route 53의 프라이빗 호스팅 영역에서 지정하는 DNS 도메인 이름을 확인하는 데 사용됩니다. 프라이빗 호스팅 영역에 대한 자세한 내용은 [Amazon Route 53 개발자 안내서](#)의 프라이빗 호스팅 영역 작업을 참조하세요.

규칙 및 고려 사항

Amazon DNS 서버를 사용할 때 다음 규칙 및 고려 사항이 적용됩니다.

- 네트워크 ACL 또는 보안 그룹을 사용하여 Amazon DNS 서버의 양방향 트래픽을 필터링할 수는 없습니다.
- Amazon EMR 같은 Hadoop 프레임워크를 사용하는 서비스에서는 인스턴스가 자신의 FQDN(정규화된 도메인 이름)을 확인해야 합니다. 그런 경우, domain-name-servers 옵션이 사용자 지정 값으로 설정되어 있는 경우 DNS 확인이 실패할 수 있습니다. DNS를 올바르게 확인하려면 **region-name.compute.internal** 도메인에 대한 쿼리를 Amazon DNS 서버로 전달하기 위해 DNS 서버 상에 조건부 전달자를 추가하는 방법을 고려하십시오. 자세한 내용은 Amazon EMR 관리 가이드의 [클러스터를 호스팅하도록 VPC를 설정](#)을 참조하십시오.
- Amazon Route 53 Resolver는 재귀 DNS 쿼리만 지원합니다.

DNS 호스트 이름

인스턴스를 시작하면 인스턴스가 프라이빗 IPv4 주소와 프라이빗 IPv6 주소에 해당하는 프라이빗 DNS 호스트 이름을 항상 수신합니다. 인스턴스에 퍼블릭 IPv4 주소가 있는 경우 VPC의 DNS 속성은 퍼블릭 IPv4 주소에 해당하는 퍼블릭 DNS 호스트 이름을 수신할 여부를 결정합니다. 자세한 내용은 [VPC의 DNS 속성](#) 단원을 참조하십시오.

Amazon에서 제공한 DNS 서버를 사용 설정하면 DNS 호스트 이름이 다음과 같이 할당되고 확인됩니다.

프라이빗 IP DNS 이름(IPv4에만 해당)

프라이빗 IP DNS 이름(IPv4 전용) 호스트 이름을 사용하여 동일 VPC의 인스턴스 간에 통신할 수 있습니다. 인스턴스가 동일한 AWS 리전에 있고 다른 인스턴스의 호스트 이름이 [RFC 1918](#): 10.0.0.0 - 10.255.255.255 (10/8 prefix), 172.16.0.0 - 172.31.255.255 (172.16/12 prefix) 및 192.168.0.0 - 192.168.255.255 (192.168/16 prefix)에서 정의한 프라이빗 주소 공간 범위에 있는 한, 다른 VPC에 있는 다른 인스턴스의 프라이빗 IP DNS 이름(IPv4 전용) 호스트 이름을 확인할 수 있습니다.

프라이빗 리소스 DNS 이름

이 인스턴스에 대해 선택한 A 및 AAAA DNS 레코드로 확인할 수 있는 RBN 기반 DNS 이름입니다. 이 DNS 호스트 이름은 이중 스택 및 IPv6 전용 서브넷의 인스턴스에 대한 인스턴스 세부 정보에서 볼 수 있습니다. RBN에 대한 자세한 내용은 [EC2 인스턴스 호스트 이름 유형](#)을 참조하십시오.

퍼블릭 IPv4 DNS

퍼블릭(외부) IPv4 DNS 호스트 이름은 us-east-1 리전의 경우 `ec2-public-ipv4-address.compute-1.amazonaws.com`, 다른 리전의 경우 `ec2-public-ipv4-address.region.compute.amazonaws.com` 형식이 됩니다. Amazon DNS 서버는 퍼블릭 DNS 호스트 이름을 인스턴스 네트워크 외부에서는 인스턴스의 퍼블릭 IPv4 주소로 확인하고, 인스턴스 네트워크 내부에서는 인스턴스의 프라이빗 IPv4 주소로 확인합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [퍼블릭 IPv4 주소 및 외부 DNS 호스트 이름](#)을 참조하십시오.

VPC의 DNS 속성

다음 VPC 속성에 따라 VPC에 제공되는 DNS 지원이 결정됩니다. 두 속성을 모두 사용하면, VPC로 시작된 인스턴스는 생성 시 퍼블릭 IPv4 주소 또는 탄력적 IP 주소가 할당된 경우 퍼블릭 DNS 호스트 이름을 받습니다. 이전에는 모두 사용하지 않았던 두 VPC 속성을 모두 사용하도록 설정하면, 그 VPC로 이미 시작된 인스턴스는 퍼블릭 IPv4 주소 또는 탄력적 IP 주소가 있는 경우 퍼블릭 DNS 호스트 이름을 받습니다.

VPC에 대해 이러한 속성이 활성화되어 있는지 확인하려면 [VPC에 대한 DNS 속성 보기 및 업데이트](#)을 참조하십시오.

속성	설명
<code>enableDnsHostnames</code>	<p>VPC가 퍼블릭 IP 주소가 있는 인스턴스에 퍼블릭 DNS 호스트 이름을 할당하도록 지원할 여부를 결정합니다.</p> <p>VPC가 기본 VPC가 아닌 경우 이 속성의 기본값은 <code>false</code>입니다. 아래 이 속성에 대한 규칙 및 고려 사항을 참고하십시오.</p>
<code>enableDnsSupport</code>	<p>VPC가 Amazon에서 제공하는 DNS 서버를 통해 DNS 확인을 지원하는지 여부를 결정합니다.</p> <p>이 속성이 <code>true</code>인 경우, Amazon 제공 DNS 서버에 대한 쿼리가 성공합니다. 자세한 내용은 Amazon DNS 서버 단원을 참조하십시오.</p> <p>이 속성의 기본값은 <code>true</code>입니다. 아래 이 속성에 대한 규칙 및 고려 사항을 참고하십시오.</p>

규칙 및 고려 사항

- 두 속성이 모두 `true`로 설정되는 경우 다음이 발생합니다.
 - 퍼블릭 IP 주소를 갖는 인스턴스가 해당하는 퍼블릭 DNS 호스트 이름을 받습니다.
 - Amazon Route 53 Resolver 서버는 Amazon에서 제공한 프라이빗 DNS 호스트 이름을 확인할 수 있습니다.
- 속성 중 하나 이상이 `false`로 설정된 경우 다음이 발생합니다.
 - 퍼블릭 IP 주소를 갖는 인스턴스가 해당하는 퍼블릭 DNS 호스트 이름을 받지 않습니다.
 - Amazon Route 53 Resolver는 Amazon에서 제공한 프라이빗 DNS 호스트 이름을 확인할 수 없습니다.
 - 사용자 지정 도메인 이름에 [DHCP 옵션 세트](#)가 있을 경우 인스턴스가 프라이빗 DNS 호스트 이름을 받습니다. Amazon Route 53 Resolver 서버를 사용하지 않는 경우에는 사용자 지정 도메인 이름 서버가 적절하게 호스트 이름을 확인해야 합니다.
- Amazon Route 53의 프라이빗 호스팅 영역에서 정의된 사용자 지정 DNS 도메인 이름을 사용하거나 프라이빗 DNS를 인터페이스 VPC 엔드포인트(AWS PrivateLink)와 함께 사용하는 경우, `enableDnsHostnames` 및 `enableDnsSupport` 속성을 `true`로 설정해야 합니다.

- Amazon Route 53 Resolver는 VPC의 IPv4 주소 범위가 [RFC 1918](#)에 의해 지정된 프라이빗 IPv4 주소 범위를 벗어나는 경우를 비롯하여 모든 주소 공간에 대해 프라이빗 DNS 호스트 이름을 프라이빗 IPv4 주소로 확인할 수 있습니다. 그러나 2016년 10월 이전에 VPC를 생성했다면 VPC의 IPv4 주소 범위가 이러한 범위를 벗어나는 경우 Amazon Route 53 Resolver에서 프라이빗 DNS 호스트 이름을 확인하지 않습니다. 이에 대한 지원을 활성화하려면 [AWS Support](#)에 문의하십시오.
- VPC 피어링을 사용하면 두 VPC 모두에 대해 두 가지 속성을 모두 활성화해야 하며 피어링 연결에 대한 DNS 확인을 활성화해야 합니다. 자세한 정보는 [VPC 피어링 연결에 대해 DNS 확인 사용 설정](#)을 참조하세요.

DNS 할당량

각 EC2 인스턴스는 Route 53 Resolver로(구체적으로 10.0.0.2 및 169.254.169.253과 같은 .2 주소) 네트워크 인터페이스별 초당 1024개의 패킷을 보낼 수 있습니다. 이 할당량은 늘릴 수 없습니다. Route 53 Resolver가 지원하는 초당 DNS 쿼리 수는 쿼리 유형, 응답 크기 및 사용 중인 프로토콜에 따라 다릅니다. 확장 가능한 DNS 아키텍처에 대한 자세한 내용과 권장 사항은 [AWS Active Directory 포함 하이브리드 DNS](#) 기술 가이드를 확인하십시오.

할당량에 도달하면 Route 53 Resolver가 트래픽을 거부합니다. 할당량에 도달하는 원인으로는 DNS 조절 문제나, Route 53 Resolver 네트워크 인터페이스를 사용하는 인스턴스 메타데이터 쿼리 등이 있습니다. VPC DNS 조절 문제를 해결하는 방법에 대한 자세한 내용은 [Amazon에서 제공한 DNS 서버에 대한 내 DNS 쿼리가 VPC DNS 조절로 인해 실패하는지 확인하려면 어떻게 해야 하나요?](#)를 참조하세요. 인스턴스 메타데이터 검색에 대한 자세한 내용은 Amazon EC2 사용 설명서에서 [인스턴스 메타데이터 검색](#)을 참조하세요.

EC2 인스턴스의 DNS 호스트 이름 보기

Amazon EC2 콘솔 또는 명령줄을 사용하면 실행 중인 인스턴스 또는 네트워크 인터페이스의 DNS 호스트 이름을 볼 수 있습니다.

퍼블릭 DNS(IPv4) 및 프라이빗 DNS 필드는 인스턴스와 연결된 VPC에 대해 DNS 옵션을 활성화한 경우 사용할 수 있습니다. 자세한 내용은 [the section called "VPC의 DNS 속성"](#) 단원을 참조하십시오.

Instance

콘솔을 사용하여 인스턴스의 DNS 호스트 이름을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 인스턴스(Instances)를 선택합니다.

3. 목록에서 해당 인스턴스를 선택합니다.
4. 해당되는 경우, 세부 정보 창의 퍼블릭 DNS(Public DNS)(IPv4) 및 프라이빗 DNS(Private DNS) 필드에 DNS 호스트 이름이 표시됩니다.

명령줄을 사용하여 인스턴스의 DNS 호스트 이름을 보려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon VPC 작업](#) 섹션을 참조하세요.

- [describe-instances](#) (AWS CLI)
- [Get-EC2Instance](#) AWS Tools for Windows PowerShell

네트워크 인터페이스

콘솔을 사용하여 네트워크 인터페이스의 프라이빗 DNS 호스트 이름을 보려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 네트워크 인터페이스(Network Interfaces)를 선택합니다.
3. 목록에서 네트워크 인터페이스를 선택합니다.
4. 세부 정보 창의 프라이빗 DNS(IPv4) 필드에 프라이빗 DNS 호스트 이름이 표시됩니다.

명령줄을 사용하여 네트워크 인터페이스의 DNS 호스트 이름을 보려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon VPC 작업](#) 단원을 참조하세요.

- [describe-network-interfaces](#)(AWS CLI)
- [Get-EC2NetworkInterface](#)(AWS Tools for Windows PowerShell)

VPC에 대한 DNS 속성 보기 및 업데이트

Amazon VPC 콘솔을 이용해 VPC의 DNS 지원 속성을 확인하고 업데이트할 수 있습니다.

콘솔을 사용하여 VPC에 대한 DNS 지원을 설명하고 업데이트하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 Your VPCs(사용자 VPC)를 선택합니다.

3. VPC에 대한 확인란을 선택합니다.
4. 세부 정보(Details)의 정보를 검토합니다. 이 예에서는 DNS 호스트 이름(DNS hostnames) 및 DNS 확인(DNS resolution)이 모두 활성화되었습니다.

Details	CIDRs	Flow logs	Tags
Details			
VPC ID vpc-e03dd489	State Available	DNS hostnames Enabled	DNS resolution Enabled

5. 이러한 설정을 업데이트하려면 Actions(작업)와 Edit VPC settings(VPC 설정 편집)를 차례로 선택합니다. 적절한 DNS 속성에 대한 Enable(활성화)을 선택하거나 선택 취소하고 Save changes(변경 사항 저장)를 선택합니다.

명령줄을 사용하여 VPC에 대한 DNS 지원을 설명하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon VPC 작업 단원](#)을 참조하세요.

- [describe-vpc-attribute](#)(AWS CLI)
- [Get-EC2VpcAttribute](#)(AWS Tools for Windows PowerShell)

명령줄을 사용하여 VPC에 대한 DNS 지원을 업데이트하려면

다음 명령 중 하나를 사용할 수 있습니다. 명령줄 인터페이스에 대한 자세한 내용은 [Amazon VPC 작업 단원](#)을 참조하십시오.

- [modify-vpc-attribute](#)(AWS CLI)
- [Edit-EC2VpcAttribute](#)(AWS Tools for Windows PowerShell)

프라이빗 호스팅 영역

프라이빗 IPv4 주소 또는 AWS 제공 프라이빗 DNS 호스트 이름을 사용하는 대신 example.com과 같은 사용자 지정 DNS 도메인 이름을 사용하여 VPC의 리소스에 액세스하려면 Route 53에서 프라이빗

호스팅 영역을 생성합니다. 프라이빗 호스팅 영역이란 인터넷에 자신의 리소스를 노출하지 않고 하나 이상의 VPC 내에 있는 도메인과 그 하위 도메인의 트래픽을 라우팅하려는 방식에 대한 정보를 담고 있는 컨테이너입니다. Route 53에서 도메인과 하위 도메인에 대한 쿼리에 응답하는 방식을 결정하는 Route 53 리소스 레코드 세트를 생성할 수 있습니다. 예를 들어 example.com에 대한 브라우저 요청이 VPC의 웹 서버로 라우팅되도록 하려는 경우, 프라이빗 호스팅 영역에 A 레코드를 생성하고 그 웹 서버의 IP 주소를 지정할 것입니다. 프라이빗 호스팅 영역의 생성에 대한 자세한 내용은 Amazon Route 53 개발자 안내서의 [프라이빗 호스팅 영역 작업](#)을 참조하세요.

사용자 지정 DNS 도메인 이름을 사용하여 리소스에 액세스하려면 VPC 내에 있는 인스턴스에 연결되어 있어야 합니다. 인스턴스에서 ping 명령(예: ping mywebserver.example.com)을 사용하여 사용자 지정 DNS 이름에서 프라이빗 호스팅 영역에 있는 리소스에 액세스 가능한지 테스트할 수 있습니다. (인스턴스의 보안 그룹 규칙에서 ping에 대한 인바운드 ICMP 트래픽 작동을 허용하는지 확인해야 합니다.)

프라이빗 호스팅 영역은 VPC 외부에서 전이적 관계를 지원하지 않습니다. 예를 들어 VPN 연결의 반대 쪽에서 사용자 지정 프라이빗 DNS 이름을 사용하여 리소스에 액세스할 수 없습니다.

Important

Amazon Route 53의 프라이빗 호스팅 영역에 정의된 사용자 지정 DNS 도메인 이름을 사용하는 경우, enableDnsHostnames 및 enableDnsSupport 속성을 둘 다 true(으)로 설정해야 합니다.

VPC의 네트워크 주소 사용량

NAU(네트워크 주소 사용량)는 VPC 크기를 계획하고 모니터링하는 데 도움이 되도록 가상 네트워크의 리소스에 적용되는 지표입니다. 각 NAU 단위가 VPC 크기를 나타내는 합계에 포함됩니다.

다음과 같은 VPC 할당량을 통해 VPC 크기가 제한되므로 VPC의 NAU를 구성하는 총 단위 수를 파악하는 것이 중요합니다.

- [네트워크 주소 사용량](#) – 단일 VPC에서 보유할 수 있는 최대 NAU 단위 수 합계입니다. 각 VPC에서는 기본적으로 최대 64,000개의 NAU 단위 수 합계를 보유할 수 있습니다. 256,000개까지 할당량 증가를 요청할 수 있습니다.
- [피어링된 네트워크 주소 사용량](#) – VPC 및 피어링된 모든 VPC의 최대 NAU 단위 수 합계입니다. VPC가 동일한 리전의 다른 VPC와 피어링된 경우 결합된 VPC에서는 기본적으로 128,000개까지 NAU 단위 수 합계를 보유할 수 있습니다. 512,000개까지 할당량 증가를 요청할 수 있습니다. 상이한 리전에서 피어링되는 VPC는 이 제한에 포함되지 않습니다.

다음과 같은 방법으로 NAU를 사용할 수 있습니다.

- 가상 네트워크를 생성하기 전에 여러 VPC에 워크로드를 분산해야 하는지 결정하는 데 도움이 되도록 NAU 단위 수 합계를 계산합니다.
- VPC를 생성한 후에는 VPC에서 NAU 할당량 제한을 초과하지 않도록 Amazon CloudWatch를 사용하여 NAU 사용량을 모니터링합니다. 자세한 내용은 [the section called “CloudWatch 지표”](#) 단원을 참조하십시오.

NAU를 계산하는 방법

NAU를 계산하는 방법을 이해하면 VPC 확장을 계획하는 데 도움이 될 수 있습니다.

다음 표에 VPC의 NAU 개수를 구성하는 리소스와 각 리소스에서 사용하는 NAU 단위 수 합계 수가 설명되어 있습니다. 일부 AWS 리소스는 단일 NAU 단위 수 합계로 표시되고 일부 리소스는 여러 NAU 단위 수 합계로 표시됩니다. 이 표를 사용하여 NAU를 계산하는 방식을 알아볼 수 있습니다.

Resource	NAU 단위 수 합계
VPC의 EC2 인스턴스에 대한 네트워크 인터페이스에 할당된 각 프라이빗 또는 퍼블릭 IPv4 및 각 IPv6 주소	1
EC2 인스턴스에 연결된 추가 네트워크 인터페이스	1
네트워크 인터페이스에 할당된 접두사	1
AZ당 Network Load Balancer	6
AZ당 Gateway Load Balancer	6
AZ당 VPC 엔드포인트	6
Transit Gateway Attachment	6
Lambda 함수	6
NAT 게이트웨이	6
EFS 탑재 대상	6

NAU 예시

다음 예시에서는 NAU를 계산하는 방법을 보여줍니다.

예시 1 - VPC 피어링을 사용하여 연결된 VPC 2개

결합된 NAU 할당량에 동일한 리전의 피어링된 VPC가 포함됩니다.

- VPC 1
 - 별도의 가용 영역에 있는 서브넷 2개의 Network Load Balancer 50개 - NAU 단위 수 합계 600개
 - 하나의 서브넷에 5,000개의 인스턴스(각각 IPv4 주소 및 IPv6 주소 포함) 및 다른 서브넷에 5,000개의 인스턴스(각각 IPv4 주소 및 IPv6 주소 포함) - 20,000개
 - Lambda 함수 100개 - NAU 단위 수 합계 600개
- VPC 2
 - 별도의 가용 영역에 있는 서브넷 2개의 Network Load Balancer 50개 - NAU 단위 수 합계 600개
 - 하나의 서브넷에 5,000개의 인스턴스(각각 IPv4 주소 및 IPv6 주소 포함) 및 다른 서브넷에 5,000개의 인스턴스(각각 IPv4 주소 및 IPv6 주소 포함) - 20,000개
 - Lambda 함수 100개 - NAU 단위 수 합계 600개
- 총 피어링 NAU 개수: 단위 42,400개
- 기본 피어링 NAU 할당량: 단위 128,000개

예시 2 - Transit Gateway를 사용하여 연결된 VPC 2개

Transit Gateway를 사용하여 연결된 VPC는 피어링된 VPC와는 달리, 결합된 NAU 할당량에 포함되지 않습니다.

- VPC 1
 - 별도의 가용 영역에 있는 서브넷 2개의 Network Load Balancer 50개 - NAU 단위 수 합계 600개
 - 하나의 서브넷에 5,000개의 인스턴스(각각 IPv4 주소 및 IPv6 주소 포함) 및 다른 서브넷에 5,000개의 인스턴스(각각 IPv4 주소 및 IPv6 주소 포함) - 20,000개
 - Lambda 함수 100개 - NAU 단위 수 합계 600개
- VPC 2
 - 별도의 가용 영역에 있는 서브넷 2개의 Network Load Balancer 50개 - NAU 단위 수 합계 600개
 - 하나의 서브넷에 5,000개의 인스턴스(각각 IPv4 주소 및 IPv6 주소 포함) 및 다른 서브넷에 5,000개의 인스턴스(각각 IPv4 주소 및 IPv6 주소 포함) - 20,000개

- Lambda 함수 100개 - NAU 단위 수 합계 600개
- VPC당 총 NAU 개수: 단위 21,200개
- VPC당 기본 NAU 할당량: 단위 64,000개

다른 계정과 VPC 공유

VPC 공유를 이용하면 여러 AWS 계정에서 Amazon EC2 인스턴스, Amazon Relational Database Service(RDS) 데이터베이스, Amazon Redshift 클러스터, AWS Lambda 함수 등의 애플리케이션 리소스를 중앙 관리형 공유 Virtual Private Cloud(VPC)로 생성할 수 있습니다. 이 모델에서 VPC(소유자)를 소유하는 계정은 AWS Organizations의 동일한 조직에 속한 다른 계정(참여자)과 한 개 또는 여러 개의 서브넷을 공유합니다. 서브넷을 공유한 후 참여자는 공유된 서브넷의 해당 애플리케이션 리소스를 보고, 생성하고, 수정하고, 삭제할 수 있습니다. 참여자는 다른 참여자 또는 VPC 소유자에 속한 리소스를 보거나 수정하거나 삭제할 수 없습니다.

VPC를 공유하여 높은 상호 연결성을 필요로 하고 동일한 신뢰 경계 내에 있는 애플리케이션에 대해 VPC 내의 암시적 라우팅을 활용할 수 있습니다. 이렇게 하면 생성 및 관리하는 VPC 수가 줄어들고 청구 및 액세스 제어에 별도의 계정을 사용할 수 있습니다. AWS PrivateLink, Transit Gateway, VPC 피어링 같은 연결 기능으로 공유 Amazon VPC를 상호 연결하여 네트워크 토폴로지를 더욱 단순화할 수 있습니다. VPC 공유 혜택에 대한 자세한 내용은 [VPC 공유: 여러 계정 및 VPC 관리에 대한 새로운 접근 방식](#)을 참조하세요.

목차

- [공유 VPC 사전 조건](#)
- [서브넷 공유](#)
- [공유 서브넷 공유 해제](#)
- [공유 서브넷의 소유자 식별](#)
- [VPC 리소스 관리](#)
- [소유자 및 참가자에 대한 책임 및 권한](#)
- [AWS 리소스 및 공유 VPC 서브넷](#)
- [VPC 공유 할당량](#)
- [퍼블릭 서브넷과 프라이빗 서브넷 공유 예](#)

공유 VPC 사전 조건

- VPC 소유자 및 참여자의 계정은 AWS Organizations에서 관리해야 합니다.
- AWS RAM 콘솔의 조직 관리 계정에서 리소스 공유를 활성화해야 합니다. 자세한 내용은 AWS RAM 사용 설명서의 [AWS Organizations 내 리소스 공유 활성화](#)를 참조하세요.
- 리소스 공유를 생성해야 합니다. 리소스 공유를 생성할 때 공유할 서브넷을 지정하거나, 나중에 다음 섹션의 절차를 사용하여 리소스 공유에 서브넷을 추가할 수 있습니다. 자세한 내용은 AWS RAM 사용 설명서의 [리소스 공유 생성](#)을 참조하세요.

서브넷 공유

기본이 아닌 서브넷을 조직의 다른 계정과 다음과 같이 공유할 수 있습니다.

콘솔을 사용하여 서브넷을 공유하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 서브넷을 선택합니다.
3. 서브넷을 선택하고 작업, 서브넷 공유를 선택합니다.
4. 리소스 공유를 선택하고 서브넷 공유를 선택합니다.

AWS CLI를 사용하여 서브넷을 공유하려면

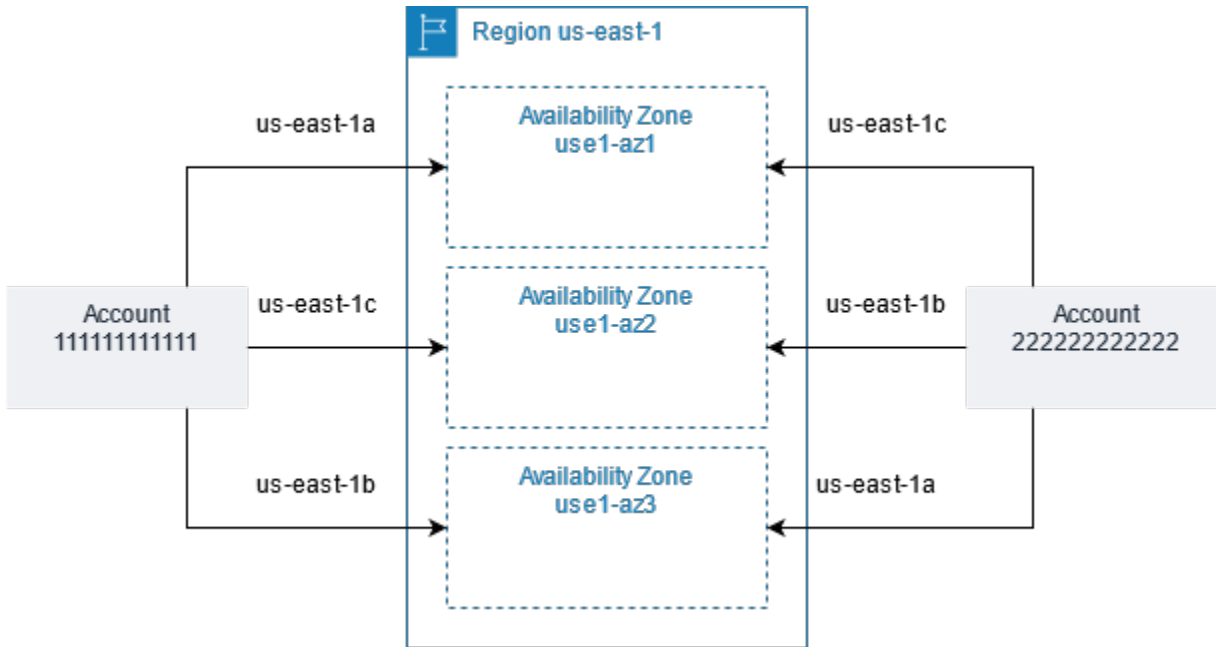
[create-resource-share](#) 명령과 [associate-resource-share](#) 명령을 사용합니다.

여러 가용 영역에서 서브넷 매핑

리전의 가용 영역에 걸쳐 리소스가 배포될 수 있도록 각 계정의 이름에 가용 영역을 독립적으로 매핑합니다. 예를 들어 AWS 계정의 us-east-1a 가용 영역은 다른 AWS 계정에 대한 us-east-1a로 위치가 동일하지 않을 수 있습니다.

VPC 공유를 위해 계정에 대해 가용 영역을 조정하려면 가용 영역에 대한 고유하고 일관된 식별자인 AZ ID를 사용해야 합니다. 예를 들어, use1-az1은 us-east-1 리전의 가용 영역 중 하나에 대한 AZ ID입니다. AZ ID를 사용하여 다른 계정과 관련된 하나의 계정의 리소스 위치를 판단합니다. Amazon VPC 콘솔에서 각 서브넷에 대한 AZ ID를 확인할 수 있습니다.

다음 다이어그램은 가용 영역 코드를 AZ ID에 매핑하는 서로 다른 두 개의 계정을 보여 줍니다.



공유 서브넷 공유 해제

소유자는 참여자와 공유한 서브넷을 언제든지 공유 해제할 수 있습니다. 소유자가 공유한 서브넷을 해제하면 다음 규칙이 적용됩니다.

- 기존 참여자 리소스는 공유 해제된 서브넷에서 계속 실행됩니다. 자동/관리형 워크플로(예: Auto Scaling 또는 노드 교체)가 있는 AWS 관리형 서비스(예: Elastic Load Balancing의 경우 일부 리소스의 공유 서브넷에 지속적으로 액세스해야 할 수 있습니다).
- 참여자는 공유 해제된 서브넷에 더 이상 새로운 리소스를 생성할 수 없습니다.
- 참여자는 서브넷에 있는 리소스를 수정, 기술하고 삭제할 수 있습니다.
- 참여자가 공유 해제된 서브넷의 리소스를 여전히 가지고 있을 경우 소유자는 공유 서브넷 또는 공유 서브넷 VPC를 삭제할 수 없습니다. 참여자가 공유 해제된 서브넷의 모든 리소스를 삭제한 후에만 소유자는 서브넷 또는 공유 서브넷 VPC를 삭제할 수 있습니다.

콘솔을 사용하여 서브넷을 공유 해제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 서브넷을 선택합니다.
3. 서브넷을 선택하고 작업, 서브넷 공유를 선택합니다.
4. 작업, 공유 중지를 선택합니다.

AWS CLI를 사용하여 서브넷을 공유 해제하려면

[disassociate-resource-share](#) 명령을 사용합니다.

공유 서브넷의 소유자 식별

참여자는 공유된 서브넷을 Amazon VPC 콘솔이나 명령줄 도구를 사용하여 볼 수 있습니다.

콘솔을 사용하여 서브넷 소유자를 식별하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 서브넷을 선택합니다. 소유자 옆에 서브넷 소유자가 표시됩니다.

AWS CLI를 사용하여 서브넷 소유자를 식별하려면

[describe-subnets](#) 명령과 [describe-vpcs](#) 명령을 사용합니다. 출력에 소유자의 ID가 포함됩니다.

VPC 리소스 관리

소유자와 참여자는 자신이 소유한 VPC 리소스에 책임이 있습니다.

소유자 리소스

VPC 소유자는 공유 VPC와 연결된 리소스를 생성, 관리 및 삭제할 책임이 있습니다. 이러한 책임에는 서브넷, 라우팅 테이블, 네트워크 ACL, 피어링 연결, 게이트웨이 엔드포인트, 인터페이스 엔드포인트, Amazon Route 53 Resolver 엔드포인트, 인터넷 게이트웨이, NAT 게이트웨이, 가상 프라이빗 게이트웨이 및 전송 게이트웨이 연결이 포함됩니다.

참여자 리소스

참여자는 공유된 VPC에서 제한된 VPC 리소스 세트를 생성할 수 있습니다. 예를 들어 참여자는 네트워크 인터페이스 및 보안 그룹을 생성하고 자신이 소유한 네트워크 인터페이스에 대한 흐름 로그를 활성화할 수 있습니다. 참여자가 생성하는 VPC 리소스는 소유자 계정이 아닌 참여자 계정의 VPC 할당량에 포함됩니다. 자세한 내용은 [VPC 공유](#) 단원을 참조하십시오.

소유자 및 참여자에 대한 청구 및 측정

- 공유 VPC에서 각 참여자는 Amazon EC2 인스턴스, Amazon Relational Database Service 데이터베이스, Amazon Redshift 클러스터 및 AWS Lambda 함수를 비롯한 애플리케이션 리소스에 대한 비용을 지불합니다. 또한 참가자는 가용 영역 간 데이터 전송은 물론 VPC 피어링 연결을 통한 데이터 전

송, 인터넷 게이트웨이 간 데이터 전송 및 AWS Direct Connect 게이트웨이 간 데이터 전송과 연결된 데이터 전송 요금을 지불합니다.

- VPC 소유자는 NAT 게이트웨이, 가상 프라이빗 게이트웨이, 전송 게이트웨이, AWS PrivateLink 및 VPC 엔드포인트에서의 데이터 처리 및 데이터 전송 요금을 시간당 요금(해당하는 경우)으로 지불합니다. 아울러 공유 VPC에서 사용되는 퍼블릭 IPv4 주소는 VPC 소유자에게 요금이 청구됩니다. 퍼블릭 IPv4 주소 요금에 대한 자세한 내용은 [Amazon VPC 요금 페이지](#)의 퍼블릭 IPv4 주소 탭을 참조하세요.
- 동일한 가용 영역(AZ-ID를 사용하여 고유하게 식별됨) 내에서의 데이터 전송은 통신 리소스의 계정 소유권과 상관없이 무료입니다.

소유자 및 참가자에 대한 책임 및 권한

공유 VPC 서브넷으로 작업할 때 VPC 리소스에는 다음과 같은 책임 및 권한이 적용됩니다.

흐름 로그

- 참가자는 자신이 소유하지 않은 공유 VPC 서브넷에서 흐름 로그를 생성, 삭제하거나 이를 설명할 수 없습니다.
- 참가자는 자신이 소유한 공유 VPC 서브넷에서 흐름 로그를 생성, 삭제하거나 이를 설명할 수 있습니다.
- VPC 소유자는 참가자가 생성한 흐름 로그를 설명하거나 삭제할 수 없습니다.

인터넷 게이트웨이 및 외부 전용 인터넷 게이트웨이

- 참가자는 공유 VPC 서브넷에서 인터넷 게이트웨이 및 외부 전용 인터넷 게이트웨이를 생성, 연결 또는 삭제할 수 없습니다. 참가자는 공유 VPC 서브넷의 인터넷 게이트웨이를 설명할 수 있습니다. 참가자는 공유 VPC 서브넷의 송신 전용 인터넷 게이트웨이를 설명할 수 없습니다.

NAT 게이트웨이

- 참가자는 공유 VPC 서브넷에서 NAT 게이트웨이를 생성, 삭제 또는 설명할 수 없습니다.

네트워크 액세스 제어 목록(NACL)

- 참가자는 공유 VPC 서브넷에서 NACL을 생성, 삭제 또는 교체할 수 없습니다. 참가자는 공유 VPC 서브넷에서 VPC 소유자가 생성한 NACL을 설명할 수 있습니다.

네트워크 인터페이스

- 참가자는 공유 VPC 서브넷에서 네트워크 인터페이스를 만들 수 있습니다. 참가자는 공유 VPC 서브넷에서 VPC 소유자가 생성한 네트워크 인터페이스에 대해 다른 방식(예: 네트워크 인터페이스 연결, 연결 해제 또는 수정)으로 작업할 수 없습니다. 참가자는 공유 VPC에서 자신이 생성한 네트워크 인터페이스를 수정 또는 삭제할 수 있습니다. 예를 들어 참가자는 자신이 생성한 네트워크 인터페이스에 IP 주소를 연결하거나 연결 해제할 수 있습니다.
- VPC 소유자는 공유 VPC 서브넷의 참가자가 소유한 네트워크 인터페이스를 설명할 수 있습니다. VPC 소유자는 참가자가 소유한 네트워크 인터페이스를 다른 방식(예: 공유 VPC 서브넷의 참가자가 소유한 네트워크 인터페이스의 연결, 연결 해제 또는 수정)으로 작업할 수 없습니다.

라우팅 테이블

- 참가자는 공유 VPC 서브넷에서 라우팅 테이블에 대한 작업(예: 라우팅 테이블 생성, 삭제 또는 연결)을 수행할 수 없습니다. 참가자는 공유 VPC 서브넷의 라우팅 테이블을 설명할 수 있습니다.

보안 그룹

- 참가자는 공유 VPC 서브넷에서 자신이 소유하는 보안 그룹을 작업할 수 있습니다(수신 및 송신 규칙 생성, 삭제, 설명 또는 수정). 참가자는 VPC 소유자가 생성한 보안 그룹을 다른 방식으로 작업할 수 없습니다.
- 참가자는 자신이 소유한 보안 그룹에 다른 참가자나 VPC 소유자에게 속하는 보안 그룹을 참조하는 규칙을 만들 수 있습니다(예: account-number/security-group-id)
- 참가자는 VPC 소유자나 다른 참가자나 소유한 보안 그룹을 사용하여 인스턴스를 시작할 수 없습니다. 참가자는 VPC의 기본 보안 그룹을 사용하여 인스턴스를 시작할 수 없습니다. 이는 소유자에게 속해 있기 때문입니다.
- 참가자는 공유 VPC 서브넷에서 참가자가 생성한 보안 그룹을 설명할 수 있습니다. VPC 소유자는 참가자가 생성한 보안 그룹을 다른 방식으로 작업할 수 없습니다. 예를 들어, VPC 소유자는 참가자가 생성한 보안 그룹을 사용하여 인스턴스를 시작할 수 없습니다.

서브넷

- 참가자는 공유 서브넷 또는 관련 속성을 수정할 수 없습니다. VPC 소유자만 수정할 수 있습니다. 참가자는 공유 VPC 서브넷에서 서브넷을 설명할 수 있습니다.
- VPC 소유자는 AWS Organizations에서 동일한 조직에 속한 다른 계정 또는 다른 조직 단위와만 서브넷을 공유할 수 있습니다. VPC 소유자는 기본 VPC에 있는 서브넷을 공유할 수 없습니다.

Transit Gateway

- 서브넷 소유자만 공유 VPC 서브넷에 전송 게이트웨이를 연결할 수 있습니다. 참여자는 선택할 수 없습니다.

VPC

- 참가자는 VPC 또는 그 해당 속성을 수정할 수 없습니다. VPC 소유자만 수정할 수 있습니다. 참가자는 VPC, 해당 속성 및 DHCP 옵션 세트를 설명할 수 있습니다.
- VPC 태그와 공유 VPC 내 리소스에 대한 태그는 참여자와 공유되지 않습니다.

AWS 리소스 및 공유 VPC 서브넷

다음 AWS 서비스는 공유 VPC 서브넷의 리소스를 지원합니다. 서비스가 공유 VPC 서브넷을 지원하는 방법에 대한 자세한 내용은 해당 서비스 문서에 대한 링크를 따르세요.

- [Amazon Aurora](#)
- [AWS CodeBuild](#)
- [AWS Database Migration Service](#)
- [Amazon EC2](#)
- [Amazon Elastic Kubernetes Service](#)
- Elastic Load Balancing
 - [Application Load Balancers](#)
 - [Gateway Load Balancers](#)
 - [Network Load Balancers](#)
- [Amazon EMR](#)
- [AWS Glue](#)
- [AWS Lambda](#)
- AWS Network Manager
 - [AWS 클라우드 WAN](#)
 - [Network Access Analyzer](#)
 - [Reachability Analyzer](#)
- [AWS PrivateLink](#)[†]

- [Amazon Relational Database Service\(RDS\)](#)
- [Amazon Redshift](#)
- [Amazon Route 53](#)
- [AWS Transit Gateway](#)
- [AWS Verified Access](#)
- Amazon VPC
 - [피어링](#)
 - [트래픽 미러링](#)
- [Amazon VPC Lattice](#)

† 공유 VPC에서 VPC 엔드포인트를 사용하여 PrivateLink를 지원하는 모든 AWS 서비스에 연결할 수 있습니다. PrivateLink를 지원하는 서비스 목록은 AWS PrivateLink 가이드에서 [AWS PrivateLink와 통합되는 AWS 서비스](#)를 참조하세요.

VPC 공유 할당량

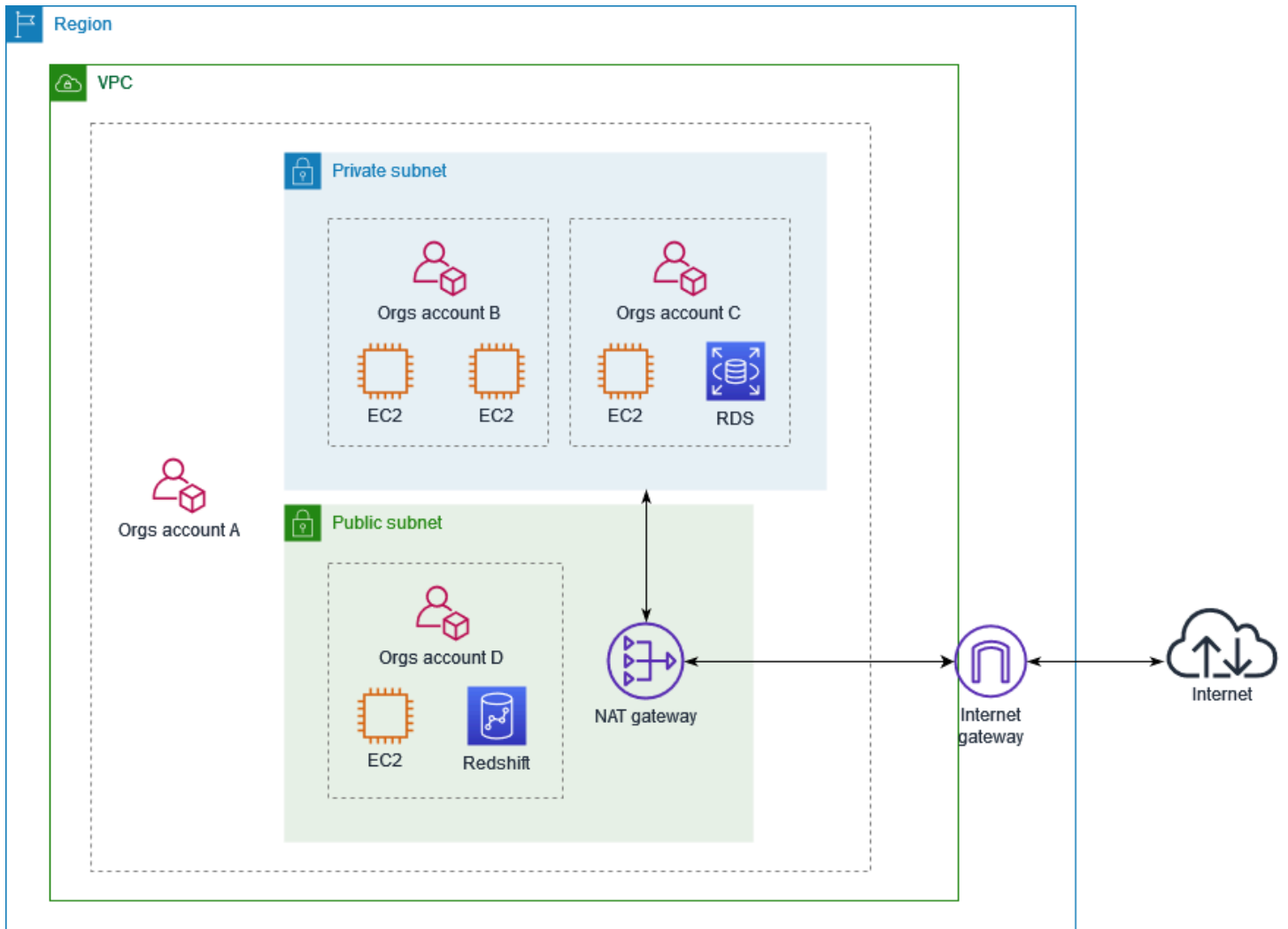
VPC 공유와 관련된 할당량이 있습니다. 자세한 내용은 [VPC 공유](#) 단원을 참조하십시오.

퍼블릭 서브넷과 프라이빗 서브넷 공유 예

계정(계정 A)에서 VPC, 서브넷, 라우팅 테이블, 게이트웨이, CIDR 범위를 포함한 인프라를 관리하고 기타 멤버 계정이 해당 애플리케이션에 대해 서브넷을 사용하도록 하려고 하는 이 시나리오를 검토하세요. 계정 D에는 인터넷 연결이 필요한 애플리케이션이 있습니다. 계정 B와 계정 C에는 인터넷에 연결할 필요가 없는 애플리케이션이 있습니다.

계정 A는 AWS Resource Access Manager를 사용하여 서브넷에 대한 리소스 공유를 만들고 퍼블릭 서브넷을 계정 D와 공유하며 프라이빗 서브넷을 계정 B 및 계정 C와 공유합니다. 계정 B, 계정 C 및 계정 D는 서브넷에서 리소스를 생성할 수 있습니다. 각 계정은 공유된 서브넷의 리소스만 보고 생성할 수 있습니다. 각 계정에서는 이러한 서브넷(예: EC2 인스턴스 및 보안 그룹)에서 만드는 리소스를 제어할 수 있습니다.

공유된 서브넷에 필요한 추가 구성은 없기 때문에 라우팅 테이블은 비공유 서브넷 라우팅 테이블과 동일합니다.



계정 A(111111111111)는 계정 D(444444444444)와 퍼블릭 서브넷을 공유합니다. 계정 D는 다음 서브넷을 볼 수 있으며, 소유자(Owner) 열에는 서브넷이 공유되었음을 나타내는 두 가지 지표가 표시됩니다.

- 소유자 계정 ID는 계정 D(444444444444)가 아니라 계정 A(111111111111)입니다.
- 소유자 계정 ID 옆에 "공유됨"이라는 단어가 표시됩니다.

Create subnet Actions

Filter by tags and attributes or search by keyword

Name	Subnet ID	State	VPC	Default subnet	Owner
	subnet-0bb1c79de301436ee	available	vpc-0ee975135d74bdcfe	No	111111111111 (shared)

로컬 영역, Wavelength 영역 또는 Outpost로 VPC 확장

서브넷과 같은 VPC 리소스를 전 세계 여러 위치에서 호스팅할 수 있습니다. 이 위치는 리전, 가용 영역, Local Zones 및 Wavelength Zone으로 구성됩니다. 각 리전은 개별 지리 영역입니다.

- 가용 영역은 각 리전 내에 있는 여러 격리된 위치입니다.
- Local Zones에서는 최종 사용자에게 가까운 여러 위치에 컴퓨팅, 스토리지 등의 리소스를 배치할 수 있습니다.
- AWS Outposts는 네이티브 AWS 서비스, 인프라 및 운영 모델을 사실상 모든 데이터 센터, 코로케이션 공간 또는 온프레미스 시설로 옮길 수 있습니다.
- Wavelength Zone을 사용하면 개발자는 5G 디바이스 및 최종 사용자에게 매우 짧은 지연 시간을 제공하는 애플리케이션을 빌드할 수 있습니다. Wavelength는 표준 AWS 컴퓨팅 및 스토리지 서비스를 통신 사업자의 5G 네트워크 엣지에 배포합니다.

AWS은 최신 기술을 탑재한 고가용성 데이터 센터를 운영하고 있습니다. 드물기는 하지만 동일한 위치에 있는 인스턴스의 가용성에 영향을 미치는 장애가 발생할 수도 있습니다. 장애의 영향을 받는 위치 한 곳에서 모든 인스턴스를 호스팅하면 인스턴스를 전혀 사용하지 못하게 될 수 있습니다.

[AWS Wavelength FAQ](#)를 참조하면 어떤 배포가 가장 적합한지 판단하는 데 도움이 됩니다.

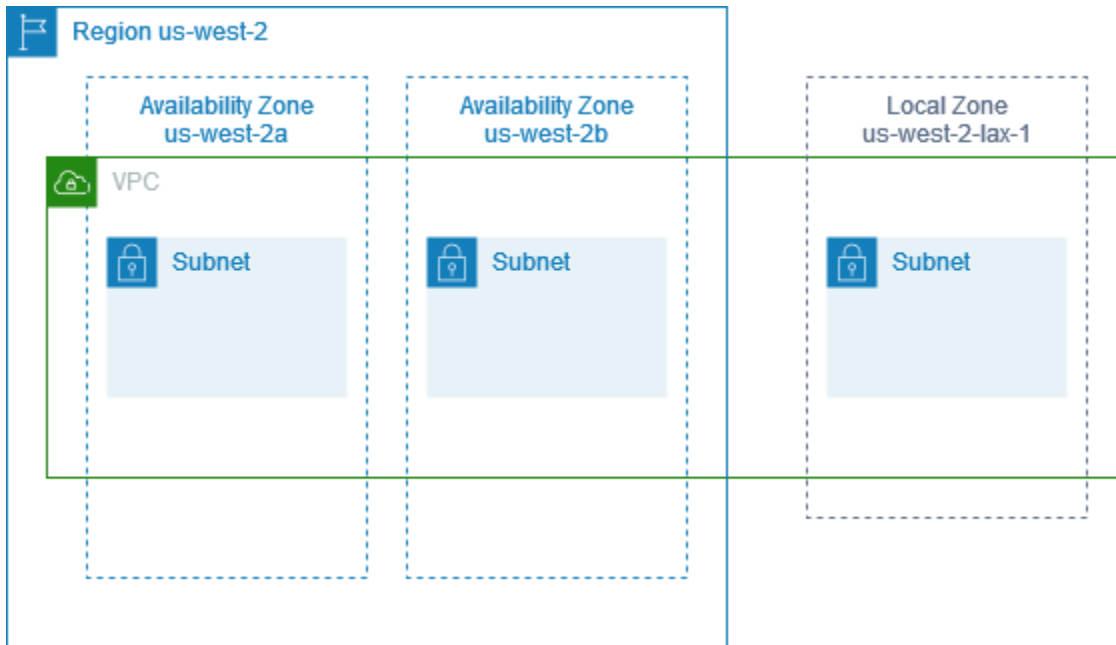
AWS Local Zones의 서브넷

AWS Local Zones를 사용하면 리소스를 최종 사용자에게 가까이 배치하고 친숙한 API 및 도구 세트를 사용하여 AWS 리전의 모든 서비스에 원활하게 연결할 수 있습니다. 로컬 영역에 서브넷을 생성하면 VPC가 해당 로컬 영역으로 확장됩니다.

로컬 영역을 사용하려면 다음과 같은 프로세스를 사용합니다.

- 로컬 영역에 옵트인합니다.
- 로컬 영역에서 서브넷을 만듭니다.
- 애플리케이션이 최종 사용자에게 더 가까이 접근하도록 로컬 영역 서브넷에서 리소스를 시작합니다.

다음 다이어그램은 여러 가용 영역과 하나의 로컬 영역을 아우르는 미국 서부(오레곤)(us-west-2) 리전의 VPC를 보여줍니다.



VPC 를 생성할 때 Amazon에서 제공하는 퍼블릭 IP 주소를 VPC 할당하도록 선택할 수 있습니다. 주소를 그룹으로 제한하는 주소에 대한 네트워크 경계 그룹을 설정하도록 설정할 수도 있습니다. 네트워크 경계 그룹을 설정하면 IP 주소가 네트워크 경계 그룹 간에 이동할 수 없습니다. 로컬 영역 네트워크 트래픽은 로컬 영역의 상위 리전을 통과하지 않고 인터넷이나 접속 지점(POP)으로 직접 이동하므로 대기 시간이 짧은 컴퓨팅에 액세스할 수 있습니다. 로컬 영역 및 해당 상위 리전의 모든 목록을 확인하려면 AWS 로컬 영역 사용 설명서의 [가용 로컬 영역](#) 섹션을 참조하세요.

로컬 영역에는 다음 규칙이 적용됩니다.

- 로컬 영역 서브넷은 라우팅 테이블, 보안 그룹, 네트워크 ACL 등의 가용 영역 서브넷과 동일한 라우팅 규칙을 따릅니다.
- 아웃바운드 인터넷 트래픽은 로컬 영역을 떠납니다.
- 로컬 영역에서 사용할 퍼블릭 IP 주소를 프로비저닝해야 합니다. 주소를 할당할 때 IP 주소가 공고되는 위치를 지정할 수 있습니다. 이를 네트워크 경계 그룹이라고 하며 이 파라미터를 설정하여 주소를 이 위치로 제한할 수 있습니다. IP 주소를 프로비저닝한 후에는 로컬 영역과 상위 리전(예: us-west-2-lax-1a에서 us-west-2로) 간에 IP 주소를 이동할 수 없습니다.
- 로컬 영역에서 IPv6를 지원하는 경우, Amazon에서 제공한 IPv6 IP 주소를 요청하여 새 VPC 또는 기존 VPC의 네트워크 경계 그룹과 연결할 수 있습니다. IPv6이 지원되는 로컬 영역의 목록을 확인하려면 AWS 로컬 영역 사용 설명서의 [고려 사항](#)을 참조하세요.
- 로컬 영역 서브넷에서는 VPC 엔드포인트를 만들 수 없습니다.

로컬 영역 작업에 대한 자세한 내용은 [AWS 로컬 영역 사용 설명서](#)를 참조하세요.

인터넷 게이트웨이에 대한 고려 사항

Local Zones에서 상위 리전의 인터넷 게이트웨이를 사용할 때 다음 사항을 고려하세요.

- 탄력적 IP 주소 또는 Amazon 자동 할당 퍼블릭 IP 주소가 있는 Local Zones에서 인터넷 게이트웨이를 사용할 수 있습니다. 연결하는 탄력적 IP 주소에는 로컬 영역의 네트워크 경계 그룹이 포함되어야 합니다. 자세한 내용은 [the section called “탄력적 IP 주소”](#) 단원을 참조하십시오.

리전에 설정된 탄력적 IP 주소를 연결할 수 없습니다.

- Local Zones에서 사용되는 탄력적 IP 주소는 리전의 탄력적 IP 주소와 동일한 할당량을 갖습니다. 자세한 내용은 [the section called “탄력적 IP 주소”](#) 단원을 참조하십시오.
- 로컬 영역 리소스와 연결된 라우팅 테이블에서 인터넷 게이트웨이를 사용할 수 있습니다. 자세한 내용은 [the section called “인터넷 게이트웨이로 라우팅”](#) 단원을 참조하십시오.

Direct Connect 게이트웨이를 사용하여 Local Zones에 액세스

온프레미스 데이터 센터에서 로컬 영역에 있는 리소스에 액세스하는 시나리오를 생각해보세요. 로컬 영역과 연결된 VPC에 가상 프라이빗 게이트웨이를 사용하여 로컬 영역을 Direct Connect 게이트웨이에 연결합니다. Direct Connect 게이트웨이는 리전의 AWS Direct Connect 위치에 연결됩니다. 온프레미스 데이터 센터에 해당 AWS Direct Connect 위치에 대한 AWS Direct Connect 연결이 있습니다.

Note

Direct Connect를 사용하여 로컬 영역의 서브넷으로 향하는 미국 내 트래픽은 로컬 영역의 상위 리전을 통과하지 않습니다. 그 대신에 트래픽에서는 로컬 영역까지 최단 경로를 이용합니다. 그러면 대기 시간이 감소하고 애플리케이션의 응답성 향상에 도움이 됩니다.

이 구성에 대해 다음 리소스를 구성합니다.

- 로컬 영역 서브넷과 연결된 VPC의 가상 프라이빗 게이트웨이. Amazon Virtual Private Cloud Console의 서브넷 세부 정보 페이지에서 서브넷의 VPC를 보거나 [describe-subnets](#)를 사용합니다.

가상 프라이빗 게이트웨이를 만드는 자세한 방법은 AWS Site-to-Site VPN 사용 설명서의 [대상 게이트웨이 생성](#)을 참조하세요.

- Direct Connect 연결. 최고의 대기 시간 성능을 위해 AWS는 서브넷을 확장할 로컬 영역에 가장 가까운 [Direct Connect 위치](#)를 사용할 것을 권장합니다.

연결 순서 지정 방법에 대한 자세한 내용은 AWS Direct Connect 사용 설명서의 [교차 연결](#)을 참조하세요.

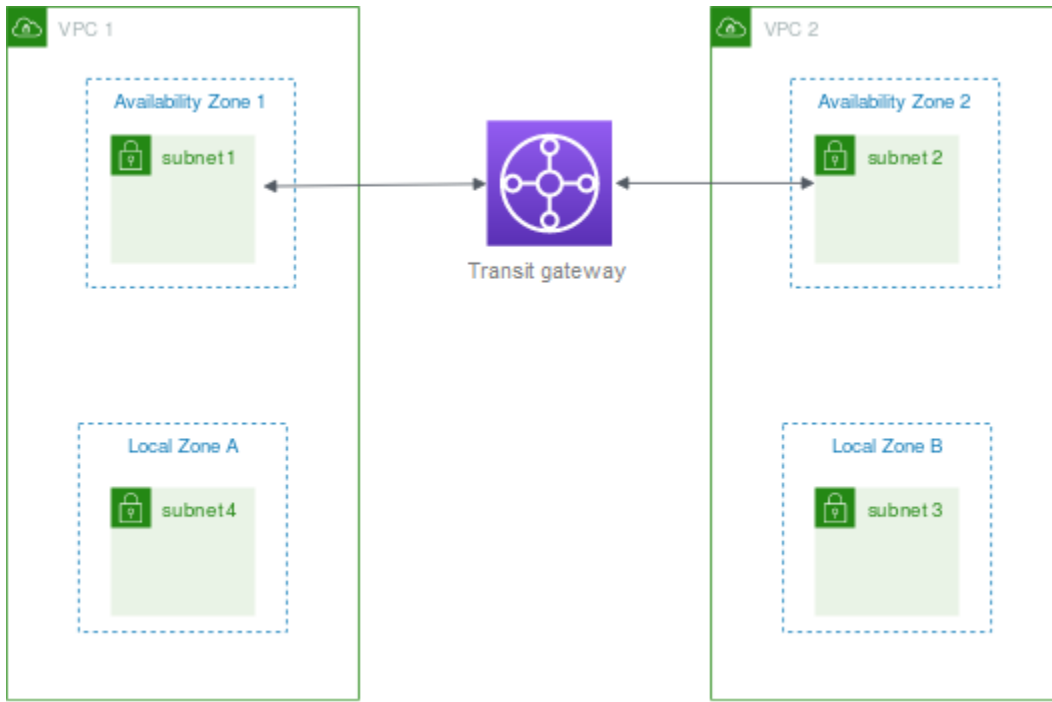
- Direct Connect 게이트웨이 Direct Connect 게이트웨이를 만드는 자세한 방법은 AWS Direct Connect 사용 설명서의 [Direct Connect 게이트웨이 생성](#)을 참조하세요.
- VPC를 Direct Connect 게이트웨이에 연결하기 위한 가상 프라이빗 게이트웨이 연결입니다. 가상 프라이빗 게이트웨이 연결을 생성하는 자세한 방법은 AWS Direct Connect 사용 설명서의 [가상 프라이빗 게이트웨이 연결 및 연결 해제](#)를 참조하세요.
- AWS Direct Connect 위치에서 온프레미스 데이터 센터로의 연결에 대한 프라이빗 가상 인터페이스입니다. Direct Connect 게이트웨이를 만드는 자세한 방법은 AWS Direct Connect 사용 설명서의 [Direct Connect 게이트웨이에 대한 프라이빗 가상 인터페이스 생성](#)을 참조하세요.

Transit Gateway에 로컬 영역 서브넷 연결

로컬 영역의 서브넷에 대해서는 Transit Gateway Attachment를 생성할 수 없습니다. 다음 다이어그램에서는 로컬 영역의 서브넷이 상위 가용 영역을 통해 전송 게이트웨이에 연결되도록 네트워크를 구성하는 방법을 보여 줍니다. Local Zones에 서브넷을 만들고 상위 가용 영역에서 서브넷을 생성합니다. 상위 가용 영역의 서브넷을 전송 게이트웨이에 연결한 다음 다른 VPC CIDR로 향하는 트래픽을 Transit Gateway Attachment에 대한 네트워크 인터페이스로 라우팅하는 각 VPC의 라우팅 테이블에 경로를 만듭니다.

Note

Transit Gateway에서 시작하는 로컬 영역의 서브넷으로 향하는 트래픽은 먼저 상위 리전을 통과합니다.



이 시나리오에서는 다음 리소스를 생성합니다.

- 각 상위 가용 영역에 있는 서브넷입니다. 자세한 내용은 [the section called “서브넷 생성”](#) 단원을 참조하십시오.
- Transit Gateway. 자세한 내용은 Amazon VPC Transit Gateway의 [전송 게이트웨이 생성](#)을 참조하십시오.
- 상위 가용 영역을 사용하는 각 VPC에 대한 Transit Gateway Attachment입니다. 자세한 내용은 Amazon VPC Transit Gateway의 [VPC에 Transit Gateway Attachment 생성](#)을 참조하십시오.
- Transit Gateway Attachment와 연결된 전송 게이트웨이 라우팅 테이블입니다. 자세한 내용은 Amazon VPC Transit Gateways의 [Transit Gateway 라우팅 테이블](#)을 참조하십시오.
- 각 VPC에 대해 다른 VPC CIDR이 대상으로 있고 Transit Gateway Attachment에 대한 네트워크 인터페이스 ID가 대상으로 있는 VPC 라우팅 테이블의 항목입니다. Transit Gateway Attachment의 네트워크 인터페이스를 찾으려면 네트워크 인터페이스 설명에서 Transit Gateway Attachment의 ID를 검색합니다. 자세한 내용은 [the section called “전송 게이트웨이에 대한 라우팅”](#) 섹션을 참조하십시오.

다음은 VPC 1의 라우팅 테이블의 예입니다.

대상 주소	대상
-------	----

대상 주소	대상
<i>VPC 1 CIDR</i>	<i>##</i>
<i>VPC 2 CIDR</i>	<i>vpc1-attachment-network-interface-id</i>

다음은 VPC 2의 라우팅 테이블의 예입니다.

대상 주소	대상
<i>VPC 2 CIDR</i>	<i>##</i>
<i>VPC 1 CIDR</i>	<i>vpc2-attachment-network-interface-id</i>

다음은 Transit Gateway 라우팅 테이블의 예입니다. 각 VPC의 CIDR 블록이 Transit Gateway 라우팅 테이블에 전파됩니다.

CIDR	연결	경로 유형
<i>VPC 1 CIDR</i>	<i>VPC 1# ##</i>	전파
<i>VPC 2 CIDR</i>	<i>VPC 2# ##</i>	전파

AWS Wavelength의 서브넷

AWS Wavelength를 사용하면 개발자는 모바일 디바이스 및 최종 사용자에게 매우 짧은 지연 시간을 제공하는 애플리케이션을 빌드할 수 있습니다. Wavelength는 표준 AWS 컴퓨팅 및 스토리지 서비스를 통신 사업자의 5G 네트워크 엣지에 배포합니다. 개발자는 Virtual Private Cloud(VPC)를 하나 이상의 Wavelength Zone으로 확장한 다음, Amazon EC2 인스턴스와 같은 AWS 리소스를 사용하여 매우 짧은 지연 시간으로 리전의 AWS 서비스에 연결해야 하는 애플리케이션을 실행할 수 있습니다.

Wavelength Zone을 사용하려면 먼저 Zone에 옵트인해야 합니다. 그런 다음 Wavelength Zone에 서브넷을 생성합니다. Wavelength Zone에서는 Amazon EC2 인스턴스, Amazon EBS 볼륨, Amazon VPC 서브넷 및 캐리어 게이트웨이를 생성할 수 있습니다. EC2, EBS 및 VPC와 함께 오케스트레이션 또는 연동되는 Amazon EC2 Auto Scaling, Amazon EKS 클러스터, Amazon ECS 클러스터, Amazon EC2 Systems Manager, Amazon CloudWatch, AWS CloudTrail, AWS CloudFormation 등의 서비스를 사용할 수도 있습니다. Wavelength의 서비스는 Amazon DynamoDB 및 Amazon RDS를 비롯한 서비스에 쉽게 액세스할 수 있도록 안정적인 고대역폭 연결을 통해 AWS 리전에 연결되는 VPC의 일부입니다.

Wavelength Zone에는 다음 규칙이 적용됩니다.

- VPC에 서브넷을 생성하여 Wavelength Zone에 연결하면 VPC가 Wavelength Zone으로 확장됩니다.
- 기본적으로 Wavelength Zone에 걸쳐 있는 VPC에서 생성하는 모든 서브넷은 로컬 경로를 포함하여 기본 VPC 라우팅 테이블을 상속합니다.
- Wavelength Zone의 서브넷에서 EC2 인스턴스를 시작할 때 통신 사업자 IP 주소를 할당합니다. 통신 사업자 게이트웨이는 인터페이스에서 인터넷 또는 모바일 디바이스로의 트래픽에 주소를 사용합니다. 통신 사업자 게이트웨이는 NAT를 사용하여 주소를 변환한 다음 트래픽을 대상으로 보냅니다. 전기 통신 사업자 네트워크의 트래픽은 통신 사업자 게이트웨이를 통해 라우팅됩니다.
- VPC 라우팅 테이블 또는 Wavelength Zone의 서브넷 라우팅 테이블의 대상을 통신 사업자 게이트웨이로 설정할 수 있습니다. 이 경우 특정 위치의 통신 사업자 네트워크에서 들어오는 인바운드 트래픽과 통신 사업자 네트워크 및 인터넷으로의 아웃바운드 트래픽이 허용됩니다. Wavelength Zone의 라우팅 옵션에 대한 자세한 내용은 AWS Wavelength 개발자 안내서의 [라우팅](#)을 참조하세요.
- Wavelength Zone의 서브넷은 IPv4 주소, DHCP 옵션 세트 및 네트워크 ACL을 포함하여 가용 영역의 서브넷과 동일한 네트워킹 구성 요소를 가지고 있습니다.
- Wavelength 영역의 서브넷에 대해서는 Transit Gateway Attachment를 생성할 수 없습니다. 대신 상위 가용 영역의 서브넷을 통해 연결을 생성한 다음 Transit Gateway를 통해 트래픽을 원하는 대상으로 라우팅합니다. 다음 섹션의 예를 참조하세요.

다중 Wavelength Zone 고려 사항

동일한 VPC의 서로 다른 Wavelength Zone에 있는 EC2 인스턴스는 서로 통신할 수 없습니다.

Wavelength Zone과 Wavelength Zone 간 통신이 필요한 경우 AWS에서는 각 Wavelength Zone마다 하나씩 여러 VPC를 사용하는 것이 좋습니다. 전송 게이트웨이를 사용하여 VPC를 연결할 수 있습니다. 이 구성을 사용하면 Wavelength Zone의 인스턴스 간에 통신할 수 있습니다.

Wavelength Zone과 Wavelength Zone 간 트래픽은 AWS 리전을 통해 라우팅됩니다. 자세한 내용은 [AWS Transit Gateway](#)를 참조하세요.

다음 다이어그램은 서로 다른 두 Wavelength Zone의 인스턴스가 통신할 수 있도록 네트워크를 구성하는 방법을 보여줍니다. 두 개의 Wavelength Zone(Wavelength Zone A 및 Wavelength Zone B)이 있습니다. 통신을 활성화하려면 다음 리소스를 생성해야 합니다.

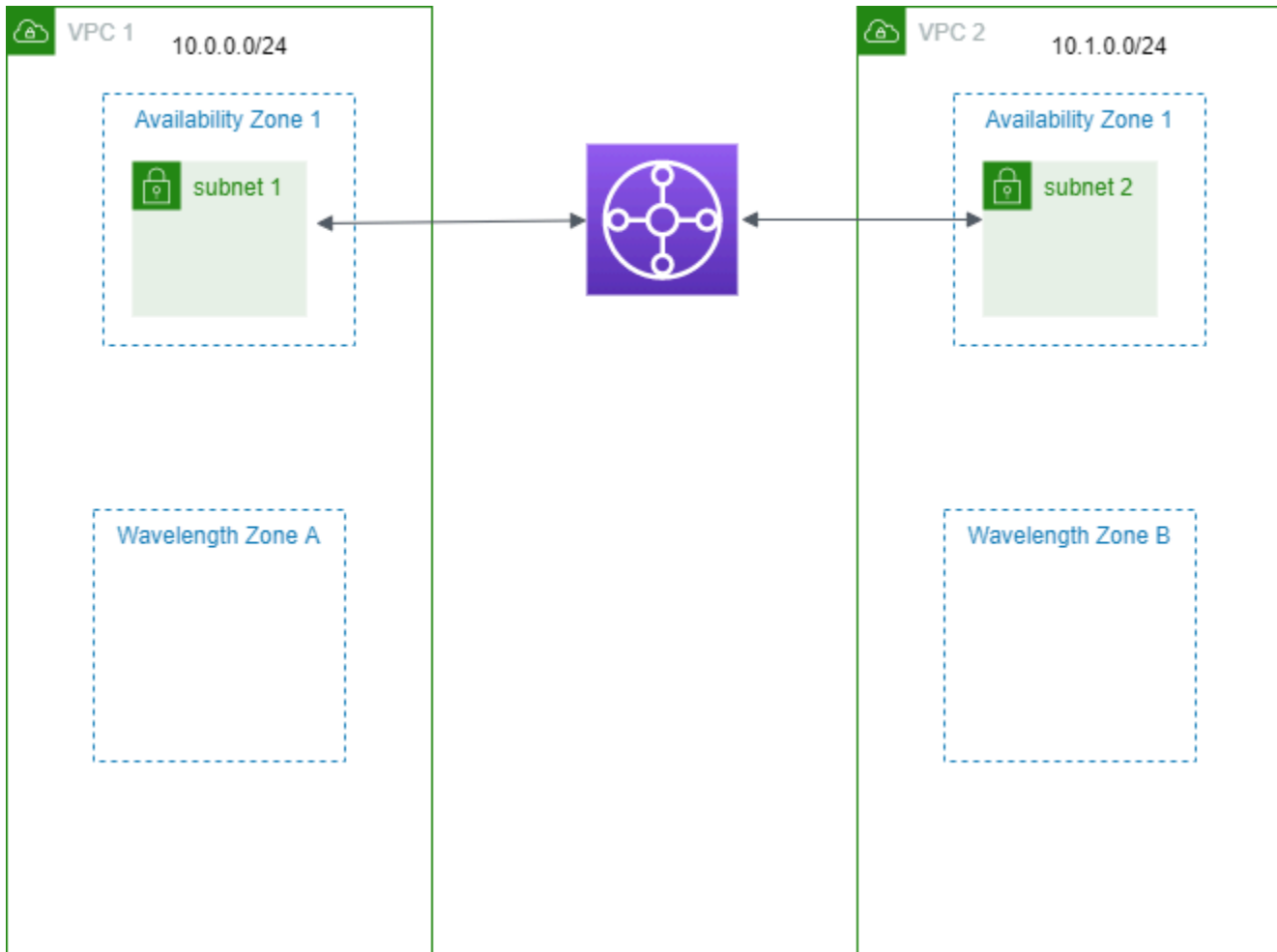
- 각 Wavelength Zone에 대해 Wavelength Zone의 상위 가용 영역인 가용 영역의 서브넷. 이 예에서는 서브넷 1과 서브넷 2를 생성합니다. 서브넷 생성에 대한 자세한 내용은 [the section called “서브넷 생성”](#) 단원을 참조하세요. [describe-availability-zones](#)를 사용하여 상위 영역을 찾습니다.
- 전송 게이트웨이. 전송 게이트웨이는 VPC를 연결합니다. Transit Gateway를 생성하는 자세한 방법은 [Amazon VPC Transit Gateways 가이드](#)의 Transit Gateway 생성을 참조하세요.
- 각 VPC에 대해 Wavelength 영역의 상위 가용 영역에 있는 Transit Gateway에 대한 VPC 연결입니다. 자세한 내용은 Amazon VPC Transit Gateway 가이드의 [VPC에 전송 게이트웨이 연결](#)을 참조하세요.
- Transit Gateway 라우팅 테이블의 각 VPC에 대한 항목. Transit Gateway 경로 생성에 대한 자세한 내용은 Amazon VPC Transit Gateways 안내서에서 [Transit Gateway 라우팅 테이블](#)을 참조하세요.
- 각 VPC에서, 대상으로 다른 VPC CIDR을 목적지로 사용하며 전송 게이트웨이 ID를 대상으로 사용하는 VPC 라우팅 테이블의 항목. 자세한 내용은 [the section called “전송 게이트웨이에 대한 라우팅”](#) 단원을 참조하십시오.

이 예에서 VPC 1의 라우팅 테이블에는 다음 항목이 있습니다.

대상 주소	대상
10.1.0.0/24	tgw-222222222222222222

VPC 2의 라우팅 테이블에는 다음 항목이 있습니다.

대상 주소	대상
10.0.0.0/24	tgw-222222222222222222



AWS Outposts의 서브넷

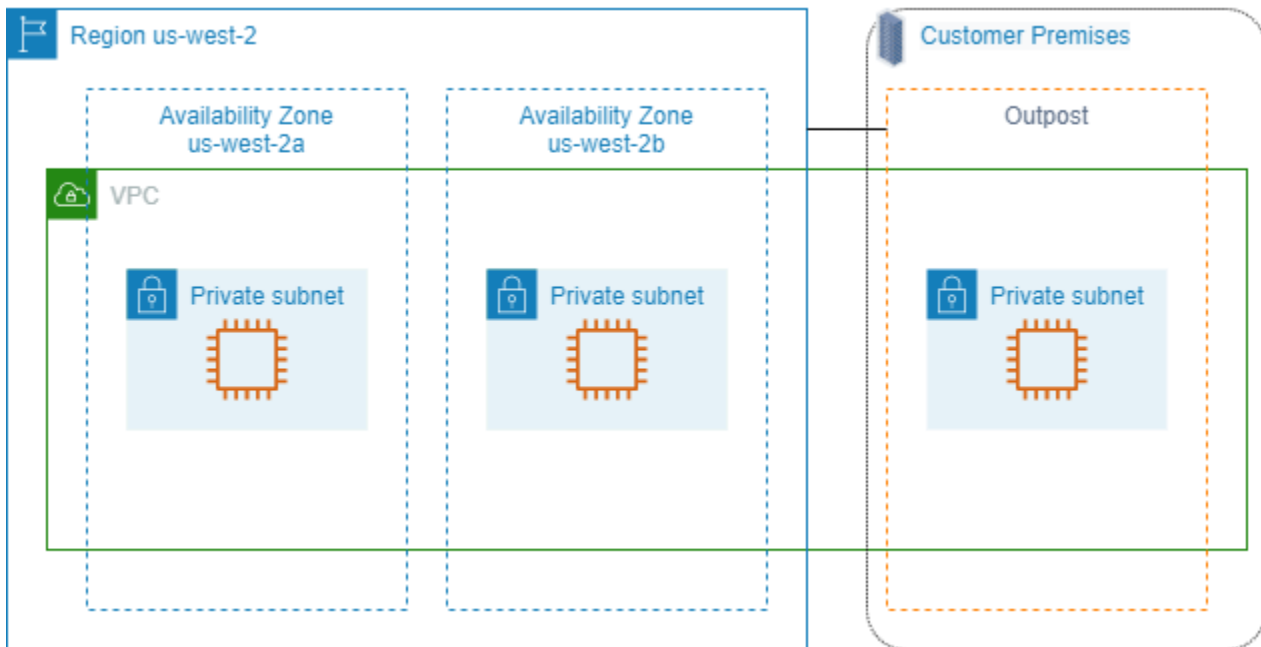
AWS Outposts는 온프레미스 및 클라우드에서 애플리케이션을 구축하고 실행할 수 있는 것과 동일한 AWS 하드웨어 인프라, 서비스, API 및 툴을 제공합니다. AWS Outposts는 온프레미스 애플리케이션 또는 시스템에 대한 짧은 대기 시간 액세스가 필요한 워크로드와 로컬에서 데이터를 저장 및 처리해야 하는 워크로드에 적합합니다. AWS Outposts에 대한 자세한 정보는 [AWS Outposts](#) 섹션을 참조하십시오.

VPC는 AWS 리전의 모든 가용 영역에 적용됩니다. Outpost를 상위 리전에 연결한 후 해당 VPC의 Outpost를 위한 서브넷을 생성하여 해당 리전의 VPC를 Outpost로 확장할 수 있습니다.

다음 규칙은 AWS Outposts에 적용됩니다.

- 서브넷은 하나의 Outposts 위치에 있어야 합니다.
- Outpost를 위한 서브넷을 생성하려면 서브넷을 생성할 때 Outpost의 Amazon 리소스 이름(ARN)을 지정합니다.

- Outposts 랙 - 로컬 게이트웨이가 VPC와 온프레미스 네트워크 간의 네트워크 연결을 처리합니다. 자세한 내용은 Outposts 랙용 AWS Outposts 사용 설명서의 [로컬 게이트웨이](#)를 참조하세요.
- Outposts 랙 - 로컬 게이트웨이가 VPC와 온프레미스 네트워크 간의 네트워크 연결을 처리합니다. 자세한 내용은 Outposts 서버 AWS Outposts 사용 설명서의 [로컬 네트워크 인터페이스](#)를 참조하세요.
- 기본적으로 Outposts의 서브넷을 포함하여 VPC에서 생성하는 모든 서브넷은 암시적으로 VPC의 기본 라우팅 테이블에 연결됩니다. 또는 사용자 지정 라우팅 테이블을 VPC의 서브넷과 명시적으로 연결하고 로컬 게이트웨이를 온프레미스 네트워크로 라우팅할 모든 트래픽에 대한 다음 홉 대상으로 사용할 수 있습니다.



VPC 삭제

VPC 사용을 마치면 이를 삭제할 수 있습니다.

요구 사항

VPC를 삭제하려면 먼저 VPC에서 [요청자 관리형 네트워크 인터페이스](#)를 생성한 모든 리소스를 종료하거나 삭제해야 합니다. 예를 들어 EC2 인스턴스를 종료하고 로드 밸런서, NAT 게이트웨이, 전송 게이트웨이 VPC 연결 및 인터페이스 VPC 엔드포인트를 삭제해야 합니다.

내용

- [콘솔을 사용하여 VPC 삭제](#)
- [명령줄을 사용하여 VPC 삭제](#)

콘솔을 사용하여 VPC 삭제

Amazon VPC 콘솔을 사용하여 VPC를 삭제하면 다음과 같은 VPC 구성 요소도 삭제됩니다.

- DHCP 옵션
- 외부 전용 인터넷 게이트웨이
- 게이트웨이 엔드포인트
- 인터넷 게이트웨이
- 네트워크 ACL
- 라우팅 테이블
- 보안 그룹
- 서브넷

콘솔을 사용하여 VPC를 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. VPC에서 모든 인스턴스를 종료합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [인스턴스 종료](#)를 참조하세요.
3. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
4. 탐색 창에서 [Your VPCs]를 선택합니다.
5. 삭제할 VPC를 선택하고 Actions, Delete VPC를 선택합니다.
6. VPC를 삭제하기 전에 삭제하거나 종료해야 하는 리소스가 있는 경우 해당 리소스가 표시됩니다. 해당 리소스를 삭제하거나 종료한 다음 다시 시도하세요. 그렇지 않으면 VPC와 함께 삭제할 리소스가 표시됩니다. 목록을 검토한 후 다음 단계로 이동합니다.
7. (선택 사항) Site-to-Site VPN 연결이 있는 경우 삭제할 옵션을 선택할 수 있습니다. 다른 VPC에서 이 고객 게이트웨이를 사용할 계획이라면 Site-to-Site VPN 연결 및 게이트웨이를 유지하는 것이 좋습니다. 그렇지 않으면 새 Site-to-Site VPN 연결을 생성한 후 고객 게이트웨이 디바이스를 다시 구성해야 합니다.
8. 확인 메시지가 나타나면 **delete**을 입력한 다음 삭제를 선택합니다.

명령줄을 사용하여 VPC 삭제

명령줄을 사용하여 VPC를 삭제하려면 먼저 VPC에서 요청자 관리형 네트워크 인터페이스를 생성한 모든 리소스를 종료하거나 삭제해야 합니다. 또한 서브넷, 보안 그룹, 네트워크 ACL, 라우팅 테이블, 인

터넷 게이트웨이 및 송신 전용 인터넷 게이트웨이와 같은 생성한 모든 VPC 리소스를 삭제하거나 분리해야 합니다. 기본 보안 그룹, 기본 라우팅 테이블 또는 기본 네트워크 ACL은 삭제할 필요가 없습니다.

다음 절차는 일반적인 VPC 리소스를 삭제한 다음 VPC를 삭제하는 데 사용하는 명령을 보여줍니다. 이러한 명령을 이 순서대로 사용해야 합니다. VPC 리소스를 추가로 생성한 경우 VPC를 삭제하기 전에 해당하는 삭제 명령도 사용해야 합니다.

AWS CLI를 사용하여 VPC 삭제

1. [delete-security-group](#) 명령을 사용하여 보안 그룹을 삭제합니다.

```
aws ec2 delete-security-group --group-id sg-id
```

2. [delete-network-acl](#) 명령을 사용하여 각 네트워크 ACL을 삭제합니다.

```
aws ec2 delete-network-acl --network-acl-id acl-id
```

3. [delete-subnet](#) 명령을 사용하여 각 서브넷을 삭제합니다.

```
aws ec2 delete-subnet --subnet-id subnet-id
```

4. [delete-route-table](#) 명령을 사용하여 각 사용자 지정 라우팅 테이블을 삭제합니다.

```
aws ec2 delete-route-table --route-table-id rtb-id
```

5. [detach-internet-gateway](#) 명령을 사용하여 VPC에서 인터넷 게이트웨이를 분리합니다.

```
aws ec2 detach-internet-gateway --internet-gateway-id igw-id --vpc-id vpc-id
```

6. [delete-internet-gateway](#) 명령을 사용하여 인터넷 게이트웨이를 삭제합니다.

```
aws ec2 delete-internet-gateway --internet-gateway-id igw-id
```

7. [듀얼 스택 VPC] [delete-egress-only-internet-gateway](#) 명령을 사용하여 송신 전용 인터넷 게이트웨이를 삭제합니다.

```
aws ec2 delete-egress-only-internet-gateway --egress-only-internet-gateway-id eigw-id
```

8. [delete-vpc](#) 명령을 사용하여 VPC를 삭제합니다.


```
aws ec2 delete-vpc --vpc-id vpc-id
```

VPC의 서브넷

서브넷은 VPC의 IP 주소 범위입니다. 특정 서브넷에서 EC2 인스턴스와 같은 AWS 리소스를 생성할 수 있습니다.

내용

- [서브넷 기본 사항](#)
- [서브넷 보안](#)
- [서브넷 생성](#)
- [서브넷 구성](#)
- [서브넷 CIDR 예약](#)
- [라우팅 테이블 구성](#)
- [서브넷 삭제](#)

서브넷 기본 사항

각 서브넷은 단일 가용 영역 내에서만 존재해야 하며, 여러 영역으로 스케일 아웃할 수 없습니다. 별도의 가용 영역에서 AWS 리소스를 시작하면 단일 가용 영역의 장애로부터 애플리케이션을 보호할 수 있습니다.

내용

- [서브넷 IP 주소 범위](#)
- [서브넷 유형](#)
- [서브넷 다이어그램](#)
- [서브넷 라우팅](#)
- [서브넷 설정](#)

서브넷 IP 주소 범위

서브넷을 만들 때 VPC 구성에 따라 다음과 같이 IP 주소를 지정합니다.

- IPv4 전용 - 서브넷에 IPv4 CIDR 블록은 있지만 IPv6 CIDR 블록은 없습니다. IPv4 전용 서브넷의 리소스는 IPv4를 통해 통신해야 합니다.

- 듀얼 스택 - 서브넷에 IPv4 CIDR 블록 및 IPv6 CIDR 블록이 둘 다 있습니다. VPC에는 IPv4 CIDR 블록 및 IPv6 CIDR 블록이 둘 다 있어야 합니다. 듀얼 스택 서브넷의 리소스는 IPv4 및 IPv6를 통해 통신할 수 있습니다.
- IPv6 전용 - 서브넷에 IPv6 CIDR 블록은 있지만 IPv4 CIDR 블록은 없습니다. VPC에 IPv6 CIDR 블록이 있어야 합니다. IPv6 전용 서브넷의 리소스는 IPv6를 통해 통신해야 합니다.

Note

IPv6 전용 서브넷에 있는 리소스에는 CIDR 블록 169.254.0.0/16로부터 IPv4 [링크-로컬](#) 주소가 할당됩니다. 이러한 주소를 사용하여 [인스턴스 메타데이터 서비스\(IMDS\)](#)와 같은 VPC 서비스와 통신할 수 있습니다.

자세한 내용은 [VPC 및 서브넷의 IP 주소 지정](#) 단원을 참조하십시오.

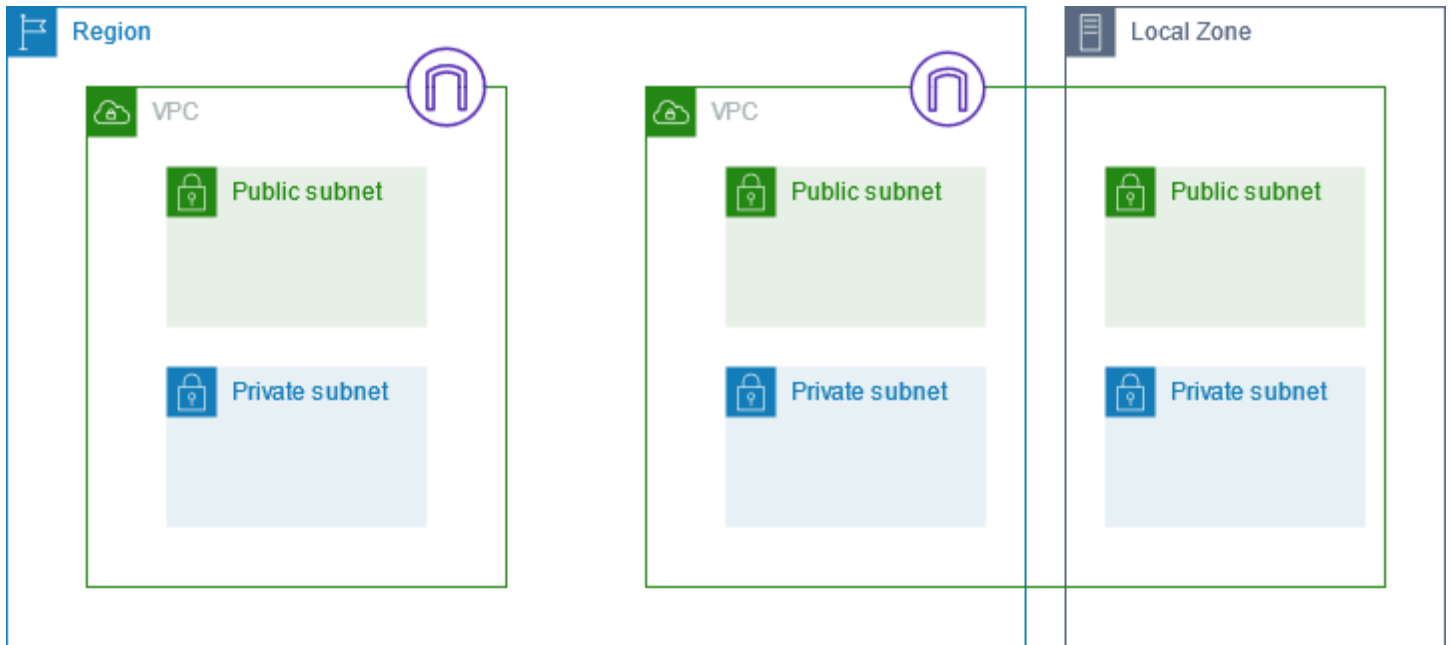
서브넷 유형

서브넷 유형은 서브넷에 대한 라우팅을 구성하는 방법에 따라 결정됩니다. 예:

- 퍼블릭 서브넷 - 서브넷에 [인터넷 게이트웨이](#)로 직접 연결되는 경로가 있습니다. 퍼블릭 서브넷의 리소스는 퍼블릭 인터넷에 액세스할 수 있습니다.
- 프라이빗 서브넷 - 서브넷에 인터넷 게이트웨이로 직접 연결되는 경로가 없습니다. 프라이빗 서브넷의 리소스에는 퍼블릭 인터넷에 액세스하기 위해 [NAT 디바이스](#)가 필요합니다.
- VPN 전용 서브넷 - 서브넷에 가상 프라이빗 게이트웨이를 통해 [Site-to-Site VPN 연결](#)으로 연결되는 경로가 있습니다. 서브넷에는 인터넷 게이트웨이에 대한 경로가 없습니다.
- 격리된 서브넷 - 서브넷에 VPC 외부 대상에 대한 경로가 없습니다. 격리된 서브넷의 리소스는 동일한 VPC의 다른 리소스와만 서로 액세스할 수 있습니다.

서브넷 다이어그램

다음 다이어그램에서는 한 리전에 있는 2개의 VPC를 보여줍니다. 각 VPC에는 퍼블릭 및 프라이빗 서브넷과 인터넷 게이트웨이가 있습니다. 다이어그램에 표시된 대로 필요에 따라 로컬 영역에 서브넷을 추가할 수 있습니다. 로컬 영역은 컴퓨팅, 스토리지 및 데이터베이스 서비스를 최종 사용자에게 더 가깝게 배치하는 AWS 인프라 배포입니다. 로컬 영역을 사용하면 최종 사용자가 10밀리초 미만의 지연 시간이 필요한 애플리케이션을 실행할 수 있습니다. 자세한 내용은 [AWS 로컬 영역](#)을 참조하세요.



서브넷 라우팅

각 서브넷은 서브넷 외부로 나가는 아웃바운드 트래픽에 대해 허용된 경로를 지정하는 라우팅 테이블이 연결되어 있어야 합니다. 생성된 각 서브넷은 자동으로 VPC의 기본 라우팅 테이블에 연결됩니다. 테이블 연결 및 기본 라우팅 테이블의 내용을 변경할 수 있습니다. 자세한 내용은 [라우팅 테이블 구성 단원](#)을 참조하십시오.

서브넷 설정

모든 서브넷은 해당 서브넷에서 생성된 네트워크 인터페이스에 퍼블릭 IPv4 주소(해당되는 경우, IPv6 주소)가 할당될 것인지 결정하는 수정 가능한 속성을 갖습니다. 여기에는 해당 서브넷에서 인스턴스를 시작할 때 인스턴스에 대해 생성된 주 네트워크 인터페이스(eth0)가 포함됩니다. 서브넷 속성에 상관없이 특정 인스턴스를 시작하는 중에 해당 인스턴스에 대한 이 설정을 재정의할 수 있습니다.

서브넷을 생성한 후 서브넷에 대한 다음 설정을 수정할 수 있습니다.

- IP 설정 자동 할당: 이 서브넷의 새 네트워크 인터페이스에 대한 퍼블릭 IPv4 또는 IPv6 주소를 자동으로 요청하도록 자동 할당 IP 설정을 구성할 수 있습니다.
- 리소스 기반 이름(RBN) 설정: 이 서브넷에서 EC2 인스턴스에 대한 호스트 이름 유형을 지정하고 DNS A 및 AAAA 레코드 쿼리가 처리되는 방법을 구성할 수 있습니다. 자세한 내용은 Amazon EC2 사용 설명서의 [Amazon EC2 인스턴스 호스트 이름 유형](#)을 참조하세요.

서브넷 보안

AWS 리소스를 보호하려면 프라이빗 서브넷을 사용하는 것이 좋습니다. Bastion Host 또는 NAT 디바이스를 사용하여 프라이빗 서브넷에 있는 EC2 인스턴스와 같은 리소스에 대한 인터넷 액세스를 제공합니다.

AWS는(는) VPC에서 리소스에 대한 보안을 강화하기 위해 사용할 수 있는 기능을 제공합니다. 보안 그룹은 EC2 인스턴스와 같은 관련 리소스에 대한 인바운드 및 아웃바운드 트래픽을 허용합니다. 네트워크 ACL은 사용하여 서브넷 수준에서 인바운드 및 아웃바운드 트래픽을 허용하거나 제어합니다. 대부분의 경우 보안 그룹은 사용자의 요구 사항을 충족할 수 있습니다. 단, 추가 보안 계층을 원하는 경우 네트워크 ACL을 사용할 수 있습니다. 자세한 내용은 [the section called “보안 그룹 및 네트워크 ACL 비교”](#) 단원을 참조하십시오.

각 서브넷에는 네트워크 ACL이 연결되어야 합니다. 생성하는 모든 서브넷은 VPC의 기본 네트워크 ACL과 자동으로 연결됩니다. 기본 네트워크 ACL은 인바운드와 아웃바운드 트래픽을 모두 허용합니다. 기본 네트워크 ACL을 업데이트하거나, 사용자 지정 네트워크 ACL을 생성하여 서브넷에 연결할 수 있습니다. 자세한 내용은 [네트워크 ACL을 사용하여 서브넷에 대한 트래픽 제어](#) 단원을 참조하십시오.

VPC 또는 서브넷에 흐름 로그를 만들어 VPC 또는 서브넷의 네트워크 인터페이스로 들어오고 나가는 모든 트래픽을 캡처할 수 있습니다. 또한 개별 네트워크 인터페이스에 흐름 로그를 만들 수도 있습니다. 자세한 내용은 [VPC 흐름 로그를 사용하여 IP 트래픽 로깅](#) 단원을 참조하십시오.

서브넷 생성

다음 절차에 따라 Virtual Private Cloud(VPC)에 대한 서브넷을 생성합니다. 필요한 연결에 따라 게이트웨이와 라우팅 테이블을 추가해야 할 수도 있습니다.

고려 사항

- VPC의 범위로부터 서브넷에 대해 IPv4 CIDR 블록을 지정해야 합니다. IPv6 CIDR 블록이 VPC와 연결되어 있는 경우 서브넷에 대해 IPv6 CIDR 블록을 지정할 수도 있습니다. 자세한 내용은 [VPC 및 서브넷의 IP 주소 지정](#) 단원을 참조하십시오.
- IPv6 전용 서브넷을 생성하는 경우 다음 사항에 유의합니다. IPv6 전용 서브넷에서 시작된 EC2 인스턴스는 IPv6 주소를 수신하지만 IPv4 주소는 수신하지 않습니다. IPv6 전용 서브넷으로 시작하는 모든 인스턴스는 [Nitro 시스템에 구축된 인스턴스](#)여야 합니다.
- 로컬 영역 또는 Wavelength 영역에 서브넷을 생성하려면 영역을 활성화해야 합니다. 자세한 내용은 Amazon EC2 사용 설명서에서 [리전 및 가용 영역](#)을 참조하세요.

VPC에 서브넷을 추가하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 서브넷(Subnets)을 선택합니다.
3. 서브넷 생성(Create subnet)을 선택합니다.
4. VPC ID에서 서브넷의 VPC를 선택합니다.
5. (선택 사항) 서브넷 이름(Subnet name)에 서브넷의 이름을 입력합니다. 이렇게 하면 Name 키와 지정한 값으로 태그가 생성됩니다.
6. 가용 영역(Availability Zone)에서 서브넷의 영역을 선택하거나 AWS에서 자동으로 선택하도록 기본값인 기본 설정 없음(No Preference)을 그대로 둘 수 있습니다.
7. IPv4 CIDR 블록의 경우 수동 입력을 선택하여 서브넷의 IPv4 CIDR 블록을 입력하거나(예: 10.0.1.0/24) IPv4 CIDR 없음을 선택합니다. Amazon VPC IP 주소 관리자(IPAM)를 사용하여 AWS 워크로드의 IP 주소를 계획, 추적 및 모니터링하는 경우 서브넷을 생성할 때 IPAM(IPAM 할당)에서 CIDR 블록을 할당할 수 있는 옵션이 제공됩니다. 서브넷 IP 할당을 위한 VPC IP 주소 공간 계획에 대한 자세한 내용은 Amazon VPC IPAM 사용 설명서의 [자습서: 서브넷 IP 할당을 위한 VPC IP 주소 공간 계획](#)을 참조하세요.
8. IPv6 CIDR 블록의 경우 수동 입력을 선택하여 서브넷을 생성할 VPC의 IPv6 CIDR을 선택합니다. 이 옵션은 VPC에 연결된 IPv6 CIDR 블록이 있는 경우에만 사용할 수 있습니다. Amazon VPC IP 주소 관리자(IPAM)를 사용하여 AWS 워크로드의 IP 주소를 계획, 추적 및 모니터링하는 경우 서브넷을 생성할 때 IPAM(IPAM 할당)에서 CIDR 블록을 할당할 수 있는 옵션이 제공됩니다. 서브넷 IP 할당을 위한 VPC IP 주소 공간 계획에 대한 자세한 내용은 Amazon VPC IPAM 사용 설명서의 [자습서: 서브넷 IP 할당을 위한 VPC IP 주소 공간 계획](#)을 참조하세요.
9. IPv6 VPC CIDR 블록을 선택합니다.
10. IPv6 서브넷 CIDR 블록의 경우 VPC CIDR과 같거나 더 구체적인 서브넷의 CIDR을 선택합니다. 예를 들어 VPC 풀 CIDR이 /50인 경우 서브넷의 넷마스크 길이를 /50에서 /64 사이로 선택할 수 있습니다. 가능한 IPv6 넷마스크 길이는 /44~/64(/4씩 증가)입니다.
11. 서브넷 생성(Create subnet)을 선택합니다.

AWS CLI를 사용하여 VPC에 서브넷을 추가하려면

[create-subnet](#) 명령을 사용합니다.

다음 단계

서브넷을 생성한 후 다음과 같이 구성할 수 있습니다.

- 라우팅 구성. 그런 다음 인터넷 게이트웨이와 같이 VPC와 연결된 게이트웨이로 트래픽을 전송하는 사용자 정의 라우팅 테이블과 라우팅을 생성할 수 있습니다. 자세한 내용은 [라우팅 테이블 구성](#) 단원을 참조하십시오.
- 서브넷 IP 주소를 수정합니다. 자세한 내용은 [the section called “서브넷 구성”](#) 단원을 참조하십시오.
- IP 주소 지정 동작 수정. 서브넷에서 시작하는 인스턴스가 퍼블릭 IPv4 주소, IPv6 주소 또는 둘 다 받도록 지정할 수 있습니다. 자세한 내용은 [서브넷 설정](#) 단원을 참조하십시오.
- RBN(리소스 기반 이름) 설정을 수정합니다. 자세한 내용은 [Amazon EC2 인스턴스 호스트 이름 유형](#)을 참조하십시오.
- 네트워크 ACL을 생성하거나 수정합니다. 자세한 내용은 [네트워크 ACL을 사용하여 서브넷에 대한 트래픽 제어](#) 단원을 참조하십시오.
- 다른 계정과 서브넷을 공유합니다. 자세한 내용은 [???](#) 단원을 참조하십시오.

서브넷 구성

다음 절차에 따라 Virtual Private Cloud(VPC)에 대한 서브넷을 구성합니다.

Tasks

- [서브넷 보기](#)
- [서브넷에 IPv6 CIDR 블록 추가](#)
- [서브넷에서 IPv6 CIDR 블록 제거](#)
- [서브넷의 퍼블릭 IPv4 주소 지정 속성 수정](#)
- [서브넷의 IPv6 주소 지정 속성 수정](#)

서브넷 보기

다음 단계 섹션에 따라 서브넷에 대한 세부 정보를 봅니다.

콘솔을 사용하여 서브넷 세부 정보를 보려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 서브넷(Subnets)을 선택합니다.
3. 서브넷의 확인란을 선택하거나 서브넷 ID를 선택하여 세부 정보 페이지를 엽니다.

AWS CLI를 사용하여 서브넷을 설명하려면

[describe-subnets](#) 명령을 사용합니다.

모든 리전의 서브넷을 보려면

<https://console.aws.amazon.com/ec2globalview/home>에서 Amazon EC2 Global View 콘솔을 엽니다. 자세한 내용은 Amazon EC2 사용 설명서의 [Amazon EC2 Global View를 사용하여 리소스 나열 및 필터링](#)을 참조하세요.

서브넷에 IPv6 CIDR 블록 추가

IPv6 CIDR 블록을 VPC의 기존 서브넷에 연결할 수 있습니다. 서브넷에는 이와 연결된 기존 IPv6 CIDR 블록이 있어서는 안 됩니다.

서브넷에 IPv6 CIDR 블록을 추가하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 서브넷(Subnets)을 선택합니다.
3. 서브넷을 선택하고 작업(Actions), IPv6 CIDR 편집(Edit IPv6 CIDRs)을 선택합니다.
4. Add IPv6 CIDR을 선택합니다.
5. VPC CIDR 블록을 선택하고 서브넷 CIDR 블록을 입력한 다음 VPC CIDR의 넷마스크 길이와 같거나 더 구체적인 넷마스크 길이를 선택합니다. 예를 들어 VPC 풀 CIDR이 /50인 경우 서브넷의 넷마스크 길이를 /50에서 /64 사이로 선택할 수 있습니다. 가능한 IPv6 넷마스크 길이는 /44~/64(/4씩 증가)입니다.
6. Save(저장)를 선택합니다.

AWS CLI를 사용하여 IPv6 CIDR 블록을 서브넷에 연결하려면

[associate-subnet-cidr-block](#) 명령을 사용합니다.

서브넷에서 IPv6 CIDR 블록 제거

서브넷에서 IPv6 지원이 더 이상 필요 없지만 IPv4 리소스 생성 및 IPv4 리소스와의 통신을 위해 서브넷을 계속 사용하려는 경우 IPv6 CIDR 블록을 제거할 수 있습니다.

IPv6 CIDR 블록을 제거하려면 먼저 서브넷의 모든 인스턴스에 할당된 모든 IPv6 주소를 할당 해제해야 합니다.

서브넷에서 IPv6 CIDR 블록을 제거하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.

2. 탐색 창에서 서브넷(Subnets)을 선택합니다.
3. 서브넷을 선택하고 작업(Actions), IPv6 CIDR 편집(Edit IPv6 CIDRs)을 선택합니다.
4. IPv6 CIDR 블록을 찾고 제거(Remove)를 선택합니다.
5. Save(저장)를 선택합니다.

AWS CLI를 사용하여 서브넷에서 IPv6 CIDR 블록을 연결 해제하려면

[disassociate-subnet-cidr-block](#) 명령을 사용합니다.

서브넷의 퍼블릭 IPv4 주소 지정 속성 수정

기본이 아닌 서브넷은 IPv4 퍼블릭 주소 지정 속성이 `false`로 기본 설정되어 있고, 기본 서브넷은 이 속성이 `true`로 기본 설정되어 있습니다. 단 Amazon EC2 인스턴스 실행 마법사에서 생성되는 기본이 아닌 서브넷은 예외로, 마법사는 이 속성을 `true`로 설정합니다. Amazon VPC 콘솔을 사용하여 이 속성을 수정할 수 있습니다.

서브넷의 퍼블릭 IPv4 주소 지정 동작을 수정하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 서브넷(Subnets)을 선택합니다.
3. 서브넷을 선택하고 작업(Actions), 서브넷 설정 편집(Edit subnet settings)을 선택합니다.
4. 퍼블릭 IPv4 주소 자동 할당 활성화(Enable auto-assign public IPv4 address) 확인란을 선택하면 선택된 서브넷에서 시작된 모든 인스턴스에 대한 퍼블릭 IPv4 주소를 요청합니다. 필요에 따라 확인란을 선택하거나 선택 취소한 후 Save를 선택합니다.

AWS CLI를 사용하여 서브넷 속성을 수정하려면

[modify-subnet-attribute](#) 명령을 사용합니다.

서브넷의 IPv6 주소 지정 속성 수정

모든 서브넷에는 IPv6 주소 지정 속성이 `false`로 기본 설정되어 있습니다. Amazon VPC 콘솔을 사용하여 이 속성을 수정할 수 있습니다. 서브넷에서 IPv6 주소 지정 속성을 사용하는 경우, 해당 서브넷에서 생성된 네트워크 인터페이스는 서브넷의 범위에 속하는 IPv6 주소를 받습니다. 서브넷에서 시작한 인스턴스는 주 네트워크 인터페이스에서 IPv6 주소를 받습니다.

서브넷에는 연결된 IPv6 CIDR 블록이 있어야 합니다.

Note

서브넷에서 IPv6 주소 지정 기능을 사용하는 경우, 네트워크 인터페이스 또는 인스턴스는 Amazon EC2 API 2016-11-15 이상 버전에서 생성된 경우에만 IPv6 주소를 받습니다. Amazon EC2 API 콘솔에서는 최신 API 버전을 사용합니다.

서브넷의 IPv6 주소 지정 동작을 수정하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 서브넷(Subnets)을 선택합니다.
3. 서브넷을 선택하고 Actions(작업), Edit subnet settings(서브넷 설정 편집)을 선택합니다.
4. 퍼블릭 IPv4 주소 자동 할당 활성화(Enable auto-assign IPv6 address) 확인란을 선택하면 선택된 서브넷에서 생성된 모든 네트워크 인터페이스에 대한 IPv6 주소를 요청합니다. 필요에 따라 확인란을 선택하거나 선택 취소한 후 Save를 선택합니다.

AWS CLI를 사용하여 서브넷 속성을 수정하려면

[modify-subnet-attribute](#) 명령을 사용합니다.

서브넷 CIDR 예약

서브넷 CIDR 예약은 AWS에서 네트워크 인터페이스에 할당할 수 없도록 따로 설정한 IPv4 또는 IPv6 주소 범위입니다. 이렇게 하면 네트워크 인터페이스에 사용할 IPv4 또는 IPv6 CIDR 블록('접두사'라고도 함)을 예약할 수 있습니다.

서브넷 CIDR 예약을 생성할 때 예약된 ID 주소를 사용하는 방법을 지정합니다. 다음과 같은 옵션을 사용할 수 있습니다.

- 접두사—AWS에서 예약된 IP 주소 범위의 주소를 네트워크 인터페이스에 할당합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [Amazon EC2 네트워크 인터페이스에 접두사 할당](#)을 참조하세요.
- 명시적—네트워크 인터페이스에 IP 주소를 수동으로 할당합니다.

다음 규칙이 서브넷 CIDR 예약에 적용됩니다.

- 서브넷 CIDR 예약을 생성할 때 IP 주소 범위에는 이미 사용 중인 주소가 포함될 수 있습니다. 서브넷 예약을 생성해도 이미 사용 중인 IP 주소는 할당 해제되지 않습니다.

- 서브넷당 여러 CIDR 범위를 예약할 수 있습니다. 동일한 VPC 내에 여러 CIDR 범위를 예약하면 CIDR 범위가 겹쳐지지 않습니다.
- 접두사 위임을 위해 서브넷에서 2개 이상의 범위를 예약하고 접두사 위임을 자동 할당하도록 구성한 경우 네트워크 인터페이스에 할당할 IP 주소를 임의로 선택합니다.
- 서브넷 예약을 삭제하면 사용하지 않은 IP 주소를 AWS에서 네트워크 인터페이스에 할당할 수 있습니다. 서브넷 예약을 삭제해도 사용 중인 IP 주소는 할당 해제되지 않습니다.

Classless Inter-Domain Routing(CIDR) 표기법에 대한 자세한 내용은 [IP 주소 지정](#) 섹션을 참조하세요.

콘솔을 사용한 서브넷 CIDR 예약 작업

다음과 같이 서브넷 CIDR 예약을 만들고 관리할 수 있습니다.

서브넷 CIDR 예약 편집

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 서브넷(Subnets)을 선택합니다.
3. 서브넷을 선택합니다.
4. 기존 서브넷 CIDR 예약에 대한 정보를 얻으려면 CIDR 예약 탭을 선택합니다.
5. 서브넷 CIDR 예약을 추가하거나 제거하려면 작업, CIDR 예약 편집을 선택한 후 다음을 수행합니다.
 - IPv4 CIDR 예약을 추가하려면 IPv4, IPv4 CIDR 예약 추가를 선택합니다. 예약 유형을 선택하고 CIDR 범위를 입력한 다음 추가를 선택합니다.
 - IPv6 CIDR 예약을 추가하려면 IPv6, IPv6 CIDR 예약 추가를 선택합니다. 예약 유형을 선택하고 CIDR 범위를 입력한 다음 추가(Add)를 선택합니다.
 - CIDR 예약을 제거하려면 서브넷 CIDR 예약에 대해 제거를 선택합니다.

AWS CLI를 사용한 서브넷 CIDR 예약 작업

AWS CLI를 사용하여 서브넷 CIDR 예약을 만들고 관리할 수 있습니다.

Tasks

- [서브넷 CIDR 예약 만들기](#)
- [서브넷 CIDR 예약 보기](#)
- [서브넷 CIDR 예약 삭제](#)

서브넷 CIDR 예약 만들기

[create-subnet-cidr-reservation](#)을 사용하여 서브넷 CIDR 예약을 생성합니다.

```
aws ec2 create-subnet-cidr-reservation --subnet-id subnet-03c51e2eEXAMPLE --
reservation-type prefix --cidr 2600:1f13:925:d240:3a1b::/80
```

출력의 예제는 다음과 같습니다.

```
{
  "SubnetCidrReservation": {
    "SubnetCidrReservationId": "scr-044f977c4eEXAMPLE",
    "SubnetId": "subnet-03c51e2ef5EXAMPLE",
    "Cidr": "2600:1f13:925:d240:3a1b::/80",
    "ReservationType": "prefix",
    "OwnerId": "123456789012"
  }
}
```

서브넷 CIDR 예약 보기

[get-subnet-cidr-reservations](#)을 사용하여 서브넷 CIDR 예약의 세부 정보를 볼 수 있습니다.

```
aws ec2 get-subnet-cidr-reservations --subnet-id subnet-05eef9fb78EXAMPLE
```

서브넷 CIDR 예약 삭제

[delete-subnet-cidr-reservation](#)을 사용하여 서브넷 CIDR 예약을 삭제합니다.

```
aws ec2 delete-subnet-cidr-reservation --subnet-cidr-reservation-
id scr-044f977c4eEXAMPLE
```

라우팅 테이블 구성

라우팅 테이블에는 서브넷 또는 게이트웨이의 네트워크 트래픽이 전송되는 위치를 결정하는 라우팅이라는 규칙 세트가 포함되어 있습니다.

내용

- [라우팅 테이블 개념](#)

- [서브넷 라우팅 테이블](#)
- [게이트웨이 라우팅 테이블](#)
- [라우팅 우선순위](#)
- [라우팅 테이블 할당량](#)
- [연결 문제 해결](#)
- [라우팅 옵션 예](#)
- [라우팅 테이블 작업](#)
- [미들박스 라우팅 마법사](#)

라우팅 테이블 개념

다음은 라우팅 테이블의 주요 개념입니다.

- 기본 라우팅 테이블 — VPC와 함께 자동으로 제공되는 라우팅 테이블입니다. 다른 라우팅 테이블과 명시적으로 연결되지 않은 모든 서브넷의 라우팅을 제어합니다.
- 사용자 지정 라우팅 테이블(Custom route table) - VPC에 대해 생성하는 라우팅 테이블입니다.
- 대상(Destination) - 트래픽을 이동할 대상 IP 주소(대상 CIDR)의 범위입니다. 예를 들어, CIDR 172.16.0.0/12가 있는 외부 회사 네트워크입니다.
- 대상(Target) - 대상 트래픽을 전송할 때 사용할 게이트웨이, 네트워크 인터페이스 또는 연결입니다 (예: 인터넷 게이트웨이).
- 라우팅 테이블 연결(Route table association) - 라우팅 테이블과 서브넷, 인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이 간의 연결입니다.
- 서브넷 라우팅 테이블(Subnet route table) - 서브넷과 연결된 라우팅 테이블입니다.
- 로컬 라우팅(Local route) - VPC 내 통신을 위한 기본 라우팅입니다.
- 전파 - 가상 프라이빗 게이트웨이를 VPC에 연결하고 라우팅 전파를 활성화한 경우 VPN 연결을 위한 경로를 서브넷 라우팅 테이블에 자동으로 추가합니다. 따라서 VPN 경로를 수동으로 추가하거나 제거할 필요가 없습니다. 자세한 내용은 Site-to-Site VPN 사용 설명서의 [Site-to-Site VPN 라우팅 옵션](#)을 참조하세요.
- 게이트웨이 라우팅 테이블(Gateway route table) - 인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이와 연결된 라우팅 테이블입니다.
- 엣지 연결(Edge association) - 인바운드 VPC 트래픽을 어플라이언스로 라우팅하는 데 사용하는 라우팅 테이블입니다. 라우팅 테이블을 인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이와 연결하고 어플라이언스의 네트워크 인터페이스를 VPC 트래픽의 대상으로 지정합니다.

- Transit 게이트웨이 라우팅 테이블(Transit gateway route table) - Transit 게이트웨이와 연결된 라우팅 테이블입니다. 자세한 내용은 Amazon VPC Transit Gateways의 [Transit Gateway 라우팅 테이블](#)을 참조하세요.
- 로컬 게이트웨이 라우팅 테이블(Local gateway route table) - Outposts 로컬 게이트웨이와 연결된 라우팅 테이블입니다. 자세한 내용은 AWS Outposts 사용 설명서의 [로컬 게이트웨이](#)를 참조하십시오.

서브넷 라우팅 테이블

VPC에는 암시적 라우터가 있으며 라우팅 테이블을 사용하여 네트워크 트래픽이 전달되는 위치를 제어합니다. VPC의 각 서브넷을 라우팅 테이블에 연결해야 합니다. 테이블에서는 서브넷에 대한 라우팅을 제어합니다(서브넷 라우팅 테이블). 서브넷을 특정 라우팅 테이블과 명시적으로 연결할 수 있습니다. 그렇지 않으면 서브넷이 기본 라우팅 테이블과 암시적으로 연결됩니다. 서브넷은 한 번에 하나의 라우팅 테이블에만 연결할 수 있지만 여러 서브넷을 동일한 서브넷 라우팅 테이블에 연결할 수 있습니다.

내용

- [경로](#)
- [기본 라우팅 테이블](#)
- [사용자 지정 라우팅 테이블](#)
- [서브넷 라우팅 테이블 연결](#)

경로

테이블의 각 라우팅은 목적지 및 대상을 지정합니다. 예를 들어 서브넷이 인터넷 게이트웨이를 통해 인터넷에 액세스할 수 있도록 하려면 서브넷 라우팅 테이블에 다음 라우팅을 추가합니다. 라우팅의 대상은 모든 IPv4 주소를 나타내는 `0.0.0.0/0`입니다. 대상은 VPC에 연결된 인터넷 게이트웨이입니다.

대상 주소	대상
0.0.0.0/0	<i>igw-id</i>

IPv4 및 IPv6 CIDR 블록은 별도로 취급됩니다. 예를 들어 대상 CIDR이 `0.0.0.0/0`인 라우팅에는 모든 IPv6 주소가 자동으로 포함되지 않습니다. 모든 IPv6 주소에 대해 대상 CIDR이 `::/0`인 라우팅을 생성해야 합니다.

AWS 리소스 전체에서 동일한 CIDR 블록 세트를 자주 참조하는 경우, [고객 관리형 접두사 목록](#)을 생성하여 함께 그룹화할 수 있습니다. 그런 다음 라우팅 테이블 항목의 대상으로 접두사 목록을 지정할 수 있습니다.

모든 라우팅 테이블에는 VPC 내부 통신을 위한 로컬 라우팅이 포함되어 있습니다. 이 라우팅은 기본적으로 모든 라우팅 테이블에 추가됩니다. VPC에 하나 이상의 IPv4 CIDR 블록이 연결되어 있는 경우, 라우팅 테이블에 각 IPv4 CIDR 블록의 로컬 경로가 포함됩니다. IPv6 CIDR 블록을 VPC와 연결한 경우, 라우팅 테이블에 IPv6 CIDR 블록의 로컬 경로가 포함됩니다. 필요에 따라 각 로컬 라우팅의 대상을 [교체하거나 복원](#)할 수 있습니다.

규칙 및 고려 사항

- 로컬 경로보다 더 구체적인 경로를 라우팅 테이블에 추가할 수 있습니다. 대상은 VPC에 있는 서브넷의 전체 IPv4 또는 IPv6 CIDR 블록과 일치해야 합니다. 대상은 NAT 게이트웨이, 네트워크 인터페이스 또는 게이트웨이 Load Balancer 엔드포인트여야 합니다.
- 라우팅 테이블에 라우팅이 여러 개 있는 경우 트래픽과 일치하는 가장 구체적인 라우팅(가장 긴 접두사 일치)을 사용하여 트래픽의 라우팅 방법을 결정합니다.
- 정확히 일치하거나 169.254.168.0/22 범위의 하위 세트인 IPv4 주소에는 경로를 추가할 수 없습니다. 이 범위는 링크 로컬 주소 공간 내에 있으며 AWS 서비스에서 사용하도록 예약되어 있습니다. 예를 들어 Amazon EC2는 인스턴스 메타데이터 서비스(IMDS) 및 Amazon DNS 서버와 같이 EC2 인스턴스에서만 액세스할 수 있는 서비스에 대해 이 범위의 주소를 사용합니다. 169.254.168.0/22보다 크지만 겹치는 CIDR 블록을 사용할 수 있지만 169.254.168.0/22의 주소로 향하는 패킷은 전달되지 않습니다.
- 정확히 일치하거나 fd00:ec2::/32 범위의 하위 세트인 IPv6 주소에는 경로를 추가할 수 없습니다. 이 범위는 고유 로컬 주소(ULA) 공간 내에 있으며 AWS 서비스에서 사용하도록 예약되어 있습니다. 예를 들어 Amazon EC2는 인스턴스 메타데이터 서비스(IMDS) 및 Amazon DNS 서버와 같이 EC2 인스턴스에서만 액세스할 수 있는 서비스에 대해 이 범위의 주소를 사용합니다. fd00:ec2::/32보다 크지만 겹치는 CIDR 블록을 사용할 수 있지만 fd00:ec2::/32의 주소로 향하는 패킷은 전달되지 않습니다.
- VPC에 대한 라우팅 경로에 미들박스 어플라이언스를 추가할 수 있습니다. 자세한 내용은 [the section called “미들박스 어플라이언스에 대한 라우팅”](#) 단원을 참조하십시오.

예

다음 예제에서 VPC에는 IPv4 CIDR 블록 및 IPv6 CIDR 블록이 모두 있다고 가정합니다. IPv4 및 IPv6 트래픽은 다음 라우팅 테이블에 표시된 것처럼 별도로 처리됩니다.

대상 주소	대상
10.0.0.0/16	Local
2001:db8:1234:1a00::/56	Local
172.31.0.0/16	pcx-11223344556677889
0.0.0.0/0	igw-12345678901234567
::/0	eigw-aabbccdde1122334

- VPC(10.0.0.0/16) 내에서 라우팅되는 IPv4 트래픽은 Local 경로로 처리됩니다.
- VPC(2001:db8:1234:1a00::/56) 내에서 라우팅되는 IPv6 트래픽은 Local 경로로 처리됩니다.
- 172.31.0.0/16의 경로는 피어링 연결로 트래픽을 전송합니다.
- 모든 IPv4 트래픽(0.0.0.0/0)의 경로는 인터넷 게이트웨이로 트래픽을 전송합니다. 따라서 VPC 내 그
리고 피어링 연결을 통한 트래픽을 제외한 모든 IPv4 트래픽은 인터넷 게이트웨이로 라우팅됩니다.
- 모든 IPv6 트래픽(::/0)의 경로는 외부 전용 인터넷 게이트웨이로 트래픽을 전송합니다. 따라서 VPC
내 트래픽을 제외한 모든 IPv6 트래픽은 외부 전용 인터넷 게이트웨이로 라우팅됩니다.

기본 라우팅 테이블

VPC를 만들면 기본 라우팅 테이블이 자동으로 생성됩니다. 서브넷이 라우팅 테이블과 명시적으로 연결되지 않은 경우 서브넷은 기본 라우팅 테이블이 기본적으로 사용됩니다. Amazon VPC 콘솔에서 라우팅 테이블(Route tables) 페이지의 기본(Main) 열에서 예(Yes)를 찾으면 VPC에 대한 기본 라우팅 테이블을 볼 수 있습니다.

기본적으로 기본이 아닌 VPC를 만들면 기본 라우팅 테이블에는 로컬 라우팅만 포함됩니다. 만약 NAT 게이트웨이를 [VPC 생성](#) 및 선택하면 Amazon VPC가 게이트웨이의 기본 라우팅 테이블에 경로를 자동으로 추가합니다.

기본 라우팅 테이블에는 다음 규칙이 적용됩니다.

- 기본 라우팅 테이블에서 라우팅을 추가 및 제거하고 수정할 수 있습니다.
- 기본 라우팅 테이블은 삭제할 수 없습니다.
- 게이트웨이 라우팅 테이블은 기본 라우팅 테이블로 설정할 수 없습니다.

- 사용자 지정 라우팅 테이블을 서브넷에 연결하여 기본 라우팅 테이블을 대체할 수 있습니다.
- 서브넷이 기본 라우팅 테이블에 명시적으로 연결되어 있지 않을 경우에도 서브넷을 기본 라우팅에 명시적으로 연결할 수 있습니다.

기본 라우팅 테이블을 변경하면 이를 수행할 수 있습니다. 기본 라우팅 테이블을 변경하면 추가되는 새로운 서브넷 또는 다른 라우팅 테이블에 명시적으로 연결되지 않은 서브넷의 기본값도 변경됩니다. 자세한 내용은 [기본 라우팅 테이블 교체](#) 단원을 참조하십시오.

사용자 지정 라우팅 테이블

기본적으로 라우팅 테이블에는 VPC 내부 통신을 위한 로컬 라우팅이 포함되어 있습니다. 만약 퍼블릭 서브넷을 [VPC 생성](#) 및 선택하면 Amazon VPC는 사용자 지정 라우팅 테이블을 만들고 인터넷 게이트웨이를 가리키는 경로를 추가합니다. VPC를 보호하는 한 가지 방법은 기본 라우팅 테이블을 원래 기본 상태로 두는 것입니다. 그런 다음 생성한 각 새 서브넷을 생성한 사용자 지정 라우팅 테이블 중 하나에 명시적으로 연결합니다. 이렇게 하면 각 서브넷이 트래픽으로 어떻게 라우팅되는지를 명시적으로 제어할 수 있습니다.

기본 라우팅 테이블에서 라우팅을 추가 및 제거하고 수정할 수 있습니다. 사용자 지정 라우팅 테이블에 연결이 없는 경우에만 삭제할 수 있습니다.

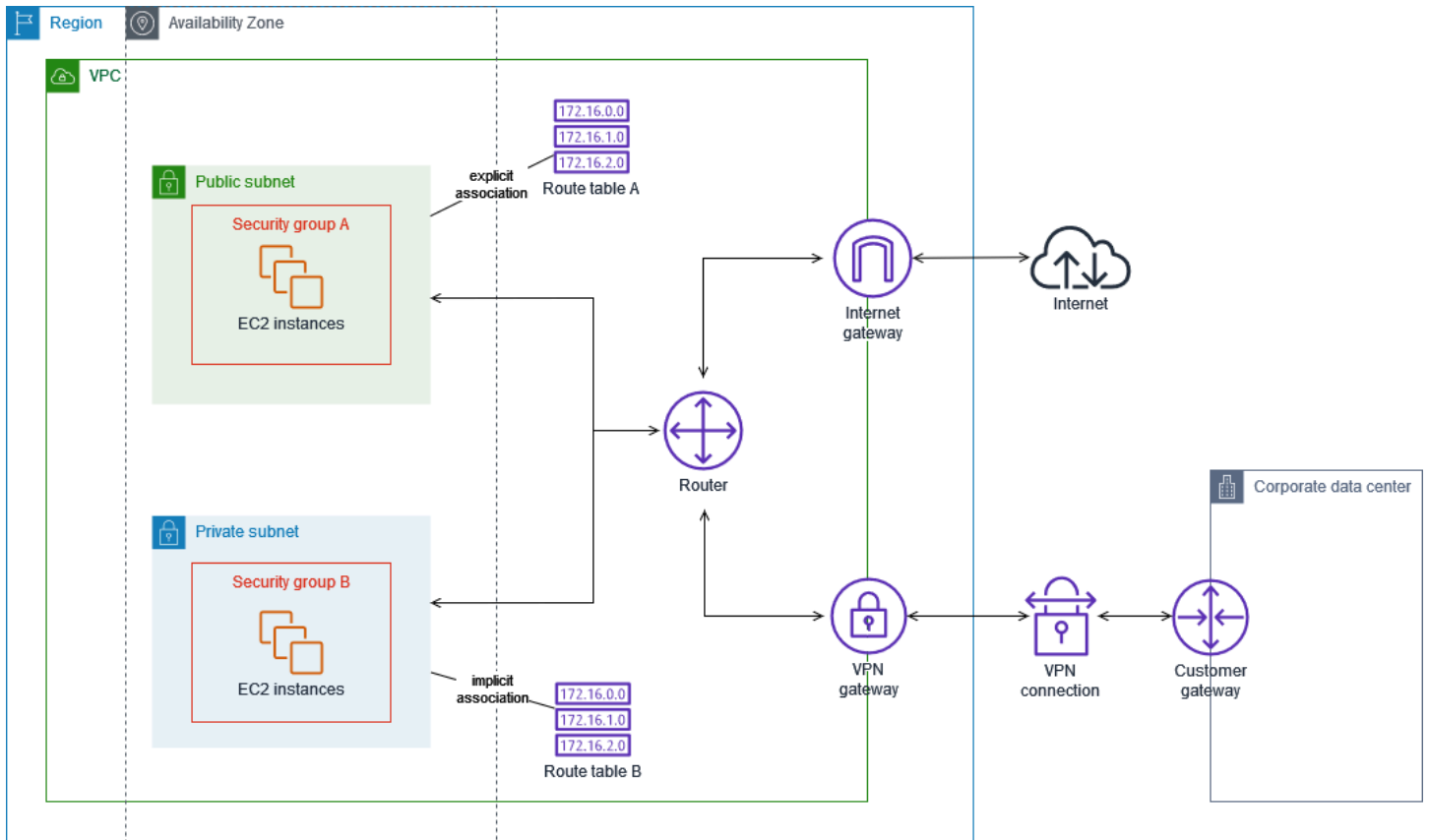
서브넷 라우팅 테이블 연결

VPC에 있는 각 서브넷을 라우팅 테이블과 연결해야 합니다. 서브넷은 사용자 지정 라우팅 테이블과 명시적으로 연결되거나 기본 라우팅 테이블과 암시적 또는 명시적으로 연결될 수 있습니다. 서브넷 및 라우팅 테이블 연결 보기에 대한 자세한 내용은 [명시적으로 연결되어 있는 서브넷 또는 게이트웨이 확인](#)을 참조하십시오.

Outposts와 연결된 VPC에 있는 서브넷은 로컬 게이트웨이의 추가 대상 유형을 가질 수 있습니다. 이 점이 Outposts가 아닌 서브넷과의 유일한 라우팅 차이점입니다.

예제 1: 암시적/명시적 서브넷 연결

다음 다이어그램에서는 인터넷 게이트웨이, 가상 프라이빗 게이트웨이, 퍼블릭 서브넷 및 VPN 전용 서브넷이 있는 VPC의 라우팅을 보여줍니다.



라우팅 테이블 A는 퍼블릭 서브넷과 명시적으로 연결되는 사용자 지정 라우팅 테이블입니다. 이 서브넷은 모든 트래픽을 인터넷 게이트웨이로 보내는 경로가 있기 때문에 퍼블릭 서브넷이 됩니다.

대상 주소	대상
<i>VPC CIDR</i>	로컬
0.0.0.0/0	<i>igw-id</i>

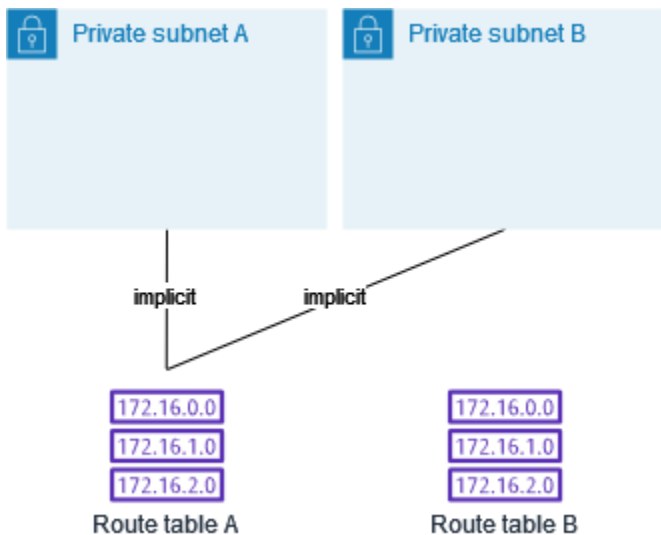
라우팅 테이블 B는 기본 라우팅 테이블입니다. 프라이빗 서브넷과는 묵시적으로 연결됩니다. 이 서브넷은 모든 트래픽을 가상 프라이빗 게이트웨이로 보내는 경로는 있지만 인터넷 게이트웨이로 보내는 경로는 없기 때문에 VPN 전용 서브넷이 됩니다. 이 VPC에 다른 서브넷을 만들고 사용자 지정 라우팅 테이블에 연결하지 않으면, 서브넷은 기본 라우팅 테이블인 이 라우팅 테이블과도 묵시적으로 연결됩니다.

대상 주소	대상
<i>VPC CIDR</i>	로컬
0.0.0.0/0	<i>vgw-id</i>

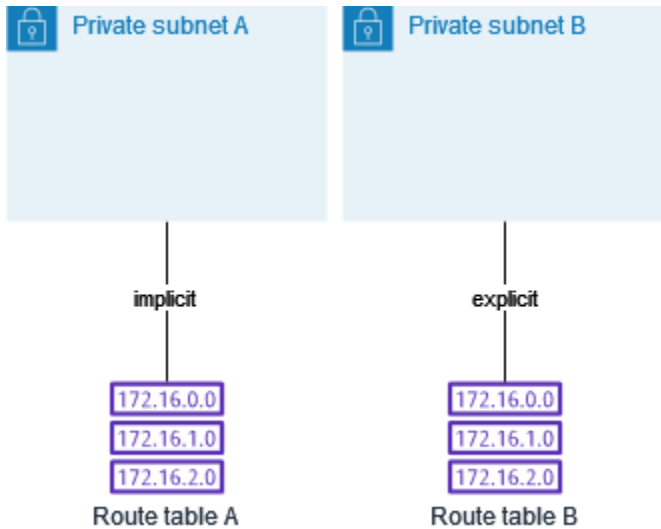
예제 2: 기본 라우팅 테이블 바꾸기

기본 라우팅 테이블을 변경할 수 있습니다. 트래픽 중단을 방지하려면 먼저 사용자 지정 라우팅 테이블을 사용하여 라우팅 변경을 테스트하는 것이 좋습니다. 테스트 결과에 만족하면 기본 라우팅 테이블을 새로운 사용자 지정 테이블로 바꿉니다.

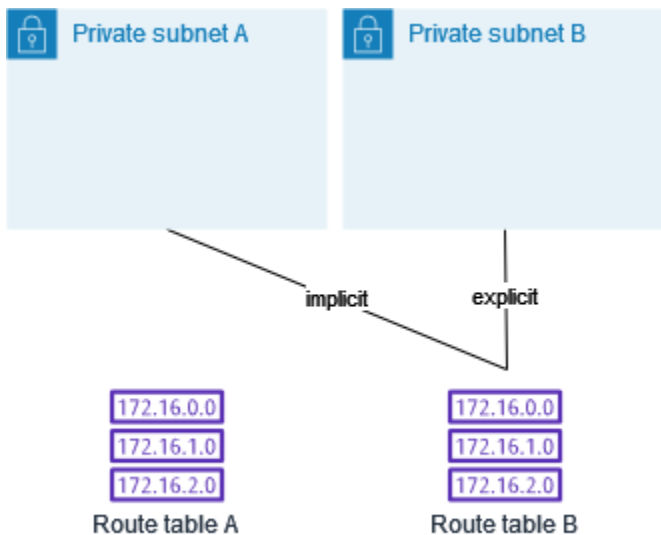
다음 구성도는 서브넷 2개와 라우팅 테이블 2개를 보여 줍니다. 서브넷 A는 기본 라우팅 테이블인 라우팅 테이블 A와 묵시적으로 연결되어 있습니다. 서브넷 B는 라우팅 테이블 A와 묵시적으로 연결되어 있습니다. 사용자 지정 라우팅 테이블인 라우팅 테이블 B는 어느 서브넷과도 연결되어 있지 않습니다.



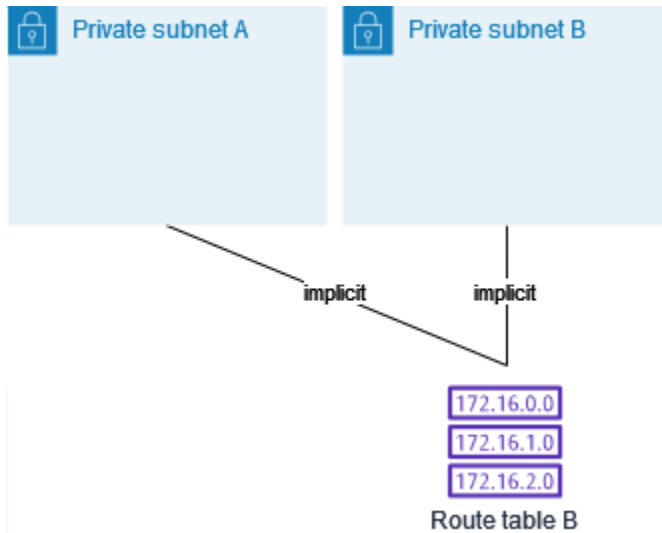
기본 라우팅 테이블을 교체하려면 먼저 서브넷 B와 라우팅 테이블 B 간에 명시적 연결을 만듭니다. 라우팅 테이블 B를 테스트합니다.



라우팅 테이블 B를 테스트한 후, 이 테이블을 기본 라우팅 테이블로 만듭니다. 서브넷 B는 여전히 라우팅 테이블 B와 명시적으로 연결되어 있습니다. 단, 라우팅 테이블 B가 이제 새로운 기본 라우팅 테이블이기 때문에 서브넷 A는 라우팅 테이블 B와 묵시적으로 연결됩니다. 라우팅 테이블 A는 더 이상 어느 서브넷과도 연결되지 않습니다.



(선택) 서브넷 B를 라우팅 테이블 B와의 연결에서 해제해도 서브넷 B와 라우팅 테이블 B의 묵시적 연결은 유지됩니다. 라우팅 테이블 A가 더 이상 필요하지 않은 경우 삭제할 수 있습니다.



게이트웨이 라우팅 테이블

라우팅 테이블을 인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이와 연결할 수 있습니다. 라우팅 테이블이 게이트웨이와 연결된 경우 이를 게이트웨이 라우팅 테이블이라고 합니다. VPC로 들어오는 트래픽의 라우팅 경로를 세밀하게 제어할 수 있는 게이트웨이 라우팅 테이블을 생성할 수 있습니다. 예를 들어 인터넷 게이트웨이를 통해 VPC로 들어오는 트래픽을 VPC의 미들박스 어플라이언스(예: 보안 어플라이언스)로 리디렉션하여 가로챌 수 있습니다.

내용

- [게이트웨이 라우팅 테이블 라우팅](#)
- [규칙 및 고려 사항](#)

게이트웨이 라우팅 테이블 라우팅

인터넷 게이트웨이와 연결된 게이트웨이 라우팅 테이블은 다음 대상을 가진 경로를 지원합니다.

- 기본 로컬 경로
- [Gateway Load Balancer 엔드포인트](#)
- 미들박스 어플라이언스에 대한 네트워크 인터페이스

가상 프라이빗 게이트웨이와 연결된 게이트웨이 라우팅 테이블은 다음 대상을 가진 경로를 지원합니다.

- 기본 로컬 경로

- [Gateway Load Balancer 엔드포인트](#)
- 미들박스 어플라이언스에 대한 네트워크 인터페이스

대상이 Gateway Load Balancer 엔드포인트 또는 네트워크 인터페이스인 경우 다음 대상이 허용됩니다.

- VPC의 전체 IPv4 또는 IPv6 CIDR 블록. 이 경우 기본 로컬 라우팅의 대상을 대체합니다.
- VPC에 있는 서브넷의 전체 IPv4 또는 IPv6 CIDR 블록입니다. 이는 기본 로컬 라우팅보다 더 구체적인 라우팅입니다.

게이트웨이 라우팅 테이블에 있는 로컬 라우팅의 대상을 VPC의 네트워크 인터페이스로 변경하면 나중에 기본 local 대상으로 복원할 수 있습니다. 자세한 내용은 [로컬 경로의 대상 교체 또는 복원](#) 단원을 참조하십시오.

예

다음 게이트웨이 라우팅 테이블에서 172.31.0.0/20 CIDR 블록이 있는 서브넷으로 향하는 트래픽은 특정 네트워크 인터페이스로 라우팅됩니다. VPC의 다른 모든 서브넷으로 향하는 트래픽은 로컬 라우팅을 사용합니다.

대상 주소	대상
172.31.0.0/16	로컬
172.31.0.0/20	<i>eni-id</i>

예

다음 게이트웨이 라우팅 테이블에서 로컬 라우팅의 대상이 네트워크 인터페이스 ID로 대체됩니다. VPC 내의 모든 서브넷으로 향하는 트래픽은 네트워크 인터페이스로 라우팅됩니다.

대상 주소	대상
172.31.0.0/16	<i>eni-id</i>

규칙 및 고려 사항

다음 중 하나라도 해당되는 경우 라우팅 테이블을 게이트웨이와 연결할 수 없습니다.

- 라우팅 테이블에 네트워크 인터페이스, Gateway Load Balancer 엔드포인트 또는 기본 로컬 라우팅이 아닌 대상이 있는 기존 라우팅이 포함된 경우
- 라우팅 테이블에 VPC의 범위를 벗어나는 CIDR 블록에 대한 기존 라우팅이 포함된 경우
- 라우팅 테이블에 대해 라우팅 전파가 활성화된 경우

또한 다음 규칙 및 고려 사항이 적용됩니다.

- 개별 VPC CIDR 블록보다 큰 범위를 포함하여 VPC의 범위를 벗어나는 CIDR 블록에는 라우팅을 추가할 수 없습니다.
- local, Gateway Load Balancer 엔드포인트 또는 네트워크 인터페이스를 대상으로 지정할 수만 있습니다. 개별 호스트 IP 주소 등 다른 유형의 대상은 지정할 수 없습니다. 자세한 내용은 [the section called “라우팅 옵션 예”](#) 단원을 참조하십시오.
- 접두사 목록을 대상으로 지정할 수 없습니다.
- 게이트웨이 라우팅 테이블을 사용하여 VPC 외부의 트래픽(예: 연결된 전송 게이트웨이를 통한 트래픽)을 제어하거나 가로챌 수 없습니다. VPC로 들어오는 트래픽을 가로채서 동일한 VPC에 있는 다른 대상으로만 리디렉션할 수 있습니다.
- 트래픽이 미들박스 어플라이언스에 도달하도록 하려면 실행 중인 인스턴스에 대상 네트워크 인터페이스가 연결되어야 합니다. 인터넷 게이트웨이를 통해 흐르는 트래픽의 경우, 대상 네트워크 인터페이스에는 퍼블릭 IP 주소도 있어야 합니다.
- 미들박스 어플라이언스를 구성할 때 [어플라이언스 고려 사항](#)을 기록해 둡니다.
- 미들박스 어플라이언스를 통해 트래픽을 라우팅하는 경우 대상 서브넷의 반환 트래픽은 동일한 어플라이언스를 통해 라우팅되어야 합니다. 비대칭 라우팅은 지원되지 않습니다.
- 라우팅 테이블 규칙은 서브넷을 떠나는 모든 트래픽에 적용됩니다. 서브넷을 떠나는 트래픽은 해당 서브넷의 게이트웨이 라우터의 MAC 주소를 대상으로 하는 트래픽으로 정의됩니다. 서브넷에 있는 다른 네트워크 인터페이스의 MAC 주소를 대상으로 하는 트래픽은 네트워크(계층 3) 대신 데이터 링크(계층 2) 라우팅을 사용하므로 규칙이 이 트래픽에 적용되지 않습니다.
- 모든 로컬 영역이 가상 프라이빗 게이트웨이와의 엣지 연결을 지원하는 것은 아닙니다. 사용 가능한 영역에 대한 자세한 내용은 AWS 로컬 영역 사용 설명서의 [고려 사항](#)을 참조하세요.

라우팅 우선순위

일반적으로 트래픽은 트래픽과 일치하는 가장 구체적인 경로를 사용하여 전달됩니다. 이를 가장 긴 접두사 일치라고 합니다. 라우팅 테이블에 겹치거나 일치하는 경로가 있는 경우 추가적인 규칙이 적용됩니다.

내용

- [가장 긴 접두사 일치](#)
- [경로 우선 순위 및 전파 경로](#)
- [라우팅 우선 순위 및 접두사 목록](#)

가장 긴 접두사 일치

IPv4 및 IPv6 주소 또는 CIDR 블록에 대한 라우팅은 서로 독립적입니다. IPv4 트래픽 또는 IPv6 트래픽과 일치하는 가장 구체적인 라우팅을 사용하여 트래픽 라우팅 방법을 결정합니다.

다음 예의 서브넷 라우팅 테이블에는 인터넷 게이트웨이를 가리키는 IPv4 인터넷 트래픽(0.0.0.0/0)에 대한 경로와 피어링 연결(172.31.0.0/16)을 가리키는 pcx-11223344556677889 IPv4 트래픽에 대한 경로가 있습니다. 172.31.0.0/16 IP 주소 범위로 향하는 서브넷의 모든 트래픽은 피어링 연결을 사용합니다. 이 라우팅은 인터넷 게이트웨이에 대한 라우팅보다 더 구체적이기 때문입니다. VPC(10.0.0.0/16) 내에서 대상으로 전송되는 트래픽은 local 라우팅이 적용되며 따라서 VPC 내에서 라우팅됩니다. 서브넷으로부터의 다른 모든 트래픽은 인터넷 게이트웨이를 사용합니다.

대상 주소	대상
10.0.0.0/16	로컬
172.31.0.0/16	pcx-11223344556677889
0.0.0.0/0	igw-12345678901234567

경로 우선 순위 및 전파 경로

가상 프라이빗 게이트웨이를 VPC에 연결하고 서브넷 라우팅 테이블에서 라우팅 전파를 활성화한 경우, Site-to-Site VPN 연결을 나타내는 라우팅은 라우팅 테이블에 전파된 라우팅으로 자동으로 나타납니다.

전파 경로의 대상이 정적 경로와 겹치는 경우 정적 경로가 우선 적용됩니다.

전파 경로의 대상이 정적 경로의 대상과 동일한 경우 대상이 다음 중 하나라면 정적 경로가 우선 적용됩니다.

- 인터넷 게이트웨이
- NAT 게이트웨이
- 네트워크 인터페이스
- 인스턴스 ID
- 게이트웨이 VPC 엔드포인트
- 전송 게이트웨이
- VPC 피어링 연결
- Gateway Load Balancer 엔드포인트

자세한 내용은 AWS Site-to-Site VPN 사용 설명서의 [라우팅 테이블 및 VPN 경로 우선 순위](#)를 참조하세요.

예를 들어 다음 라우팅 테이블에는 인터넷 게이트웨이에 대한 정적 경로와 가상 프라이빗 게이트웨이에 대한 전파 경로가 있습니다. 두 라우팅은 모두 대상 주소가 172.31.0.0/24입니다. 인터넷 게이트웨이에 대한 정적 경로가 우선 적용되므로 대상이 172.31.0.0/24인 모든 트래픽이 인터넷 게이트웨이로 라우팅됩니다.

대상 주소	대상	전파
10.0.0.0/16	로컬	아니요
172.31.0.0/24	vgw-11223344556677889	예
172.31.0.0/24	igw-12345678901234567	아니요

라우팅 우선 순위 및 접두사 목록

라우팅 테이블이 접두사 목록을 참조하는 경우 다음 규칙이 적용됩니다.

- 라우팅 테이블에 접두사 목록이 있는 정적 경로와 겹치는 CIDR 블록 대상 정적 경로가 포함된 경우, CIDR 블록이 있는 정적 경로가 우선 적용됩니다.

- 라우팅 테이블에 접두사 목록을 참조하는 라우팅과 일치하는 전파된 라우팅이 포함된 경우 접두사 목록을 참조하는 라우팅이 먼저 적용됩니다. 겹치는 경로의 경우 전파된 경로, 정적 경로 또는 접두사 목록을 참조하는 경로인지와 관계없이 보다 구체적인 경로가 항상 우선합니다.
- 라우팅 테이블이 서로 다른 대상에 겹치는 CIDR 블록이 있는 여러 접두사 목록을 참조하는 경우 우선 적용할 라우팅을 임의로 선택합니다. 그 이후에는 동일한 라우팅이 항상 우선합니다.

라우팅 테이블 할당량

VPC당 생성할 수 있는 라우팅 테이블 수에는 할당량이 있습니다. 라우팅 테이블당 추가할 수 있는 라우팅 수에도 할당량이 있습니다. 자세한 내용은 [Amazon VPC 할당량](#) 단원을 참조하십시오.

연결 문제 해결

Reachability Analyzer는 정적 구성 분석 도구입니다. Reachability Analyzer를 사용하여 VPC의 두 리소스 간 네트워크 연결성을 분석하고 디버깅할 수 있습니다. Reachability Analyzer에서는 연결할 수 있는 경우 이러한 리소스 간 가상 경로에 대한 흠별 세부 정보가 생성되고, 그렇지 않다면 차단 구성 요소가 식별됩니다. 예를 들면 누락되거나 잘못 구성된 라우팅 테이블 경로가 식별될 수 있습니다.

자세한 내용은 [Reachability Analyzer 사용 설명서](#)를 참조하십시오.

라우팅 옵션 예

다음 주제에서는 VPC의 특정 게이트웨이 또는 연결을 위한 라우팅을 설명합니다.

내용

- [인터넷 게이트웨이로 라우팅](#)
- [NAT 디바이스로 라우팅](#)
- [가상 프라이빗 게이트웨이로 라우팅](#)
- [AWS Outposts 로컬 게이트웨이로 라우팅](#)
- [VPC 피어링 연결로 라우팅](#)
- [게이트웨이 VPC 엔드포인트로 라우팅](#)
- [외부 전용 인터넷 게이트웨이로 라우팅](#)
- [전송 게이트웨이에 대한 라우팅](#)
- [미들박스 어플라이언스에 대한 라우팅](#)
- [접두사 목록을 사용한 라우팅](#)
- [Gateway Load Balancer 엔드포인트로의 라우팅](#)

인터넷 게이트웨이로 라우팅

서브넷 라우팅 테이블의 라우팅을 인터넷 게이트웨이에 추가하여 서브넷을 퍼블릭 서브넷으로 만들 수 있습니다. 이렇게 하려면 인터넷 게이트웨이를 생성하여 VPC에 연결한 다음, IPv4 트래픽에 대한 대상 주소가 0.0.0.0/0인 라우팅이나 IPv6 트래픽에 대한 대상 주소가 ::/0인 라우팅을 추가하고, 인터넷 게이트웨이 ID(igw-xxxxxxxxxxxxxxxxxx)의 대상을 추가합니다.

대상 주소	대상
0.0.0.0/0	<i>igw-id</i>
::/0	<i>igw-id</i>

자세한 내용은 [인터넷 게이트웨이를 사용하여 인터넷에 연결](#) 단원을 참조하십시오.

NAT 디바이스로 라우팅

프라이빗 서브넷의 인스턴스가 인터넷에 연결되도록 하려면 NAT 게이트웨이를 생성하거나 퍼블릭 서브넷에서 NAT 인스턴스를 시작할 수 있습니다. 그런 다음 IPv4 인터넷 트래픽(0.0.0.0/0)을 NAT 디바이스로 라우팅하는 프라이빗 서브넷의 라우팅 테이블에 대한 라우팅을 추가합니다.

대상 주소	대상
0.0.0.0/0	<i>nat-gateway-id</i>

또한 NAT 게이트웨이 사용에 대한 불필요한 데이터 처리 비용을 피하거나 특정 트래픽을 비공개로 라우팅하기 위해 다른 대상에 대한 보다 구체적인 라우팅을 생성할 수 있습니다. 다음 예에서 Amazon S3 트래픽(pl-xxxxxxx, 특전 리전의 Amazon S3에 대한 IP 주소 범위가 포함된 접두사 목록)은 게이트웨이 VPC 엔드포인트로 라우팅되고 10.25.0.0/16 트래픽은 VPC 피어링 연결로 라우팅됩니다. 이러한 IP 주소 범위는 0.0.0.0/0보다 더 구체적입니다. 인스턴스가 Amazon S3 또는 피어 VPC로 트래픽을 보내면 트래픽이 게이트웨이 VPC 엔드포인트 또는 VPC 피어링 연결로 전송됩니다. 다른 모든 트래픽은 NAT 게이트웨이로 전송됩니다.

대상 주소	대상
0.0.0.0/0	<i>nat-gateway-id</i>

대상 주소	대상
pl-xxxxxxxxx	vpce-id
10.25.0.0/16	pcx-id

자세한 내용은 [NAT 디바이스](#) 단원을 참조하십시오.

가상 프라이빗 게이트웨이로 라우팅

AWS Site-to-Site VPN 연결을 사용하여 VPC의 인스턴스를 사용자의 네트워크와 통신하도록 할 수 있습니다. 이렇게 하려면 가상 프라이빗 게이트웨이를 생성하여 VPC에 연결합니다. 그런 다음 네트워크 대상 및 가상 프라이빗 게이트웨이(vgw-xxxxxxxxxxxxxxxxxxx)의 대상이 있는 서브넷 라우팅 테이블에 라우팅을 추가합니다.

대상 주소	대상
10.0.0.0/16	vgw-id

이제 Site-to-Site VPN 연결을 만들고 구성할 수 있습니다. 자세한 내용은 AWS Site-to-Site VPN 사용 설명서의 [AWS Site-to-Site VPN이란 무엇입니까?](#) 및 [라우팅 테이블 및 VPN 경로 우선 순위](#)를 참조하십시오.

가상 프라이빗 게이트웨이의 Site-to-Site VPN 연결은 IPv6 트래픽을 지원하지 않습니다. 그러나 가상 프라이빗 게이트웨이를 통해 AWS Direct Connect 연결로 라우팅되는 IPv6 트래픽은 지원합니다. 자세한 내용은 [AWS Direct Connect 사용 설명서](#)를 참조하십시오.

AWS Outposts 로컬 게이트웨이로 라우팅

이 섹션에서는 AWS Outposts 로컬 게이트웨이로 라우팅을 위한 라우팅 테이블 구성에 대해 설명합니다.

내용

- [Outpost 서브넷과 온프레미스 네트워크 간 트래픽 활성화](#)
- [Outpost 전체에서 동일한 VPC에 있는 서브넷 간 트래픽 활성화](#)

Outpost 서브넷과 온프레미스 네트워크 간 트래픽 활성화

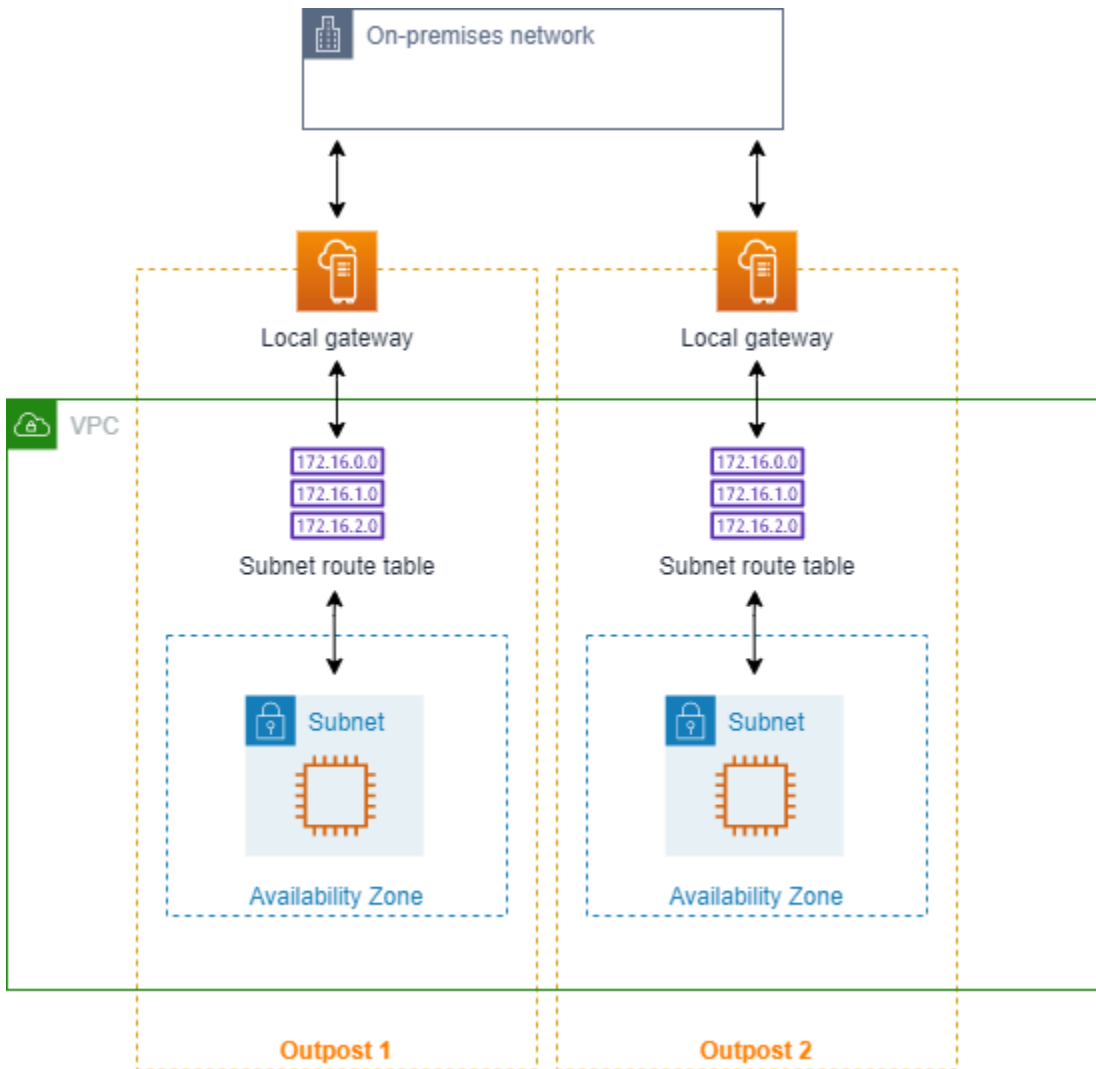
AWS Outposts와 연결된 VPC에 있는 서브넷은 로컬 게이트웨이의 추가 대상 유형을 가질 수 있습니다. 대상 주소가 192.168.10.0/24인 로컬 게이트웨이 트래픽을 고객 네트워크로 라우팅하려는 경우를 생각해 보세요. 이렇게 하려면 대상 네트워크와 로컬 게이트웨이(lgw-xxxx)의 대상을 사용하여 다음 라우팅을 추가합니다.

대상 주소	대상
192.168.10.0/24	<i>lgw-id</i>

Outpost 전체에서 동일한 VPC에 있는 서브넷 간 트래픽 활성화

Outpost 로컬 게이트웨이와 온프레미스 네트워크를 사용하여 여러 Outpost에서 동일한 VPC에 있는 서브넷 간 통신을 설정할 수 있습니다.

이 기능을 사용하면 서로 다른 가용 영역에 고정된 Outpost 랙 간에 연결을 설정하여 Outpost 랙에서 실행되는 온프레미스 애플리케이션을 위한 다중 가용 영역 아키텍처와 유사한 아키텍처를 구축할 수 있습니다.



이 기능을 활성화하려면 Outpost 랙 서브넷 라우팅 테이블에 해당 라우팅 테이블의 로컬 라우팅보다 더 구체적이고 대상 유형이 로컬 게이트웨이인 경로를 추가합니다. 경로의 대상은 다른 Outpost의 VPC에 있는 서브넷의 전체 IPv4 블록과 일치해야 합니다. 통신해야 하는 모든 Outpost 서브넷에 대해 이 구성을 반복합니다.

⚠ Important

- 이 기능을 사용하려면 [직접 VPC 라우팅](#)을 사용해야 합니다. [고객 소유의 IP 주소](#)는 사용할 수 없습니다.
- 서브넷이 서로 액세스할 수 있도록 Outpost 로컬 게이트웨이가 연결된 온프레미스 네트워크에 필요한 라우팅이 있어야 합니다.
- 서브넷의 리소스에 보안 그룹을 사용하려면 IP 주소 범위를 Outpost 서브넷의 소스 또는 대상으로 포함하는 규칙을 사용해야 합니다. 보안 그룹 ID는 사용할 수 없습니다.

- 여러 Outpost 간의 VPC 내 통신 지원을 활성화하기 위해 기존 Outpost 랙의 업데이트가 필요할 수 있습니다. 이 기능이 작동하지 않으면 [AWS Support](#)에 문의하세요.

Example 예

CIDR이 10.0.0.0/16인 VPC, CIDR이 10.0.1.0/24인 Outpost 1 서브넷, CIDR이 10.0.2.0/24인 Outpost 2 서브넷의 경우 Outpost 1 서브넷의 라우팅 테이블 항목은 다음과 같습니다.

대상 주소	대상
10.0.0.0/16	로컬
10.0.2.0/24	<i>lgw-1-id</i>

Outpost 2 서브넷의 라우팅 테이블 항목은 다음과 같습니다.

대상 주소	대상
10.0.0.0/16	로컬
10.0.1.0/24	<i>lgw-2-id</i>

VPC 피어링 연결로 라우팅

VPC 피어링 연결은 프라이빗 IPv4 주소를 사용하여 두 VPC 간에 트래픽을 라우팅할 수 있게 해주는 두 VPC 사이의 네트워킹 연결입니다. 동일한 네트워크에 속하는 경우와 같이 VPC의 인스턴스가 서로 통신할 수 있습니다.

VPC 피어링 연결에서 VPC 간 트래픽 라우팅을 활성화하려면 VPC 피어링 연결을 가리키는 하나 이상의 서브넷 라우팅 테이블에 라우팅을 추가해야 합니다. 이렇게 하면 피어링 연결에서 다른 VPC의 CIDR 블록 전체 또는 일부에 액세스할 수 있습니다. 마찬가지로, 다른 VPC의 소유자는 트래픽을 다시 사용자의 VPC로 라우팅하기 위해 소유자 자신의 서브넷 라우팅 테이블에 라우팅을 추가해야 합니다.

예를 들어 다음과 같은 정보를 가진 두 VPC 사이에 VPC 피어링 연결(`pcx-11223344556677889`)이 있다고 합시다.

- VPC A: CIDR 블록은 10.0.0.0/16

- VPC B: CIDR 블록은 172.31.0.0/16

VPC 간에 트래픽을 활성화하고 어느 한 VPC의 전체 IPv4 CIDR 블록에 대한 액세스를 허용하기 위해 VPC A의 라우팅 테이블은 다음과 같이 구성됩니다.

대상 주소	대상
10.0.0.0/16	로컬
172.31.0.0/16	pcx-11223344556677889

VPC B의 라우팅 테이블은 다음과 같이 구성됩니다.

대상 주소	대상
172.31.0.0/16	로컬
10.0.0.0/16	pcx-11223344556677889

VPC와 인스턴스가 IPv6 통신을 할 수 있는 경우, VPC 피어링 연결은 VPC의 인스턴스 간 IPv6 통신도 지원할 수 있습니다. VPC 간 IPv6 트래픽을 라우팅할 수 있게 하려면, VPC 피어링 연결을 가리키는 라우팅 테이블에 대한 경로를 추가하여 피어 VPC의 IPv6 CIDR 블록 전부 또는 일부에 액세스해야 합니다.

예를 들어 VPC가 위와 같이 동일한 VPC 피어링 연결(pcx-11223344556677889)을 사용하여 다음과 같은 정보를 갖고 있다고 가정합니다.

- VPC A: IPv6 CIDR 블록은 2001:db8:1234:1a00::/56입니다.
- VPC B: IPv6 CIDR 블록은 2001:db8:5678:2b00::/56입니다.

VPC 피어링 연결을 통해 IPv6 통신을 할 수 있도록 하려면 VPC A에 대한 서브넷 라우팅 테이블에 다음 라우팅을 추가합니다.

대상 주소	대상
10.0.0.0/16	로컬

대상 주소	대상
172.31.0.0/16	pcx-11223344556677889
2001:db8:5678:2b00::/56	pcx-11223344556677889

다음 라우팅을 VPC B에 대한 라우팅 테이블에 추가합니다.

대상 주소	대상
172.31.0.0/16	로컬
10.0.0.0/16	pcx-11223344556677889
2001:db8:1234:1a00::/56	pcx-11223344556677889

VPC 피어링 연결에 대한 자세한 내용은 [Amazon VPC 사용 설명서](#)를 참조하십시오.

게이트웨이 VPC 엔드포인트로 라우팅

게이트웨이 VPC 엔드포인트를 사용하면 VPC와 다른 AWS 서비스 사이에 프라이빗 연결을 생성할 수 있습니다. 게이트웨이 엔드포인트를 생성할 때 VPC에서 게이트웨이 엔드포인트에서 사용하는 서브넷 라우팅 테이블을 지정합니다. 경로는 각각의 라우팅 테이블에 자동으로 추가되며 이때 서비스의 접두사 목록 ID(p1-xxxxxxx)를 지정하는 목적지 및 엔드포인트 ID(vpce-xxxxxxxxxxxxxxxx)를 포함한 대상도 함께 추가됩니다. 엔드포인트 경로를 명시적으로 삭제하거나 수정할 수는 없지만, 엔드포인트에서 사용되는 라우팅 테이블을 변경할 수는 있습니다.

엔드포인트에 대한 라우팅과 AWS 서비스에 대한 경로에 대한 자세한 내용은 [게이트웨이 엔드포인트에 대한 라우팅](#)을 참조하세요.

외부 전용 인터넷 게이트웨이로 라우팅

VPC에 외부 전용 인터넷 게이트웨이를 생성하여 프라이빗 서브넷의 인스턴스가 인터넷에 대한 아웃바운드 통신을 시작하도록 하되 인터넷에서 해당 인스턴스와의 연결을 시작하지 못하도록 할 수 있습니다. 외부 전용 인터넷 게이트웨이는 IPv6 트래픽에만 사용됩니다. 외부 전용 인터넷 게이트웨이에 대한 라우팅을 구성하려면 IPv6 인터넷 트래픽(::

대상 주소	대상
::/0	<i>eigw-id</i>

자세한 내용은 [송신 전용 인터넷 게이트웨이를 사용하여 아웃바운드 IPv6 트래픽 활성화](#) 단원을 참조하십시오.

전송 게이트웨이에 대한 라우팅

전송 게이트웨이에 VPC를 연결할 때 전송 게이트웨이를 통해 라우팅할 트래픽에 대한 라우팅을 서브넷 라우팅 테이블에 추가해야 합니다.

전송 게이트웨이에 연결된 VPC 3개가 있는 다음 시나리오를 고려하세요. 이 시나리오에서는 모든 연결이 전송 게이트웨이 라우팅 테이블과 연결되어 전송 게이트웨이 라우팅 테이블에 전파됩니다. 따라서 모든 연결은 패킷을 서로 라우팅할 수 있으며 전송 게이트웨이는 단순한 계층 3 IP 허브 역할을 합니다.

예를 들어, 다음과 같은 정보를 가진 두 VPC가 있다고 가정합니다.

- VPC A: 10.1.0.0/16, 연결 ID tgw-attach-111111111111111111
- VPC B: 10.2.0.0/16, 연결 ID tgw-attach-222222222222222222

VPC 간에 트래픽을 활성화하고 전송 게이트웨이에 대한 액세스를 허용하려면 VPC A의 라우팅 테이블을 다음과 같이 구성해야 합니다.

대상 주소	대상
10.1.0.0/16	로컬
10.0.0.0/8	<i>tgw-id</i>

다음은 VPC 연결에 대한 전송 게이트웨이 라우팅 테이블 항목의 예입니다.

대상 주소	대상
10.1.0.0/16	tgw-attach-111111111111111111

대상 주소	대상
10.2.0.0/16	tgw-attach-222222222222222222

전송 게이트웨이 라우팅 테이블에 대한 자세한 내용은 Amazon VPC Transit Gateways의 [라우팅](#)을 참조하십시오.

미들박스 어플라이언스에 대한 라우팅

VPC에 대한 라우팅 경로에 미들박스 어플라이언스를 추가할 수 있습니다. 다음은 몇 가지 가능한 사용 사례입니다.

- 인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이를 통해 VPC로 들어오는 트래픽을 VPC의 미들박스 어플라이언스로 전송하여 가로챍니다. 미들박스 라우팅 마법사를 사용하여 AWS가 게이트웨이, 미들박스 및 대상 서브넷에 적합한 라우팅 테이블을 자동으로 구성할 수 있습니다. 자세한 내용은 [the section called “미들박스 라우팅 마법사”](#) 단원을 참조하십시오.
- 두 서브넷 간의 트래픽을 미들박스 어플라이언스로 보냅니다. 이렇게 하려면 다른 서브넷의 서브넷 CIDR과 일치하는 서브넷 라우팅 테이블에 대한 경로를 생성하고 Gateway Load Balancer 엔드포인트, NAT 게이트웨이, Network Firewall 엔드포인트 또는 어플라이언스의 네트워크 인터페이스를 대상으로 지정합니다. 또는 서브넷에서 다른 서브넷으로 모든 트래픽을 리디렉션하려면 로컬 경로의 대상을 Gateway Load Balancer 엔드포인트, NAT 게이트웨이 또는 네트워크 인터페이스로 바꿉니다.

필요에 맞게 어플라이언스를 구성할 수 있습니다. 예를 들어 모든 트래픽을 차단하는 보안 어플라이언스 또는 WAN 가속 어플라이언스를 구성할 수 있습니다. 어플라이언스는 VPC의 서브넷에 Amazon EC2 인스턴스로 배포되며 서브넷의 탄력적 네트워크 인터페이스(네트워크 인터페이스)로 표시됩니다.

대상 서브넷 라우팅 테이블에 대해 라우팅 전파를 활성화한 경우 라우팅 우선 순위를 알고 있어야 합니다. 가장 구체적인 라우팅에 대해 우선 순위를 두며 라우팅이 일치하면 전파된 라우팅보다 정적 라우팅에 우선 순위를 둡니다. 라우팅을 검토하여 트래픽이 올바르게 라우팅되고 라우팅 전파를 활성화 또는 비활성화할 경우 의도하지 않은 결과가 없는지 확인합니다(예: 점보 프레임을 지원하는 AWS Direct Connect 연결에 대해 라우팅 전파가 필요함).

인바운드 VPC 트래픽을 어플라이언스로 라우팅하려면 라우팅 테이블을 인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이와 연결하고 어플라이언스의 네트워크 인터페이스를 VPC 트래픽의 대상으로 지

정합니다. 자세한 내용은 [게이트웨이 라우팅 테이블](#) 단원을 참조하십시오. 서브넷에서 다른 서브넷의 미들박스 어플라이언스로 아웃바운드 트래픽을 라우팅할 수도 있습니다.

미들박스 라우팅 예제에 대해서는 [미들박스 시나리오](#)을(를) 참조하세요.

내용

- [어플라이언스 고려 사항](#)
- [게이트웨이와 어플라이언스 간 트래픽 라우팅](#)
- [어플라이언스로 서브넷 간 트래픽 라우팅](#)

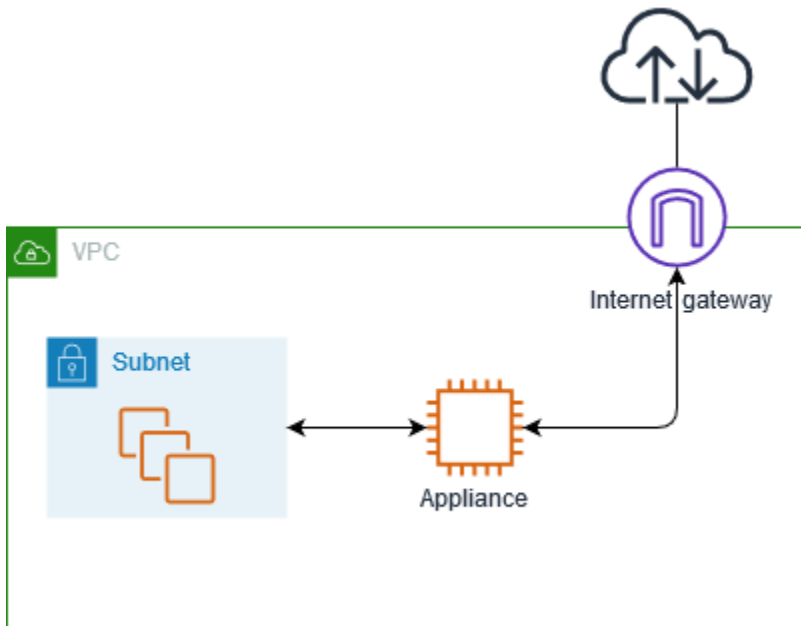
어플라이언스 고려 사항

[AWS Marketplace](#)에서 서드 파티 어플라이언스를 선택하거나 직접 어플라이언스를 구성할 수 있습니다. 어플라이언스를 생성하거나 구성할 때 다음 사항에 유의하세요.

- 어플라이언스는 원본 또는 대상 트래픽과 별도의 서브넷에 구성되어야 합니다.
- 어플라이언스에서 원본/대상 확인을 비활성화해야 합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [소스 또는 대상 변경 확인](#)을 참조하세요.
- 어플라이언스를 통해 동일한 서브넷에 있는 호스트 간에 트래픽을 라우팅할 수 없습니다.
- 어플라이언스는 NAT(네트워크 주소 변환)를 수행할 필요가 없습니다.
- 로컬 경로보다 더 구체적인 경로를 라우팅 테이블에 추가할 수 있습니다. 보다 구체적인 경로를 사용하여 VPC 내의 서브넷 간의 트래픽(동서 트래픽)을 미들박스 어플라이언스로 리디렉션할 수 있습니다. 경로의 대상은 VPC에 있는 서브넷의 전체 IPv4 또는 IPv6 CIDR 블록과 일치해야 합니다.
- IPv6 트래픽을 가로채려면 VPC, 서브넷 및 어플라이언스가 IPv6를 지원하는지 확인합니다. 가상 프라이빗 게이트웨이는 IPv6 트래픽을 지원하지 않습니다.

게이트웨이와 어플라이언스 간 트래픽 라우팅

인바운드 VPC 트래픽을 어플라이언스로 라우팅하려면 라우팅 테이블을 인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이와 연결하고 어플라이언스의 네트워크 인터페이스를 VPC 트래픽의 대상으로 지정합니다. 다음 예제에서 VPC에는 인터넷 게이트웨이, 어플라이언스 및 인스턴스가 있는 서브넷이 있습니다. 인터넷으로부터의 트래픽은 어플라이언스를 통해 라우팅됩니다.



이 라우팅 테이블을 인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이와 연결합니다. 첫 번째 항목은 로컬 경로입니다. 두 번째 항목은 서브넷을 대상으로 하는 IPv4 트래픽을 어플라이언스에 대한 네트워크 인터페이스로 전송합니다. 이는 로컬 경로보다 더 구체적인 경로입니다.

대상 주소	대상
<i>VPC CIDR</i>	로컬
<i>### CIDR</i>	<i>##### ### ID</i>

또는 로컬 경로의 대상을 어플라이언스의 네트워크 인터페이스로 대체할 수 있습니다. 이렇게 하면 향후 VPC에 추가하는 서브넷으로 향하는 트래픽을 포함하여 모든 트래픽이 해당 어플라이언스로 자동 라우팅됩니다.

대상 주소	대상
<i>VPC CIDR</i>	<i>##### ### ID</i>

서브넷에서 다른 서브넷의 어플라이언스로 트래픽을 라우팅하려면 트래픽을 어플라이언스의 네트워크 인터페이스로 라우팅하는 라우팅을 서브넷 라우팅 테이블에 추가합니다. 대상은 로컬 라우팅의 대상보다 덜 구체적이어야 합니다. 예를 들어 인터넷으로 향하는 트래픽의 경우 대상에 대해 0.0.0.0/0(모든 IPv4 주소)을 지정합니다.

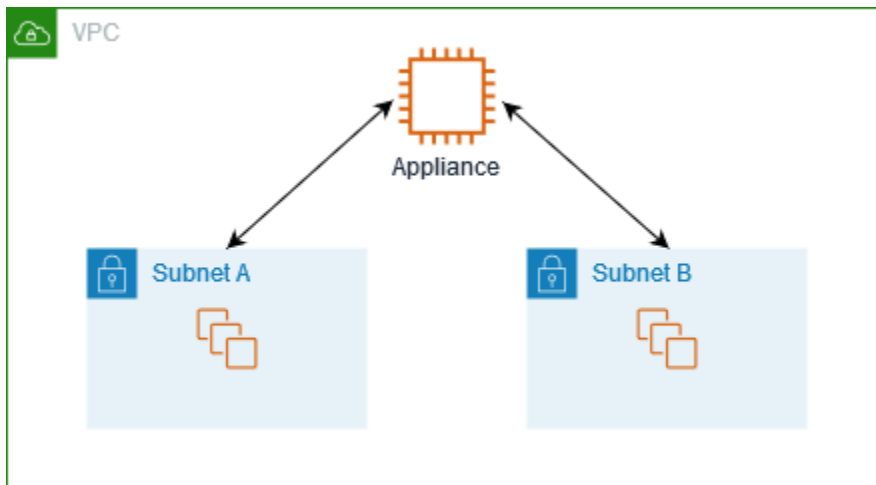
대상 주소	대상
<i>VPC CIDR</i>	로컬
0.0.0.0/0	<i>##### ##### ##### ID</i>

그런 다음 어플라이언스의 서브넷과 연결된 라우팅 테이블에서 트래픽을 인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이로 다시 전송하는 라우팅을 추가합니다.

대상 주소	대상
<i>VPC CIDR</i>	로컬
0.0.0.0/0	<i>igw-id</i>

어플라이언스로 서브넷 간 트래픽 라우팅

특정 서브넷을 대상으로 하는 트래픽을 어플라이언스의 네트워크 인터페이스로 라우팅할 수 있습니다. 다음 예제에서 VPC는 서브넷 두 개와 어플라이언스로 구성됩니다. 서브넷 간에 트래픽은 어플라이언스를 통해 라우팅됩니다.



보안 그룹

미들박스 어플라이언스를 통해 서로 다른 서브넷에 있는 인스턴스 간에 트래픽을 라우팅하는 경우 두 인스턴스에 대한 보안 그룹에서 인스턴스 간에 트래픽이 흐르도록 허용해야 합니다. 각 인스턴스의 보안 그룹은 다른 인스턴스의 프라이빗 IP 주소 또는 다른 인스턴스가 포함된 서브넷의 CIDR 범위를 소

스로 참조해야 합니다. 다른 인스턴스의 보안 그룹을 소스로 참조하면 인스턴스 간에 트래픽이 흐를 수 없습니다.

라우팅

다음은 서브넷 A에 대한 라우팅 테이블의 예제입니다. 첫 번째 항목을 사용하면 VPC 인스턴스가 서로 통신할 수 있습니다. 두 번째 항목은 서브넷 A에서 서브넷 B로의 모든 트래픽을 장치의 네트워크 인터페이스로 라우팅합니다.

대상 주소	대상
<i>VPC CIDR</i>	로컬
<i>### B CIDR</i>	<i>##### ##### ##### ID</i>

다음은 서브넷 B에 대한 라우팅 테이블의 예제입니다. 첫 번째 항목을 사용하면 VPC 인스턴스가 서로 통신할 수 있습니다. 두 번째 항목은 서브넷 B에서 서브넷 A로의 모든 트래픽을 장치의 네트워크 인터페이스로 라우팅합니다.

대상 주소	대상
<i>VPC CIDR</i>	로컬
<i>### A CIDR</i>	<i>##### ##### ##### ID</i>

또는 로컬 경로의 대상을 어플라이언스의 네트워크 인터페이스로 대체할 수 있습니다. 이렇게 하면 향후 VPC에 추가하는 서브넷으로 향하는 트래픽을 포함하여 모든 트래픽이 해당 어플라이언스로 자동 라우팅됩니다.

대상 주소	대상
<i>VPC CIDR</i>	<i>##### ##### ##### ID</i>

접두사 목록을 사용한 라우팅

AWS 리소스 전체에서 동일한 CIDR 블록 세트를 자주 참조하는 경우, [고객 관리형 접두사 목록](#)을 생성하여 함께 그룹화할 수 있습니다. 그런 다음 라우팅 테이블 항목의 대상으로 접두사 목록을 지정할 수

있습니다. 나중에 라우팅 테이블을 업데이트할 필요 없이 접두사 목록에 대한 항목을 추가하거나 제거할 수 있습니다.

예를 들어, 여러 VPC 연결이 있는 전송 게이트웨이가 있습니다. VPC는 다음과 같은 CIDR 블록이 있는 두 개의 특정 VPC 연결과 통신할 수 있어야 합니다.

- 10.0.0.0/16
- 10.2.0.0/16

두 항목이 모두 포함된 접두사 목록을 만듭니다. 서브넷 라우팅 테이블에서 라우팅을 생성하고 접두사 목록을 대상 주소로 지정한 다음 전송 게이트웨이를 대상으로 지정합니다.

대상 주소	대상
172.31.0.0/16	로컬
pl-123abc123abc123ab	<i>tgw-id</i>

접두사 목록의 최대 항목 수는 라우팅 테이블에 있는 항목 수와 같습니다.

Gateway Load Balancer 엔드포인트로의 라우팅

Gateway Load Balancer를 사용하면 방화벽 같은 가상 어플라이언스 플릿에 트래픽을 분산할 수 있습니다. [VPC 엔드포인트 서비스 구성](#)을 생성하여 로드 밸런서를 서비스로 구성할 수 있습니다. 그런 다음 VPC에 [Gateway Load Balancer 엔드포인트](#)를 생성하여 VPC를 서비스에 연결합니다.

트래픽을 Gateway Load Balancer로 라우팅하려면(예: 보안 검사) Gateway Load Balancer 엔드포인트를 라우팅 테이블에서 대상으로 지정합니다.

Gateway Load Balancer 뒤에 있는 보안 어플라이언스의 예제는 [the section called “보안 어플라이언스를 사용하여 트래픽 검사”](#)를 참조하세요.

라우팅 테이블에 Gateway Load Balancer 엔드포인트를 지정하려면 VPC 엔드포인트의 ID를 사용합니다. 예를 들어 10.0.1.0/24에 대한 트래픽을 Gateway Load Balancer 엔드포인트로 라우팅하려면 다음 경로를 추가합니다.

대상 주소	대상
10.0.1.0/24	<i>vpc-endpoint-id</i>

자세한 내용은 [Gateway Load Balancers](#)를 참조하세요.

라우팅 테이블 작업

이 섹션에서는 라우팅 테이블을 사용하는 방법에 대해 설명합니다.

내용

- [서브넷의 라우팅 테이블 확인](#)
- [명시적으로 연결되어 있는 서브넷 또는 게이트웨이 확인](#)
- [사용자 지정 라우팅 테이블 생성](#)
- [라우팅 테이블에서 경로 추가 및 제거](#)
- [경로 전파 활성화 또는 비활성화](#)
- [서브넷을 라우팅 테이블과 연결](#)
- [서브넷의 라우팅 테이블 변경](#)
- [라우팅 테이블에서 서브넷의 연결 끊기](#)
- [기본 라우팅 테이블 교체](#)
- [게이트웨이를 라우팅 테이블과 연결](#)
- [라우팅 테이블에서 게이트웨이 연결 해제](#)
- [로컬 경로의 대상 교체 또는 복원](#)
- [라우팅 테이블 삭제](#)

서브넷의 라우팅 테이블 확인

Amazon VPC 콘솔에서 서브넷의 세부 정보를 살펴보면 서브넷이 연결되어 있는 라우팅 테이블을 확인할 수 있습니다.

서브넷의 라우팅 테이블을 확인하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 서브넷(Subnets)을 선택합니다.

3. 서브넷을 선택합니다.
4. 라우팅 테이블 ID와 경로에 대한 정보를 보려면 라우팅 테이블(Route Table) 탭을 선택합니다. 기본 라우팅 테이블에 대한 연결인지, 해당 연결이 명시적인지 확인하려면 [명시적으로 연결되어 있는 서브넷 또는 게이트웨이 확인](#) 단원을 참조하십시오.

명시적으로 연결되어 있는 서브넷 또는 게이트웨이 확인

라우팅 테이블과 명시적으로 연결된 서브넷 또는 게이트웨이와 그 수를 확인할 수 있습니다.

기본 라우팅 테이블에는 명시적 및 암시적 서브넷 연결이 있을 수 있습니다. 사용자 지정 라우팅 테이블에는 명시적 연결만 있습니다.

어떤 라우팅 테이블과도 명시적으로 연결되지 않은 서브넷은 기본 라우팅 테이블과 암시적으로 연결됩니다. 서브넷을 기본 라우팅 테이블과 명시적으로 연결할 수 있습니다. 이 작업을 해야 하는 대표적인 이유는 [기본 라우팅 테이블 교체](#)를 참조하십시오.

콘솔을 사용하여 명시적으로 연결된 서브넷을 확인하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블을 선택합니다.
3. 명시적 서브넷 연결(Explicit subnet association) 열을 선택하여 명시적으로 연결된 서브넷을 확인하고 기본(Main) 열을 선택하여 기본 라우팅 테이블인지도 확인합니다.
4. 라우팅 테이블을 선택하고 서브넷 연결(Subnet associations) 탭을 선택합니다.
5. 명시적으로 서브넷 연결(Explicit subnet associations) 아래에 있는 서브넷은 라우팅 테이블과 명시적으로 연결되어 있습니다. 명시적 연결이 없는 서브넷(Subnets without explicit associations) 아래에 있는 서브넷은 라우팅 테이블로 같은 VPC에 속해 있지만 어떤 라우팅 테이블과도 연결되지 않아 VPC 기본 라우팅 테이블과 암시적으로 연결되어 있습니다.

콘솔을 사용하여 명시적으로 연결된 게이트웨이를 확인하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블을 선택합니다.
3. 라우팅 테이블을 선택하고 엣지 연결(Edge associations) 탭을 선택합니다.

명령줄을 사용하여 하나 이상의 라우팅 테이블을 설명하고 해당 연결을 보려면

- [describe-route-tables](#)(AWS CLI)

- [Get-EC2RouteTable](#)(AWS Tools for Windows PowerShell)

사용자 지정 라우팅 테이블 생성

Amazon VPC 콘솔을 사용하여 VPC에 대한 사용자 지정 라우팅 테이블을 만들 수 있습니다.

콘솔을 사용하여 사용자 지정 라우팅 테이블을 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블을 선택합니다.
3. 라우팅 테이블 생성을 선택합니다.
4. (선택 사항) 이름(Name)에 라우팅 테이블의 이름을 입력합니다.
5. VPC에서 VPC를 선택합니다.
6. (선택 사항) 태그를 추가하려면 새 태그 추가(Add new tag)를 선택하고 태그 키와 태그 값을 입력합니다.
7. 라우팅 테이블 생성을 선택합니다.

명령줄을 사용하여 사용자 지정 라우팅 테이블을 생성하려면

- [create-route-table](#)(AWS CLI)
- [New-EC2RouteTable](#)(AWS Tools for Windows PowerShell)

라우팅 테이블에서 경로 추가 및 제거

라우팅 테이블에서 경로를 추가, 삭제 및 수정할 수 있습니다. 사용자가 추가한 경로만 수정할 수 있습니다.

Site-to-Site VPN 연결에 대한 정적 라우팅 작업에 대한 자세한 내용은 AWS Site-to-Site VPN 사용 설명서의 [Site-to-Site VPN 연결에 대한 고정 경로 편집](#)을 참조하세요.

콘솔을 사용하여 라우팅 테이블의 경로를 업데이트하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블(Route Tables)를 선택한 후 라우팅 테이블을 선택합니다.
3. 작업(Actions), 라우팅 편집(Edit routes)을 선택합니다.

4. 라우팅을 추가하려면 라우팅 추가(Add route)를 선택합니다. 대상에서 대상 CIDR 블록, 단일 IP 주소 또는 접두사 목록의 ID를 입력합니다.
5. 라우팅을 수정하려면 대상(Destination)에서 대상 CIDR 블록 또는 단일 IP 주소를 바꿉니다. 대상(Target)에서 대상을 선택합니다.
6. 라우팅을 삭제하려면 제거(Remove)를 선택합니다.
7. 변경 사항 저장(Save changes)을 선택합니다.

명령줄을 사용하여 라우팅 테이블의 경로를 업데이트하려면

- [create-route](#)(AWS CLI)
- [replace-route](#)(AWS CLI)
- [delete-route](#)(AWS CLI)
- [New-EC2Route](#)(AWS Tools for Windows PowerShell)
- [Set-EC2Route](#)(AWS Tools for Windows PowerShell)
- [Remove-EC2Route](#)(AWS Tools for Windows PowerShell)

Note

명령줄 도구 또는 API를 사용하여 라우팅을 추가하면 대상 CIDR 블록이 표준 형식으로 자동 수정됩니다. 예를 들어 CIDR 블록에 대해 100.68.0.18/18을 지정하면 대상 CIDR 블록이 100.68.0.0/18인 라우팅이 생성됩니다.

경로 전파 활성화 또는 비활성화

라우팅 전파를 사용하면 가상 프라이빗 게이트웨이가 라우팅 테이블에 경로를 자동으로 전파할 수 있습니다. 따라서 VPN 경로를 수동으로 추가하거나 제거할 필요가 없습니다.

이 프로세스를 완료하려면 가상 프라이빗 게이트웨이가 있어야 합니다.

자세한 내용은 Site-to-Site VPN 사용 설명서의 [Site-to-Site VPN 라우팅 옵션](#)을 참조하세요.

콘솔을 사용하여 라우팅 전파를 활성화하는 방법

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블(Route Tables)를 선택한 후 라우팅 테이블을 선택합니다.

3. 작업, Edit route propagation(라우팅 속성 편집)을 선택합니다.
4. 가상 프라이빗 게이트웨이 옆에 있는 [활성화(Enable)] 확인란을 선택한 후 [저장(Save)]을 선택합니다.

명령줄을 사용하여 라우팅 전파를 활성화하려면

- [enable-vgw-route-propagation](#)(AWS CLI)
- [Enable-EC2VgwRoutePropagation](#)(AWS Tools for Windows PowerShell)

콘솔을 사용하여 라우팅 전파를 비활성화하는 방법

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블(Route Tables)를 선택한 후 라우팅 테이블을 선택합니다.
3. 작업, Edit route propagation(라우팅 속성 편집)을 선택합니다.
4. 가상 프라이빗 게이트웨이 옆에 있는 활성화(Enable) 확인란을 선택 취소한 후 저장(Save)을 선택합니다.

명령줄을 사용하여 정적 라우팅을 비활성화하려면

- [disable-vgw-route-propagation](#)(AWS CLI)
- [Disable-EC2VgwRoutePropagation](#)(AWS Tools for Windows PowerShell)

서브넷을 라우팅 테이블과 연결

특정 서브넷에 라우팅 테이블의 경로를 적용하려면 라우팅 테이블을 서브넷과 연결해야 합니다. 라우팅 테이블은 여러 서브넷과 연결될 수 있습니다. 그러나 서브넷은 한 번에 하나의 라우팅 테이블에만 연결할 수 있습니다. 테이블과 명시적으로 연결되지 않은 서브넷은 기본적으로 기본 라우팅 테이블과 암시적으로 연결됩니다.

콘솔을 사용하여 라우팅 테이블을 서브넷과 연결하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블(Route Tables)를 선택한 후 라우팅 테이블을 선택합니다.
3. [서브넷 연결(Subnet associations)] 탭에서 [서브넷 연결 편집(Edit subnet associations)]을 선택합니다.

4. 라우팅 테이블과 연결할 서브넷에 대한 확인란을 선택합니다.
5. [연결 저장(Save associations)]을 선택합니다.

명령줄을 사용하여 서브넷을 라우팅 테이블과 연결하려면

- [associate-route-table](#)(AWS CLI)
- [Register-EC2RouteTable](#)(AWS Tools for Windows PowerShell)

서브넷의 라우팅 테이블 변경

서브넷에 대한 라우팅 테이블 연결을 변경할 수 있습니다.

라우팅 테이블을 변경하면 새 라우팅 테이블에 동일한 대상의 동일한 트래픽에 대한 라우팅이 포함되어 있지 않은 경우 서브넷의 기존 연결이 삭제됩니다.

콘솔을 사용하여 서브넷 라우팅 테이블 연결을 변경하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Subnets]를 선택한 후 서브넷을 선택합니다.
3. 라우팅 테이블(Route table) 탭에서 라우팅 테이블 연결 편집(Edit route table association)을 선택합니다.
4. 라우팅 테이블 ID(Route table ID)에서 새 라우팅 테이블을 선택합니다.
5. Save(저장)를 선택합니다.

명령줄을 사용하여 서브넷과 연결된 라우팅 테이블을 변경하려면

- [replace-route-table-association](#)(AWS CLI)
- [Set-EC2RouteTableAssociation](#)(AWS Tools for Windows PowerShell)

라우팅 테이블에서 서브넷의 연결 끊기

서브넷과 라우팅 테이블의 연결을 해제할 수 있습니다. 서브넷에 다른 라우팅 테이블을 명시적으로 연결하지 않는 한 서브넷에는 기본 라우팅 테이블이 암시적으로 연결됩니다.

콘솔을 사용하여 라우팅 테이블에서 서브넷을 연결 해제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.

2. 탐색 창에서 라우팅 테이블(Route Tables)를 선택한 후 라우팅 테이블을 선택합니다.
3. 서브넷 연결(Subnet associations) 탭에서 서브넷 연결 편집(Edit subnet associations)을 선택합니다.
4. 서브넷의 확인란을 선택 취소합니다.
5. [연결 저장(Save associations)]을 선택합니다.

명령줄을 사용하여 라우팅 테이블에서 서브넷을 연결 해제하려면

- [disassociate-route-table](#)(AWS CLI)
- [Unregister-EC2RouteTable](#)(AWS Tools for Windows PowerShell)

기본 라우팅 테이블 교체

VPC에서 어떤 라우팅 테이블이 기본 라우팅 테이블인지를 변경할 수 있습니다.

콘솔을 사용하여 기본 라우팅 테이블을 바꾸려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블(Route Tables)을 선택한 후 새로운 기본 라우팅 테이블을 선택합니다.
3. 작업(Actions), 기본 라우팅 테이블 설정(Set main route table)을 선택합니다.
4. 확인 메시지가 나타나면 **set**을 입력한 다음 확인(OK)을 선택합니다.

명령줄을 사용하여 기본 라우팅 테이블을 바꾸려면

- [replace-route-table-association](#)(AWS CLI)
- [Set-EC2RouteTableAssociation](#)(AWS Tools for Windows PowerShell)

다음 절차에서는 서브넷과 기본 라우팅 테이블 간의 명시적 연결을 제거하는 방법을 설명합니다. 결과적으로, 서브넷과 기본 라우팅 테이블 사이에 암시적 연결이 설정됩니다. 이 프로세스는 임의의 라우팅 테이블에서 서브넷의 연결을 끊는 프로세스와 같습니다.

기본 라우팅 테이블과의 명시적 연결을 제거하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블(Route Tables)를 선택한 후 라우팅 테이블을 선택합니다.

3. 서브넷 연결(Subnet associations) 탭에서 서브넷 연결 편집(Edit subnet associations)을 선택합니다.
4. 서브넷에 대한 확인란을 선택 취소합니다.
5. [연결 저장(Save associations)]을 선택합니다.

게이트웨이를 라우팅 테이블과 연결

인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이를 라우팅 테이블과 연결할 수 있습니다. 자세한 내용은 [게이트웨이 라우팅 테이블](#) 단원을 참조하십시오.

콘솔을 사용하여 게이트웨이를 라우팅 테이블과 연결하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블(Route Tables)를 선택한 후 라우팅 테이블을 선택합니다.
3. 엣지 연결(Edge associations) 탭에서 엣지 연결 편집(Edit edge associations)을 선택합니다.
4. 게이트웨이에 대한 확인란을 선택합니다.
5. Save changes(변경 사항 저장)를 선택합니다.

AWS CLI를 사용하여 게이트웨이를 라우팅 테이블과 연결하려면

[associate-route-table](#) 명령을 사용합니다. 다음 예제에서는 인터넷 게이트웨이 igw-11aa22bb33cc44dd1을 라우팅 테이블 rtb-01234567890123456과 연결합니다.

```
aws ec2 associate-route-table --route-table-id rtb-01234567890123456 --gateway-id igw-11aa22bb33cc44dd1
```

라우팅 테이블에서 게이트웨이 연결 해제

라우팅 테이블에서 인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이의 연결을 해제할 수 있습니다.

콘솔을 사용하여 게이트웨이를 라우팅 테이블과 연결하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블(Route Tables)를 선택한 후 라우팅 테이블을 선택합니다.
3. 엣지 연결(Edge associations) 탭에서 엣지 연결 편집(Edit edge associations)을 선택합니다.
4. 게이트웨이의 확인란을 선택 취소합니다.
5. Save changes(변경 사항 저장)를 선택합니다.

명령줄을 사용하여 라우팅 테이블에서 게이트웨이를 연결 해제하려면

- [disassociate-route-table](#)(AWS CLI)
- [Unregister-EC2RouteTable](#)(AWS Tools for Windows PowerShell)

로컬 경로의 대상 교체 또는 복원

기본 로컬 경로의 대상을 변경할 수 있습니다. 로컬 라우팅의 대상을 교체하면 나중에 기본 local 대상으로 복원할 수 있습니다. VPC에 [여러 CIDR 블록](#)이 있는 경우 라우팅 테이블에 CIDR 블록당 하나씩 여러 개의 로컬 라우팅이 있습니다. 필요에 따라 각 로컬 라우팅의 대상을 교체하거나 복원할 수 있습니다.

콘솔을 사용하여 로컬 라우팅을 업데이트하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블(Route Tables)를 선택한 후 라우팅 테이블을 선택합니다.
3. 라우팅(Routes) 탭에서 라우팅 편집(Edit routes)을 선택합니다.
4. 로컬 라우팅에서 대상(Target)을 지운 다음 새로운 대상을 선택합니다.
5. Save changes(변경 사항 저장)를 선택합니다.

콘솔을 사용하여 로컬 라우팅의 대상을 복원하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블(Route Tables)를 선택한 후 라우팅 테이블을 선택합니다.
3. 작업(Actions), Edit routes(라우팅 편집)를 선택합니다.
4. 라우팅에서 대상(Target)을 지운 다음 로컬(local)을 선택합니다.
5. Save changes(변경 사항 저장)를 선택합니다.

AWS CLI를 사용하여 로컬 라우팅의 대상을 대체하려면

[replace-route](#) 명령을 사용합니다. 다음 예제에서는 로컬 라우팅의 대상을 eni-11223344556677889로 바꿉니다.

```
aws ec2 replace-route --route-table-id rtb-01234567890123456 --destination-cidr-block 10.0.0.0/16 --network-interface-id eni-11223344556677889
```

AWS CLI를 사용하여 로컬 라우팅의 대상을 복원하려면

다음 예제에서는 라우팅 테이블 `rtb-01234567890123456`의 로컬 대상을 복원합니다.

```
aws ec2 replace-route --route-table-id rtb-01234567890123456 --destination-cidr-block
10.0.0.0/16 --local-target
```

라우팅 테이블 삭제

라우팅 테이블과 연결된 서브넷이 없는 경우에만 라우팅 테이블을 삭제할 수 있습니다. 기본 라우팅 테이블은 삭제할 수 없습니다.

콘솔을 사용하여 라우팅 테이블을 삭제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블(Route Tables)를 선택한 후 라우팅 테이블을 선택합니다.
3. 작업, 라우팅 테이블 삭제를 차례로 선택합니다.
4. 확인 메시지가 나타나면 **delete**를 입력한 다음 삭제>Delete)를 선택합니다.

명령줄을 사용하여 라우팅 테이블을 삭제하려면

- [delete-route-table](#)(AWS CLI)
- [Remove-EC2RouteTable](#)(AWS Tools for Windows PowerShell)

미들박스 라우팅 마법사

트래픽을 보안 어플라이언스로 리디렉션하는 것과 같이 VPC로 들어오고 나가는 트래픽 라우팅 경로에 세분화된 제어를 구성하려는 경우 VPC 콘솔에서 미들박스 라우팅 마법사를 사용할 수 있습니다. 미들박스 라우팅 마법사는 필요에 따라 트래픽을 리디렉션하는 데 필요한 라우팅 테이블과 라우팅(흡)을 자동으로 생성하도록 도와줍니다.

미들박스 라우팅 마법사는 다음과 같은 시나리오에 대한 라우팅을 구성하는 데 도움을 줄 수 있습니다.

- 미들박스 어플라이언스(예: 보안 어플라이언스로 구성된 Amazon EC2 인스턴스)로 트래픽을 라우팅합니다.
- Gateway Load Balancer 엔드포인트로 라우팅합니다. 자세한 내용은 [로드 밸런서 게이트웨이 사용 설명서](#)를 참조하십시오.

자세한 내용은 [the section called “미들박스 시나리오”](#) 단원을 참조하십시오.

목차

- [미들박스 라우팅 마법사 사전 조건](#)
- [미들박스 경로 관리](#)
- [미들박스 라우팅 마법사 고려 사항](#)
- [미들박스 시나리오](#)

미들박스 라우팅 마법사 사전 조건

[the section called “미들박스 라우팅 마법사 고려 사항”](#) 섹션을 검토합니다. 그런 다음 미들박스 라우팅 마법사를 사용하기 전에 다음 정보가 있는지 확인합니다.

- VPC.
- 인터넷 게이트웨이, 가상 프라이빗 게이트웨이 또는 네트워크 인터페이스와 같은 VPC에서 트래픽이 발생하거나 트래픽이 시작되는 리소스입니다.
- 미들박스 네트워크 인터페이스 또는 Gateway Load Balancer 엔드포인트
- 트래픽에 대한 대상 서브넷입니다.

미들박스 경로 관리

미들박스 라우팅 마법사는 Amazon Virtual Private Cloud Console에서 사용할 수 있습니다.

내용

- [미들박스 라우팅 마법사를 사용하여 경로 생성](#)
- [미들박스 경로 수정](#)
- [미들박스 라우팅 마법사 라우팅 테이블 보기](#)
- [미들박스 라우팅 마법사 구성 삭제](#)

미들박스 라우팅 마법사를 사용하여 경로 생성

미들박스 라우팅 마법사를 사용하여 경로를 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 사용자 VPC(Your VPCs)를 선택합니다.

3. VPC 선택한 다음작업(Actions), 미들박스 경로 관리(Manage middlebox routes)를 선택합니다.
4. 경로 생성(Create route)을 선택합니다.
5. 세부 정보 지정(Specify routes) 페이지에서 다음 작업을 수행합니다.
 - 소스(Source)에서 트래픽의 소스를 선택합니다. 가상 프라이빗 게이트웨이를 선택한 경우 대상 IPv4 CIDR(Destination IPv4 CIDR)에 가상 프라이빗 게이트웨이에서 VPC로 들어오는 온프레미스 트래픽에 대한 CIDR을 입력합니다.
 - 미들박스(Middlebox)에서 미들박스 어플라이언스와 연결된 네트워크 인터페이스 ID를 선택하거나 Gateway Load Balancer 엔드포인트를 사용하는 경우 VPC 엔드포인트 ID를 선택합니다.
 - 대상 서브넷(Destination subnet)에서 대상 서브넷을 선택합니다.
6. (선택 사항) 다른 대상 서브넷을 추가하려면 서브넷 추가(Add additional subnet)를 선택하고 다음 중 하나를 수행합니다.
 - 미들박스(Middlebox)에서 미들박스 어플라이언스와 연결된 네트워크 인터페이스 ID를 선택하거나 Gateway Load Balancer 엔드포인트를 사용하는 경우 VPC 엔드포인트 ID를 선택합니다.
 - 여러 서브넷에 대해 동일한 미들박스 어플라이언스를 사용해야 합니다.
 - 대상 서브넷(Destination subnet)에서 대상 서브넷을 선택합니다.
7. (선택 사항) 다른 소스를 추가하려면 소스 추가(Add source)를 선택한 다음 이전 단계를 반복합니다.
8. Next(다음)를 선택합니다.
9. 검토 및 생성(Review and create) 페이지에서 경로를 확인한 다음 경로 생성(Create routes)을 선택합니다.

미들박스 경로 수정

게이트웨이, 미들박스 또는 대상 서브넷을 변경하여 경로 구성을 편집할 수 있습니다.

수정 사항이 있으면 미들박스 라우팅 마법사는 다음의 작업을 자동으로 수행합니다.

- 게이트웨이, 미들박스 및 대상 서브넷에 대한 새 라우팅 테이블을 생성합니다.
- 필요한 경로를 새 라우팅 테이블에 추가합니다.
- 미들박스 라우팅 마법사가 리소스와 연결된 현재 라우팅 테이블을 연결 해제합니다.
- 미들박스 라우팅 마법사가 리소스를 사용해 생성한 새 라우팅 테이블을 연결합니다.

미들박스 라우팅 마법사를 사용하여 미들박스 경로를 수정하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 사용자 VPC(Your VPCs)를 선택합니다.
3. VPC 선택한 다음작업(Actions), 미들박스 경로 관리(Manage middlebox routes)를 선택합니다.
4. 라우팅 편집(Edit routes)을 선택합니다.
5. 게이트웨이를 변경하려면 소스(Source)에서 트래픽이 VPC로 들어가는 게이트웨이를 선택합니다. 가상 프라이빗 게이트웨이를 선택한 경우 대상 IPv4 CIDR(Destination IPv4 CIDR)에 대상 서브넷 CIDR을 입력합니다.
6. 다른 대상 서브넷을 추가하려면 서브넷 추가(Add additional subnet)를 선택하고 다음 중 하나를 수행합니다.
 - 미들박스(Middlebox)에서 미들박스 어플라이언스와 연결된 네트워크 인터페이스 ID를 선택하거나 Gateway Load Balancer 엔드포인트를 사용하는 경우 VPC 엔드포인트 ID를 선택합니다.

여러 서브넷에 대해 동일한 미들박스 어플라이언스를 사용해야 합니다.

 - 대상 서브넷(Destination subnet)에서 대상 서브넷을 선택합니다.
7. Next(다음)를 선택합니다.
8. 검토 및 업데이트(Review and update) 페이지에서 미들박스 라우팅 마법사에 의해 생성될 라우팅 테이블 및 경로 목록이 표시됩니다. 경로를 확인한 다음 확인 대화 상자에서 경로 업데이트(Update routes)를 선택합니다.

미들박스 라우팅 마법사 라우팅 테이블 보기

미들박스 라우팅 마법사 라우팅 테이블을 보려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 사용자 VPC(Your VPCs)를 선택합니다.
3. VPC 선택한 다음작업(Actions), 미들박스 경로 관리(Manage middlebox routes)를 선택합니다.
4. 미들박스 라우팅 테이블(Middlebox route tables) 아래에서 숫자는 미들박스 라우팅 마법사가 생성한 경로 수를 나타냅니다. 경로를 볼 숫자를 선택합니다.

별도의 라우팅 테이블 페이지에 미들박스 라우팅 마법사 경로를 표시합니다.

미들박스 라우팅 마법사 구성 삭제

미들박스 라우팅 마법사 구성을 더 이상 사용하지 않기로 결정한 경우 라우팅 테이블을 수동으로 삭제해야 합니다.

미들박스 라우팅 마법사 구성을 삭제하려면

1. 미들박스 라우팅 마법사 라우팅 테이블을 봅니다. 자세한 내용은 [the section called “미들박스 라우팅 마법사 라우팅 테이블 보기”](#) 단원을 참조하세요.
작업을 수행하고 나면 미들박스 라우팅 마법사가 생성한 라우팅 테이블이 별도의 라우팅 테이블 페이지에 표시됩니다.
2. 표시된 각 라우팅 테이블을 삭제합니다. 자세한 내용은 [the section called “라우팅 테이블 삭제”](#) 단원을 참조하십시오.

미들박스 라우팅 마법사 고려 사항

미들박스 라우팅 마법사를 사용할 때는 다음 사항을 고려합니다.

- 트래픽을 검사하려면 소스로 인터넷 게이트웨이 또는 가상 프라이빗 게이트웨이를 사용할 수 있습니다.
- 동일한 VPC 내의 여러 미들박스 구성에서 동일한 미들박스를 사용하는 경우 미들박스가 두 서브넷에 대해 동일한 홉 위치에 있는지 확인합니다.
- 어플라이언스는 원본 또는 대상 서브넷과는 별도의 서브넷에 구성되어야 합니다.
- 어플라이언스에서 원본/대상 확인을 비활성화해야 합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [소스 또는 대상 변경 확인](#)을 참조하세요.
- 미들박스 라우팅 마법사가 생성하는 라우팅 테이블과 경로는 할당량에 포함됩니다. 자세한 내용은 [the section called “라우팅 테이블”](#) 단원을 참조하십시오.
- 리소스(예: 네트워크 인터페이스)를 삭제하면 리소스와 라우팅 테이블과의 연결이 제거됩니다. 리소스가 대상이면 경로 대상이 블랙홀로 설정됩니다. 라우팅 테이블은 삭제되지 않습니다.
- 미들박스 서브넷 및 대상 서브넷은 기본이 아닌 라우팅 테이블과 연결되어야 합니다.

Note

미들박스 라우팅 마법사를 사용하여 생성한 라우팅 테이블을 수정하거나 삭제할 때에는 미들박스 라우팅 마법사를 사용하는 것이 좋습니다.

미들박스 시나리오

다음 예에서는 미들박스 라우팅 마법사의 시나리오를 설명합니다.

내용

- [서브넷을 대상으로 하는 트래픽 검사](#)
- [보안 VPC에서 어플라이언스를 사용하여 트래픽 검사](#)
- [서브넷 간 트래픽 검사](#)

서브넷을 대상으로 하는 트래픽 검사

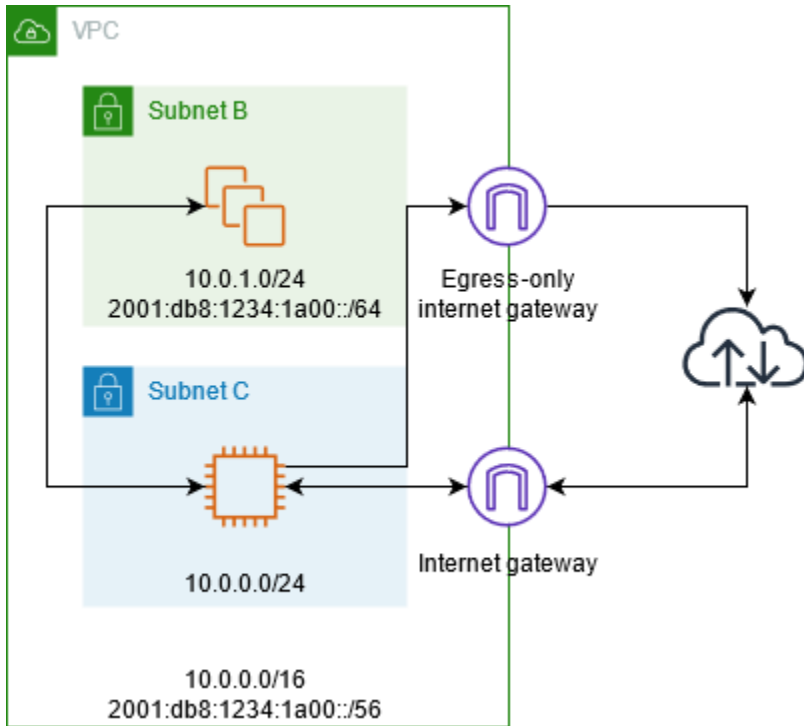
인터넷 게이트웨이를 통해 VPC 로 들어오는 트래픽이 있고, EC2 인스턴스에 설치된 방화벽 어플라이언스를 사용하여 서브넷(서브넷 B로 가정)을 대상으로 하는 모든 트래픽을 검사하려는 시나리오를 생각해봅시다. 방화벽 어플라이언스는 VPC의 서브넷 B와는 별도인 서브넷(서브넷 C로 가정)의 EC2 인스턴스에 설치 및 구성해야 합니다. 그런 다음 미들박스 라우팅 마법사를 사용하여 서브넷 B와 인터넷 게이트웨이 간의 트래픽 경로를 구성할 수 있습니다.

미들박스 라우팅 마법사는 다음의 작업을 자동으로 수행합니다.

- 다음의 라우팅 테이블을 생성합니다.
 - 인터넷 게이트웨이의 라우팅 테이블
 - 대상 서브넷의 라우팅 테이블
 - 미들박스 서브넷의 라우팅 테이블
- 다음 단원에 설명된 내용에 따라 필요한 라우팅을 새 라우팅 테이블에 추가합니다.
- 인터넷 게이트웨이, 서브넷 B 및 서브넷 C와 연결된 현재 라우팅 테이블의 연결을 해제합니다.
- 라우팅 테이블 A를 인터넷 게이트웨이(미들박스 라우팅 마법사의 소스(Source))와 연결하고 라우팅 테이블 C를 서브넷 C(미들박스 라우팅 마법사의 미들박스(Middlebox))와 연결하고 라우팅 테이블 B를 서브넷 B(미들박스 라우팅 마법사의 대상(Destination))과 연결합니다.
- 미들박스 라우팅 마법사에 의해 생성되었음을 나타내는 태그와 생성 날짜를 나타내는 태그를 생성합니다.

미들박스 라우팅 마법사는 기존 라우팅 테이블을 수정하지 않습니다. 새 라우팅 테이블을 생성한 다음 게이트웨이 및 서브넷 리소스와 연결합니다. 리소스가 이미 기존 라우팅 테이블과 명시적으로 연결되어 있는 경우 기존 라우팅 테이블의 연결이 우선적으로 해제된 다음 새 라우팅 테이블이 리소스와 연결됩니다. 기존 라우팅 테이블은 삭제되지 않습니다.

미들박스 라우팅 마법사를 사용하지 않는 경우 서브넷과 인터넷 게이트웨이에 라우팅 테이블을 수동으로 구성한 다음 할당해야 합니다.



인터넷 게이트웨이 라우팅 테이블

인터넷 게이트웨이의 라우팅 테이블에 다음의 경로를 추가합니다.

대상 주소	대상	용도
<i>10.0.0.0/16</i>	로컬	IPv4에 대한 로컬 경로
<i>10.0.1.0/24</i>	<i>appliance-eni</i>	서브넷 B를 대상으로 하는 IPv4 트래픽을 미들박스로 라우팅
<i>2001:db8:1234:1a00::/56</i>	로컬	IPv6에 대한 로컬 경로
<i>2001:db8:1234:1a00::/64</i>	<i>appliance-eni</i>	서브넷 B를 대상으로 하는 IPv6 트래픽을 미들박스로 라우팅

인터넷 게이트웨이와 VPC 간에는 엣지 연결이 있습니다.

미들박스 라우팅 마법사를 사용하면 다음의 태그가 라우팅 테이블과 연결됩니다.

- 키는 “Origin”이고 값은 “미들박스 마법사”입니다.
- 키는 “date_created”이고 값은 생성 시간입니다(예: “2021-02-18T22:25:49.137Z”)

대상 서브넷 라우팅 테이블

대상 서브넷(예제 다이어그램의 서브넷 B)에 대한 라우팅 테이블에 다음 경로를 추가합니다.

대상 주소	대상	용도
<i>10.0.0.0/16</i>	로컬	IPv4에 대한 로컬 경로
0.0.0.0/0	<i>appliance-eni</i>	인터넷을 대상으로 하는 IPv4 트래픽을 미들박스로 라우팅
<i>2001:db8:1234:1a00::/56</i>	로컬	IPv6에 대한 로컬 경로
:::0	<i>appliance-eni</i>	인터넷을 대상으로 하는 IPv6 트래픽을 미들박스로 라우팅

미들박스 서브넷과의 서브넷 연결이 있습니다.

미들박스 라우팅 마법사를 사용하면 다음의 태그가 라우팅 테이블과 연결됩니다.

- 키는 “Origin”이고 값은 “미들박스 마법사”입니다.
- 키는 “date_created”이고 값은 생성 시간입니다(예: “2021-02-18T22:25:49.137Z”)

미들박스 서브넷 라우팅 테이블

대상 서브넷(예제 다이어그램의 서브넷 C)에 대한 라우팅 테이블에 다음 경로를 추가합니다.

대상 주소	대상	용도
<i>10.0.0.0/16</i>	로컬	IPv4에 대한 로컬 경로
0.0.0.0/0	<i>igw-id</i>	IPv4 트래픽을 인터넷 게이트웨이로 라우팅

대상 주소	대상	용도
<code>2001:db8:1234:1a00::/56</code>	로컬	IPv6에 대한 로컬 경로
<code>::/0</code>	<code>eigw-id</code>	송신 전용 인터넷 게이트웨이로 IPv6 트래픽 라우팅

대상 서브넷과의 서브넷 연결이 있습니다.

미들박스 라우팅 마법사를 사용하면 다음의 태그가 라우팅 테이블과 연결됩니다.

- 키는 “Origin”이고 값은 “미들박스 마법사”입니다.
- 키는 “date_created”이고 값은 생성 시간입니다(예: “2021-02-18T22:25:49.137Z”)

보안 VPC에서 어플라이언스를 사용하여 트래픽 검사

Gateway Load Balancer 뒤에 구성된 보안 어플라이언스 플릿을 사용하여 인터넷 게이트웨이에서 VPC로 들어오고 서브넷을 대상으로 하는 트래픽을 검사해야 하는 시나리오를 고려합니다. 서비스 소비자 VPC의 소유자는 VPC의 서브넷에 Gateway Load Balancer 엔드포인트를 생성합니다(엔드포인트 네트워크 인터페이스로 표시됨). 인터넷 게이트웨이를 통해 VPC로 들어오는 모든 트래픽은 먼저 검사할 수 있도록 Gateway Load Balancer 엔드포인트로 라우팅된 후 애플리케이션 서브넷으로 라우팅됩니다. 마찬가지로 애플리케이션 서브넷에서 나가는 모든 트래픽은 검사할 수 있도록 먼저 Gateway Load Balancer 엔드포인트로 라우팅된 후 인터넷으로 라우팅됩니다.

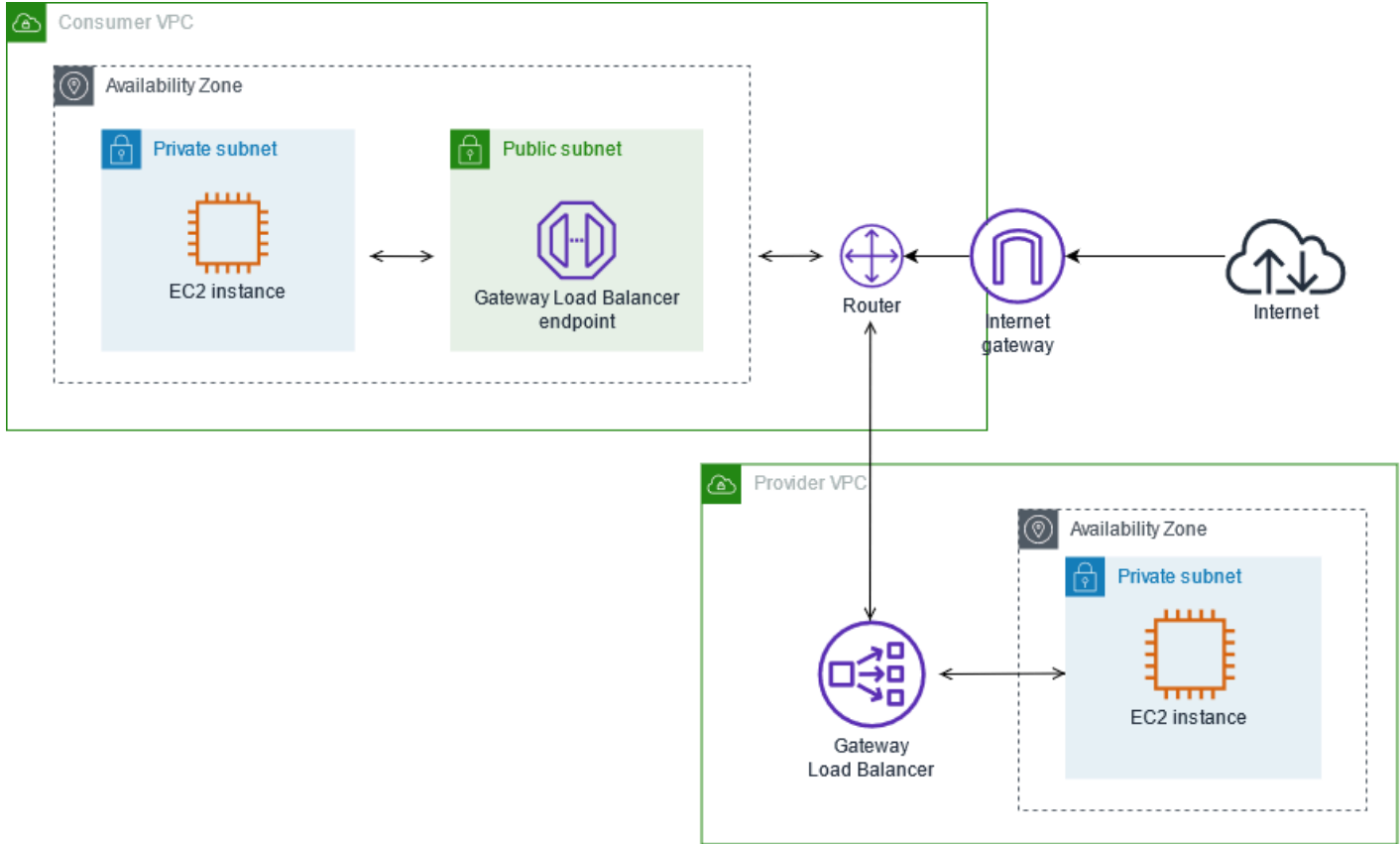
미들박스 라우팅 마법사는 다음의 작업을 자동으로 수행합니다.

- 라우팅 테이블을 생성합니다.
- 필요한 경로를 새 라우팅 테이블에 추가합니다.
- 서브넷과 연결된 현재 라우팅 테이블의 연결을 해제합니다.
- 미들박스 라우팅 마법사가 생성하는 라우팅 테이블을 서브넷과 연결합니다.
- 미들박스 라우팅 마법사에 의해 생성되었음을 나타내는 태그와 생성 날짜를 나타내는 태그를 생성합니다.

미들박스 라우팅 마법사는 기존 라우팅 테이블을 수정하지 않습니다. 새 라우팅 테이블을 생성한 다음 게이트웨이 및 서브넷 리소스와 연결합니다. 리소스가 이미 기존 라우팅 테이블과 명시적으로 연결되

어 있는 경우 기존 라우팅 테이블의 연결이 우선적으로 해제된 다음 새 라우팅 테이블이 리소스와 연결됩니다. 기존 라우팅 테이블은 삭제되지 않습니다.

미들박스 라우팅 마법사를 사용하지 않는 경우 서브넷과 인터넷 게이트웨이에 라우팅 테이블을 수동으로 구성한 다음 할당해야 합니다.



인터넷 게이트웨이 라우팅 테이블

인터넷 게이트웨이의 라우팅 테이블에는 다음의 경로가 포함됩니다.

대상 주소	대상	용도
<i>### VPC CIDR</i>	로컬	로컬 경로
<i>##### ### CIDR</i>	<i>endpoint-id</i>	애플리케이션 서브넷을 대상으로 하는 트래픽을 Gateway Load Balancer 엔드포인트로 라우팅합니다.

게이트웨이와의 엣지 연결이 있습니다.

미들박스 라우팅 마법사를 사용하면 다음의 태그가 라우팅 테이블과 연결됩니다.

- 키는 “Origin”이고 값은 “미들박스 마법사”입니다.
- 키는 “date_created”이고 값은 생성 시간입니다(예: “2021-02-18T22:25:49.137Z”)

애플리케이션 서브넷 라우팅 테이블

애플리케이션 서브넷의 라우팅 테이블에는 다음과 같은 경로가 있습니다.

대상 주소	대상	용도
<i>### VPC CIDR</i>	로컬	로컬 경로
0.0.0.0/0	<i>endpoint-id</i>	애플리케이션 서버에서 트래픽이 인터넷으로 라우팅되기 전에 Gateway Load Balancer 엔드포인트로 라우팅합니다.

미들박스 라우팅 마법사를 사용하면 다음의 태그가 라우팅 테이블과 연결됩니다.

- 키는 “Origin”이고 값은 “미들박스 마법사”입니다.
- 키는 “date_created”이고 값은 생성 시간입니다(예: “2021-02-18T22:25:49.137Z”)

공급자 서브넷 라우팅 테이블

공급자 서브넷의 라우팅 테이블에는 다음과 같은 경로가 있습니다.

대상 주소	대상	용도
<i>### VPC CIDR</i>	로컬	로컬 경로 인터넷에서 시작된 트래픽이 애플리케이션 서버로 라우팅되게 합니다.
0.0.0.0/0	<i>igw-id</i>	모든 트래픽을 인터넷 게이트웨이로 라우팅

미들박스 라우팅 마법사를 사용하면 다음의 태그가 라우팅 테이블과 연결됩니다.

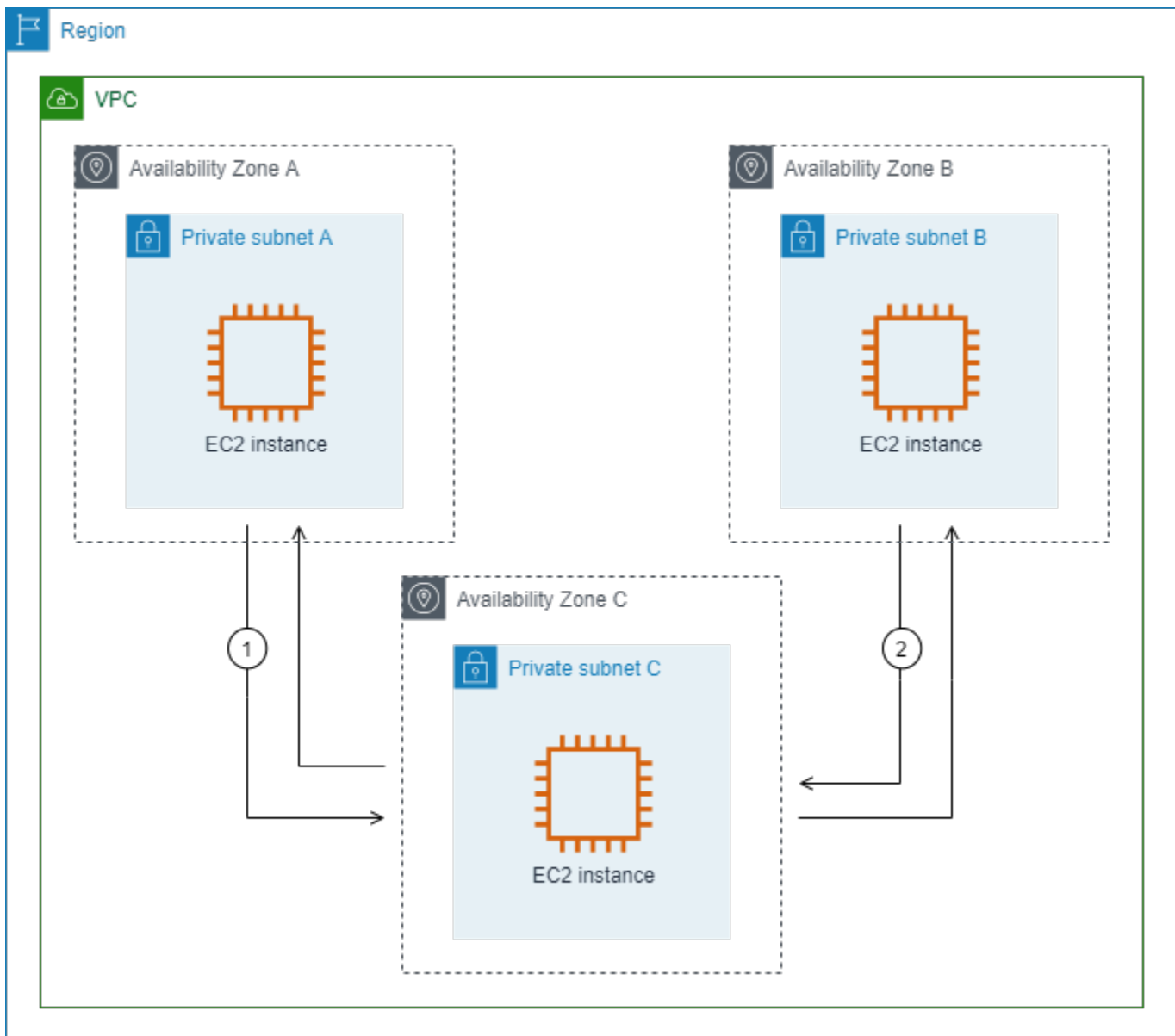
- 키는 “Origin”이고 값은 “미들박스 마법사”입니다.

- 키는 “date_created”이고 값은 생성 시간입니다(예: “2021-02-18T22:25:49.137Z”)

서브넷 간 트래픽 검사

VPC에 여러 개의 서브넷이 있고 방화벽 어플라이언스를 통해 해당 서브넷 간의 트래픽을 검사하려는 시나리오를 생각해봅시다. VPC에 별도의 서브넷에 있는 EC2 인스턴스에 방화벽 어플라이언스를 구성하고 설치합니다.

다음 다이어그램은 서브넷 C의 EC2 인스턴스에 설치된 방화벽 어플라이언스를 보여줍니다. 해당 어플라이언스는 서브넷 A에서 서브넷 B로(1 참조), 서브넷 B에서 서브넷 A(2 참조)로 이동하는 트래픽을 모두 검사합니다.



VPC 및 미들박스 서브넷에 대한 기본 라우팅 테이블을 사용합니다. 서브넷 A와 B에는 각각 사용자 지정 라우팅 테이블이 있습니다.

미들박스 라우팅 마법사는 다음의 작업을 자동으로 수행합니다.

- 라우팅 테이블을 생성합니다.
- 필요한 경로를 새 라우팅 테이블에 추가합니다.
- 서브넷과 연결된 현재 라우팅 테이블의 연결을 해제합니다.
- 미들박스 라우팅 마법사가 생성하는 라우팅 테이블을 서브넷과 연결합니다.
- 미들박스 라우팅 마법사에 의해 생성되었음을 나타내는 태그와 생성 날짜를 나타내는 태그를 생성합니다.

미들박스 라우팅 마법사는 기존 라우팅 테이블을 수정하지 않습니다. 새 라우팅 테이블을 생성한 다음 게이트웨이 및 서브넷 리소스와 연결합니다. 리소스가 이미 기존 라우팅 테이블과 명시적으로 연결되어 있는 경우 기존 라우팅 테이블의 연결이 우선적으로 해제된 다음 새 라우팅 테이블이 리소스와 연결됩니다. 기존 라우팅 테이블은 삭제되지 않습니다.

미들박스 라우팅 마법사를 사용하지 않는 경우 서브넷과 인터넷 게이트웨이에 라우팅 테이블을 수동으로 구성한 다음 할당해야 합니다.

서브넷 A에 대한 사용자 지정 라우팅 테이블

서브넷 A의 라우팅 테이블에는 다음과 같은 경로가 있습니다.

대상 주소	대상	용도
<i>VPC CIDR</i>	로컬	로컬 경로
<i>### B CIDR</i>	<i>appliance-eni</i>	서브넷 B를 대상으로 하는 트래픽을 미들박스로 라우팅

미들박스 라우팅 마법사를 사용하면 다음의 태그가 라우팅 테이블과 연결됩니다.

- 키는 “Origin”이고 값은 “미들박스 마법사”입니다.
- 키는 “date_created”이고 값은 생성 시간입니다(예: “2021-02-18T22:25:49.137Z”)

서브넷 B에 대한 사용자 지정 라우팅 테이블

서브넷 B의 라우팅 테이블에는 다음과 같은 경로가 있습니다.

대상 주소	대상	용도
<i>VPC CIDR</i>	로컬	로컬 경로
<i>### A CIDR</i>	<i>appliance-eni</i>	서브넷 A를 대상으로 하는 트래픽을 미들박스로 라우팅

미들박스 라우팅 마법사를 사용하면 다음의 태그가 라우팅 테이블과 연결됩니다.

- 키는 “Origin”이고 값은 “미들박스 마법사”입니다.
- 키는 “date_created”이고 값은 생성 시간입니다(예: “2021-02-18T22:25:49.137Z”)

기본 라우팅 테이블

서브넷 C는 기본 라우팅 테이블을 사용합니다. 기본 라우팅 테이블에는 다음과 같은 경로가 있습니다.

대상 주소	대상	용도
<i>VPC CIDR</i>	로컬	로컬 경로

미들박스 라우팅 마법사를 사용하면 다음의 태그가 라우팅 테이블과 연결됩니다.

- 키는 “Origin”이고 값은 “미들박스 마법사”입니다.
- 키는 “date_created”이고 값은 생성 시간입니다(예: “2021-02-18T22:25:49.137Z”)

서브넷 삭제

서브넷이 더 이상 필요하지 않으면 삭제할 수 있습니다. 네트워크 인터페이스가 포함된 서브넷은 삭제할 수 없습니다. 예를 들어 서브넷의 모든 인스턴스를 종료해야 서브넷을 삭제할 수 있습니다.

콘솔을 사용하여 서브넷을 삭제하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 서브넷의 모든 인스턴스를 종료합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [인스턴스 종료](#)를 참조하세요.
3. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
4. 탐색 창에서 서브넷(Subnets)을 선택합니다.
5. 서브넷을 선택한 다음 작업(Actions), 서브넷 삭제>Delete subnet)를 선택합니다.
6. 확인 메시지가 나타나면 **delete**를 입력한 다음 삭제>Delete)를 선택합니다.

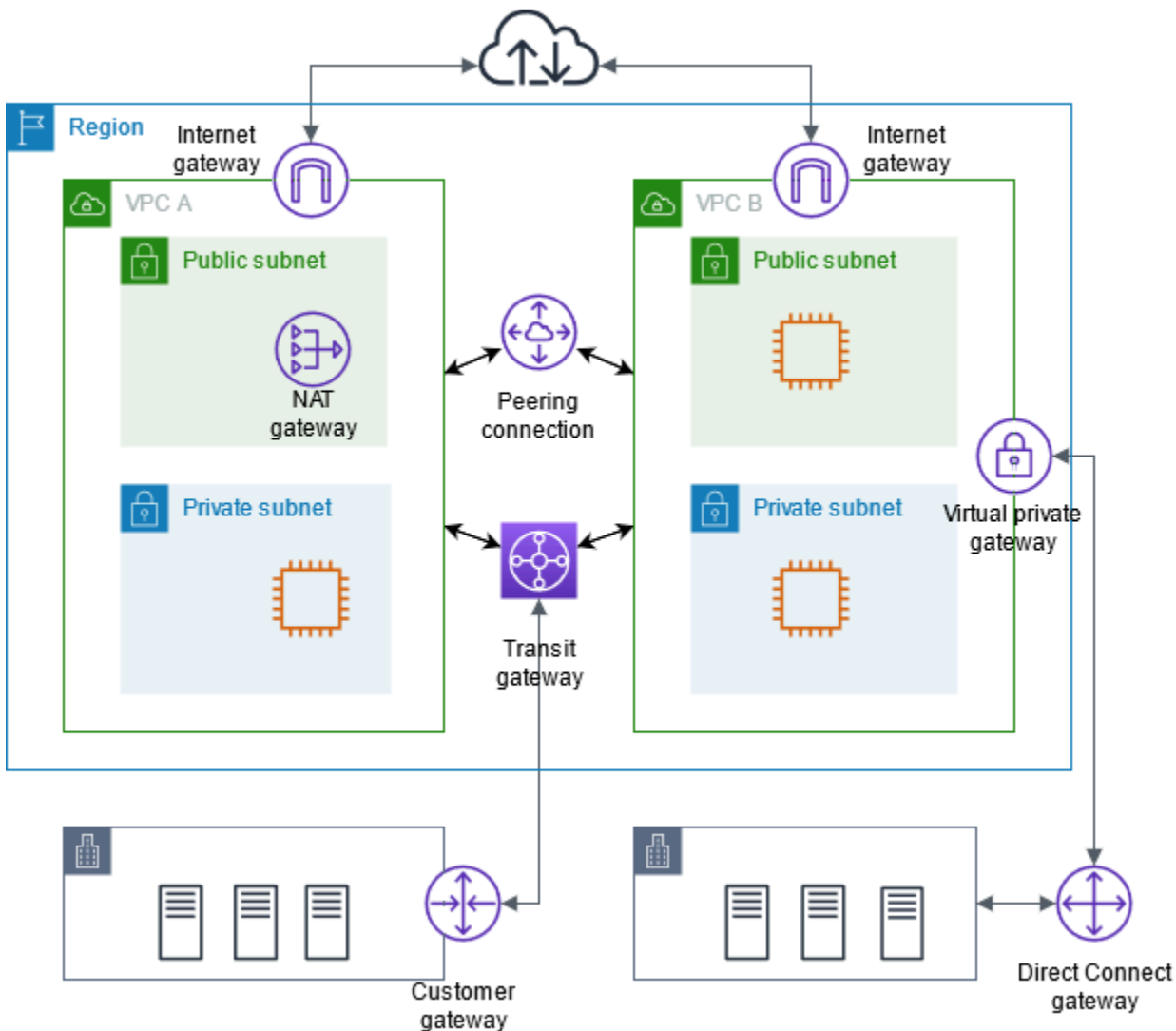
AWS CLI를 사용하여 서브넷을 삭제하려면

[delete-subnet](#) 명령을 사용합니다.

다른 네트워크에 VPC 연결

Virtual Private Cloud(VPC)를 다른 네트워크에 연결할 수 있습니다. 다른 VPC, 인터넷 또는 온프레미스 네트워크를 예로 들 수 있습니다.

다음 다이어그램에서는 이러한 연결 옵션 중 일부를 보여줍니다. VPC A는 인터넷 게이트웨이를 통해 인터넷에 연결됩니다. VPC A의 프라이빗 서브넷에 있는 EC2 인스턴스는 VPC A의 퍼블릭 서브넷에 있는 NAT 게이트웨이를 사용하여 인터넷에 연결할 수 있습니다. VPC B는 인터넷 게이트웨이를 통해 인터넷에 연결됩니다. VPC B의 퍼블릭 서브넷에 있는 EC2 인스턴스는 인터넷 게이트웨이를 사용하여 인터넷에 연결할 수 있습니다. VPC A와 VPC B는 VPC 피어링 연결과 전송 게이트웨이를 통해 서로 연결됩니다. 전송 게이트웨이에는 데이터 센터에 대한 VPN 연결이 있습니다. VPC B에는 데이터 센터에 대한 AWS Direct Connect 연결이 있습니다.



자세한 내용은 [Amazon Virtual Private Cloud 연결 옵션](#)을 참조하세요.

내용

- [인터넷 게이트웨이를 사용하여 인터넷에 연결](#)
- [송신 전용 인터넷 게이트웨이를 사용하여 아웃바운드 IPv6 트래픽 활성화](#)
- [NAT 디바이스를 사용하여 인터넷 또는 다른 네트워크에 연결](#)
- [탄력적 IP 주소를 VPC의 리소스와 연결](#)
- [전송 게이트웨이를 사용하여 다른 VPC 및 네트워크에 VPC 연결](#)
- [AWS Virtual Private Network를 사용하여 VPC를 원격 네트워크에 연결](#)
- [VPC 피어링을 사용하여 VPC 연결](#)

인터넷 게이트웨이를 사용하여 인터넷에 연결

인터넷 게이트웨이는 수평 확장되고 가용성이 높은 중복 VPC 구성 요소로, VPC와 인터넷 간에 통신할 수 있게 해줍니다. IPv4 트래픽 및 IPv6 트래픽을 지원합니다. 네트워크 트래픽에 가용성 위험이나 대역폭 제약이 발생하지 않습니다.

리소스에 퍼블릭 IPv4 주소 또는 IPv6 주소가 있는 경우 인터넷 게이트웨이를 사용하면 퍼블릭 서브넷(예: EC2 인스턴스)의 리소스가 인터넷에 연결할 수 있습니다. 마찬가지로 인터넷의 리소스는 퍼블릭 IPv4 주소 또는 IPv6 주소를 사용하여 서브넷의 리소스에 대한 연결을 시작할 수 있습니다. 예를 들어 인터넷 게이트웨이를 사용하면 로컬 컴퓨터로 AWS의 EC2 인스턴스에 연결할 수 있습니다.

인터넷 게이트웨이는 VPC 라우팅 테이블에서 인터넷 라우팅 가능 트래픽에 대한 대상을 제공합니다. IPv4 통신의 경우 인터넷 게이트웨이는 Network Address Translation(NAT)도 수행합니다. IPv6를 사용하는 통신의 경우 IPv6 주소가 퍼블릭이므로 NAT가 필요하지 않습니다. 자세한 내용은 [IP 주소 및 NAT](#)를 참조하세요.

인터넷 액세스 구성

인스턴스가 인터넷에서 트래픽을 수신하거나 전송할 수 있도록 하려면 다음을 수행하세요.

- [인터넷 게이트웨이를 생성하여 VPC에 연결](#)합니다.
- 인터넷 바인딩 트래픽을 인터넷 게이트웨이로 전달하는 서브넷의 라우팅 테이블에 [경로를 추가](#)합니다.
- 서브넷의 인스턴스에 퍼블릭 IPv4 주소나 IPv6 주소가 있는지 확인합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [인스턴스 IP 주소 지정](#)을 참조하세요.
- [보안 그룹](#) 및 [네트워크 액세스 제어 목록](#)에서 원하는 인터넷 트래픽이 인스턴스로, 그리고 인스턴스에서 흐르도록 허용되는지 확인합니다.

인스턴스에 퍼블릭 IP 주소를 할당하지 않고 인터넷에 액세스할 수 있도록 하려면 대신 NAT 디바이스를 사용하면 됩니다. NAT 디바이스를 사용하면 프라이빗 서브넷의 인스턴스를 인터넷에 연결하는 한편, 인터넷의 호스트가 해당 인스턴스와의 연결을 시작하지 못하도록 할 수 있습니다. 자세한 내용은 [NAT 디바이스](#) 단원을 참조하십시오.

퍼블릭 및 프라이빗 서브넷

서브넷이 인터넷 게이트웨이로 향하는 라우팅이 있는 라우팅 테이블과 연결되는 경우, 이를 퍼블릭 서브넷이라고 합니다. 서브넷이 인터넷 게이트웨이로 향하는 라우팅이 없는 라우팅 테이블과 연결되는 경우 이를 프라이빗 서브넷이라고 합니다.

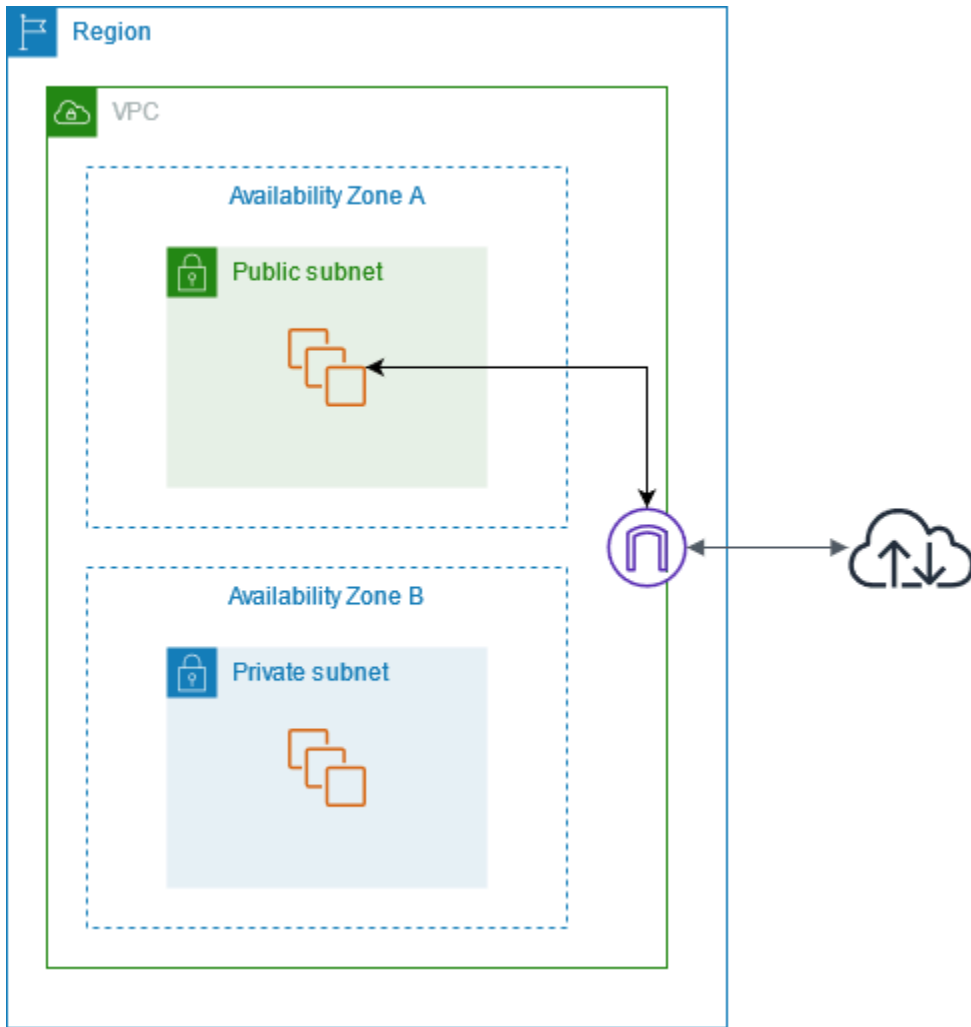
퍼블릭 서브넷의 라우팅 테이블에서 라우팅 테이블에 명시적으로 알려지지 않은 모든 대상에 대한 인터넷 게이트웨이의 라우팅을 지정할 수 있습니다(IPv4의 경우 $0.0.0.0/0$ 또는 IPv6의 경우 $::/0$). 또는 라우팅을 더 좁은 범위의 IP 주소(예: AWS 외부에 있는 회사 퍼블릭 엔드포인트의 퍼블릭 IPv4 주소 또는 VPC 외부에 있는 다른 Amazon EC2 인스턴스의 탄력적 IP 주소)로 지정할 수 있습니다.

IP 주소 및 NAT

IPv4 인터넷 통신이 가능하게 하려면, 인스턴스에 퍼블릭 IPv4 주소가 있어야 합니다. 인스턴스에 퍼블릭 IPv4 주소를 자동 할당하도록 인스턴스에 탄력적 IP 주소를 할당하도록 VPC를 구성할 수 있습니다. 사용자의 인스턴스는 VPC 및 서브넷 내부에서 정의된 프라이빗(내부) IP 주소 공간만 인식합니다. 인터넷 게이트웨이는 사용자의 인스턴스를 대신하여 논리적으로 일대일 NAT를 제공하므로, 트래픽이 VPC 서브넷을 떠나 인터넷으로 이동할 때 회신 주소 필드는 프라이빗 IP 주소가 아니라, 인스턴스의 퍼블릭 IPv4 주소 또는 탄력적 IP 주소(EIP)로 설정됩니다. 반대로, 인스턴스의 퍼블릭 IPv4 주소 또는 탄력적 IP 주소를 대상 주소로 하는 트래픽에는 트래픽이 VPC로 전달되기 전에 인스턴스의 프라이빗 IPv4 주소로 변환되는 대상 주소가 있습니다.

IPv6를 위해 인터넷을 통한 통신을 가능케 하려면, VPC 및 서브넷에 연결된 IPv6 CIDR 블록이 있어야 하고, 서브넷의 범위에 속한 IPv6 주소가 인스턴스에 할당되어야 합니다. IPv6 주소는 전역적으로 고유하므로 퍼블릭으로 기본 설정되어 있습니다.

다음 다이어그램에서 가용 영역 A의 서브넷은 퍼블릭 서브넷입니다. 이 서브넷의 라우팅 테이블에는 인터넷 바운드 IPv4 트래픽을 인터넷 게이트웨이로 전송하는 경로가 있습니다. 퍼블릭 서브넷의 인스턴스에는 퍼블릭 IP 주소 또는 탄력적 IP 주소가 있어서 인터넷 게이트웨이를 통해 인터넷과의 통신을 지원할 수 있어야 합니다. 비교해 보면 가용 영역 B의 서브넷은 라우팅 테이블에 인터넷 게이트웨이에 대한 경로가 없기 때문에 프라이빗 서브넷에 해당합니다. 인터넷 게이트웨이로 향하는 경로가 없기 때문에 퍼블릭 IP 주소가 있더라도 프라이빗 서브넷은 인터넷과 통신할 수 없습니다.



기본 VPC와 기본이 아닌 VPC에 대한 인터넷 액세스

다음 테이블에서 VPC가 IPv4 또는 IPv6를 통한 인터넷 액세스에 필요한 구성 요소와 함께 자동으로 제공되는지를 개괄적으로 제시합니다.

구성 요소	기본 VPC	기본이 아닌 VPC
인터넷 게이트웨이	예	아니요
IPv4 트래픽을 위한 인터넷 게이트웨이로 가는 경로가 있는 라우팅 테이블(0.0.0.0/0)	예	아니요

구성 요소	기본 VPC	기본이 아닌 VPC
IPv6 트래픽을 위한 인터넷 게이트웨이로 가는 경로가 있는 라우팅 테이블(::/0)	아니요	아니요
서브넷에서 시작된 인스턴스에 자동 할당된 퍼블릭 IPv4 주소	예(기본 서브넷)	아니요(기본이 아닌 서브넷)
서브넷에서 시작된 인스턴스에 자동 할당된 IPv6 주소	아니요(기본 서브넷)	아니요(기본이 아닌 서브넷)

기본 VPC에 대한 자세한 내용은 [기본 VPC](#) 단원을 참조하십시오. VPC 생성에 대한 자세한 내용은 [VPC 생성](#) 단원을 참조하십시오.

인터넷 게이트웨이 작업

다음은 인터넷 게이트웨이를 사용하여 VPC의 서브넷에서 인터넷 액세스를 지원하는 방법을 설명합니다. 인터넷 액세스를 제거하려면 VPC에서 인터넷 게이트웨이를 분리한 후 삭제할 수 있습니다.

Tasks

- [인터넷 게이트웨이 생성](#)
- [VPC에 인터넷 게이트웨이 연결](#)
- [VPC에서 인터넷 게이트웨이 분리](#)
- [인터넷 게이트웨이 삭제](#)

인터넷 게이트웨이 생성

인터넷 게이트웨이를 생성하려면 다음 절차를 따르세요.

인터넷 게이트웨이를 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 인터넷 게이트웨이(Internet gateways)를 선택합니다.
3. 인터넷 게이트웨이 생성을 선택합니다.
4. (선택 사항) 인터넷 게이트웨이에 이름을 입력합니다.

5. (선택 사항) 태그를 추가하려면 Add new tag(새 태그 추가)를 선택하고 태그 키와 태그 값을 입력합니다.
6. 인터넷 게이트웨이 생성을 선택합니다.
7. (선택 사항) 지금 인터넷 게이트웨이를 VPC에 연결하려면 화면 상단의 배너에서 VPC에 연결을 선택하고 사용 가능한 VPC를 선택한 다음 인터넷 게이트웨이 연결을 선택합니다. 그렇지 않으면 나중에 인터넷 게이트웨이를 VPC에 연결할 수 있습니다.

VPC에 인터넷 게이트웨이 연결

인터넷 게이트웨이를 사용하려면 이를 VPC에 연결해야 합니다.

인터넷 게이트웨이를 VPC에 연결하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 인터넷 게이트웨이(Internet gateways)를 선택합니다.
3. 해당 인터넷 게이트웨이의 확인란을 선택합니다.
4. 작업, VPC에 연결을 선택합니다.
5. 사용 가능한 VPC를 선택합니다.
6. 인터넷 게이트웨이 연결을 선택합니다.

VPC에서 인터넷 게이트웨이 분리

VPC로 시작하는 인스턴스에 대해 더 이상 인터넷 액세스가 필요하지 않으면, VPC에서 인터넷 게이트웨이를 분리할 수 있습니다. VPC에 퍼블릭 IP 주소 또는 탄력적 IP 주소가 연결된 리소스가 있는 경우에는 인터넷 게이트웨이를 분리할 수 없습니다.

인터넷 게이트웨이 분리

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 인터넷 게이트웨이(Internet gateways)를 선택합니다.
3. 해당 인터넷 게이트웨이의 확인란을 선택합니다.
4. 작업, VPC에서 분리를 선택합니다.
5. 확인 메시지가 나타나면 인터넷 게이트웨이 분리를 선택합니다.

인터넷 게이트웨이 삭제

더 이상 필요하지 않게 된 인터넷 게이트웨이를 삭제할 수 있습니다. 아직 VPC에 연결되어 있는 인터넷 게이트웨이는 삭제할 수 없습니다.

인터넷 게이트웨이 삭제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 인터넷 게이트웨이(Internet gateways)를 선택합니다.
3. 해당 인터넷 게이트웨이의 확인란을 선택합니다.
4. 작업, 인터넷 게이트웨이 삭제를 선택합니다.
5. 확인 메시지가 나타나면 **delete**를 입력한 다음 인터넷 게이트웨이 삭제를 선택합니다.

API 및 명령 개요

명령줄 또는 API를 사용하여 이 페이지에서 설명하는 작업을 수행할 수 있습니다. 명령줄 인터페이스 및 사용 가능한 API 작업 목록에 대한 자세한 내용은 [Amazon VPC 작업](#) 단원을 참조하십시오.

인터넷 게이트웨이 생성

- [create-internet-gateway](#)(AWS CLI)
- [New-EC2InternetGateway](#)(AWS Tools for Windows PowerShell)

VPC에 인터넷 게이트웨이 연결

- [attach-internet-gateway](#)(AWS CLI)
- [Add-EC2InternetGateway](#)(AWS Tools for Windows PowerShell)

인터넷 게이트웨이 설명

- [describe-internet-gateways](#)(AWS CLI)
- [Get-EC2InternetGateway](#)(AWS Tools for Windows PowerShell)

VPC에서 인터넷 게이트웨이 분리

- [detach-internet-gateway](#)(AWS CLI)

- [Dismount-EC2InternetGateway](#)(AWS Tools for Windows PowerShell)

인터넷 게이트웨이 삭제

- [delete-internet-gateway](#)(AWS CLI)
- [Remove-EC2InternetGateway](#)(AWS Tools for Windows PowerShell)

요금

인터넷 게이트웨이에는 요금이 부과되지 않지만 인터넷 게이트웨이를 사용하는 EC2 인스턴스에는 데이터 전송 요금이 부과됩니다. 자세한 내용은 [Amazon EC2 온디맨드 요금](#)을 참조하세요.

송신 전용 인터넷 게이트웨이를 사용하여 아웃바운드 IPv6 트래픽 활성화

외부 전용 인터넷 게이트웨이는 수평 확장되고 가용성이 높은 중복 VPC 구성 요소로서, VPC의 인스턴스에서 인터넷으로 IPv6을 통한 아웃바운드 통신을 가능케 하되 인터넷에서 해당 인스턴스와의 IPv6 연결을 시작하지 못하게 할 수 있습니다.

Note

외부 전용 인터넷 게이트웨이는 IPv6 트래픽에만 사용됩니다. IPv4를 통한 아웃바운드 전용 인터넷 통신을 사용하려면 NAT 게이트웨이를 사용하십시오. 자세한 내용은 [NAT 게이트웨이](#) 단원을 참조하십시오.

목차

- [외부 전용 인터넷 게이트웨이 기본 사항](#)
- [송신 전용 인터넷 게이트웨이 작업](#)
- [API 및 CLI 개요](#)
- [요금](#)

외부 전용 인터넷 게이트웨이 기본 사항

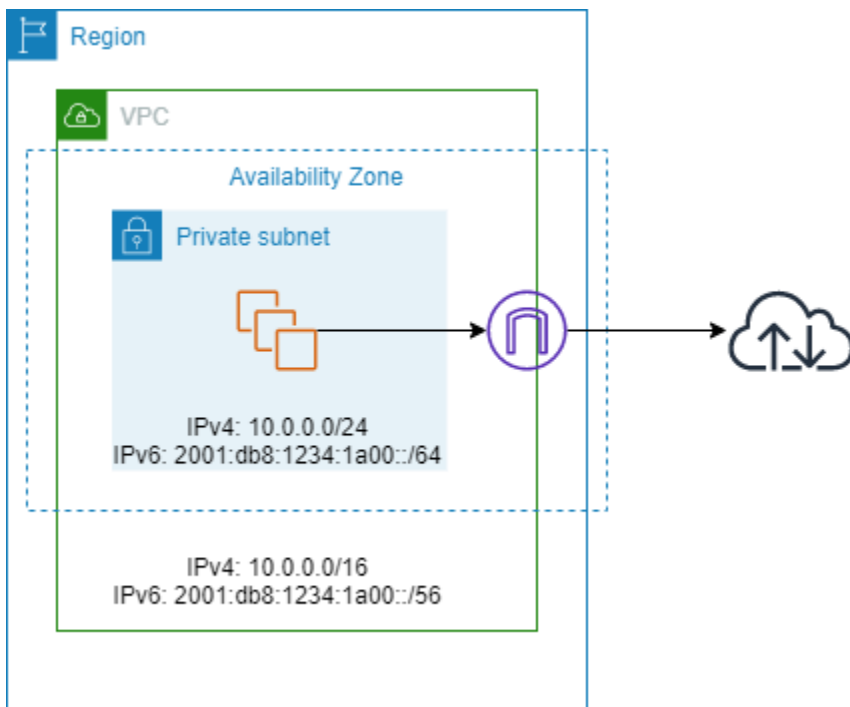
IPv6 주소는 전역적으로 고유하므로 퍼블릭으로 기본 설정되어 있습니다. 인스턴스가 인터넷에 액세스할 수 있게 하되 인터넷 상의 리소스가 해당 인스턴스와의 통신을 시작하지 못하게 하려면 외부 전용 인터넷 게이트웨이를 사용하면 됩니다. 이렇게 하려면 VPC에 외부 전용 인터넷 게이트웨이를 만들어 라우팅 테이블에 모든 IPv6 트래픽(:::/0)을 가리키는 라우팅을 추가하거나 IPv6 주소의 특정 범위를 외부 전용 인터넷 게이트웨이에 추가합니다. 라우팅 테이블에 연결된 서브넷의 IPv6 트래픽은 외부 전용 인터넷 게이트웨이로 라우팅됩니다.

외부 전용 인터넷 게이트웨이는 상태 저장 방식으로서, 서브넷의 인스턴스에서 인터넷 또는 기타 AWS 서비스로 트래픽을 전달한 다음, 다시 인스턴스로 응답을 보냅니다.

외부 전용 인터넷 게이트웨이에는 다음과 같은 특성이 있습니다.

- 보안 그룹은 외부 전용 인터넷 게이트웨이와 연결할 수 없습니다. 프라이빗 서브넷의 인스턴스에 대한 보안 그룹을 사용하여 해당 인스턴스에서 주고받는 트래픽을 제어할 수 있습니다.
- 네트워크 ACL을 사용하여 외부 전용 인터넷 게이트웨이가 트래픽을 라우팅하는 서브넷에서 주고받는 트래픽을 제어할 수 있습니다.

다음 다이어그램에서 VPC에는 IPv4 및 IPv6 CIDR 블록이 모두 있고 서브넷에는 IPv4 및 IPv6 CIDR 블록이 모두 있습니다. VPC에는 송신 전용 인터넷 게이트웨이가 있습니다.



다음은 서브넷과 연결된 라우팅 테이블의 예입니다. 모든 인터넷 바인딩 IPv6 트래픽(::/0)을 외부 전용 인터넷 게이트웨이로 전송하는 경로가 있습니다.

대상 주소	대상
10.0.0.0/16	로컬
2001:db8:1234:1a00:/64	로컬
::/0	<i>eigw-id</i>

송신 전용 인터넷 게이트웨이 작업

다음 작업에서는 프라이빗 서브넷에 대해 외부 전용(아웃바운드) 인터넷 게이트웨이를 생성하고 서브넷에 대한 라우팅을 구성하는 방법에 대해 설명합니다.

작업

- [외부 전용 인터넷 게이트웨이 생성](#)
- [송신 전용 인터넷 게이트웨이 보기](#)
- [사용자 지정 라우팅 테이블 생성](#)
- [외부 전용 인터넷 게이트웨이 삭제](#)

외부 전용 인터넷 게이트웨이 생성

Amazon VPC 콘솔을 사용하여 VPC에 대한 외부 전용 인터넷 게이트웨이를 만들 수 있습니다.

외부 전용 인터넷 게이트웨이를 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 Egress Only Internet Gateways를 선택합니다.
3. Create Egress Only Internet Gateway를 선택합니다.
4. (선택) 태그를 추가하거나 제거할 수 있습니다.

[태그 추가] 새 태그 추가를 선택하고 다음을 수행합니다.

- 키에서 키 이름을 입력합니다.
- 값에 키 값을 입력합니다.

[태그 제거] 태그의 키와 값 오른쪽에 있는 제거를 선택합니다.

5. 외부 전용 인터넷 게이트웨이를 생성할 VPC를 선택합니다.
6. Create를 선택합니다.

송신 전용 인터넷 게이트웨이 보기

Amazon VPC 콘솔에서 외부 전용 인터넷 게이트웨이에 대한 정보를 볼 수 있습니다.

외부 전용 인터넷 게이트웨이에 대한 정보를 보려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 Egress Only Internet Gateways를 선택합니다.
3. 세부 정보 창에서 관련 정보를 조회할 외부 전용 인터넷 게이트웨이를 선택합니다.

사용자 지정 라우팅 테이블 생성

VPC 외부 위치를 대상 주소로 하는 트래픽을 외부 전용 인터넷 게이트웨이로 전송하려면 사용자 지정 라우팅 테이블을 생성하고 트래픽을 게이트웨이로 전송하는 라우팅을 추가한 다음, 이를 서브넷과 연결해야 합니다.

사용자 지정 라우팅 테이블을 만들고 외부 전용 인터넷 게이트웨이에 라우팅을 추가하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [라우팅 테이블(Route Tables)]을 선택한 다음, [라우팅 테이블 생성(Create Route Table)]을 선택합니다.
3. [라우팅 테이블 생성(Create route table)] 대화 상자에서 선택적으로 라우팅 테이블의 이름을 지정한 다음 VPC를 선택하고 [라우팅 테이블 생성(Create route table)]을 선택합니다.
4. 방금 생성한 사용자 지정 라우팅 테이블을 선택합니다. 세부 정보 창에는 경로, 연결 및 경로 전파 작업을 위한 탭이 표시됩니다.
5. [경로(Routes)] 탭에서 [경로 편집(Edit routes)]을 선택하고, [대상 주소(Destination)] 상자에서 `::/0`을 지정하고, [대상(Target)] 목록에서 송신 전용 인터넷 게이트웨이 ID를 선택한 다음, [변경 사항 저장(Save changes)]을 선택합니다.
6. [서브넷 연결(Subnet associations)] 탭에서 [서브넷 연결 편집(Edit subnet associations)]을 선택하고 서브넷에 대한 확인란을 선택합니다. Save를 선택합니다.

또는 서브넷과 연결된 기존 라우팅 테이블에 경로를 추가할 수도 있습니다. 기존 라우팅 테이블을 선택하고, 위의 5단계 및 6단계를 수행하여 외부 전용 인터넷 게이트웨이에 대한 라우팅을 추가합니다.

라우팅 테이블에 대한 자세한 내용은 [라우팅 테이블 구성](#) 단원을 참조하십시오.

외부 전용 인터넷 게이트웨이 삭제

외부 전용 인터넷 게이트웨이가 더 이상 필요하지 않으면 이를 삭제할 수 있습니다. 삭제된 외부 전용 인터넷 게이트웨이를 가리키는 라우팅 테이블의 모든 라우팅은 그 라우팅을 수동으로 삭제하거나 업데이트할 때까지 blackhole 상태로 남아 있습니다.

외부 전용 인터넷 게이트웨이를 삭제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 외부 전용 인터넷 게이트웨이를 선택하고 해당하는 외부 전용 인터넷 게이트웨이를 선택합니다.
3. 삭제를 선택합니다.
4. 확인 대화 상자에서 Delete Egress Only Internet Gateway를 선택합니다.

API 및 CLI 개요

명령줄 또는 API를 사용하여 이 페이지에서 설명하는 작업을 수행할 수 있습니다. 명령줄 인터페이스 및 사용 가능한 API 작업 목록에 대한 자세한 내용은 [Amazon VPC 작업](#) 단원을 참조하십시오.

외부 전용 인터넷 게이트웨이 생성

- [create-egress-only-internet-gateway](#) (AWS CLI)
- [New-EC2EgressOnlyInternetGateway](#) (AWS Tools for Windows PowerShell)

외부 전용 인터넷 게이트웨이 설명

- [describe-egress-only-internet-gateways](#) (AWS CLI)
- [Get-EC2EgressOnlyInternetGatewayList](#) (AWS Tools for Windows PowerShell)

외부 전용 인터넷 게이트웨이 삭제

- [delete-egress-only-internet-gateway](#) (AWS CLI)

- [Remove-EC2EgressOnlyInternetGateway](#) (AWS Tools for Windows PowerShell)

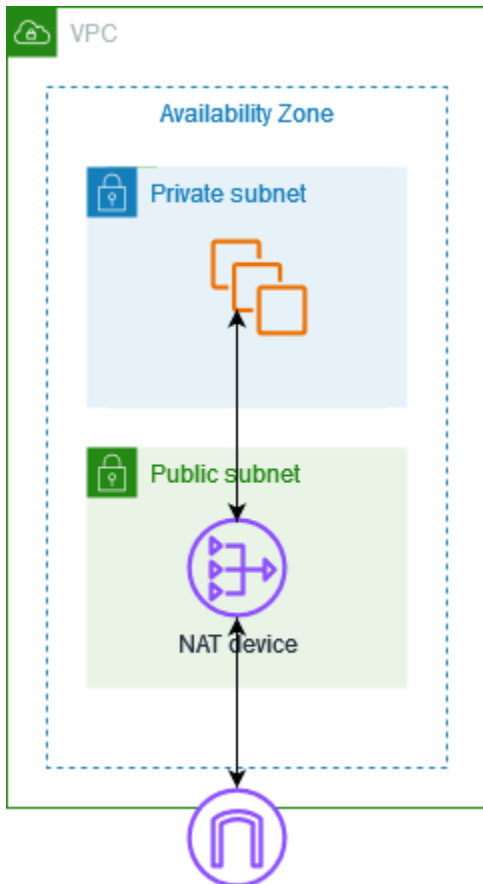
요금

외부 전용 인터넷 게이트웨이에는 요금이 부과되지 않지만 인터넷 게이트웨이를 사용하는 EC2 인스턴스에는 데이터 전송 요금이 부과됩니다. 자세한 내용은 [Amazon EC2 온디맨드 요금](#)을 참조하세요.

NAT 디바이스를 사용하여 인터넷 또는 다른 네트워크에 연결

NAT 디바이스를 사용하여 프라이빗 서브넷의 리소스가 인터넷, 다른 VPC 또는 온프레미스 네트워크에 연결되도록 허용할 수 있습니다. 이러한 인스턴스는 VPC 외부의 서비스와 통신할 수 있지만 원치 않는 연결 요청은 받을 수는 없습니다.

예를 들면 다음 다이어그램에서는 인터넷 게이트웨이를 통해 프라이빗 서브넷의 EC2 인스턴스를 인터넷에 연결할 수 있는 퍼블릭 서브넷의 NAT 디바이스를 보여줍니다. NAT 디바이스는 인스턴스의 소스 IPv4 주소를 NAT 디바이스의 주소로 바꿉니다. 인스턴스에 응답 트래픽을 전송할 때 NAT 디바이스는 주소를 원래 소스 IPv4 주소로 다시 변환합니다.



⚠ Important

- 이 문서에서는 NAT라는 용어를 일반적인 IT 용례에 따라 사용하지만, 실제로 NAT 디바이스는 주소 변환과 포트 주소 변환(PAT)을 모두 담당합니다.
- AWS에서 제공하는 NAT 게이트웨이라는 관리형 NAT 디바이스를 사용하거나, EC2 인스턴스에서 NAT 디바이스(여기서는 NAT 인스턴스라고 함)를 생성할 수 있습니다. NAT 게이트웨이는 더 나은 가용성과 대역폭을 제공하고 관리에 소요되는 작업이 줄어들기 때문에 권장합니다.

내용

- [NAT 게이트웨이](#)
- [NAT 인스턴스](#)
- [NAT 게이트웨이 및 NAT 인스턴스 비교](#)

NAT 게이트웨이

NAT 게이트웨이는 NAT(네트워크 주소 변환) 서비스입니다. 프라이빗 서브넷의 인스턴스가 VPC 외부의 서비스에 연결할 수 있지만 외부 서비스에서 이러한 인스턴스와의 연결을 시작할 수 없도록 NAT 게이트웨이를 사용할 수 있습니다.

NAT 게이트웨이를 만들 때 다음 연결 유형 중 하나를 지정합니다.

- 퍼블릭 - (기본값) 프라이빗 서브넷의 인스턴스는 퍼블릭 NAT 게이트웨이를 통해 인터넷에 연결할 수 있지만 인터넷에서 원치 않는 인바운드 연결을 수신할 수 없습니다. 퍼블릭 서브넷에서 퍼블릭 NAT 게이트웨이를 생성하고 생성 시 탄력적 IP 주소를 NAT 게이트웨이와 연결해야 합니다. 트래픽을 NAT 게이트웨이에서 VPC용 인터넷 게이트웨이로 라우팅합니다. 또는 퍼블릭 NAT 게이트웨이를 사용하여 다른 VPC 또는 온프레미스 네트워크에 연결할 수 있습니다. 이 경우 NAT 게이트웨이에서 Transit Gateway 또는 가상 프라이빗 게이트웨이를 통해 트래픽을 라우팅합니다.
- 프라이빗- 프라이빗 서브넷의 인스턴스는 프라이빗 NAT 게이트웨이를 통해 다른 VPC 또는 온프레미스 네트워크에 연결할 수 있습니다. 트래픽을 NAT 게이트웨이에서 Transit Gateway 또는 가상 프라이빗 게이트웨이를 통해 트래픽을 라우팅할 수 있습니다. 탄력적 IP 주소를 프라이빗 NAT 게이트웨이에 연결할 수 없습니다. 프라이빗 NAT 게이트웨이를 사용하여 VPC에 인터넷 게이트웨이를 연결할 수 있지만 프라이빗 NAT 게이트웨이에서 인터넷 게이트웨이로 트래픽을 라우팅하는 경우 인터넷 게이트웨이가 트래픽을 삭제합니다.

프라이빗 및 퍼블릭 NAT 게이트웨이는 모두 인스턴스의 소스 프라이빗 IPv4 주소를 NAT 게이트웨이의 프라이빗 IPv4 주소에 매핑하지만, 퍼블릭 NAT 게이트웨이의 경우 인터넷 게이트웨이는 퍼블릭 NAT 게이트웨이의 프라이빗 IPv4 주소를 NAT 게이트웨이와 연결된 탄력적 IP 주소에 매핑합니다. 인스턴스에 응답 트래픽을 전송할 때 인스턴스는 퍼블릭 또는 프라이빗 NAT 게이트웨어 여부에 상관없이 NAT 게이트웨이는 주소를 원래 소스 IP 주소로 다시 변환합니다.

Important

퍼블릭 또는 프라이빗 NAT 게이트웨이를 사용하여 트래픽을 전송 게이트웨이와 가상 프라이빗 게이트웨이로 라우팅할 수 있습니다.

프라이빗 NAT 게이트웨이를 사용하여 전송 게이트웨이 또는 가상 프라이빗 게이트웨이에 연결하는 경우 대상에 대한 트래픽은 프라이빗 NAT 게이트웨이의 프라이빗 IP 주소에서 전송됩니다.

퍼블릭 NAT 게이트웨이를 사용하여 전송 게이트웨이 또는 가상 프라이빗 게이트웨이에 연결하는 경우 인터넷 게이트웨이를 사용하지 않는 한 대상에 대한 트래픽은 퍼블릭 NAT 게이트웨이의 프라이빗 IP 주소에서 전송됩니다. 퍼블릭 NAT 게이트웨이는 인터넷 게이트웨이와 함께 사용할 때만 EIP를 소스 IP 주소로 사용합니다.

내용

- [NAT 게이트웨이 기본 사항](#)
- [NAT 게이트웨이 사용 제어](#)
- [NAT 게이트웨이 작업](#)
- [API 및 CLI 개요](#)
- [NAT Gateway 사용 사례](#)
- [DNS64 및 NAT64](#)
- [Amazon CloudWatch를 사용하여 NAT 게이트웨이 모니터링](#)
- [NAT 게이트웨이 문제 해결](#)
- [요금](#)

NAT 게이트웨이 기본 사항

각 NAT 게이트웨이는 특정 가용 영역에 생성되고 해당 영역에서 중복성을 통해 구현됩니다. 각 가용 영역에서 만들 수 있는 NAT 게이트웨이 개수에는 할당량이 있습니다. 자세한 내용은 [Amazon VPC 할당량](#) 단원을 참조하세요.

여러 가용 영역에 리소스가 있고 NAT 게이트웨이 하나를 공유하는 경우, NAT 게이트웨이의 가용 영역이 다운되면 다른 가용 영역의 리소스도 인터넷에 액세스할 수 없게 됩니다. 복원력 향상을 위해 각 가용 영역에 NAT 게이트웨이를 생성하고 리소스가 동일한 가용 영역의 NAT 게이트웨이를 사용하도록 라우팅을 구성합니다.

NAT 게이트웨이에 적용되는 특성 및 규칙은 다음과 같습니다.

- NAT 게이트웨이는 TCP, UDP, ICMP 등의 프로토콜을 지원합니다.
- NAT 게이트웨이는 IPv4 또는 IPv6 트래픽에 대해 지원됩니다. IPv6 트래픽의 경우 NAT 게이트웨이가 NAT64를 수행합니다. 이를 DNS64(Route 53 Resolver에서 사용 가능)와 함께 사용하면 Amazon VPC의 서브넷에 있는 IPv6 워크로드가 IPv4 리소스와 통신할 수 있습니다. 이러한 IPv4 서비스는 동일한 VPC(별도의 서브넷에 있음) 또는 다른 VPC, 온프레미스 환경 또는 인터넷에 존재할 수 있습니다.
- NAT 게이트웨이는 5Gbps의 대역폭을 지원하며 최대 100Gbps까지 자동 확장합니다. 더 많은 대역폭이 필요한 경우 리소스를 여러 서브넷으로 분할하고 각 서브넷에 NAT 게이트웨이를 만들 수 있습니다.
- NAT 게이트웨이는 초당 백만 개의 패킷을 처리할 수 있으며 초당 최대 천만 개의 패킷을 자동으로 확장할 수 있습니다. 이 제한을 초과하면 NAT 게이트웨이가 패킷을 삭제합니다. 패킷 손실을 방지하려면 리소스를 여러 서브넷으로 분할하고 각 서브넷에 대해 별도의 NAT 게이트웨이를 생성합니다.
- 각 IPv4 주소는 각 고유 대상에 대해 55,000개까지 동시 연결을 지원할 수 있습니다. 고유 대상은 대상 IP 주소, 대상 포트 및 프로토콜(TCP/UDP/ICMP)의 고유한 조합으로 식별됩니다. 최대 8개의 IPv4 주소를 NAT 게이트웨이에 연결하여 이 제한을 늘릴 수 있습니다(기본 IPv4 주소 1개 및 보조 IPv4 주소 7개). 기본적으로 2개의 탄력적 IP 주소를 퍼블릭 NAT 게이트웨이에 연결하는 것으로 제한됩니다. 할당량 조정을 요청하여 이 제한을 늘릴 수 있습니다. 자세한 내용은 [탄력적 IP 주소](#) 단원을 참조하십시오.
- NAT 게이트웨이에 할당할 프라이빗 IPv4 주소를 선택하거나 서브넷의 IPv4 주소 범위에서 자동으로 할당하도록 할 수 있습니다. 할당된 프라이빗 IPv4 주소는 프라이빗 NAT 게이트웨이를 삭제할 때까지 유지됩니다. 프라이빗 IPv4 주소는 분리할 수 없으며, 추가 프라이빗 IPv4 주소를 연결할 수 없습니다.
- 보안 그룹을 NAT 게이트웨이와 연결할 수 없습니다. 보안 그룹을 인스턴스에 연결하여 인바운드 및 아웃바운드 트래픽을 제어할 수 있습니다.
- 네트워크 ACL을 사용하여 NAT 게이트웨이에 대해 서브넷에서 주고받는 트래픽을 제어할 수 있습니다. NAT 게이트웨이는 포트 1024 - 65535를 사용합니다. 자세한 내용은 [네트워크 ACL을 사용하여 서브넷에 대한 트래픽 제어](#) 단원을 참조하십시오.
- NAT 게이트웨이는 네트워크 인터페이스를 수신합니다. 인터페이스에 할당할 프라이빗 IPv4 주소를 선택하거나 서브넷의 IPv4 주소 범위에서 자동으로 할당하도록 할 수 있습니다. Amazon EC2 콘솔

을 사용하여 NAT 게이트웨이에 대한 네트워크 인터페이스를 볼 수 있습니다. 자세한 내용은 [네트워크 인터페이스에 대한 세부 정보 보기](#)를 참조하세요. 이 네트워크 인터페이스의 속성을 수정할 수 없습니다.

- VPC 피어링 연결을 통해 NAT 게이트웨이로 트래픽을 라우팅할 수 없습니다. 트래픽이 가상 프라이빗 게이트웨이를 통한 하이브리드 연결(사이트 간 VPN 또는 Direct Connect)을 통해 도착하면 NAT 게이트웨이를 통해 트래픽을 라우팅할 수 없습니다. 트래픽이 전송 게이트웨이를 통한 하이브리드 연결(사이트 간 VPN 또는 Direct Connect)을 통해 도착하면 NAT 게이트웨이를 통해 트래픽을 라우팅할 수 있습니다.
- NAT 게이트웨이에서 지원되는 트래픽의 최대 전송 단위(MTU)는 8,500이지만, 다음 사항에 유의해야 합니다.
 - 퍼블릭 NAT 게이트웨이를 사용하여 인터넷을 통해 리소스와 통신할 때 잠재적 패킷 손실을 방지하려면 EC2 인스턴스의 MTU 설정이 1,500바이트를 초과하지 않아야 합니다. 인스턴스의 MTU 확인 및 설정에 대한 자세한 내용은 Amazon EC2 사용 설명서의 [Linux 인스턴스에서 MTU 확인 및 설정](#)을 참조하세요.
 - NAT 게이트웨이에서는 FRAG_NEEDED ICMPv4 패킷 및 패킷이 너무 큼(PTB) ICMPv6 패킷을 통한 경로 MTU 검색(PMTUD)이 지원됩니다.
 - NAT 게이트웨이에서는 모든 패킷에 MSS(최대 세그먼트 크기) 클램핑이 강제로 적용됩니다. 자세한 내용은 [RFC879](#)를 참조하세요.

NAT 게이트웨이 사용 제어

기본적으로 사용자에게는 NAT 게이트웨이를 사용할 권한이 없습니다. 사용자에게 NAT 게이트웨이를 생성, 설명, 삭제할 수 있는 권한을 부여하는 정책이 연결된 IAM 역할을 만들 수 있습니다. 자세한 내용은 [Amazon VPC용 Identity and Access Management](#) 단원을 참조하십시오.

NAT 게이트웨이 작업

Amazon VPC 콘솔을 사용하여 NAT 게이트웨이를 생성하고 관리할 수 있습니다.

Tasks

- [NAT 게이트웨이 만들기](#)
- [보조 IP 주소 연결 편집](#)
- [NAT 게이트웨이 태그 지정](#)
- [NAT 게이트웨이 삭제](#)

NAT 게이트웨이 만들기

다음 절차에 따라 NAT 게이트웨이를 생성하세요.

관련 할당량

- 계정에 할당된 EIP 수를 모두 사용한 경우 퍼블릭 NAT 게이트웨이를 생성할 수 없습니다. EIP 할당량 및 그 조절 방법에 대한 자세한 내용은 [탄력적 IP 주소](#) 섹션을 참조하세요.
- 프라이빗 NAT 게이트웨이에 최대 8개의 프라이빗 IPv4 주소를 할당할 수 있습니다. 이 제한은 조정할 수 없습니다.
- 기본적으로 2개의 탄력적 IP 주소를 퍼블릭 NAT 게이트웨이에 연결하는 것으로 제한됩니다. 할당량 조정을 요청하여 이 제한을 늘릴 수 있습니다. 자세한 내용은 [탄력적 IP 주소](#) 단원을 참조하십시오.

NAT 게이트웨이를 만들려면

- <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
- 탐색 창에서 NAT 게이트웨이를 선택합니다.
- NAT 게이트웨이 생성을 선택합니다.
- (선택 사항) NAT 게이트웨이의 이름을 지정합니다. 이렇게 하면 키가 **Name**이고 값이 사용자가 지정한 이름인 태그가 생성됩니다.
- NAT 게이트웨이를 생성할 서브넷을 선택합니다.
- 연결 유형에서 기본 선택 퍼블릭을 그대로 사용하여 퍼블릭 NAT 게이트웨이를 생성하거나 프라이빗을 선택하여 프라이빗 NAT 게이트웨이를 생성합니다. 퍼블릭과 프라이빗 NAT 게이트웨이 간의 차이점에 대한 자세한 정보는 [NAT 게이트웨이](#)를 참조하세요.
- 퍼블릭을 선택한 경우 다음을 수행합니다. 그렇지 않으면 8단계로 건너뛴니다.
 - 탄력적 IP 할당 ID를 선택하여 EIP를 NAT 게이트웨이에 할당하거나 탄력적 IP 할당을 선택하여 퍼블릭 NAT 게이트웨이에 사용할 EIP를 자동으로 할당합니다. 기본적으로 2개의 탄력적 IP 주소를 퍼블릭 NAT 게이트웨이에 연결하는 것으로 제한됩니다. 할당량 조정을 요청하여 이 제한을 늘릴 수 있습니다. 자세한 내용은 [탄력적 IP 주소](#) 단원을 참조하십시오.

Important

퍼블릭 NAT 게이트웨이에 EIP를 할당하는 경우 EIP의 네트워크 경계 그룹은 반드시 퍼블릭 NAT 게이트웨이를 시작하는 대상 가용 영역의 네트워크 경계 그룹과 일치해야 합니다. 일치하지 않을 경우 NAT 게이트웨이가 시작되지 않습니다. 서브넷의 세부 정보를 확인하면 해당 서브넷 AZ의 네트워크 경계 그룹을 확인할 수 있습니다. 마찬가지로

EIP 주소의 세부 정보를 확인하면 EIP의 네트워크 경계 그룹을 확인할 수 있습니다. 네트워크 경계 그룹 및 EIP에 대한 자세한 내용을 확인하려면 [탄력적 IP 주소 할당](#) 섹션을 참조하세요.

2. (선택 사항) 추가 설정을 선택하고 기본 프라이빗 IP 주소 - 선택 사항에 NAT 게이트웨이의 프라이빗 IPv4 주소를 입력합니다. 주소를 입력하지 않으면 AWS는 NAT 게이트웨이가 있는 서브넷에서 무작위로 NAT 게이트웨이에 프라이빗 IPv4 주소를 자동으로 할당합니다.
3. 11단계로 건너뛩니다.
8. 프라이빗을 선택한 경우 추가 설정, 프라이빗 IPv4 주소 할당 방법에서 다음 중 하나를 선택합니다.
 - 자동 할당: AWS가 NAT 게이트웨이의 기본 프라이빗 IPv4 주소를 선택합니다. 자동 할당된 프라이빗 IPv4 주소 수에서 NAT 게이트웨이의 보조 프라이빗 IPv4 주소 수를 필요한 경우 지정할 수 있습니다. AWS가 NAT 게이트웨이의 서브넷에서 임의로 해당 IP 주소를 선택합니다.
 - 사용자 지정: 기본 프라이빗 IPv4 주소에서 NAT 게이트웨이의 기본 프라이빗 IPv4 주소를 선택합니다. 보조 프라이빗 IPv4 주소에서 NAT 게이트웨이에 최대 7개의 보조 프라이빗 IPv4 주소를 필요한 경우 지정할 수 있습니다.
9. 8단계에서 사용자 지정을 선택한 경우 이 단계를 건너뛰세요. 자동 할당을 선택한 경우 자동 할당된 프라이빗 IP 주소 수에서 AWS가 이 프라이빗 NAT 게이트웨이에 할당할 보조 IPv4 주소 수를 선택합니다. IPv4 주소를 최대 7개까지 선택할 수 있습니다.

Note

보조 IPv4 주소는 선택 사항이며 NAT 게이트웨이를 사용하는 워크로드가 단일 대상(동일한 대상 IP, 대상 포트 및 프로토콜)에 대한 동시 연결 55,000개를 초과하는 경우 지정하거나 할당해야 합니다. 보조 IPv4 주소는 사용 가능한 포트 수를 늘리므로 워크로드가 NAT 게이트웨이를 사용하여 설정할 수 있는 동시 연결 수에 대한 제한이 늘어납니다.

10. 9단계에서 자동 할당을 선택한 경우 이 단계를 건너뛰세요. 사용자 지정을 선택한 경우 다음을 수행합니다.
 1. 기본 프라이빗 IPv4 주소에 프라이빗 IPv4 주소를 입력합니다.
 2. 보조 프라이빗 IPv4 주소에 최대 7개의 보조 프라이빗 IPv4 주소를 입력합니다.
11. (선택 사항) NAT 게이트웨이에 태그를 추가하려면 새로운 태그 추가(Add new tag)를 선택하고 키 이름과 값을 입력합니다. 최대 50개의 태그를 추가할 수 있습니다.
12. NAT 게이트웨이 생성을 선택합니다.

13. NAT 게이트웨이의 초기 상태는 Pending입니다. 상태가 Available(으)로 변경되면 NAT 게이트웨이가 사용 준비 상태가 됩니다. 필요에 따라 라우팅 테이블을 업데이트해야 합니다. 예를 보려면 [the section called “사용 사례”](#)을 참조하세요.

NAT 게이트웨이의 상태가 Failed 상태로 바뀌면 생성 중에 오류가 발생한 것입니다. 자세한 내용은 [NAT 게이트웨이 생성 실패](#) 단원을 참조하십시오.

보조 IP 주소 연결 편집

각 IPv4 주소는 각 고유 대상에 대해 55,000개까지 동시 연결을 지원할 수 있습니다. 고유 대상은 대상 IP 주소, 대상 포트 및 프로토콜(TCP/UDP/ICMP)의 고유한 조합으로 식별됩니다. 최대 8개의 IPv4 주소를 NAT 게이트웨이에 연결하여 이 제한을 늘릴 수 있습니다(기본 IPv4 주소 1개 및 보조 IPv4 주소 7개). 기본적으로 2개의 탄력적 IP 주소를 퍼블릭 NAT 게이트웨이에 연결하는 것으로 제한됩니다. 할당량 조정을 요청하여 이 제한을 늘릴 수 있습니다. 자세한 내용은 [탄력적 IP 주소](#) 단원을 참조하십시오.

[NAT 게이트웨이 CloudWatch 지표](#), ErrorPortLocation, PacketsDropCount를 사용하여 NAT 게이트웨이에서 포트 할당 오류가 발생하거나 패킷이 손실되는지 확인할 수 있습니다. 이 문제를 해결하려면 NAT 게이트웨이에 보조 IPv4 주소를 추가하세요.

고려 사항

- 프라이빗 NAT 게이트웨이를 생성할 때 또는 이 섹션의 절차를 사용하여 NAT 게이트웨이를 생성한 후에 보조 프라이빗 IPv4 주소를 추가할 수 있습니다. 이 섹션의 절차를 사용하여 NAT 게이트웨이를 생성한 후에만 퍼블릭 NAT 게이트웨이에 보조 EIP 주소를 추가할 수 있습니다.
- NAT 게이트웨이는 IPv4 주소를 최대 8개(기본 IPv4 주소 1개 및 보조 IPv4 주소 7개) 연결할 수 있습니다. 프라이빗 NAT 게이트웨이에 최대 8개의 프라이빗 IPv4 주소를 할당할 수 있습니다. 기본적으로 2개의 탄력적 IP 주소를 퍼블릭 NAT 게이트웨이에 연결하는 것으로 제한됩니다. 할당량 조정을 요청하여 이 제한을 늘릴 수 있습니다. 자세한 내용은 [탄력적 IP 주소](#) 단원을 참조하십시오.

보조 IPv4 주소 연결을 편집하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 NAT 게이트웨이를 선택합니다.
3. 보조 IPv4 주소 연결을 편집하려는 NAT 게이트웨이를 선택합니다.
4. 작업을 선택한 다음 보조 IP 주소 연결 편집을 선택합니다.

5. 프라이빗 NAT 게이트웨이의 보조 IPv4 주소 연결을 편집하는 경우 작업에서 새 IPv4 주소 할당 또는 기존 IPv4 주소 할당 해제를 선택합니다. 퍼블릭 NAT 게이트웨이의 보조 IPv4 주소 연결을 편집하는 경우 작업에서 새 IPv4 주소 연결 또는 기존 IPv4 주소 연결 해제를 선택합니다.
6. 다음 중 하나를 수행하십시오.

- 새 IPv4 주소를 할당하거나 연결하도록 선택한 경우 다음을 수행합니다.
 1. 이 단계는 필수입니다. 프라이빗 IPv4 주소를 선택해야 합니다. 프라이빗 IPv4 주소 할당 방법을 선택합니다.
 - 자동 할당: AWS가 기본 프라이빗 IPv4 주소를 자동으로 선택합니다. AWS가 NAT 게이트웨이에 할당할 보조 프라이빗 IPv4 주소를 최대 7개까지 할당할지 선택합니다. AWS는 NAT 게이트웨이가 있는 서브넷에서 무작위로 주소를 자동으로 선택하고 할당합니다.
 - 사용자 지정: 기본 프라이빗 IPv4 주소와 최대 7개의 보조 프라이빗 IPv4 주소를 선택하여 NAT 게이트웨이에 할당합니다.
 2. 탄력적 IP 할당 ID에서 보조 IPv4 주소로 추가할 EIP를 선택합니다. 이 단계는 필수입니다. 프라이빗 IPv4 주소와 함께 EIP를 선택해야 합니다. 프라이빗 IP 주소 할당 방법으로 사용자 지정을 선택한 경우 추가하는 각 EIP의 프라이빗 IPv4 주소도 입력해야 합니다.

⚠ Important

퍼블릭 NAT 게이트웨이에 보조 EIP를 할당하는 경우 EIP의 네트워크 경계 그룹은 반드시 퍼블릭 NAT 게이트웨이가 있는 가용 영역의 네트워크 경계 그룹과 일치해야 합니다. 동일하지 않을 경우 EIP가 할당되지 않습니다. 서브넷의 세부 정보를 확인하면 해당 서브넷 AZ의 네트워크 경계 그룹을 확인할 수 있습니다. 마찬가지로 EIP 주소의 세부 정보를 확인하면 EIP의 네트워크 경계 그룹을 확인할 수 있습니다. 네트워크 경계 그룹 및 EIP에 대한 자세한 내용을 확인하려면 [탄력적 IP 주소 할당](#) 섹션을 참조하세요.

NAT 게이트웨이는 IP 주소를 8개까지 연결할 수 있습니다. 퍼블릭 NAT 게이트웨이인 경우 리전당 EIP에 대한 기본 할당량 제한이 있습니다. 자세한 내용은 [탄력적 IP 주소](#) 단원을 참조하십시오.

- 새 IPv4 주소를 할당 해제하거나 연결 해제하도록 선택한 경우 다음을 완료합니다.
 1. 할당 해제할 기존 보조 IP 주소에서 할당 해제하려는 보조 IP 주소를 선택합니다.
 2. (선택 사항) 연결 드레인 기간에 연결이 아직 진행 중인 경우 IP 주소를 강제로 해제하기 전에 대기할 최대 시간(초)을 입력합니다. 값을 입력하지 않으면 기본값은 350초입니다.

7. **Save changes(변경 사항 저장)**를 선택합니다.

NAT 게이트웨이의 상태가 `Failed` 상태로 바뀌면 생성 중에 오류가 발생한 것입니다. 자세한 내용은 [NAT 게이트웨이 생성 실패](#) 단원을 참조하세요.

NAT 게이트웨이 태그 지정

NAT 게이트웨이에 태그를 지정하면 조직의 요구에 따라 이를 식별 또는 분류할 수 있습니다. 태그 사용에 대한 자세한 내용은 [Amazon EC2 사용 설명서](#)의 Amazon EC2 리소스에 태깅을 참조하세요.

NAT 게이트웨이는 비용 할당 태그를 지원합니다. 따라서 태그를 사용하여 AWS 청구서를 구성하고 고유한 원가 구조를 반영할 수도 있습니다. 자세한 내용은 [AWS Billing 사용 설명서](#)에서 비용 할당 태그 사용을 참조하세요. 태그를 사용한 비용 할당 보고서 설정에 대한 자세한 내용은 [AWS 계정 결제 정보](#)의 월간 비용 할당 보고서를 참조하세요.

NAT 게이트웨이에 태그를 지정하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [NAT Gateways]를 선택합니다.
3. 태그를 지정하려는 NAT 게이트웨이를 선택하고 작업을 선택합니다. 태그 관리를 선택합니다.
4. 새 태그 추가를 선택하고 태그의 키와 값을 정의합니다. 최대 50개의 태그를 추가할 수 있습니다.
5. Save(저장)를 선택합니다.

NAT 게이트웨이 삭제

NAT 게이트웨이가 더 이상 필요하지 않으면 삭제할 수 있습니다. NAT 게이트웨이를 삭제하면 해당 항목은 한 시간 동안 Amazon VPC 콘솔에 표시된 후 자동으로 제거됩니다. 이 항목을 직접 제거할 수는 없습니다.

NAT 게이트웨이를 삭제하면 탄력적 IP 주소가 연결 해제되지만 계정에서 주소가 해제되지는 않습니다. NAT 게이트웨이를 삭제하면 NAT 게이트웨이 경로는 경로를 삭제하거나 업데이트할 때까지 `blackhole` 상태로 유지됩니다.

NAT 게이트웨이를 삭제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [NAT Gateways]를 선택합니다.
3. NAT 게이트웨이에 대한 라디오 버튼을 선택한 후, 작업, NAT 게이트웨이 삭제를 선택합니다.
4. 확인 메시지가 나타나면 **delete**를 입력한 다음 삭제를 선택합니다.

5. 퍼블릭 NAT 게이트웨이와 연결된 탄력적 IP 주소가 더 이상 필요 없는 경우 해당 주소를 릴리스하는 것이 좋습니다. 자세한 내용은 [탄력적 IP 주소 릴리스](#) 단원을 참조하십시오.

API 및 CLI 개요

이 페이지에서 설명한 작업은 명령줄이나 API를 사용하여 수행할 수 있습니다. 명령줄 인터페이스 및 사용 가능한 API 작업 목록에 대한 자세한 내용은 [Amazon VPC 작업](#)를 참조하세요.

프라이빗 NAT 게이트웨이에 프라이빗 IPv4 주소 할당

- [assign-private-nat-gateway-address](#)(AWS CLI)
- [Register-EC2PrivateNatGatewayAddress](#)(AWS Tools for Windows PowerShell)
- [AssignPrivateNatGatewayAddress](#)(Amazon EC2 Query API)

탄력적 IP 주소(EIP) 및 프라이빗 IPv4 주소를 퍼블릭 NAT 게이트웨이와 연결

- [associate-nat-gateway-address](#)(AWS CLI)
- [Register-EC2NatGatewayAddress](#)(AWS Tools for Windows PowerShell)
- [AssociateNatGatewayAddress](#)(Amazon EC2 Query API)

NAT 게이트웨이 만들기

- [create-nat-gateway](#)(AWS CLI)
- [New-EC2NatGateway](#)(AWS Tools for Windows PowerShell)
- [CreateNatGateway](#)(Amazon EC2 쿼리 API)

NAT 게이트웨이 삭제

- [delete-nat-gateway](#)(AWS CLI)
- [Remove-EC2NatGateway](#)(AWS Tools for Windows PowerShell)
- [DeleteNatGateway](#)(Amazon EC2 쿼리 API)

NAT 게이트웨이 설명

- [describe-nat-gateways](#)(AWS CLI)

- [Get-EC2NatGateway](#)(AWS Tools for Windows PowerShell)
- [DescribeNatGateways](#)(Amazon EC2 쿼리 API)

퍼블릭 NAT 게이트웨이에서 보조 탄력적 IP 주소(EIP) 연결 해제

- [disassociate-nat-gateway-address](#)(AWS CLI)
- [Unregister-EC2NatGatewayAddress](#)(AWS Tools for Windows PowerShell)
- [DisassociateNatGatewayAddress](#)(Amazon EC2 Query API)

NAT 게이트웨이 태그 지정

- [create-tags](#)(AWS CLI)
- [New-EC2Tag](#)(AWS Tools for Windows PowerShell)
- [CreateTags](#)(Amazon EC2 쿼리 API)

프라이빗 NAT 게이트웨이에서 보조 IPv4 주소 할당 해제

- [unassign-private-nat-gateway-address](#)(AWS CLI)
- [Unregister-EC2PrivateNatGatewayAddress](#)(AWS Tools for Windows PowerShell)
- [UnassignPrivateNatGatewayAddress](#)(Amazon EC2 Query API)

NAT Gateway 사용 사례

다음은 퍼블릭 및 프라이빗 NAT 게이트웨이의 사용 사례입니다.

시나리오

- [프라이빗 서브넷에서 인터넷 액세스](#)
- [허용 목록에 있는 IP 주소를 사용하여 네트워크에 액세스](#)
- [중첩되는 네트워크 간에 통신 사용](#)

프라이빗 서브넷에서 인터넷 액세스

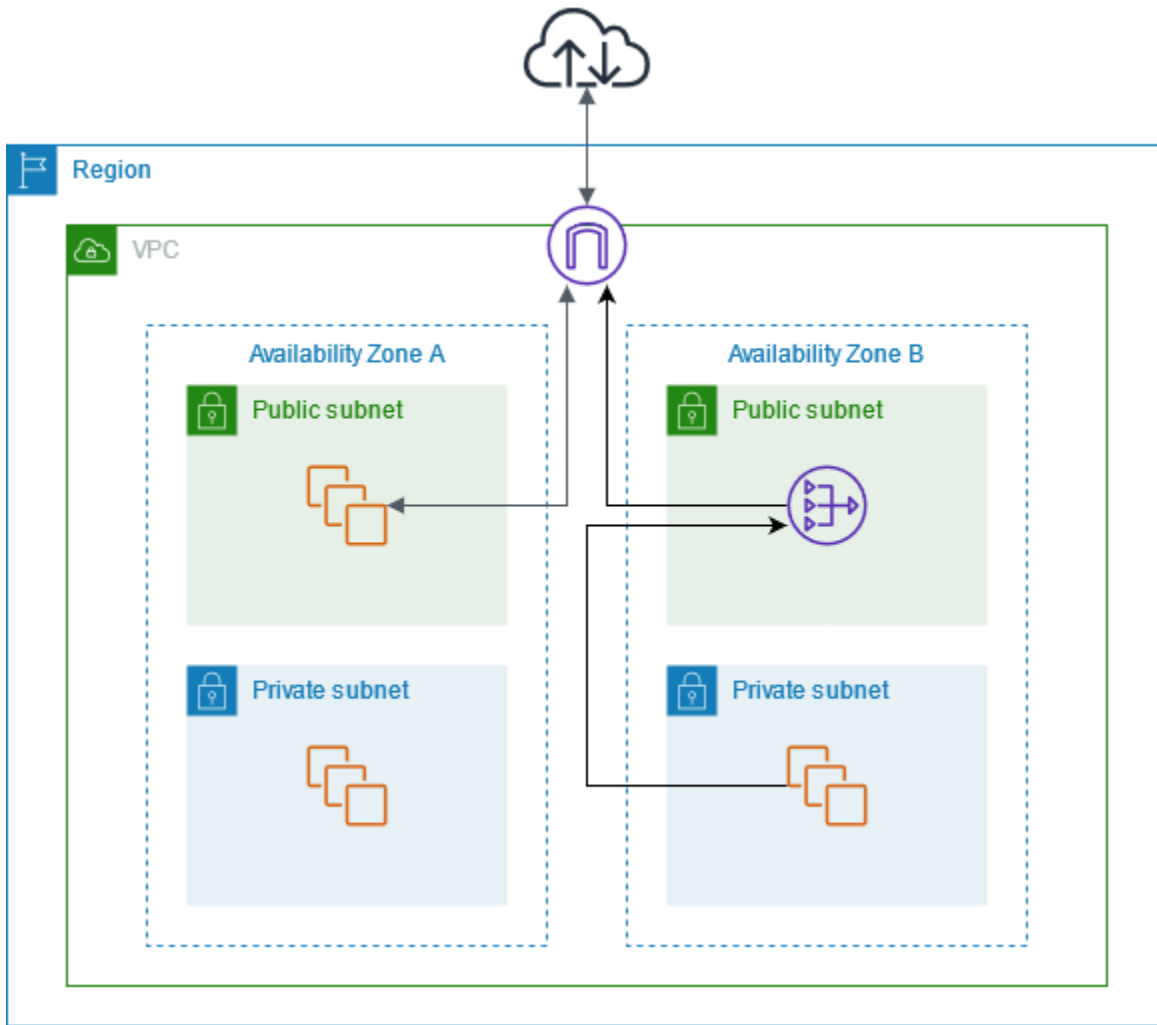
퍼블릭 NAT 게이트웨이를 사용하여 프라이빗 서브넷의 인스턴스가 아웃바운드 트래픽을 인터넷으로 전송할 수 있도록 하는 동시에 인터넷이 인스턴스에 대한 연결을 설정하는 것을 방지할 수 있습니다.

내용

- [개요](#)
- [라우팅](#)
- [퍼블릭 NAT 게이트웨이 테스트](#)

개요

다음 다이어그램에서 이 사용 사례를 보여줍니다. 각 가용 영역에 2개의 서브넷이 있는 2개의 가용 영역이 있습니다. 각 서브넷에 대한 라우팅 테이블에 따라 트래픽이 라우팅되는 방법이 결정됩니다. 가용 영역 A에서 퍼블릭 서브넷의 인스턴스는 인터넷 게이트웨이에 대한 경로를 통해 인터넷에 연결할 수 있지만 프라이빗 서브넷의 인스턴스는 인터넷에 대한 경로가 없습니다. 가용 영역 B에서 퍼블릭 서브넷은 NAT 게이트웨이를 포함하고 프라이빗 서브넷의 인스턴스는 퍼블릭 서브넷의 NAT 게이트웨이 경로를 통해 인터넷에 연결할 수 있습니다. 프라이빗 및 퍼블릭 NAT 게이트웨이는 모두 인스턴스의 소스 프라이빗 IPv4 주소를 프라이빗 NAT 게이트웨이의 프라이빗 IPv4 주소에 매핑하지만, 퍼블릭 NAT 게이트웨이의 경우 인터넷 게이트웨이는 퍼블릭 NAT 게이트웨이의 프라이빗 IPv4 주소를 NAT 게이트웨이와 연결된 엘라스틱 IP 주소에 매핑합니다. 인스턴스에 응답 트래픽을 전송할 때 인스턴스는 퍼블릭 또는 프라이빗 NAT 게이트웨어 여부에 상관없이 NAT 게이트웨이는 주소를 원래 소스 IP 주소로 다시 변환합니다.



가용 영역 A의 프라이빗 서브넷에 있는 인스턴스도 인터넷에 연결해야 하는 경우 이 서브넷에서 가용 영역 B의 NAT 게이트웨이로 가는 경로를 생성할 수 있습니다. 또는 인터넷 액세스가 필요한 리소스가 포함된 각 가용 영역에 NAT 게이트웨이를 생성하여 복원력을 향상시킬 수 있습니다. 예시 다이어그램은 [the section called “프라이빗 서버”](#) 섹션을 참조하세요.

라우팅

다음은 가용 영역 A의 퍼블릭 서브넷과 연결된 라우팅 테이블입니다. 첫 번째 항목은 로컬 경로입니다. 서브넷의 인스턴스가 프라이빗 IP 주소를 사용하여 VPC의 다른 인스턴스와 통신할 수 있도록 합니다. 두 번째 항목은 다른 모든 서브넷 트래픽을 인터넷 게이트웨이로 전송하여 서브넷의 인스턴스가 인터넷에 액세스할 수 있도록 합니다.

대상 주소	대상
<i>VPC CIDR</i>	로컬

대상 주소	대상
0.0.0.0/0	<i>internet-gateway-id</i>

다음은 가용 영역 A의 프라이빗 서브넷과 연결된 라우팅 테이블입니다. 첫 번째 항목은 로컬 경로이며 이 경로는 서브넷의 인스턴스가 프라이빗 IP 주소를 사용하여 VPC의 다른 인스턴스와 통신할 수 있도록 합니다. 이 서브넷의 인스턴스에는 인터넷에 대한 액세스 권한이 없습니다.

대상 주소	대상
<i>VPC CIDR</i>	로컬

다음은 가용 영역 B의 퍼블릭 서브넷과 연결된 라우팅 테이블입니다. 첫 번째 항목은 로컬 경로이며, 이 경로는 서브넷의 인스턴스가 프라이빗 IP 주소를 사용하여 VPC의 다른 인스턴스와 통신할 수 있도록 합니다. 두 번째 항목은 다른 모든 서브넷 트래픽을 인터넷 게이트웨이로 전송하여 서브넷의 NAT 게이트웨이가 인터넷에 액세스할 수 있도록 합니다.

대상 주소	대상
<i>VPC CIDR</i>	로컬
0.0.0.0/0	<i>internet-gateway-id</i>

다음은 가용 영역 B의 프라이빗 서브넷과 연결된 라우팅 테이블입니다. 첫 번째 항목은 로컬 경로입니다. 서브넷의 인스턴스가 프라이빗 IP 주소를 사용하여 VPC의 다른 인스턴스와 통신할 수 있도록 합니다. 두 번째 항목에서는 기타 서브넷 트래픽을 모두 NAT 게이트웨이로 전송합니다.

대상 주소	대상
<i>VPC CIDR</i>	로컬
0.0.0.0/0	<i>nat-gateway-id</i>

자세한 내용은 [the section called “라우팅 테이블 작업”](#) 단원을 참조하십시오.

퍼블릭 NAT 게이트웨이 테스트

NAT 게이트웨이를 만들고 라우팅 테이블을 업데이트한 후에는 프라이빗 서브넷의 인스턴스에서 인터넷의 원격 주소를 ping하여 인터넷에 연결할 수 있는지 테스트할 수 있습니다. 이렇게 하는 방법의 예는 [인터넷 연결 테스트](#) 섹션을 참조하세요.

인터넷에 연결할 수 있는 경우 인터넷 트래픽이 NAT 게이트웨이를 통해 라우팅되는지도 확인할 수 있습니다.

- 프라이빗 서브넷의 인스턴스에서 전송되는 트래픽의 경로를 추적합니다. 이렇게 하려면 프라이빗 서브넷의 Linux 인스턴스에서 traceroute 명령을 실행합니다. 출력에서 홉 중 하나(일반적으로 첫 번째 홉)에 NAT 게이트웨이의 프라이빗 IP 주소가 보아야 합니다.
- 프라이빗 서브넷의 인스턴스에서 소스 IP 주소에 연결할 때 이를 표시하는 타사 웹 사이트 또는 도구를 사용합니다. 소스 IP 주소는 NAT 게이트웨이의 탄력적 IP 주소여야 합니다.

이러한 테스트가 실패하는 경우 [NAT 게이트웨이 문제 해결](#) 섹션을 참조하세요.

인터넷 연결 테스트

다음 예제는 프라이빗 서브넷의 인스턴스가 인터넷에 연결할 수 있는지 테스트하는 방법을 보여 줍니다.

1. 퍼블릭 서브넷에서 인스턴스를 시작합니다(이 인스턴스를 Bastion Host로 사용). 시작 마법사에서 Amazon Linux AMI를 선택하고 인스턴스에 퍼블릭 IP 주소를 할당해야 합니다. 보안 그룹 규칙이 로컬 네트워크의 IP 주소 범위에서 전송되는 인바운드 SSH 트래픽을 허용하고, 프라이빗 서브넷의 IP 주소 범위로 전송되는 아웃바운드 SSH 트래픽을 허용하는지 확인합니다. 이 테스트에서는 인바운드 및 아웃바운드 SSH 트래픽 모두에 0.0.0.0/0을 사용할 수 있습니다.
2. 프라이빗 서브넷에서 인스턴스를 시작합니다. 시작 마법사에서 Amazon Linux AMI를 선택해야 합니다. 인스턴스에 퍼블릭 IP 주소를 할당하지 마세요. 보안 그룹 규칙이 퍼블릭 서브넷에서 시작한 인스턴스의 프라이빗 IP 주소에서 전송되는 인바운드 SSH 트래픽 및 모든 아웃바운드 ICMP 트래픽을 허용하는지 확인합니다. 퍼블릭 서브넷에서 인스턴스를 시작하는 데 사용한 것과 동일한 키 페어를 선택해야 합니다.
3. 로컬 컴퓨터에서 SSH 에이전트 전달을 구성하고, 퍼블릭 서브넷의 Bastion Host에 연결합니다. 자세한 내용은 [Linux 또는 macOS에 대한 SSH 에이전트 전달을 구성하려면](#) 또는 [Windows에 대한 SSH 에이전트 전달 구성 단원을](#) 참조하세요.
4. Bastion Host에서 프라이빗 서브넷의 인스턴스에 연결한 다음, 프라이빗 서브넷의 인스턴스에서 인터넷 연결을 테스트합니다. 자세한 내용은 [인터넷 연결을 테스트하려면](#) 단원을 참조하세요.

Linux 또는 macOS에 대한 SSH 에이전트 전달을 구성하려면

1. 로컬 시스템에서 인증 에이전트에 프라이빗 키를 추가합니다.

Linux의 경우 다음 명령을 사용합니다.

```
ssh-add -c mykeypair.pem
```

macOS의 경우 다음 명령을 사용합니다.

```
ssh-add -K mykeypair.pem
```

2. 다음 예제와 같이 `-A` 옵션으로 퍼블릭 서브넷의 인스턴스에 연결하여 SSH 에이전트 전달을 활성화하고 해당 인스턴스의 퍼블릭 주소를 사용합니다.

```
ssh -A ec2-user@54.0.0.123
```

Windows에 대한 SSH 에이전트 전달 구성

Windows에서 사용 가능한 OpenSSH 클라이언트를 사용하거나 선호하는 SSH 클라이언트(예: PuTTY)를 설치할 수 있습니다.

OpenSSH

[Getting started with OpenSSH for Windows](#)의 설명에 따라 Windows용 OpenSSH를 설치합니다. 그런 다음 인증 에이전트에 키를 추가합니다. 자세한 내용은 [Key-based authentication in OpenSSH for Windows](#)를 참조하세요.

PuTTY

1. Pageant가 아직 설치되어 있지 않으면 [PuTTY 다운로드 페이지](#)에서 Pageant를 다운로드하여 설치합니다.
2. 프라이빗 키를 .ppk 형식으로 변환합니다. 자세한 내용은 [Amazon EC2 사용 설명서의 PuTTYgen](#)을 사용하여 프라이빗 키 변환을 참조하세요.
3. Pageant를 시작하고 작업 표시줄의 Pageant 아이콘을 마우스 오른쪽 버튼으로 클릭한 다음 추가(Add Key)를 선택합니다. 생성한 .ppk 파일을 선택하고 필요에 따라 암호를 입력한 다음 열기(Open)를 선택합니다.

4. PuTTY 세션을 시작하고 퍼블릭 IP 주소를 사용하여 퍼블릭 서브넷의 인스턴스에 연결합니다. 자세한 내용은 [Linux 인스턴스 연결](#)을 참조하세요. [Auth] 범주에서 [Allow agent forwarding] 옵션을 선택하고 [Private key file for authentication] 상자를 공백 상태로 둡니다.

인터넷 연결을 테스트하려면

1. 다음 예제와 같이 퍼블릭 서브넷의 인스턴스에서 프라이빗 IP 주소를 사용하여 프라이빗 서브넷의 인스턴스에 연결합니다.

```
ssh ec2-user@10.0.1.123
```

2. 프라이빗 인스턴스에서 ICMP가 활성화된 웹 사이트에 대해 ping 명령을 실행하여 인터넷에 연결할 수 있는지 테스트합니다.

```
ping ietf.org
```

```
PING ietf.org (4.31.198.44) 56(84) bytes of data.
64 bytes from mail.ietf.org (4.31.198.44): icmp_seq=1 ttl=47 time=86.0 ms
64 bytes from mail.ietf.org (4.31.198.44): icmp_seq=2 ttl=47 time=75.6 ms
...
```

키보드에서 [Ctrl+C]를 눌러 ping 명령을 취소합니다. ping 명령이 실패할 경우 [인스턴스에서 인터넷에 액세스할 수 없음](#)을 참조하세요.

3. (선택 사항) 더 이상 필요하지 않으면 인스턴스를 종료합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [인스턴스 종료](#)를 참조하세요.

허용 목록에 있는 IP 주소를 사용하여 네트워크에 액세스

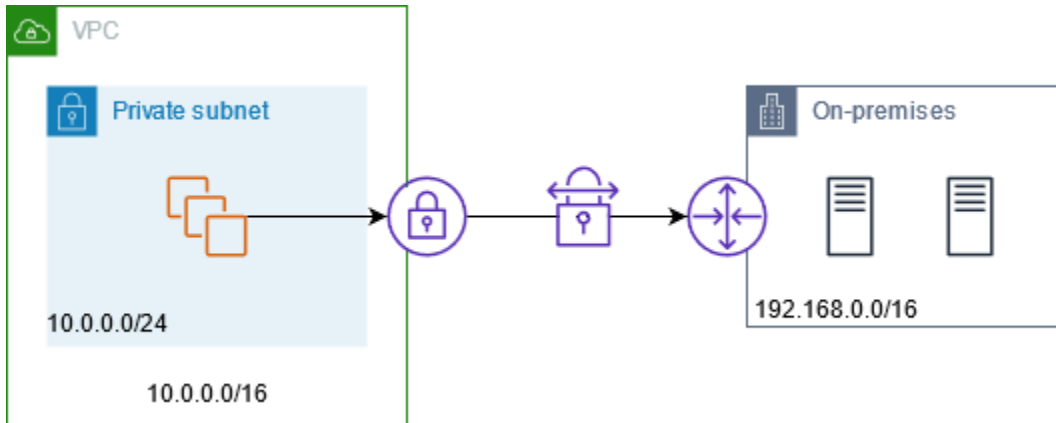
프라이빗 NAT 게이트웨이를 통해 허용 목록에 있는 주소 풀을 사용하여 VPC에서 온프레미스 네트워크로의 통신을 사용할 수 있습니다. 각 인스턴스에 허용 목록에 있는 IP 주소 범위의 별도 IP 주소를 할당하는 대신 허용 목록에 있는 IP 주소 범위의 IP 주소를 사용하여 프라이빗 NAT 게이트웨이를 통해 온프레미스 네트워크로 향하는 서브넷의 트래픽을 라우팅할 수 있습니다.

내용

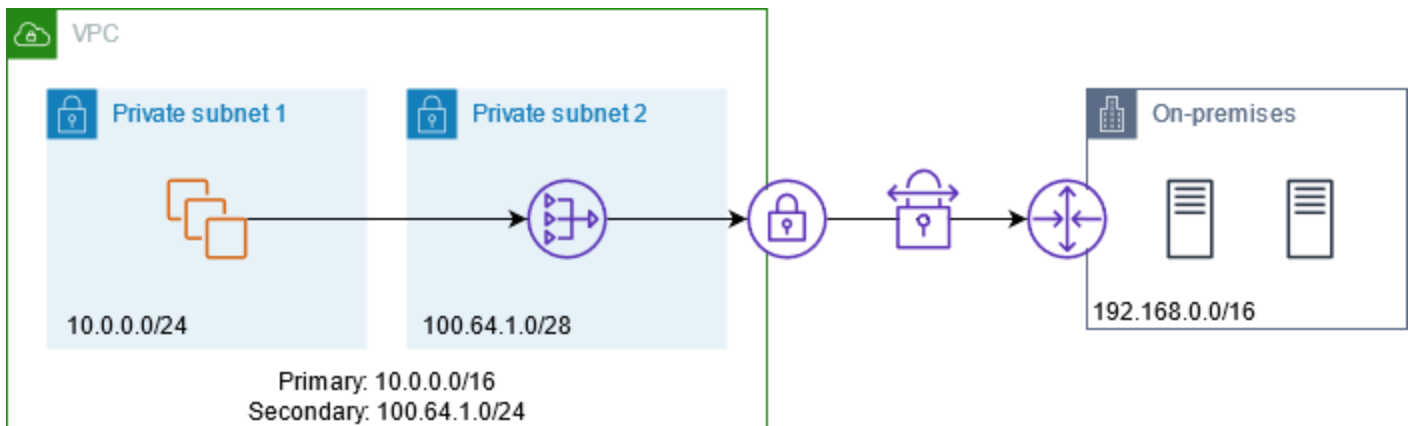
- [개요](#)
- [리소스](#)
- [라우팅](#)

개요

다음 다이어그램은 인스턴스가 AWS VPN을 통해 온프레미스 리소스에 액세스할 수 있는 방법을 보여줍니다. 인스턴스의 트래픽은 VPN 연결을 통해 가상 프라이빗 게이트웨이로 라우팅되고 고객 게이트웨이로 라우팅된 후 온프레미스 네트워크의 대상으로 라우팅됩니다. 그러나 대상에서 100.64.1.0/28과 같은 특정 IP 주소 범위의 트래픽만 허용된다고 가정해 보겠습니다. 이 경우에는 이러한 인스턴스의 트래픽이 온프레미스 네트워크에 도달하지 못하게 됩니다.



다음 다이어그램은 이 시나리오를 위한 구성의 주요 구성 요소를 보여줍니다. VPC에는 원본 IP 주소 범위와 함께 허용된 IP 주소 범위가 있습니다. VPC에는 프라이빗 NAT 게이트웨이가 있는 허용된 IP 주소 범위의 서브넷이 있습니다. 온프레미스 네트워크로 향하는 인스턴스의 트래픽은 VPN 연결로 라우팅되기 전에 NAT 게이트웨이로 전송됩니다. 온프레미스 네트워크는 허용된 IP 주소 범위에서 제공되는 NAT 게이트웨이의 소스 IP 주소가 있는 인스턴스에서 트래픽을 수신합니다.



리소스

다음과 같이 리소스를 생성하거나 업데이트합니다.

- 허용된 IP 주소 범위를 VPC와 연결합니다.
- 허용된 IP 주소 범위의 VPC에서 서브넷을 생성합니다.

- 새 서브넷에서 프라이빗 NAT 게이트웨이를 생성합니다.
- 서브넷의 라우팅 테이블을 인스턴스로 업데이트하여 온프레미스 네트워크로 향하는 트래픽을 NAT 게이트웨이로 전송합니다. 온프레미스 네트워크로 향하는 트래픽을 가상 프라이빗 게이트웨이로 전송하는 프라이빗 NAT 게이트웨이가 있는 서브넷의 라우팅 테이블에 경로를 추가합니다.

라우팅

다음은 첫 번째 서브넷과 연결된 라우팅 테이블입니다. 각 VPC CIDR에 대한 로컬 경로가 있습니다. 로컬 경로를 사용하면 서브넷의 리소스가 프라이빗 IP 주소를 사용하여 VPC의 다른 리소스와 통신할 수 있습니다. 세 번째 항목은 온프레미스 네트워크로 향하는 트래픽을 프라이빗 NAT 게이트웨이로 전송합니다.

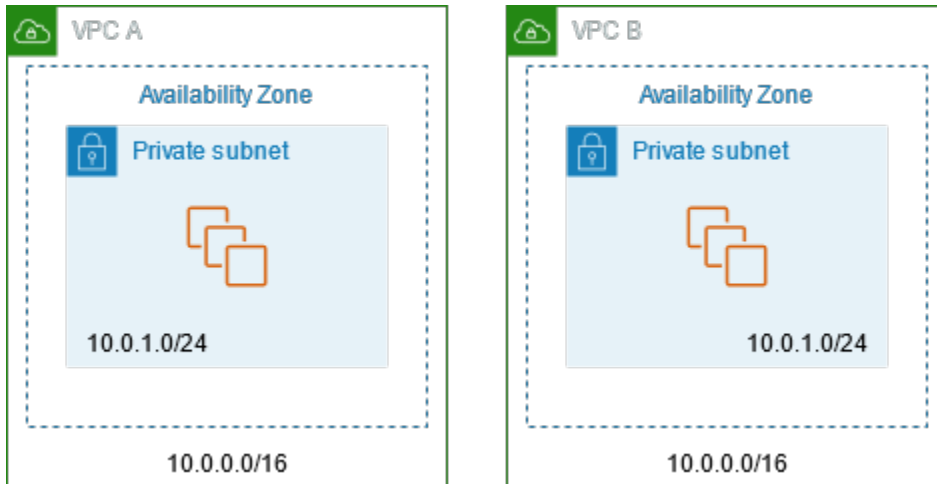
대상 주소	대상
<i>10.0.0.0/16</i>	로컬
<i>100.64.1.0/24</i>	로컬
<i>192.168.0.0/16</i>	<i>nat-gateway-id</i>

다음은 두 번째 서브넷과 연결된 라우팅 테이블입니다. 각 VPC CIDR에 대한 로컬 경로가 있습니다. 로컬 경로를 사용하면 서브넷의 리소스가 프라이빗 IP 주소를 사용하여 VPC의 다른 리소스와 통신할 수 있습니다. 세 번째 항목은 온프레미스 네트워크로 향하는 트래픽을 가상 프라이빗 게이트웨이로 전송합니다.

대상 주소	대상
<i>10.0.0.0/16</i>	로컬
<i>100.64.1.0/24</i>	로컬
<i>192.168.0.0/16</i>	<i>vgw-id</i>

중첩되는 네트워크 간에 통신 사용

프라이빗 NAT 게이트웨이를 사용하여 네트워크 간에 CIDR 범위가 중첩되더라도 통신할 수 있습니다. 예를 들어, VPC A의 인스턴스가 VPC B의 인스턴스에서 제공되는 서비스에 액세스해야 한다고 가정해 보십시오.



내용

- [개요](#)
- [리소스](#)
- [라우팅](#)

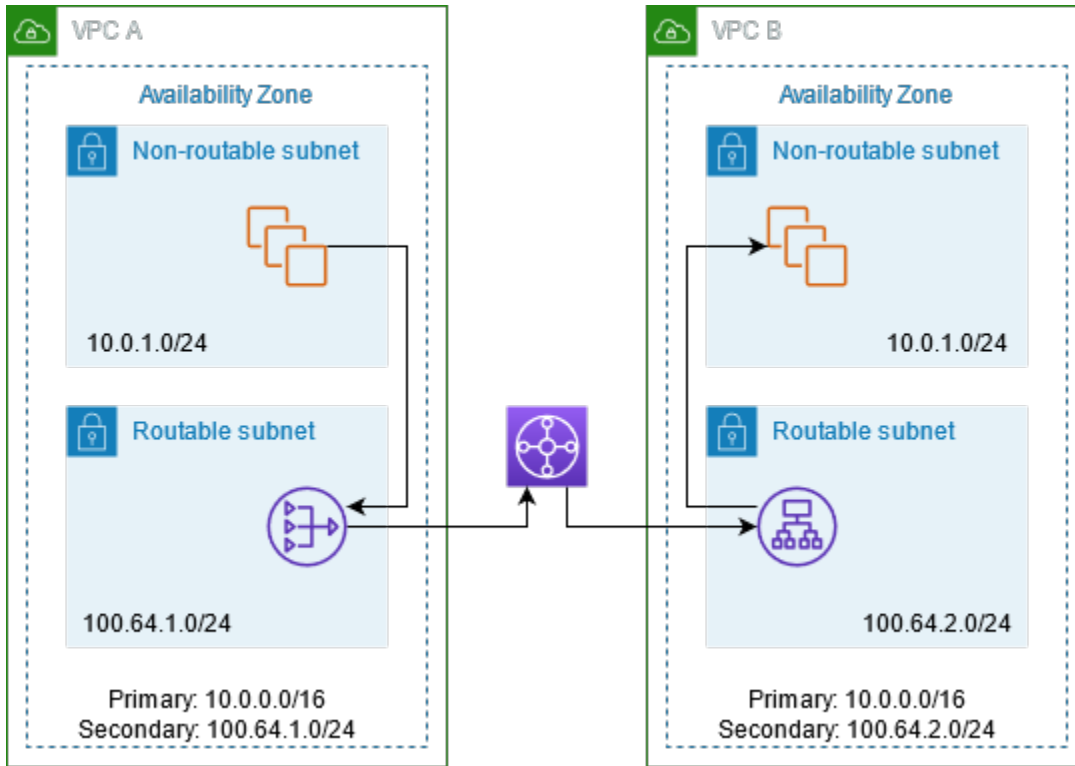
개요

다음 다이어그램은 이 시나리오를 위한 구성의 주요 구성 요소를 보여줍니다. 먼저 IP 관리 팀이 중첩될 수 있는 주소 범위(라우팅 불가능한 주소 범위) 및 중첩될 수 없는 주소 범위(라우팅 가능한 주소 범위)를 결정합니다. IP 관리 팀은 라우팅 가능한 주소 범위 풀에서 요청이 있을 경우 프로젝트에 주소 범위를 할당합니다.

각 VPC에는 라우팅이 불가능한 원래 IP 주소 범위와 IP 관리 팀에 의해 할당된 라우팅 가능한 IP 주소 범위가 있습니다. VPC A에는 프라이빗 NAT 게이트웨이가 포함된 라우팅 가능한 범위의 서브넷이 있습니다. 프라이빗 NAT 게이트웨이는 해당 서브넷에서 해당 IP 주소를 가져옵니다. VPC B에는 Application Load Balancer가 포함된 라우팅 가능한 범위의 서브넷이 있습니다. Application Load Balancer는 해당 서브넷에서 해당 IP 주소를 가져옵니다.

VPC B의 라우팅 불가능한 서브넷에 있는 인스턴스로 향하는 VPC A의 라우팅 불가능한 서브넷에 있는 인스턴스의 트래픽은 프라이빗 NAT 게이트웨이를 통해 전송된 다음 전송 게이트웨이로 라우팅됩니다. 전송 게이트웨이는 트래픽을 Application Load Balancer로 전송합니다. Application Load

Balancer는 VPC B의 라우팅 불가능한 서브넷에 있는 대상 인스턴스 중 하나로 해당 트래픽을 라우팅합니다. 전송 게이트웨이에서 Application Load Balancer로의 이 트래픽에는 프라이빗 NAT 게이트웨이의 소스 IP 주소가 있습니다. 따라서 로드 밸런서의 응답 트래픽은 프라이빗 NAT 게이트웨이의 주소를 대상으로 사용합니다. 응답 트래픽은 전송 게이트웨이로 보내진 후 프라이빗 NAT 게이트웨이로 라우팅됩니다. NAT 게이트웨이는 대상을 VPC A의 라우팅 불가능한 서브넷에 있는 인스턴스로 변환합니다.



리소스

다음과 같이 리소스를 생성하거나 업데이트합니다.

- 할당된 라우팅 가능한 IP 주소 범위를 각각의 해당 VPC에 연결합니다.
- 라우팅 가능한 IP 주소 범위의 VPC A에서 서브넷을 생성하고 이 새 서브넷에서 프라이빗 NAT 게이트웨이를 생성합니다.
- 라우팅 가능한 IP 주소 범위의 VPC B에서 서브넷을 생성하고 이 새 서브넷에서 Application Load Balancer를 생성합니다. 로드 밸런서의 대상 그룹에 라우팅 불가능한 서브넷의 인스턴스를 등록합니다.
- VPC를 연결할 전송 게이트웨이를 생성합니다. 경로 전파를 사용 중지했는지 확인합니다. 각 VPC를 전송 게이트웨이에 연결하는 경우에는 VPC의 라우팅 가능한 주소 범위를 사용하세요.
- VPC A의 라우팅 불가능한 서브넷의 라우팅 테이블을 업데이트하여 VPC B의 라우팅 가능한 주소 범위로 향하는 모든 트래픽을 프라이빗 NAT 게이트웨이로 전송합니다. VPC A의 라우팅 가능한 서

브넷의 라우팅 테이블을 업데이트하여 VPC B의 라우팅 가능한 주소 범위로 향하는 모든 트래픽을 전송 게이트웨이로 전송합니다.

- VPC B의 라우팅 가능한 서브넷의 라우팅 테이블을 업데이트하여 VPC A의 라우팅 가능한 주소 범위로 향하는 모든 트래픽을 전송 게이트웨이로 전송합니다.

라우팅

다음은 VPC A의 라우팅 불가능한 서브넷에 대한 라우팅 테이블입니다.

대상 주소	대상
<i>10.0.0.0/16</i>	로컬
<i>100.64.1.0/24</i>	로컬
<i>100.64.2.0/24</i>	<i>nat-gateway-id</i>

다음은 VPC A의 라우팅 가능한 서브넷에 대한 라우팅 테이블입니다.

대상 주소	대상
<i>10.0.0.0/16</i>	로컬
<i>100.64.1.0/24</i>	로컬
<i>100.64.2.0/24</i>	<i>transit-gateway-id</i>

다음은 VPC B의 라우팅 불가능한 서브넷에 대한 라우팅 테이블입니다.

대상 주소	대상
<i>10.0.0.0/16</i>	로컬
<i>100.64.2.0/24</i>	로컬

다음은 VPC B의 라우팅 가능한 서브넷에 대한 라우팅 테이블입니다.

대상 주소	대상
<i>10.0.0.0/16</i>	로컬
<i>100.64.2.0/24</i>	로컬
<i>100.64.1.0/24</i>	<i>transit-gateway-id</i>

다음은 전송 게이트웨이 라우팅 테이블입니다.

CIDR	연결	경로 유형
<i>100.64.1.0/24</i>	<i>VPC A ##</i>	정적
<i>100.64.2.0/24</i>	<i>VPC B ##</i>	정적

DNS64 및 NAT64

NAT 게이트웨이는 IPv6에서 IPv4로의 네트워크 주소 변환을 지원합니다(일반적으로 NAT64라고 함). NAT64는 IPv6 AWS 리소스가 동일한 VPC 또는 다른 VPC, 온프레미스 네트워크 또는 인터넷을 통해 IPv4 리소스와 통신하도록 지원합니다. Amazon Route 53 Resolver 기반 DNS64를 포함하여 NAT64를 사용하거나 자체 DNS64 서버를 사용할 수 있습니다.

내용

- [DNS64란 무엇입니까?](#)
- [NAT64란 무엇입니까?](#)
- [DNS64 및 NAT64 구성](#)

DNS64란 무엇입니까?

VPC에서 실행되는 IPv6 전용 워크로드에는 IPv6 네트워크 패킷만 보내고 받을 수 있습니다. DNS64 없이는 IPv4 전용 서비스에 대한 DNS 쿼리는 응답으로 IPv4 대상 주소를 출력하며 IPv6 전용 서비스와 통신할 수 없습니다. 이러한 통신 격차를 해소하기 위해 서브넷에 대해 DNS64를 활성화하면 서브넷 내의 모든 해당 AWS 리소스에 적용됩니다. DNS64에서는 Amazon Route 53 Resolver가 쿼리한 서비스에 대한 DNS 레코드를 조회하고 다음 중 하나를 수행합니다.

- 레코드에 IPv6 주소가 포함되어 있으면 원래 레코드를 반환하고 IPv6을 통한 변환 없이 연결이 설정됩니다.
- DNS 레코드의 대상과 연결된 IPv6 주소가 없는 경우 Route 53 Resolver는 레코드의 IPv4 주소에 대하여 RFC6052(64:ff9b::/96)에 정의된 잘 알려진 /96 접두사를 가장하여 주소를 합성합니다. IPv6 전용 서비스는 네트워크 패킷을 합성된 IPv6 주소로 보냅니다. 그런 다음 NAT 게이트웨이를 통해 이 트래픽을 라우팅해야 합니다. 이 게이트웨이는 트래픽에 대해 필요한 변환을 수행하여 서브넷의 IPv6 서비스가 해당 서브넷 외부의 IPv4 서비스에 액세스할 수 있도록 합니다.

AWS CLI로 [modify-subnet-attribute](#)를 사용하거나 서브넷을 선택하고 작업(Actions) > 서브넷 설정 편집(Edit subnet settings)을 선택하여 VPC 콘솔에서 서브넷의 DNS64를 사용 또는 사용 중지할 수 있습니다.

NAT64란 무엇입니까?

NAT64를 사용하면 Amazon VPC의 IPv6 전용 서비스가 동일한 VPC(다른 서브넷에 있음) 또는 연결된 VPC, 온프레미스 네트워크 또는 인터넷을 통해 IPv4 전용 서비스와 통신할 수 있습니다.

NAT64는 기존 NAT 게이트웨이 또는 새로 만든 NAT 게이트웨이에서 자동으로 사용할 수 있습니다. 이 기능을 활성화하거나 비활성화할 수 없습니다. NAT 게이트웨이가 있는 서브넷은 이중 스택 서브넷이 아니어도 NAT64가 작동합니다.

DNS64를 사용하도록 설정한 후 IPv6 전용 서비스가 NAT 게이트웨이를 통해 합성된 IPv6 주소로 네트워크 패킷을 전송하면 다음과 같은 일이 발생합니다.

- 64:ff9b::/96 접두사에서 NAT 게이트웨이는 원래 대상이 IPv4임을 인식하고 다음을 대체하여 IPv6 패킷을 IPv4로 변환합니다.
 - 인터넷 게이트웨이에 의해 탄력적 IP 주소로 변환되는 자체 프라이빗 IP가 있는 소스 IPv6
 - 64:ff9b::/96 접두사를 잘라서 대상 IPv6에서 IPv4로 연결합니다.
- NAT 게이트웨이는 인터넷 게이트웨이, 가상 프라이빗 게이트웨이 또는 전송 게이트웨이를 통해 변환된 IPv4 패킷을 대상으로 전송하고 연결을 시작합니다.
- IPv4 전용 호스트는 IPv4 응답 패킷을 다시 반환합니다. 연결이 설정된 후 NAT 게이트웨이는 외부 호스트에서 응답 IPv4 패킷을 수락합니다.
- 응답 IPv4 패킷은 NAT 게이트웨이로 향합니다. 이 게이트웨이는 IP(대상 IP)를 호스트의 IPv6 주소로 바꾸고 소스 IPv4 주소 패킷을 64:ff9b::/96로 다시 가장하여 패킷을 수신합니다. 그런 다음 패킷은 로컬 경로를 따라 호스트로 흐릅니다.

이러한 방식으로 NAT 게이트웨이를 사용하면 서브넷의 IPv6 전용 워크로드가 서브넷 외부의 IPv4 전용 서비스와 통신할 수 있습니다.

DNS64 및 NAT64 구성

이 단원의 단계에 따라 IPv4 전용 서비스와의 통신을 사용하도록 DNS64 및 NAT64를 구성합니다.

내용

- [AWS CLI를 통해 인터넷에서 IPv4 전용 서비스와의 통신을 활성화합니다.](#)
- [온프레미스 환경에서 IPv4 전용 서비스와의 통신 사용](#)

AWS CLI를 통해 인터넷에서 IPv4 전용 서비스와의 통신을 활성화합니다.

서브넷 외부의 IPv4 전용 서비스와 통신해야 하는 IPv6 전용 워크로드를 포함한 서브넷이 있는 경우, 이 예에서는 이러한 IPv6 전용 서비스가 인터넷에서 IPv4 전용 서비스와 통신하도록 설정하는 방법을 보여 줍니다.

먼저 퍼블릭 서브넷(IPv6 전용 워크로드가 포함된 서브넷과는 별개)에서 NAT 게이트웨이를 구성해야 합니다. 예를 들어 NAT 게이트웨이가 포함된 서브넷에는 인터넷 게이트웨이를 가리키는 `0.0.0.0/0` 경로가 있어야 합니다.

이러한 IPv6 전용 서비스가 인터넷에서 IPv4 전용 서비스에 연결할 수 있도록 하려면 다음 단계를 완료 하세요.

1. IPv6 전용 워크로드가 포함된 서브넷의 라우팅 테이블에 다음 세 가지 경로를 추가합니다.
 - NAT 게이트웨이를 가리키는 IPv4 경로(있는 경우).
 - NAT 게이트웨이를 가리키는 `64:ff9b::/96` 경로. 이렇게 하면 IPv4 전용 서비스로 향하는 IPv6 전용 워크로드의 트래픽이 NAT 게이트웨이를 통해 라우팅될 수 있습니다.
 - 송신 전용 인터넷 게이트웨이(또는 인터넷 게이트웨이)를 가리키는 IPv6 `::/0` 경로.

참고로 `::/0`이 인터넷 게이트웨이를 가리키면 외부 IPv6 호스트(VPC 외부)가 IPv6을 통한 연결을 시작할 수 있습니다.

```
aws ec2 create-route --route-table-id rtb-34056078 --destination-cidr-block
0.0.0.0/0 --nat-gateway-id nat-05dba92075d71c408
```

```
aws ec2 create-route --route-table-id rtb-34056078 --destination-ipv6-cidr-block 64:ff9b::/96 --nat-gateway-id nat-05dba92075d71c408
```

```
aws ec2 create-route --route-table-id rtb-34056078 --destination-ipv6-cidr-block ::/0 --egress-only-internet-gateway-id eigw-c0a643a9
```

2. IPv6 전용 워크로드가 포함된 서브넷에서 DNS64 기능을 활성화합니다.

```
aws ec2 modify-subnet-attribute --subnet-id subnet-1a2b3c4d --enable-dns64
```

이제 프라이빗 서브넷의 리소스가 인터넷의 IPv4 및 IPv6 서비스와 상태 저장 연결을 설정할 수 있습니다. 64:ff9b::/96 트래픽에 대한 송신 및 수신 트래픽을 허용하도록 보안 그룹 및 NACL을 적절히 구성합니다.

온프레미스 환경에서 IPv4 전용 서비스와의 통신 사용

Amazon Route 53 Resolver를 사용하면 VPC를 온프레미스 네트워크로 또는 그 반대로 DNS 쿼리를 전달할 수 있습니다. 이를 위해 다음 정책을 사용할 수 있습니다.

- VPC에 Route 53 Resolver 아웃바운드 엔드포인트를 생성하고 Route 53 Resolver가 쿼리를 전달할 IPv4 주소를 할당합니다. 온프레미스 DNS 해석기의 경우 이 주소는 DNS 쿼리가 시작되는 IP 주소이므로 IPv4 주소여야 합니다.
- Route 53 Resolver가 온프레미스 해석기에 전달할 DNS 쿼리의 도메인 이름을 지정하는 규칙을 하나 이상 만듭니다. 또한 온프레미스 해석기의 IPv4 주소를 지정합니다.
- 이제 Route 53 Resolver 아웃바운드 엔드포인트를 설정했으므로 IPv6 전용 워크로드가 포함된 서브넷에서 DNS64를 활성화하고 NAT 게이트웨이를 통해 온프레미스 네트워크를 대상으로 하는 모든 데이터를 라우팅해야 합니다.

온프레미스 네트워크의 IPv4 전용 대상에서 DNS64 작동 방식:

1. VPC의 Route 53 Resolver 아웃바운드 엔드포인트에 IPv4 주소를 할당합니다.
2. IPv6 서비스의 DNS 쿼리는 IPv6을 통한 Route 53 Resolver로 이동합니다. Route 53 Resolver는 전달 규칙과 쿼리를 일치시키고 온프레미스 해석기의 IPv4 주소를 가져옵니다.

3. Route 53 Resolver는 쿼리 패킷을 IPv6에서 IPv4로 변환하여 아웃바운드 엔드포인트로 전달합니다. 엔드포인트의 각 IP 주소는 DNS 해석기의 온프레미스 IPv4 주소로 요청을 전달하는 하나의 ENI를 나타냅니다.
4. 온프레미스 해석기는 IPv4를 사용하여 아웃바운드 엔드포인트를 통해 Route 53 Resolver로 응답 패킷을 다시 보냅니다.
5. 쿼리가 DNS64 지원 서브넷에서 수행되었다고 가정하면 Route 53 Resolver는 다음 두 가지 작업을 수행합니다.
 - a. 응답 패킷의 내용을 확인합니다. 레코드에 IPv6 주소가 있는 경우 콘텐츠는 그대로 유지되지만 IPv4 레코드만 포함된 경우에는 IPv4 주소에 대하여 64:ff9b::/96을 가장하여 IPv6 레코드도 합성합니다.
 - b. 콘텐츠를 다시 패키징하여 IPv6을 통해 VPC 서비스로 전송합니다.

Amazon CloudWatch를 사용하여 NAT 게이트웨이 모니터링

CloudWatch를 이용하여 NAT 게이트웨이를 모니터링하면 NAT 게이트웨이에 대한 정보를 수집하고, 거의 실시간에 가까운 읽기 가능한 지표를 만들 수 있습니다. 이 정보를 사용하여 NAT 게이트웨이를 모니터링하고 문제를 해결할 수 있습니다. NAT 게이트웨이 지표 데이터는 1분마다 제공되며, 통계는 15개월 동안 기록됩니다.

Amazon CloudWatch에 대한 자세한 내용은 [Amazon CloudWatch 사용 설명서](#)를 참조하세요. 요금에 대한 자세한 정보는 [Amazon CloudWatch 요금](#)을 참조하세요.

NAT 게이트웨이 지표 및 차원

NAT 게이트웨이에 사용할 수 있는 측정치는 아래와 같습니다. 설명 옆에는 각 지표에 대한 설명과 [단위](#) 및 [통계](#)가 포함됩니다.

지표	설명
ActiveConnectionCount	NAT 게이트웨이를 통한 동시 활성 TCP 연결의 총 수입니다. 0의 값은 NAT 게이트웨이를 통한 활성 연결이 없음을 나타냅니다. 단위: 개수

지표	설명
	통계: 가장 유용한 통계는 Max입니다.
BytesInFromDestination	<p>NAT 게이트웨이가 대상으로부터 수신한 바이트 수입입니다.</p> <p>BytesOutToSource 값이 BytesInFromDestination 값보다 작은 경우, NAT 게이트웨이 처리 중에 데이터 손실이 있거나 NAT 게이트웨이가 적극적으로 차단하는 트래픽이 있을 수도 있습니다.</p> <p>단위: 바이트</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>
BytesInFromSource	<p>NAT 게이트웨이가 VPC 내 클라이언트로부터 수신한 바이트 수입입니다.</p> <p>BytesOutToDestination 값이 BytesInFromSource 값보다 작은 경우, NAT 게이트웨이 처리 중에 데이터 손실이 있을 수도 있습니다.</p> <p>단위: 바이트</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>

지표	설명
BytesOutToDestination	<p>NAT 게이트웨이를 통해 대상으로 전송된 바이트 수입입니다.</p> <p>0보다 큰 값은 NAT 게이트웨이 뒤에 있는 클라이언트에서 인터넷으로 가는 트래픽이 있음을 나타냅니다. BytesOutToDestination 값이 BytesInFromSource 값보다 작은 경우, NAT 게이트웨이 처리 중에 데이터 손실이 있을 수도 있습니다.</p> <p>단위: 바이트</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>
BytesOutToSource	<p>NAT 게이트웨이를 통해 VPC 내 클라이언트로 전송된 바이트 수입입니다.</p> <p>0보다 큰 값은 인터넷에서 NAT 게이트웨이 뒤에 있는 클라이언트로 오는 트래픽이 있음을 나타냅니다. BytesOutToSource 값이 BytesInFromDestination 값보다 작은 경우, NAT 게이트웨이 처리 중에 데이터 손실이 있거나 NAT 게이트웨이가 적극적으로 차단하는 트래픽이 있을 수도 있습니다.</p> <p>단위: 바이트</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>

지표	설명
ConnectionAttemptCount	<p>NAT 게이트웨이를 통해 이루어진 연결 시도 횟수.</p> <p>ConnectionEstablishedCount 값이 ConnectionAttemptCount 값보다 작은 경우, NAT 게이트웨이 뒤의 클라이언트가 응답이 없는 새 연결을 시도했음을 나타냅니다.</p> <p>단위: 수</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>
ConnectionEstablishedCount	<p>NAT 게이트웨이를 통해 설정된 연결 수.</p> <p>ConnectionEstablishedCount 값이 ConnectionAttemptCount 값보다 작은 경우, NAT 게이트웨이 뒤의 클라이언트가 응답이 없는 새 연결을 시도했음을 나타냅니다.</p> <p>단위: 수</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>
ErrorPortAllocation	<p>NAT 게이트웨이가 소스 포트 할당에 실패한 횟수.</p> <p>0보다 큰 값은 너무 많은 동시 연결이 NAT 게이트웨이를 통해 열려 있음을 나타냅니다.</p> <p>단위: 개수</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>

지표	설명
IdleTimeoutCount	<p>활성 상태가 유휴 상태로 전환된 연결 수입니다. 활성 연결은 적절하게 종료되지 않고 직전 350 초 동안 활동이 없는 경우 유휴 상태로 전환됩니다.</p> <p>0보다 큰 값은 유휴 상태로 이동된 연결이 있었음을 나타냅니다. IdleTimeoutCount 값이 증가하는 경우, NAT 게이트웨이 뒤의 클라이언트가 부실 연결을 재사용하고 있음을 나타낼 수도 있습니다.</p> <p>단위: 수</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>
PacketsDropCount	<p>NAT 게이트웨이가 삭제한 패킷 수입니다.</p> <p>삭제된 패킷 수를 전체 패킷 트래픽의 백분율로 계산하려면 $\text{PacketsDropCount} / (\text{PacketsInFromSource} + \text{PacketsInFromDestination}) * 100$ 공식을 사용합니다. 이 값이 NAT 게이트웨이의 총 트래픽 중 0.01퍼센트를 초과한다면 Amazon VPC 서비스에 문제가 있는 것일 수 있습니다. NAT 게이트웨이에서 패킷이 삭제되는 원인이 될 수 있는 서비스 관련 문제는 AWS 서비스 상태 대시보드를 사용하여 식별할 수 있습니다.</p> <p>단위: 개</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>

지표	설명
PacketsInFromDestination	<p>NAT 게이트웨이가 대상으로부터 수신한 패킷 수입입니다.</p> <p>PacketsOutToSource 값이 PacketsInFromDestination 값보다 작은 경우, NAT 게이트웨이 처리 중에 데이터 손실이 있거나 NAT 게이트웨이가 적극적으로 차단하는 트래픽이 있을 수도 있습니다.</p> <p>단위: 수</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>
PacketsInFromSource	<p>NAT 게이트웨이가 VPC 내 클라이언트로부터 수신한 패킷 수입입니다.</p> <p>PacketsOutToDestination 값이 PacketsInFromSource 값보다 작은 경우, NAT 게이트웨이 처리 중에 데이터 손실이 있을 수도 있습니다.</p> <p>단위: 수</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>
PacketsOutToDestination	<p>NAT 게이트웨이를 통해 대상으로 전송된 패킷 수입입니다.</p> <p>0보다 큰 값은 NAT 게이트웨이 뒤에 있는 클라이언트에서 인터넷으로 가는 트래픽이 있음을 나타냅니다. PacketsOutToDestination 값이 PacketsInFromSource 값보다 작은 경우, NAT 게이트웨이 처리 중에 데이터 손실이 있을 수도 있습니다.</p> <p>단위: 수</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>

지표	설명
PacketsOutToSource	<p>NAT 게이트웨이를 통해 VPC 내 클라이언트로 전송된 패킷 수입니다.</p> <p>0보다 큰 값은 인터넷에서 NAT 게이트웨이 뒤에 있는 클라이언트로 오는 트래픽이 있음을 나타냅니다. PacketsOutToSource 값이 PacketsInFromDestination 값보다 작은 경우, NAT 게이트웨이 처리 중에 데이터 손실이 있거나 NAT 게이트웨이가 적극적으로 차단하는 트래픽이 있을 수도 있습니다.</p> <p>단위: 수</p> <p>통계: 가장 유용한 통계는 Sum입니다.</p>
PeakBytesPerSecond	<p>이 지표는 특정 분당 가장 높은 초당 10초 바이트 평균값을 보고합니다.</p> <p>단위: 개</p> <p>통계: 가장 유용한 통계는 Maximum입니다.</p>
PeakPacketsPerSecond	<p>이 지표는 60초 동안 10초마다 평균 패킷 속도 (초당 처리된 패킷 수)를 계산한 다음 최대 6개 속도(가장 높은 평균 패킷 속도)를 보고합니다.</p> <p>단위: 개</p> <p>통계: 가장 유용한 통계는 Maximum입니다.</p>

지표 데이터를 필터링하려면 다음 차원을 사용하세요.

차원	설명
NatGatewayId	NAT 게이트웨이 ID를 기준으로 측정치 데이터를 필터링합니다.

NAT 게이트웨이 CloudWatch 지표 보기

NAT 게이트웨이 지표는 1분 가격으로 CloudWatch로 전송됩니다. 지표는 먼저 서비스 네임스페이스 별로 그룹화된 다음, 각 네임스페이스 내에서 가능한 차원 조합에 따라 그룹화됩니다. NAT 게이트웨이에 대해 다음과 같이 측정치를 볼 수 있습니다.

CloudWatch 콘솔을 사용하여 지표를 보려면

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 지표, 모든 지표를 선택합니다.
3. NATGateway 지표 네임스페이스를 선택합니다.
4. 지표 차원을 선택합니다.

를 사용하여 지표를 보려면AWS CLI

명령 프롬프트에서 다음 명령을 사용하여 NAT 게이트웨이 서비스에 사용 가능한 지표 목록을 확인합니다.

```
aws cloudwatch list-metrics --namespace "AWS/NATGateway"
```

NAT 게이트웨이를 모니터링하기 위한 CloudWatch 경보 생성

경보로 인해 상태가 변경되면 Amazon SNS 메시지를 보내는 CloudWatch 경보를 생성할 수 있습니다. 경보는 지정한 기간 동안 단일 지표를 감시합니다. 경보는 기간 수에 대한 주어진 임계값과 지표 값을 비교하여 Amazon SNS 주제에 알림을 보냅니다.

예를 들어 NAT 게이트웨이로 들어오거나 나가는 트래픽의 양을 모니터링하는 경보를 만들 수 있습니다. 아래 경보는 NAT 게이트웨이를 통해 VPC의 클라이언트에서 인터넷으로 가는 아웃바운드 트래픽의 양을 모니터링합니다. 그리고 15분 동안 바이트 수가 임계값인 5,000,000에 도달하면 알림을 보냅니다.

NAT 게이트웨이를 통한 아웃바운드 트래픽에 대한 경보를 만들려면

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 경보(Alarms) 모든 경보(All Alarms)를 선택합니다.
3. 경보 생성(Create alarm)을 선택하세요.
4. 지표 선택(Select metric)을 선택하세요.

5. NATGateway 지표 네임스페이스를 선택한 다음 지표 차원을 선택합니다. 지표를 가져오면 NAT 게이트웨이에 대한 BytesOutToDestination 지표 옆 확인란을 선택한 다음 지표 선택을 선택합니다.
6. 경보를 다음과 같이 구성한 다음 다음(Next)을 선택합니다.
 - 통계(Statistic)에서 합계(Sum)를 선택합니다.
 - 기간에서 15분을 선택합니다.
 - 항상에서 초과/같음을 선택하고 임계값으로 5000000을 입력합니다.
7. 알림에서 기존 SNS 주제를 선택하거나 새 주제 생성을 선택하여 새로 생성합니다. Next(다음)를 선택합니다.
8. 경보의 이름과 설명을 입력하고 다음을 선택합니다.
9. 경보 구성을 마쳤으면 경보 생성을 선택합니다.

다른 예와 같이, 포트 할당 오류를 모니터링하는 경보를 만들고 이 값이 3회 연속 5분간 0보다 클 경우에 알림을 보낼 수 있습니다.

경보를 만들어 포트 할당 오류를 모니터링하려면

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 경보(Alarms) 모든 경보(All Alarms)를 선택합니다.
3. 경보 생성(Create alarm)을 선택하세요.
4. 지표 선택(Select metric)을 선택하세요.
5. NATGateway 지표 네임스페이스를 선택한 다음 지표 차원을 선택합니다. 지표를 가져오면 NAT 게이트웨이에 대한 ErrorPortAllocation 지표 옆 확인란을 선택한 다음 지표 선택을 선택합니다.
6. 경보를 다음과 같이 구성한 다음 다음(Next)을 선택합니다.
 - 통계에서 최대를 선택합니다.
 - 기간에서 5분을 선택합니다.
 - 항상에서 초과를 선택하고 임계값으로 0을 입력합니다.
 - 추가 구성에서 경보를 생성할 데이터 포인트에 대해 3을 입력합니다.
7. 알림에서 기존 SNS 주제를 선택하거나 새 주제 생성을 선택하여 새로 생성합니다. Next(다음)를 선택합니다.
8. 경보의 이름과 설명을 입력하고 다음을 선택합니다.
9. 경보 구성이 완료되면 경보 생성을 선택합니다.

자세한 정보는 [Amazon CloudWatch 사용 설명서](#)의 Amazon CloudWatch 경보 사용을 참조하세요.

NAT 게이트웨이 문제 해결

다음 주제는 NAT 게이트웨이를 만들거나 사용할 때 발생할 수 있는 일반적인 문제를 해결하는 데 도움이 됩니다.

문제

- [NAT 게이트웨이 생성 실패](#)
- [NAT 게이트웨이 할당량](#)
- [탄력적 IP 주소 할당량](#)
- [가용 영역이 지원되지 않음](#)
- [NAT 게이트웨이가 더 이상 표시되지 않음](#)
- [NAT 게이트웨이가 Ping 명령에 응답하지 않음](#)
- [인스턴스에서 인터넷에 액세스할 수 없음](#)
- [대상에 대한 TCP 연결 실패](#)
- [경로 추적 출력에 NAT 게이트웨이 프라이빗 IP 주소가 표시되지 않음](#)
- [350초 후 인터넷 연결이 끊어짐](#)
- [IPSec 연결을 설정할 수 없음](#)
- [추가 연결을 시작할 수 없음](#)

NAT 게이트웨이 생성 실패

문제

NAT 게이트웨이를 생성하면 상태가 Failed가 됩니다.

Note

실패한 NAT 게이트웨이는 일반적으로 약 1시간 내에 자동으로 삭제됩니다.

원인

NAT 게이트웨이를 생성할 때 오류가 발생했습니다. 반환된 상태 메시지를 통해 오류 원인을 확인할 수 있습니다.

Solution

오류 메시지를 보려면 Amazon VPC 콘솔 열 연 후 NAT 게이트웨이를 선택하세요. NAT 게이트웨이에 대한 라디오 버튼을 선택한 후, 세부 정보 탭에서 상태 메시지를 찾습니다.

다음 표에는 Amazon VPC 콘솔에 표시되는 오류의 예상 원인이 나와 있습니다. 표시된 수정 단계를 적용한 후 NAT 게이트웨이를 다시 만들어 볼 수 있습니다.

표시된 오류	원인	솔루션
서브넷에 이 NAT 게이트웨이를 만들 수 있는 사용 가능한 주소가 부족함	지정한 서브넷에 사용 가능한 프라이빗 IP 주소가 없습니다. NAT 게이트웨이에는 서브넷의 범위에서 할당된 프라이빗 IP 주소가 있는 네트워크 인터페이스가 필요합니다.	Amazon VPC 콘솔의 서브넷 페이지로 이동하여 서브넷에서 사용 가능한 IP 주소 개수를 확인합니다. 서브넷의 세부 정보 창에서 사용 가능한 IP를 볼 수 있습니다. 서브넷에서 사용 가능한 IP 주소를 만들려면 사용되지 않은 네트워크 인터페이스를 삭제하거나 필요 없는 인스턴스를 종료할 수 있습니다.
네트워크 vpc-xxxxxxx에 연결된 인터넷 게이트웨이가 없음	인터넷 게이트웨이가 있는 VPC에서 NAT 게이트웨이를 만들어야 합니다.	인터넷 게이트웨이를 생성하여 VPC에 연결합니다. 자세한 내용은 인터넷 게이트웨이 작업 단원을 참조하십시오.
탄력적 IP 주소 eipalloc-xxxxxxx가 이미 연결되어 있음	지정한 탄력적 IP 주소가 다른 리소스와 이미 연결되어 있으므로 NAT 게이트웨이와 연결할 수 없습니다.	탄력적 IP 주소와 연결된 리소스를 확인합니다. Amazon VPC 콘솔에서 탄력적 IP 페이지로 이동하여 인스턴스 ID 또는 네트워크 인터페이스 ID에 지정된 값을 확인합니다. 해당 리소스에 탄력적 IP 주소가 필요 없는 경우 주소를 연결 해제할 수 있습니다. 또는 계정에 새로운 탄력적 IP 주소를 할당합니다. 자세한 내용은 탄력적 IP

표시된 오류	원인	솔루션
		주소 작업 단원을 참조하십시오.

NAT 게이트웨이 할당량

NAT 게이트웨이를 생성하려고 하면 다음 오류가 발생합니다.

Performing this operation would exceed the limit of 5 NAT gateways

원인

해당 가용 영역에 대한 NAT 게이트웨이 수가 할당량에 도달했습니다.

Solution

계정에 대한 이 NAT 게이트웨이 할당량에 도달한 경우, 다음 중 하나를 수행할 수 있습니다.

- Service Quotas 콘솔을 사용하여 [가용 영역 할당량당 NAT 게이트웨이](#) 증가를 요청합니다.
- NAT 게이트웨이의 상태를 확인합니다. Pending, Available 또는 Deleting의 상태는 할당량에 포함됩니다. NAT 게이트웨이를 최근에 삭제한 경우 상태가 Deleting에서 Deleted로 바뀔 때까지 몇 분간 기다립니다. 그런 다음 새 NAT 게이트웨이를 생성해 보세요.
- 특정 가용 영역에 NAT 게이트웨이가 필요 없는 경우 할당량에 도달하지 않은 가용 영역에서 NAT 게이트웨이를 만들어 봅니다.

자세한 내용은 [Amazon VPC 할당량](#) 단원을 참조하세요.

탄력적 IP 주소 할당량

문제

퍼블릭 NAT 게이트웨이에 탄력적 IP 주소를 할당하려고 하면 다음 오류가 발생합니다.

The maximum number of addresses has been reached.

원인

해당 리전의 계정에 대한 탄력적 IP 주소 수가 할당량에 도달했습니다.

Solution

탄력적 IP 주소 할당량에 도달한 경우 다른 리소스에서 탄력적 IP 주소의 연결을 해제할 수 있습니다. 또는 Service Quotas 콘솔을 사용하여 [탄력적 IP 할당량](#) 증가를 요청할 수 있습니다.

가용 영역이 지원되지 않음

문제

NAT 게이트웨이를 생성하려고 하면 NotAvailableInZone 오류가 발생합니다.

원인

제약이 있는 가용 영역(확장이 제약되어 있는 영역)에서 NAT 게이트웨이를 생성하려 했을지도 모릅니다.

Solution

이러한 가용 영역에서는 NAT 게이트웨이를 지원하지 않습니다. 다른 가용 영역에서 NAT 게이트웨이를 만들고 제약이 있는 영역의 프라이빗 서브넷에 사용할 수 있습니다. 리소스와 NAT 게이트웨이가 동일한 영역에 있도록 제약이 없는 가용 영역으로 리소스를 이동할 수도 있습니다.

NAT 게이트웨이가 더 이상 표시되지 않음

문제

NAT 게이트웨이를 만들었지만 Amazon VPC 콘솔에 표시되지 않습니다.

원인

NAT 게이트웨이를 만드는 동안 오류가 발생하여 생성에 실패했을 수 있습니다. Failed 상태의 NAT 게이트웨이는 Amazon VPC 콘솔에 1시간 가량 표시됩니다. 한 시간 후에는 자동으로 삭제됩니다.

Solution

[NAT 게이트웨이 생성 실패](#)에서 정보를 검토하고 새 NAT 게이트웨이를 만들어 봅니다.

NAT 게이트웨이가 Ping 명령에 응답하지 않음

문제

인터넷에서(예를 들어 홈 컴퓨터에서) 또는 VPC의 인스턴스에서 NAT 게이트웨이의 탄력적 IP 주소 또는 프라이빗 IP 주소를 ping하려고 시도하는 경우 응답을 얻을 수 없습니다.

원인

NAT 게이트웨이는 프라이빗 서브넷의 인스턴스에서 인터넷으로만 트래픽을 전달합니다.

Solution

NAT 게이트웨이가 작동하는지 테스트하려면 [퍼블릭 NAT 게이트웨이 테스트](#)를 참조하세요.

인스턴스에서 인터넷에 액세스할 수 없음

문제

퍼블릭 NAT 게이트웨이를 생성하고 테스트 단계를 수행했지만 ping 명령이 실패하거나 프라이빗 서브넷의 인스턴스가 인터넷에 액세스할 수 없습니다.

원인

이 문제의 원인은 다음 중 하나일 수 있습니다.

- NAT 게이트웨이가 트래픽을 제공할 준비가 되지 않았습니다.
- 라우팅 테이블이 올바르게 구성되지 않았습니다.
- 보안 그룹 또는 네트워크 ACL이 인바운드 또는 아웃바운드 트래픽을 차단하고 있습니다.
- 지원되지 않는 프로토콜을 사용하고 있습니다.

Solution

다음 정보를 확인하세요.

- NAT 게이트웨이가 Available 상태인지 확인합니다. Amazon VPC 콘솔에서 NAT Gateways 페이지로 이동하고 세부 정보 창에서 상태 정보를 봅니다. NAT 게이트웨이가 실패 상태인 경우 게이트웨이가 생성될 때 오류가 발생했을 수 있습니다. 자세한 내용은 [NAT 게이트웨이 생성 실패](#) 단원을 참조하세요.
- 라우팅 테이블을 올바르게 구성했는지 확인합니다:
 - NAT 게이트웨이는 인터넷 트래픽을 인터넷 게이트웨이로 라우팅하는 라우팅 테이블이 있는 퍼블릭 서브넷에 있어야 합니다.
 - 인스턴스는 인터넷 트래픽을 NAT 게이트웨이로 라우팅하는 라우팅 테이블이 있는 프라이빗 서브넷에 있어야 합니다.
 - 전체 또는 일부 인터넷 트래픽을 NAT 게이트웨이 대신 다른 디바이스로 라우팅하는 다른 라우팅 테이블 항목이 있는지 확인합니다.
- 프라이빗 인스턴스에 대한 보안 그룹 규칙이 아웃바운드 인터넷 트래픽을 허용하는지 확인합니다. ping 명령이 작동하려면 규칙이 아웃바운드 ICMP 트래픽도 허용해야 합니다.

NAT 게이트웨이 자체는 모든 아웃바운드 트래픽과 아웃바운드 요청에 대한 응답으로 받는 트래픽을 허용합니다(따라서 상태 저장).

- 프라이빗 서브넷 및 퍼블릭 서브넷과 연결된 네트워크 ACL에 인바운드 또는 아웃바운드 인터넷 트래픽을 차단하는 규칙이 없는지 확인합니다. ping 명령이 작동하려면 규칙이 인바운드 및 아웃바운드 ICMP 트래픽도 허용해야 합니다.

흐름 로그를 활성화하여 네트워크 ACL 또는 보안 그룹 규칙으로 인해 끊어진 연결을 진단할 수 있습니다. 자세한 내용은 [VPC 흐름 로그를 사용하여 IP 트래픽 로깅](#) 단원을 참조하세요.

- ping 명령을 사용하는 경우 ICMP가 활성화된 호스트를 ping하고 있는지 확인합니다. ICMP가 활성화되지 않은 경우 회신 패킷을 받지 못합니다. 이를 테스트하려면 사용자 자신의 컴퓨터의 명령줄 터미널에서 똑같은 ping 명령을 수행하세요.
- 인스턴스가 다른 리소스, 예를 들어 프라이빗 서브넷의 다른 인스턴스를 ping할 수 있는지 확인합니다(보안 그룹 규칙이 이 작업을 허용한다고 가정함).
- 연결이 TCP, UDP 또는 ICMP 프로토콜만 사용하는지 확인합니다.

대상에 대한 TCP 연결 실패

문제

프라이빗 서브넷의 인스턴스에서 NAT 게이트웨이를 통해 특정 대상에 연결할 때 일부 TCP 연결은 성공하지만 일부는 실패하거나 시간이 초과됩니다.

원인

이 문제의 원인은 다음 중 하나일 수 있습니다.

- 대상 엔드포인트가 조각난 TCP 패킷으로 응답하고 있습니다. NAT 게이트웨이는 TCP 또는 ICMP에 대한 IP 조각화를 지원하지 않습니다. 자세한 내용은 [NAT 게이트웨이 및 NAT 인스턴스 비교](#) 단원을 참조하세요.
- tcp_tw_recycle 옵션이 원격 서버에서 활성화되었으며, 이 옵션은 NAT 디바이스 뒤에 여러 연결이 있는 경우 문제를 일으키는 것으로 알려져 있습니다.

솔루션

다음을 수행하여 연결하려는 엔드포인트가 조각난 TCP 패킷으로 응답하는지 확인하세요.

1. 퍼블릭 IP 주소가 있는 퍼블릭 서브넷의 인스턴스를 사용하여 특정 엔드포인트로부터 조각화를 유발할 정도로 큰 응답을 트리거합니다.
2. tcpdump 유틸리티를 사용하여 엔드포인트가 조각화된 패킷을 전송하는지 확인합니다.

⚠ Important

이러한 확인을 수행하려면 퍼블릭 서브넷의 인스턴스를 사용해야 합니다. 원래 연결이 실패한 인스턴스, NAT 게이트웨이 뒤 프라이빗 서브넷의 인스턴스 또는 NAT 인스턴스는 사용할 수 없습니다.

대량 ICMP 패킷을 전송 또는 수신하는 진단 도구가 패킷 손실을 보고할 것입니다. 예를 들어 NAT 게이트웨이 뒤에서는 `ping -s 10000 example.com` 명령이 작동하지 않습니다.

3. 엔드포인트가 조각화된 TCP 패킷을 전송하는 경우 NAT 게이트웨이 대신 NAT 인스턴스를 사용할 수 있습니다.

원격 서버에 액세스할 수 있는 경우 다음을 수행하여 `tcp_tw_recycle` 옵션이 사용 가능한지 확인할 수 있습니다.

1. 서버에서 다음 명령을 실행합니다.

```
cat /proc/sys/net/ipv4/tcp_tw_recycle
```

1이 출력될 경우 `tcp_tw_recycle` 옵션이 활성화된 것입니다.

2. `tcp_tw_recycle`이 활성화된 경우 비활성화하는 것이 좋습니다. 연결을 재사용해야 하는 경우 `tcp_tw_reuse` 옵션을 사용하는 것이 더 안전합니다.

원격 서버에 액세스할 수 없는 경우 프라이빗 서브넷의 인스턴스에서 `tcp_timestamps` 옵션을 일시적으로 비활성화하여 테스트할 수 있습니다. 그런 다음 원격 서버에 다시 연결합니다. 연결에 성공하면 원격 서버에서 `tcp_tw_recycle`이 활성화된 것이 이전 오류의 원인일 수 있습니다. 가능하면 원격 서버 소유자에게 이 옵션이 활성화되어 있는지 확인하고 비활성화하도록 요청하세요.

경로 추적 출력에 NAT 게이트웨이 프라이빗 IP 주소가 표시되지 않음

문제

인스턴스가 인터넷에 액세스할 수 있지만, traceroute 명령을 수행할 때 출력에 NAT 게이트웨이의 프라이빗 IP 주소가 표시되지 않습니다.

원인

인스턴스가 인터넷 게이트웨이 등의 다른 게이트웨이를 사용하여 인터넷에 액세스하고 있습니다.

Solution

인스턴스가 위치하고 있는 서브넷의 라우팅 테이블에서 다음 정보를 확인합니다.

- 인터넷 트래픽을 NAT 게이트웨이로 보내는 경로가 있는지 확인합니다.
- 인터넷 트래픽을 가상 프라이빗 게이트웨이 또는 인터넷 게이트웨이와 같은 다른 디바이스로 보내는 보다 구체적인 경로가 없는지 확인합니다.

350초 후 인터넷 연결이 끊어짐

문제

인스턴스에서 인터넷에 액세스할 수 있지만 350초 후에 연결이 끊어집니다.

원인

NAT 게이트웨이를 사용하는 연결이 350초 이상 유희 상태인 경우 연결이 시간 초과됩니다.

연결 제한 시간이 초과하면 NAT 게이트웨이는 연결을 계속하려고 하는 NAT 게이트웨이 뒤의 리소스로 RST 패킷을 반환합니다(FIN 패킷을 보내지 않음).

Solution

연결이 끊어지지 않도록 하려면 연결을 통해 더 많은 트래픽을 시작합니다. 또는 인스턴스에서 350초 미만의 값으로 TCP keepalive를 활성화할 수 있습니다.

IPSec 연결을 설정할 수 없음

문제

대상에 대한 IPsec 연결을 설정할 수 없습니다.

원인

NAT 게이트웨이는 현재 IPSec 프로토콜을 지원하지 않습니다.

Solution

NAT-Traversal(NAT-T)을 사용하여 NAT 게이트웨이에 대해 지원되는 프로토콜인 UDP의 IPsec 트래픽을 캡슐화할 수 있습니다. NAT-T 및 IPsec 구성을 테스트하여 IPsec 트래픽이 삭제되지 않는지 확인하세요.

추가 연결을 시작할 수 없음

문제

대상에 대해 NAT 게이트웨이를 통한 기존 연결이 있지만 추가 연결을 설정할 수 없습니다.

원인

단일 NAT 게이트웨이에 대한 동시 연결 제한에 도달했을 수 있습니다. 자세한 내용은 [NAT 게이트웨이 기본 사항](#) 단원을 참조하세요. 프라이빗 서브넷의 인스턴스가 많은 수의 연결을 생성하는 경우 이 제한에 도달할 수 있습니다.

Solution

다음 중 하나를 수행하십시오.

- 가용 영역당 하나의 NAT 게이트웨이를 만들고 해당 영역에 클라이언트를 분산합니다.
- 퍼블릭 서브넷에서 추가 NAT 게이트웨이를 만들고 각각 다른 NAT 게이트웨이에 대한 경로가 있는 여러 프라이빗 서브넷으로 클라이언트를 분할합니다.
- 클라이언트가 대상에 대해 생성할 수 있는 연결 수를 제한합니다.
- CloudWatch의 [IdleTimeoutCount](#) 지표를 사용하여 유휴 접속의 증가를 모니터링합니다. 유휴 상태의 연결을 닫아서 용량을 확보합니다.
- 다양한 IP 주소를 이용하여 NAT 게이트웨이를 생성하거나 보조 IP 주소를 기존의 NAT 게이트웨이에 추가합니다. 각각의 신규 IPv4 주소에서는 최대 55,000개의 동시 연결이 지원됩니다. 자세한 내용은 [NAT 게이트웨이 만들기](#) 또는 [보조 IP 주소 연결 편집](#) 단원을 참조하세요.

요금

NAT 게이트웨이를 프로비저닝하면 NAT 게이트웨이를 사용할 수 있는 시간당 요금 및 처리하는 데이터 기가바이트당 요금이 부과됩니다 자세한 내용은 [Amazon VPC 요금](#)을 참조하세요.

다음 전략은 NAT 게이트웨이에 대한 데이터 전송 요금을 줄이는 데 도움이 될 수 있습니다.

- AWS 리소스가 가용 영역에서 상당한 양의 트래픽을 전송하거나 수신하는 경우 리소스가 NAT 게이트웨이와 동일한 가용 영역에 있는지 확인합니다. 또는 리소스가 있는 각 가용 영역에 NAT 게이트웨이를 생성합니다.

- NAT 게이트웨이를 통과하는 대부분의 트래픽이 인터페이스 엔드포인트 또는 게이트웨이 엔드포인트를 지원하는 AWS 서비스를 사용하는 경우 이러한 서비스에 대한 인터페이스 엔드포인트 또는 게이트웨이 엔드포인트를 만드는 것이 좋습니다. 잠재적인 비용 절감에 관한 자세한 내용은 [AWS PrivateLink 요금](#)을 참조하세요.

NAT 인스턴스

NAT 인스턴스는 Network Address Translation(NAT)을 제공합니다. NAT 인스턴스를 사용하면 프라이빗 서브넷의 리소스가 인터넷이나 온프레미스 네트워크와 같은 Virtual Private Cloud(VPC) 외부의 대상과 통신할 수 있습니다. 프라이빗 서브넷의 리소스는 인터넷으로 향하는 아웃바운드 IPv4 트래픽을 시작할 수 있지만 인터넷에서 시작된 인바운드 트래픽을 수신할 수는 없습니다.

Important

NAT AMI는 2020년 12월 31일에 표준 지원이 종료되고 2023년 12월 31일에 유지 관리 지원이 종료된 Amazon Linux AMI, 2018.03의 마지막 버전을 기반으로 구축되었습니다. 자세한 내용은 [Amazon Linux AMI 지원 종료](#) 블로그 게시물을 참조하세요.

기존 NAT AMI 사용하는 경우 AWS는 [NAT 게이트웨이로의 마이그레이션](#)을 권장합니다. NAT 게이트웨이는 더 나은 가용성과 향상된 대역폭을 제공하면서 관리 작업은 간소화합니다. 자세한 내용은 [NAT 게이트웨이 및 NAT 인스턴스 비교](#) 단원을 참조하십시오.

NAT 게이트웨이보다 NAT 인스턴스가 사용 사례와 더 잘 일치하는 경우 [the section called "NAT AMI 생성"](#)에 설명된 대로 Amazon Linux 현재 버전에서 자체 NAT AMI를 생성할 수 있습니다.

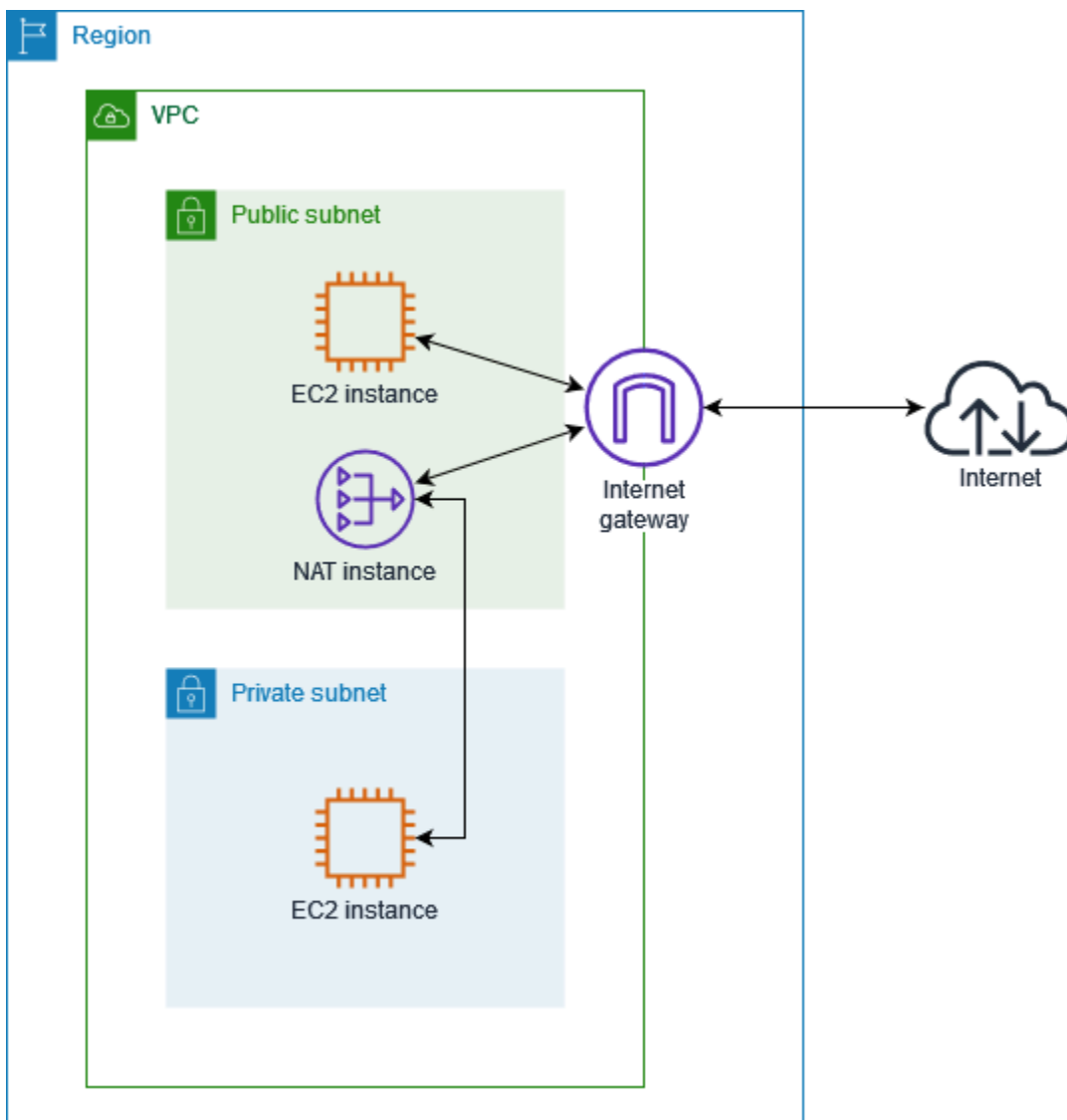
내용

- [NAT 인스턴스 기본 사항](#)
- [NAT 인스턴스의 VPC 생성](#)
- [NAT 인스턴스에 대한 보안 그룹 생성](#)
- [NAT AMI 생성](#)
- [NAT 인스턴스 시작](#)
- [원본/대상 확인 비활성화](#)
- [라우팅 테이블 업데이트](#)
- [NAT 인스턴스 테스트](#)

NAT 인스턴스 기본 사항

다음 그림에서는 NAT 인스턴스 기본 사항을 보여줍니다. 프라이빗 서브넷과 연결된 라우팅 테이블은 프라이빗 서브넷의 인스턴스에서 퍼블릭 서브넷의 NAT 인스턴스로 인터넷 트래픽을 전송합니다. 그러면 NAT 인스턴스는 인터넷 게이트웨이로 트래픽을 전송합니다. 트래픽은 NAT 인스턴스의 퍼블릭 IP 주소로 귀속됩니다. NAT 인스턴스는 응답에 대해 높은 포트 번호를 지정합니다. 즉, 응답이 되돌아 오면 NAT 인스턴스가 응답에 대한 포트 번호를 기준으로 프라이빗 서브넷에 있는 인스턴스로 이 응답을 보냅니다.

NAT 인스턴스는 인터넷에 액세스할 수 있어야 하므로, 퍼블릭 서브넷(인터넷 게이트웨이로 가는 경로가 있는 라우팅 테이블이 있는 서브넷)에 있어야 하며, NAT 인스턴스에는 퍼블릭 IP 주소 또는 탄력적 IP 주소가 있어야 합니다.



NAT 인스턴스를 시작하려면 NAT AMI를 생성하고 NAT 인스턴스용 보안 그룹을 생성한 다음 VPC로 NAT 인스턴스를 시작합니다.

NAT 인스턴스 할당량은 리전의 인스턴스 할당량에 따라 다릅니다. 자세한 내용은 AWS 일반 참조의 [Amazon EC2 서비스 할당량](#)을 참조하세요.

NAT 인스턴스의 VPC 생성

다음 절차를 따라 퍼블릭 서브넷 및 프라이빗 서브넷이 있는 VPC를 생성합니다.

VPC를 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. VPC 생성을 선택합니다.
3. Resources to create(생성할 리소스)에서 VPC and more(VPC 등)를 선택합니다.
4. Name tag auto-generation(이름 태그 자동 생성)에 VPC의 이름을 입력합니다.
5. 서브넷을 구성하려면 다음을 수행합니다.
 - a. Number of Availability Zones(가용 영역 수)에서 필요에 따라 1 또는 2를 선택합니다.
 - b. Number of public subnets(퍼블릭 서브넷 수)에서 가용 영역당 하나의 퍼블릭 서브넷이 있는지 확인합니다.
 - c. Number of private subnets(프라이빗 서브넷 수)에서 가용 영역당 하나의 프라이빗 서브넷이 있는지 확인합니다.
6. VPC 생성을 선택합니다.

NAT 인스턴스에 대한 보안 그룹 생성

다음 표에 설명된 규칙을 사용하여 보안 그룹을 생성합니다. 이 규칙을 사용하면 NAT 인스턴스가 프라이빗 서브넷에 있는 인스턴스로부터 오는 인터넷 트래픽뿐 아니라, 네트워크에서 오는 SSH 트래픽도 수신할 수 있습니다. 또한 NAT 인스턴스는 인터넷으로 트래픽을 전송할 수 있으며 따라서 프라이빗 서브넷의 인스턴스가 소프트웨어 업데이트를 받을 수 있습니다.

다음에는 권장 규칙이 나와 있습니다.

인바운드

소스	프로토콜	포트 범위	설명
#### ### CIDR	TCP	80	프라이빗 서브넷의 서버로부터의 인바운드 HTTP 트래픽 허용
#### ### CIDR	TCP	443	프라이빗 서브넷의 서버로부터의 인바운드 HTTPS 트래픽 허용
##### ### IP ## ##	TCP	22	네트워크로부터 NAT 인스턴스에 대한 인바운드 SSH 액세스 허용 (인터넷 게이트웨이를 통해)

아웃바운드

대상 주소	프로토콜	포트 범위	설명
0.0.0.0/0	TCP	80	인터넷에 대한 아웃바운드 HTTP 액세스 허용
0.0.0.0/0	TCP	443	인터넷에 대한 아웃바운드 HTTPS 액세스 허용

보안 그룹을 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹 생성을 선택합니다.
4. 보안 그룹의 이름과 설명을 입력합니다.
5. VPC인 경우 NAT 인스턴스에 대한 VPC ID를 선택합니다.
6. 다음과 같이 인바운드 규칙(Inbound Rules) 하에 인바운드 트래픽에 대한 규칙을 추가합니다.
 - a. 규칙 추가를 선택합니다. 유형으로 HTTP를 선택하고 소스(Source)에 대한 프라이빗 서브넷의 IP 주소 범위를 입력합니다.
 - b. 규칙 추가를 선택합니다. 유형으로 HTTPS를 선택하고 소스(Source)에 대한 프라이빗 서브넷의 IP 주소 범위를 입력합니다.

- c. 규칙 추가를 선택합니다. 유형으로 SSH를 선택하고 소스(Source)에 대한 네트워크의 IP 주소 범위를 입력합니다.
7. 다음과 같이 아웃바운드 규칙(Outbound Rules) 하에 아웃바운드 트래픽에 대한 규칙을 추가합니다.
 - a. 규칙 추가를 선택합니다. 유형으로 HTTP를 선택하고 대상에 0.0.0.0/0을 입력합니다.
 - b. 규칙 추가를 선택합니다. 유형으로 HTTPS를 선택하고 대상에 0.0.0.0/0을 입력합니다.
8. 보안 그룹 생성을 선택합니다.

자세한 내용은 [보안 그룹](#) 단원을 참조하십시오.

NAT AMI 생성

EC2 인스턴스에서 NAT를 실행하기 위해 NAT AMI가 구성됩니다. NAT AMI를 생성한 다음 NAT AMI를 사용하여 NAT 인스턴스를 시작해야 합니다.

NAT AMI에 Amazon Linux 이외의 운영 체제를 사용하려는 경우 해당 운영 체제의 설명서를 참조하여 NAT 구성 방법을 확인하십시오. 인스턴스를 재부팅한 후에도 이 설정이 유지되도록 설정을 저장해야 합니다.

Amazon Linux용 NAT AMI 생성

1. AL2023 또는 Amazon Linux 2를 실행하는 EC2 인스턴스를 시작합니다. NAT 인스턴스용으로 생성한 보안 그룹을 지정해야 합니다.
2. 인스턴스에 연결하고 인스턴스에서 다음 명령을 실행하여 iptables를 활성화합니다.

```
sudo yum install iptables-services -y
sudo systemctl enable iptables
sudo systemctl start iptables
```

3. 인스턴스에서 다음 작업을 수행하여 IP 전달을 활성화하고 IP 전달이 재부팅 후에도 계속 유지되도록 합니다.
 - a. nano 또는 vim 등의 텍스트 편집기를 사용하여 구성 파일(/etc/sysctl.d/custom-ip-forwarding.conf)을 생성합니다.
 - b. 구성 파일에 다음 줄을 추가합니다.

```
net.ipv4.ip_forward=1
```

- c. 구성 파일을 저장하고 텍스트 편집기를 종료합니다.
- d. 다음 명령을 실행하여 구성 파일을 적용합니다.

```
sudo sysctl -p /etc/sysctl.d/custom-ip-forwarding.conf
```

4. 인스턴스에서 다음 명령을 실행하고, 기본 네트워크 인터페이스의 이름을 기록해 둡니다. 다음 단계에서 이 정보가 필요합니다.

```
netstat -i
```

다음 예제 출력에서 `docker0`은 도커에 의해 생성된 네트워크 인터페이스이고, `eth0`은 기본 네트워크 인터페이스이고, `lo`는 루프백 인터페이스입니다.

Iface	MTU	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
docker0	1500	0	0	0	0	0	0	0	0	BMU
eth0	9001	7276052	0	0	0	5364991	0	0	0	BMRU
lo	65536	538857	0	0	0	538857	0	0	0	LRU

다음 예제 출력에서 기본 네트워크 인터페이스는 `enX0`입니다.

Iface	MTU	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
enX0	9001	1076	0	0	0	1247	0	0	0	BMRU
lo	65536	24	0	0	0	24	0	0	0	LRU

다음 예제 출력에서 기본 네트워크 인터페이스는 `ens5`입니다.

Iface	MTU	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
ens5	9001	14036	0	0	0	2116	0	0	0	BMRU
lo	65536	12	0	0	0	12	0	0	0	LRU

5. 인스턴스에서 다음 명령을 실행하여 NAT를 구성합니다. 기본 네트워크 인터페이스가 `eth0`이 아닌 경우 `eth0`을 이전 단계에서 기록한 기본 네트워크 인터페이스로 바꿉니다.

```
sudo /sbin/iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
sudo /sbin/iptables -F FORWARD
sudo service iptables save
```

6. EC2 인스턴스에서 NAT AMI를 생성합니다. 자세한 내용은 [Amazon EC2 사용 설명서](#)의 인스턴스에서 Linux AMI 생성을 참조하세요.

NAT 인스턴스 시작

다음 절차에 따라 생성한 VPC, 보안 그룹, NAT AMI를 사용하여 NAT 인스턴스를 시작합니다.

NAT 인스턴스를 시작하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 대시보드에서 인스턴스 시작을 선택합니다.
3. 이름(Name)에 NAT 인스턴스의 이름을 입력합니다.
4. 애플리케이션 및 OS 이미지에서 NAT AMI를 선택합니다(더 많은 AMI 찾아보기, 내 AMI 선택).
5. 인스턴스 유형에서 NAT 인스턴스에 필요한 컴퓨팅, 메모리, 스토리지 리소스를 제공하는 인스턴스 유형을 선택합니다.
6. 키 페어에서 기존 키 페어를 선택하거나 새 키 페어 생성을 선택합니다.
7. Network settings(네트워크 설정)에서 다음을 수행합니다.
 - a. 편집을 선택합니다.
 - b. [VPC]에 대해 생성된 VPC를 선택합니다.
 - c. 서브넷에서 생성한 퍼블릭 서브넷을 선택합니다.
 - d. Auto-assign Public IP(퍼블릭 IP 자동 할당)에서 Enable(활성화)을 선택합니다. 또는 NAT 인스턴스를 시작한 후 탄력적 IP 주소를 할당하고 이를 NAT 인스턴스에 할당합니다.
 - e. 방화벽에서 기존 보안 그룹 선택을 선택한 다음 생성한 보안 그룹을 선택합니다.
8. 인스턴스 시작을 선택합니다. 인스턴스 ID를 선택하여 인스턴스 세부 정보 페이지를 엽니다. 인스턴스 상태가 실행 중으로 변경되고 상태 확인이 성공할 때까지 기다립니다.
9. NAT 인스턴스의 소스/대상 확인을 비활성화합니다([원본/대상 확인 비활성화](#) 참조).
10. NAT 인스턴스로 트래픽을 보내기 위한 라우팅 테이블을 업데이트합니다([라우팅 테이블 업데이트](#) 참조).

원본/대상 확인 비활성화

각각의 EC2 인스턴스는 기본적으로 원본/대상 확인을 수행합니다. 이는 인스턴스가 보내거나 받는 트래픽의 원본 또는 대상이어야 한다는 의미입니다. 하지만, NAT 인스턴스는 원본 또는 대상이 그 자신이 아닐 때 트래픽을 보내고 받을 수 있어야 합니다. 따라서 NAT 인스턴스에서 원본/대상 확인을 비활성화해야 합니다.

소스/대상 확인을 비활성화하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 Instances(인스턴스)를 선택합니다.
3. NAT 인스턴스를 선택합니다.
4. 작업, 네트워킹, 소스/대상 확인 변경을 선택합니다.
5. 소스/대상 확인에서 중지를 선택합니다.
6. Save(저장)를 선택합니다.
7. NAT 인스턴스에 보조 네트워크 인터페이스가 있는 경우 [네트워크 인터페이스(Network interfaces)]에서 [네트워킹(Networking)] 탭을 선택합니다. 인터페이스 ID를 선택하여 네트워크 인터페이스 페이지로 이동합니다. [작업(Actions)], [소스/대상 변경. 확인(Change source/dest)]을 선택하고 [활성화(Enable)]을 지우고 [저장(Save)]을 선택합니다.

라우팅 테이블 업데이트

프라이빗 서브넷의 라우팅 테이블에는 NAT 인스턴스로 인터넷 트래픽을 보내는 경로가 있어야 합니다.

라우팅 테이블을 업데이트하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 라우팅 테이블을 선택합니다.
3. 프라이빗 서브넷에 대한 라우팅 테이블을 선택합니다.
4. 경로(Routes) 탭에서 경로 편집(Edit routes) 및 경로 추가(Add route)를 차례로 선택합니다.
5. 대상(Destination)에 0.0.0.0/0을, 타겟(Target)에 NAT 인스턴스의 인스턴스 ID를 입력합니다.
6. Save changes(변경 사항 저장)를 선택합니다.

자세한 내용은 [라우팅 테이블 구성](#) 단원을 참조하십시오.

NAT 인스턴스 테스트

NAT 인스턴스를 시작하고 위의 구성 단계를 완료한 후, NAT 인스턴스를 bastion 서버로 사용하여 프라이빗 서브넷의 인스턴스가 NAT 인스턴스를 통해 인터넷에 액세스할 수 있는지 여부를 테스트할 수 있습니다.

Tasks

- [1단계: NAT 인스턴스 보안 그룹 업데이트](#)
- [2단계: 프라이빗 서브넷에서 테스트 인스턴스 시작](#)
- [3단계: ICMP 지원 웹사이트 ping](#)
- [4단계: 정리](#)

1단계: NAT 인스턴스 보안 그룹 업데이트

프라이빗 서브넷의 인스턴스가 ping 트래픽을 NAT 인스턴스로 전송하도록 허용하려면 인바운드 및 아웃바운드 ICMP 트래픽을 허용하는 규칙을 추가합니다. NAT 인스턴스가 Bastion 서버 역할을 하도록 허용하려면 프라이빗 서브넷으로의 아웃바운드 SSH 트래픽을 허용하는 규칙을 추가합니다.

NAT 인스턴스의 보안 그룹 업데이트

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. NAT 인스턴스와 연결된 보안 그룹의 확인란을 선택합니다.
4. [인바운드 규칙(Inbound rules)] 탭에서 [인바운드 규칙 편집(Edit inbound rules)]을 선택합니다.
5. [Add another rule]을 선택합니다. [유형(Type)]에서 [모든 ICMP - IPv4(All ICMP - IPv4)]를 선택합니다. 소스(Source)에서 사용자 정의(Custom)를 선택하고 프라이빗 서브넷의 IP 주소 범위를 입력합니다. 규칙 저장을 선택합니다.
6. 아웃바운드 규칙(Outbound rules) 탭에서 아웃바운드 규칙 편집(Edit outbound rules)을 선택합니다.
7. [Add another rule]을 선택합니다. [유형(Type)]에서 SSH를 선택합니다. Destination(대상)에서 사용자 정의(Custom)를 선택하고 프라이빗 서브넷의 IP 주소 범위를 입력합니다.
8. [Add another rule]을 선택합니다. [유형(Type)]에서 [모든 ICMP - IPv4(All ICMP - IPv4)]를 선택합니다. 대상(Destination)에 대해 어디서나 - IPv4(Anywhere - IPv4)를 선택합니다. 규칙 저장을 선택합니다.

2단계: 프라이빗 서브넷에서 테스트 인스턴스 시작

프라이빗 서브넷으로 인스턴스를 시작합니다. NAT 인스턴스에서 SSH 액세스를 허용해야 하며 NAT 인스턴스에 사용한 것과 동일한 키 쌍을 사용해야 합니다.

프라이빗 서브넷에서 테스트 인스턴스 시작

1. <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다.

2. 대시보드에서 인스턴스 시작을 선택합니다.
3. 프라이빗 서브넷을 선택합니다.
4. 이 인스턴스에는 퍼블릭 IP 주소를 할당하지 마세요.
5. 이 인스턴스의 보안 그룹은 NAT 인스턴스 또는 퍼블릭 서브넷의 IP 주소 범위에서 오는 인바운드 SSH 액세스와 아웃바운드 ICMP 트래픽을 허용해야 합니다.
6. NAT 인스턴스에 사용한 것과 동일한 키 페어를 선택합니다.

3단계: ICMP 지원 웹사이트 ping

프라이빗 서브넷의 테스트 인스턴스가 NAT 인스턴스를 사용하여 인터넷과 통신할 수 있는지 확인하려면 ping 명령을 실행합니다.

프라이빗 인스턴스에서 인터넷 연결을 테스트하려면

1. 로컬 컴퓨터에서 SSH 에이전트 전달을 구성하여 NAT 인스턴스를 Bastion 서버로 사용할 수 있습니다.

Linux and macOS

```
ssh-add key.pem
```

Windows

아직 설치되지 않은 경우 [Pageant를 다운로드하여 설치합니다.](#)

PuTTYgen을 사용하여 [프라이빗 키를 .ppk 형식으로 변환합니다.](#)

Pageant를 시작하고 작업 표시줄에서 Pageant 아이콘(숨겨져 있을 수 있음)을 마우스 오른쪽 버튼으로 클릭한 다음 키 추가를 선택합니다. 생성한 .ppk 파일을 선택하고 필요한 경우 암호를 입력한 다음 열기를 선택합니다.

2. 로컬 컴퓨터에서 NAT 인스턴스에 연결합니다.

Linux and macOS

```
ssh -A ec2-user@nat-instance-public-ip-address
```

Windows

PuTTY를 사용하여 NAT 인스턴스에 연결합니다. 인증의 경우 에이전트 전달 허용을 선택하고 인증을 위한 프라이빗 키 파일을 비워 두어야 합니다.

3. NAT 인스턴스에서 ping 명령을 실행하여 ICMP에 대해 활성화된 웹 사이트를 지정합니다.

```
[ec2-user@ip-10-0-4-184]$ ping ietf.org
```

NAT 인스턴스에 인터넷 액세스 권한이 있는지 확인하려면 다음과 같은 출력을 수신했는지 확인한 다음 Ctrl+C를 눌러 ping 명령을 취소합니다. 아니면 NAT 인스턴스가 퍼블릭 서브넷에 있는지 확인합니다(라우팅 테이블에 인터넷 게이트웨이에 대한 경로가 있음).

```
PING ietf.org (104.16.45.99) 56(84) bytes of data.
64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=1 ttl=33 time=7.88 ms
64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=2 ttl=33 time=8.09 ms
64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=3 ttl=33 time=7.97 ms
...
```

4. NAT 인스턴스에서 프라이빗 IP 주소를 사용하여 프라이빗 서브넷의 인스턴스에 연결합니다.

```
[ec2-user@ip-10-0-4-184]$ ssh ec2-user@private-server-private-ip-address
```

5. 프라이빗 인스턴스에서 ping 명령을 실행하여 인터넷에 연결할 수 있는지 테스트합니다.

```
[ec2-user@ip-10-0-135-25]$ ping ietf.org
```

프라이빗 인스턴스가 NAT 인스턴스를 통해 인터넷에 액세스할 수 있는지 확인하려면 다음과 같은 출력을 수신했는지 확인한 다음 Ctrl+C를 눌러 ping 명령을 취소합니다.

```
PING ietf.org (104.16.45.99) 56(84) bytes of data.
64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=1 ttl=33 time=8.76 ms
64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=2 ttl=33 time=8.26 ms
64 bytes from 104.16.45.99 (104.16.45.99): icmp_seq=3 ttl=33 time=8.27 ms
...
```

문제 해결

프라이빗 서브넷의 서버에서 ping 명령이 실패하는 경우 다음 단계에 따라 문제를 해결하십시오.

- ICMP가 활성화된 웹 사이트를 ping했는지 확인합니다. 수행되지 않았으면 서버가 응답 패킷을 수신할 수 없습니다. 이를 테스트하려면 사용자 컴퓨터의 명령줄 터미널에서 똑같은 ping 명령을 수행하세요.
- NAT 인스턴스의 보안 그룹이 프라이빗 서브넷에서 오는 인바운드 ICMP 트래픽을 허용하는지 확인합니다. 허용하지 않으면, NAT 인스턴스가 프라이빗 인스턴스로부터 ping 명령을 수신할 수 없습니다.
- NAT 인스턴스에 대해 소스/대상 확인을 비활성화했는지 확인합니다. 자세한 내용은 [원본/대상 확인 비활성화](#) 단원을 참조하십시오.
- 라우팅 테이블을 올바르게 구성했는지 확인합니다. 자세한 내용은 [라우팅 테이블 업데이트](#) 단원을 참조하십시오.

4단계: 정리

프라이빗 서브넷에 테스트 서버가 더 이상 필요하지 않은 경우 더 이상 요금이 청구되지 않도록 인스턴스를 종료합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [인스턴스 종료](#)를 참조하세요.

NAT 인스턴스가 더 이상 필요하지 않은 경우 요금이 더 이상 청구되지 않도록 중지하거나 종료할 수 있습니다. NAT AMI를 생성한 경우 필요할 때마다 새 NAT 인스턴스를 생성할 수 있습니다.

NAT 게이트웨이 및 NAT 인스턴스 비교

NAT 인스턴스와 NAT 게이트웨이의 차이점을 세부적으로 요약하면 다음과 같습니다. NAT 게이트웨이는 더 나은 가용성과 대역폭을 제공하고 관리에 소요되는 작업이 줄어들기 때문에 권장합니다.

속성	NAT 게이트웨이	NAT 인스턴스
가용성	고가용성. 각 가용 영역의 NAT 게이트웨이는 중복적으로 구현됩니다. 각 가용 영역에 하나의 NAT 게이트웨이를 만들어 아키텍처가 영역에 종속되지 않도록 합니다.	스크립트를 사용하여 인스턴스 간의 장애 조치를 관리합니다.
대역폭	최대 100Gbps까지 확장합니다.	인스턴스 유형의 대역폭에 따라 다릅니다.
유지 관리	AWS에서 관리합니다. 유지 관리 작업을 수행할 필요가 없습니다.	사용자가 관리합니다(예: 인스턴스에 소프트웨어 업데이트 또는 운영 체제 패치 설치).

속성	NAT 게이트웨이	NAT 인스턴스
성능	소프트웨어가 NAT 트래픽 처리에 최적화되어 있습니다.	NAT를 수행하도록 구성된 일반 AMI입니다.
비용	사용하는 NAT 게이트웨이 수, 사용 기간, NAT 게이트웨이를 통해 보내는 데이터의 양에 따라 요금이 청구됩니다.	사용하는 NAT 인스턴스 수, 사용 기간, 인스턴스 유형과 크기에 따라 요금이 청구됩니다.
유형 및 크기	균일하게 제공되므로, 유형 또는 크기를 결정할 필요가 없습니다.	예상 워크로드에 따라 적합한 인스턴스 유형과 크기를 선택합니다.
퍼블릭 IP 주소	생성할 때 퍼블릭 NAT 게이트웨이와 연결할 탄력적 IP 주소를 선택합니다.	탄력적 IP 주소 또는 퍼블릭 IP 주소를 NAT 인스턴스와 함께 사용합니다. 새 탄력적 IP 주소를 인스턴스와 연결하여 언제든지 퍼블릭 IP 주소를 변경할 수 있습니다.
프라이빗 IP 주소	게이트웨이를 만들 때 서브넷의 IP 주소 범위에서 자동으로 선택됩니다.	인스턴스를 시작할 때 서브넷의 IP 주소 범위에서 특정 프라이빗 IP 주소를 할당합니다.
보안 그룹	보안 그룹을 NAT 게이트웨이와 연결할 수 없습니다. 보안 그룹을 NAT 게이트웨이 기반 리소스와 연결하여 인바운드 및 아웃바운드 트래픽을 제어할 수 있습니다.	NAT 인스턴스 뒤의 리소스 및 NAT 인스턴스와 연결하여 인바운드 및 아웃바운드 트래픽을 제어합니다.
네트워크 ACL	네트워크 ACL을 사용하여 NAT 게이트웨이가 위치하고 있는 서브넷에서 보내고 받는 트래픽을 제어합니다.	네트워크 ACL을 사용하여 NAT 인스턴스가 위치하고 있는 서브넷에서 보내고 받는 트래픽을 제어합니다.
흐름 로그	흐름 로그를 사용하여 트래픽을 캡처합니다.	흐름 로그를 사용하여 트래픽을 캡처합니다.
포트 전달	지원하지 않음.	포트 전달을 지원하려면 구성을 수동으로 사용자 지정합니다.
Bastion 서버	지원하지 않음.	Bastion 서버로 사용합니다.

속성	NAT 게이트웨이	NAT 인스턴스
트래픽 지표	NAT 게이트웨이에 대한 CloudWatch 지표를 확인합니다.	인스턴스에 대한 CloudWatch 지표를 확인합니다.
제한 시간 초과 동작	연결 제한 시간이 초과하면 NAT 게이트웨이는 연결을 계속하려고 하는 NAT 게이트웨이 뒤의 리소스로 RST 패킷을 반환합니다(FIN 패킷을 보내지 않음).	연결 제한 시간이 초과하면 NAT 인스턴스는 NAT 인스턴스 뒤의 리소스로 FIN 패킷을 전송하여 연결을 닫습니다.
IP 조각화	UDP 프로토콜에서 IP 조각화된 패킷의 전달을 지원합니다. TCP 및 ICMP 프로토콜에 대해서는 조각화를 지원하지 않습니다. 이러한 프로토콜의 조각화된 패킷은 삭제됩니다.	UDP, TCP 및 ICMP 프로토콜에 대해 IP 조각화된 패킷의 재수집을 지원합니다.

NAT 인스턴스에서 NAT 게이트웨이로 마이그레이션

이미 NAT 인스턴스를 사용하는 경우 이를 NAT 게이트웨이로 대체하는 것이 좋습니다. NAT 인스턴스와 동일한 서브넷에 NAT 게이트웨이를 만든 다음, NAT 인스턴스를 가리키는 라우팅 테이블의 기존 경로를 NAT 게이트웨이를 가리키는 경로로 대체할 수 있습니다. 현재 NAT 인스턴스에 사용하는 것과 동일한 탄력적 IP 주소를 NAT 게이트웨이에 사용하려는 경우에도 먼저 NAT 인스턴스의 탄력적 IP 주소를 연결 해제하고 NAT 게이트웨이를 만들 때 이 주소를 게이트웨이에 연결해야 합니다.

NAT 인스턴스에서 NAT 게이트웨이로 라우팅을 변경하거나 NAT 인스턴스에서 탄력적 IP 주소의 연결을 해제하면 현재 연결이 끊어지고 연결을 다시 설정해야 합니다. 중요한 작업(또는 NAT 인스턴스를 통해 작동하는 기타 작업)이 실행 중이지 않은지 확인합니다.

탄력적 IP 주소를 VPC의 리소스와 연결

탄력적 IP 주소는 동적 클라우드 컴퓨팅을 위해 고안된 고정 퍼블릭 IPv4 주소입니다. 계정의 모든 VPC에 있는 모든 인스턴스 또는 네트워크 인터페이스에 탄력적 IP 주소를 연결할 수 있습니다. 탄력적 IP 주소로 VPC의 다른 인스턴스에 주소를 신속하게 다시 매핑하여 인스턴스의 오류를 숨길 수 있습니다.

탄력적 IP 주소 개념 및 규칙

탄력적 IP 주소를 사용하려면 먼저 계정에서 사용할 수 있도록 할당합니다. 그런 다음, VPC의 인스턴스 또는 네트워크 인터페이스와 연결할 수 있습니다. 탄력적 IP 주소는 명시적으로 릴리스될 때까지 AWS 계정에 할당되어 있습니다.

탄력적 IP 주소는 네트워크 인터페이스의 속성입니다. 인스턴스에 연결된 네트워크 인터페이스를 업데이트하여 탄력적 IP 주소를 인스턴스와 연결할 수 있습니다. 탄력적 IP 주소를 인스턴스에 직접 연결하는 대신에 네트워크 인터페이스와 연결할 경우, 네트워크 인터페이스의 모든 속성들을 한 인스턴스에서 다른 인스턴스로 한 번에 옮길 수 있는 장점이 있습니다. 자세한 내용은 Amazon EC2 사용 설명서의 [탄력적 네트워크 인터페이스](#)를 참조하세요.

다음 규칙이 적용됩니다.

- 탄력적 IP 주소는 한 번에 단일 인스턴스 또는 네트워크 인터페이스와 연결할 수 있습니다.
- 인스턴스 또는 네트워크 인터페이스 간에 탄력적 IP 주소를 이동할 수 있습니다.
- 탄력적 IP 주소를 인스턴스의 eth0 네트워크 인터페이스와 연결하면, 현재 퍼블릭 IPv4 주소(있는 경우)는 EC2-VPC 퍼블릭 IP 주소 풀로 연결 해제됩니다. 탄력적 IP 주소의 연결을 해제하면 몇 분 내에 자동으로 eth0 네트워크 인터페이스에 새 퍼블릭 IPv4 주소가 배정됩니다. 인스턴스에 보조 네트워크 인터페이스를 연결한 경우에는 이 동작이 적용되지 않습니다.
- 탄력적 IP 주소는 5개로 제한됩니다. 주소를 절약하기 위해 NAT 디바이스를 사용할 수 있습니다. 자세한 내용은 [NAT 디바이스를 사용하여 인터넷 또는 다른 네트워크에 연결](#) 단원을 참조하십시오.
- IPv6에 대한 탄력적 IP 주소는 지원되지 않습니다.
- VPC에 사용하도록 할당된 탄력적 IP 주소에 태그를 지정할 수 있지만, 비용 할당 태그가 지원되지 않습니다. 탄력적 IP 주소를 복구하는 경우, 태그가 복구되지 않습니다.
- 보안 그룹 및 네트워크 ACL이 원본 IP 주소의 트래픽을 허용하면 인터넷에서 탄력적 IP 주소에 액세스할 수 있습니다. VPC 내에서 인터넷으로 다시 돌아가는 응답 트래픽에는 인터넷 게이트웨이가 필요합니다. 자세한 내용은 [보안 그룹](#) 및 [네트워크 ACL](#) 단원을 참조하세요.
- 탄력적 IP 주소에 대해 다음 옵션 중 하나를 사용할 수 있습니다.
 - Amazon이 탄력적 IP 주소를 제공하도록 합니다. 이 옵션을 선택하면 탄력적 IP 주소를 네트워크 경계 그룹과 연결할 수 있습니다. 이는 CIDR 블록을 공고하는 위치입니다. 네트워크 경계 그룹을 설정하면 CIDR 블록이 이 그룹으로 제한됩니다.
 - 고유 IP 주소를 사용합니다. 고유 IP 주소 가져오기에 대한 자세한 내용은 Amazon EC2 사용 설명서의 [고유 IP 주소 가져오기\(BYOIP\)](#)를 참조하세요.

탄력적 IP 주소는 리전별입니다. Global Accelerator를 사용하여 글로벌 IP 주소를 프로비저닝하는 것에 대한 자세한 내용은 AWS Global Accelerator 개발자 안내서의 [지역의 정적 IP 주소 대신 글로벌 정적 IP 주소 사용](#)을 참조하세요.

탄력적 IP 주소 작업

다음 섹션에서는 탄력적 IP 주소를 이용한 작업 방법에 대해 살펴보겠습니다.

작업

- [탄력적 IP 주소 할당](#)
- [탄력적 IP 주소 연결](#)
- [탄력적 IP 주소 보기](#)
- [탄력적 IP 주소 태그](#)
- [탄력적 IP 주소 연결 해제](#)
- [탄력적 IP 주소 전송](#)
- [탄력적 IP 주소 릴리스](#)
- [탄력적 IP 주소 복구](#)
- [API 및 명령 개요](#)

탄력적 IP 주소 할당

탄력적 IP를 사용하기 전에 VPC에서 사용할 탄력적 IP를 할당해야 합니다.

탄력적 IP 주소 할당

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Elastic IPs]를 선택합니다.
3. 탄력적 IP 주소 할당을 선택합니다.
4. (선택 사항) 탄력적 IP 주소(EIP)를 할당할 때는 EIP를 할당할 네트워크 경계 그룹을 선택합니다. 네트워크 경계 그룹은 AWS가 퍼블릭 IP 주소를 알리는 가용 영역, Local Zone 또는 Wavelength Zone의 집합입니다. Local Zone 및 Wavelength Zone은 AWS 네트워크와 해당 영역의 리소스에 액세스하는 고객 간의 지연 시간 또는 물리적 거리를 최소화하기 위해 리전의 AZ와 다른 네트워크 경계 그룹을 가질 수 있습니다.

⚠ Important

EIP와 연결될 AWS 리소스와 동일한 네트워크 경계 그룹에 EIP를 할당해야 합니다. 한 네트워크 경계 그룹의 EIP는 해당 네트워크 경계 그룹의 영역에서만 광고될 수 있으며 다른 네트워크 경계 그룹이 나타내는 다른 영역에서는 광고될 수 없습니다.

Local Zone 또는 Wavelength Zone을 활성화한 경우(자세한 내용은 [Local Zone 활성화](#) 또는 [Wavelength Zone 활성화](#) 참조) AZ, Local Zone 또는 Wavelength Zone에 대한 네트워크 경계 그룹을 선택할 수 있습니다. EIP와 연결된 AWS 리소스는 동일한 네트워크 경계 그룹에 있어야 하므로 네트워크 경계 그룹을 신중하게 선택하세요. EC2 콘솔을 사용하여 가용 영역, Local Zone 또는 Wavelength Zone이 속한 네트워크 경계 그룹을 볼 수 있습니다([Local Zones](#) 참조). 일반적으로 리전의 모든 가용 영역은 동일한 네트워크 경계 그룹에 속하지만 Local Zone 또는 Wavelength Zone은 별도의 자체 네트워크 경계 그룹에 속합니다.

Local Zone 또는 Wavelength Zone이 활성화되지 않은 경우 EIP를 할당하면 해당 리전의 모든 AZ를 나타내는 네트워크 경계 그룹(예:us-west-2)이 미리 정의되어 있으며 변경할 수 없습니다. 즉, 이 네트워크 경계 그룹에 할당한 EIP는 현재 속한 리전의 모든 AZ에 광고됩니다.

5. 퍼블릭 IPv4 주소 풀(Public IPv4 address pool)에서 다음 중 하나를 선택합니다.

- Amazon의 IP 주소 풀 — IPv4 주소를 Amazon의 IP 주소 풀에서 할당하려는 경우.
- [내 퍼블릭 IPv4 주소 풀(My pool of public IPv4 addresses)] - AWS 계정으로 가져온 IP 주소 풀에서 IPv4 주소를 할당하려는 경우. IP 주소 풀이 없는 경우에는 이 옵션을 사용할 수 없습니다.
- Customer owned pool of IPv4 addresses(고객 소유 IPv4 주소 풀)—Outpost에서 사용하기 위해 온프레미스 네트워크에서 만든 풀에서 IPv4 주소를 할당하려는 경우. Outpost가 없는 경우 이 옵션만 사용할 수 있습니다.

6. (선택) 태그를 추가하거나 제거할 수 있습니다.

[태그 추가] 새 태그 추가를 선택하고 다음을 수행합니다.

- 키에서 키 이름을 입력합니다.
- 값에 키 값을 입력합니다.

[태그 제거] 태그의 키와 값 오른쪽에 있는 제거를 선택합니다.

7. [Allocate]를 선택합니다.

탄력적 IP 주소 연결

VPC에서 실행 중인 인스턴스 또는 네트워크 인터페이스에 탄력적 IP를 연결할 수 있습니다.

탄력적 IP 주소를 인스턴스와 연결하면 DNS 호스트 이름이 활성화된 경우 인스턴스가 퍼블릭 DNS 호스트 이름을 받습니다. 자세한 내용은 [VPC의 DNS 속성](#) 단원을 참조하십시오.

인스턴스 또는 네트워크 인터페이스에 탄력적 IP 주소를 연결하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Elastic IPs]를 선택합니다.
3. VPC에서 사용하기 위해 할당한 탄력적 IP 주소를 선택한 후(범위(Scope) 열에 vpc 값이 표시됨), 작업(Actions), 주소 연결(Associate Address)을 선택합니다.
4. [Instance] 또는 [Network interface]를 선택한 다음 인스턴스 또는 네트워크 인터페이스 ID를 선택합니다. 탄력적 IP 주소를 연결할 프라이빗 IP 주소를 선택합니다. 연결을 선택합니다.

탄력적 IP 주소 보기

계정에 할당된 탄력적 IP 주소를 볼 수 있습니다.

탄력적 IP 주소를 보려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Elastic IPs]를 선택합니다.
3. 표시되는 목록을 필터링하려면 검색 상자에 탄력적 IP 주소의 일부 또는 속성 중 하나를 입력하기 시작합니다.

탄력적 IP 주소 태그

탄력적 IP 주소에 태그를 적용하면 조직의 요구에 따라 이를 식별 또는 분류할 수 있습니다.

탄력적 IP 주소를 태그하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Elastic IPs]를 선택합니다.
3. 탄력적 IP 주소를 선택하고 [Tags]를 선택합니다.
4. 태그 추가/편집(Add/Edit Tags)을 선택하고, 필요에 따라 태그 키 및 값을 입력하고, 저장(Save)을 선택합니다.

탄력적 IP 주소 연결 해제

탄력적 IP 주소가 연결된 리소스를 변경하려면 먼저 현재 연결된 리소스에서 해당 주소를 연결 해제해야 합니다.

엘라스틱 IP 주소를 연결 해제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Elastic IPs]를 선택합니다.
3. 탄력적 IP 주소를 선택한 다음 작업(Actions), 탄력적 IP 주소 연결 해제(Disassociate Elastic IP address)를 선택합니다.
4. 메시지가 나타나면 주소 연결 해제(Disassociate address)를 선택합니다.

탄력적 IP 주소 전송

이 섹션에서는 하나의 AWS 계정에서 다른 계정으로 탄력적 IP 주소를 전송하는 방법을 설명합니다. 탄력적 IP 주소 전송은 다음 상황에서 유용할 수 있습니다.

- 조직 구조 조정 - 탄력적 IP 주소 전송을 사용하여 워크로드를 하나의 AWS 계정에서 다른 계정으로 빠르게 이동합니다. 새 탄력적 IP 주소가 보안 그룹 및 NACL의 허용 목록에 추가될 때까지 기다리지 않아도 됩니다.
- 중앙 집중식 보안 관리 - 중앙 집중식 AWS 보안 계정을 사용하여 보안 규정 준수를 위해 검증된 탄력적 IP 주소를 추적하고 전송합니다.
- 재해 복구 - 탄력적 IP 주소 전송을 사용하여 긴급 상황 동안 공용 인터넷 워크로드의 IP를 신속하게 다시 매핑합니다.

탄력적 IP 주소 전송에는 요금이 부과되지 않습니다.

Tasks

- [탄력적 IP 주소 전송 활성화](#)
- [탄력적 IP 주소 전송 비활성화](#)
- [전송된 탄력적 IP 주소 수락](#)

탄력적 IP 주소 전송 활성화

이 섹션에서는 전송된 탄력적 IP 주소를 수락하는 방법을 설명합니다. 전송을 위한 탄력적 IP 주소 활성화와 관련해 다음과 같은 제한 사항에 유의하세요.

- 모든 AWS 계정(소스 계정)의 탄력적 IP 주소를 동일한 AWS 리전의 다른 AWS 계정(전송 계정)으로 전송할 수 있습니다.
- 탄력적 IP 주소를 전송할 때 AWS 계정 간에 2단계 핸드셰이크가 발생합니다. 소스 계정에서 전송이 시작되면 전송 계정은 7일 내에 탄력적 IP 주소 전송을 수락해야 합니다. 이 7일 동안 소스 계정은 대기 중인 전송을 볼 수 있습니다(예: AWS 콘솔에서 보거나 [describe-address-transfers](#) AWS CLI 명령 사용). 7일이 지나면 전송이 만료되고 탄력적 IP 주소의 소유권이 소스 계정으로 반환됩니다.
- 수락된 전송은 전송이 수락된 후 3일 동안 소스 계정에서 볼 수 있습니다(예: AWS 콘솔에서 보거나 [describe-address-transfers](#) AWS CLI 명령 사용).
- AWS에서는 보류 중인 탄력적 IP 주소 전송 요청에 대해 전송 계정에 알리지 않습니다. 소스 계정의 소유자는 반드시 수락해야 하는 탄력적 IP 주소 전송 요청이 있음을 전송 계정 소유자에게 알려야 합니다.
- 전송되는 탄력적 IP 주소와 연결된 모든 태그는 전송이 완료된 후에 재설정됩니다.
- AWS 계정에 가져온 퍼블릭 IPv4 주소 풀(일반적으로 고유 IP 주소 가져오기(BYOIP) 주소 풀이라고 함)에서 할당된 탄력적 IP 주소는 전송할 수 없습니다.
- 역방향 DNS 레코드가 연결되어 있는 탄력적 IP 주소를 전송하려고 할 경우 전송 프로세스를 시작할 수 있지만, 연결된 DNS 레코드가 제거될 때까지 전송 계정이 전송을 수락할 수 없습니다.
- AWS Outposts를 활성화하고 구성한 경우 고객 소유의 IP 주소 풀(COIP)에서 탄력적 IP 주소를 할당할 수 있습니다. CoIP에서 할당된 탄력적 IP 주소는 전송할 수 없습니다. 하지만 AWS RAM을 사용하여 다른 계정과 CoIP를 공유할 수 있습니다. 자세한 내용은 AWS Outposts 사용 설명서에서 [고객 소유 IP 주소](#)를 참조하세요.
- Amazon VPC IPAM을 사용하여 AWS Organizations의 조직 내 계정으로 탄력적 IP 주소 전송을 추적할 수 있습니다. 자세한 내용은 [IP 주소 기록 보기](#)를 참조하세요. 그러나 탄력적 IP 주소가 조직 외부의 AWS 계정으로 전송되는 경우 탄력적 IP 주소에 대한 IPAM 감사 기록은 손실됩니다.

소스 계정으로 이 단계를 수행해야 합니다.

탄력적 IP 주소 전송 활성화

1. 소스 AWS 계정을 사용 중인지 확인하세요.
2. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
3. 탐색 창에서 탄력적 IP(Elastic IPs)를 선택합니다.

4. 전송을 활성화할 탄력적 IP 주소를 하나 이상 선택하고 Actions(작업), Enable transfer(전송 활성화)를 선택합니다.
5. 탄력적 IP 주소를 여러 개 전송하는 경우 Transfer type(전송 유형) 옵션이 표시됩니다. 다음 옵션 중 하나를 선택합니다:
 - 탄력적 IP 주소를 단일 AWS 계정으로 전송하려는 경우 Single account(단일 계정)를 선택합니다.
 - 탄력적 IP 주소를 여러 AWS 계정으로 전송하려는 경우 Multiple accounts(다중 계정)를 선택합니다.
6. Transfer account ID(전송 계정 ID)에 탄력적 IP 주소를 전송하려는 AWS 계정의 ID를 입력합니다.
7. 텍스트 상자에 **enable**을 입력하여 전송을 확인합니다.
8. 제출을 선택합니다.
9. 전송을 수락하려면 [전송된 탄력적 IP 주소 수락](#) 섹션을 참조하세요. 전송을 비활성화하려면 [탄력적 IP 주소 전송 비활성화](#) 섹션을 참조하세요.

탄력적 IP 주소 전송 비활성화

이 섹션에서는 탄력적 IP 전송을 활성화한 후 이를 비활성화하는 방법을 설명합니다.

전송을 활성화한 소스 계정으로 다음 단계를 수행해야 합니다.

탄력적 IP 주소 전송 비활성화

1. 소스 AWS 계정을 사용 중인지 확인하세요.
2. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
3. 탐색 창에서 탄력적 IP(Elastic IPs)를 선택합니다.
4. 탄력적 IP의 리소스 목록에서 Transfer status(전송 상태) 열을 표시하는 속성이 활성화되어 있는지 확인합니다.
5. Transfer status(전송 상태)가 Pending(보류 중)인 탄력적 IP 주소를 하나 이상 선택하고 Actions(작업), Disable transfer(전송 비활성화)를 선택합니다.
6. 텍스트 상자에 **disable**을 입력하여 확인합니다.
7. 제출을 선택합니다.

전송된 탄력적 IP 주소 수락

이 섹션에서는 전송된 탄력적 IP 주소를 수락하는 방법을 설명합니다.

탄력적 IP 주소를 전송할 때 AWS 계정 간에 2단계 핸드셰이크가 발생합니다. 소스 계정에서 전송이 시작되면 전송 계정은 7일 내에 탄력적 IP 주소 전송을 수락해야 합니다. 이 7일 동안 소스 계정은 대기 중인 전송을 볼 수 있습니다(예: AWS 콘솔에서 보거나 [describe-address-transfers](#) AWS CLI 명령 사용). 7일이 지나면 전송이 완료되고 탄력적 IP 주소의 소유권이 소스 계정으로 반환됩니다.

전송을 수락할 때 발생할 수 있는 다음 예외 사항과 이에 대한 해결 방법을 참고하세요.

- **AddressLimitExceed**: 전송 계정이 탄력적 IP 주소 할당량을 초과한 경우, 소스 계정에서 탄력적 IP 주소 전송을 활성화할 수 있지만 전송 계정이 전송을 수락하려고 하면 이 예외가 발생합니다. 기본적으로 모든 AWS 계정은 리전당 탄력적 IP 주소 5개로 제한됩니다. 제한 상황에 대한 지침은 Amazon EC2 사용 설명서의 [탄력적 IP 주소 제한](#)을 참조하세요.
- **InvalidTransfer.addressCustomPtrSet**: 귀하 또는 조직 내 다른 사용자가 귀하가 전송하려는 탄력적 IP 주소를 역방향 DNS 조회를 사용하도록 구성한 경우, 소스 계정에서 탄력적 IP 주소 전송을 활성화할 수 있지만 전송 계정이 전송을 수락하려고 하면 이 예외가 발생합니다. 이 문제를 해결하기 위해서는 소스 계정에서 탄력적 IP 주소의 DNS 레코드를 제거해야 합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [역방향 DNS 레코드 제거](#)를 참조하세요.
- **InvalidTransfer.addressAssociated**: 탄력적 IP 주소가 ENI 또는 EC2 인스턴스와 연결된 경우, 소스 계정에서 탄력적 IP 주소 전송을 활성화할 수 있지만 전송 계정이 전송을 수락하려고 하면 이 예외가 발생합니다. 이 문제를 해결하기 위해서는 탄력적 IP 주소의 연결을 해제해야 합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [탄력적 IP 주소 연결 해제](#)를 참조하세요.

기타 예외 사항은 [AWS Support에 문의하세요](#).

전송 계정으로 이 단계를 수행해야 합니다.

탄력적 IP 주소 전송 수락

1. 전송 계정을 사용 중인지 확인하세요.
2. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
3. 탐색 창에서 탄력적 IP(Elastic IPs)를 선택합니다.
4. Actions(작업), Accept transfer(전송 수락)를 선택합니다.
5. 전송을 수락하면 전송 중인 탄력적 IP 주소와 연결된 태그가 탄력적 IP 주소로 전송되지 않습니다. 수락하려는 탄력적 IP 주소의 Name(이름) 태그를 정의하려는 경우 Create a tag with a key of 'Name' and a value that you specify('Name' 키와 지정한 값으로 태그 생성)를 선택합니다.
6. 전송할 탄력적 IP 주소를 입력합니다.

7. 전송된 탄력적 IP 주소를 여러 개 수락 중인 경우 Add address(주소 추가)를 선택하여 추가 탄력적 IP 주소를 입력합니다.
8. 제출을 선택합니다.

탄력적 IP 주소 릴리스

탄력적 IP 주소가 더 이상 필요하지 않으면 해당 주소를 릴리스하는 것이 좋습니다. VPC에서 사용하기 위해 할당한 탄력적 IP 주소에 대한 요금은 발생하지만 인스턴스와 연결된 주소에 대해서는 요금이 발생하지 않습니다. 탄력적 IP 주소를 인스턴스 또는 네트워크 인터페이스와 연결해서는 안 됩니다.

엘라스틱 IP 주소를 해제합니다

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Elastic IPs]를 선택합니다.
3. 탄력적 IP 주소를 선택한 다음 작업(Actions), 탄력적 IP 주소 릴리스(Release Elastic IP addresses)를 선택합니다.
4. 메시지가 나타나면 [Release]를 선택합니다.

탄력적 IP 주소 복구

탄력적 IP 주소를 해제했지만 생각이 바뀐 경우 복구할 수 있습니다. 탄력적 IP 주소가 다른 AWS 계정에 할당되었거나, 복구로 인해 탄력적 IP 주소 할당량을 초과하는 경우에는 탄력적 IP 주소를 복구할 수 없습니다.

Amazon EC2 API 또는 명령줄 도구를 사용하여 탄력적 IP 주소를 복구할 수 있습니다.

AWS CLI를 사용하여 탄력적 IP 주소를 복구하려면

[allocate-address](#) 명령을 사용한 다음 --address 파라미터를 사용하여 IP 주소를 지정합니다.

```
aws ec2 allocate-address --domain vpc --address 203.0.113.3
```

API 및 명령 개요

명령줄 또는 API를 사용하여 이 섹션에서 설명하는 태스크를 수행할 수 있습니다. 명령줄 인터페이스 및 사용 가능한 API 작업 목록에 대한 자세한 내용은 [Amazon VPC 작업](#) 단원을 참조하십시오.

탄력적 IP 주소 전송 수락

- [accept-address-transfer](#)(AWS CLI)
- [Approve-EC2AddressTransfer](#)(AWS Tools for Windows PowerShell)

탄력적 IP 주소 할당

- [allocate-address](#) (AWS CLI)
- [New-EC2Address](#)(AWS Tools for Windows PowerShell)

인스턴스 또는 네트워크 인터페이스에 탄력적 IP 주소 연결

- [associate-address](#)(AWS CLI)
- [Register-EC2Address](#)(AWS Tools for Windows PowerShell)

탄력적 IP 주소 전송 설명

- [describe-address-transfers](#)(AWS CLI)
- [Get-EC2AddressTransfer](#)(AWS Tools for Windows PowerShell)

탄력적 IP 주소 전송 비활성화

- [disable-address-transfer](#)(AWS CLI)
- [Disable-EC2AddressTransfer](#)(AWS Tools for Windows PowerShell)

탄력적 IP 주소 연결 해제

- [disassociate-address](#)(AWS CLI)
- [Unregister-EC2Address](#)(AWS Tools for Windows PowerShell)

탄력적 IP 주소 전송 활성화

- [enable-address-transfer](#)(AWS CLI)
- [Enable-EC2AddressTransfer](#)(AWS Tools for Windows PowerShell)

탄력적 IP 주소 릴리스

- [release-address](#)(AWS CLI)
- [Remove-EC2Address](#)(AWS Tools for Windows PowerShell)

탄력적 IP 주소 태그

- [create-tags](#)(AWS CLI)
- [New-EC2Tag](#)(AWS Tools for Windows PowerShell)

탄력적 IP 주소 보기

- [describe-addresses](#)(AWS CLI)
- [Get-EC2Address](#)(AWS Tools for Windows PowerShell)

요금

탄력적 IP 주소의 효율적인 사용을 위해 소액의 시간당 요금이 부과됩니다. 자세한 내용은 [Amazon VPC 요금](#)의 퍼블릭 IPv4 주소를 참조하세요.

전송 게이트웨이를 사용하여 다른 VPC 및 네트워크에 VPC 연결

VPC, VPN 연결 및 AWS Direct Connect 연결 간에 트래픽을 라우팅하는 중앙 허브 역할을 하는 전송 게이트웨이를 사용하여 Virtual Private Cloud(VPC)와 온프레미스 네트워크를 연결할 수 있습니다. 자세한 내용은 [AWS Transit Gateway](#)를 참조하세요.

다음 표에서는 전송 게이트웨이의 몇 가지 일반적인 사용 사례를 설명하고 Amazon VPC 전송 게이트웨이의 추가 정보에 대한 링크를 제공합니다.

예	사용량
중앙 집중식 라우터	Transit Gateway를 모든 VPC, AWS Direct Connect 및 AWS Site-to-Site VPN 연결을 연결하는 중앙 집중식 라우터로 구성합니다. 자세한 내용은 예: 중앙 집중식 라우터 를 참조하세요.
격리된 VPC	Transit Gateway를 여러 격리된 라우터로 구성합니다. 이는 여러 개의 Transit Gateway를 사용하는 것과 유사하지만 라우팅 및 연

예	사용량
	결이 변경될 수 있는 경우 더 많은 유연성을 제공합니다. 자세한 내용은 예: 격리된 VPC 를 참조하세요.
공유 서비스를 사용하는 격리된 VPC	Transit Gateway를 공유 서비스를 사용하는 여러 격리된 라우터로 구성합니다. 이는 여러 개의 Transit Gateway를 사용하는 것과 유사하지만 라우팅 및 연결이 변경될 수 있는 경우 더 많은 유연성을 제공합니다. 자세한 내용은 예: 공유 서비스를 사용하는 격리된 VPC 를 참조하세요.

AWS Virtual Private Network를 사용하여 VPC를 원격 네트워크에 연결

다음 VPN 연결 옵션을 사용하여 VPC를 원격 네트워크 및 사용자에게 연결할 수 있습니다.

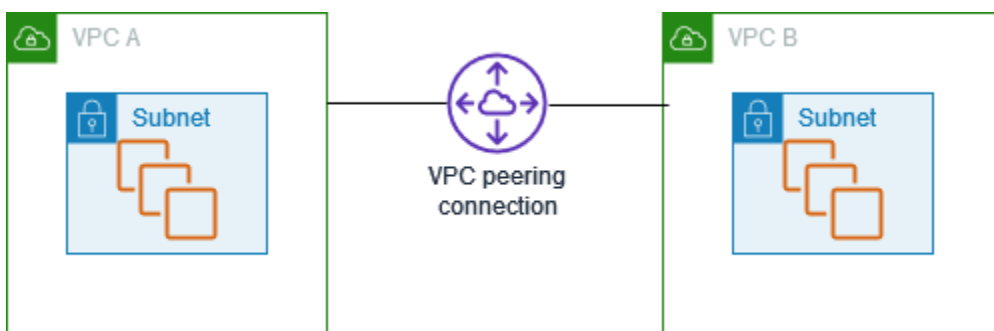
VPN 연결 옵션	설명
AWS Site-to-Site VPN	VPC와 원격 네트워크 사이에 IPsec VPN 연결을 생성할 수 있습니다. AWS 측 Site-to-Site VPN 연결에서 가상 프라이빗 게이트웨이 또는 Transit Gateway는 자동 장애 조치를 위한 2개의 VPN 엔드포인트(터널)를 제공합니다. Site-to-Site VPN 원격 연결 측에서 고객 게이트웨이 디바이스를 구성합니다. 자세한 내용은 AWS Site-to-Site VPN 사용 설명서 를 참조하세요.
AWS Client VPN	AWS Client VPN은 AWS 리소스와 온프레미스 네트워크에 안전하게 액세스할 수 있게 해주는 관리형 클라이언트 기반 VPN 서비스입니다. AWS Client VPN을 사용하여 사용자가 연결할 수 있는 엔드포인트를 구성하여 보안 TLS VPN 세션을 설정할 수 있습니다. 이렇게 하면 클라이언트가 OpenVPN 기반 VPN 클라이언트를 사용하여 어느 위치에서든 온프레미스 또는 AWS의 리소스에 액세스할 수 있습니다. 자세한 내용은 AWS Client VPN 관리자 안내서 를 참조하세요.
AWS VPN CloudHub	원격 네트워크가 두 개 이상인 경우(예: 여러 지사 사무실), 가상 프라이빗 게이트웨이를 통해 AWS Site-to-Site VPN 연결을 여러 개 만들고 이들 네트워크 사이의 통신을 활성화할 수 있습니다. 자세한 내용은 AWS Site-

VPN 연결 옵션	설명
	to-Site VPN 사용 설명서의 VPN CloudHub를 사용하여 사이트 간 보안 통신 제공 을 참조하세요.
타사 소프트웨어 VPN 어플라이언스	서드 파티 소프트웨어 VPN 어플라이언스를 실행 중인 VPC에서 Amazon EC2 인스턴스를 사용하여 원격 네트워크에 대한 VPN 연결을 생성할 수 있습니다. AWS는 서드 파티 소프트웨어 VPN 어플라이언스를 제공하거나 유지 관리하지 않지만, 파트너와 오픈 소스 커뮤니티에서 제공하는 다양한 제품 중에서 선택할 수 있습니다. AWS Marketplace 에서 서드 파티 소프트웨어 VPN 어플라이언스를 찾으세요.

또한 AWS Direct Connect를 사용하여 원격 네트워크에서 VPC까지 전용 프라이빗 연결을 생성할 수도 있습니다. 이 연결을 AWS Site-to-Site VPN과 결합하여 IPsec 암호화 연결을 생성할 수 있습니다. 자세한 정보는 AWS Direct Connect 사용 설명서의 [AWS Direct Connect\(이\)란 무엇입니까?](#) 섹션을 참조하세요.

VPC 피어링을 사용하여 VPC 연결

VPC 피어링 연결은 비공개적으로 두 VPC 간에 트래픽을 라우팅할 수 있도록 하기 위한 두 VPC 사이의 네트워킹 연결입니다. 피어링된 VPC의 리소스는 동일한 네트워크 내에 있는 것처럼 서로 통신할 수 있습니다. 자체 VPC 간, 다른 AWS 계정에서 VPC를 사용하여 또는 다른 AWS 리전에서 VPC를 사용하여 VPC 피어링 연결을 생성할 수 있습니다. 피어링된 VPC 간의 트래픽은 공용 인터넷을 통과하지 않습니다.



AWS는 VPC의 기존 인프라를 사용하여 VPC 피어링 연결을 생성합니다. VPC 피어링 연결은 게이트웨이나 AWS Site-to-Site VPN 연결이 아니며 별도의 물리적 하드웨어에 의존하지 않습니다. 그러므로 통신 또는 대역폭 병목에 대한 단일 지점 장애가 없습니다.

자세한 내용은 [Amazon VPC 피어링 가이드](#)를 참조하세요.

VPC 모니터링

다음 도구를 사용하여 Virtual Private Cloud(VPC)에서 트래픽 또는 네트워크 액세스를 모니터링할 수 있습니다.

VPC 흐름 로그

VPC 흐름 로그를 사용하여 VPC의 네트워크 인터페이스에서 송수신되는 트래픽에 대한 세부 정보를 캡처할 수 있습니다.

Amazon VPC IP 주소 관리자(IPAM)

IPAM을 사용하여 워크로드의 IP 주소를 계획, 추적 및 모니터링할 수 있습니다. 자세한 내용은 [IP 주소 관리자](#)를 참조하세요.

트래픽 미러링

이 기능을 사용하여 Amazon EC2 인스턴스의 네트워크 인터페이스에서 네트워크 트래픽을 복사하고 심층 패킷 검사를 위해 대역 외 보안 및 모니터링 어플라이언스로 전송할 수 있습니다. 네트워크 및 보안 이상을 감지하고, 운영 인사이트를 얻고, 규정 준수 및 보안 제어를 구현하고, 문제를 해결할 수 있습니다. 자세한 내용은 [트래픽 미러링](#)을 참조하세요.

Reachability Analyzer

이 도구를 사용하여 VPC에 있는 두 리소스 간의 네트워크 연결성을 분석하고 디버깅할 수 있습니다. 소스 및 대상 리소스를 지정한 후 Reachability Analyzer는 연결할 수 있는 경우 두 리소스 간의 가상 경로에 대한 출발 세부 정보를 생성하고 연결할 수 없는 경우 차단 구성 요소를 식별합니다. 자세한 내용은 [Reachability Analyzer](#)를 참조하세요.

Network Access Analyzer

Network Access Analyzer를 사용하여 리소스에 대한 네트워크 액세스를 파악할 수 있습니다. 이를 통해 네트워크 보안 태세의 개선 사항을 식별하고 네트워크가 특정 규정 준수 요구 사항을 충족함을 입증할 수 있습니다. 자세한 내용은 [Network Access Analyzer](#)를 참조하세요.

CloudTrail 로그

AWS CloudTrail을 사용하여 Amazon VPC API 호출에 대한 자세한 정보를 캡처할 수 있습니다. 생성된 CloudTrail 로그를 사용하여 어떤 요청이 이루어졌는지, 어떤 소스 IP 주소에서 요청을 했는지, 누가 언제 요청했는지 등을 확인할 수 있습니다. 자세한 내용은 Amazon EC2 API 참조(Amazon EC2 API Reference)의 [AWS CloudTrail를 사용하여 Amazon EC2 Amazon EBS 및 Amazon VPC](#)

[API 호출 로깅\(Logging Amazon EC2 Amazon EBS, and Amazon VPC API calls using \)](#)을 참조하세요.

VPC 흐름 로그를 사용하여 IP 트래픽 로깅

VPC 흐름 로그는 VPC의 네트워크 인터페이스에서 전송되고 수신되는 IP 트래픽에 대한 정보를 수집할 수 있는 기능입니다. 흐름 로그 데이터가 게시될 수 있는 위치는 Amazon CloudWatch Logs, Amazon S3 또는 Amazon Data Firehose입니다. 흐름 로그를 생성하면 구성된 로그 그룹, 버킷 또는 전송 스트림의 흐름 로그 레코드를 검색하고 볼 수 있습니다.

흐름 로그는 다음과 같은 여러 작업에 도움이 될 수 있습니다.

- 지나치게 제한적인 보안 그룹 규칙 진단
- 인스턴스에 도달하는 트래픽 모니터링
- 네트워크 인터페이스를 오가는 트래픽 방향 결정

흐름 로그 데이터는 네트워크 트래픽 경로 외부에서 수집되므로 네트워크 처리량이나 지연 시간에 영향을 주지 않습니다. 네트워크 성능에 영향을 주지 않고 흐름 로그를 생성하거나 삭제할 수 있습니다.

Note

이 섹션에서는 VPC의 흐름 로그에 대해서만 설명합니다. 버전 6에 도입된 전송 게이트웨이의 흐름 로그에 대한 자세한 내용은 Amazon VPC Transit Gateway 사용 설명서의 [Logging network traffic using Transit Gateway Flow Logs](#)를 참조하세요.

내용

- [흐름 로그 기본 사항](#)
- [흐름 로그 레코드](#)
- [흐름 로그 레코드의 예](#)
- [흐름 로그 제한](#)
- [요금](#)
- [흐름 로그 작업](#)
- [CloudWatch Logs에 흐름 로그 게시](#)

- [Amazon S3에 흐름 로그 게시](#)
- [Amazon Data Firehose에 흐름 로그 게시](#)
- [Amazon Athena를 사용하여 흐름 로그 쿼리](#)
- [VPC 흐름 로그 문제 해결](#)

흐름 로그 기본 사항

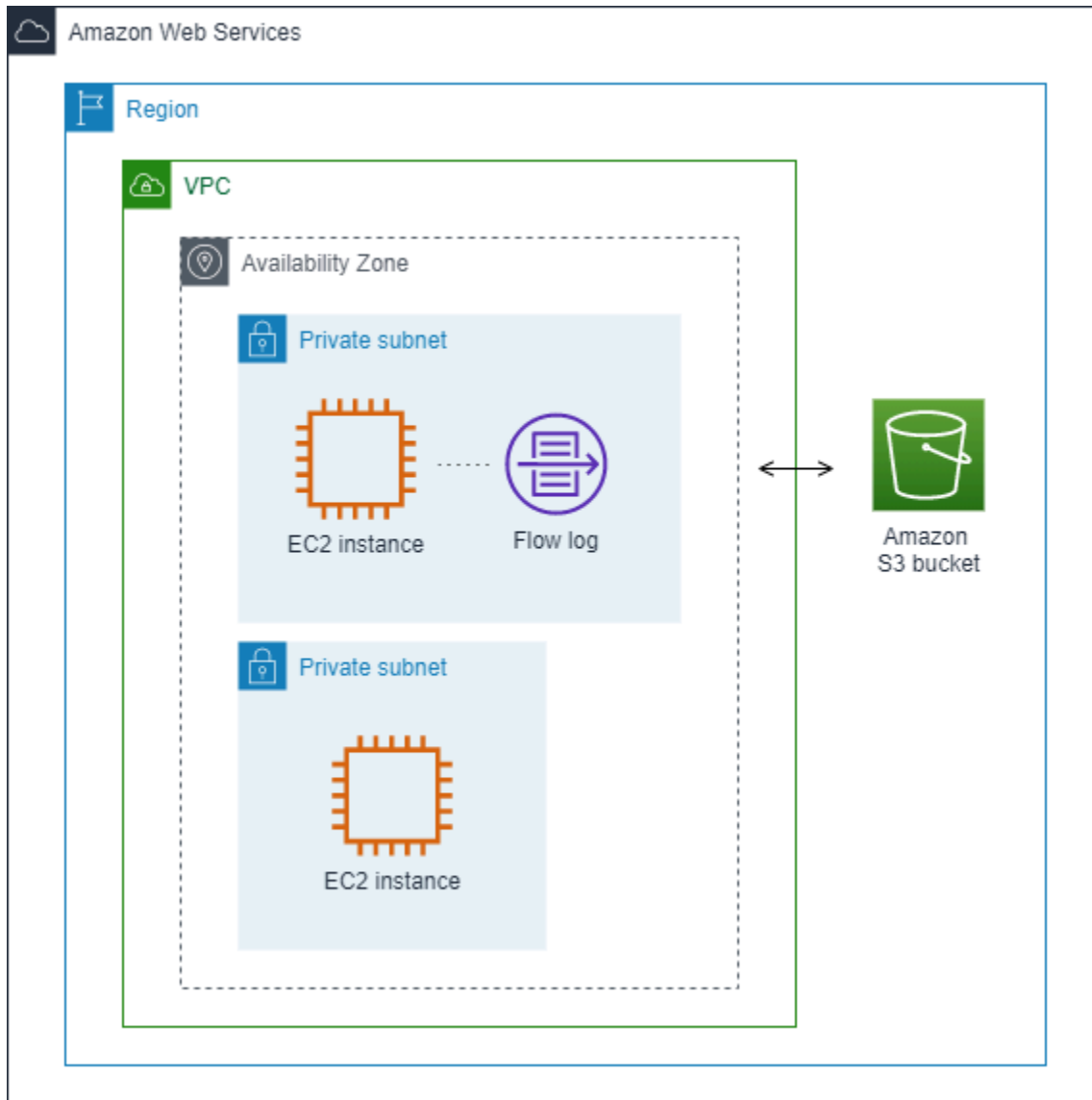
VPC, 서브넷 또는 네트워크 인터페이스에 대한 흐름 로그를 생성할 수 있습니다. 서브넷이나 VPC에 대한 흐름 로그를 생성할 경우, VPC 또는 서브넷의 각 네트워크 인터페이스가 모니터링됩니다.

모니터링된 네트워크 인터페이스를 위한 흐름 로그 데이터는 트래픽 흐름을 설명하는 필드로 구성된 로그 이벤트인 흐름 로그 레코드로서 기록됩니다. 자세한 내용은 [흐름 로그 레코드](#) 단원을 참조하세요.

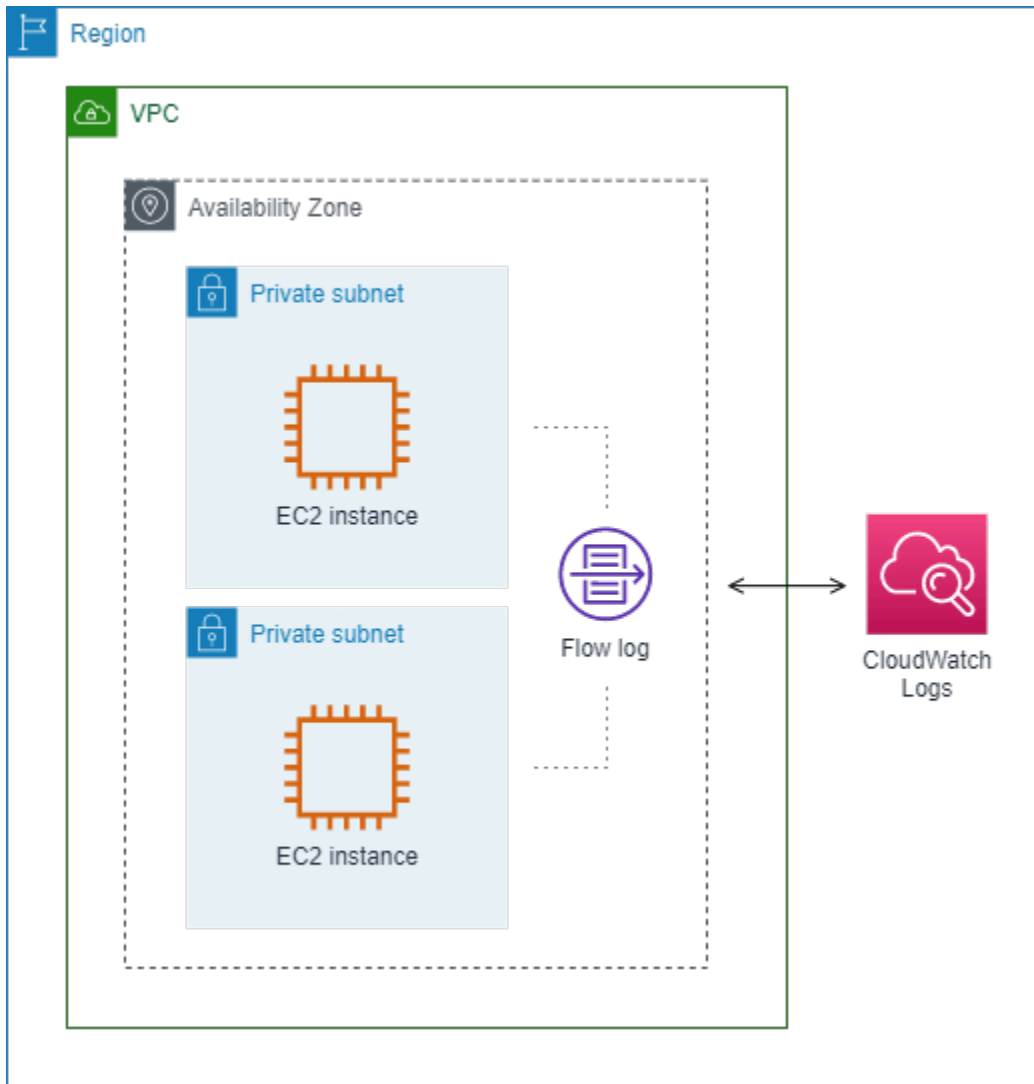
흐름 로그를 생성하려면 다음을 지정합니다.

- 흐름 로그를 생성할 리소스
- 캡처할 트래픽 유형(허용된 트래픽, 거부된 트래픽 또는 모든 트래픽)
- 흐름 로그 데이터를 게시할 대상

다음 예에서는 프라이빗 서브넷의 EC2 인스턴스 중 하나의 네트워크 인터페이스에 대해 허용된 트래픽을 캡처하고 흐름 로그 레코드를 Amazon S3 버킷에 게시하는 흐름 로그를 생성합니다.



다음 예에서 흐름 로그는 서브넷의 모든 트래픽을 캡처하고 흐름 로그 레코드를 Amazon CloudWatch Logs에 게시합니다. 흐름 로그는 서브넷의 모든 네트워크 인터페이스에 대한 트래픽을 캡처합니다.



흐름 로그를 생성한 후에는, 데이터를 수집하여 선택된 대상에 게시하는 데 몇 분의 시간이 소요될 수 있습니다. 흐름 로그는 네트워크 인터페이스에 대한 로그 스트림을 실시간으로 캡처하지 않습니다. 자세한 내용은 [흐름 로그 생성](#) 단원을 참조하십시오.

서브넷이나 VPC에 대한 흐름 로그를 생성한 후 서브넷에서 하나의 인스턴스를 시작할 경우, 해당 네트워크 인터페이스에 대한 네트워크 트래픽이 생기는 즉시 새로운 네트워크 인터페이스에 대한 (CloudWatch Logs용) 로그 스트림 또는 (Amazon S3용) 로그 파일 객체가 생성됩니다.

다음과 같은 다른 AWS 서비스에서 생성한 네트워크 인터페이스에 대한 흐름 로그를 생성할 수 있습니다.

- Elastic Load Balancing
- Amazon RDS
- Amazon ElastiCache

- Amazon Redshift
- Amazon WorkSpaces
- NAT 게이트웨이
- 전송 게이트웨이

네트워크 인터페이스 유형에 관계없이 Amazon EC2 콘솔 또는 Amazon EC2 API를 사용하여 네트워크 인터페이스에 대한 흐름 로그를 작성해야 합니다.

흐름 로그에 태그를 적용할 수 있습니다. 각 태그는 사용자가 정의하는 키와 선택적 값으로 구성됩니다. 태그는 흐름 로그를 용도나 소유자별로 구성하는 데 도움이 될 수 있습니다.

흐름 로그가 더 이상 필요하지 않을 경우 삭제할 수 있습니다. 흐름 로그를 삭제하면 리소스에 대한 흐름 로그 서비스가 비활성화되어 생성되거나 게시되는 새 흐름 로그 레코드가 없습니다. 흐름 로그를 삭제해도 기존 흐름 로그 데이터는 삭제되지 않습니다. 흐름 로그를 삭제하면 작업을 마무리했을 때 대상에서 직접 흐름 로그 데이터를 삭제할 수 있습니다. 자세한 내용은 [흐름 로그 삭제](#) 단원을 참조하십시오.

흐름 로그 레코드

흐름 로그 레코드는 VPC에 네트워크 흐름을 나타냅니다. 기본적으로 각 레코드는 캡처 기간이라고도 하는 집계 간격 내에 발생하는 네트워크 인터넷 프로토콜(IP) 트래픽 흐름(네트워크 인터페이스별로 5 튜플을 특징으로 함)을 캡처합니다.

각 레코드는 필드가 공백으로 구분되어 있는 문자열입니다. 레코드에는 소스, 대상, 프로토콜 등 IP 흐름의 다양한 구성 요소에 대한 값이 포함됩니다.

흐름 로그를 생성할 때 흐름 로그 레코드의 기본 형식을 사용하거나 사용자 지정 형식을 지정할 수 있습니다.

목차

- [집계 간격](#)
- [기본 형식](#)
- [사용자 지정 형식](#)
- [사용 가능한 필드](#)

집계 간격

집계 간격은 특정 흐름이 캡처되어 흐름 로그 레코드로 집계되는 기간입니다. 기본적으로 최대 집계 간격은 10분입니다. 흐름 로그를 만들 때 선택적으로 최대 집계 간격을 1분으로 지정할 수 있습니다. 최대 집계 간격이 1분인 흐름 로그는 최대 집계 간격이 10분인 흐름 로그보다 더 많은 양의 흐름 로그 레코드를 생성합니다.

네트워크 인터페이스가 [NITRO 기반 인스턴스](#)에 연결된 경우 집계 간격은 지정된 최대 집계 간격에 관계없이 항상 1분 이하입니다.

집계 간격 내에서 데이터를 캡처한 후에는 데이터를 처리하고 CloudWatch Logs 또는 Amazon S3에 게시하느라 추가 시간이 걸립니다. 흐름 로그 서비스는 일반적으로 로그를 약 5분 만에 CloudWatch Logs로 전송하고 약 10분 만에 Amazon S3로 전송합니다. 그러나 로그 전달에 최선의 노력을 기울이고 로그가 일반적인 전달 시간을 초과하여 지연될 수 있습니다.

기본 형식

기본 형식의 흐름 로그 레코드에는 [사용 가능한 필드](#) 테이블에 표시되는 순서대로 버전 2 필드가 포함됩니다. 기본 형식을 사용자 정의하거나 변경할 수 없습니다. 추가 필드 또는 다른 필드 하위 세트를 캡처하려면 사용자 지정 형식을 지정합니다.

사용자 지정 형식

사용자 지정 형식을 사용하면 흐름 로그 레코드에 포함되는 필드와 그 순서를 지정할 수 있습니다. 이를 통해 요구 사항에 맞는 흐름 로그를 만들고 관련이 없는 필드를 생략할 수 있습니다. 사용자 지정 형식을 사용하면 게시된 흐름 로그에서 특정 정보를 추출하기 위해 별도의 프로세스가 필요하지 않습니다. 사용 가능한 흐름 로그 필드를 얼마든지 지정할 수 있지만 하나 이상을 지정해야 합니다.

사용 가능한 필드

다음 표는 흐름 로그 레코드에 사용 가능한 모든 필드를 설명합니다. 버전(Version) 열은 해당 필드를 도입한 VPC 흐름 로그의 버전을 나타냅니다. 기본 형식에는 모든 버전 2 필드가 테이블에 표시되는 순서와 동일하게 포함됩니다.

Amazon S3 흐름 로그 데이터를 게시할 때 필드의 데이터 유형은 흐름 로그 형식에 따라 다릅니다. 형식이 일반 텍스트인 경우 모든 필드는 STRING 유형입니다. 형식이 Parquet 인 경우 필드 데이터 유형에 대한 표를 참조하십시오.

필드를 적용할 수 없거나 특정 레코드에 대해 계산할 수 없는 경우 레코드는 해당 항목에 대해 '-' 기호를 표시합니다. 패킷 헤더에서 직접 제공되지 않는 메타데이터 필드는 최선의 작업 수준 근사값이며 해당 값이 누락되거나 정확하지 않을 수 있습니다.

필드	설명	버전
version	VPC 흐름 로그 버전. 기본 형식을 사용하는 경우, 버전은 2입니다. 사용자 지정 형식을 사용하는 경우, 버전은 지정된 필드 중에서 가장 높은 버전입니다. 예를 들어 버전 2의 필드만 지정한다면 버전은 2가 됩니다. 버전 2, 3 및 4의 필드를 혼합하여 지정한다면 버전은 4가 됩니다. Parquet 데이터 유형: INT_32	2
account-id	트래픽이 기록되는 소스 네트워크 인터페이스 소유자의 AWS 계정 ID입니다. 네트워크 인터페이스가 AWS 서비스에 의해 생성된 경우, 예를 들어 VPC 엔드포인트 또는 Network Load Balancer 생성 시 이 필드의 레코드에 unknown이 표시될 수도 있습니다. Parquet 데이터 유형: 문자열	2
interface-id	트래픽이 기록되는 네트워크 인터페이스 ID. Parquet 데이터 유형: 문자열	2
srcaddr	들어오는 트래픽의 소스 주소 또는 네트워크 인터페이스의 나가는 트래픽의 네트워크 인터페이스의 IPv4 또는 IPv6 주소. 네트워크 인터페이스의 IPv4 주소는 항상 해당 프라이빗 IPv4 주소입니다. 또한 pkt-srcaddr 단원도 참조하십시오. Parquet 데이터 유형: 문자열	2
dstaddr	나가는 트래픽의 대상 주소 또는 네트워크 인터페이스의 들어오는 트래픽의 네트워크 인터페이스의 IPv4 또는 IPv6 주소. 네트워크 인터페이스의 IPv4 주소는 항상 해당 프라이빗 IPv4 주소입니다. 또한 pkt-dstaddr 단원도 참조하십시오. Parquet 데이터 유형: 문자열	2
srcport	트래픽의 소스 포트 Parquet 데이터 유형: INT_32	2
dstport	트래픽의 대상 포트	2

필드	설명	버전
	Parquet 데이터 유형: INT_32	
protocol	트래픽의 IANA 프로토콜 번호. 자세한 정보는 지정된 인터넷 프로토콜 번호 단원을 참조하십시오. Parquet 데이터 유형: INT_32	2
packets	흐름 중 전송된 패킷 수. Parquet 데이터 유형: INT_64	2
bytes	흐름 중 전송된 바이트 수. Parquet 데이터 유형: INT_64	2
start	흐름의 첫 번째 패킷이 집계 간격 내에서 수신된 시간(단위: Unix 초)입니다. 이 시간은 패킷이 네트워크 인터페이스에서 전송되거나 수신된 후 최대 60초가 될 수 있습니다. Parquet 데이터 유형: INT_64	2
end	집계 간격 내에서 흐름의 마지막 패킷을 수신한 시간(단위: Unix 초)입니다. 이 시간은 패킷이 네트워크 인터페이스에서 전송되거나 수신된 후 최대 60초가 될 수 있습니다. Parquet 데이터 유형: INT_64	2
action	트래픽과 연결된 작업 <ul style="list-style-type: none"> ACCEPT — 트래픽이 수락되었습니다. REJECT — 트래픽이 거부되었습니다. 예를 들어 보안 그룹이나 네트워크 ACL에서 트래픽을 허용하지 않았거나 연결이 닫힌 후 패킷이 도착했습니다. Parquet 데이터 유형: 문자열	2

필드	설명	버전
log-status	<p>흐름 로그의 로깅 상태:</p> <ul style="list-style-type: none"> • OK — 선택된 대상에 정상적으로 로깅됩니다. • NODATA — 집계 간격 중 네트워크 인터페이스에서 전송하거나 수신된 네트워크 트래픽이 없었습니다. • SKIPDATA — 집계 간격 중 일부 흐름 로그 레코드를 건너뛰었습니다. 내부 용량 제한 또는 내부 오류가 원인일 수 있습니다. <p>Parquet 데이터 유형: 문자열</p>	2
vpc-id	<p>트래픽이 기록되는 네트워크 인터페이스를 포함하는 VPC의 ID.</p> <p>Parquet 데이터 유형: 문자열</p>	3
subnet-id	<p>트래픽이 기록되는 네트워크 인터페이스를 포함하는 서브넷의 ID.</p> <p>Parquet 데이터 유형: 문자열</p>	3
instance-id	<p>인스턴스를 소유한 경우 트래픽이 기록되는 네트워크 인터페이스와 연결된 인스턴스의 ID입니다. NAT 게이트웨이의 네트워크 인터페이스 같은 요청자 관리 네트워크 인터페이스에는 '-' 기호를 반환합니다.</p> <p>Parquet 데이터 유형: 문자열</p>	3

필드	설명	버전
tcp-flags	<p>다음 TCP 플래그의 비트 마스크 값:</p> <ul style="list-style-type: none"> • FIN — 1 • SYN — 2 • RST — 4 • SYN-ACK — 18 <p>지원되는 플래그가 기록되지 않는 경우 TCP 플래그 값은 0입니다. 예를 들어 tcp-flags는 ACK 또는 PSH 플래그 로깅을 지원하지 않으므로 이러한 지원되지 않는 플래그가 있는 트래픽에 대한 레코드의 결과는 tcp-flags 값 0이 됩니다. 하지만 지원되지 않는 플래그에 지원되는 플래그가 동반된 경우 지원되는 플래그의 값을 보고합니다. 예를 들어 ACK가 SYN-ACK의 일부인 경우 18을 보고합니다. 그리고 SYN+ECE와 같은 레코드가 있는 경우 SYN은 지원되는 플래그이고 ECE는 지원되지 않으므로 TCP 플래그 값은 2입니다. 어떤 이유로든 플래그 조합이 유효하지 않아 값을 계산할 수 없는 경우 값은 '-'입니다. 플래그가 전송되지 않는 경우 TCP 플래그 값은 0입니다.</p> <p>TCP 플래그는 집계 간격 동안 OR일 수 있습니다. 짧은 연결의 경우 SYN-ACK 및 FIN에 대해 19, SYN 및 FIN에 대해 3과 같이 흐름 로그 레코드의 동일한 행에 플래그가 설정될 수 있습니다. 문제 해결 예는 TCP 플래그 시퀀스을(를) 참조하십시오.</p> <p>TCP 플래그에 대한 일반 정보(예: FIN, SYN 및 ACK와 같은 플래그의 의미)는 Wikipedia에서 TCP 세그먼트 구조를 참조하십시오.</p> <p>Parquet 데이터 유형: INT_32</p>	3
type	<p>트래픽 유형입니다. 가능한 값: IPv4 IPv6 EFA. 자세한 내용은 Elastic Fabric Adapter(EFA) 섹션을 참조하십시오.</p> <p>Parquet 데이터 유형: 문자열</p>	3

필드	설명	버전
pkt-srcaddr	<p>트래픽의 패킷 수준(원본) 소스 IP 주소입니다. 이 필드를 srcaddr 필드와 함께 사용하면 트래픽이 흐르는 중간 계층의 IP 주소와 트래픽의 원래 소스 IP 주소를 구별 할 수 있습니다. 대표적인 경우는 트래픽이 NAT 게이트웨이에 대한 네트워크 인터페이스를 통과하거나 Amazon EKS의 포드 IP 주소가 포드가 실행 중인(VPC 내 통신용) 인스턴스 노드의 네트워크 인터페이스 IP 주소와 다른 경우입니다.</p> <p>Parquet 데이터 유형: 문자열</p>	3
pkt-dstaddr	<p>트래픽의 패킷 수준(원본) 대상 IP 주소입니다. 이 필드를 dstaddr 필드와 함께 사용하면 트래픽이 흐르는 중간 계층의 IP 주소와 트래픽의 최종 대상 IP 주소를 구별 할 수 있습니다. 대표적인 경우는 트래픽이 NAT 게이트웨이에 대한 네트워크 인터페이스를 통과하거나 Amazon EKS의 포드 IP 주소가 포드가 실행 중인(VPC 내 통신용) 인스턴스 노드의 네트워크 인터페이스 IP 주소와 다른 경우입니다.</p> <p>Parquet 데이터 유형: 문자열</p>	3
region	<p>트래픽이 기록되는 네트워크 인터페이스가 포함된 리전입니다.</p> <p>Parquet 데이터 유형: 문자열</p>	4
az-id	<p>트래픽이 기록되는 네트워크 인터페이스가 포함된 가용 영역의 ID입니다. 하위 위치에서 트래픽이 발생한 경우 레코드는 이 필드에 대해 '-' 기호를 표시합니다.</p> <p>Parquet 데이터 유형: 문자열</p>	4
sublocation-type	<p>sublocation-id 필드에 반환되는 하위 위치 유형입니다. 가능한 값: wavelength outpost localzone 트래픽이 하위 위치에서 발생하지 않는 경우 레코드는 이 필드에 대해 '-' 기호를 표시합니다.</p> <p>Parquet 데이터 유형: 문자열</p>	4

필드	설명	버전
sublocation-id	트래픽이 기록되는 네트워크 인터페이스가 포함된 하위 위치의 ID입니다. 트래픽이 하위 위치에서 발생하지 않는 경우 레코드는 이 필드에 대해 '-' 기호를 표시합니다. Parquet 데이터 유형: 문자열	4
pkt-src-aws-service	소스 IP 주소가 AWS 서비스용인 경우 pkt-srcaddr 필드에 대한 IP 주소 범위 의 하위 집합 이름입니다. pkt-srcaddr이 결친 범위 에 속하는 경우 pkt-src-aws-service는 AWS 서비스 코드 중 하나만 표시합니다. 가능한 값: AMAZON AMAZON_APPFLOW AMAZON_CONNECT API_GATEWAY CHIME_MEETINGS CHIME_VOICECONNECTOR CLOUD9 CLOUDFRONT CODEBUILD DYNAMODB EBS EC2 EC2_INSTANCE_CONNECT GLOBALACCELERATOR KINESIS_VIDEO_STREAMS ROUTE53 ROUTE53_HEALTHCHECKS ROUTE53_HEALTHCHECKS_PUBLISHING ROUTE53_RESOLVER S3 WORKSPACES_GATEWAYS. Parquet 데이터 유형: 문자열	5
pkt-dst-aws-service	대상 IP 주소가 AWS 서비스용인 경우 pkt-dstaddr 필드에 대한 IP 주소 범위의 하위 집합 이름입니다. 가능한 값 목록은 pkt-src-aws-service 필드를 참조하십시오. Parquet 데이터 유형: 문자열	5
flow-direction	트래픽이 캡처되는 인터페이스에 대한 흐름 방향입니다. 가능한 값: ingress egress Parquet 데이터 유형: 문자열	5

필드	설명	버전
traffic-path	<p>송신 트래픽이 대상으로 이동하는 경로입니다. 트래픽이 송신 트래픽인지 여부를 확인하려면 flow-direction 필드를 확인하십시오. 가능한 값은 다음과 같습니다. 적용되는 값이 없는 경우 필드는 -로 설정됩니다.</p> <ul style="list-style-type: none"> • 1 - VPC에서 네트워크 인터페이스를 생성하는 리소스를 포함하여 동일한 VPC의 다른 리소스를 통해 • 2 — 인터넷 게이트웨이 또는 게이트웨이 VPC 엔드포인트를 통해 • 3 — 가상 프라이빗 게이트웨이를 통해 • 4 — 리전 내 VPC 피어링 연결을 통해 • 5 — 리전 간 VPC 피어링 연결을 통해 • 6 — 로컬 게이트웨이를 통해 • 7 — 게이트웨이 VPC 엔드포인트를 통해(Nitro 기반 인스턴스에만 해당) • 8 — 인터넷 게이트웨이를 통해(Nitro 기반 인스턴스만 해당) <p>Parquet 데이터 유형: INT_32</p>	5
ecs-cluster-arn	<p>트래픽이 실행 중인 ECS 태스크에서 발생하는 경우 ECS 클러스터의 AWS 리소스 이름(ARN)입니다. 구독에 이 필드를 포함하려면 ecs:ListClusters를 직접적으로 호출할 수 있는 권한이 필요합니다.</p> <p>Parquet 데이터 유형: 문자열</p>	7
ecs-cluster-name	<p>트래픽이 실행 중인 ECS 태스크에서 발생하는 경우 ECS 클러스터의 이름입니다. 구독에 이 필드를 포함하려면 ecs:ListClusters를 직접적으로 호출할 수 있는 권한이 필요합니다.</p> <p>Parquet 데이터 유형: 문자열</p>	7
ecs-container-instance-arn	<p>트래픽이 EC2 인스턴스에서 실행 중인 ECS 태스크에서 발생하는 경우 ECS 컨테이너 인스턴스의 ARN입니다. 용량 공급자가 AWS Fargate인 경우 이 필드는 '-'가 됩니다. 구독에 이 필드를 포함하려면 ecs:ListClusters와 ecs:ListContainerInstances를 직접적으로 호출할 수 있는 권한이 필요합니다.</p> <p>Parquet 데이터 유형: 문자열</p>	7

필드	설명	버전
ecs-container-instance-id	트래픽이 EC2 인스턴스에서 실행 중인 ECS 태스크에서 발생하는 경우 ECS 컨테이너 인스턴스의 ID입니다. 용량 공급자가 AWS Fargate인 경우 이 필드는 '-'가 됩니다. 구독에 이 필드를 포함하려면 ecs:ListClusters와 ecs:ListContainerInstances를 직접적으로 호출할 수 있는 권한이 필요합니다. Parquet 데이터 유형: 문자열	7
ecs-container-id	트래픽이 실행 중인 ECS 태스크에서 발생하는 경우 컨테이너의 Docker 런타임 ID입니다. ECS 태스크에 컨테이너가 하나 이상 있는 경우 이 ID는 첫 번째 컨테이너의 Docker 런타임 ID가 됩니다. 구독에 이 필드를 포함하려면 ecs:ListClusters를 직접적으로 호출할 수 있는 권한이 필요합니다. Parquet 데이터 유형: 문자열	7
ecs-second-container-id	트래픽이 실행 중인 ECS 태스크에서 발생하는 경우 컨테이너의 Docker 런타임 ID입니다. ECS 태스크에 컨테이너가 두 개 이상 있는 경우 이 ID는 두 번째 컨테이너의 Docker 런타임 ID가 됩니다. 구독에 이 필드를 포함하려면 ecs:ListClusters를 직접적으로 호출할 수 있는 권한이 필요합니다. Parquet 데이터 유형: 문자열	7
ecs-service-name	트래픽이 실행 중인 ECS 태스크에서 발생하고 ECS 태스크가 ECS 서비스에 의해 시작되는 경우 ECS 서비스의 이름입니다. ECS 태스크가 ECS 서비스에 의해 시작되지 않는 경우 이 필드는 '-'가 됩니다. 구독에 이 필드를 포함하려면 ecs:ListClusters와 ecs:ListServices를 직접적으로 호출할 수 있는 권한이 필요합니다. Parquet 데이터 유형: 문자열	7
ecs-task-definition-arn	트래픽이 실행 중인 ECS 태스크에서 발생하는 경우 ECS 태스크 정의의 ARN입니다. 구독에 이 필드를 포함하려면 ecs:ListClusters와 ecs:ListTaskDefinitions를 직접적으로 호출할 수 있는 권한이 필요합니다. Parquet 데이터 유형: 문자열	7

필드	설명	버전
ecs-task-arn	트래픽이 실행 중인 ECS 태스크에서 발생하는 경우 ECS 태스크의 ARN입니다. 구독에 이 필드를 포함하려면 ecs:ListClusters와 ecs:ListTasks를 직접적으로 호출할 수 있는 권한이 필요합니다. Parquet 데이터 유형: 문자열	7
ecs-task-id	트래픽이 실행 중인 ECS 태스크에서 발생하는 경우 ECS 태스크의 ID입니다. 구독에 이 필드를 포함하려면 ecs:ListClusters와 ecs:ListTasks를 직접적으로 호출할 수 있는 권한이 필요합니다. Parquet 데이터 유형: 문자열	7

흐름 로그 레코드의 예

다음은 특정 트래픽 흐름을 캡처하는 흐름 로그 레코드의 예입니다.

흐름 로그 레코드 형식에 대한 자세한 내용은 [흐름 로그 레코드](#)를 참조하십시오. 흐름 로그를 생성하는 방법에 대한 자세한 내용은 [흐름 로그 작업](#) 단원을 참조하십시오.

목차

- [허용 및 거부된 트래픽](#)
- [데이터가 없고 건너뛴 레코드](#)
- [보안 그룹 및 네트워크 ACL 규칙](#)
- [IPv6 트래픽](#)
- [TCP 플래그 시퀀스](#)
- [NAT 게이트웨이를 통한 트래픽](#)
- [전송 게이트웨이를 통한 트래픽](#)
- [서비스 이름, 트래픽 경로 및 흐름 방향](#)

허용 및 거부된 트래픽

다음은 기본 흐름 로그 레코드의 예입니다.

이 예시에서는 프라이빗 IP 주소가 172.31.16.21이고 계정 123456789010의 ID가 eni-1235b8ca123456789인 네트워크 인터페이스로 IP 주소 172.31.16.139의 SSH 트래픽(대상 포트 22, TCP 프로토콜)이 허용되었습니다.

```
2 123456789010 eni-1235b8ca123456789 172.31.16.139 172.31.16.21 20641 22 6 20 4249
1418530010 1418530070 ACCEPT OK
```

이 예제에서는 계정 123456789010에서 네트워크 인터페이스 eni-1235b8ca123456789에 대한 RDP 트래픽(대상 포트 3389, TCP 프로토콜)이 거부되었습니다.

```
2 123456789010 eni-1235b8ca123456789 172.31.9.69 172.31.9.12 49761 3389 6 20 4249
1418530010 1418530070 REJECT OK
```

데이터가 없고 건너뛴 레코드

다음은 기본 흐름 로그 레코드의 예입니다.

이 예에서는 집계 간격 동안 데이터가 기록되지 않았습니다.

```
2 123456789010 eni-1235b8ca123456789 - - - - - 1431280876 1431280934 - NODATA
```

이 예에서는 집계 간격 동안 레코드를 건너뛰었습니다. VPC 흐름 로그는 내부 용량 초과로 집계 간격 동안 흐름 로그 데이터를 캡처할 수 없는 경우 레코드를 건너뛵니다. 건너뛴 단일 레코드는 집계 간격 중 네트워크 인터페이스에 대해 캡처되지 않은 여러 흐름을 나타낼 수 있습니다.

```
2 123456789010 eni-1111111111aaaaaaaa - - - - - 1431280876 1431280934 - SKIPDATA
```

보안 그룹 및 네트워크 ACL 규칙

너무 제한적이거나 허용적인 보안 그룹 규칙 또는 네트워크 ACL 규칙을 진단하기 위해 흐름 로그를 사용할 경우 이러한 리소스의 상태 저장 여부를 알아야 합니다. 보안 그룹은 상태가 저장됩니다. 보안 그룹의 규칙에서 허용하지 않더라도 허용된 트래픽에 응답할 수 있다는 뜻입니다. 반대로 네트워크 ACL은 상태를 저장하지 않으므로 허용된 트래픽에 대한 응답은 네트워크 ACL 규칙을 따릅니다.

예를 들어 홈 컴퓨터(IP 주소: 203.0.113.12)에서 인스턴스(네트워크 인터페이스의 프라이빗 IP 주소: 172.31.16.139)로 ping 명령을 사용합니다. 보안 그룹의 인바운드 규칙은 ICMP 트래픽을 허용하지만 아웃바운드 규칙은 ICMP 트래픽을 허용하지 않습니다. 보안 그룹은 상태 저장이므로 인스턴스의 응답 ping이 허용됩니다. 네트워크 ACL은 인바운드 ICMP 트래픽을 허용하지만 아웃바운드 ICMP 트래픽은 허용하지 않습니다. 왜냐하면 네트워크 ACL은 상태를 저장하지 않아서 응답 ping이 홈 컴퓨터에 도달하지 않기 때문입니다. 이는 기본 흐름 로그에서 다음과 같은 2가지 흐름 로그 레코드로 표시됩니다.

- 네트워크 ACL과 보안 그룹이 모두 허용했으며 따라서 인스턴스에 접속하도록 허용된 요청 ping에 대한 ACCEPT 레코드

- 네트워크 ACL이 거부한 응답 ping에 대한 REJECT 레코드

```
2 123456789010 eni-1235b8ca123456789 203.0.113.12 172.31.16.139 0 0 1 4 336 1432917027
1432917142 ACCEPT OK
```

```
2 123456789010 eni-1235b8ca123456789 172.31.16.139 203.0.113.12 0 0 1 4 336 1432917094
1432917142 REJECT OK
```

네트워크 ACL이 아웃바운드 ICMP 트래픽을 허용한 경우, 흐름 로그에 두 가지 ACCEPT 레코드(하나는 요청 ping에 대한 레코드, 다른 하나는 응답 ping에 대한 레코드)가 표시됩니다. 보안 그룹이 인바운드 ICMP 트래픽을 거부한 경우, 흐름 로그에는 하나의 REJECT 레코드만 표시됩니다. 해당 트래픽이 인스턴스에 접속하도록 허용되지 않았기 때문입니다.

IPv6 트래픽

다음은 기본 흐름 로그 레코드의 예입니다. 이 예제에서는 계정 123456789010에서, IPv6 주소 2001:db8:1234:a100:8d6e:3477:df66:f105로부터 네트워크 인터페이스 eni-1235b8ca123456789로의 SSH 트래픽(포트 22)이 허용되었습니다.

```
2 123456789010 eni-1235b8ca123456789 2001:db8:1234:a100:8d6e:3477:df66:f105
2001:db8:1234:a102:3304:8879:34cf:4071 34892 22 6 54 8855 1477913708 1477913820 ACCEPT
OK
```

TCP 플래그 시퀀스

이 섹션에는 아래와 같은 순서로 다음 필드를 캡처하는 사용자 지정 흐름 로그의 예가 포함되어 있습니다.

```
version vpc-id subnet-id instance-id interface-id account-id type srcaddr dstaddr
srcport dstport pkt-srcaddr pkt-dstaddr protocol bytes packets start end action tcp-
flags log-status
```

이 섹션에서 예제의 tcp-flags 필드는 흐름 로그의 마지막 두 번째 값으로 표시됩니다. TCP 플래그는 트래픽의 방향(예: 연결을 시작한 서버)을 식별하는 데 도움이 됩니다.

Note

tcp-flags 옵션에 대한 자세한 내용 및 각 TCP 플래그에 대한 설명은 [사용 가능한 필드의 내용](#)을 참조하십시오.

다음 레코드(오후 7:47:55 오후에 시작하고 오후 7:48:53에 끝남)에서는 클라이언트가 포트 5001에서 실행 중인 서버에 대한 두 개의 연결을 시작했습니다. 클라이언트의 다른 소스 포트(43416 및 43418)에서 서버가 두 개의 SYN 플래그(2)를 수신했습니다. 각 SYN에 대해 SYN-ACK가 서버에서 해당 포트의 클라이언트(18)로 전송되었습니다.

```
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
eni-1235b8ca123456789 123456789010 IPv4 52.213.180.42 10.0.0.62 43416 5001
52.213.180.42 10.0.0.62 6 568 8 1566848875 1566848933 ACCEPT 2 OK
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
eni-1235b8ca123456789 123456789010 IPv4 10.0.0.62 52.213.180.42 5001 43416 10.0.0.62
52.213.180.42 6 376 7 1566848875 1566848933 ACCEPT 18 OK
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
eni-1235b8ca123456789 123456789010 IPv4 52.213.180.42 10.0.0.62 43418 5001
52.213.180.42 10.0.0.62 6 100701 70 1566848875 1566848933 ACCEPT 2 OK
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
eni-1235b8ca123456789 123456789010 IPv4 10.0.0.62 52.213.180.42 5001 43418 10.0.0.62
52.213.180.42 6 632 12 1566848875 1566848933 ACCEPT 18 OK
```

두 번째 집계 간격에서 이전 흐름 중에 설정된 연결 중 하나가 닫힙니다. 클라이언트는 포트 43418 연결을 위해 FIN 플래그(1)를 서버로 보냈습니다. 서버가 43418 포트에서 클라이언트로 FIN을 보냈습니다.

```
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
eni-1235b8ca123456789 123456789010 IPv4 10.0.0.62 52.213.180.42 5001 43418 10.0.0.62
52.213.180.42 6 63388 1219 1566848933 1566849113 ACCEPT 1 OK
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
eni-1235b8ca123456789 123456789010 IPv4 52.213.180.42 10.0.0.62 43418 5001
52.213.180.42 10.0.0.62 6 23294588 15774 1566848933 1566849113 ACCEPT 1 OK
```

단일 집계 간격 내에서 열리고 닫히는 짧은 연결(예: 몇 초)의 경우 동일한 방향으로 트래픽 흐름을 위해 흐름 로그 레코드에서 같은 줄에 플래그가 설정될 수 있습니다. 다음 예제에서는 동일한 집계 간격 내에서 연결이 설정되고 완료됩니다. 첫 번째 줄에서 TCP 플래그 값은 3입니다. 이는 클라이언트에서 서버로 전송된 SYN 및 FIN 메시지가 있음을 나타냅니다. 두 번째 줄에서 TCP 플래그 값은 19입니다. 이는 서버에서 클라이언트로 전송된 SYN-ACK 및 FIN 메시지가 있음을 나타냅니다.

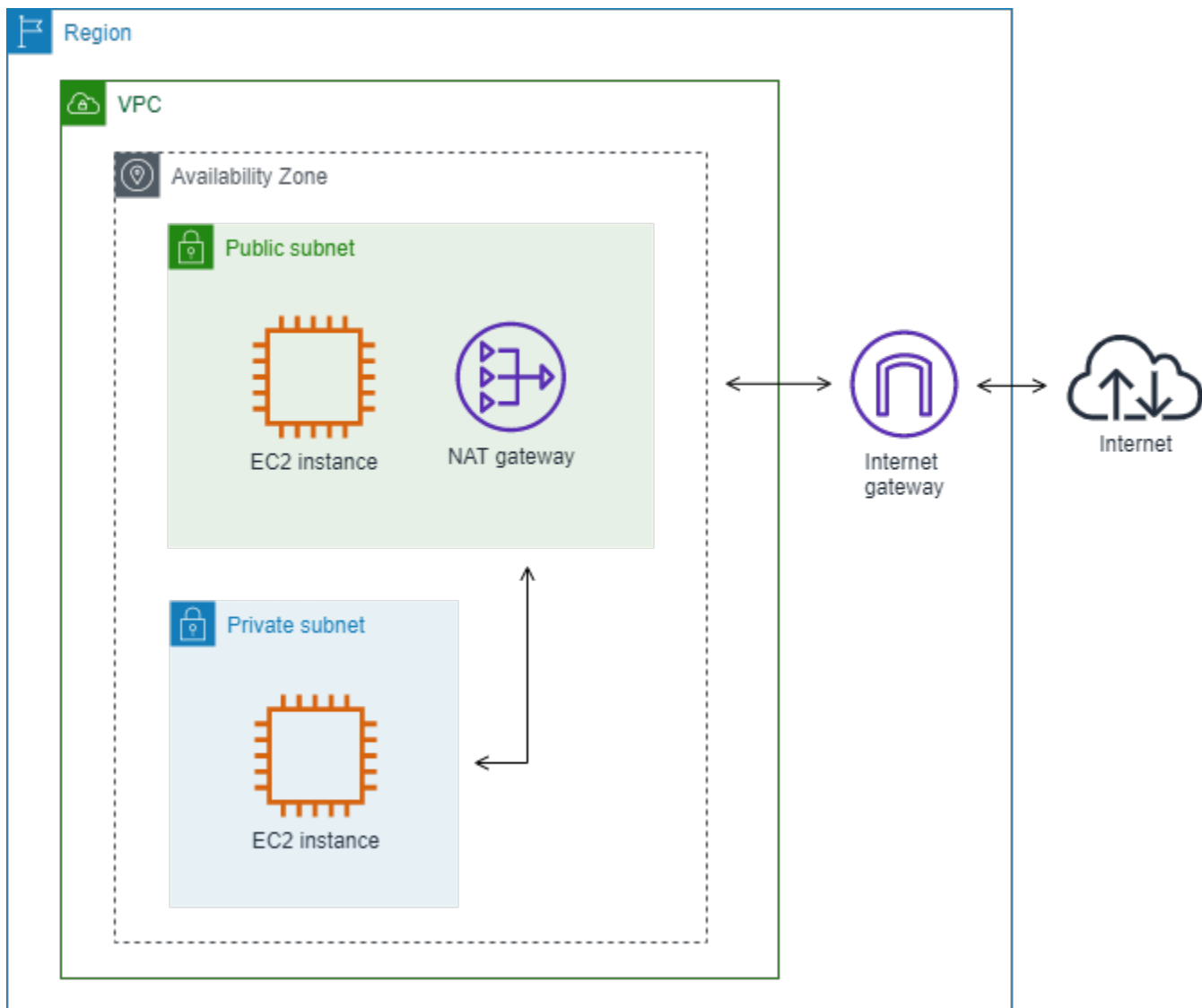
```

3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
eni-1235b8ca123456789 123456789010 IPv4 52.213.180.42 10.0.0.62 43638 5001
52.213.180.42 10.0.0.62 6 1260 17 1566933133 1566933193 ACCEPT 3 OK
3 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-01234567890123456
eni-1235b8ca123456789 123456789010 IPv4 10.0.0.62 52.213.180.42 5001 43638 10.0.0.62
52.213.180.42 6 967 14 1566933133 1566933193 ACCEPT 19 OK

```

NAT 게이트웨이를 통한 트래픽

이 예제에서 프라이빗 서브넷의 인스턴스는 퍼블릭 서브넷에 있는 NAT 게이트웨이를 통해 인터넷에 액세스합니다.



NAT 게이트웨이 네트워크 인터페이스에 대한 다음 사용자 정의 흐름 로그는 다음 필드를 다음 순서로 캡처합니다.


```
instance-id interface-id srcaddr dstaddr pkt-srcaddr pkt-dstaddr
```

흐름 로그는 NAT 게이트웨이 네트워크 인터페이스를 통해 인스턴스 IP 주소(10.0.1.5)에서 인터넷의 호스트(203.0.113.5)로의 트래픽 흐름을 보여줍니다. NAT 게이트웨이 네트워크 인터페이스는 요청자 관리 네트워크 인터페이스이므로 흐름 로그 레코드는 instance-id 필드에 '-' 기호를 표시합니다. 다음 줄은 소스 인스턴스에서 NAT 게이트웨이 네트워크 인터페이스로의 트래픽을 보여줍니다. dstaddr 및 pkt-dstaddr 필드의 값은 다릅니다. dstaddr 필드에는 NAT 게이트웨이 네트워크 인터페이스의 프라이빗 IP 주소가 표시되고, pkt-dstaddr 필드에는 인터넷에 있는 호스트의 최종 대상 IP 주소가 표시됩니다.

```
- eni-1235b8ca123456789 10.0.1.5 10.0.0.220 10.0.1.5 203.0.113.5
```

다음 두 줄은 NAT 게이트웨이 네트워크 인터페이스에서 인터넷의 대상 호스트로의 트래픽과 호스트에서 NAT 게이트웨이 네트워크 인터페이스로의 응답 트래픽을 보여줍니다.

```
- eni-1235b8ca123456789 10.0.0.220 203.0.113.5 10.0.0.220 203.0.113.5
- eni-1235b8ca123456789 203.0.113.5 10.0.0.220 203.0.113.5 10.0.0.220
```

다음 줄은 NAT 게이트웨이 네트워크 인터페이스에서 소스 인스턴스로의 응답 트래픽을 보여줍니다. srcaddr 및 pkt-srcaddr 필드의 값은 다릅니다. srcaddr 필드에는 NAT 게이트웨이 네트워크 인터페이스의 프라이빗 IP 주소가 표시되고, pkt-srcaddr 필드에는 인터넷에 있는 호스트의 IP 주소가 표시됩니다.

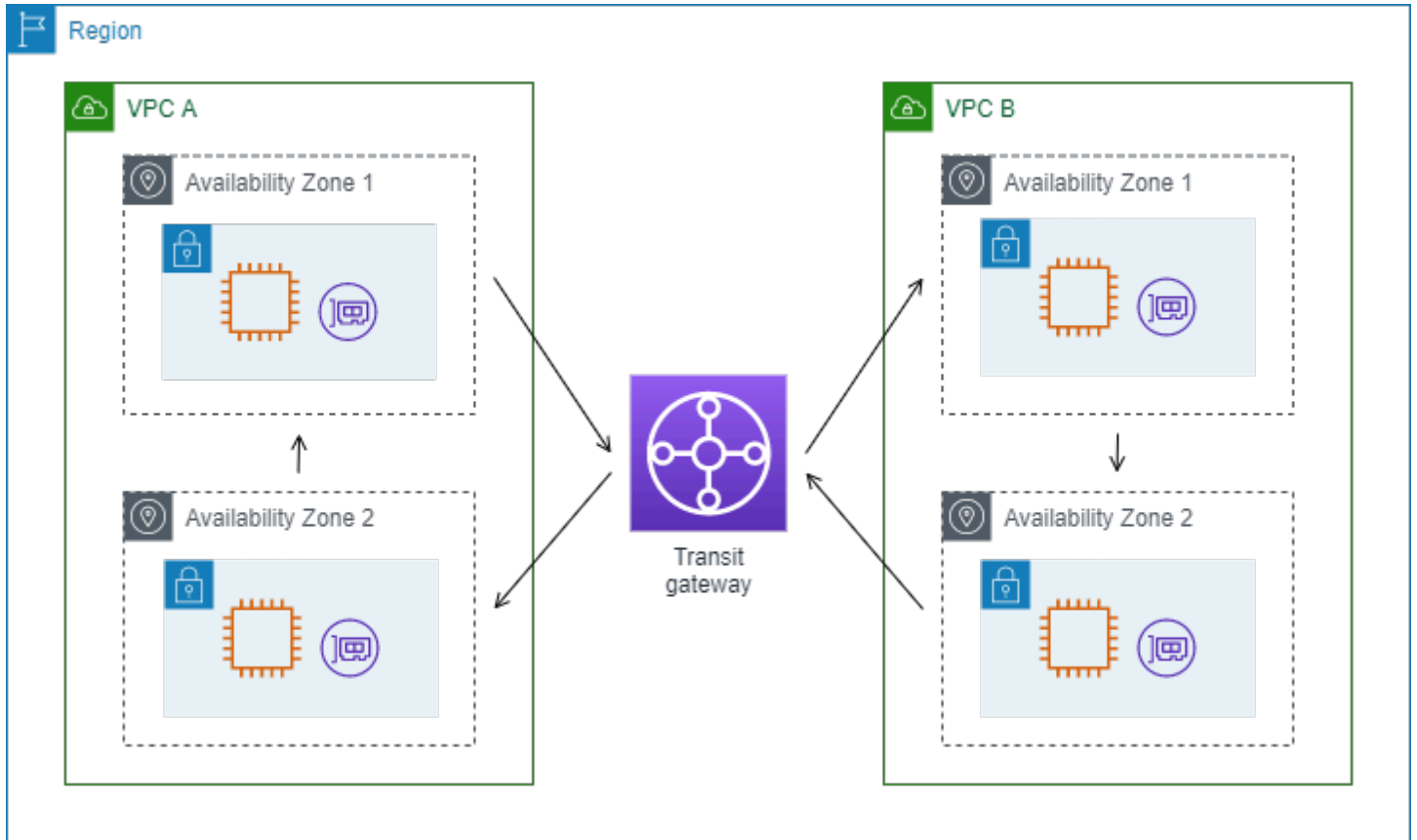
```
- eni-1235b8ca123456789 10.0.0.220 10.0.1.5 203.0.113.5 10.0.1.5
```

위와 동일한 필드 세트를 사용하여 다른 사용자 정의 흐름 로그를 작성합니다. 프라이빗 서브넷에서 인스턴스의 네트워크 인터페이스에 대한 흐름 로그를 생성합니다. 이 경우 instance-id 필드는 네트워크 인터페이스와 연결된 인스턴스의 ID를 반환하며, dstaddr 및 pkt-dstaddr 필드와 srcaddr 및 pkt-srcaddr 필드 사이에는 차이가 없습니다. NAT 게이트웨이의 네트워크 인터페이스와 달리 이 네트워크 인터페이스는 트래픽의 중간 네트워크 인터페이스가 아닙니다.

```
i-01234567890123456 eni-1111aaaa2222bbbb3 10.0.1.5 203.0.113.5 10.0.1.5 203.0.113.5
#Traffic from the source instance to host on the internet
i-01234567890123456 eni-1111aaaa2222bbbb3 203.0.113.5 10.0.1.5 203.0.113.5 10.0.1.5
#Response traffic from host on the internet to the source instance
```

전송 게이트웨이를 통한 트래픽

이 예에서 VPC A의 클라이언트는 전송 게이트웨이를 통해 VPC B의 웹 서버에 연결합니다. 클라이언트와 서버가 서로 다른 가용 영역에 있습니다. 트래픽은 탄력적 네트워크 인터페이스 ID 하나 (이 예에서 ID가 eni-11111111111111111임)를 사용하여 VPC B의 서버에 도착하고 다른 ID(예: eni-22222222222222222)를 사용하여 VPC B를 떠납니다.



다음 형식으로 VPC B에 대한 사용자 지정 흐름 로그를 생성합니다.

```
version interface-id account-id vpc-id subnet-id instance-id srcaddr dstaddr srcport
dstport protocol tcp-flags type pkt-srcaddr pkt-dstaddr action log-status
```

흐름 로그 레코드의 다음 줄은 웹 서버의 네트워크 인터페이스에서의 트래픽 흐름을 보여줍니다. 첫 번째 줄은 클라이언트의 요청 트래픽이고 마지막 줄은 웹 서버의 응답 트래픽입니다.

```
3 eni-33333333333333333 123456789010 vpc-abcdefab012345678 subnet-22222222bbbbbbbbb
i-01234567890123456 10.20.33.164 10.40.2.236 39812 80 6 3 IPv4 10.20.33.164
10.40.2.236 ACCEPT OK
...
```

```
3 eni-333333333333333333 123456789010 vpc-abcdefab012345678 subnet-22222222bbbbbbbbb
i-01234567890123456 10.40.2.236 10.20.33.164 80 39812 6 19 IPv4 10.40.2.236
10.20.33.164 ACCEPT OK
```

다음 줄은 subnet-11111111aaaaaaaa 서브넷의 전송 게이트웨이에 대한 요청자 관리 네트워크 인터페이스인 eni-11111111111111111111에서의 요청 트래픽입니다. 흐름 로그 레코드는 instance-id 필드에 '-' 기호를 표시합니다. srcaddr 필드에는 전송 게이트웨이 네트워크 인터페이스의 프라이빗 IP 주소가 표시되고, pkt-srcaddr 필드에는 VPC A의 클라이언트의 원본 IP 주소가 표시됩니다.

```
3 eni-111111111111111111 123456789010 vpc-abcdefab012345678 subnet-11111111aaaaaaaa -
10.40.1.175 10.40.2.236 39812 80 6 3 IPv4 10.20.33.164 10.40.2.236 ACCEPT OK
```

다음 줄은 subnet-22222222bbbbbbbbb 서브넷의 전송 게이트웨이에 대한 요청자 관리 네트워크 인터페이스인 eni-22222222222222222222에서의 응답 트래픽입니다. dstaddr 필드에는 전송 게이트웨이 네트워크 인터페이스의 프라이빗 IP 주소가 표시되고, pkt-dstaddr 필드에는 VPC A의 클라이언트의 IP 주소가 표시됩니다.

```
3 eni-222222222222222222 123456789010 vpc-abcdefab012345678 subnet-22222222bbbbbbbbb -
10.40.2.236 10.40.2.31 80 39812 6 19 IPv4 10.40.2.236 10.20.33.164 ACCEPT OK
```

서비스 이름, 트래픽 경로 및 흐름 방향

다음은 사용자 지정 흐름 로그 레코드의 필드 예제입니다.

```
version srcaddr dstaddr srcport dstport protocol start end type packets bytes account-
id vpc-id subnet-id instance-id interface-id region az-id sublocation-type sublocation-
id action tcp-flags pkt-srcaddr pkt-dstaddr pkt-src-aws-service pkt-dst-aws-service
traffic-path flow-direction log-status
```

다음 예제에서는 레코드에 버전 5 필드가 포함되어 있으므로 버전이 5입니다. EC2 인스턴스가 Amazon S3 서비스를 호출합니다. 흐름 로그는 인스턴스의 네트워크 인터페이스에서 캡처됩니다. 첫 번째 레코드의 흐름 방향은 ingress이고 두 번째 레코드의 흐름 방향은 egress입니다. egress 레코드의 traffic-path는 8로, 트래픽이 인터넷 게이트웨이를 통해 전송된다는 것을 나타냅니다. traffic-path 트래픽에 대해서는 ingress 필드가 지원되지 않습니다. pkt-srcaddr 또는 pkt-dstaddr이 퍼블릭 IP 주소인 경우 서비스 이름이 표시됩니다.

```
5 52.95.128.179 10.0.0.71 80 34210 6 1616729292 1616729349 IPv4 14 15044
123456789012 vpc-abcdefab012345678 subnet-aaaaaaaa012345678 i-0c50d5961bcb2d47b
```

```
eni-1235b8ca123456789 ap-southeast-2 apse2-az3 - - ACCEPT 19 52.95.128.179 10.0.0.71
S3 - - ingress OK
5 10.0.0.71 52.95.128.179 34210 80 6 1616729292 1616729349 IPv4 7 471 123456789012 vpc-
abcdefab012345678 subnet-aaaaaaaa012345678 i-0c50d5961bcb2d47b eni-1235b8ca123456789
ap-southeast-2 apse2-az3 - - ACCEPT 3 10.0.0.71 52.95.128.179 - S3 8 egress OK
```

흐름 로그 제한

흐름 로그를 사용하려면 다음과 같은 제한 사항을 알아 두어야 합니다.

- 피어 VPC가 본인의 계정이 아닌 한, 본인의 VPC와 피어링된 VPC에 대해 흐름 로그를 활성화할 수 없습니다.
- 흐름 로그를 만든 후에는 구성 또는 흐름 로그 레코드 형식을 변경할 수 없습니다. 예를 들어 다른 IAM 역할을 흐름 로그와 연결하거나 흐름 로그 레코드에서 필드를 추가 또는 제거할 수 없습니다. 대신에 흐름 로그를 삭제한 후 필요한 구성으로 새로운 흐름 로그를 생성할 수 있습니다.
- 네트워크 인터페이스에 IPv4 주소가 여러 개 있고 트래픽이 보조 프라이빗 IPv4 주소로 전송되는 경우, 흐름 로그는 `dstaddr` 필드에 주 프라이빗 IPv4 주소를 표시합니다. 원래 대상 IP 주소를 캡처하려면 `pkt-dstaddr` 필드로 흐름 로그를 작성하십시오.
- 트래픽이 네트워크 인터페이스로 전송된 경우 대상이 네트워크 인터페이스의 IP 주소가 아니면 흐름 로그에 `dstaddr` 필드의 기본 프라이빗 IPv4 주소가 표시됩니다. 원래 대상 IP 주소를 캡처하려면 `pkt-dstaddr` 필드로 흐름 로그를 작성하십시오.
- 트래픽이 네트워크 인터페이스에서 전송된 경우 원본이 네트워크 인터페이스의 IP 주소가 아니면, 흐름 로그에 `srcaddr` 필드의 기본 프라이빗 IPv4 주소가 표시됩니다. 원래 원본 IP 주소를 캡처하려면 `pkt-srcaddr` 필드로 흐름 로그를 작성하십시오.
- 트래픽이 네트워크 인터페이스를 오가는 경우 흐름 로그의 `srcaddr`, `dstaddr` 필드는 패킷 원본 또는 대상에 관계없이 항상 기본 프라이빗 IPv4 주소를 표시합니다. 패킷 소스 또는 대상을 캡처하려면 `pkt-srcaddr` 및 `pkt-dstaddr` 필드를 사용하여 흐름 로그를 작성하십시오.
- 네트워크 인터페이스가 [NITRO 기반 인스턴스](#)에 연결된 경우 집계 간격은 지정된 최대 집계 간격에 관계없이 항상 1분 이하입니다.

흐름 로그는 모든 IP 트래픽을 캡처하지는 않습니다. 다음 트래픽 유형은 기록되지 않습니다.

- 인스턴스가 Amazon DNS 서버에 연결할 때 생성한 트래픽. 자체 DNS 서버를 사용할 경우 DNS 서버에 대한 모든 트래픽은 기록됩니다.
- Amazon Windows 라이선스 인증을 위해 Windows 인스턴스에서 생성한 트래픽.
- 인스턴스 메타데이터를 위해 169.254.169.254와 주고받는 트래픽.

- Amazon Time Sync Service를 위해 169.254.169.123와 주고받는 트래픽.
- DHCP 트래픽.
- 미러링된 트래픽.
- 기본 VPC 라우터의 예약된 IP 주소로 보내는 트래픽.
- 엔드포인트 네트워크 인터페이스와 Network Load Balancer 네트워크 인터페이스 간의 트래픽.

버전 7에서 사용할 수 있는 ECS 필드에만 적용되는 제한 사항:

- ECS 필드로 흐름 로그 구독을 생성하려면 계정에 하나 이상의 ECS 클러스터가 포함되어 있어야 합니다.
- 흐름 로그 구독의 소유자가 기본 ECS 태스크를 소유하지 않은 경우 ECS 필드는 계산되지 않습니다. 예를 들어, 서브넷(SubnetA)을 다른 계정(AccountB)과 공유한 다음 SubnetA에 대한 흐름 로그 구독을 생성하는 경우 AccountB가 공유 서브넷에서 ECS 태스크를 시작하면 구독은 AccountB에서 시작한 ECS 태스크에서 트래픽 로그를 수신하지만 보안 문제로 인해 이러한 로그의 ECS 필드는 계산되지 않습니다.
- VPC/서브넷 리소스 수준에서 ECS 필드를 사용하여 흐름 로그 구독을 생성하는 경우 비ECS 네트워크 인터페이스에 대해 생성된 모든 트래픽도 구독에 전달됩니다. 비ECS IP 트래픽의 경우 ECS 필드 값은 '-'가 됩니다. 예를 들어 서브넷(subnet-000000)이 있고 ECS 필드(f1-00000000)로 이 서브넷에 대한 흐름 로그 구독을 생성합니다. subnet-000000에서 인터넷에 연결되어 있고 IP 트래픽을 활발하게 생성하는 EC2 인스턴스(i-00000000)를 시작합니다. 또한 동일한 서브넷에서 실행 중인 ECS 태스크(ECS-Task-1)를 시작합니다. i-00000000과 ECS-Task-1 모두 IP 트래픽을 생성하므로 흐름 로그 구독 f1-00000000은 두 엔터티 모두에 대한 트래픽 로그를 제공합니다. 그러나 ECS-Task-1만이 logFormat에 포함된 ECS 필드에 대한 실제 ECS 메타데이터를 갖습니다. i-00000000 관련 트래픽의 경우 이러한 필드의 값은 '-'가 됩니다.
- ecs-container-id와 ecs-second-container-id는 VPC 흐름 로그 서비스가 ECS 이벤트 스트림에서 수신할 때 순서가 지정됩니다. ECS 콘솔 또는 DescribeTask API 직접 호출에서 볼 수 있는 순서와 동일하다는 보장은 없습니다. 태스크가 계속 실행되는 동안 컨테이너가 STOPPED 상태가 되면 로그에 계속 표시될 수 있습니다.
- ECS 메타데이터와 IP 트래픽 로그는 서로 다른 두 소스에서 가져온 것입니다. 업스트림 종속성에서 필요한 모든 정보를 얻는 즉시 ECS 트래픽 계산이 시작됩니다. 새 태스크를 시작하면 1) 기본 네트워크 인터페이스에 대한 IP 트래픽이 수신되는 경우와 2) ECS 태스크가 현재 실행 중임을 나타내는 태스크에 대한 메타데이터가 포함된 ECS 이벤트가 수신되는 경우 ECS 필드 계산이 시작됩니다. 태스크를 중지하면 1) 기본 네트워크 인터페이스에 대한 IP 트래픽이 더 이상 수신되지 않거나 하루 이상 지연되는 IP 트래픽이 수신되는 경우와 2) ECS 태스크가 더 이상 실행되지 않음을 나타내는 태스크에 대한 메타데이터가 포함된 ECS 이벤트가 수신되는 경우 ECS 필드 계산이 중지됩니다.

- awsvpc [네트워크](#) 모드에서 시작된 ECS 태스크만 지원됩니다.

요금

흐름 로그를 게시하면 벤딩 로그에 대한 데이터 모으기 및 보관 요금이 적용됩니다. 벤딩 로그를 게시할 때 요금에 대해 자세히 알아보려면 [Amazon CloudWatch Pricing](#)(Amazon CloudWatch 요금)을 열고 Logs(로그)를 선택하고 Vended Logs(벤딩 로그)를 찾으세요.

흐름 로그 게시의 요금을 추적하려는 경우 대상 리소스에 비용 할당 태그를 적용할 수 있습니다. 이후에 AWS 비용 할당 보고서에 이러한 태그로 집계된 사용량 및 비용이 포함됩니다. 비즈니스 범주를 나타내는 태그(예: 비용 센터, 애플리케이션 이름 또는 소유자)를 적용하여 비용을 정리할 수 있습니다. 자세한 내용은 다음 자료를 참조하세요.

- AWS Billing 사용 설명서의 [비용 할당 태그 사용](#)
- Amazon CloudWatch Logs 사용 설명서의 [Amazon CloudWatch Logs의 로그 그룹에 태그 지정](#)
- Amazon Simple Storage Service 사용 설명서의 [비용 할당 S3 버킷 태그 사용](#)
- Amazon Data Firehose 개발자 안내서의 [전송 스트림 태그 지정](#)

흐름 로그 작업

Amazon EC2 및 Amazon VPC의 콘솔을 사용하여 흐름 로그를 연동할 수 있습니다.

Tasks

- [흐름 로그 사용 제어](#)
- [흐름 로그 생성](#)
- [흐름 로그 보기](#)
- [흐름 로그 태그 지정](#)
- [흐름 로그 삭제](#)
- [API 및 CLI 개요](#)

흐름 로그 사용 제어

기본적으로 사용자에게는 흐름 로그 사용 권한이 없습니다. 사용자에게 흐름 로그를 생성, 설명, 삭제하는 권한을 부여하는 정책이 연결된 IAM 역할을 만들 수 있습니다.

다음은 사용자에게 흐름 로그를 생성, 설명 및 삭제할 수 있는 전체 권한을 부여하는 정책의 예입니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DeleteFlowLogs",
        "ec2:CreateFlowLogs",
        "ec2:DescribeFlowLogs"
      ],
      "Resource": "*"
    }
  ]
}
```

자세한 내용은 [the section called “Amazon VPC가 IAM과 작동하는 방식” 단원을 참조하십시오.](#)

흐름 로그 생성

VPC, 서브넷 또는 네트워크 인터페이스에 대한 흐름 로그를 생성할 수 있습니다. 흐름 로그를 생성할 때 흐름 로그의 대상을 지정해야 합니다. 자세한 내용은 다음 자료를 참조하세요.

- [the section called “CloudWatch Logs에 게시하는 흐름 로그 생성”](#)
- [the section called “Amazon S3에 게시하는 흐름 로그 생성”](#)
- [the section called “Amazon Data Firehose에 게시하는 흐름 로그 생성”](#)

흐름 로그 보기

리소스의 흐름 로그(예: 네트워크 인터페이스)에 대한 정보를 볼 수 있습니다. 흐름 로그의 ID, 흐름 로그 구성, 흐름 로그 상태에 대한 정보 등이 표시됩니다.

흐름 로그에 대한 정보를 보는 방법

1. 다음 중 하나를 수행하십시오.
 - <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다. 탐색 창에서 네트워크 인터페이스(Network Interfaces)를 선택합니다. 네트워크 인터페이스의 확인란을 선택합니다.

- <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 Your VPCs(사용자 VPC)를 선택합니다. VPC에 대한 확인란을 선택합니다.
 - <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 서브넷(Subnets)을 선택합니다. 서브넷의 확인란을 선택합니다.
2. Flow Logs(흐름 로그)를 선택합니다.
 3. (선택 사항) 흐름 로그 데이터를 보려면 로그 대상을 엽니다.

흐름 로그 태그 지정

언제든지 흐름 로그의 태그를 추가하거나 제거할 수 있습니다.

흐름 로그의 태그를 관리하는 방법

1. 다음 중 하나를 수행하십시오.
 - <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다. 탐색 창에서 네트워크 인터페이스(Network Interfaces)를 선택합니다. 네트워크 인터페이스의 확인란을 선택합니다.
 - <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 Your VPCs(사용자 VPC)를 선택합니다. VPC에 대한 확인란을 선택합니다.
 - <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 서브넷(Subnets)을 선택합니다. 서브넷의 확인란을 선택합니다.
2. Flow Logs(흐름 로그)를 선택합니다.
3. 작업(Actions), 태그 관리(Manage tags)를 선택합니다.
4. 태그를 추가하려면 Add new tag(새 태그 추가)를 선택하고 키와 값을 입력합니다. 태그를 제거하려면 제거를 선택합니다.
5. 태그 추가 또는 제거를 마쳤으면 Save(저장)를 선택합니다.

흐름 로그 삭제

언제든지 흐름 로그를 삭제할 수 있습니다. 흐름 로그를 삭제하면 데이터 수집 중단까지 몇 분 정도 걸릴 수 있습니다.

흐름 로그를 삭제해도 대상의 로그 데이터가 삭제되거나 대상 리소스가 수정되지 않습니다. 대상 서비스의 콘솔을 사용하여 기존 흐름 로그 데이터를 대상에서 직접 삭제하고 대상 리소스를 정리해야 합니다.

흐름 로그를 삭제하는 방법

1. 다음 중 하나를 수행하십시오.
 - <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다. 탐색 창에서 네트워크 인터페이스(Network Interfaces)를 선택합니다. 네트워크 인터페이스의 확인란을 선택합니다.
 - <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 Your VPCs(사용자 VPC)를 선택합니다. VPC에 대한 확인란을 선택합니다.
 - <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 서브넷(Subnets)을 선택합니다. 서브넷의 확인란을 선택합니다.
2. Flow Logs(흐름 로그)를 선택합니다.
3. Actions(작업), Delete flow logs(흐름 로그 삭제)를 선택합니다.
4. 확인 메시지가 나타나면 **delete**를 입력한 다음 삭제>Delete)를 선택합니다.

API 및 CLI 개요

이 페이지에서 설명한 작업은 명령줄이나 API를 사용하여 수행할 수 있습니다. 명령줄 인터페이스 및 사용 가능한 API 작업 목록에 대한 자세한 내용은 [Amazon VPC 작업](#) 단원을 참조하십시오.

흐름 로그 생성

- [create-flow-logs](#)(AWS CLI)
- [New-EC2FlowLog](#)(AWS Tools for Windows PowerShell)
- [CreateFlowLogs](#)(Amazon EC2 쿼리 API)

흐름 로그 설명

- [describe-flow-logs](#)(AWS CLI)
- [Get-EC2FlowLog](#)(AWS Tools for Windows PowerShell)
- [DescribeFlowLogs](#)(Amazon EC2 쿼리 API)

흐름 로그 태그 지정

- [create-tags](#) 및 [delete-tags](#)(AWS CLI)
- [New-EC2Tag](#) 및 [Remove-EC2Tag](#)(AWS Tools for Windows PowerShell)

- [CreateTags](#) 및 [DeleteTags](#)(Amazon EC2 쿼리 API)

흐름 로그 삭제

- [delete-flow-logs](#)(AWS CLI)
- [Remove-EC2FlowLog](#)(AWS Tools for Windows PowerShell)
- [DeleteFlowLogs](#)(Amazon EC2 쿼리 API)

CloudWatch Logs에 흐름 로그 게시

흐름 로그는 흐름 로그 데이터를 Amazon CloudWatch에 직접 게시할 수 있습니다.

CloudWatch Logs에 게시하는 경우 흐름 로그 데이터는 로그 그룹에 게시되고, 각 네트워크 인터페이스는 로그 그룹에 고유의 로그 스트림을 가집니다. 로그 스트림에는 흐름 로그 레코드가 포함됩니다. 여러 개의 흐름 로그를 생성하여, 그 데이터를 같은 로그 그룹에 게시할 수 있습니다. 같은 로그 그룹의 하나 이상의 흐름 로그에 동일한 네트워크 인터페이스가 있을 경우 로그 스트림은 하나로 병합됩니다. 한 흐름 로그에서는 거부된 트래픽을 캡처하고, 다른 흐름 로그에서는 허용된 트래픽을 캡처하도록 지정한 경우, 병합된 로그 스트림은 모든 트래픽을 캡처합니다.

CloudWatch Logs에서 timestamp 필드는 흐름 로그 레코드에서 캡처된 시작 시간에 해당합니다. ingestionTime 필드는 CloudWatch Logs에서 흐름 로그 레코드가 수신된 날짜와 시간을 나타냅니다. 이 타임스탬프는 흐름 로그 레코드에 캡처된 종료 시간보다 이후입니다.

CloudWatch Logs에 대한 자세한 내용은 [Amazon CloudWatch Logs 사용 설명서](#)의 CloudWatch Logs로 전송된 로그를 참조하세요.

요금

CloudWatch Logs에 흐름 로그를 게시할 때는 Vended 로그에 대한 데이터 수집 및 아카이브 요금이 부과됩니다. 자세히 알아보려면 [Amazon CloudWatch Pricing](#)(Amazon CloudWatch 요금)을 열고, Logs(로그)를 선택하고, Vended Logs(벤딩 로그)를 찾으세요.

내용

- [CloudWatch Logs에 흐름 로그를 게시하는 IAM 역할](#)
- [CloudWatch Logs에 흐름 로그를 게시하는 IAM 보안 주체의 권한](#)
- [CloudWatch Logs에 게시하는 흐름 로그 생성](#)
- [흐름 로그 레코드 보기](#)

- [흐름 로그 레코드 검색](#)
- [CloudWatch Logs에서 흐름 로그 레코드 처리](#)

CloudWatch Logs에 흐름 로그를 게시하는 IAM 역할

흐름 로그와 연결된 IAM 역할에는 CloudWatch Logs의 지정된 로그 그룹에 흐름 로그를 게시할 권한이 있어야 합니다. IAM 역할은 AWS 계정에 속해야 합니다.

IAM 역할에 연결된 IAM 정책에는 최소한 다음과 같은 권한이 포함되어야 합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "logs:DescribeLogGroups",
        "logs:DescribeLogStreams"
      ],
      "Resource": "*"
    }
  ]
}
```

흐름 로그 서비스에서 역할을 수임할 수 있는 다음과 같은 신뢰 정책이 역할에 있는지 확인하세요.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "vpc-flow-logs.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

혼동된 대리자 문제로부터 자신을 보호하기 위하여 `aws:SourceAccount` 및 `aws:SourceArn` 조건 키를 사용할 것을 권장합니다. 예를 들어 이전 신뢰 정책에 다음 조건 블록을 추가할 수 있습니다. 소스 계정은 흐름 로그의 소유자이고 소스 ARN은 흐름 로그 ARN입니다. 흐름 로그 ID를 모르는 경우 ARN의 해당 부분을 와일드카드(*)로 바꾼 다음 흐름 로그를 만든 후 정책을 업데이트할 수 있습니다.

```
"Condition": {
  "StringEquals": {
    "aws:SourceAccount": "account_id"
  },
  "ArnLike": {
    "aws:SourceArn": "arn:aws:ec2:region:account_id:vpc-flow-log/flow-log-id"
  }
}
```

흐름 로그에 대한 IAM 역할 생성

위에 설명된 대로 기존 역할을 업데이트할 수 있습니다. 또는 다음과 같은 절차에 따라 흐름 로그에서 사용할 새 역할을 생성할 수 있습니다. 이 역할은 흐름 로그를 생성할 때 지정합니다.

흐름 로그에 대한 IAM 역할 생성

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 정책을 선택합니다.
3. [정책 생성(Create policy)]을 선택합니다.
4. [Create policy(정책 생성)] 페이지에서 다음을 수행합니다.
 - a. JSON을 선택합니다.
 - b. 이 창의 내용을 이 섹션의 시작 부분에 있는 권한 정책으로 대체합니다.
 - c. Next(다음)를 선택합니다.
 - d. 정책의 이름과 설명(선택 사항) 및 태그를 입력한 다음에 정책 생성을 선택합니다.
5. 탐색 창에서 역할을 선택합니다.
6. 역할 생성을 선택합니다.
7. Trusted entity type(신뢰할 수 있는 엔터티 유형)에서 Custom trust policy(사용자 지정 정책)를 선택합니다. Custom trust policy(사용자 지정 정책)에서 "Principal": {}, 을(를) 다음으로 대체하고 Next(다음)를 선택합니다.

```
"Principal": {
```

```
"Service": "vpc-flow-logs.amazonaws.com"
},
```

8. Add permissions(권한 추가) 페이지에서 이 절차의 앞부분에서 생성한 정책의 확인란을 선택한 후 Next(다음)를 선택합니다.
9. 역할 이름을 입력하고 선택적으로 설명을 제공합니다.
10. 역할 생성(Create role)을 선택합니다.

CloudWatch Logs에 흐름 로그를 게시하는 IAM 보안 주체의 권한

요청을 수행하는 데 사용 중인 IAM 보안 주체에 iam:PassRole 작업을 직접적으로 호출할 수 있는 권한이 있는지 확인하세요.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["iam:PassRole"],
      "Resource": "arn:aws:iam::account-id:role/flow-log-role-name"
    }
  ]
}
```

CloudWatch Logs에 게시하는 흐름 로그 생성

VPC, 서브넷 또는 네트워크 인터페이스에 대한 흐름 로그를 생성할 수 있습니다. 특정 IAM 역할을 사용하는 사용자로 이러한 단계를 수행하는 경우 iam:PassRole 작업을 사용할 수 있는 권한이 있는지 확인하십시오. 자세한 내용은 [CloudWatch Logs에 흐름 로그를 게시하는 IAM 보안 주체의 권한](#) 단원을 참조하십시오.

전제 조건

- [the section called “CloudWatch Logs에 흐름 로그를 게시하는 IAM 역할”](#)에 설명된 대로 IAM 역할을 생성합니다.

콘솔을 사용하여 흐름 로그를 생성하는 방법

1. 다음 중 하나를 수행하십시오.

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다. 탐색 창에서 네트워크 인터페이스(Network Interfaces)를 선택합니다. 네트워크 인터페이스의 확인란을 선택합니다.
 - <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 Your VPCs(사용자 VPC)를 선택합니다. VPC에 대한 확인란을 선택합니다.
 - <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 서브넷(Subnets)을 선택합니다. 서브넷의 확인란을 선택합니다.
2. 작업(Actions), 흐름 로그 생성(Create flow log)을 선택합니다.
 3. 필터(Filter)에 기록할 트래픽 유형을 지정합니다. 모두(All)를 선택하여 수락된 트래픽 및 거부된 트래픽을 기록하거나, 거부(Reject)를 선택하여 거부된 트래픽만 기록하거나, 수락(Accept)을 선택해 수락된 트래픽만 기록합니다.
 4. Maximum aggregation interval(최대 집계 간격)에서 흐름이 캡처되어 흐름 로그 레코드로 집계되는 최대 기간을 선택합니다.
 5. Destination(대상)에서 Send to CloudWatch Logs(CloudWatch Logs로 전송))를 선택합니다.
 6. 대상 로그 그룹에서 기존 로그 그룹의 이름을 선택하거나 이 흐름 로그를 생성할 때 생성될 새 로그 그룹의 이름을 입력합니다.
 7. IAM 역할(IAM role)에서 CloudWatch Logs에 로그를 게시할 권한이 있는 역할의 이름을 지정합니다.
 8. 로그 레코드 형식(Log record format)에서 흐름 로그 레코드의 형식을 선택합니다.
 - 기본 형식을 사용하려면 AWS 기본 형식(default format)을 선택하세요.
 - 사용자 지정 형식을 사용하려면 사용자 지정 형식(Custom format)을 선택하고 로그 형식(Log format)에서 필드를 선택합니다.
 9. 추가 메타데이터에서 Amazon ECS의 메타데이터를 로그 형식으로 포함할지 선택합니다.
 10. (선택 사항) 새 태그 추가(Add new tag)를 선택하여 흐름 로그에 태그를 적용합니다.
 11. 흐름 로그 생성(Create flow log)을 선택합니다.

명령줄을 사용하여 흐름 로그를 만들려면 다음을 수행합니다.

다음 명령 중 하나를 사용합니다.

- [create-flow-logs](#)(AWS CLI)
- [New-EC2FlowLogs](#)(AWS Tools for Windows PowerShell)

다음 AWS CLI 예시에서는 지정된 서브넷에 대해 수락된 모든 트래픽을 캡처하는 흐름 로그를 생성합니다. 흐름 로그가 지정된 로그 그룹에 전송됩니다. `--deliver-logs-permission-arn` 파라미터에서는 CloudWatch Logs에 게시하는 데 필요한 IAM 역할을 지정합니다.

```
aws ec2 create-flow-logs --resource-type Subnet --resource-ids subnet-1a2b3c4d --
traffic-type ACCEPT --log-group-name my-flow-logs --deliver-logs-permission-arn
arn:aws:iam::123456789101:role/publishFlowLogs
```

흐름 로그 레코드 보기

CloudWatch Logs 콘솔을 사용하여 흐름 로그 레코드를 볼 수 있습니다. 흐름 로그를 생성하면 콘솔에 표시되는 데 몇 분 정도 걸릴 수도 있습니다.

콘솔을 사용하여 CloudWatch Logs에 게시된 흐름 로그 레코드를 보는 방법

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 로그(Logs), 로그 그룹(Log groups)을 선택합니다.
3. 흐름 로그가 있는 로그 그룹의 이름을 선택하여 세부 정보 페이지를 엽니다.
4. 흐름 로그 레코드가 있는 로그 스트림의 이름을 선택합니다. 자세한 내용은 [흐름 로그 레코드](#) 단원을 참조하십시오.

명령줄을 사용하여 CloudWatch Logs에 게시된 흐름 로그 레코드를 보는 방법

- [get-log-events](#)(AWS CLI)
- [Get-CWLogEvent](#)(AWS Tools for Windows PowerShell)

흐름 로그 레코드 검색

CloudWatch Logs 콘솔을 사용하여 CloudWatch Logs에 게시된 흐름 로그 레코드를 검색할 수 있습니다. [지표 필터](#)를 사용하여 흐름 로그 레코드를 필터링할 수 있습니다. 흐름 로그 레코드는 공백으로 구분됩니다.

CloudWatch Logs 콘솔을 사용하여 흐름 로그 레코드를 검색하려면

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 로그, 로그 그룹을 선택합니다.

3. 흐름 로그가 있는 로그 그룹을 선택한 다음에 검색 중인 네트워크 인터페이스를 아는 경우 로그 스트림을 선택합니다. 또는 Search log group(로그 그룹 검색)을 선택합니다. 로그 그룹에 네트워크 인터페이스가 많거나 선택한 시간 범위에 따라 시간이 걸릴 수 있습니다.
4. 이벤트 필터링에서 아래 문자열을 입력합니다. 여기서는 흐름 로그 레코드가 [기본 형식](#)을 사용한다고 가정합니다.

```
[version, accountid, interfaceid, srcaddr, dstaddr, srcport, dstport, protocol, packets, bytes, start, end, action, logstatus]
```

5. 필드의 값을 지정하여 필요에 따라 필터를 수정합니다. 다음 예제에서는 특정 원본 IP 주소를 기준으로 필터링합니다.

```
[version, accountid, interfaceid, srcaddr = 10.0.0.1, dstaddr, srcport, dstport, protocol, packets, bytes, start, end, action, logstatus]
[version, accountid, interfaceid, srcaddr = 10.0.2.*, dstaddr, srcport, dstport, protocol, packets, bytes, start, end, action, logstatus]
```

다음 예제에서는 대상 포트, 바이트 수 및 트래픽이 거부되었는지 여부를 기준으로 필터링합니다.

```
[version, accountid, interfaceid, srcaddr, dstaddr, srcport, dstport = 80 || dstport = 8080, protocol, packets, bytes, start, end, action, logstatus]
[version, accountid, interfaceid, srcaddr, dstaddr, srcport, dstport = 80 || dstport = 8080, protocol, packets, bytes >= 400, start, end, action = REJECT, logstatus]
```

CloudWatch Logs에서 흐름 로그 레코드 처리

흐름 로그 레코드는 CloudWatch Logs에서 수집한 다른 로그 이벤트처럼 사용할 수 있습니다. 로그 데이터 및 지표 필터 모니터링에 대한 자세한 내용은 Amazon CloudWatch 사용 설명서의 [로그 데이터 검색 및 필터링](#)을 참조하세요.

예: 흐름 로그에 대한 CloudWatch 지표 필터 및 경보 생성

이 예에서는 eni-1a2b3c4d에 대한 흐름 로그를 사용합니다. TCP 포트 22(SSH)를 거쳐 인스턴스에 연결하려는 시도가 한 시간 내에 10번 이상 거부된 경우 이를 알려 주는 알림을 만들 수 있습니다. 우선 경보를 만들려는 트래픽의 패턴과 일치하는 지표 필터를 만들어야 합니다. 그런 다음 지표 필터에 대한 경보를 만듭니다.

거부된 SSH 트래픽에 대한 지표 필터와 필터에 대한 경보를 만들려면

1. <https://console.aws.amazon.com/cloudwatch/>에서 CloudWatch 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 로그, 로그 그룹을 선택합니다.
3. 로그 그룹에 대한 확인란을 선택한 다음 작업(Actions), 지표 필터 생성(Create metric filter)을 선택합니다.
4. 필터 패턴(Filter Pattern)에서는 다음과 같은 문자열을 입력합니다.


```
[version, account, eni, source, destination, srcport, destport="22", protocol="6", packets, bytes, windowstart, windowend, action="REJECT", flowlogstatus]
```
5. 테스트할 로그 데이터 선택에서 네트워크 인터페이스에 대한 로그 스트림을 선택합니다. (선택 사항) 필터 패턴과 일치하는 로그 데이터 행을 보려면 패턴 테스트를 선택합니다.
6. 준비가 되면 다음을 선택합니다.
7. 필터 이름, 지표 네임스페이스 및 지표 이름을 입력합니다. 지표 값을 1로 설정합니다. 완료되면 다음을 선택하고 지표 필터 생성을 선택합니다.
8. 탐색 창에서 경보, 모든 경보를 선택합니다.
9. Create alarm(경보 생성)을 선택하세요.
10. 생성한 지표 이름을 선택한 후 지표 선택을 선택합니다.
11. 경보를 다음과 같이 구성한 다음 다음(Next)을 선택합니다.
 - Statistic(통계)에서 Sum(합계)를 선택합니다. 이것으로 지정된 기간 동안 데이터 포인트의 총 수를 캡처할 수 있습니다.
 - 기간에서 1시간을 선택합니다.
 - TimeSinceLastActive가 다음과 같은 때마다에는 크거나 같음을 선택하고 10을 임계값으로 입력합니다.
 - 추가 구성에서 경보에 대한 데이터 포인트를 기본값 1로 남겨둡니다.
12. Next(다음)를 선택합니다.
13. 알림에서 기존 SNS 주제를 선택하거나 새 주제 생성을 선택하여 새로 생성합니다. Next(다음)를 선택합니다.
14. 경보의 이름과 설명을 입력하고 다음을 선택합니다.
15. 경보 미리 보기를 완료했으면 경보 생성을 선택합니다.

Amazon S3에 흐름 로그 게시

흐름 로그는 흐름 로그 데이터를 Amazon S3에 게시할 수 있습니다.

Amazon S3에 게시하는 경우 흐름 로그 데이터가 지정해 놓은 기존 Amazon S3 버킷에 게시됩니다. 모니터링된 모든 네트워크 인터페이스에 대한 흐름 로그 레코드는 버킷에 저장된 일련의 로그 파일 객체에 게시됩니다. 흐름 로그가 VPC에 대한 데이터를 캡처하면, 흐름 로그가 모든 네트워크 인터페이스에 대한 흐름 로그 레코드를 선택된 VPC에 게시합니다.

흐름 로그와 함께 사용할 Amazon S3 버킷을 만드는 방법은 Amazon Simple Storage Service 사용 설명서의 [버킷 생성](#)을 참조하세요.

다중 계정 로깅에 대한 자세한 내용은 AWS 솔루션 라이브러리의 [중앙 로깅](#)을 참조하세요.

CloudWatch Logs에 대한 자세한 내용은 Amazon CloudWatch Logs 사용 설명서의 [Amazon S3로 전송된 로그](#)를 참조하세요.

요금

Amazon S3에 흐름 로그를 게시할 때는 Vended 로그에 대한 데이터 수집 및 아카이브 요금이 부과됩니다. 자세히 알아보려면 [Amazon CloudWatch Pricing](#)(Amazon CloudWatch 요금)을 열고, Logs(로그)를 선택하고, Vended Logs(벤딩 로그)를 찾으세요.

내용

- [흐름 로그 파일](#)
- [Amazon S3에 흐름 로그를 게시하는 IAM 보안 주체의 권한](#)
- [Amazon S3 버킷의 흐름 로그에 대한 권한](#)
- [SSE-KMS를 사용할 경우 필요한 키 정책](#)
- [Amazon S3 로그 파일 권한](#)
- [Amazon S3에 게시하는 흐름 로그 생성](#)
- [흐름 로그 레코드 보기](#)
- [Amazon S3에서 흐름 로그 레코드 처리](#)

흐름 로그 파일

VPC Flow Logs는 VPC에서 들어오고 나가는 IP 트래픽에 대한 데이터를 로그 레코드로 수집하고 해당 레코드를 로그 파일로 집계한 다음 5분 간격으로 Amazon S3 버킷에 로그 파일을 게시합니다. 여러 파

일이 게시될 수 있으며 각 로그 파일에는 이전 5분 동안 기록된 IP 트래픽에 대한 흐름 로그 레코드의 일부 또는 전체가 포함될 수 있습니다.

Amazon S3에서 흐름 로그 파일의 마지막 수정(Last modified) 필드는 파일이 Amazon S3 버킷에 업로드된 날짜와 시간을 나타냅니다. 파일 이름의 타임스탬프보다 이후이며 파일을 Amazon S3 버킷에 업로드하는 데 걸리는 시간에 따라 다릅니다.

로그 파일 형식

로그 파일에 대해 다음 형식 중 하나를 지정할 수 있습니다. 각 파일은 단일 Gzip 파일로 압축됩니다.

- 텍스트(Text) – 일반 텍스트. 이것은 기본 형식입니다.
- Parquet – Apache Parquet은 열 기반 데이터 형식입니다. Parquet 형식의 데이터에 대한 쿼리는 일반 텍스트 데이터에 대한 쿼리에 비해 10배에서 100배 빠릅니다. Gzip 압축을 사용하는 Parquet 형식 데이터는 Gzip 압축을 사용하는 일반 텍스트보다 스토리지 공간을 20% 적게 사용합니다.

Note

Gzip 압축을 사용하는 Parquet 형식 데이터가 집계 기간별로 100KB 미만이라면 데이터를 Parquet 형식으로 저장할 경우 Parquet 파일 메모리 요구 사항으로 인해 Gzip 압축을 사용하는 일반 텍스트보다 더 많은 공간을 차지할 수 있습니다.

로그 파일 옵션

필요한 경우 다음과 같은 옵션을 지정할 수 있습니다.

- Hive 호환 S3 접두사 – 분할을 Hive 호환 도구로 가져오는 대신 Hive 호환 접두사를 활성화합니다. 쿼리 실행 전에 MSCK REPAIR TABLE 명령을 사용합니다.
- 시간당 분할 – 대량의 로그가 있고 일반적으로 특정 시간까지 쿼리를 타겟팅하는 경우 로그를 시간별로 분할하여 더 결과를 빠르게 얻고 쿼리 비용을 절감할 수 있습니다.

로그 파일 S3 버킷 구조

로그 파일은 흐름 로그의 ID, 리전, 생성된 날짜 및 대상 옵션에 따라 폴더 구조를 사용하여 지정된 Amazon S3 버킷에 저장됩니다.

기본적으로 파일은 다음 위치로 전달됩니다.

```
bucket-and-optional-prefix/AWSLogs/account_id/vpcflowlogs/region/year/month/day/
```

Hive 호환 S3 접두사를 사용하도록 설정하면 파일이 다음 위치로 전달됩니다.

```
bucket-and-optional-prefix/AWSLogs/aws-account-id=account_id/aws-service=vpcflowlogs/  
aws-region=region/year=year/month=month/day=day/
```

시간별 분할을 사용하도록 설정하면 파일이 다음 위치로 전달됩니다.

```
bucket-and-optional-prefix/AWSLogs/account_id/vpcflowlogs/region/year/month/day/hour/
```

Hive 호환 분할을 사용하도록 설정하고 시간당 흐름 로그를 분할하면 파일이 다음 위치로 전달됩니다.

```
bucket-and-optional-prefix/AWSLogs/aws-account-id=account_id/aws-service=vpcflowlogs/  
aws-region=region/year=year/month=month/day=day/hour=hour/
```

로그 파일 이름

로그 파일의 파일 이름은 흐름 로그 ID, 리전 및 생성 날짜 및 시간을 기반으로 합니다. 파일 이름은 다음 형식을 사용합니다.

```
aws_account_id_vpcflowlogs_region_flow_log_id_YYYYMMDDTHHmmZ_hash.log.gz
```

예를 들어, 다음은 June 20, 2018, 16:20 UTC에 us-east-1 리전의 리소스에 대해 AWS 계정 123456789012에서 생성한 흐름 로그에 대한 로그 파일의 예를 보여 줍니다. 종료 시간이 16:20:00에서 16:24:59 사이인 흐름 로그 레코드가 파일에 포함됩니다.

```
123456789012_vpcflowlogs_us-east-1_fl-1234abcd_20180620T1620Z_fe123456.log.gz
```

Amazon S3에 흐름 로그를 게시하는 IAM 보안 주체의 권한

흐름 로그를 생성하는 IAM 보안 주체는 대상 Amazon S3 버킷에 흐름 로그를 게시하는 데 필요한 다음 권한이 있는 IAM 역할을 사용해야 합니다.

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

```

```

        "logs:CreateLogDelivery",
        "logs>DeleteLogDelivery"
    ],
    "Resource": "*"
}
]
}

```

Amazon S3 버킷의 흐름 로그에 대한 권한

기본적으로 Amazon S3 버킷과 버킷에 포함된 객체는 비공개입니다. 버킷 소유자만이 해당 버킷과 그 안에 저장된 객체에 액세스할 수 있습니다. 그러나 버킷 소유자는 액세스 정책을 작성하여 다른 리소스 및 사용자에게 액세스 권한을 부여할 수 있습니다.

흐름 로그를 생성하는 사용자가 버킷을 소유하고 해당 버킷에 대한 PutBucketPolicy 및 GetBucketPolicy 권한을 소유한 경우, 다음 정책을 해당 버킷에 자동으로 연결합니다. 이 정책은 버킷에 연결된 모든 기존 정책을 덮어씁니다.

그렇지 않으면 버킷 소유자가 이 정책을 버킷에 추가하고 흐름 로그 작성자의 AWS 계정 ID 지정 또는 흐름 로그 생성이 실패합니다. 자세한 내용은 Amazon Simple Storage Service 사용 설명서의 [버킷 정책 사용](#)을 참조하세요.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AWSLogDeliveryWrite",
      "Effect": "Allow",
      "Principal": {
        "Service": "delivery.logs.amazonaws.com"
      },
      "Action": "s3:PutObject",
      "Resource": "my-s3-arn/*",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": account_id,
          "s3:x-amz-acl": "bucket-owner-full-control"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:logs:region:account_id:*"
        }
      }
    }
  ]
}

```

```

    },
    {
      "Sid": "AWSLogDeliveryAclCheck",
      "Effect": "Allow",
      "Principal": {
        "Service": "delivery.logs.amazonaws.com"
      },
      "Action": [
        "s3:GetBucketAcl",
        "s3:ListBucket"
      ],
      "Resource": "arn:aws:s3:::bucket_name",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": account_id
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:logs:region:account_id:*"
        }
      }
    }
  ]
}

```

*my-s3-arn*에 지정한 ARN은 Hive 호환 S3 접두사를 사용하는지 여부에 따라 다릅니다.

- 기본 접두사

```
arn:aws:s3:::bucket_name/optional_folder/AWSLogs/account_id/*
```

- Hive 호환 S3 접두사

```
arn:aws:s3:::bucket_name/optional_folder/AWSLogs/aws-account-id=account_id/*
```

개별 AWS 계정 ARN 대신 로그 전송 서비스 보안 주체에 이 권한들을 부여하는 것이 좋습니다. 또한 [혼동된 대리자 문제](#)로부터 보호하려면 `aws:SourceAccount` 및 `aws:SourceArn` 조건 키를 사용하는 것 좋습니다. 소스 계정은 흐름 로그의 소유자이고 원본 ARN은 로그 서비스의 와일드카드(*) ARN입니다.

SSE-KMS를 사용할 경우 필요한 키 정책

해당 S3 버킷에서 Amazon S3-관리형 키(SSE-S3)를 사용한 서버 측 암호화 또는 KMS 키(SSE-KMS)를 사용한 서버 측 암호화를 활성화하여 Amazon S3 버킷의 데이터를 보호할 수 있습니다. 자세한 내용은 Amazon S3 사용 설명서의 [서버 측 암호화를 사용하여 데이터 보호](#)를 참조하세요.

SSE-S3를 선택하면 추가 구성이 필요하지 않습니다. Amazon S3는 암호화 키를 처리합니다.

SSE-KMS를 선택하면 고객 관리형 키 ARN을 사용해야 합니다. 키 ID를 사용하는 경우 흐름 로그를 생성할 때 [LogDestination 전송 불가](#) 오류가 발생할 수 있습니다. 또한 로그 전달 계정이 S3 버킷에 쓸 수 있도록 고객 관리형 키에 대한 키 정책을 업데이트해야 합니다. SSE-KMS와 함께 사용하는 데 필요한 키 정책에 대한 자세한 내용은 Amazon CloudWatch Logs 사용 설명서의 [Amazon S3 버킷 서버 측 암호화](#)를 참조하십시오.

Amazon S3 로그 파일 권한

필요한 버킷 정책 외에도, Amazon S3는 ACL(액세스 제어 목록)을 사용하여 흐름 로그에서 생성한 로그 파일에 대한 액세스를 관리합니다. 기본적으로 버킷 소유자는 각 로그 파일에 대한 FULL_CONTROL 권한을 보유하고 있습니다. 로그 전송 소유자가 버킷 소유자와 다른 경우에는 권한이 없습니다. 로그 전송 계정에는 READ 및 WRITE 권한이 부여됩니다. 자세한 내용은 Amazon Simple Storage Service 사용 설명서의 [액세스 제어 목록\(ACL\) 개요](#)를 참조하십시오.

Amazon S3에 게시하는 흐름 로그 생성

Amazon S3 버킷을 생성하고 구성한 후에는 네트워크 인터페이스, 서브넷 및 VPC에 대한 흐름 로그를 생성할 수 있습니다.

콘솔을 사용하여 흐름 로그를 생성하는 방법

1. 다음 중 하나를 수행하십시오.

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다. 탐색 창에서 네트워크 인터페이스(Network Interfaces)를 선택합니다. 네트워크 인터페이스의 확인란을 선택합니다.
- <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 Your VPCs(사용자 VPC)를 선택합니다. VPC에 대한 확인란을 선택합니다.
- <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 서브넷(Subnets)을 선택합니다. 서브넷의 확인란을 선택합니다.

2. 작업(Actions), 흐름 로그 생성(Create flow log)을 선택합니다.

3. 필터(Filter)에서 로깅할 IP 트래픽 데이터의 유형을 지정합니다.
 - 수락 - 수락한 트래픽만 로그합니다.
 - 거부 - 거부한 트래픽만 로그합니다.
 - 모두 - 허용 및 거부된 트래픽을 로그합니다.
4. 최대 집계 간격(Maximum aggregation interval)에서 흐름이 캡처되어 흐름 로그 레코드로 집계되는 최대 기간을 선택합니다.
5. 대상에서 Amazon S3 버킷으로 전송(Send to an Amazon S3 bucket)을 선택합니다.
6. S3 버킷 ARN(S3 bucket ARN)의 경우 기존 Amazon S3 버킷의 Amazon 리소스 이름(ARN)을 지정합니다. 필요한 경우 하위 폴더를 포함할 수 있습니다. 예를 들어 my-bucket이란 이름의 버킷에 my-logs이란 이름의 하위 폴더를 지정하려면 다음 ARN을 사용하십시오.

```
arn:aws:s3:::my-bucket/my-logs/
```

버킷에 AWSLogs를 하위 폴더 이름으로 사용할 수 없습니다. 이것은 예약된 용어입니다.

버킷을 소유한 경우, 자동으로 리소스 정책을 생성하여 버킷에 연결합니다. 자세한 내용은 [Amazon S3 버킷의 흐름 로그에 대한 권한](#) 단원을 참조하십시오.

7. 로그 레코드 형식에서 흐름 로그 레코드의 형식을 지정합니다.
 - 기본 흐름 로그 레코드 형식을 사용하려면 AWS 기본 형식을 선택하세요.
 - 사용자 지정 형식을 만들려면 사용자 지정 형식을 선택하십시오. 로그 형식(Log format)에 대해 흐름 로그 레코드에 포함할 필드를 선택하세요.
8. 추가 메타데이터에서 Amazon ECS의 메타데이터를 로그 형식으로 포함할지 선택합니다.
9. 로그 파일 형식의 경우 로그 파일의 형식을 지정합니다.
 - Text - 일반 텍스트. 이것은 기본 형식입니다.
 - Parquet - Apache Parquet은 열 기반 데이터 형식입니다. Parquet 형식의 데이터에 대한 쿼리는 일반 텍스트 데이터에 대한 쿼리에 비해 10배에서 100배 빠릅니다. Gzip 압축을 사용하는 Parquet 형식 데이터는 Gzip 압축을 사용하는 일반 텍스트보다 스토리지 공간을 20% 적게 사용합니다.
10. (선택 사항) Hive 호환 S3 접두사를 사용하려면 Hive 호환 S3 접두사, 활성화를 선택합니다.
11. (선택 사항) 흐름 로그를 시간당 분할하려면 1시간마다 (60분)을 선택합니다.
12. (선택 사항) 흐름 로그에 태그를 추가하려면 새 태그 추가를 선택하여 태그 키와 값을 지정하십시오.
13. 흐름 로그 생성(Create flow log)을 선택합니다.

명령줄 도구를 사용하여 Amazon S3에 게시하는 흐름 로그를 만들려면

다음 명령 중 하나를 사용합니다.

- [create-flow-logs](#)(AWS CLI)
- [New-EC2FlowLogs](#)(AWS Tools for Windows PowerShell)

다음 AWS CLI 예시에서는 지정된 VPC의 모든 트래픽을 캡처하고 지정된 Amazon S3 버킷에 흐름 로그를 전송하는 흐름 로그를 생성합니다. `--log-format` 파라미터는 흐름 로그 레코드의 사용자 지정 형식을 지정합니다.

```
aws ec2 create-flow-logs --resource-type VPC --resource-ids vpc-00112233344556677 --
traffic-type ALL --log-destination-type s3 --log-destination arn:aws:s3:::flow-log-
bucket/custom-flow-logs/ --log-format '${version} ${vpc-id} ${subnet-id} ${instance-
id} ${srcaddr} ${dstaddr} ${srcport} ${dstport} ${protocol} ${tcp-flags} ${type} ${pkt-
srcaddr} ${pkt-dstaddr}'
```

흐름 로그 레코드 보기

Amazon S3 콘솔을 사용하여 흐름 로그 레코드를 볼 수 있습니다. 흐름 로그를 생성하면 콘솔에 표시 되는 데 몇 분 정도 걸릴 수도 있습니다.

Amazon S3에 게시된 흐름 로그 레코드를 보려면

1. <https://console.aws.amazon.com/s3/>에서 Amazon S3 콘솔을 엽니다.
2. 버킷 이름을 선택하여 세부 정보 페이지를 엽니다.
3. 로그 파일이 있는 폴더로 이동합니다. 예: `prefix/AWSLogs/account_id/vpcflowlogs/region/year/month/day/`.
4. 파일 이름 옆에 있는 확인란을 선택한 다음에 Download(다운로드)를 선택합니다.

Amazon S3에서 흐름 로그 레코드 처리

로그 파일은 압축된 상태입니다. Amazon S3 콘솔을 사용해 로그 파일을 열면 압축이 해제되고 흐름 로그 레코드가 표시됩니다. 파일을 다운로드하는 경우, 압축을 해제해야 흐름 로그 레코드를 볼 수 있습니다.

또한 Amazon Athena를 사용해 로그 파일의 흐름 로그 레코드를 쿼리할 수도 있습니다. Amazon Athena는 표준 SQL을 사용해 Amazon S3에 저장된 데이터를 더 쉽게 분석할 수 있는 대화식 쿼리 서

비스입니다. 자세한 내용은 Amazon Athena 사용 설명서의 [Amazon VPC 흐름 로그 쿼리 방법](#)을 참조하세요.

Amazon Data Firehose에 흐름 로그 게시

흐름 로그에서는 흐름 로그 데이터를 Amazon Data Firehose에 직접 게시할 수 있습니다.

Amazon Data Firehose에 게시할 때 흐름 로그 데이터는 일반 텍스트 형식으로 Amazon Data Firehose 전송 스트림에 게시됩니다.

요금

표준 모으기 및 전송 요금이 적용됩니다. 자세히 알아보려면 [Amazon CloudWatch Pricing](#)(Amazon CloudWatch 요금)을 열고, Logs(로그)를 선택하고, Vended Logs(벤딩 로그)를 찾으세요.

내용

- [교차 계정 전송에 대한 IAM 역할](#)
- [Amazon Data Firehose에 게시하는 흐름 로그 생성](#)
- [Amazon Data Firehose의 흐름 로그 레코드 처리](#)

교차 계정 전송에 대한 IAM 역할

Amazon Data Firehose에 게시할 때 모니터링할 리소스와 동일한 계정(소스 계정) 또는 상이한 계정(대상 계정)에 있는 전송 스트림을 선택할 수 있습니다. Amazon Data Firehose에 대한 흐름 로그의 교차 계정 전송을 활성화하려면 소스 계정에서 IAM 역할을 생성하고 대상 계정에서 IAM 역할을 생성해야 합니다.

역할

- [소스 계정 역할](#)
- [대상 계정 역할](#)

소스 계정 역할

소스 계정에서 다음과 같은 권한을 부여하는 역할을 생성합니다. 이 예시에서는 역할 이름이 mySourceRole이지만, 이 역할에 대해 다른 이름을 선택할 수 있습니다. 마지막 명령문에서는 대상 계정의 역할에 이 역할 수임을 허용합니다. 조건문에서는 지정된 리소스를 모니터링할 때만 이 역할이 로그 전송 서비스에만 전달되도록 합니다. 정책을 생성할 때 모니터링 중인 VPC, 네트워크 인터페이스 또는 서브넷을 조건 키(iam:AssociatedResourceARN)로 지정합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::source-account:role/mySourceRole",
      "Condition": {
        "StringEquals": {
          "iam:PassedToService": "delivery.logs.amazonaws.com"
        },
        "StringLike": {
          "iam:AssociatedResourceARN": [
            "arn:aws:ec2:region:source-account:vpc/vpc-00112233344556677"
          ]
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogDelivery",
        "logs>DeleteLogDelivery",
        "logs>ListLogDeliveries",
        "logs:GetLogDelivery"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": "arn:aws:iam::destination-account:role/AWSLogDeliveryFirehoseCrossAccountRole"
    }
  ]
}

```

로그 전송 서비스에서 역할을 수임할 수 있는 다음과 같은 신뢰 정책이 이 역할에 있는지 확인하세요.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {

```

```

    "Effect": "Allow",
    "Principal": {
      "Service": "delivery.logs.amazonaws.com"
    },
    "Action": "sts:AssumeRole"
  }
]
}

```

소스 계정에서 다음과 같은 절차를 사용하여 역할을 생성합니다.

소스 계정 역할을 생성하는 방법

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 정책을 선택합니다.
3. [정책 생성(Create policy)]을 선택합니다.
4. [Create policy(정책 생성)] 페이지에서 다음을 수행합니다.
 - a. JSON을 선택합니다.
 - b. 이 창의 내용을 이 섹션의 시작 부분에 있는 권한 정책으로 대체합니다.
 - c. Next(다음)를 선택합니다.
 - d. 정책의 이름과 설명(선택 사항) 및 태그를 입력한 다음에 정책 생성을 선택합니다.
5. 탐색 창에서 역할을 선택합니다.
6. 역할 생성을 선택합니다.
7. Trusted entity type(신뢰할 수 있는 엔터티 유형)에서 Custom trust policy(사용자 지정 정책)를 선택합니다. Custom trust policy(사용자 지정 신뢰 정책)에서 로그 전송 서비스를 지정하는 다음으로 "Principal": {}, 을 대체합니다. Next(다음)를 선택합니다.

```

"Principal": {
  "Service": "delivery.logs.amazonaws.com"
},

```

8. Add permissions(권한 추가) 페이지에서 이 절차의 앞부분에서 생성한 정책의 확인란을 선택한 후 Next(다음)를 선택합니다.
9. 역할 이름을 입력하고 선택적으로 설명을 제공합니다.
10. 역할 생성(Create role)을 선택합니다.

대상 계정 역할

대상 계정에서 AWSLogDeliveryFirehoseCrossAccountRole로 시작하는 이름으로 역할을 생성합니다. 이 역할에서는 다음과 같은 권한을 부여해야 합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:CreateServiceLinkedRole",
        "firehose:TagDeliveryStream"
      ],
      "Resource": "*"
    }
  ]
}
```

이 역할을 수입할 수 있도록 소스 계정에서 생성한 역할이 허용되는 다음과 같은 신뢰 정책이 이 역할에 있는지 확인하세요.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::source-account:role/mySourceRole"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

대상 계정에서 다음과 같은 절차를 사용하여 역할을 생성합니다.

대상 계정 역할을 생성하는 방법

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 정책을 선택합니다.
3. [정책 생성(Create policy)]을 선택합니다.

4. [Create policy(정책 생성)] 페이지에서 다음을 수행합니다.
 - a. JSON을 선택합니다.
 - b. 이 창의 내용을 이 섹션의 시작 부분에 있는 권한 정책으로 대체합니다.
 - c. Next(다음)를 선택합니다.
 - d. AWSLogDeliveryFirehoseCrossAccountRole로 시작하는 정책 이름을 입력한 다음에 정책 생성(Create policy)을 선택합니다.
5. 탐색 창에서 역할을 선택합니다.
6. 역할 생성을 선택합니다.
7. Trusted entity type(신뢰할 수 있는 엔터티 유형)에서 Custom trust policy(사용자 지정 정책)를 선택합니다. Custom trust policy(사용자 지정 신뢰 정책)에서 소스 계정 역할을 지정하는 다음으로 "Principal": {}, 을 대체합니다. Next(다음)를 선택합니다.

```
"Principal": {
  "AWS": "arn:aws:iam::source-account:role/mySourceRole"
},
```

8. Add permissions(권한 추가) 페이지에서 이 절차의 앞부분에서 생성한 정책의 확인란을 선택한 후 Next(다음)를 선택합니다.
9. 역할 이름을 입력하고 선택적으로 설명을 제공합니다.
10. 역할 생성(Create role)을 선택합니다.

Amazon Data Firehose에 게시하는 흐름 로그 생성

VPC, 서브넷 또는 네트워크 인터페이스에 대한 흐름 로그를 생성할 수 있습니다.

필수 조건

- 대상 Amazon Data Firehose 전송 스트림을 생성합니다. Direct Put을 원본으로 사용합니다. 자세한 내용은 [Amazon Data Firehose 전송 스트림 생성](#)을 참조하세요.
- 흐름 로그를 다른 계정에 게시하는 경우 [the section called “교차 계정 전송에 대한 IAM 역할”](#)에 설명된 대로 필수 IAM 역할을 생성합니다.

Amazon Data Firehose에 게시하는 흐름 로그 생성 방법

1. 다음 중 하나를 수행하십시오.

- <https://console.aws.amazon.com/ec2/>에서 Amazon EC2 콘솔을 엽니다. 탐색 창에서 네트워크 인터페이스(Network Interfaces)를 선택합니다. 네트워크 인터페이스의 확인란을 선택합니다.
 - <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 Your VPCs(사용자 VPC)를 선택합니다. VPC에 대한 확인란을 선택합니다.
 - <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다. 탐색 창에서 서브넷(Subnets)을 선택합니다. 서브넷의 확인란을 선택합니다.
2. 작업(Actions), 흐름 로그 생성(Create flow log)을 선택합니다.
 3. 필터(Filter)에 기록할 트래픽 유형을 지정합니다.
 - 수락 – 수락한 트래픽만 로그합니다.
 - 거부 – 거부한 트래픽만 로그합니다.
 - 모두 – 수락 및 거부한 트래픽을 로그합니다.
 4. 최대 집계 간격(Maximum aggregation interval)에서 흐름이 캡처되어 흐름 로그 레코드로 집계되는 최대 기간을 선택합니다.
 5. 대상(Destination)에서는 다음과 같은 옵션 중 하나를 선택합니다.
 - 동일한 계정의 Amazon Data Firehose로 보내기 - 전송 스트림과 모니터링할 리소스가 동일한 계정에 있습니다.
 - 다른 계정의 Amazon Data Firehose로 보내기 - 전송 스트림과 모니터링할 리소스가 상이한 계정에 있습니다.
 6. Amazon Data Firehose 스트림 이름에는 생성한 전송 스트림을 선택합니다.
 7. [교차 계정 전송만 해당] IAM roles(IAM 역할)에서는 필수 역할을 지정합니다([the section called “교차 계정 전송에 대한 IAM 역할”](#) 참조).
 8. 로그 레코드 형식에서 흐름 로그 레코드의 형식을 지정합니다.
 - 기본 흐름 로그 레코드 형식을 사용하려면 AWS 기본 형식을 선택하세요.
 - 사용자 지정 형식을 만들려면 사용자 지정 형식을 선택하십시오. 로그 형식(Log format)에 대해 흐름 로그 레코드에 포함할 필드를 선택하세요.
 9. 추가 메타데이터에서 Amazon ECS의 메타데이터를 로그 형식으로 포함할지 선택합니다.
 10. (선택 사항) 태그 추가를 선택하여 흐름 로그에 태그를 적용합니다.
 11. 흐름 로그 생성(Create flow log)을 선택합니다.

명령줄 도구를 사용하여 Amazon Data Firehose에 게시하는 흐름 로그를 생성하는 방법

다음 명령 중 하나를 사용합니다.

- [create-flow-logs](#)(AWS CLI)
- [New-EC2FlowLogs](#)(AWS Tools for Windows PowerShell)

다음 AWS CLI 예시에서는 지정된 VPC의 모든 트래픽을 캡처하고 동일한 계정의 지정된 Amazon Data Firehose 전송 스트림에 흐름 로그를 전송하는 흐름 로그를 생성합니다.

```
aws ec2 create-flow-logs --traffic-type ALL \
  --resource-type VPC \
  --resource-ids vpc-00112233344556677 \
  --log-destination-type kinesis-data-firehose \
  --log-destination arn:aws:firehose:us-
east-1:123456789012:deliverystream:flowlogs_stream
```

다음 AWS CLI 예시에서는 지정된 VPC의 모든 트래픽을 캡처하고 상이한 계정의 지정된 Amazon Data Firehose 전송 스트림에 흐름 로그를 전송하는 흐름 로그를 생성합니다.

```
aws ec2 create-flow-logs --traffic-type ALL \
  --resource-type VPC \
  --resource-ids vpc-00112233344556677 \
  --log-destination-type kinesis-data-firehose \
  --log-destination arn:aws:firehose:us-
east-1:123456789012:deliverystream:flowlogs_stream \
  --deliver-logs-permission-arn arn:aws:iam::source-account:role/mySourceRole \
  --deliver-cross-account-role arn:aws:iam::destination-account:role/
AWSLogDeliveryFirehoseCrossAccountRole
```

Amazon Data Firehose의 흐름 로그 레코드 처리

전송 스트림에 대해 구성한 대상에서 흐름 로그 데이터를 가져올 수 있습니다.

Amazon Athena를 사용하여 흐름 로그 쿼리

Amazon Athena는 표준 SQL을 사용하여 흐름 로그와 같은 Amazon S3의 데이터를 분석할 수 있는 대화형 쿼리 서비스입니다. Athena를 VPC 흐름 로그와 함께 사용하여 VPC를 통해 흐르는 트래픽에 대해 유용한 인사이트를 빠르게 얻을 수 있습니다. 예를 들어 Virtual Private Cloud(VPC)의 어떤 리소스가 상위 대화자인지 식별하거나 TCP 연결이 가장 많이 거부된 IP 주소를 식별할 수 있습니다.

옵션

- VPC를 통해 흐르는 트래픽에 대한 인사이트를 얻기 위해 실행할 수 있는 필수 AWS 리소스와 사전 정의된 쿼리를 생성하는 CloudFormation 템플릿을 생성하여 VPC 흐름 로그와 Athena의 통합을 간소화하고 자동화할 수 있습니다.
- Athena를 사용하여 고유한 쿼리를 생성할 수 있습니다. 자세한 내용은 Amazon Athena 사용 설명서의 [Amazon Athena를 사용하여 쿼리 실행](#)을 참조하십시오.

요금

쿼리 실행에 대해 표준 [Amazon Athena 요금](#)이 발생합니다. 반복 일정에 따라 새 파티션을 로드하는 Lambda 함수에 대해 표준 [AWS Lambda 요금](#)이 발생합니다(파티션 로드 빈도를 지정하고 시작 날짜와 종료 날짜는 지정하지 않은 경우).

미리 정의된 쿼리를 사용하려면

- [콘솔을 사용하여 CloudFormation 템플릿 생성](#)
- [AWS CLI를 사용하여 CloudFormation 템플릿 생성](#)
- [사전 정의된 쿼리 실행](#)

콘솔을 사용하여 CloudFormation 템플릿 생성

첫 번째 흐름 로그가 S3 버킷으로 전송된 후 CloudFormation 템플릿을 생성하고 해당 템플릿을 사용하여 스택을 생성함으로써 Athena와 통합할 수 있습니다.

요구 사항

- 선택한 리전은 AWS Lambda 및 Amazon Athena를 지원해야 합니다.
- 선택한 리전에 Amazon S3 버킷이 있어야 합니다.
- 흐름 로그의 로그 레코드 형식에는 실행하려는 미리 정의된 특정 쿼리에서 사용하는 필드가 포함되어야 합니다.

콘솔을 사용하여 템플릿을 생성하려면

1. 다음 중 하나를 수행하세요.

- Amazon VPC 콘솔을 엽니다. 탐색 창에서 [VPC(Your VPCs)]를 선택한 후 해당 VPC를 선택합니다.

- Amazon VPC 콘솔을 엽니다. 탐색 창에서 [서브넷(Subnets)]을 선택한 후 해당 서브넷을 선택합니다.
 - Amazon EC2 콘솔을 엽니다. 탐색 창에서 [네트워크 인터페이스(Network Interfaces)]를 선택한 후 해당 네트워크 인터페이스를 선택합니다.
2. [흐름 로그(Flow logs)] 탭에서 Amazon S3에 게시하는 흐름 로그를 선택한 후 [작업(Actions)], [Athena 통합 생성(Generate Athena integration)]을 차례로 선택합니다.
 3. 파티션 로드 빈도를 지정합니다. 없음(None)을 선택하는 경우 과거 날짜를 사용하여 파티션 시작 및 종료 날짜를 지정해야 합니다. [매일(Daily)], [매주(Weekly)] 또는 [매월(Monthly)]을 선택하는 경우 파티션 시작 및 종료 날짜는 선택 사항입니다. 시작 날짜와 종료 날짜를 지정하지 않으면 CloudFormation 템플릿은 반복 일정에 따라 새 파티션을 로드하는 Lambda 함수를 생성합니다.
 4. 생성된 템플릿을 저장할 S3 버킷을 선택하고, 쿼리 결과를 저장할 S3 버킷을 선택하거나 생성합니다.
 5. [Athena 통합 생성(Generate Athena integration)]을 선택합니다.
 6. (선택 사항) 성공 메시지에서 링크를 선택하여 CloudFormation 템플릿용으로 지정한 버킷으로 이동하고, 템플릿을 사용자 지정합니다.
 7. 성공 메시지에서 [CloudFormation 스택 생성(Create CloudFormation stack)]을 선택하여 AWS CloudFormation 콘솔에서 [스택 생성(Create Stack)] 마법사를 엽니다. 생성된 CloudFormation 템플릿의 URL은 템플릿(Template) 섹션에 명시되어 있습니다. 마법사를 완료하여 템플릿에 지정된 리소스를 생성합니다.

CloudFormation 템플릿으로 생성된 리소스

- Athena 데이터베이스. 데이터베이스 이름은 `vpcflowlogsathenadatabase<flow-logs-subscription-id>`입니다.
- Athena 작업 그룹. 작업 그룹 이름은 `<flow-log-subscription-id><partition-load-frequency><start-date><end-date>workgroup`입니다.
- 흐름 로그 레코드에 해당하는 분할된 Athena 테이블. 테이블 이름은 `<flow-log-subscription-id><partition-load-frequency><start-date><end-date>`입니다.
- Athena 명명된 쿼리 세트. 자세한 내용은 [사전 정의된 쿼리](#) 단원을 참조하십시오.
- 지정된 일정(매일, 매주 또는 매월)에 따라 테이블에 새 파티션을 로드하는 Lambda 함수입니다.
- Lambda 함수를 실행할 권한을 부여하는 IAM 역할입니다.

AWS CLI를 사용하여 CloudFormation 템플릿 생성

첫 번째 흐름 로그가 S3 버킷으로 전송된 후 CloudFormation 템플릿을 생성하고 해당 템플릿을 사용하여 Athena와 통합할 수 있습니다.

다음 [get-flow-logs-integration-template](#) 명령을 사용하여 CloudFormation 템플릿을 생성합니다.

```
aws ec2 get-flow-logs-integration-template --cli-input-json file://config.json
```

다음은 config.json 파일의 예입니다.

```
{
  "FlowLogId": "fl-12345678901234567",
  "ConfigDeliveryS3DestinationArn": "arn:aws:s3:::my-flow-logs-athena-integration/
templates/",
  "IntegrateServices": {
    "AthenaIntegrations": [
      {
        "IntegrationResultS3DestinationArn": "arn:aws:s3:::my-flow-logs-
analysis/athena-query-results/",
        "PartitionLoadFrequency": "monthly",
        "PartitionStartDate": "2021-01-01T00:00:00",
        "PartitionEndDate": "2021-12-31T00:00:00"
      }
    ]
  }
}
```

다음 [create-stack](#) 명령을 사용하여 생성된 CloudFormation 템플릿으로 스택을 생성합니다.

```
aws cloudformation create-stack --stack-name my-vpc-flow-logs --template-body file://
my-cloudformation-template.json
```

사전 정의된 쿼리 실행

생성된 CloudFormation 템플릿은 AWS 네트워크의 트래픽에 대한 의미 있는 인사이트를 빠르게 얻기 위해 실행할 수 있는 사전 정의된 쿼리 세트를 제공합니다. 스택을 생성하고 모든 리소스가 올바르게 생성되었는지 확인한 후 사전 정의된 쿼리 중 하나를 실행할 수 있습니다.

콘솔을 사용하여 사전 정의된 쿼리를 실행하려면

1. Athena 콘솔을 엽니다.

2. 탐색 창에서 쿼리 편집기(Query editor)를 선택합니다. 작업 그룹(Workgroups)에서 CloudFormation 템플릿으로 생성한 작업 그룹을 선택합니다.
3. 저장된 쿼리(Saved queries)를 선택하고 필요에 따라 파라미터를 수정한 후 쿼리를 실행합니다. 사용 가능한 사전 정의된 쿼리 목록은 [사전 정의된 쿼리](#)를 참조하세요.
4. 쿼리 결과(Query results)에서 쿼리 결과를 확인합니다.

사전 정의된 쿼리

다음은 Athena 명명된 쿼리의 전체 목록입니다. 템플릿을 생성할 때 제공되는 미리 정의된 쿼리는 흐름 로그에 대한 로그 레코드 형식의 일부인 필드에 따라 달라집니다. 따라서 템플릿에 이러한 미리 정의된 쿼리가 모두 포함되어 있지 않을 수 있습니다.

- VpcFlowLogsAcceptedTraffic - 보안 그룹 및 네트워크 ACL에 따라 허용된 TCP 연결입니다.
- VpcFlowLogsAdminPortTraffic - 관리 포트에서 요청을 처리하는 애플리케이션에서 기록된 트래픽이 가장 많은 상위 10개의 IP 주소입니다.
- VpcFlowLogsIPv4Traffic - 기록된 IPv4 트래픽의 총 바이트 수입니다.
- VpcFlowLogsIPv6Traffic - 기록된 IPv6 트래픽의 총 바이트 수입니다.
- VpcFlowLogsRejectedTCPTraffic - 보안 그룹 또는 네트워크 ACL에 따라 거부된 TCP 연결입니다.
- VpcFlowLogsRejectedTraffic - 보안 그룹 또는 네트워크 ACL에 따라 거부된 트래픽입니다.
- VpcFlowLogsSshRdpTraffic - SSH 및 RDP 트래픽입니다.
- VpcFlowLogsTopTalkers - 기록된 트래픽이 많은 IP 주소 50개입니다.
- VpcFlowLogsTopTalkersPacketLevel - 기록된 트래픽이 가장 많은 50개의 패킷 수준 IP 주소입니다.
- VpcFlowLogsTopTalkingInstances - 기록된 트래픽이 가장 많은 50개 인스턴스의 ID입니다.
- VpcFlowLogsTopTalkingSubnets - 기록된 트래픽이 가장 많은 50개 서브넷의 ID입니다.
- VpcFlowLogsTopTCPTraffic - 특정 소스 IP 주소에 대해 기록된 모든 TCP 트래픽입니다.
- VpcFlowLogsTopTCPTraffic - 기록된 바이트 수가 가장 많은 50쌍의 소스 및 대상 IP 주소입니다.
- VpcFlowLogsTotalBytesTransferredPacketLevel - 기록된 바이트 수가 가장 많은 50쌍의 패킷 수준 소스 및 대상 IP 주소입니다.
- VpcFlowLogsTrafficFrmSrcAddr - 특정 소스 IP 주소에 대해 기록된 트래픽입니다.
- VpcFlowLogsTrafficToDstAddr - 특정 대상 IP 주소에 대해 기록된 트래픽입니다.

VPC 흐름 로그 문제 해결

다음은 흐름 로그로 작업할 때 발생할 수 있는 문제입니다.

문제

- [불완전한 흐름 로그 레코드](#)
- [흐름 로그가 활성화되었지만 흐름 로그 레코드 또는 로그 그룹이 없음](#)
- ['LogDestinationNotFoundException' 또는 'LogDestination에 대한 액세스가 거부됨' 오류](#)
- [Amazon S3 버킷 정책 제한 초과](#)
- [LogDestination 전송 불가](#)

불완전한 흐름 로그 레코드

문제

흐름 로그 레코드가 불완전하거나 더 이상 게시되지 않습니다.

원인

흐름 로그를 CloudWatch Logs 로그 그룹으로 전달하는 데 문제가 있을지도 모릅니다.

Solution

Amazon EC2 콘솔 또는 Amazon VPC 콘솔에서 해당 리소스에 대한 흐름 로그(Flow Logs) 탭을 확인합니다. 자세한 내용은 [흐름 로그 보기](#) 섹션을 참조하세요. 흐름 로그 테이블의 [Status] 열에는 오류가 표시됩니다. 또는 [describe-flow-logs](#) 명령을 사용하여 DeliverLogsErrorMessage 필드에 반환된 값을 확인하십시오. 다음 중 하나의 오류가 표시될 수 있습니다.

- **Rate limited:** 이 오류는 CloudWatch Logs 조절이 적용된 경우, 즉 네트워크 인터페이스에 대한 흐름 로그 레코드의 수가 특정 시간 범위 내에 게시될 수 있는 최대 레코드의 수보다 많은 경우에 발생할 수 있습니다. 이 오류는 만들 수 있는 CloudWatch Logs 로그 그룹 수 할당량에 도달한 경우에 발생하기도 합니다. 자세한 내용은 Amazon CloudWatch 사용 설명서의 [CloudWatch Service Quotas](#)를 참조하세요.
- **Access error:** 이 오류는 다음과 같은 이유로 발생할 수 있습니다.
 - 흐름 로그용 IAM 역할에 CloudWatch log 그룹에 흐름 로그를 게시할 권한이 없습니다.
 - IAM 역할이 흐름 로그 서비스와 신뢰 관계를 갖지 않습니다.
 - 신뢰 관계는 흐름 로그 서비스를 주체로 지정하지 않습니다.

자세한 내용은 [CloudWatch Logs에 흐름 로그를 게시하는 IAM 역할](#) 단원을 참조하십시오.

- Unknown error: 흐름 로그 서비스에서 내부 오류가 발생했습니다.

흐름 로그가 활성화되었지만 흐름 로그 레코드 또는 로그 그룹이 없음

문제

흐름 로그를 생성했고 Amazon VPC 또는 Amazon EC2 콘솔에서 흐름 로그를 Active로 표시합니다. 하지만 CloudWatch Logs에서 어떠한 로그 스트림도 볼 수 없거나 Amazon S3 버킷에서 로그 파일을 볼 수 없습니다.

가능한 원인

- 흐름 로그가 아직 생성되는 중입니다. 경우에 따라 흐름 로그를 생성한 후 로그 그룹이 생성되고 데이터가 표시되기까지 10분 이상 걸릴 수 있습니다.
- 네트워크 인터페이스에 대해 기록된 트래픽이 아직 없습니다. CloudWatch Logs의 로그 그룹은 트래픽이 기록될 때만 생성됩니다.

Solution

로그 그룹이 생성되거나 트래픽이 기록될 때까지 몇 분 정도 기다리세요.

'LogDestinationNotFoundException' 또는 'LogDestination에 대한 액세스가 거부됨' 오류

문제

흐름 로그를 생성할 때 Access Denied for LogDestination 또는 LogDestinationNotFoundException 오류가 발생합니다.

가능한 원인

- Amazon S3 버킷에 데이터를 게시하는 흐름 로그를 생성하는 경우 이 오류는 지정된 S3 버킷을 찾을 수 없거나 버킷 정책에서 로그를 버킷에 전달할 수 없음을 나타냅니다.
- Amazon CloudWatch Logs에 데이터를 게시하는 흐름 로그를 생성하는 경우 이 오류는 IAM 역할이 로그를 로그 그룹에 전달할 수 없음을 나타냅니다.

Solution

- Amazon S3에 게시하는 경우 기존 S3 버킷에 ARN을 지정했는지, 그리고 그 ARN의 형식이 올바른지 확인합니다. S3 버킷을 소유하지 않은 경우 [버킷 정책](#)이 필수 권한을 보유하고 있고 ARN에서 계정 ID와 버킷 이름을 올바르게 사용하는지 확인합니다.
- CloudWatch Logs Logs에 게시하는 경우 [IAM 역할](#)에 필수 권한이 있는지 확인합니다.

Amazon S3 버킷 정책 제한 초과

문제

흐름 로그를 생성할 때 LogDestinationPermissionIssueException 오류가 발생합니다.

가능한 원인

Amazon S3 버킷 정책은 크기가 20KB로 제한됩니다.

Amazon S3 버킷에 게시하는 흐름 로그가 생성될 때마다 폴더 경로를 포함하는 지정된 버킷 ARN을 버킷 정책의 Resource 요소에 자동으로 추가합니다.

동일한 버킷에 게시하는 여러 개의 흐름 로그를 생성하면 버킷 정책 제한을 초과할 수 있습니다.

Solution

- 더 이상 필요 없는 흐름 로그 항목을 제거하여 버킷 정책을 정리합니다.
- 개별 흐름 로그 항목을 다음으로 대체하여 전체 버킷에 권한을 부여합니다.

```
arn:aws:s3:::bucket_name/*
```

전체 버킷에 권한을 부여할 경우, 새 흐름 로그 구독이 버킷 정책에 새 권한을 추가합니다.

LogDestination 전송 불가

문제

흐름 로그를 생성할 때 LogDestination <bucket name> is undeliverable 오류가 발생합니다.

가능한 원인

대상 Amazon S3 버킷은 AWS KMS(SSE-KMS)로 서버 측 암호화를 사용하여 암호화되며 버킷의 기본 암호화는 KMS 키 ID입니다.

Solution

값은 KMS 키 ARN이어야 합니다. 기본 S3 암호화 유형을 KMS 키 ID에서 KMS 키 ARN으로 변경합니다. 자세한 내용은 Amazon Simple Storage Service 사용 설명서의 [기본 암호화 사용 설정](#)을 참조하세요.

VPC의 CloudWatch 지표

Amazon VPC에서는 VPC에 대한 데이터를 Amazon CloudWatch에 게시합니다. 지표라고 알려진 정렬된 시계열 데이터 집합으로 VPC에 대한 통계를 검색할 수 있습니다. 지표를 모니터링할 변수, 데이터는 시간에 따른 해당 변수의 값으로 생각하세요. 자세한 설명은 [Amazon CloudWatch 사용자 가이드](#)를 참조하세요.

내용

- [NAU 지표 및 차원](#)
- [NAU 모니터링 활성화 또는 비활성화](#)
- [NAU CloudWatch 경보 예시](#)

NAU 지표 및 차원

[네트워크 주소 사용량\(NAU\)](#)은 VPC 크기에 대한 계획을 세우고 이를 모니터링하는 데 도움이 되도록 가상 네트워크의 리소스에 적용되는 지표입니다. NAU를 모니터링하는 비용은 없습니다. VPC에 대한 NAU 또는 피어링된 NAU 할당량이 소진되는 경우 새 EC2 인스턴스를 시작하거나 새 리소스(예: Network Load Balancer, VPC 엔드포인트, Lambda 함수, Transit Gateway Attachment 또는 NAT 게이트웨이)를 프로비저닝할 수 없기 때문에 NAU 모니터링이 도움이 됩니다.

VPC에 대한 네트워크 주소 사용량 모니터링을 활성화한 경우 Amazon VPC에서는 NAU와 관련된 지표를 Amazon CloudWatch로 보냅니다. VPC의 크기는 VPC에 포함된 NAU(네트워크 주소 사용량) 단위 수로 측정합니다.

이러한 지표를 사용하여 VPC 성장 속도를 파악하거나, VPC에서 언제 크기 제한에 도달할지 예측하거나, 크기 임계값을 초과할 때 경보를 생성할 수 있습니다.

AWS/EC2네임스페이스에는 NAU 모니터링에 대한 다음과 같은 지표가 포함됩니다.

지표	설명
NetworkAddressUsage	<p>VPC당 NAU 개수.</p> <p>보고 기준</p> <ul style="list-style-type: none"> • 24시간마다. <p>측정기준</p> <ul style="list-style-type: none"> • 이름: Per-VPC Metrics, 값: VPC ID.
NetworkAddressUsagePeered	<p>VPC 및 피어링되는 모든 VPC에 대한 NAU 개수.</p> <p>보고 기준</p> <ul style="list-style-type: none"> • 24시간마다. <p>측정기준</p> <ul style="list-style-type: none"> • 이름: Per-VPC Metrics, 값: VPC ID.

AWS/Usage네임스페이스에는 NAU 모니터링에 대한 다음과 같은 지표가 포함됩니다.

지표	설명
ResourceCount	<p>VPC당 NAU 개수.</p> <p>보고 기준</p> <ul style="list-style-type: none"> • 24시간마다. <p>측정기준</p> <ul style="list-style-type: none"> • 이름: Service, 값: EC2 • 이름: Type, 값: Resource

지표	설명
	<ul style="list-style-type: none"> 이름: Resource, 값: VPC ID. 이름: Class, 값: NetworkAddressUsage
ResourceCount	<p>VPC 및 피어링되는 모든 VPC에 대한 NAU 개수.</p> <p>보고 기준</p> <ul style="list-style-type: none"> 24시간마다. <p>측정기준</p> <ul style="list-style-type: none"> 이름: Service, 값: EC2 이름: Type, 값: Resource 이름: Resource, 값: VPC ID. 이름: Class, 값: NetworkAddressUsagePeered
ResourceCount	<p>VPC 전체의 NAU 사용량 결합 보기.</p> <p>보고 기준</p> <ul style="list-style-type: none"> 24시간마다. <p>측정기준</p> <ul style="list-style-type: none"> 이름: Service, 값: EC2 이름: Type, 값: Resource 이름: Resource, 값: VPC 이름: Class, 값: NetworkAddressUsage

지표	설명
ResourceCount	<p>피어링된 VPC의 NAU 사용량 결합 보기.</p> <p>보고 기준</p> <ul style="list-style-type: none"> • 24시간마다. <p>측정기준</p> <ul style="list-style-type: none"> • 이름: Service, 값: EC2 • 이름: Type, 값: Resource • 이름: Resource, 값: VPC • 이름: Class, 값: NetworkAddressUsagePeered

NAU 모니터링 활성화 또는 비활성화

CloudWatch에서 NAU 지표를 보려면 먼저 모니터링할 각 VPC의 모니터링을 활성화해야 합니다.

NAU 모니터링을 활성화하거나 비활성화하는 방법

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 사용자 VPC(Your VPCs)를 선택합니다.
3. VPC의 확인란을 선택합니다.
4. Actions(작업), Edit VPC settings(VPC 설정 편집)를 선택합니다.
5. 다음 중 하나를 수행하십시오.
 - 모니터링을 활성화하려면 Network mapping units metrics settings(네트워크 매핑 단위 지표 설정), Enable network address usage metrics(네트워크 주소 사용량 지표 활성화)를 선택합니다.
 - 모니터링을 비활성화하려면 Network mapping units metrics settings(네트워크 매핑 단위 지표 설정), Enable network address usage metrics(네트워크 주소 사용량 지표 활성화)를 선택 취소합니다.

명령줄을 사용하여 모니터링을 활성화하거나 비활성화하는 방법

- [modify-vpc-attribute](#)(AWS CLI)
- [Edit-EC2VpcAttribute](#)(AWS Tools for Windows PowerShell)

NAU CloudWatch 경보 예시

다음과 같은 AWS CLI 명령과 예제 `.json`을 사용하여 임계값이 50,000 NAU인 VPC의 NAU 사용률을 추적하는 Amazon CloudWatch 경보 및 SNS 알림을 생성할 수 있습니다. 이 샘플에서는 먼저 Amazon SNS 주제를 생성해야 합니다. 자세한 내용은 Amazon Simple Notification Service 개발자 안내서의 [Amazon SNS 시작하기](#)를 참조하십시오.

```
aws cloudwatch put-metric-alarm --cli-input-json file://nau-alarm.json
```

다음은 `nau-alarm.json`의 예제입니다.

```
{
  "Namespace": "AWS/EC2",
  "MetricName": "NetworkAddressUsage",
  "Dimensions": [{
    "Name": "Per-VPC Metrics",
    "Value": "vpc-0123456798"
  }],
  "AlarmActions": ["arn:aws:sns:us-west-1:123456789012:my_sns_topic"],
  "ComparisonOperator": "GreaterThanThreshold",
  "Period": 86400,
  "EvaluationPeriods": 1,
  "Threshold": 50000,
  "AlarmDescription": "Tracks NAU utilization of the VPC with 50k NAUs as the
threshold",
  "AlarmName": "VPC NAU Utilization",
  "Statistic": "Maximum"
}
```

Amazon Virtual Private Cloud에서의 보안

AWS는 클라우드 보안을 가장 중요하게 생각합니다. AWS 고객으로서 여러분은 가장 높은 보안 요구 사항을 충족하기 위해 설계된 데이터 센터 및 네트워크 아키텍처의 혜택을 받게 됩니다.

보안은 AWS와 여러분의 공동 책임입니다. [공동 책임 모델](#)은 이 사항을 클라우드의 보안 및 클라우드 내 보안으로 설명합니다.

- 클라우드의 보안: AWS는 AWS 클라우드에서 AWS 서비스를 실행하는 인프라를 보호할 책임이 있습니다. AWS는 안전하게 사용할 수 있는 서비스 또한 제공합니다. 타사 감사자는 [AWS 규정 준수 프로그램](#)의 일환으로 보안 효과를 정기적으로 테스트하고 검증합니다. Amazon Virtual Private Cloud에 적용되는 규정 준수 프로그램에 대한 자세한 내용은 [규정 준수 프로그램 제공 범위 내 AWS 서비스를 참조](#)하세요.
- 클라우드 내 보안 – 귀하의 책임은 귀하가 사용하는 AWS서비스에 의해 결정됩니다. 또한 귀하는 귀사의 데이터의 민감도, 귀사의 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

이 설명서는 Amazon VPC를 사용할 때 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 다음 주제에서는 보안 및 규정 준수 목적에 맞게 Amazon VPC를 구성하는 방법을 보여줍니다. 또한 Amazon VPC 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법을 배우게 됩니다.

내용

- [Amazon Virtual Private Cloud에서의 보안](#)
- [Amazon VPC용 Identity and Access Management](#)
- [Amazon VPC의 인프라 보안](#)
- [보안 그룹을 사용하여 AWS 리소스에 대한 트래픽 제어](#)
- [네트워크 ACL을 사용하여 서브넷에 대한 트래픽 제어](#)
- [Amazon Virtual Private Cloud에서의 복원성](#)
- [Amazon Virtual Private Cloud에 대한 규정 준수 확인](#)
- [VPC에 대한 보안 모범 사례](#)

Amazon Virtual Private Cloud에서의 보안

AWS [공동 책임 모델](#)은 Amazon Virtual Private Cloud에서의 데이터 보호에 적용됩니다. 이 모델에서 설명하는 것처럼 AWS는 모든 AWS 클라우드를 실행하는 글로벌 인프라를 보호할 책임이 있습니다. 사용자는 인프라에서 호스팅되는 콘텐츠를 관리해야 합니다. 사용하는 AWS 서비스의 보안 구성과 관리 작업에 대한 책임도 사용자에게 있습니다. 데이터 프라이버시에 대한 자세한 내용은 [데이터 프라이버시 FAQ](#)를 참조하세요. 유럽의 데이터 보호에 대한 자세한 내용은 AWS 보안 블로그의 [AWS 공동 책임 모델 및 GDPR](#) 블로그 게시물을 참조하세요.

데이터를 보호하려면 AWS 계정보안 인증 정보를 보호하고 AWS IAM Identity Center 또는 AWS Identity and Access Management(IAM)를 통해 개별 사용자 계정을 설정하는 것이 좋습니다. 이렇게 하면 개별 사용자에게 자신의 직무를 충실히 이행하는 데 필요한 권한만 부여됩니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정에 멀티 팩터 인증 설정(MFA)을 사용하세요.
- SSL/TLS를 사용하여 AWS 리소스와 통신하세요. TLS 1.2는 필수이며 TLS 1.3를 권장합니다.
- AWS CloudTrail로 API 및 사용자 활동 로깅을 설정하세요.
- AWS 암호화 솔루션을 AWS 서비스 내의 모든 기본 보안 컨트롤과 함께 사용하세요.
- Amazon S3에 저장된 민감한 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용하세요.
- 명령줄 인터페이스 또는 API를 통해 AWS에 액세스할 때 FIPS 140-2 검증된 암호화 모듈이 필요한 경우, FIPS 엔드포인트를 사용합니다. 사용 가능한 FIPS 엔드포인트에 대한 자세한 내용은 [FIPS\(Federal Information Processing Standard\) 140-2](#)를 참조하세요.

고객의 이메일 주소와 같은 기밀 정보나 중요한 정보는 태그나 이름 필드와 같은 자유 양식 필드에 입력하지 않는 것이 좋습니다. 여기에는 Amazon VPC 또는 기타 AWS 서비스에서 콘솔, API, AWS CLI 또는 AWS SDK를 사용하여 작업하는 경우가 포함됩니다. 이름에 사용되는 태그 또는 자유 형식 텍스트 필드에 입력하는 모든 데이터는 청구 또는 진단 로그에 사용될 수 있습니다. 외부 서버에 URL을 제공할 때 해당 서버에 대한 요청을 검증하기 위해 보안 인증 정보를 URL에 포함해서는 안 됩니다.

Amazon VPC의 인터넷워크 트래픽 프라이버시

Amazon Virtual Private Cloud는 Virtual Private Cloud(VPC)의 보안을 강화하고 모니터링하는 데 사용할 수 있는 여러 기능을 제공합니다.

- 보안 그룹: 보안 그룹은 리소스 수준(예: EC2 인스턴스)에서 특정 인바운드 및 아웃바운드 트래픽을 허용합니다. 인스턴스를 시작할 때 하나 이상의 보안 그룹과 연결할 수 있습니다. VPC의 각 인스턴

스는 서로 다른 보안 그룹 세트에 속할 수 있습니다. 인스턴스를 시작할 때 보안 그룹을 지정하지 않을 경우 해당 VPC에 대해 인스턴스는 기본 보안 그룹과 자동으로 연결됩니다. 자세한 내용은 [보안 그룹](#) 단원을 참조하십시오.

- 네트워크 액세스 제어 목록(ACL): 네트워크 ACL은 서브넷 수준에서 특정 인바운드 또는 아웃바운드 트래픽을 허용하거나 거부합니다. 자세한 내용은 [네트워크 ACL을 사용하여 서브넷에 대한 트래픽 제어](#) 단원을 참조하십시오.
- 흐름 로그: 흐름 로그는 VPC의 네트워크 인터페이스에서 양방향으로 이동하는 IP 트래픽에 대한 정보를 캡처합니다. VPC, 서브넷 또는 개별 네트워크 인터페이스에 대한 흐름 로그를 생성할 수 있습니다. 흐름 로그 데이터는 CloudWatch Logs 또는 Amazon S3에 게시되며 과도하게 제한하거나 과도하게 허용하는 보안 그룹과 네트워크 ACL 규칙을 진단하는 데 도움이 됩니다. 자세한 내용은 [VPC 흐름 로그를 사용하여 IP 트래픽 로깅](#) 단원을 참조하십시오.
- 트래픽 미러링: Amazon EC2 인스턴스의 탄력적 네트워크 인터페이스에서 네트워크 트래픽을 복사할 수 있습니다. 그런 다음 트래픽을 대역 외 보안 및 모니터링 어플라이언스에 보낼 수 있습니다. 자세한 내용은 [트래픽 미러링 안내서](#)를 참조하십시오.

Amazon VPC용 Identity and Access Management

AWS Identity and Access Management(IAM)는 관리자가 AWS 리소스에 대한 액세스를 안전하게 제어할 수 있도록 지원하는 AWS 서비스입니다. IAM 관리자는 어떤 사용자가 Amazon VPC 리소스를 사용할 수 있는 인증(로그인) 및 권한(권한 있음)을 받을 수 있는지를 제어합니다. IAM은 추가 비용 없이 사용할 수 있는 AWS 서비스입니다.

내용

- [고객](#)
- [ID로 인증](#)
- [정책을 사용하여 액세스 관리](#)
- [Amazon VPC가 IAM과 작동하는 방식](#)
- [Amazon VPC 정책 예](#)
- [Amazon VPC 자격 증명 및 액세스 문제 해결](#)
- [Amazon Virtual Private Cloud에 대한 AWS 관리형 정책](#)

고객

AWS Identity and Access Management(IAM)를 사용하는 방법은 Amazon VPC에서 수행하는 작업에 따라 달라집니다.

서비스 사용자 – Amazon VPC 서비스를 사용하여 작업을 수행하는 경우 필요한 자격 증명과 권한을 관리자가 제공합니다. 다른 Amazon VPC 기능을 사용하여 작업을 수행한다면 추가 권한이 필요할 수 있습니다. 액세스 권한 관리 방식을 이해하면 적절한 권한을 관리자에게 요청할 수 있습니다. Amazon VPC의 기능에 액세스할 수 없다면 [Amazon VPC 자격 증명 및 액세스 문제 해결](#)를 참조하십시오.

서비스 관리자 – 회사에서 Amazon VPC 리소스를 책임지고 있다면 사용하는 서비스에 대한 완전한 액세스 권한이 있을 것입니다. 서비스 관리자는 직원이 액세스해야 하는 Amazon VPC 기능과 리소스를 결정합니다. IAM 관리자에게 요청을 제출하여 서비스 사용자의 권한을 변경합니다. 이 페이지의 정보를 검토하여 IAM의 기본 개념을 이해하십시오. 회사가 Amazon VPC에서 IAM을 사용하는 방법에 대해 자세히 알아보려면 [Amazon VPC가 IAM과 작동하는 방식](#)을 참조하십시오.

IAM 관리자 - IAM 관리자라면 Amazon VPC에 대한 액세스 권한 관리 정책 작성 방법을 자세히 알고 싶을 것입니다. 정책 예제를 보려면 [Amazon VPC 정책 예](#)를 참조하십시오.

ID로 인증

인증은 ID 보안 인증을 사용하여 AWS에 로그인하는 방식입니다. AWS 계정 루트 사용자(이)나, IAM 사용자로 또는 IAM 역할을 수임하여 인증(AWS에 로그인)되어야 합니다.

자격 증명 소스를 통해 제공된 보안 인증 정보를 사용하여 연동 자격 증명으로 AWS에 로그인할 수 있습니다. AWS IAM Identity Center (IAM Identity Center) 사용자, 회사의 Single Sign-On 인증, Google 또는 Facebook 보안 인증이 페더레이션형 ID의 예입니다. 연동 자격 증명으로 로그인할 때 관리자가 이전에 IAM 역할을 사용하여 ID 페더레이션을 설정했습니다. 연동을 사용하여 AWS에 액세스하면 간접적으로 역할을 수임합니다.

사용자 유형에 따라 AWS Management Console 또는 AWS 액세스 포털에 로그인할 수 있습니다. AWS에 로그인하는 방법에 대한 자세한 내용은 AWS 로그인 사용 설명서의 [AWS 계정에 로그인하는 방법](#)을 참조하세요.

AWS에 프로그래밍 방식으로 액세스하는 경우, AWS에서는 보안 인증 정보를 사용하여 요청에 암호화 방식으로 서명할 수 있는 소프트웨어 개발 키트(SDK) 및 명령줄 인터페이스(CLI)를 제공합니다. AWS 도구를 사용하지 않는 경우 요청에 직접 서명해야 합니다. 권장 방법을 사용하여 요청에 직접 서명하는 방법에 대한 자세한 내용은 IAM 사용 설명서의 [AWS API 요청에 서명](#)을 참조하세요.

사용하는 인증 방법에 상관없이 추가 보안 정보를 제공해야 할 수도 있습니다. 예를 들어, AWS(은)는 다중 인증(MFA)을 사용하여 계정의 보안을 강화하는 것을 권장합니다. 자세한 내용은 AWS IAM Identity Center 사용 설명서의 [다중 인증](#) 및 IAM 사용 설명서의 [AWS에서 다중 인증\(MFA\) 사용](#)을 참조하세요.

AWS 계정 루트 사용자

AWS 계정(을)를 생성할 때는 해당 계정의 모든 AWS 서비스 및 리소스에 대한 완전한 액세스 권한이 있는 단일 로그인 ID로 시작합니다. 이 자격 증명은 AWS 계정 루트 사용자라고 하며, 계정을 생성할 때 사용한 이메일 주소와 암호로 로그인하여 액세스합니다. 일상적인 태스크에 루트 사용자를 사용하지 않을 것을 강력히 권장합니다. 루트 사용자 보안 인증 정보를 보호하고 루트 사용자만 수행할 수 있는 태스크를 수행하는 데 사용하세요. 루트 사용자로 로그인해야 하는 전체 작업 목록은 IAM 사용 설명서의 [루트 사용자 보안 인증이 필요한 태스크](#)를 참조하세요.

IAM 사용자 및 그룹

[IAM 사용자](#)는 단일 개인 또는 애플리케이션에 대한 특정 권한을 가지고 있는 AWS 계정 내 자격 증명입니다. 가능하면 암호 및 액세스 키와 같은 장기 자격 증명이 있는 IAM 사용자를 생성하는 대신 임시 자격 증명을 사용하는 것이 좋습니다. 하지만 IAM 사용자의 장기 자격 증명이 필요한 특정 사용 사례가 있는 경우 액세스 키를 교체하는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 [장기 보안 인증이 필요한 사용 사례의 경우 정기적으로 액세스 키 교체](#)를 참조하세요.

[IAM 그룹](#)은 IAM 사용자 컬렉션을 지정하는 자격 증명입니다. 사용자는 그룹으로 로그인할 수 없습니다. 그룹을 사용하여 여러 사용자의 권한을 한 번에 지정할 수 있습니다. 그룹을 사용하면 대규모 사용자 집합의 권한을 더 쉽게 관리할 수 있습니다. 예를 들어, IAMAdmins라는 그룹이 있고 이 그룹에 IAM 리소스를 관리할 권한을 부여할 수 있습니다.

사용자는 역할과 다릅니다. 사용자는 한 사람 또는 애플리케이션과 고유하게 연결되지만, 역할은 해당 역할이 필요한 사람이라면 누구나 수임할 수 있습니다. 사용자는 영구적인 장기 보안 인증을 가지고 있지만, 역할은 임시 보안 인증만 제공합니다. 자세한 정보는 IAM 사용 설명서의 [IAM 사용자를 만들어야 하는 경우\(역할이 아님\)](#)를 참조하세요.

IAM 역할

[IAM 역할](#)은 특정 권한을 가지고 있는 AWS 계정 계정 내 ID입니다. IAM 사용자와 유사하지만, 특정 개인과 연결되지 않습니다. [역할 전환](#)하여 AWS Management Console에서 IAM 역할을 임시로 수임할 수 있습니다. AWS CLI 또는 AWSAPI 태스크를 호출하거나 사용자 지정 URL을 사용하여 역할을 수임할 수 있습니다. 역할 사용 방법에 대한 자세한 정보는 IAM 사용 설명서의 [IAM 역할 사용](#)을 참조하세요.

임시 보안 인증이 있는 IAM 역할은 다음과 같은 상황에서 유용합니다.

- 페더레이션 사용자 액세스 - 연동 자격 증명에 권한을 부여하려면 역할을 생성하고 해당 역할의 권한을 정의합니다. 연동 자격 증명이 인증되면 역할이 연결되고 역할에 정의된 권한이 부여됩니다. 페더레이션 역할에 대한 자세한 내용은 IAM 사용 설명서의 [타사 자격 증명 공급자의 역할 만들기](#)를 참조하세요. IAM Identity Center를 사용하는 경우 권한 세트를 구성합니다. 인증 후 아이덴티티가 액세스할 수 있는 항목을 제어하기 위해 IAM Identity Center는 권한 세트를 IAM의 역할과 연관 짓습니다. 권한 세트에 대한 자세한 내용은 AWS IAM Identity Center 사용 설명서의 [권한 세트](#)를 참조하세요.
- 임시 IAM 사용자 권한 - IAM 사용자 또는 역할은 IAM 역할을 수임하여 특정 태스크에 대한 다양한 권한을 임시로 받을 수 있습니다.
- 크로스 계정 액세스 - IAM 역할을 사용하여 다른 계정의 사용자(신뢰할 수 있는 보안 주체)가 내 계정의 리소스에 액세스하도록 허용할 수 있습니다. 역할은 계정 간 액세스를 부여하는 기본적인 방법입니다. 그러나 일부 AWS 서비스를 사용하면 정책을 리소스에 직접 연결할 수 있습니다(역할을 프록시로 사용하는 대신). 크로스 계정 액세스를 위한 역할과 리소스 기반 정책의 차이점을 알아보려면 IAM 사용 설명서의 [IAM 역할과 리소스 기반 정책의 차이](#)를 참조하세요.
- 교차 서비스 액세스 - 일부 AWS 서비스는 다른 AWS 서비스의 기능을 사용합니다. 예컨대, 어떤 서비스에서 호출을 수행하면 일반적으로 해당 서비스는 Amazon EC2에서 애플리케이션을 실행하거나 Amazon S3에 객체를 저장합니다. 서비스는 호출하는 보안 주체의 권한을 사용하거나, 서비스 역할을 사용하거나, 또는 서비스 연결 역할을 사용하여 이 작업을 수행할 수 있습니다.
- 전달 액세스 세션(FAS) - IAM 사용자 또는 역할을 사용하여 AWS에서 작업을 수행하는 사람은 보안 주체로 간주됩니다. 일부 서비스를 사용하는 경우 다른 서비스에서 다른 작업을 시작하는 작업을 수행할 수 있습니다. FAS는 AWS 서비스를 직접 호출하는 보안 주체의 권한과 요청하는 AWS 서비스를 함께 사용하여 다운스트림 서비스에 대한 요청을 수행합니다. FAS 요청은 서비스에서 완료를 위해 다른 AWS 서비스 또는 리소스와의 상호 작용이 필요한 요청을 받은 경우에만 이루어 집니다. 이 경우 두 작업을 모두 수행할 수 있는 권한이 있어야 합니다. FAS 요청 시 정책 세부 정보는 [전달 액세스 세션](#)을 참조하세요.
- 서비스 역할 - 서비스 역할은 서비스가 사용자를 대신하여 태스크를 수행하기 위해 맡는 [IAM 역할](#)입니다. IAM 관리자는 IAM 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 자세한 정보는 IAM 사용 설명서의 [AWS 서비스에 대한 권한을 위임할 역할 생성](#)을 참조하세요.
- 서비스 연결 역할 - 서비스 연결 역할은 AWS 서비스에 연결된 서비스 역할의 한 유형입니다. 서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 링크 역할은 AWS 계정에 나타나고, 서비스가 소유합니다. IAM 관리자는 서비스 연결 역할의 권한을 볼 수 있지만 편집할 수는 없습니다.
- Amazon EC2에서 실행 중인 애플리케이션 - IAM 역할을 사용하여 EC2 인스턴스에서 실행되고 AWS CLI 또는 AWS API 요청을 수행하는 애플리케이션의 임시 보안 인증을 관리할 수 있습니다. 이

는 EC2 인스턴스 내에 액세스 키를 저장할 때 권장되는 방법입니다. EC2 인스턴스에 AWS 역할을 할당하고 해당 역할을 모든 애플리케이션에서 사용할 수 있도록 하려면 인스턴스에 연결된 인스턴스 프로파일을 생성합니다. 인스턴스 프로파일에는 역할이 포함되어 있으며 EC2 인스턴스에서 실행되는 프로그램이 임시 보안 인증을 얻을 수 있습니다. 자세한 정보는 IAM 사용 설명서의 [IAM 역할을 사용하여 Amazon EC2 인스턴스에서 실행되는 애플리케이션에 권한 부여](#)를 참조하세요.

IAM 역할을 사용할지 또는 IAM 사용자를 사용할지를 알아보려면 IAM 사용 설명서의 [IAM 역할\(사용자 대신\)](#)을 생성하는 경우를 참조하십시오.

정책을 사용하여 액세스 관리

정책을 생성하고 AWSID 또는 리소스에 연결하여 AWS내 액세스를 제어합니다. 정책은 ID 또는 리소스와 연결될 때 해당 권한을 정의하는 AWS의 객체입니다. AWS는 보안 주체(사용자, 루트 사용자 또는 역할 세션)가 요청을 보낼 때 이러한 정책을 평가합니다. 정책에서 권한은 요청이 허용되거나 거부되는지를 결정합니다. 대부분의 정책은 AWS에 JSON 문서로 저장됩니다. JSON 정책 문서의 구조와 콘텐츠에 대한 자세한 정보는 IAM 사용 설명서의 [JSON 정책 개요](#)를 참조하세요.

관리자는 AWSJSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지를 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

기본적으로, 사용자와 역할에는 어떠한 권한도 없습니다. 사용자에게 사용자가 필요한 리소스에서 작업을 수행할 권한을 부여하려면 IAM 관리자가 IAM 정책을 생성하면 됩니다. 그런 다음 관리자가 IAM 정책을 역할에 추가하고, 사용자가 역할을 수입할 수 있습니다.

IAM 정책은 작업을 수행하기 위해 사용하는 방법과 상관없이 작업에 대한 권한을 정의합니다. 예를 들어, iam:GetRole태스크를 허용하는 정책이 있다고 가정합니다. 해당 정책이 있는 사용자는 AWS Management Console, AWS CLI 또는 AWSAPI에서 역할 정보를 가져올 수 있습니다.

ID 기반 정책

ID 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 자격 증명에 연결할 수 있는 JSON 권한 정책 문서입니다. 이러한 정책은 사용자와 역할이 어떤 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는지를 제어합니다. 자격 증명 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 [IAM 정책 생성](#)을 참조하세요.

자격 증명 기반 정책은 인라인 정책 또는 관리형 정책으로 한층 더 분류할 수 있습니다. 인라인 정책은 단일 사용자, 그룹 또는 역할에 직접 포함됩니다. 관리형 정책은 AWS 계정에 속한 다수의 사용자, 그룹 및 역할에 독립적으로 추가할 수 있는 정책입니다. 관리형 정책에는 AWS관리형 정책과 고객 관리

형 정책이 포함되어 있습니다. 관리형 정책 또는 인라인 정책을 선택하는 방법을 알아보려면 IAM 사용 설명서의 [관리형 정책과 인라인 정책의 선택](#)을 참조하세요.

리소스 기반 정책

리소스 기반 정책은 리소스에 연결하는 JSON 정책 설명서입니다. 리소스 기반 정책의 예는 IAM 역할 신뢰 정책과 Amazon S3 버킷 정책입니다. 리소스 기반 정책을 지원하는 서비스에서 서비스 관리자는 이러한 정책을 사용하여 특정 리소스에 대한 액세스를 통제할 수 있습니다. 정책이 연결된 리소스의 경우 정책은 지정된 보안 주체가 해당 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는지를 정의합니다. 리소스 기반 정책에서 [보안 주체를 지정](#)해야 합니다. 보안 주체에는 계정, 사용자, 역할, 페더레이션 사용자 또는 AWS 서비스가 포함될 수 있습니다.

리소스 기반 정책은 해당 서비스에 있는 인라인 정책입니다. 리소스 기반 정책에서는 IAM의 AWS 관리형 정책을 사용할 수 없습니다.

액세스 제어 목록(ACLs)

액세스 제어 목록(ACL)은 어떤 보안 주체(계정 멤버, 사용자 또는 역할)가 리소스에 액세스할 수 있는 권한을 가지고 있는지를 제어합니다. ACLs는 JSON 정책 문서 형식을 사용하지 않지만 리소스 기반 정책과 유사합니다.

Amazon S3, AWS WAF 및 Amazon VPC는 ACL을 지원하는 대표적인 서비스입니다. ACL에 대해 자세히 알아보려면 Amazon Simple Storage Service 개발자 안내서의 [액세스 제어 목록\(ACL\) 개요](#)를 참조하세요.

기타 정책 타입

AWS는 비교적 일반적이지 않은 추가 정책 유형을 지원합니다. 이러한 정책 타입은 더 일반적인 정책 타입에 따라 사용자에게 부여되는 최대 권한을 설정할 수 있습니다.

- 권한 경계 – 권한 경계는 보안 인증 기반 정책에 따라 IAM 엔터티(IAM 사용자 또는 역할)에 부여할 수 있는 최대 권한을 설정하는 고급 기능입니다. 개체에 대한 권한 경계를 설정할 수 있습니다. 그 결과로 얻는 권한은 엔터티의 자격 증명 기반 정책과 그 권한 경계의 교집합입니다. Principal 필드에서 사용자나 역할을 보안 주체로 지정하는 리소스 기반 정책은 권한 경계를 통해 제한되지 않습니다. 이러한 정책 중 하나에 포함된 명시적 거부 허용을 재정의합니다. 권한 경계에 대한 자세한 정보는 IAM 사용 설명서의 [IAM 엔터티에 대한 권한 경계](#)를 참조하세요.
- 서비스 제어 정책(SCP) – SCP는 AWS Organizations에서 조직 또는 조직 단위(OU)에 최대 권한을 지정하는 JSON 정책입니다. AWS Organizations는 기업이 소유하는 여러 개의 AWS 계정을 그룹화하고 중앙에서 관리하기 위한 서비스입니다. 조직에서 모든 기능을 활성화할 경우 서비스 제

어 정책(SCP)을 임의의 또는 모든 계정에 적용할 수 있습니다. SCP는 각 AWS 계정 루트 사용자를 비롯하여 멤버 계정의 엔터티에 대한 권한을 제한합니다. 조직 및 SCP에 대한 자세한 정보는 AWS Organizations 사용 설명서의 [SCP 작동 방식](#)을 참조하세요.

- 세션 정책 – 세션 정책은 역할 또는 페더레이션 사용자에게 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 결과적으로 얻는 세션의 권한은 사용자 또는 역할 자격 증명 기반 정책의 교차 및 세션 정책입니다. 또한 권한을 리소스 기반 정책에서 가져올 수도 있습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 자세한 정보는 IAM 사용 설명서의 [세션 정책](#)을 참조하세요.

여러 정책 타입

여러 정책 타입이 요청에 적용되는 경우 결과 권한은 이해하기가 더 복잡합니다. 여러 정책 유형이 관련될 때 AWS가 요청을 허용할지 여부를 결정하는 방법을 알아보려면 IAM 사용 설명서의 [정책 평가 로직](#)을 참조하세요.

Amazon VPC가 IAM과 작동하는 방식

IAM을 사용하여 Amazon VPC에 대한 액세스를 관리하기 전에 Amazon VPC에서 사용할 수 있는 IAM 기능을 이해해야 합니다. Amazon VPC 및 기타 AWS 서비스에서 IAM을 사용하는 방법을 개괄적으로 알아보려면 IAM 사용 설명서의 [IAM으로 작업하는 AWS 서비스](#)를 참조하세요.

목차

- [작업](#)
- [리소스](#)
- [조건 키](#)
- [Amazon VPC 리소스 기반 정책](#)
- [태그 기반 권한 부여](#)
- [IAM 역할](#)

IAM 자격 증명 기반 정책을 사용하면 허용 또는 거부된 작업을 지정할 수 있습니다. 일부 작업의 경우 작업이 허용 또는 거부되는 리소스 및 조건을 지정할 수 있습니다. Amazon VPC는 특정 작업, 리소스 및 조건 키를 지원합니다. JSON 정책에서 사용하는 모든 요소에 대해 알아보려면 IAM 사용 설명서의 [IAM JSON 정책 요소 참조](#)를 참조하세요.

작업

관리자는 AWSJSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지를 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

JSON 정책의 Action요소는 정책에서 액세스를 허용하거나 거부하는 데 사용할 수 있는 태스크를 설명합니다. 일반적으로 정책 작업의 이름은 연결된 AWSAPI 작업의 이름과 동일합니다. 일치하는 API 작업이 없는 권한 전용 작업 같은 몇 가지 예외도 있습니다. 정책에서 여러 작업이 필요한 몇 가지 작업도 있습니다. 이러한 추가 작업을 일컬어 종속 작업이라고 합니다.

연결된 작업을 수행할 수 있는 권한을 부여하기 위한 정책에 작업을 포함시킵니다.

Amazon VPC는 API 네임스페이스를 Amazon EC2와 공유합니다. Amazon VPC의 정책 작업은 작업 앞에 ec2: 접두사를 사용합니다. 예를 들어 CreateVpc API 작업을 사용하여 사용자에게 VPC를 생성할 수 있는 권한을 부여하려면 ec2:CreateVpc 작업에 대한 액세스 권한을 부여합니다. 정책 설명에는 Action 또는 NotAction 요소가 반드시 추가되어야 합니다.

단일 명령문에서 여러 작업을 지정하려면 다음 예제와 같이 쉼표로 구분합니다.

```
"Action": [
    "ec2:action1",
    "ec2:action2"
]
```

와일드카드(*)를 사용하여 여러 작업을 지정할 수 있습니다. 예를 들어, Describe라는 단어로 시작하는 모든 작업을 지정하려면 다음 작업을 포함합니다.

```
"Action": "ec2:Describe*"
```

Amazon VPC 작업 목록을 보려면 서비스 승인 참조의 [Amazon EC2에서 정의한 작업을](#) 참조하세요.

리소스

관리자는 AWSJSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지를 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지 지정할 수 있습니다.

Resource JSON 정책 요소는 작업이 적용되는 하나 이상의 개체를 지정합니다. 문장에는 Resource 또는 NotResource 요소가 반드시 추가되어야 합니다. 모범 사례에 따라 [Amazon 리소스 이름\(ARN\)](#)을 사용하여 리소스를 지정합니다. 리소스 수준 권한이라고 하는 특정 리소스 타입을 지원하는 작업에 대해 이 작업을 수행할 수 있습니다.

작업 나열과 같이 리소스 수준 권한을 지원하지 않는 작업의 경우, 와일드카드(*)를 사용하여 해당 문이 모든 리소스에 적용됨을 나타냅니다.

```
"Resource": "*"
```

VPC 리소스에는 다음 예제와 같이 ARN이 있습니다.

```
arn:${Partition}:ec2:${Region}:${Account}:vpc/${VpcId}
```

예를 들어, 명령문에서 vpc-1234567890abcdef0 VPC를 지정하려면 다음 예제에 표시된 ARN을 사용합니다.

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:vpc/vpc-1234567890abcdef0"
```

특정 리전에서 특정 계정에 속하는 모든 VPC를 지정하려면 와일드카드(*)를 사용합니다.

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:vpc/*"
```

리소스 생성 작업과 같은 일부 Amazon VPC 작업은 특정 리소스에서 수행할 수 없습니다. 이러한 경우 와일드카드(*)를 사용해야 합니다.

```
"Resource": "*"
```

다양한 Amazon EC2 API 작업에는 여러 리소스가 관여합니다. 단일 문에서 여러 리소스를 지정하려면 ARN을 쉼표로 구분합니다.

```
"Resource": [
    "resource1",
    "resource2"
]
```

Amazon VPC 리소스 유형 및 해당 ARN 목록을 보려면 서비스 승인 참조의 [Amazon EC2에서 정의한 리소스 유형](#)을 참조하세요.

조건 키

관리자는 AWSJSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지를 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지 지정할 수 있습니다.

Condition 요소(또는 Condition 블록)를 사용하면 정책이 발효되는 조건을 지정할 수 있습니다. Condition 요소는 옵션입니다. 같거나 작음과 같은 [조건 연산자](#)를 사용하여 정책의 조건을 요청의 값과 일치시키는 조건식을 생성할 수 있습니다.

한 문에서 여러 Condition요소를 지정하거나 단일 Condition요소에서 여러 키를 지정하는 경우 AWS는 논리적 AND태스크를 사용하여 평가합니다. 단일 조건 키의 여러 값을 지정하는 경우 AWS는 논리적 OR태스크를 사용하여 조건을 평가합니다. 명문의 권한을 부여하기 전에 모든 조건을 충족해야 합니다.

조건을 지정할 때 자리 표시자 변수를 사용할 수도 있습니다. 예를 들어, IAM 사용자에게 IAM 사용자 이름으로 태그가 지정된 경우에만 리소스에 액세스할 수 있는 권한을 부여할 수 있습니다. 자세한 정보는 IAM 사용 설명서의 [IAM 정책 요소: 변수 및 태그](#)를 참조하세요.

AWS는 전역 조건 키와 서비스별 조건 키를 지원합니다. 모든 AWS 전역 조건 키를 보려면 IAM 사용 설명서의 [AWS 전역 조건 컨텍스트 키](#)를 참조하세요.

모든 Amazon EC2 작업은 `aws:RequestedRegion` 및 `ec2:Region` 조건 키를 지원합니다. 자세한 내용은 [예: 특정 리전에 대한 액세스 제한](#)을 참조하세요.

Amazon VPC는 자체 조건 키 세트를 정의하며 일부 전역 조건 키 사용도 지원합니다. Amazon VPC 조건 키 목록을 보려면 서비스 승인 참조의 [Amazon EC2에 사용되는 조건 키](#)를 참조하세요. 조건 키를 사용할 수 있는 작업과 리소스를 알아보려면 [Amazon EC2에서 정의한 작업](#)을 참조하세요.

Amazon VPC 리소스 기반 정책

리소스 기반 정책은 지정된 보안 주체가 Amazon VPC 리소스에 대해 수행할 수 있는 작업 및 관련 조건을 지정하는 JSON 정책 문서입니다.

교차 계정 액세스를 활성화하려는 경우 전체 계정이나 다른 계정의 IAM 개체를 [리소스 기반 정책의 보안 주체](#)로 지정할 수 있습니다. 리소스 기반 정책에 교차 계정 보안 주체를 추가하는 것은 트러스트 관계 설정의 절반밖에 되지 않는다는 것을 유념하십시오. 보안 주체와 리소스가 다른 AWS 계정에 있는 경우 보안 주체 엔터티가 리소스에 액세스할 권한도 부여해야 합니다. 엔터티에 보안 인증 기반 정책을 연결하여 권한을 부여합니다. 하지만 리소스 기반 정책이 동일 계정의 보안 주체에 액세스를 부여하는 경우 추가 자격 증명 기반 정책이 필요하지 않습니다. 자세한 정보는 IAM 사용 설명서의 [IAM 역할과 리소스 기반 정책의 차이](#)를 참조하세요.

태그 기반 권한 부여

태그를 Amazon VPC 리소스에 연결하거나 요청을 통해 태그를 전달할 수 있습니다. 태그를 기반으로 액세스를 제어하려면 조건 키를 사용하여 정책의 [조건 요소](#)에 태그 정보를 제공합니다. 자세한 내용은

Amazon EC2 사용 설명서의 [생성 시 리소스 태깅 및 리소스 태그를 사용하여 EC2 리소스에 대한 액세스 제어](#)를 참조하세요.

리소스의 태그를 기반으로 리소스에 대한 액세스를 제한하는 자격 증명 기반 정책의 예제는 [특정 VPC로 인스턴스 시작](#)에서 확인할 수 있습니다.

IAM 역할

[IAM 역할](#)은 특정 권한을 가지고 있는 AWS 계정 내 엔터티입니다.

임시 자격 증명 사용

임시 자격 증명을 사용하여 페더레이션을 통해 로그인하거나, IAM 역할을 맡거나, 교차 계정 역할을 맡을 수 있습니다. [AssumeRole](#) 또는 [GetFederationToken](#) 같은 AWS STS API 작업을 호출하여 임시 보안 자격 증명을 가져옵니다.

Amazon VPC는 임시 자격 증명 사용을 지원합니다.

서비스 연결 역할

[서비스 연결 역할](#)을 사용하면 AWS 서비스에서 다른 서비스의 리소스에 액세스하여 사용자 대신 작업을 완료할 수 있습니다. 서비스 연결 역할은 IAM 계정에 나타나고 서비스가 소유합니다. IAM 관리자는 서비스 연결 역할의 권한을 볼 수 있지만 편집은 할 수 없습니다.

[전송 게이트웨이](#)는 서비스 연결 역할을 지원합니다.

서비스 역할

이 기능을 사용하면 서비스가 사용자를 대신하여 [서비스 역할](#)을 수입할 수 있습니다. 이 역할을 사용하면 서비스가 다른 서비스의 리소스에 액세스해 사용자를 대신해 작업을 완료할 수 있습니다. 서비스 역할은 IAM 계정에 나타나고, 해당 계정이 소유합니다. 즉, IAM 관리자가 이 역할에 대한 권한을 변경할 수 있습니다. 그러나 권한을 변경하면 서비스의 기능이 손상될 수 있습니다.

Amazon VPC는 흐름 로그에 대한 서비스 역할을 지원합니다. 흐름 로그를 만들 때 흐름 로그 서비스에서 CloudWatch Logs에 액세스할 수 있는 역할을 선택해야 합니다. 자세한 내용은 [the section called "CloudWatch Logs에 흐름 로그를 게시하는 IAM 역할"](#) 단원을 참조하십시오.

Amazon VPC 정책 예

기본적으로 IAM 역할은 VPC 리소스를 생성하거나 수정할 수 있는 권한이 없습니다. 또한 AWS Management Console, AWS CLI 또는 AWS API를 사용해 태스크를 수행할 수 없습니다. IAM 관리자는 지정된 리소스에서 특정 API 작업을 수행할 수 있는 권한을 역할에 부여하는 IAM 정책을 생성해야 합니다. 그런 다음 관리자는 해당 권한이 필요한 IAM 역할에 이러한 정책을 연결해야 합니다.

이러한 예제 JSON 정책 문서를 사용하여 IAM 자격 증명 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 [IAM 정책 생성](#)을 참조하세요.

내용

- [정책 모범 사례](#)
- [Amazon VPC 콘솔 사용](#)
- [퍼블릭 서브넷이 포함된 VPC 만들기](#)
- [VPC 리소스 수정 및 삭제](#)
- [보안 그룹 관리](#)
- [보안 그룹 규칙 관리](#)
- [특정 서브넷으로 인스턴스 시작](#)
- [특정 VPC로 인스턴스 시작](#)
- [추가 Amazon VPC 정책 예](#)

정책 모범 사례

ID 기반 정책에 따라 계정에서 사용자가 Amazon VPC 리소스를 생성, 액세스 또는 삭제할 수 있는지 여부가 결정됩니다. 이 작업으로 인해 AWS 계정에 비용이 발생할 수 있습니다. 자격 증명 기반 정책을 생성하거나 편집할 때는 다음 지침과 권장 사항을 따르십시오.

- AWS 관리형 정책으로 시작하고 최소 권한을 향해 나아가기 - 사용자 및 워크로드에 권한 부여를 시작하려면 많은 일반 사용 사례에 대한 권한을 부여하는 AWS 관리형 정책을 사용합니다. AWS 계정에서 사용할 수 있습니다. 사용 사례에 고유한 AWS 고객 관리형 정책을 정의하여 권한을 줄이는 것이 좋습니다. 자세한 정보는 IAM 사용 설명서의 [AWS 관리형 정책](#) 또는 [AWS 직무에 대한 관리형 정책](#)을 참조하세요.
- 최소 권한 적용 - IAM 정책을 사용하여 권한을 설정하는 경우 태스크를 수행하는 데 필요한 권한만 부여합니다. 이렇게 하려면 최소 권한으로 알려진 특정 조건에서 특정 리소스에 대해 수행할 수 있는 작업을 정의합니다. IAM을 사용하여 권한을 적용하는 방법에 대한 자세한 정보는 IAM 사용 설명서에 있는 [IAM의 정책 및 권한](#)을 참조하세요.
- IAM 정책의 조건을 사용하여 액세스 추가 제한 - 정책에 조건을 추가하여 작업 및 리소스에 대한 액세스를 제한할 수 있습니다. 예를 들어 SSL을 사용하여 모든 요청을 전송해야 한다고 지정하는 정책 조건을 작성할 수 있습니다. AWS CloudFormation과 같이, 특정 AWS 서비스를 통해 사용되는 경우에만 서비스 작업에 대한 액세스 권한을 부여할 수도 있습니다. 자세한 정보는 IAM 사용 설명서의 [IAM JSON 정책 요소: 조건](#)을 참조하세요.

- IAM Access Analyzer를 통해 IAM 정책을 검증하여 안전하고 기능적인 권한 보장 – IAM Access Analyzer에서는 IAM 정책 언어(JSON)와 모범 사례가 정책에서 준수되도록 신규 및 기존 정책을 검증합니다. IAM Access Analyzer는 100개 이상의 정책 확인 항목과 실행 가능한 추천을 제공하여 안전하고 기능적인 정책을 작성하도록 돕습니다. 자세한 정보는 IAM 사용 설명서의 [IAM Access Analyzer 정책 검증](#)을 참조하세요.
- 다중 인증(MFA) 필요 – AWS 계정 계정에 IAM 사용자 또는 루트 사용자가 필요한 시나리오가 있는 경우 추가 보안을 위해 MFA를 설정합니다. API 작업을 직접 호출할 때 MFA가 필요하다면 정책에 MFA 조건을 추가합니다. 자세한 정보는 IAM 사용 설명서의 [MFA 보호 API 액세스 구성](#)을 참조하세요.

IAM의 모범 사례에 대한 자세한 내용은 IAM 사용 설명서의 [IAM의 보안 모범 사례](#)를 참조하세요.

Amazon VPC 콘솔 사용

Amazon VPC 콘솔에 액세스하려면 최소한의 권한 집합이 있어야 합니다. 이러한 권한은 AWS 계정에서 Amazon VPC 리소스에 대한 세부 정보를 나열하고 볼 수 있도록 허용해야 합니다. 최소 필수 권한보다 더 제한적인 자격 증명 기반 정책을 만들면 콘솔이 해당 정책에 연결된 개체(IAM 역할)에 대해 의도대로 작동하지 않습니다.

다음 정책은 VPC 콘솔에 리소스를 나열할 수 있는 권한을 역할에 부여하지만 리소스를 생성, 업데이트 또는 삭제할 수는 없습니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeAccountAttributes",
        "ec2:DescribeAddresses",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeClassicLinkInstances",
        "ec2:DescribeClientVpnEndpoints",
        "ec2:DescribeCustomerGateways",
        "ec2:DescribeDhcpOptions",
        "ec2:DescribeEgressOnlyInternetGateways",
        "ec2:DescribeFlowLogs",
        "ec2:DescribeInternetGateways",
        "ec2:DescribeManagedPrefixLists",
        "ec2:DescribeMovingAddresses",
```

```

    "ec2:DescribeNatGateways",
    "ec2:DescribeNetworkAcls",
    "ec2:DescribeNetworkInterfaceAttribute",
    "ec2:DescribeNetworkInterfacePermissions",
    "ec2:DescribeNetworkInterfaces",
    "ec2:DescribePrefixLists",
    "ec2:DescribeRouteTables",
    "ec2:DescribeSecurityGroupReferences",
    "ec2:DescribeSecurityGroups",
    "ec2:DescribeSecurityGroupRules",
    "ec2:DescribeStaleSecurityGroups",
    "ec2:DescribeSubnets",
    "ec2:DescribeTags",
    "ec2:DescribeTrafficMirrorFilters",
    "ec2:DescribeTrafficMirrorSessions",
    "ec2:DescribeTrafficMirrorTargets",
    "ec2:DescribeTransitGateways",
    "ec2:DescribeTransitGatewayVpcAttachments",
    "ec2:DescribeTransitGatewayRouteTables",
    "ec2:DescribeVpcAttribute",
    "ec2:DescribeVpcClassicLink",
    "ec2:DescribeVpcClassicLinkDnsSupport",
    "ec2:DescribeVpcEndpoints",
    "ec2:DescribeVpcEndpointConnectionNotifications",
    "ec2:DescribeVpcEndpointConnections",
    "ec2:DescribeVpcEndpointServiceConfigurations",
    "ec2:DescribeVpcEndpointServicePermissions",
    "ec2:DescribeVpcEndpointServices",
    "ec2:DescribeVpcPeeringConnections",
    "ec2:DescribeVpcs",
    "ec2:DescribeVpnConnections",
    "ec2:DescribeVpnGateways",
    "ec2:GetManagedPrefixListAssociations",
    "ec2:GetManagedPrefixListEntries"
  ],
  "Resource": "*"
}
]
}

```

AWS CLI 또는 AWS API만 호출하는 역할에는 최소 콘솔 권한을 허용할 필요가 없습니다. 그 대신, 역할이 수행해야 하는 API 작업과 일치하는 작업에만 액세스를 허용합니다.

퍼블릭 서브넷이 포함된 VPC 만들기

다음 예시에서는 역할이 VPC, 서브넷, 라우팅 테이블 및 인터넷 게이트웨이를 만들 수 있도록 설정합니다. 또한 역할은 인터넷 게이트웨이를 VPC에 연결하고 라우팅 테이블에 라우팅을 생성할 수 있습니다. ec2:ModifyVpcAttribute 작업을 통해 역할은 VPC로 시작된 각 인스턴스가 DNS 호스트 이름을 수신할 수 있도록 VPC에 대한 DNS 호스트 이름을 활성화합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "ec2:CreateVpc",
      "ec2:CreateSubnet",
      "ec2:DescribeAvailabilityZones",
      "ec2:CreateRouteTable",
      "ec2:CreateRoute",
      "ec2:CreateInternetGateway",
      "ec2:AttachInternetGateway",
      "ec2:AssociateRouteTable",
      "ec2:ModifyVpcAttribute"
    ],
    "Resource": "*"
  }
]
```

또한 위의 정책은 역할이 Amazon VPC 콘솔에서 VPC를 생성할 수 있도록 허용합니다.

VPC 리소스 수정 및 삭제

역할이 수정하거나 삭제할 수 있는 VPC 리소스를 제어할 수 있습니다. 예를 들어 역할은 다음 정책을 통해 Purpose=Test 태그가 있는 라우팅 테이블에서 작업하고 삭제할 수 있습니다. 또한 이 정책은 역할이 Purpose=Test 태그가 있는 인터넷 게이트웨이를 삭제할 수만 있도록 지정합니다. 역할은 이 태그가 없는 라우팅 테이블 또는 인터넷 게이트웨이를 사용할 수 없습니다.

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

```

    "Action": "ec2:DeleteInternetGateway",
    "Resource": "arn:aws:ec2:*:*:internet-gateway/*",
    "Condition": {
      "StringEquals": {
        "ec2:ResourceTag/Purpose": "Test"
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "ec2:DeleteRouteTable",
      "ec2:CreateRoute",
      "ec2:ReplaceRoute",
      "ec2:DeleteRoute"
    ],
    "Resource": "arn:aws:ec2:*:*:route-table/*",
    "Condition": {
      "StringEquals": {
        "ec2:ResourceTag/Purpose": "Test"
      }
    }
  }
]
}

```

보안 그룹 관리

다음 정책을 사용하여 역할은 보안 그룹을 관리할 수 있습니다. 첫 번째 문을 통해 역할은 Stack=test 태그가 있는 모든 보안 그룹을 삭제할 수 있고 Stack=test 태그가 있는 모든 보안 그룹에 대한 인바운드 및 아웃바운드 규칙을 관리할 수 있습니다. 두 번째 문은 역할이 Stack=Test 태그로 만든 보안 그룹에 태그를 지정하도록 요구합니다. 세 번째 문을 통해 역할은 보안 그룹을 만들 때 태그를 생성할 수 있습니다. 네 번째 문을 통해 역할은 모든 보안 그룹 및 보안 그룹 규칙을 볼 수 있습니다. 다섯 번째 문을 통해 역할은 VPC에서 보안 그룹을 생성할 수 있습니다.

Note

AWS CloudFormation 서비스에서 이 정책을 사용하여 필수 태그가 있는 보안 그룹을 생성할 수 없습니다. 태그가 필요한 ec2:CreateSecurityGroup 작업에서 조건을 제거하면 정책이 작동합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:RevokeSecurityGroupIngress",
        "ec2:AuthorizeSecurityGroupEgress",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:UpdateSecurityGroupRuleDescriptionsEgress",
        "ec2:RevokeSecurityGroupEgress",
        "ec2>DeleteSecurityGroup",
        "ec2:ModifySecurityGroupRules",
        "ec2:UpdateSecurityGroupRuleDescriptionsIngress"
      ],
      "Resource": "arn:aws:ec2:*:*:security-group/*",
      "Condition": {
        "StringEquals": {
          "ec2:ResourceTag/Stack": "test"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:CreateSecurityGroup",
      "Resource": "arn:aws:ec2:*:*:security-group/*",
      "Condition": {
        "StringEquals": {
          "aws:RequestTag/Stack": "test"
        },
        "ForAllValues:StringEquals": {
          "aws:TagKeys": "Stack"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:CreateTags",
      "Resource": "arn:aws:ec2:*:*:security-group/*",
      "Condition": {
        "StringEquals": {
          "ec2:CreateAction": "CreateSecurityGroup"
        }
      }
    }
  ]
}

```

```

    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "ec2:DescribeSecurityGroupRules",
      "ec2:DescribeVpcs",
      "ec2:DescribeSecurityGroups"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": "ec2:CreateSecurityGroup",
    "Resource": "arn:aws:ec2:*:*:vpc/*"
  }
]
}

```

역할이 인스턴스와 연결된 보안 그룹을 변경할 수 있도록 하려면 정책에 `ec2:ModifyInstanceAttribute` 작업을 추가합니다.

역할이 네트워크 인터페이스의 보안 그룹을 변경할 수 있도록 하려면 정책에 `ec2:ModifyNetworkInterfaceAttribute` 작업을 추가합니다.

보안 그룹 규칙 관리

다음 정책은 역할이 모든 보안 그룹 및 보안 그룹 규칙을 조회하고 특정 VPC의 보안 그룹에 대한 인바운드 및 아웃바운드 규칙을 추가 및 제거하며 지정된 VPC에 대한 규칙 설명을 수정할 권한을 부여합니다. 첫 번째 문은 `ec2:Vpc` 조건 키를 사용하여 특정 VPC 대한 권한 범위를 지정할 수 있습니다.

두 번째 문은 역할에 모든 보안 그룹, 보안 그룹 규칙 및 태그를 설명하는 권한을 부여합니다. 이를 통해 역할은 보안 그룹 규칙을 수정하기 위해 볼 수 있습니다.

```

{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "ec2:AuthorizeSecurityGroupIngress",
      "ec2:RevokeSecurityGroupIngress",
      "ec2:UpdateSecurityGroupRuleDescriptionsIngress",
      "ec2:AuthorizeSecurityGroupEgress",

```



```

    "ec2:RevokeSecurityGroupEgress",
    "ec2:UpdateSecurityGroupRuleDescriptionsEgress",
    "ec2:ModifySecurityGroupRules"
  ],
  "Resource": "arn:aws:ec2:region:account-id:security-group/*",
  "Condition": {
    "ArnEquals": {
      "ec2:Vpc": "arn:aws:ec2:region:account-id:vpc/vpc-id"
    }
  }
},
{
  "Effect": "Allow",
  "Action": [
    "ec2:DescribeSecurityGroups",
    "ec2:DescribeSecurityGroupRules",
    "ec2:DescribeTags"
  ],
  "Resource": "*"
},
{
  "Effect": "Allow",
  "Action": [
    "ec2:ModifySecurityGroupRules"
  ],
  "Resource": "arn:aws:ec2:region:account-id:security-group-rule/*"
}
]
}

```

특정 서브넷으로 인스턴스 시작

다음 정책은 역할에 인스턴스를 특정 서브넷으로 시작하고 요청에 특정 보안 그룹을 사용하는 권한을 부여합니다. 이 정책에서는 서브넷에 대한 ARN과 보안 그룹에 대한 ARN을 지정하여 이런 권한을 부여합니다. 역할이 다른 서브넷으로 인스턴스를 시작하거나 다른 보안 그룹을 사용하여 시작하려고 하면 (또 다른 정책 또는 설명문에서 역할에 그런 권한을 부여하지 않는 한) 요청이 실패하게 됩니다.

또한, 이 정책에서는 네트워크 인터페이스 리소스를 사용할 권한도 부여합니다. 서브넷으로 시작할 때 기본적으로 RunInstances 요청은 기본 네트워크 인터페이스를 생성하므로, 역할은 인스턴스를 시작할 때 이 리소스를 생성할 권한이 필요합니다.

```
{
```

```

"Version": "2012-10-17",
"Statement": [{
  "Effect": "Allow",
  "Action": "ec2:RunInstances",
  "Resource": [
    "arn:aws:ec2:region::image/ami-*",
    "arn:aws:ec2:region:account:instance/*",
    "arn:aws:ec2:region:account:subnet/subnet-id",
    "arn:aws:ec2:region:account:network-interface/*",
    "arn:aws:ec2:region:account:volume/*",
    "arn:aws:ec2:region:account:key-pair/*",
    "arn:aws:ec2:region:account:security-group/sg-id"
  ]
}]
}

```

특정 VPC로 인스턴스 시작

다음 정책에서는 역할에 특정 VPC 내에 있는 임의의 서브넷으로 인스턴스를 시작하는 권한을 부여합니다. 이 정책에서는 조건 키(ec2:Vpc)를 서브넷 리소스에 적용함으로써 이런 권한을 부여합니다.

또한, 이 정책에서는 역할에 "department=dev" 태그가 있는 AMI만 사용하여 인스턴스를 시작하는 권한을 부여합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "ec2:RunInstances",
    "Resource": "arn:aws:ec2:region:account-id:subnet/*",
    "Condition": {
      "ArnEquals": {
        "ec2:Vpc": "arn:aws:ec2:region:account-id:vpc/vpc-id"
      }
    }
  }
],
{
  "Effect": "Allow",
  "Action": "ec2:RunInstances",
  "Resource": "arn:aws:ec2:region::image/ami-*",
  "Condition": {
    "StringEquals": {

```

```

        "ec2:ResourceTag/department": "dev"
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": "ec2:RunInstances",
    "Resource": [
      "arn:aws:ec2:region:account:instance/*",
      "arn:aws:ec2:region:account:volume/*",
      "arn:aws:ec2:region:account:network-interface/*",
      "arn:aws:ec2:region:account:key-pair/*",
      "arn:aws:ec2:region:account:security-group/*"
    ]
  }
]
}

```

추가 Amazon VPC 정책 예

다음 문서에서 Amazon VPC와 관련된 추가 IAM 정책 예제를 확인할 수 있습니다.

- [관리형 접두사 목록](#)
- [트래픽 미러링](#)
- [전송 게이트웨이](#)
- [VPC 엔드포인트 및 VPC 엔드포인트 서비스](#)
- [VPC 엔드포인트 정책](#)
- [VPC 피어링](#)
- [AWS Wavelength](#)

Amazon VPC 자격 증명 및 액세스 문제 해결

다음 정보를 사용하여 Amazon VPC 및 IAM으로 작업할 때 발생할 수 있는 일반적인 문제를 진단하고 수정할 수 있습니다.

문제

- [Amazon VPC에서 작업을 수행할 권한이 없음](#)
- [iam:PassRole을 수행하도록 인증되지 않음](#)

- [내 AWS 계정 외부의 사용자가 내 Amazon VPC 리소스에 액세스할 수 있도록 허용하려고 합니다.](#)

Amazon VPC에서 작업을 수행할 권한이 없음

AWS Management Console에서 작업을 수행할 권한이 없다는 메시지가 나타나는 경우 관리자에게 문의하여 도움을 받아야 합니다. 관리자는 로그인 보안 인증 정보를 제공한 사람입니다.

다음 오류 예는 mateojackson IAM 사용자가 콘솔을 사용하여 서브넷에 대한 세부 정보를 보려고 하지만 ec2:DescribeSubnets 권한이 없는 IAM 역할에 속하는 경우에 발생합니다.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
ec2:DescribeSubnets on resource: subnet-id
```

이 경우 Mateo는 관리자에게 정책을 업데이트하여 자신에게 서브넷에 대한 액세스를 허용하도록 요청합니다.

iam:PassRole을 수행하도록 인증되지 않음

iam:PassRole 작업을 수행할 수 있는 권한이 없다는 오류가 수신되면 Amazon VPC에 역할을 전달할 수 있도록 정책을 업데이트해야 합니다.

일부 AWS 서비스에서는 새 서비스 역할 또는 서비스 연결 역할을 생성하는 대신 해당 서비스에 기존 역할을 전달할 수 있습니다. 이렇게 하려면 사용자가 서비스에 역할을 전달할 수 있는 권한을 가지고 있어야 합니다.

다음 예제 오류는 marymajor라는 IAM 사용자가 콘솔을 사용하여 Amazon VPC에서 작업을 수행하려고 하는 경우에 발생합니다. 하지만 작업을 수행하려면 서비스 역할이 부여한 권한이 서비스에 있어야 합니다. Mary는 서비스에 역할을 전달할 수 있는 권한을 가지고 있지 않습니다.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

이 경우 Mary가 iam:PassRole 작업을 수행할 수 있도록 Mary의 정책을 업데이트해야 합니다.

도움이 필요한 경우 AWS 관리자에게 문의하십시오. 관리자는 로그인 자격 증명을 제공한 사람입니다.

내 AWS 계정 외부의 사용자가 내 Amazon VPC 리소스에 액세스할 수 있도록 허용하려고 합니다.

다른 계정의 사용자 또는 조직 외부의 사람이 리소스에 액세스하는 데 사용할 수 있는 역할을 생성할 수 있습니다. 역할을 수임할 신뢰할 수 있는 사람을 지정할 수 있습니다. 리소스 기반 정책 또는 액세스

제어 목록(ACL)을 지원하는 서비스의 경우 이러한 정책을 사용하여 다른 사람에게 리소스에 대한 액세스 권한을 부여할 수 있습니다.

자세히 알아보려면 다음을 참조하십시오.

- Amazon VPC에서 이러한 기능을 지원하는지 여부를 알아보려면 [Amazon VPC가 IAM과 작동하는 방식](#)을 참조하십시오.
- 소유하고 있는 AWS 계정의 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 [자신이 소유한 다른 AWS 계정의 IAM 사용자에게 대한 액세스 권한 제공](#)을 참조하세요.
- 리소스에 대한 액세스 권한을 타사 AWS 계정에게 제공하는 방법을 알아보려면 IAM 사용 설명서의 [타사가 소유한 AWS 계정에 대한 액세스 제공](#)을 참조하세요.
- ID 페더레이션을 통해 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 [외부에서 인증된 사용자에게 액세스 권한 제공\(ID 페더레이션\)](#)을 참조하세요.
- 교차 계정 액세스를 위한 역할과 리소스 기반 정책 사용의 차이점을 알아보려면 IAM 사용 설명서의 [IAM 역할과 리소스 기반 정책의 차이](#)를 참조하십시오.

Amazon Virtual Private Cloud에 대한 AWS 관리형 정책

AWS 관리형 정책은 AWS에서 생성되고 관리되는 독립 실행형 정책입니다. AWS 관리형 정책은 사용자, 그룹 및 역할에 권한 할당을 시작할 수 있도록 많은 일반 사용 사례에 대한 권한을 제공하도록 설계되었습니다.

AWS 관리형 정책은 모든 AWS 고객이 사용할 수 있기 때문에 특정 사용 사례에 대해 최소 권한을 부여하지 않을 수 있습니다. 사용 사례에 고유한 [고객 관리형 정책](#)을 정의하여 권한을 줄이는 것이 좋습니다.

AWS 관리형 정책에서 정의한 권한은 변경할 수 없습니다. 만약 AWS가 AWS 관리형 정책에 정의된 권한을 업데이트할 경우 정책이 연결되어 있는 모든 보안 주체 엔터티(사용자, 그룹 및 역할)에도 업데이트가 적용됩니다. 새로운 AWS 서비스를 시작하거나 새로운 API 작업을 기존 서비스에 이용하는 경우 AWS가 AWS 관리형 정책을 업데이트할 가능성이 높습니다.

자세한 내용은 IAM 사용자 설명서의 [AWS 관리형 정책](#)을 참조하세요.

AWS 관리형 정책: AmazonVPCFullAccess

AmazonVPCFullAccess 정책을 IAM 자격 증명에 연결할 수 있습니다. 이 정책은 Amazon VPC 대한 전체 액세스를 허용하는 권한을 부여합니다.

이 정책의 권한을 보려면 AWS 관리형 정책 참조에서 [AmazonVPCFullAccess](#)를 참조하세요.

AWS 관리형 정책: AmazonVPCReadOnlyAccess

AmazonVPCReadOnlyAccess 정책을 IAM 자격 증명에 연결할 수 있습니다. 이 정책은 Amazon VPC 대한 읽기 전용 액세스를 허용하는 권한을 부여합니다.

이 정책의 권한을 보려면 AWS 관리형 정책 참조에서 [AmazonVPCReadOnlyAccess](#)를 참조하세요.

AWS 관리형 정책: AmazonVPCCrossAccountNetworkInterfaceOperations

AmazonVPCCrossAccountNetworkInterfaceOperations 정책을 IAM 보안 인증에 연결할 수 있습니다. 이 정책은 ID가 네트워크 인터페이스를 만들어 교차 계정 리소스에 연결할 수 있는 권한을 부여합니다.

이 정책의 권한을 보려면 AWS 관리형 정책 참조에서 [AmazonVPCCrossAccountNetworkInterfaceOperations](#)를 참조하세요.

AWS 관리형 정책에 대한 Amazon VPC 업데이트

이 서비스가 2021년 3월에 이러한 변경 사항을 추적하기 시작한 이후 Amazon VPC용 AWS 관리형 정책 업데이트에 대한 세부 정보를 확인합니다.

변경 사항	설명	날짜
the section called “AmazonVPCFullAccess” - 기존 정책에 대한 업데이트	VPC에서 사용할 수 있는 보안 그룹을 가져올 수 있는 GetSecurityGroupsForVpc 작업이 추가되었습니다.	2024년 2월 8일
the section called “AmazonVPCReadOnlyAccess” -기존 정책 업데이트	VPC에서 사용할 수 있는 보안 그룹을 가져올 수 있는 GetSecurityGroupsForVpc 작업이 추가되었습니다.	2024년 2월 8일
the section called “AmazonVPCCrossAccountNetworkInterfaceOperations” -기존 정책 업데이트	AssignIpv6Addresses 및 UnassignIpv6Addresses 작업이 추가되었으며, 이를 통해 네트워크 인터페이스와 연결된 IPv6 주소를 관리할 수 있습니다.	2023년 9월 25일

변경 사항	설명	날짜
the section called “AmazonVPCReadOnlyAccess” -기존 정책 업데이트	보안 그룹 규칙 을 볼 수 있는 DescribeSecurityGroupRules 작업이 추가되었습니다.	2021년 8월 2일
the section called “AmazonVPCFullAccess” -기존 정책 업데이트	보안 그룹 규칙 을 보고 수정할 수 있는 DescribeSecurityGroupRules 및 ModifySecurityGroupRules 작업이 추가되었습니다.	2021년 8월 2일
the section called “AmazonVPCFullAccess” -기존 정책 업데이트	통신업체 게이트웨이, IPv6 풀, 로컬 게이트웨이 및 로컬 게이트웨이 라우팅 테이블에 대한 작업이 추가되었습니다.	2021년 6월 23일
the section called “AmazonVPCReadOnlyAccess” -기존 정책 업데이트	통신업체 게이트웨이, IPv6 풀, 로컬 게이트웨이 및 로컬 게이트웨이 라우팅 테이블에 대한 작업이 추가되었습니다.	2021년 6월 23일

Amazon VPC의 인프라 보안

관리형 서비스인 Amazon Virtual Private Cloud는 AWS 글로벌 네트워크 보안으로 보호됩니다. AWS 보안 서비스와 AWS의 인프라 보호 방법에 대한 자세한 내용은 [AWS 클라우드 보안](#)을 참조하세요. 인프라 보안에 대한 모범 사례를 사용하여 AWS 환경을 설계하려면 보안 원칙 AWS Well-Architected 프레임워크의 [인프라 보호](#)를 참조하세요.

AWS에서 게시한 API 호출을 사용하여 네트워크를 통해 Amazon VPC에 액세스합니다. 고객은 다음을 지원해야 합니다.

- 전송 계층 보안(TLS) TLS 1.2는 필수이며 TLS 1.3을 권장합니다.
- DHE(Ephemeral Diffie-Hellman) 또는 ECDHE(Elliptic Curve Ephemeral Diffie-Hellman)와 같은 완전 전송 보안(PFS)이 포함된 암호 제품군 Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 요청은 액세스 키 ID 및 IAM 주체와 관련된 비밀 액세스 키를 사용하여 서명해야 합니다. 또는 [AWS Security Token Service](#)(AWS STS)을 사용하여 임시 보안 자격 증명을 생성하여 요청에 서명할 수 있습니다.

네트워크 격리

Virtual Private Cloud(VPC)는 AWS 클라우드에서 논리적으로 격리된 고유한 영역의 가상 네트워크입니다. 별도의 VPC를 사용하여 워크로드별 또는 조직체별로 인프라를 격리합니다.

서브넷은 VPC의 IP 주소 범위입니다. 인스턴스를 시작할 때 VPC의 서브넷에서 인스턴스를 시작합니다. 서브넷을 사용하여 단일 VPC 내의 애플리케이션 티어(예: 웹, 애플리케이션 및 데이터베이스)를 격리합니다. 인터넷에서 직접 액세스하면 안 되는 경우 프라이빗 서브넷을 인스턴스에 사용합니다.

[AWS PrivateLink](#)를 사용하여 VPC의 리소스에서 프라이빗 IP 주소를 사용하여 AWS 서비스에 연결하고 서비스를 VPC 직접 호스팅된 것처럼 이용하도록 할 수 있습니다. 따라서 AWS 서비스에 액세스하기 위해 인터넷 게이트웨이나 NAT 장치를 사용할 필요가 없습니다.

네트워크 트래픽 제어

EC2 인스턴스와 같은 VPC의 네트워크 트래픽을 제어하기 위해 다음 옵션을 고려해 보세요.

- [보안 그룹](#)을 VPC에 대한 네트워크 액세스를 제어하는 기본 메커니즘으로 활용합니다. 필요한 경우 [네트워크 ACL](#)을 사용하여 상태 비저장의 거친 네트워크 제어를 제공합니다. 보안 그룹은 상태 저장 패킷 필터링을 수행하고 다른 보안 그룹을 참조하는 규칙을 만들 수 있기 때문에 네트워크 ACL보다 다재다능합니다. 네트워크 ACL은 보조 제어 장치(예: 트래픽의 특정 하위 집합 거부) 또는 상위 수준의 서브넷 가드레일로 효과를 발휘할 수 있습니다. 또한 네트워크 ACL은 전체 서브넷에 적용되므로 인스턴스가 올바른 보안 그룹 없이 시작될 경우 이를 심층 방어 기능으로 사용할 수 있습니다.
- 인터넷에서 직접 액세스하면 안 되는 경우 프라이빗 서브넷을 인스턴스에 사용합니다. 프라이빗 서브넷에 있는 인스턴스에서 인터넷에 액세스하려면 Bastion Host 또는 NAT 게이트웨이를 사용합니다.
- 연결 요구 사항을 지원하기 위해 최소 네트워크 경로로 서브넷 [라우팅 테이블](#)을 구성합니다.
- 보안 그룹 또는 네트워크 인터페이스를 추가로 사용하여 일반 애플리케이션 트래픽과 별도로 Amazon EC2 인스턴스 관리 트래픽을 제어하고 감사하는 방법을 고려해 보십시오. 따라서 변경 제어를 위한 특별한 IAM 정책을 고객이 구현할 수 있으므로 보안 그룹 규칙 또는 자동화된 규칙 확인 스크립트의 변경 사항을 감사하기가 쉬워집니다. 또한 네트워크 인터페이스가 여러이면 호스트 기반 라우팅 정책을 생성하거나 서브넷에 할당된 네트워크 인터페이스에 따라 다양한 VPC 서브넷 라우팅 규칙을 활용하는 기능 등 네트워크 트래픽 제어의 옵션이 늘어납니다.

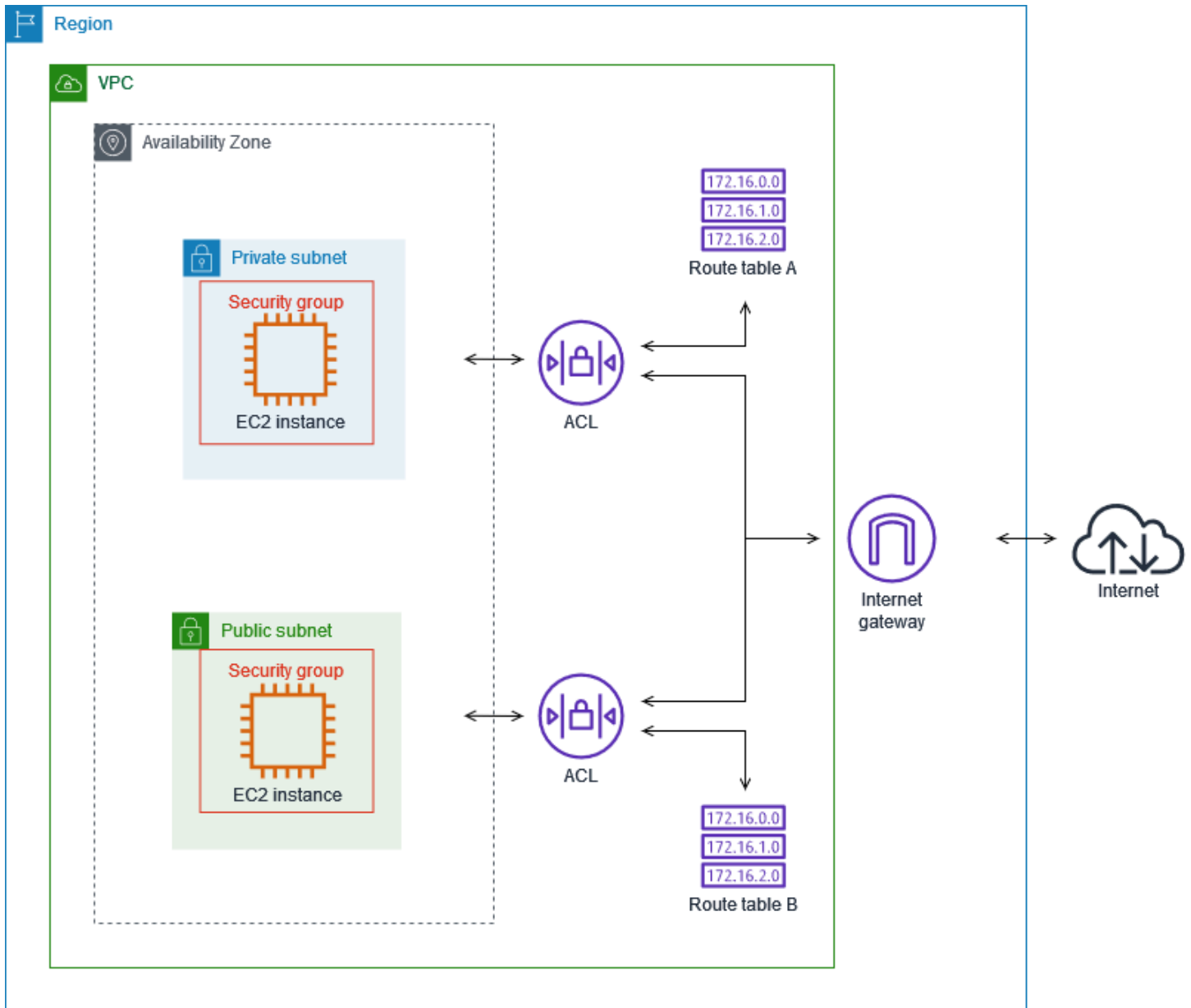
- AWS Virtual Private Network 또는 AWS Direct Connect를 사용하여 원격 네트워크에서 VPC로 프라이빗 연결을 설정합니다. 자세한 내용은 [네트워크-Amazon VPC 연결 옵션](#)을 참조하세요.
- [VPC 흐름 로그](#)를 사용하여 인스턴스에 도달하는 트래픽을 모니터링합니다.
- [AWS Security Hub](#)를 사용하여 인스턴스에서 의도하지 않게 네트워크에 액세스할 수 있는지 확인합니다.
- [AWS Network Firewall](#)을 사용하여 VPC의 서브넷을 일반적인 네트워크 위협으로부터 보호합니다.

보안 그룹 및 네트워크 ACL 비교

다음 표는 보안 그룹과 네트워크 ACL의 근본적인 차이를 요약한 것입니다.

보안 그룹	네트워크 ACL
인스턴스 레벨에서 운영됩니다.	서브넷 레벨에서 운영됩니다.
인스턴스와 연결된 경우에만 인스턴스에 적용됨	연결된 서브넷에서 배포된 모든 인스턴스에 적용됨(보안 그룹 규칙이 지나치게 허용적일 경우 추가 보안 계층 제공)
허용 규칙만 지원	허용 및 거부 규칙 지원
트래픽 허용 여부를 결정하기 전에 모든 규칙을 평가	트래픽 허용 여부를 결정할 때 가장 낮은 번호의 규칙부터 순서대로 규칙을 평가합니다.
상태 저장: 규칙에 관계없이 반환 트래픽이 허용됨	상태 비저장: 반환 트래픽이 규칙에 따라 명시적으로 허용되어야 함

다음 다이어그램은 보안 그룹과 네트워크 ACL에서 제공하는 보안 계층을 보여 줍니다. 예를 들어, 인터넷 게이트웨이의 트래픽은 라우팅 테이블의 라우팅을 사용하여 적절한 서브넷에 라우팅됩니다. 서브넷과 연결된 네트워크 ACL 규칙은 서브넷에 허용되는 트래픽 유형을 제어합니다. 인스턴스와 연결된 보안 그룹 규칙은 인스턴스에 허용되는 트래픽 유형을 제어합니다.



보안 그룹만 사용하여 인스턴스를 보호할 수 있습니다. 그러나 네트워크 ACL을 추가 방어 계층으로 추가할 수 있습니다. 자세한 내용은 [예: 서브넷의 인스턴스에 대한 액세스 제어](#) 단원을 참조하십시오.

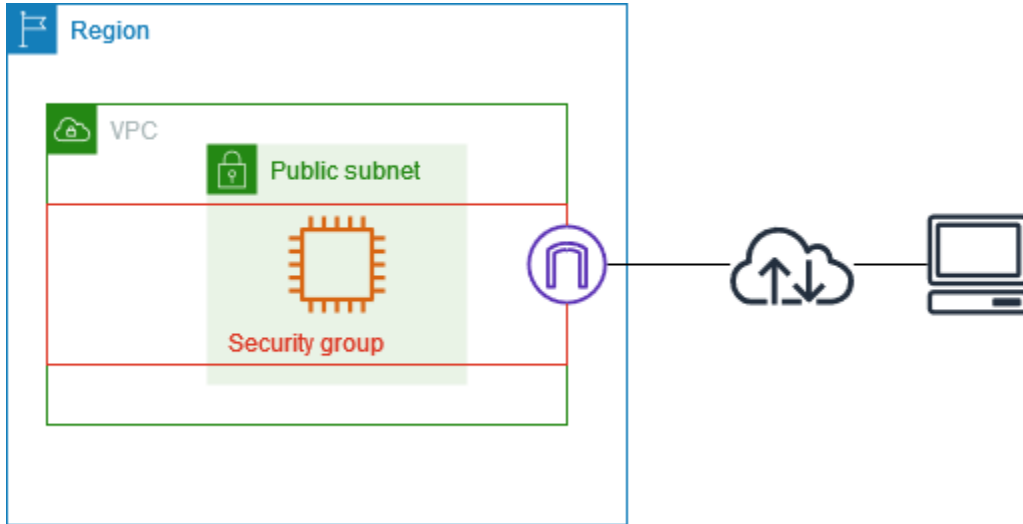
보안 그룹을 사용하여 AWS 리소스에 대한 트래픽 제어

보안 그룹은 연결된 리소스에 도달하고 나갈 수 있는 트래픽을 제어합니다. 예를 들어 보안 그룹을 EC2 인스턴스와 연결하면 인스턴스에 대한 인바운드 및 아웃바운드 트래픽을 제어합니다.

VPC를 생성할 경우 VPC는 기본 보안 그룹과 함께 제공됩니다. 각각 고유한 인바운드 및 아웃바운드 규칙이 있는 추가 보안 그룹을 생성할 수 있습니다. 각 인바운드 규칙에 대해 소스, 포트 범위 및 프로토

콜을 지정할 수 있습니다. 각 아웃바운드 규칙에 대해 대상, 포트 범위 및 프로토콜을 지정할 수 있습니다.

다음 다이어그램은 서브넷, 인터넷 게이트웨이 및 보안 그룹이 있는 VPC를 보여줍니다. 이 서브넷에는 EC2 인스턴스가 포함되어 있습니다. 보안 그룹은 인스턴스에 할당됩니다. 보안 그룹은 가상 방화벽의 기능을 수행합니다. 인스턴스에 연결되는 트래픽은 보안 그룹 규칙에서 허용되는 트래픽이 유일합니다. 예를 들어, 보안 그룹에 해당 네트워크에서 인스턴스로 이동하는 ICMP 트래픽을 허용하는 규칙이 포함되어 있는 경우 컴퓨터에서 인스턴스를 ping할 수 있습니다. 보안 그룹에 SSH 트래픽을 허용하는 규칙이 포함되어 있지 않은 경우에는 SSH를 사용하여 인스턴스에 연결할 수 없습니다.



내용

- [보안 그룹 기본 사항](#)
- [보안 그룹 예시](#)
- [보안 그룹 규칙](#)
- [VPC에 대한 기본 보안 그룹](#)
- [보안 그룹 작업](#)

요금

보안 그룹을 사용해도 추가 요금이 부과되지 않습니다.

보안 그룹 기본 사항

- 보안 그룹은 보안 그룹과 동일한 VPC에 만들어진 리소스에만 할당할 수 있습니다. 리소스에 여러 개의 보안 그룹을 할당할 수 있습니다.

- 보안 그룹을 생성할 때 이름과 설명을 제공해야 합니다. 다음 규칙이 적용됩니다.
 - 보안 그룹 이름은 VPC 내에서 고유해야 합니다.
 - 이름과 설명은 최대 255자일 수 있습니다.
 - 이름과 설명은 다음과 같은 문자로 제한됩니다. a-z, A-Z, 0-9, 공백 및 . _ : / () # , @ [] + = & ; { } ! \$ *
 - 이름에 후행 공백이 포함되어 있으면 이름 끝의 공백을 자릅니다. 예를 들어 이름에 “테스트 보안 그룹”을 입력하면 “테스트 보안 그룹”으로 저장됩니다.
 - 보안 그룹 이름은 sg-로 시작할 수 없습니다.
- 보안 그룹은 상태가 저장됩니다. 예를 들어 사용자가 인스턴스에서 요청을 전송하면 해당 요청의 응답 트래픽은 인바운드 보안 그룹 규칙에 관계없이 인스턴스에 도달할 수 있습니다. 허용된 인바운드 트래픽에 대한 응답은 아웃바운드 규칙에 관계없이 인스턴스를 떠날 수 있습니다.
- 보안 그룹은 다음에서 송수신되는 트래픽을 필터링하지 않습니다.
 - Amazon Domain Name Services(DNS)
 - Amazon Dynamic Host Configuration Protocol(DHCP)
 - Amazon EC2 인스턴스 메타데이터
 - Amazon ECS 태스크 메타데이터 엔드포인트
 - Windows 인스턴스에 대한 라이선스 활성화
 - Amazon Time Sync Service
 - 기본 VPC 라우터에서 사용하는 예약된 IP 주소
- VPC당 생성할 수 있는 보안 그룹의 개수, 각 보안 그룹에 추가할 수 있는 규칙의 개수, 그리고 네트워크 인터페이스에 연결할 수 있는 보안 그룹의 개수에는 할당량이 있습니다. 자세한 내용은 [Amazon VPC 할당량](#) 단원을 참조하십시오.

모범 사례

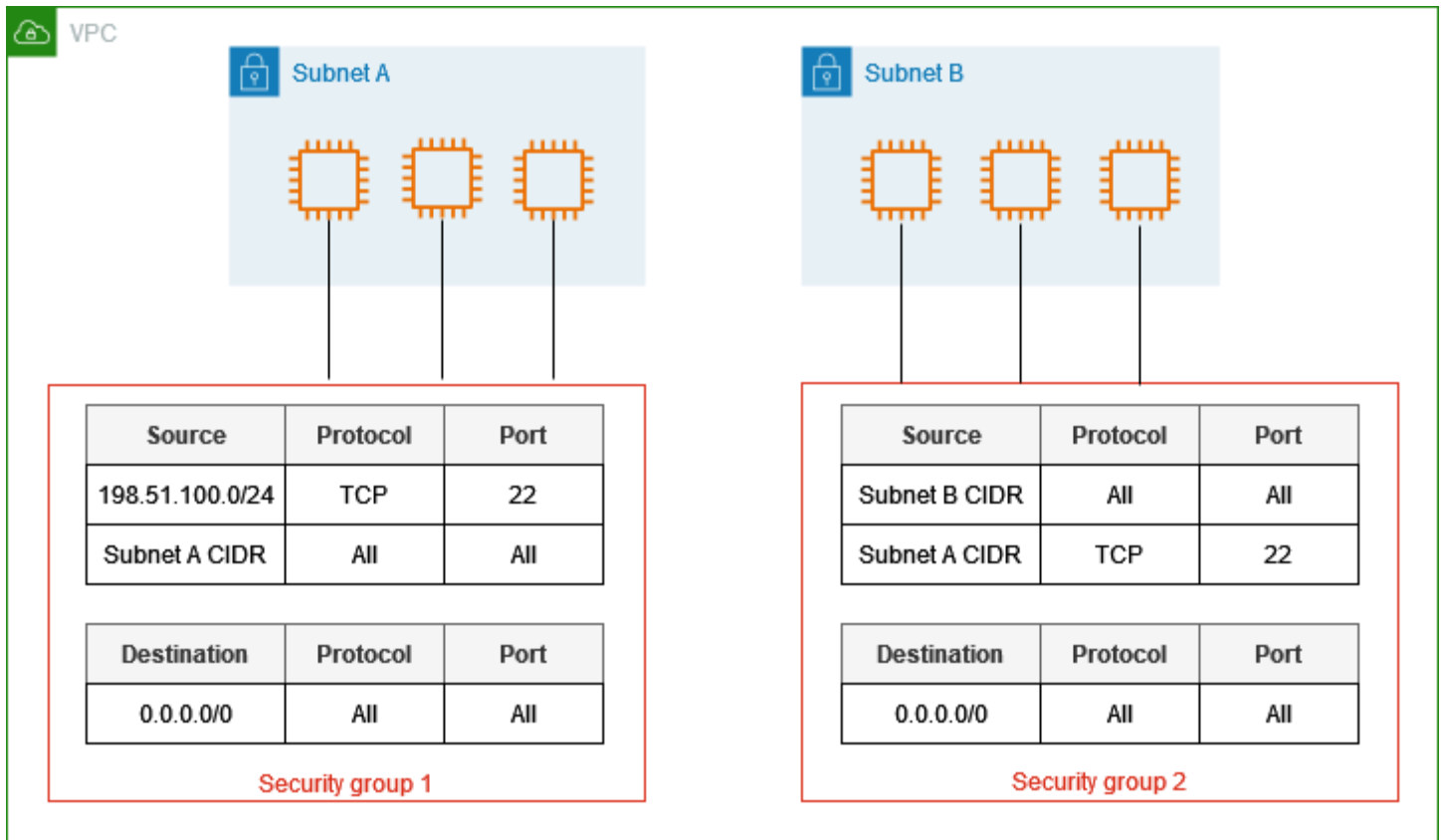
- 특정 IAM 보안 주체에만 보안 그룹을 생성 및 수정하는 권한을 부여합니다.
- 오류 위험을 줄이려면 필요한 최소 보안 그룹 수를 생성합니다. 각 보안 그룹을 사용하여 함수 및 보안 요구 사항이 비슷한 리소스에 대한 액세스 권한을 관리합니다.
- EC2 인스턴스에 액세스할 수 있도록 포트 22(SSH) 또는 3389(RDP)에 대한 인바운드 규칙을 추가하는 경우 특정 IP 주소 범위만 권한을 부여합니다. 0.0.0.0/0(IPv4)과 :::(IPv6)를 지정하면 누구든지 지정된 프로토콜을 사용하여 모든 IP 주소에서 인스턴스에 액세스할 수 있게 됩니다.
- 큰 포트 범위를 열지 마세요. 각 포트를 통한 액세스 권한이 해당 포트를 필요로 하는 원본 또는 대상으로 제한되도록 하세요.

- 보안 그룹과 비슷한 규칙으로 네트워크 ACL을 생성하여 VPC에 추가 보안 계층을 추가해 보세요. 보안 그룹과 네트워크 ACL의 차이에 대한 자세한 정보는 [보안 그룹 및 네트워크 ACL 비교](#) 단원을 참조하세요.

보안 그룹 예시

다음 다이어그램에서는 보안 그룹과 서브넷이 각각 2개인 VPC를 보여줍니다. 서브넷 A의 인스턴스는 연결 요구 사항이 동일한 보안 그룹 1과 연결됩니다. 서브넷 B의 인스턴스는 연결 요구 사항이 동일한 보안 그룹 2와 연결됩니다. 보안 그룹 규칙에서는 다음의 트래픽이 허용됩니다.

- 보안 그룹 1의 첫 번째 인바운드 규칙은 지정된 주소 범위(예: 자체 네트워크의 범위)에서 서브넷 A의 인스턴스로 이동하는 SSH 트래픽을 허용합니다.
- 보안 그룹 1의 두 번째 인바운드 규칙에서는 서브넷 A의 인스턴스가 프로토콜과 포트를 사용하여 상호간에 통신하는 것이 허용됩니다.
- 보안 그룹 2의 첫 번째 인바운드 규칙에서는 서브넷 B의 인스턴스가 프로토콜과 포트를 사용하여 상호간에 통신하는 것이 허용됩니다.
- 보안 그룹 2의 두 번째 인바운드 규칙에서는 서브넷 A의 인스턴스가 SSH를 사용하여 서브넷 B의 인스턴스와 통신하는 것이 허용됩니다.
- 두 보안 그룹 모두 모든 트래픽을 허용하는 기본 아웃바운드 규칙을 사용합니다.



보안 그룹 규칙

보안 그룹의 규칙은 보안 그룹과 연결된 리소스에 도달하도록 허용된 인바운드 트래픽을 제어합니다. 인스턴스에서 나갈 수 있는 아웃바운드 트래픽을 제어합니다.

보안 그룹의 규칙을 추가하거나 제거할 수 있습니다(인바운드 또는 아웃바운드 액세스 권한 부여 또는 취소라고도 함). 규칙은 인바운드 트래픽(수신)이나 아웃바운드 트래픽(송신)에 적용됩니다. 특정 소스 또는 대상에 대한 액세스 권한을 부여할 수 있습니다.

내용

- [보안 그룹 규칙 기본 사항](#)
- [보안 그룹 규칙의 구성 요소](#)
- [보안 그룹 참조](#)
- [보안 그룹 크기](#)
- [무효 보안 그룹 규칙](#)
- [보안 그룹 규칙으로 작업](#)
- [규칙 예](#)

• [연결 문제 해결](#)

보안 그룹 규칙 기본 사항

- 허용 규칙을 지정할 수 있지만 거부 규칙은 지정할 수 없습니다.
- 보안 그룹을 처음 만들 때 인바운드 규칙이 없습니다. 따라서 보안 그룹에 인바운드 규칙을 추가하기 전에는 어떤 인바운드 트래픽도 허용되지 않습니다.
- 보안 그룹을 처음 생성하면 리소스의 모든 아웃바운드 트래픽을 허용하는 아웃바운드 규칙이 있습니다. 규칙을 제거할 수 있으며 특정 아웃바운드 트래픽만 허용하는 아웃바운드 규칙을 추가할 수 있습니다. 보안 그룹에 아웃바운드 규칙이 없는 경우 어떤 아웃바운드 트래픽도 허용되지 않습니다.
- 여러 보안 그룹을 리소스와 연결하면 각 보안 그룹의 규칙이 집계되어 액세스 허용 여부를 결정하는데 사용되는 단일 규칙 집합을 형성합니다.
- 규칙을 추가, 업데이트 또는 제거할 때 변경 사항은 보안 그룹과 연결된 모든 리소스에 자동으로 적용됩니다. 일부 규칙 변경 사항이 미치는 효과는 트래픽의 추적 방법에 따라 다를 수 있습니다. 자세한 내용은 Amazon EC2 사용 설명서의 [연결 추적](#)을 참조하세요.
- 보안 그룹 규칙을 생성하면 AWS에서는 규칙에 고유한 ID가 할당됩니다. API 또는 CLI를 사용하여 규칙을 수정하거나 삭제할 때 규칙의 ID를 사용할 수 있습니다.

제한 사항

보안 그룹은 'VPC+2 IP 주소'라고도 하는 Route 53 Resolver와 주고받는 DNS 요청을 차단할 수 없습니다(Amazon Route 53 개발자 안내서의 [Amazon Route 53 Resolver](#) 또는 [AmazonProvidedDNS](#) 참조). Route 53 Resolver를 통해 DNS 요청을 필터링하려면 [Route 53 Resolver DNS 방화벽](#)을 사용하세요.

보안 그룹 규칙의 구성 요소

- 프로토콜: 허용할 프로토콜. 가장 일반적인 프로토콜은 6(TCP), 17(UDP) 및 1(ICMP)입니다.
- 포트 범위: TCP, UDP 또는 사용자 지정 프로토콜의 경우 허용할 포트의 범위. 단일 포트 번호(예: 22) 또는 포트 번호의 범위(예: 7000-8000)를 지정할 수 있습니다.
- ICMP 유형 및 코드: ICMP의 경우, ICMP 유형과 코드. 예를 들어 ICMP 에코 요청에 대해 유형 8을 사용하고 ICMPv6 에코 요청에 대해 유형 128을 입력합니다.
- 소스 또는 대상: 허용할 트래픽에 대한 소스(인바운드 규칙) 또는 대상(아웃바운드 규칙)입니다. 다음 중 하나를 지정하세요.
 - 단일 IPv4 주소. /32 접두사 길이를 사용해야 합니다. 예: 203.0.113.1/32.

- 단일 IPv6 주소. /128 접두사 길이를 사용해야 합니다. 예: 2001:db8:1234:1a00::123/128.
- CIDR 블록 표기법으로 표시된 IPv4 주소의 범위. 예를 들면 203.0.113.0/24입니다.
- CIDR 블록 표기법으로 표시된 IPv6 주소의 범위. 예를 들면 2001:db8:1234:1a00::/64입니다.
- 접두사 목록의 ID. 예를 들면 p1-1234abc1234abc123입니다. 자세한 내용은 [the section called “관리형 접두사 목록”](#) 단원을 참조하십시오.
- 보안 그룹의 ID. 예를 들면 sg-1234567890abcdef0입니다. 자세한 내용은 [the section called “보안 그룹 참조”](#) 단원을 참조하십시오.
- (선택 사항) 설명: 나중에 쉽게 식별할 수 있도록 규칙에 대한 설명을 입력할 수 있습니다. 설명 길이는 최대 255자입니다. 허용되는 문자는 a-z, A-Z, 0-9, 공백 및 .-:/()#,@[]+=;{}!\$*입니다.

보안 그룹 참조

보안 그룹을 규칙의 소스 또는 대상으로 지정할 경우 규칙은 보안 그룹과 연결된 모든 인스턴스에 영향을 줍니다. 인스턴스는 인스턴스의 프라이빗 IP 주소를 사용하여 지정된 프로토콜 및 포트를 통해 지정된 방향으로 통신할 수 있습니다.

예를 들어 다음은 보안 그룹 sg-0abcdef1234567890을(를) 참조하는 보안 그룹에 대한 인바운드 규칙을 나타냅니다. 이 규칙에서는 sg-0abcdef1234567890과(와) 연결된 인스턴스에서 발생한 인바운드 SSH 트래픽이 허용됩니다.

소스	프로토콜	포트 범위
<i>sg-0abcdef1234567890</i>	TCP	22

보안 그룹 규칙에서 보안 그룹을 참조할 때 다음 사항에 유의하세요.

- 두 보안 그룹 모두 동일한 VPC 또는 피어링된 VPC에 속해야 합니다.
- 참조된 보안 그룹의 규칙은 해당 그룹을 참조하는 보안 그룹에 추가되지 않습니다.
- 인바운드 규칙의 경우 보안 그룹과 연결된 EC2 인스턴스는 참조된 보안 그룹과 연결된 EC2 인스턴스의 프라이빗 IP 주소로부터 인바운드 트래픽을 수신할 수 있습니다.
- 아웃바운드 규칙의 경우 보안 그룹과 연결된 EC2 인스턴스는 참조된 보안 그룹과 연결된 EC2 인스턴스의 프라이빗 IP 주소로 아웃바운드 트래픽을 보낼 수 있습니다.

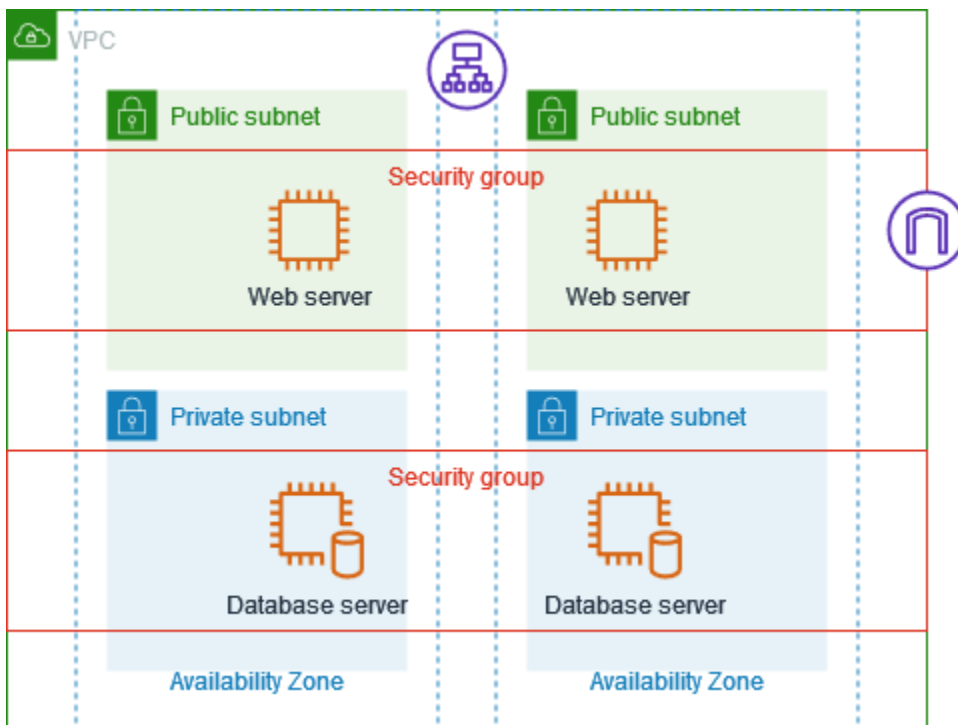
제한 사항

미들박스 어플라이언스를 통해 서로 다른 서브넷에 있는 두 인스턴스 간의 트래픽을 전달하도록 경로를 구성하는 경우 두 인스턴스에 대한 보안 그룹이 인스턴스 간에 트래픽이 흐르도록 허용해야 합니다. 각 인스턴스의 보안 그룹은 다른 인스턴스의 프라이빗 IP 주소 또는 다른 인스턴스가 포함된 서브넷의 CIDR 범위를 소스로 참조해야 합니다. 다른 인스턴스의 보안 그룹을 소스로 참조하면 인스턴스 간에 트래픽이 흐를 수 없습니다.

예

다음 다이어그램은 두 개의 가용 영역에 있는 서브넷, 인터넷 게이트웨이 및 Application Load Balancer가 있는 VPN을 보여 줍니다. 각 가용 영역에는 웹 서버용 퍼블릭 서브넷과 데이터베이스 서버용 프라이빗 서브넷이 있습니다. 로드 밸런서, 웹 서버 및 데이터베이스 서버에는 별도의 보안 그룹이 있습니다. 다음 보안 그룹 규칙을 생성하여 트래픽을 허용하세요.

- 인터넷의 HTTP 및 HTTPS 트래픽을 허용하도록 로드 밸런서 보안 그룹에 규칙을 추가합니다. 소스는 0.0.0.0/0입니다.
- 로드 밸런서의 HTTP 및 HTTPS 트래픽만 허용하도록 웹 서버의 보안 그룹에 규칙을 추가합니다. 소스는 로드 밸런서에 대한 보안 그룹입니다.
- 웹 서버의 데이터베이스 요청을 허용하도록 데이터베이스 서버의 보안 그룹에 규칙을 추가합니다. 소스는 웹 서버에 대한 보안 그룹입니다.



보안 그룹 크기

소스 또는 대상의 유형에 따라 각 규칙이 보안 그룹당 가질 수 있는 최대 규칙 수에 포함되는 방식이 결정됩니다.

- CIDR 블록을 참조하는 규칙은 하나의 규칙으로 계산됩니다.
- 다른 보안 그룹을 참조하는 규칙은 참조된 보안 그룹의 크기와 상관없이 하나의 규칙으로 계산됩니다.
- 고객이 관리하는 접두사 목록을 참조하는 규칙은 접두사 목록의 최대 크기로 계산됩니다. 예를 들어 접두사 목록의 최대 크기가 20인 경우 이 접두사 목록을 참조하는 규칙은 20개의 규칙으로 계산됩니다.
- AWS 관리형 접두사 목록을 참조하는 규칙은 접두사 목록의 가중치로 계산됩니다. 예를 들어 접두사 목록의 가중치가 10인 경우 이 접두사 목록을 참조하는 규칙은 10개의 규칙으로 계산됩니다. 자세한 내용은 [the section called “사용 가능한 AWS 관리형 접두사 목록”](#) 단원을 참조하십시오.

무효 보안 그룹 규칙

VPC에 다른 VPC와의 VPC 피어링 연결이 있는 경우 또는 다른 계정에 의해 공유된 VPC를 사용하는 경우 VPC의 보안 그룹 규칙은 해당 피어 VPC 또는 공유 VPC의 보안 그룹을 참조할 수 있습니다. 이를 통해 참조된 보안 그룹과 연결된 리소스가 참조하는 보안 그룹과 연결된 리소스와 서로 통신할 수 있습니다.

공유 VPC의 보안 그룹을 삭제하거나 VPC 피어링 연결을 삭제하는 경우 보안 그룹 규칙은 무효로 표시됩니다. 다른 보안 그룹 규칙과 같은 방법으로 무효 보안 그룹 규칙을 삭제할 수 있습니다. 자세한 내용은 Amazon VPC 피어링 안내서의 [무효 보안 그룹 규칙 작업](#)을 참조하세요.

보안 그룹 규칙으로 작업

다음 작업은 보안 그룹 규칙 작업 방법을 보여 줍니다.

필요한 권한

- [보안 그룹 규칙 관리](#)

Tasks

- [보안 그룹에 규칙 추가](#)
- [보안 그룹 규칙 업데이트](#)

- [보안 그룹 규칙 태깅](#)
- [보안 그룹 규칙 삭제](#)

보안 그룹에 규칙 추가

보안 그룹에 규칙을 추가할 경우 보안 그룹과 연결된 리소스에 새 규칙이 자동으로 적용됩니다.

VPC 피어링 연결이 있는 경우, 피어 VPC의 보안 그룹을 보안 그룹 규칙의 소스 또는 대상으로 참조할 수 있습니다. 자세한 내용은 Amazon VPC 피어링 가이드의 [피어링된 VPC 보안 그룹을 참조하도록 보안 그룹 업데이트](#)를 참조하십시오.

보안 그룹 규칙 관리에 필요한 권한에 대한 자세한 정보는 [보안 그룹 규칙 관리](#) 섹션을 참조하세요.

Warning

Anywhere-IPv4를 선택하면 모든 IPv4 주소의 트래픽이 허용됩니다. Anywhere-IPv6을 선택하면 모든 IPv6 주소의 트래픽이 허용됩니다. 포트 22(SSH) 또는 3389(RDP)에 대한 규칙을 추가할 때 특정 IP 주소 범위만 인스턴스에 액세스할 수 있는 권한을 부여합니다.

콘솔을 사용하여 규칙을 추가하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹을 선택합니다.
4. 작업(Actions), 인바운드 규칙 편집(Edit inbound rules) 또는 작업(Actions), 아웃바운드 규칙 편집(Edit outbound rules)을 선택합니다.
5. 각 규칙에 대해 규칙 추가(Add rule)를 선택하고 다음을 수행합니다.
 - a. 유형(Type)에서 허용할 프로토콜 유형을 선택합니다.
 - TCP 또는 UDP의 경우 허용할 포트 범위를 입력해야 합니다.
 - 사용자 지정 ICMP의 경우 프로토콜(Protocol)에서 ICMP 유형 이름을 선택하고, 해당되는 경우 포트 범위(Port range)에서 코드 이름을 선택해야 합니다.
 - 다른 유형에 대해 프로토콜과 포트 범위가 자동으로 구성됩니다.
 - b. 소스 유형(Source type)(인바운드 규칙) 또는 대상 유형(Destination type)(아웃바운드 규칙)에서 다음 중 하나를 수행하여 트래픽을 허용합니다.

- [사용자 지정(Custom)]을 선택한 다음 CIDR 표기법의 IP 주소, CIDR 블록, 다른 보안 그룹 또는 접두사 목록을 입력합니다.
 - 임의 IPv4 주소의 트래픽을 허용하거나(인바운드 규칙) 트래픽이 모든 IPv4 주소(아웃바운드 규칙)에 도달하도록 허용하려면 Anywhere-IPv4를 선택합니다. 이 작업을 수행하면 0.0.0.0/0 IPv4 CIDR 블록에 대해 규칙이 자동으로 추가됩니다.
 - 임의 IPv6 주소의 트래픽을 허용하거나(인바운드 규칙) 트래픽이 모든 IPv6 주소(아웃바운드 규칙)에 도달하도록 허용하려면 Anywhere-IPv6를 선택합니다. 이 작업을 수행하면 ::/0 IPv6 CIDR 블록에 대해 규칙이 자동으로 추가됩니다.
 - [내 IP(My IP)]를 선택하여 로컬 컴퓨터의 퍼블릭 IPv4 주소에서 전송된 트래픽만 허용하거나(인바운드 규칙) 로컬 컴퓨터의 퍼블릭 IPv4 주소로 트래픽을 허용하세요.
- c. (선택 사항) [설명(Description)]에서 규칙에 대한 간단한 설명을 지정합니다.
6. 규칙 저장을 선택합니다.

AWS CLI를 사용하여 보안 그룹에 규칙을 추가하려면

[authorize-security-group-ingress](#) 및 [authorize-security-group-egress](#) 명령을 사용합니다.

보안 그룹 규칙 업데이트

규칙을 업데이트하면 업데이트된 규칙은 보안 그룹과 연결된 모든 리소스에 자동으로 적용됩니다.

보안 그룹 규칙 관리에 필요한 권한에 대한 자세한 정보는 [보안 그룹 규칙 관리](#) 섹션을 참조하세요.

콘솔을 사용하여 규칙을 업데이트하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹을 선택합니다.
4. 작업, 인바운드 규칙 편집 또는 작업, 아웃바운드 규칙 편집을 선택합니다.
5. 필요에 따라 규칙을 업데이트합니다.
6. 규칙 저장을 선택합니다.

AWS CLI를 사용하여 보안 그룹 규칙을 업데이트하려면

[modify-security-group-rules](#), [update-security-group-rule-descriptions-ingress](#) 및 [update-security-group-rule-descriptions-egress](#) 명령을 사용합니다.

보안 그룹 규칙 태깅

리소스에 태그를 추가하면 용도, 소유자 또는 환경과 같은 기준으로 태그를 구성하고 식별할 수 있습니다. 보안 그룹 규칙에 태그를 추가할 수 있습니다. 태그 키는 각 보안 그룹 규칙별로 고유해야 합니다. 보안 그룹 규칙에 이미 연결된 키로 태그를 추가하면 해당 태그의 값이 업데이트됩니다.

콘솔을 사용하여 규칙을 태깅하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹을 선택합니다.
4. 인바운드 규칙 또는 아웃바운드 규칙 탭에서 해당 규칙의 확인란을 선택한 다음 태그 관리를 선택합니다.
5. [태그 관리(Manage tags)] 페이지에는 해당 규칙에 할당된 모든 태그가 표시됩니다. 태그를 추가하려면 태그 추가(Add tag)를 선택한 다음 태그 키와 값을 입력합니다. 태그를 삭제하려면 삭제할 태그 옆에 있는 제거(Remove)를 선택합니다.
6. Save changes(변경 사항 저장)를 선택합니다.

AWS CLI를 사용하여 규칙을 태깅하려면

[create-tags](#) 명령을 사용합니다.

보안 그룹 규칙 삭제

보안 그룹에서 규칙을 삭제할 경우 보안 그룹과 연결된 인스턴스에 해당 변경 내용이 자동으로 적용됩니다.

콘솔을 사용하여 보안 그룹 규칙을 삭제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹을 선택합니다.
4. [작업(Actions)]을 선택한 다음, 인바운드 규칙을 제거하려면 [인바운드 규칙 편집(Edit inbound rules)]을, 아웃바운드 규칙을 제거하려면 [아웃바운드 규칙 편집(Edit outbound rules)]을 선택합니다.
5. 삭제할 규칙 옆에 있는 [삭제>Delete] 버튼을 선택합니다.

6. 규칙 저장을 선택합니다. 또는 변경 사항 미리 보기를 선택하여 변경 내용을 검토한 다음 확인을 선택합니다.

AWS CLI를 사용하여 보안 그룹 규칙을 삭제하려면

[revoke-security-group-ingress](#) 및 [revoke-security-group-egress](#) 명령을 사용합니다.

규칙 예

웹 서버

다음은 웹 서버의 보안 그룹에 대한 규칙의 예시입니다. 웹 서버는 IPv4 및 IPv6 주소로부터 HTTP 및 HTTPS 트래픽을 수신하고, SQL 또는 MySQL 트래픽을 데이터베이스 서버에 전송할 수 있습니다.

Warning

EC2 인스턴스에 액세스할 수 있도록 포트 22(SSH) 또는 3389(RDP)에 대한 규칙을 추가하는 경우 특정 IP 주소 범위만 권한을 부여하는 것이 좋습니다. 0.0.0.0/0(IPv4)과 ::/0(IPv6)를 지정하면 누구든지 지정된 프로토콜을 사용하여 모든 IP 주소에서 인스턴스에 액세스할 수 있게 됩니다.

인바운드

소스	프로토콜	포트 범위	설명
0.0.0.0/0	TCP	80	모든 IPv4 주소에서 이루어지는 인바운드 HTTP 액세스 허용
::/0	TCP	80	모든 IPv6 주소에서 이루어지는 인바운드 HTTP 액세스 허용
0.0.0.0/0	TCP	443	모든 IPv4 주소에서 이루어지는 인바운드 HTTPS 액세스 허용

소스	프로토콜	포트 범위	설명
::/0	TCP	443	모든 IPv6 주소에서 이루어지는 인바운드 HTTPS 액세스 허용
##### ### IPv4 ## ##	TCP	22	(선택 사항) 네트워크에서 IPv4 IP 주소의 인바운드 SSH 액세스 허용
##### IPv6 ## ##	TCP	22	(선택 사항) 네트워크에서 IPv6 IP 주소의 인바운드 SSH 액세스 허용
##### ### IPv4 ## ##	TCP	3389	(선택 사항) 네트워크에서 IPv4 IP 주소의 인바운드 RDP 액세스 허용
##### IPv6 ## ##	TCP	3389	(선택 사항) 네트워크에서 IPv6 IP 주소의 인바운드 RDP 액세스 허용
# ## ## ID	모두	모두	(선택 사항) 이 보안 그룹과 연결된 다른 서버의 인바운드 트래픽 허용

아웃바운드

대상 주소	프로토콜	포트 범위	설명
Microsoft SQL Server ##### ## ## ID	TCP	1433	아웃바운드 Microsoft SQL Server 액세스 허용
MySQL# ##### ## ## ID	TCP	3306	아웃바운드 MySQL 액세스 허용

데이터베이스 서버

데이터베이스 서버에는 MySQL 또는 Microsoft SQL Server와 같은 인바운드 특정 프로토콜을 허용하는 규칙이 필요합니다. 예를 들어, Amazon EC2 사용 설명서의 [데이터베이스 서버 규칙](#)을 참조하세요. Amazon RDS DB 인스턴스의 보안 그룹에 대한 자세한 내용은 Amazon RDS 사용 설명서의 [보안 그룹을 통한 액세스 제어](#)를 참조하세요.

연결 문제 해결

Reachability Analyzer는 정적 구성 분석 도구입니다. Reachability Analyzer를 사용하여 VPC의 두 리소스 간 네트워크 연결성을 분석하고 디버깅할 수 있습니다. Reachability Analyzer에서는 연결할 수 있는 경우 이러한 리소스 간 가상 경로에 대한 출발 세부 정보가 생성되고, 그렇지 않다면 차단 구성 요소가 식별됩니다. 예를 들면 누락되거나 잘못 구성된 보안 그룹 규칙이 식별될 수 있습니다.

자세한 내용은 [Reachability Analyzer 사용 설명서](#)를 참조하십시오.

VPC에 대한 기본 보안 그룹

기본 VPC와 사용자가 생성한 VPC는 기본 보안 그룹과 함께 제공됩니다. 기본 보안 그룹의 이름은 "default"입니다.

기본 보안 그룹을 사용하는 대신 특정 리소스 또는 리소스 그룹에 대한 보안 그룹을 만드는 것이 좋습니다. 단, 일부 리소스는 생성 시점에 보안 그룹을 연결하지 않으면 기본 보안 그룹이 연결됩니다. 예를 들어 EC2 인스턴스를 시작할 때 보안 그룹을 지정하지 않은 경우 인스턴스는 그 VPC의 기본 보안 그룹과 연결됩니다.

기본 보안 그룹 기본 사항

- 기본 보안 그룹에 대한 규칙을 변경할 수 있습니다.
- 기본 보안 그룹을 삭제할 수 없습니다. 기본 보안 그룹을 삭제하려고 하면 Client.CannotDelete라는 오류 코드가 반환됩니다.

기본 규칙

다음 표에서는 기본 보안 그룹의 기본 규칙을 설명합니다.

인바운드

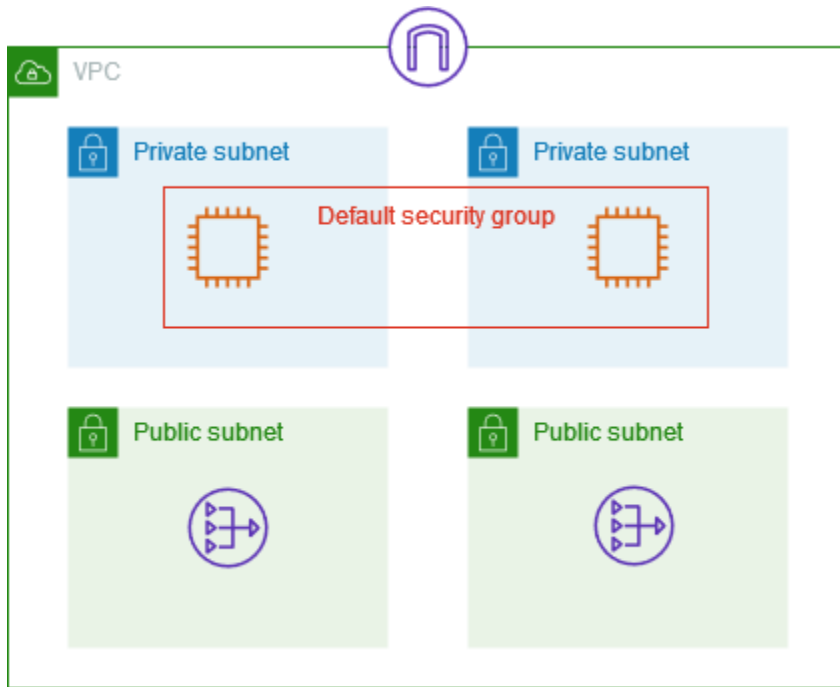
소스	프로토콜	포트 범위	설명
<i>sg-1234567890abcdef0</i>	모두	모두	이 보안 그룹에 할당된 모든 리소스로부터의 인바운드 트래픽을 허용합니다. 소스는 보안 그룹의 ID입니다.

아웃바운드

대상 주소	프로토콜	포트 범위	설명
0.0.0.0/0	모두	모두	모든 아웃바운드 IPv4 트래픽을 허용합니다.
:::0	모두	모두	모든 아웃바운드 IPv6 트래픽을 허용합니다. 이 규칙은 VPC에 연결된 IPv6 CIDR 블록이 있는 경우에만 추가됩니다.

예

다음 다이어그램은 기본 보안 그룹, 인터넷 게이트웨이 및 NAT 게이트웨이가 있는 VPC를 보여 줍니다. 기본 보안에는 기본 규칙만 포함되며 VPC에서 실행되는 두 개의 EC2 인스턴스와 연결됩니다. 이 시나리오에서 각 인스턴스는 모든 포트 및 프로토콜에서 다른 인스턴스로부터 인바운드 트래픽을 수신할 수 있습니다. 기본 규칙에서는 인스턴스가 인터넷 게이트웨이 또는 NAT 게이트웨이로부터 트래픽을 수신하는 것이 허용되지 않습니다. 인스턴스에서 반드시 추가 트래픽을 수신해야 하는 경우 필수 규칙이 있는 보안 그룹을 만들고 새 보안 그룹을 기본 보안 그룹 대신 인스턴스에 연결하는 것을 권장합니다.



보안 그룹 작업

다음 작업은 보안 그룹 작업 방법을 보여 줍니다.

Tasks

- [보안 그룹 생성](#)
- [보안 그룹 보기](#)
- [보안 그룹 태깅](#)
- [보안 그룹 삭제](#)
- [Firewall Manager를 사용하여 보안 그룹 관리](#)

필요한 권한

시작하기 전에 사전 허가를 받았는지 확인하세요.

- [보안 그룹 관리](#)
- [보안 그룹 규칙 관리](#)

보안 그룹의 규칙은 보안 그룹과 연결된 리소스에 도달하도록 허용된 인바운드 트래픽을 제어합니다. 보안 그룹 규칙에 대한 자세한 내용은 [보안 그룹 규칙](#) 부분을 참조하세요.

보안 그룹 생성

기본적으로 처음에 새 보안 그룹에는 리소스에서 나가는 모든 트래픽을 허용하는 아웃바운드 규칙만 적용됩니다. 인바운드 트래픽을 사용하거나 아웃바운드 트래픽을 제한하려면 규칙을 추가해야 합니다.

콘솔을 사용하여 보안 그룹을 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹 생성을 선택합니다.
4. 보안 그룹의 이름과 설명을 입력합니다. 보안 그룹을 생성한 후에는 보안 그룹에 대한 이름과 설명을 변경할 수 없습니다.
5. VPC에서 VPC를 선택합니다. 보안 그룹은 보안 그룹이 생성된 VPC에서만 사용할 수 있습니다.
6. 보안 그룹 규칙을 지금 추가하거나 나중에 추가할 수 있습니다. 자세한 내용은 [보안 그룹에 규칙 추가](#) 단원을 참조하십시오.
7. 태그를 지금 추가하거나 나중에 추가할 수 있습니다. 태그를 추가하려면 새 태그 추가(Add new tag)를 선택한 다음 태그 키와 값을 입력합니다.
8. 보안 그룹 생성을 선택합니다.

보안 그룹을 생성한 후 다음 중 하나를 수행할 수 있습니다.

- 인스턴스를 시작할 때 EC2 인스턴스에 보안 그룹을 할당하거나 인스턴스에 현재 할당된 보안 그룹을 변경할 수 있습니다. 자세한 내용은 Amazon EC2 사용 설명서의 [인스턴스 실행](#) 또는 [보안 그룹 변경](#)을 참조하세요.
- 보안 그룹 규칙을 추가합니다. 보안 그룹의 규칙은 보안 그룹과 연결된 리소스에 도달하도록 허용된 인바운드 트래픽을 제어합니다. 보안 그룹 규칙에 대한 자세한 내용은 [보안 그룹 규칙으로 작업](#) 부분을 참조하세요.

AWS CLI를 사용하여 보안 그룹을 생성하려면

[create-security-group](#) 명령을 사용합니다.

보안 그룹 보기

다음과 같이 보안 그룹에 대한 정보를 볼 수 있습니다.

콘솔을 사용하여 보안 그룹을 보려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹이 나열됩니다. 해당 인바운드 및 아웃바운드 규칙을 포함하여 특정 보안 그룹에 대한 세부 정보를 보려면 보안 그룹을 선택합니다. 보안 그룹 규칙 업데이트에 대한 자세한 내용은 [보안 그룹 규칙 업데이트](#)를 참조하세요.

리전 간의 모든 보안 그룹을 보려면

<https://console.aws.amazon.com/ec2globalview/home>에서 Amazon EC2 Global View 콘솔을 엽니다. 자세한 내용은 Amazon EC2 사용 설명서의 [Amazon EC2 Global View를 사용하여 리소스 나열 및 필터링](#)을 참조하세요.

AWS CLI를 사용하여 보안 그룹을 보려면

[describe-security-groups](#) 및 [describe-security-group-rules](#) 명령을 사용합니다.

보안 그룹 태깅

리소스에 태그를 추가하면 용도, 소유자 또는 환경과 같은 기준으로 태그를 구성하고 식별할 수 있습니다. 보안 그룹에 태그를 추가할 수 있습니다. 태그 키는 각 보안 그룹별로 고유해야 합니다. 규칙에 이미 연결된 키로 태그를 추가하면 해당 태그의 값이 업데이트됩니다.

콘솔을 사용하여 보안 그룹을 태깅하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹의 확인란을 선택합니다.
4. 작업(Actions), 태그 관리(Manage tags)를 선택합니다. 태그 관리(Manage tags) 페이지에는 보안 그룹에 할당된 모든 태그가 표시됩니다.
5. 태그를 추가하려면 새 태그 추가를 선택한 다음 태그 키와 태그 값을 입력합니다. 태그를 삭제하려면 삭제할 태그 옆의 [제거(Remove)]를 선택합니다.
6. [Save changes]를 선택합니다.

AWS CLI를 사용하여 보안 그룹을 태깅하려면

[create-tags](#) 명령을 사용합니다.

보안 그룹 삭제

리소스와 연결되지 않은 경우에만 보안 그룹을 삭제할 수 있습니다. 기본 보안 그룹을 삭제할 수 없습니다.

콘솔을 사용하면 한 번에 두 개 이상의 보안 그룹을 삭제할 수 있습니다. 명령줄 또는 API를 사용하면 한 번에 한 개의 보안 그룹만 삭제할 수 있습니다.

콘솔을 사용하여 보안 그룹을 삭제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹을 선택한 다음 작업, 보안 그룹 삭제를 선택합니다.
4. 확인 메시지가 나타나면 삭제를 선택합니다.

AWS CLI를 사용하여 보안 그룹을 삭제하려면

[delete-security-group](#) 명령을 사용합니다.

Firewall Manager를 사용하여 보안 그룹 관리

AWS Firewall Manager는 여러 계정과 리소스에서 보안 그룹 관리 및 유지 관리 작업을 간소화합니다. Firewall Manager를 사용하면 단일 중앙 관리자 계정에서 조직의 보안 그룹을 구성하고 감사할 수 있습니다. Firewall Manager는 새로 추가한 리소스를 포함한 모든 계정과 리소스에 자동으로 규칙과 보호를 적용합니다. Firewall Manager는 조직 전체를 보호해야 하거나 중앙 관리자 계정으로 보호할 새 리소스를 자주 추가하는 경우에 특히 유용합니다.

Firewall Manager를 사용하면 다음과 같은 방법으로 보안 그룹을 중앙에서 관리할 수 있습니다.

- 조직 전체에서 공통 기본 보안 그룹 구성: 공통 보안 그룹 정책을 사용하면 조직 전체에서 계정과 리소스에 대한 보안 그룹의 연결을 중앙에서 제어할 수 있습니다. 조직에서 정책을 적용할 위치와 방법을 지정합니다.
- 조직의 기존 보안 그룹 감사: 보안 그룹 감사 정책을 사용하여 조직의 보안 그룹에서 사용 중인 기존 규칙을 확인할 수 있습니다. 모든 계정, 특정 계정 또는 태그가 지정된 조직 내 리소스를 감사하도록 정책의 범위를 지정할 수 있습니다. Firewall Manager가 자동으로 새 계정 및 리소스를 감지하고 감사합니다. 감사 규칙을 생성하여 조직 내에서 허용 또는 금지할 보안 그룹 규칙에 대한 가드레일을 설정하고 사용되지 않거나 중복된 보안 그룹을 확인할 수 있습니다.

- 규정 미준수 리소스에 대한 보고서 가져오기 및 문제 해결: 기준 및 감사 정책에 대한 규정 미준수 리소스에 대한 보고서와 알림을 받을 수 있습니다. 또한 자동 문제 해결 워크플로를 설정하여 Firewall Manager에서 감지한 모든 규정 미준수 리소스의 문제를 해결할 수도 있습니다.

Firewall Manager를 사용하여 보안 그룹을 관리하는 방법에 대한 자세한 내용은 AWS Firewall Manager 개발자 안내서의 다음 리소스를 참조하세요.

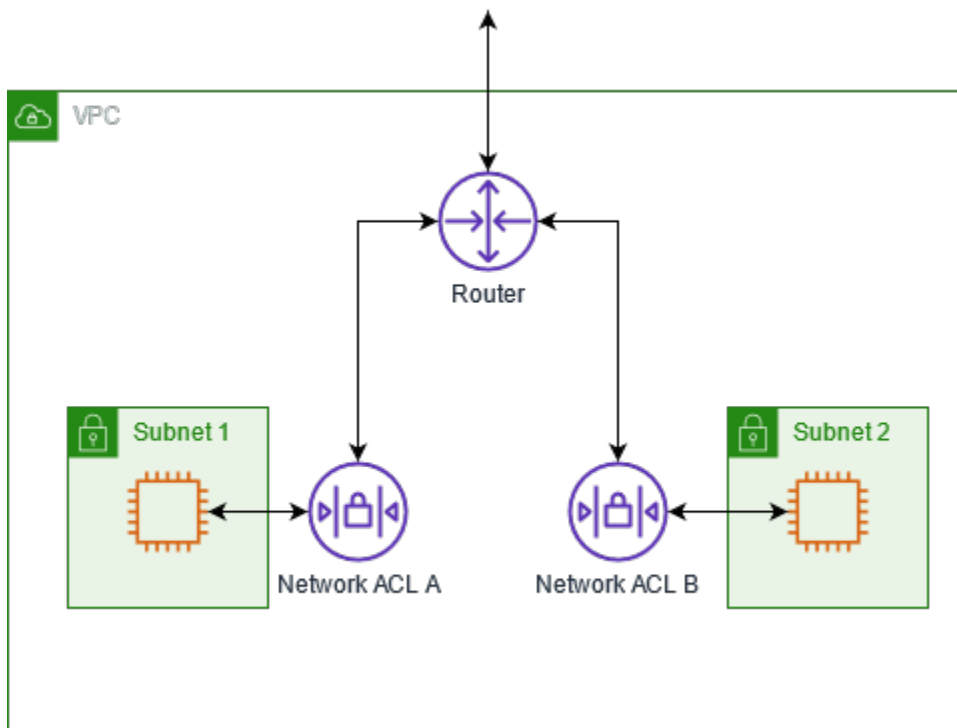
- [AWS Firewall Manager 필수 조건](#)
- [AWS Firewall Manager Amazon VPC 보안 그룹 정책 시작하기](#)
- [AWS Firewall Manager](#)에서 보안 그룹 정책의 작동 방식
- [보안 그룹 정책 사용 사례](#)

네트워크 ACL을 사용하여 서브넷에 대한 트래픽 제어

네트워크 액세스 제어 목록(ACL)은 서브넷 수준에서 특정 인바운드 또는 아웃바운드 트래픽을 허용하거나 거부합니다. VPC에 대한 기본 네트워크 ACL을 사용하거나 보안 그룹에 대한 규칙과 유사한 규칙을 사용하여 VPC에 대한 사용자 지정 네트워크 ACL을 생성하여 VPC에 보안 계층을 추가할 수 있습니다.

네트워크 ACL을 사용해도 추가 요금이 부과되지 않습니다.

다음 다이어그램에서는 서브넷이 2개인 VPC를 보여줍니다. 각 서브넷에 네트워크 ACL이 있습니다. 트래픽이 VPC로 들어오면(예: 피어링된 VPC, VPN 연결 또는 인터넷의 트래픽) 라우터에서 트래픽을 대상으로 보냅니다. 네트워크 ACL A에서는 서브넷 1로 향하는 트래픽 중 서브넷 1로 들어가도록 허용되는 트래픽과 서브넷 1 외부 위치로 향하는 트래픽 중 서브넷 1에서 나가도록 허용되는 트래픽을 결정합니다. 마찬가지로, 네트워크 ACL B에서는 서브넷 2에 들어오고 나갈 수 있는 트래픽을 결정합니다.



보안 그룹과 네트워크 ACL의 차이점에 대한 자세한 내용은 [보안 그룹 및 네트워크 ACL 비교](#)를 참조하세요.

내용

- [네트워크 ACL 기본 사항](#)
- [네트워크 ACL 규칙](#)
- [기본 네트워크 ACL](#)
- [사용자 지정 네트워크 ACL](#)
- [사용자 지정 네트워크 ACL 및 기타 AWS 서비스](#)
- [휘발성 포트](#)
- [경로 MTU 검색](#)
- [네트워크 ACL 작업](#)
- [예: 서브넷의 인스턴스에 대한 액세스 제어](#)
- [연결 문제 해결](#)

네트워크 ACL 기본 사항

다음은 네트워크 ACL에 대해 알아야 할 기본 사항입니다.

- VPC는 수정 가능한 기본 네트워크 ACL과 함께 자동으로 제공됩니다. 기본적으로 모든 인바운드 및 아웃바운드 IPv4 트래픽을 허용하며, 해당되는 경우 IPv6 트래픽도 허용합니다.
- 사용자 지정 네트워크 ACL을 생성하고 서브넷과 연결하여 서브넷 수준에서 특정 인바운드 또는 아웃바운드 트래픽을 허용하거나 거부할 수 있습니다.
- VPC에 있는 각 서브넷을 네트워크 ACL과 연결해야 합니다. 서브넷을 네트워크 ACL에 명시적으로 연결하지 않을 경우, 서브넷은 기본 네트워크 ACL에 자동적으로 연결됩니다.
- 네트워크 ACL을 여러 서브넷과 연결할 수 있습니다. 그러나 서브넷은 한 번에 하나의 네트워크 ACL에만 연결할 수 있습니다. 네트워크 ACL을 서브넷과 연결하면 이전 연결은 제거됩니다.
- 네트워크 ACL에는 인바운드 규칙과 아웃바운드 규칙이 있습니다. 각 규칙에서는 트래픽을 허용하거나 거부할 수 있습니다. 각 규칙에는 1부터 32766까지 번호가 있습니다. 규칙은 트래픽 허용 또는 거부가 결정될 때 가장 낮은 번호의 규칙부터 순서대로 평가됩니다. 트래픽이 규칙과 일치하면 규칙이 적용되며 추가 규칙은 평가되지 않습니다. 필요한 경우 나중에 새 규칙을 삽입할 수 있도록 증분 방식으로(예: 10 또는 100 단위씩 증분) 규칙을 생성하여 시작하는 것이 좋습니다.
- 네트워크 ACL 규칙은 트래픽이 서브넷 내에서 라우팅될 때가 아니라 서브넷에 들어오고 나갈 때 평가됩니다.
- NACL은 상태 비저장 목록이므로 이전에 전송했거나 수신한 트래픽에 대한 정보가 저장되지 않습니다. 예를 들어, 서브넷에 대한 특정 인바운드 트래픽을 허용하는 NACL 규칙을 생성하는 경우 해당 트래픽에 대한 응답이 자동으로 허용되지 않습니다. 이는 보안 그룹의 작동 방식과 대조적입니다. 보안 그룹은 상태 저장 그룹이므로 이전에 전송했거나 수신한 트래픽에 대한 정보가 저장됩니다. 예를 들어, 보안 그룹이 EC2 인스턴스에 대한 인바운드 트래픽을 허용하는 경우 아웃바운드 보안 그룹 규칙에 관계없이 응답이 자동으로 허용됩니다.
- 네트워크 ACL은 Route 53 Resolver(VPC+2 IP 주소 또는 AmazonProvidedDNS라고도 함)에서 송수신되는 DNS 요청을 차단할 수 없습니다. Route 53 Resolver를 통해 DNS 요청을 필터링하려면 [Route 53 Resolver DNS 방화벽](#)을 활성화합니다(Amazon Route 53 개발자 안내서 참조).
- 네트워크 ACL은 인스턴스 메타데이터 서비스(IMDS)에 대한 트래픽을 차단할 수 없습니다. IMDS에 대한 액세스를 관리하려면 Amazon EC2 사용 설명서의 [인스턴스 메타데이터 옵션 구성](#)을 참조하세요.
- 네트워크 ACL은 다음에서 송수신되는 트래픽을 필터링하지 않습니다.
 - Amazon Domain Name Services(DNS)
 - Amazon Dynamic Host Configuration Protocol(DHCP)
 - Amazon EC2 인스턴스 메타데이터
 - Amazon ECS 태스크 메타데이터 엔드포인트
 - Windows 인스턴스에 대한 라이선스 활성화

- Amazon Time Sync Service
- 기본 VPC 라우터에서 사용하는 예약된 IP 주소
- VPC당 네트워크 ACL 수 및 네트워크 ACL당 규칙 수에 대한 할당량(제한이라고도 함)이 있습니다. 자세한 내용은 [Amazon VPC 할당량](#) 단원을 참조하십시오.

네트워크 ACL 규칙

기본 네트워크 ACL에 규칙을 추가 또는 제거하거나, VPC에 대한 네트워크 ACL을 추가로 생성할 수 있습니다. 네트워크 ACL에서 규칙을 추가하거나 제거할 때 네트워크 ACL이 연결되어 있는 서브넷에 변경 사항이 자동으로 적용됩니다.

다음은 네트워크 ACL 규칙 중 일부입니다.

- 규칙 번호. 번호가 가장 낮은 규칙부터 평가됩니다. 규칙에 일치하는 트래픽이 있으면 이와 모순되는 상위 규칙이 있더라도 적용됩니다.
- 유형. 트래픽 유형(예: SSH)입니다. 모든 트래픽 또는 사용자 지정 범위를 지정할 수도 있습니다.
- 프로토콜. 표준 프로토콜 번호를 가진 어떤 프로토콜이든 지정할 수 있습니다. 자세한 내용은 [프로토콜 번호](#)를 참조하십시오. ICMP를 프로토콜로 지정하면 ICMP 유형과 코드 중 일부 또는 전부를 지정할 수 있습니다.
- 포트 범위. 트래픽에 대한 수신 포트 또는 포트 범위입니다. 예를 들어, HTTP 트래픽의 경우 80입니다.
- 소스: . [인바운드 규칙만 해당] 트래픽의 소스(CIDR 범위)입니다.
- 대상 [아웃바운드 규칙만 해당] 트래픽의 대상(CIDR 범위)입니다.
- 허용/거부. 지정된 트래픽을 허용 또는 거부 할지 여부입니다.

명령줄 도구 또는 Amazon EC2 API를 사용하여 규칙을 추가하면 CIDR 범위가 표준 형식으로 자동 수정됩니다. 예를 들어 CIDR 범위에 100.68.0.18/18을 지정하면 100.68.0.0/18 CIDR 범위를 가진 규칙이 작성됩니다.

기본 네트워크 ACL

기본 네트워크 ACL은 연결된 서브넷을 드나드는 트래픽 흐름을 모두 허용하도록 구성되어 있습니다. 각 네트워크 ACL에는 규칙 번호가 별표(*)로 되어 있는 규칙도 포함되어 있습니다. 이 규칙은 패킷이 번호가 매겨진 다른 어떤 규칙과도 일치하지 않을 경우에는 거부되도록 되어 있습니다. 이 규칙을 수정하거나 제거할 수 없습니다.

다음은 IPv4만을 지원하는 VPC에 대한 기본 네트워크 ACL의 예시입니다.

인바운드

규칙 #	Type	프로토콜	포트 범위	소스	허용/거부
100	모든 IPv4 트래픽	모두	모두	0.0.0.0/0	ALLOW
*	모든 IPv4 트래픽	모두	모두	0.0.0.0/0	DENY

아웃바운드

규칙 #	Type	프로토콜	포트 범위	대상 주소	허용/거부
100	모든 IPv4 트래픽	모두	모두	0.0.0.0/0	ALLOW
*	모든 IPv4 트래픽	모두	모두	0.0.0.0/0	DENY

IPv6 CIDR 블록이 있는 VPC를 생성하거나 IPv6 CIDR 블록을 기존 VPC와 연결하는 경우, 모든 IPv6 트래픽이 서브넷으로 그리고 서브넷으로부터 전송되도록 허용하는 규칙이 자동으로 추가됩니다. 또한 규칙 번호가 별표로 되어 있어 패킷이 번호가 매겨진 다른 어떤 규칙과도 일치하지 않을 경우에는 거부되도록 하는 규칙도 추가합니다. 이 규칙은 수정하거나 제거할 수 없습니다. 다음은 IPv4와 IPv6를 지원하는 VPC에 대한 기본 네트워크 ACL의 예시입니다.

Note

기본 네트워크 ACL의 인바운드 규칙을 수정한 경우 IPv6 블록을 VPC에 연결할 때 인바운드 IPv6 트래픽에 대한 ALLOW 규칙이 자동으로 추가되지 않습니다. 이와 마찬가지로 아웃바운드 규칙을 수정한 경우에는 아웃바운드 IPv6 트래픽에 대한 ALLOW 규칙이 자동으로 추가되지 않습니다.

인바운드

규칙 #	Type	프로토콜	포트 범위	소스	허용/거부
100	모든 IPv4 트래픽	모두	모두	0.0.0.0/0	허용
101	모든 IPv6 트래픽	모두	모두	::/0	ALLOW
*	모든 트래픽	모두	모두	0.0.0.0/0	DENY
*	모든 IPv6 트래픽	모두	모두	::/0	DENY

아웃바운드

규칙 #	Type	프로토콜	포트 범위	대상 주소	허용/거부
100	모든 트래픽	모두	모두	0.0.0.0/0	허용
101	모든 IPv6 트래픽	모두	모두	::/0	ALLOW
*	모든 트래픽	모두	모두	0.0.0.0/0	DENY
*	모든 IPv6 트래픽	모두	모두	::/0	DENY

사용자 지정 네트워크 ACL

다음은 IPv4만 지원하는 VPC에 대한 사용자 지정 네트워크 ACL의 예시입니다. 여기에는 HTTP 및 HTTPS 트래픽 수신을 허용하는 인바운드 규칙이 포함됩니다(100 및 110). 휘발성 포트 32768~65535를 포함하는 해당 인바운드 트래픽(140)에 대한 응답을 활성화하는 해당 아웃바운드 규칙이 있습니다. 적절한 휘발성 포트 범위를 선택하는 방법에 대한 자세한 내용은 [휘발성 포트](#) 단원을 참조하십시오.

이 네트워크 ACL에는 서브넷으로의 SSH 및 RDP 트래픽을 허용하는 인바운드 규칙도 포함됩니다. 아웃바운드 규칙 120은 응답이 서브넷을 떠날 수 있도록 합니다.

이 네트워크 ACL에는 서브넷 외부로의 아웃바운드 HTTP 및 HTTPS 트래픽을 허용하는 아웃바운드 규칙(100 및 110)이 있습니다. 휘발성 포트 32768~65535를 포함하는 해당 아웃바운드 트래픽(140)에 대한 응답을 활성화하는 해당 인바운드 규칙이 있습니다.

각 네트워크 ACL에는 규칙 번호가 별표로 된 기본 규칙이 포함되어 있습니다. 이 규칙은 패킷이 다른 어떤 규칙과도 일치하지 않을 경우에는 거부되도록 되어 있습니다. 이 규칙을 수정하거나 제거할 수 없습니다.

인바운드

규칙 #	Type	프로토콜	포트 범위	소스	허용/거부	설명
100	HTTP	TCP	80	0.0.0.0/0	ALLOW	어떤 IPv4 주소에서 이루어지는 인바운드 HTTP 트래픽도 모두 허용
110	HTTPS	TCP	443	0.0.0.0/0	ALLOW	어떤 IPv4 주소에서 이루어지는 인바운드 HTTPS 트래픽도 모두 허용
120	SSH	TCP	22	192.0.2.0 /24	ALLOW	홈 네트워크의 퍼블릭 IPv4 주소 범위로부터의 인바운드 SSH 트래픽 허용(인터넷 게이트웨이를 통해)
130	RDP	TCP	3389	192.0.2.0 /24	ALLOW	홈 네트워크의 퍼블릭 IPv4 주소 범위로부터 웹 서버로의 인바운드 RDP 트래픽 허용(인터넷 게이트웨이를 통해)
140	사용자 지정 TCP	TCP	32768-65535	0.0.0.0/0	ALLOW	인터넷으로부터의 인바운드 리턴 IPv4 트래픽 허용(즉, 서브넷에

규칙 #	Type	프로토콜	포트 범위	소스	허용/거부	설명
						서 시작되는 요청에 대해). 이 범위는 예시일 뿐입니다.
*	모든 트래픽	모두	모두	0.0.0.0/0	DENY	이전 규칙에서 아직 처리하지 않은 모든 인바운드 IPv4 트래픽 거부 (수정 불가)

아웃바운드

규칙 #	Type	프로토콜	포트 범위	대상 주소	허용/거부	설명
100	HTTP	TCP	80	0.0.0.0/0	ALLOW	서브넷에서 인터넷으로의 아웃바운드 IPv4 HTTP 트래픽 허용
110	HTTPS	TCP	443	0.0.0.0/0	ALLOW	서브넷에서 인터넷으로의 아웃바운드 IPv4 HTTPS 트래픽 허용
120	SSH	TCP	1024~65535	192.0.2.0/24	ALLOW	홈 네트워크의 퍼블릭 IPv4 주소 범위의 아웃바운드 반환 SSH 트래픽이 허용됩니다 (인터넷 게이트웨이 경우).
140	사용자 지정 TCP	TCP	32768-65535	0.0.0.0/0	ALLOW	인터넷에서 클라이언트에 대한 아웃바운드 IPv4 응답 허용(예: 서브넷에 있는 웹 서버를 방문하는 사람들에게 웹 페이지 제공).

규칙 #	Type	프로토콜	포트 범위	대상 주소	허용/거부	설명
						이 범위는 예시일 뿐입니다.
*	모든 트래픽	모두	모두	0.0.0.0/0	DENY	이전 규칙에서 아직 처리하지 않은 모든 아웃바운드 IPv4 트래픽 거부(수정 불가)

패킷이 서브넷에 도착할 때, 이를 서브넷이 연결되어 있는 ACL의 인바운드 규칙에 대해 평가합니다(규칙 목록의 맨 위에서 시작하여 맨 아래로 이동). 다음은 패킷이 SSL 포트(443)로 전달되는 경우 평가가 이루어지는 방식입니다. 이 패킷은 처음으로 평가되는 규칙(규칙 100)과 일치하지 않습니다. 이 패킷은 패킷을 서브넷으로 수신되도록 허용하는 두 번째 규칙(110)과도 일치하지 않습니다. 패킷이 포트 139(NetBIOS)로 보내진 경우 어떠한 규칙과도 일치하지 않으며, * 규칙이 최종적으로 해당 패킷을 거부합니다.

광범위한 포트를 합법적으로 열어야 하지만 거부하려는 범위 내에 특정 포트가 있는 경우 거부 규칙을 추가할 수 있습니다. 광범위한 포트 트래픽을 허용하는 규칙보다 거부 규칙을 먼저 테이블에 배치하십시오.

사용 사례에 따라 허용 규칙을 추가합니다. 예를 들어 DNS 확인을 위해 포트 53에서 아웃바운드 TCP 및 UDP 액세스를 허용하는 규칙을 추가할 수 있습니다. 추가하는 모든 규칙에 대해 응답 트래픽을 허용하는 해당 인바운드 또는 아웃바운드 규칙이 있는지 확인합니다.

다음은 연결된 IPv6 CIDR 블록이 있는 VPC에 대한 사용자 지정 네트워크 ACL의 예시입니다. 이 네트워크 ACL에는 모든 IPv6 HTTP 및 HTTPS 트래픽에 대한 규칙이 포함됩니다. 이 경우 IPv4 트래픽에 대한 기존 규칙 사이에 새 규칙이 삽입되었습니다. IPv4 규칙 다음에 규칙을 더 높은 번호의 규칙으로 추가할 수도 있습니다. IPv4 및 IPv6 트래픽은 분리되어 있습니다. 따라서 IPv4 트래픽에 대한 규칙 중 어느 것도 IPv6 트래픽에 적용되지 않습니다.

인바운드

규칙 #	Type	프로토콜	포트 범위	소스	허용/거부	설명
100	HTTP	TCP	80	0.0.0.0/0	ALLOW	어떤 IPv4 주소에서 이루어지는 인바운드 HTTP 트래픽도 모두 허용

규칙 #	Type	프로토콜	포트 범위	소스	허용/거부	설명
105	HTTP	TCP	80	::/0	허용	어떤 IPv6 주소에서 이루어지는 인바운드 HTTP 트래픽도 모두 허용
110	HTTPS	TCP	443	0.0.0.0/0	ALLOW	어떤 IPv4 주소에서 이루어지는 인바운드 HTTPS 트래픽도 모두 허용
115	HTTPS	TCP	443	::/0	허용	어떤 IPv6 주소에서 이루어지는 인바운드 HTTPS 트래픽도 모두 허용
120	SSH	TCP	22	192.0.2.0/24	ALLOW	홈 네트워크의 퍼블릭 IPv4 주소 범위로부터의 인바운드 SSH 트래픽 허용(인터넷 게이트웨이를 통해)
130	RDP	TCP	3389	192.0.2.0/24	ALLOW	홈 네트워크의 퍼블릭 IPv4 주소 범위로부터 웹 서버로의 인바운드 RDP 트래픽 허용(인터넷 게이트웨이를 통해)
140	사용자 지정 TCP	TCP	32768-65535	0.0.0.0/0	ALLOW	인터넷으로부터의 인바운드 리턴 IPv4 트래픽 허용(즉, 서브넷에서 시작되는 요청에 대해). 이 범위는 예시일 뿐입니다.

규칙 #	Type	프로토콜	포트 범위	소스	허용/거부	설명
145	사용자 지정 TCP	TCP	32768-65535	::/0	ALLOW	인터넷으로부터의 인바운드 리턴 IPv6 트래픽 허용(즉, 서브넷에서 시작되는 요청에 대해). 이 범위는 예시일 뿐입니다.
*	모든 트래픽	모두	모두	0.0.0.0/0	DENY	이전 규칙에서 아직 처리하지 않은 모든 인바운드 IPv4 트래픽 거부(수정 불가)
*	모든 트래픽	모두	모두	::/0	DENY	이전 규칙에서 아직 처리하지 않은 모든 인바운드 IPv6 트래픽 거부(수정 불가)

아웃바운드

규칙 #	Type	프로토콜	포트 범위	대상 주소	허용/거부	설명
100	HTTP	TCP	80	0.0.0.0/0	ALLOW	서브넷에서 인터넷으로의 아웃바운드 IPv4 HTTP 트래픽 허용
105	HTTP	TCP	80	::/0	허용	서브넷에서 인터넷으로의 아웃바운드 IPv6 HTTP 트래픽 허용
110	HTTPS	TCP	443	0.0.0.0/0	ALLOW	서브넷에서 인터넷으로의 아웃바운드 IPv4 HTTPS 트래픽 허용

규칙 #	Type	프로토콜	포트 범위	대상 주소	허용/거부	설명
115	HTTPS	TCP	443	::/0	허용	서브넷에서 인터넷으로의 아웃바운드 IPv6 HTTPS 트래픽 허용
140	사용자 지정 TCP	TCP	32768-65535	0.0.0.0/0	ALLOW	인터넷에서 클라이언트에 대한 아웃바운드 IPv4 응답 허용(예: 서브넷에 있는 웹 서버를 방문하는 사람들에게 웹 페이지 제공). 이 범위는 예시일 뿐입니다.
145	사용자 지정 TCP	TCP	32768-65535	::/0	허용	인터넷에서 클라이언트에 대한 아웃바운드 IPv6 응답 허용(예: 서브넷에 있는 웹 서버를 방문하는 사람들에게 웹 페이지 제공). 이 범위는 예시일 뿐입니다.
*	모든 트래픽	모두	모두	0.0.0.0/0	DENY	이전 규칙에서 아직 처리하지 않은 모든 아웃바운드 IPv4 트래픽 거부(수정 불가)
*	모든 트래픽	모두	모두	::/0	DENY	이전 규칙에서 아직 처리하지 않은 모든 아웃바운드 IPv6 트래픽 거부(수정 불가)

사용자 지정 네트워크 ACL 및 기타 AWS 서비스

사용자 지정 네트워크 ACL을 생성하는 경우 다른 AWS 서비스를 사용하여 생성하는 리소스에 미칠 수 있는 영향에 주의해야 합니다.

Elastic Load Balancing을 사용하는 경우 사용자의 백엔드 인스턴스에 대한 서브넷에 소스가 0.0.0.0/0 또는 서브넷의 CIDR인 모든 트래픽에 대해 거부 규칙을 추가한 네트워크 ACL이 있으면 로드 밸런서가 인스턴스에 대한 상태 확인을 수행할 수 없습니다. 로드 밸런서 및 백엔드 인스턴스에 권장되는 네트워크 ACL 규칙에 대한 자세한 내용은 Classic Load Balancer 사용 설명서의 [VPC의 로드 밸런서를 위한 네트워크 ACL](#)을 참조하십시오.

취발성 포트

이전 단원에서 예로 든 네트워크 ACL에는 32768-65535 범위의 취발성 포트가 사용됩니다. 하지만 사용하거나 통신하는 클라이언트의 유형에 따라 다른 범위의 네트워크 ACL을 사용할 수 있습니다.

요청을 시작하는 클라이언트가 취발성 포트 범위를 선택합니다. 범위는 클라이언트의 운영 체제에 따라 다릅니다.

- 다수의 Linux 커널(Amazon Linux 커널 포함)이 포트 32768-61000을 사용합니다.
- Elastic Load Balancing에서 시작된 요청은 포트 1024-65535를 사용합니다.
- Windows Server 2003까지의 Windows 운영 체제에서는 포트 1025-5000을 사용합니다.
- Windows Server 2008 이상 버전은 포트 49152-65535를 사용합니다.
- NAT 게이트웨이는 포트 1024 - 65535를 사용합니다.
- AWS Lambda 함수는 포트 1024-65535를 사용합니다.

예를 들어, 인터넷을 통해 Windows 10 클라이언트로부터 VPC에 있는 웹 서버로 요청이 수신되는 경우, 네트워크 ACL에는 포트 49152-65535로 트래픽을 전달할 수 있도록 하는 아웃바운드 규칙이 있어야 합니다.

VPC의 인스턴스가 요청을 시작하는 클라이언트인 경우, 사용자의 네트워크 ACL에 인스턴스의 유형(Amazon Linux, Windows Server 2008 등)에 특정한 취발성 포트 트래픽을 전달할 수 있도록 하는 인바운드 규칙이 있어야 합니다.

실제로는 VPC에서 퍼블릭 쪽 인스턴스로 향하는 트래픽을 시작할 수도 있는 다양한 유형의 클라이언트를 포괄하기 위해, 취발성 포트 1024-65535를 열 수 있습니다. 하지만 그 범위 내에 있는 악성 포트의 트래픽을 거부하기 위한 규칙을 ACL에 추가할 수도 있습니다. 광범위한 임시 포트를 여는 허용 규칙보다 거부 규칙을 먼저 테이블에 배치해야 합니다.

경로 MTU 검색

경로 MTU 검색을 사용하여 두 디바이스 간의 경로 MTU를 확인할 수 있습니다. 경로 MTU는 발신 호스트와 수신 호스트 간의 경로에서 지원되는 최대 패킷 크기입니다.

IPv4의 경우 호스트가 수신 호스트의 MTU 또는 경로를 따르는 디바이스의 MTU보다 큰 패킷을 전송하는 경우 수신 호스트 또는 디바이스가 패킷을 삭제한 다음 Destination Unreachable: Fragmentation Needed and Don't Fragment was Set(유형 3, 코드 4)과 같은 ICMP 메시지를 반환합니다. 이렇게 하면 전송 호스트에 페이로드를 여러 개의 작은 패킷으로 분할한 다음 다시 전송하도록 지시합니다.

IPv6 프로토콜은 네트워크의 조각화를 지원하지 않습니다. 호스트가 수신 호스트의 MTU 또는 경로를 따르는 디바이스의 MTU보다 큰 패킷을 전송하는 경우 수신 호스트 또는 디바이스가 패킷을 삭제한 다음 ICMPv6 Packet Too Big (PTB)(유형 2)과 같은 ICMP 메시지를 반환합니다. 이렇게 하면 전송 호스트에 페이로드를 여러 개의 작은 패킷으로 분할한 다음 다시 전송하도록 지시합니다.

서브넷에 있는 호스트 간의 최대 MTU(전송 단위)가 다르거나 인스턴스가 인터넷을 통해 피어와 통신하는 경우 인바운드 및 아웃바운드 네트워크 ACL 규칙을 추가해야 합니다. 이렇게 하면 경로 MTU 검색이 올바르게 작동하고 패킷 손실을 방지할 수 있습니다. 유형에 대해 사용자 지정 ICMP 규칙을 선택하고 포트 범위(유형 3, 코드 4)에 대해 대상에 연결할 수 없음, 조각화 필요, DF 플래그 설정을 선택합니다. traceroute를 사용할 경우에는 다음 규칙도 추가합니다. 즉, 유형에 사용자 지정 ICMP 규칙, 포트 범위에 시간 초과, TTL 전송 만료(유형 11, 코드 0)를 선택합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [EC2 인스턴스에 대한 네트워크 MTU\(최대 전송 단위\)](#)를 참조하세요.

네트워크 ACL 작업

다음 작업은 Amazon VPC 콘솔을 사용한 네트워크 ACL 작업 방법을 보여 줍니다.

작업

- [네트워크 ACL 연결 확인](#)
- [네트워크 ACL 생성](#)
- [규칙 추가 및 삭제](#)
- [서브넷을 네트워크 ACL과 연결](#)
- [서브넷에서 네트워크 ACL 연결 해제](#)
- [서브넷의 네트워크 ACL 변경](#)
- [네트워크 ACL 삭제](#)

- [API 및 명령 개요](#)
- [Firewall Manager를 사용하여 네트워크 ACL 관리](#)

네트워크 ACL 연결 확인

Amazon VPC 콘솔을 사용하여 특정 서브넷과 연결되어 있는 네트워크 ACL을 확인할 수 있습니다. 네트워크 ACL은 복수의 서브넷과 연결될 수 있으므로 특정 네트워크 ACL과 연결되어 있는 서브넷을 확인할 수도 있습니다.

서브넷과 연결되어 있는 네트워크 ACL을 확인하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Subnets]를 선택한 후 서브넷을 선택합니다.

서브넷과 연결된 네트워크 ACL은 네트워크 ACL의 규칙과 함께 [Network ACL] 탭에 포함되어 있습니다.

네트워크 ACL과 연결되어 있는 서브넷을 확인하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Network ACLs]를 선택합니다. [Associated With] 열에 각 네트워크 ACL에 연결된 서브넷의 수가 표시됩니다.
3. 네트워크 ACL을 선택합니다.
4. 세부 정보 창에서 서브넷 연결을 선택하여 네트워크 ACL과 연결된 서브넷을 표시합니다.

네트워크 ACL 생성

VPC에 대한 사용자 지정 네트워크 ACL을 생성할 수 있습니다. 기본적으로, 생성된 네트워크 ACL은 사용자가 규칙을 추가할 때까지는 모든 인바운드 및 아웃바운드 트래픽을 차단하며, 사용자가 명시적으로 특정 서브넷과 연결할 때까지는 서브넷과 연결되지 않습니다.

네트워크 ACL을 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Network ACLs]를 선택합니다.
3. [Create Network ACL]을 선택합니다.

- 네트워크 ACL 생성 대화 상자에서 선택적으로 네트워크 ACL의 이름을 지정한 다음 VPC 목록에서 VPC의 ID를 선택합니다. 그런 다음 예, 생성을 선택합니다.

규칙 추가 및 삭제

ACL에서 규칙을 추가하거나 삭제할 때 ACL과 연관된 서브넷이 변경될 수 있습니다. 서브넷의 인스턴스를 종료하고 다시 시작할 필요가 없습니다. 변경 사항은 잠시 후 적용됩니다.

Important

동시에 규칙을 추가하고 삭제하는 경우 매우 주의하세요. 네트워크 ACL 규칙은 VPC에 들어가거나 나올 수 있는 네트워크 트래픽 유형을 정의합니다. 인바운드 또는 아웃바운드 규칙을 삭제한 다음 [Amazon VPC 할당량](#)에서 허용된 항목보다 많은 새 항목을 추가하면 삭제하도록 선택한 항목이 제거되고 새 항목이 추가되지 않습니다. 이로 인해 예기치 않은 연결 문제가 발생하고 의도하지 않게 VPC에 대한 액세스가 차단될 수 있습니다.

Amazon EC2 API 또는 명령줄 도구를 사용하는 경우에는 규칙을 수정할 수 없습니다. 규칙을 추가 및 삭제할 수만 있습니다. Amazon VPC 콘솔을 사용하는 경우에는 기존 규칙의 항목을 수정할 수 있습니다. 콘솔은 기존 규칙을 제거하고 새 규칙을 추가합니다. ACL에서 규칙의 순서를 변경할 필요가 있는 경우에는 새 규칙 번호와 함께 새 규칙을 추가한 후에 원래 규칙은 삭제해야 합니다.

네트워크 ACL에 규칙을 추가하려면

- <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
- 탐색 창에서 [Network ACLs]를 선택합니다.
- 세부 정보 창에서 추가해야 할 규칙의 유형에 따라 [Inbound Rules] 또는 [Outbound Rules] 탭을 선택한 후 [Edit]를 선택합니다.
- [Rule #]에서 규칙 번호를 입력합니다(예: 100). 규칙 번호가 네트워크 ACL에서 이미 사용되고 있는 번호이면 안 됩니다. 규칙은 가장 낮은 번호부터 시작해서 순서대로 처리됩니다.

순차 번호(101, 102, 103)를 사용하는 대신 규칙 번호 간에 간격을 두는 것이 좋습니다(예: 100, 200, 300). 그러면 기존 규칙의 번호를 다시 매길 필요 없이 새 규칙을 더 쉽게 추가할 수 있습니다.

- [Type] 목록에서 규칙을 선택합니다. 예를 들어 HTTP에 대한 규칙을 추가하려면 [HTTP]를 선택합니다. 모든 TCP 트래픽을 허용하는 규칙을 추가하려면 [All TCP]를 선택합니다. 이런 옵션 중 일부에 대해서는(예: HTTP) 포트가 자동으로 입력됩니다. 나열되지 않은 프로토콜을 사용하려면 [Custom Protocol Rule]을 선택합니다.

6. (선택 사항) 사용자 지정 프로토콜 규칙을 생성할 경우 [Protocol] 목록에서 프로토콜의 번호와 이름을 선택합니다. 자세한 내용은 [프로토콜 번호의 IANA 목록](#)을 참조하십시오.
7. (선택 사항) 선택한 프로토콜에 포트 번호가 필요한 경우 해당 포트 번호를 입력하거나 하이픈으로 구분된 포트 범위(예: 49152-65535)를 입력합니다.
8. (인바운드 또는 아웃바운드 규칙인지에 따라) [Source] 또는 [Destination] 필드에 규칙이 적용되는 CIDR 범위를 입력합니다.
9. [Allow/Deny] 목록에서 지정된 트래픽을 허용하려면 [ALLOW]를, 지정된 트래픽을 거부하려면 [DENY]를 선택합니다.
10. (선택 사항) 또 다른 규칙을 추가하려면 [Add another rule]을 선택하고 필요에 따라 4~9단계를 반복합니다.
11. 마치면 [Save]를 선택합니다.

네트워크 ACL에서 규칙을 삭제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Network ACLs]를 선택한 후 네트워크 ACL을 선택합니다.
3. 세부 정보 창에서 [Inbound Rules] 또는 [Outbound Rules] 탭을 선택한 후 [Edit]를 선택합니다. 삭제하려는 규칙에서 [Remove]를 선택한 후 [Save]를 선택합니다.

서브넷을 네트워크 ACL과 연결

특정 서브넷에 네트워크 ACL의 규칙을 적용하려면 서브넷을 네트워크 ACL과 연결해야 합니다. 네트워크 ACL을 여러 서브넷과 연결할 수 있습니다. 그러나 서브넷은 하나의 네트워크 ACL에만 연결될 수 있습니다. 기본적으로, 특정 ACL과 연결되지 않은 서브넷은 기본 네트워크 ACL과 연결됩니다.

서브넷을 네트워크 ACL과 연결하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Network ACLs]를 선택한 후 네트워크 ACL을 선택합니다.
3. 세부 정보 창의 [Subnet Associations] 탭에서 [Edit]를 선택합니다. 네트워크 ACL과 연결할 서브넷에 대한 [Associate] 확인란을 선택한 후 [Save]를 선택합니다.

서브넷에서 네트워크 ACL 연결 해제

서브넷에서 사용자 지정 네트워크 ACL을 연결 해제할 수 있습니다. 서브넷이 사용자 지정 네트워크 ACL과의 연결이 끊어지면 기본 네트워크 ACL과 자동으로 연결됩니다.

네트워크 ACL에서 서브넷의 연결을 끊으려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Network ACLs]를 선택한 후 네트워크 ACL을 선택합니다.
3. 세부 정보 창에서 [Subnet Associations] 탭을 선택합니다.
4. [Edit]를 선택한 다음, 서브넷에 대한 [Associate] 확인란을 선택 취소합니다. Save를 선택합니다.

서브넷의 네트워크 ACL 변경

서브넷과 연결되어 있는 네트워크 ACL을 변경할 수 있습니다. 예를 들어 서브넷을 생성하면 생성된 서브넷이 처음에는 기본 네트워크 ACL과 연결됩니다. 서브넷을 사용자가 생성한 사용자 지정 네트워크 ACL과 대신 연결할 수도 있을 것입니다.

서브넷의 네트워크 ACL을 변경한 후에는 서브넷의 인스턴스를 종료했다가 다시 시작할 필요가 없습니다. 변경 사항은 잠시 후 적용됩니다.

서브넷의 네트워크 ACL 연결을 변경하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Subnets]를 선택한 후 서브넷을 선택합니다.
3. [Network ACL] 탭을 선택한 후 [Edit]를 선택합니다.
4. 를 로 변경 목록에서 서브넷과 연결할 네트워크 ACL을 선택한 다음 저장을 선택합니다.

네트워크 ACL 삭제

네트워크 ACL과 연결된 서브넷이 없는 경우에만 네트워크 ACL을 삭제할 수 있습니다. 기본 네트워크 ACL은 삭제할 수 없습니다.

네트워크 ACL을 삭제하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 탐색 창에서 [Network ACLs]를 선택합니다.

3. 네트워크 ACL을 선택한 후 [Delete]를 선택합니다.
4. 확인 대화 상자에서 [Yes, Delete]를 선택합니다.

API 및 명령 개요

명령줄 또는 API를 사용하여 이 페이지에서 설명하는 작업을 수행할 수 있습니다. 명령줄 인터페이스 및 사용 가능한 API 목록에 대한 자세한 내용은 [Amazon VPC 작업](#) 섹션을 참조하십시오.

VPC에 대한 네트워크 ACL 만들기

- [create-network-acl](#)(AWS CLI)
- [New-EC2NetworkAcl](#)(AWS Tools for Windows PowerShell)

한 개 이상의 네트워크 ACL에 대해 설명

- [describe-network-acls](#)(AWS CLI)
- [Get-EC2NetworkAcl](#)(AWS Tools for Windows PowerShell)

네트워크 ACL에 규칙 추가

- [create-network-acl-entry](#)(AWS CLI)
- [New-EC2NetworkAclEntry](#)(AWS Tools for Windows PowerShell)

네트워크 ACL에서 규칙 삭제

- [delete-network-acl-entry](#)(AWS CLI)
- [Remove-EC2NetworkAclEntry](#)(AWS Tools for Windows PowerShell)

네트워크 ACL에 있는 기존 규칙 바꾸기

- [replace-network-acl-entry](#)(AWS CLI)
- [Set-EC2NetworkAclEntry](#)(AWS Tools for Windows PowerShell)

네트워크 ACL 연결 바꾸기

- [replace-network-acl-association](#)(AWS CLI)

- [Set-EC2NetworkAclAssociation](#)(AWS Tools for Windows PowerShell)

네트워크 ACL 삭제

- [delete-network-acl](#)(AWS CLI)
- [Remove-EC2NetworkAcl](#)(AWS Tools for Windows PowerShell)

Firewall Manager를 사용하여 네트워크 ACL 관리

AWS Firewall Manager는 여러 계정과 리소스 간에 네트워크 ACL 관리 및 유지 관리 작업을 간소화합니다. Firewall Manager를 사용하여 조직의 계정 및 서브넷을 모니터링하고 정의한 네트워크 ACL 구성을 자동으로 적용할 수 있습니다. Firewall Manager는 조직 전체를 보호해야 하거나 중앙 관리자 계정으로 자동 보호할 새 서브넷을 자주 추가하는 경우에 특히 유용합니다.

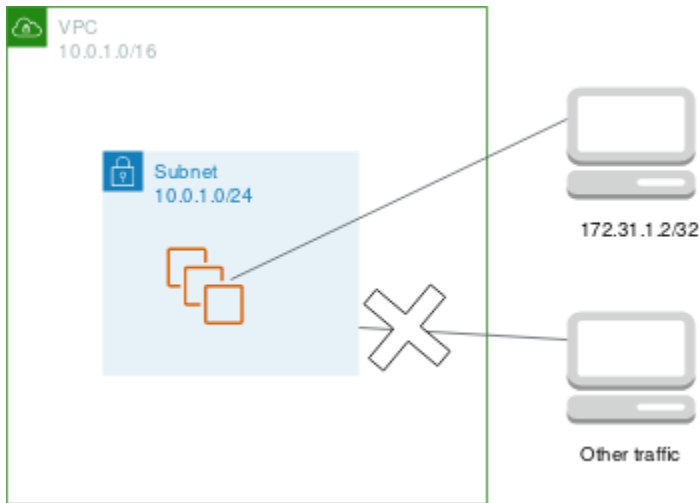
Firewall Manager 네트워크 ACL 정책을 사용하면 단일 관리자 계정을 사용하여 조직 전체에서 사용하는 네트워크 ACL에 정의하려는 최소 규칙 세트를 구성, 모니터링 및 관리할 수 있습니다. 조직의 어떤 계정과 서브넷이 Firewall Manager 정책 범위 내에 속하는지 지정합니다. Firewall Manager는 범위 내 서브넷에 대한 네트워크 ACL의 규정 준수 상태를 보고하며, 비준수 네트워크 ACL을 자동으로 수정하여 규정을 준수하도록 Firewall Manager를 구성할 수 있습니다.

Firewall Manager를 사용하여 네트워크 ACL을 관리하는 방법에 대한 자세한 내용은 AWS Firewall Manager 개발자 안내서의 다음 리소스를 참조하세요.

- [AWS Firewall Manager 필수 조건](#)
- [AWS Firewall Manager Amazon VPC 네트워크 ACL 정책 시작하기](#)
- [Amazon Virtual Private Cloud 네트워크 액세스 제어 목록\(ACL\) 정책](#)

예: 서브넷의 인스턴스에 대한 액세스 제어

이 예에서는 서브넷의 인스턴스가 서로 통신할 수 있으며, 신뢰할 수 있는 원격 컴퓨터로부터 액세스될 수 있습니다. 원격 컴퓨터는 로컬 네트워크의 컴퓨터이거나 다른 서브넷 또는 VPC의 인스턴스일 수 있습니다. 원격 컴퓨터를 통해 인스턴스에 연결하여 관리 작업을 수행할 수 있습니다. 사용자의 보안 그룹 규칙 및 네트워크 ACL 규칙이 원격 컴퓨터의 IP 주소(172.31.1.2/32)로부터의 액세스를 허용합니다. 인터넷 또는 다른 네트워크로부터의 다른 모든 트래픽은 거부됩니다. 이 시나리오는 인스턴스에 대한 보안 그룹 또는 보안 그룹 규칙을 변경할 수 있고 네트워크 ACL을 방어의 백업 계층으로 사용할 수 있는 유연성을 제공합니다.



다음은 인스턴스와 연결할 보안 그룹의 예입니다. 보안 그룹은 상태가 저장됩니다. 따라서 인바운드 트래픽에 대한 응답을 허용하는 규칙은 필요하지 않습니다.

인바운드

프로토콜 유형	프로토콜	포트 범위	소스	설명
모든 트래픽	모두	모두	sg-123456 7890abcdef0	이 보안 그룹과 연결된 모든 인스턴스는 서로 통신할 수 있습니다.
SSH	TCP	22	172.31.1.2/32	원격 컴퓨터로부터의 인바운드 SSH 액세스를 허용합니다.

아웃바운드

프로토콜 유형	프로토콜	포트 범위	대상 주소	설명
모든 트래픽	모두	모두	sg-123456 7890abcdef0	이 보안 그룹과 연결된 모든 인스턴스는 서로 통신할 수 있습니다.

다음은 인스턴스의 서브넷과 연결할 네트워크 ACL 예제입니다. 네트워크 ACL 규칙은 서브넷의 모든 인스턴스에 적용됩니다. 네트워크 ACL은 상태가 저장되지 않습니다. 따라서 인바운드 트래픽에 대한 응답을 허용하는 규칙이 필요합니다.

인바운드

규칙 #	Type	프로토콜	포트 범위	소스	허용/거부	설명
100	SSH	TCP	22	172.31.1. 2/32	허용	원격 컴퓨터로부터의 인바운드 트래픽을 허용합니다.
*	모든 트래픽	모두	모두	0.0.0.0/0	DENY	다른 모든 인바운드 트래픽을 거부합니다.

아웃바운드

규칙 #	Type	프로토콜	포트 범위	대상 주소	허용/거부	설명
100	사용자 지정 TCP	TCP	1024~65535	172.31.1. 2/32	허용	원격 컴퓨터에 대한 아웃바운드 응답을 허용합니다.
*	모든 트래픽	모두	모두	0.0.0.0/0	DENY	다른 모든 아웃바운드 트래픽을 거부합니다.

잘못하여 보안 그룹 규칙을 너무 허용하는 경우, 이 예제의 네트워크 ACL 규칙이 지정된 IP 주소의 액세스만 계속 허용하게 됩니다. 예를 들어 다음 보안 그룹에는 모든 IP 주소에서 인바운드 SSH 액세스를 허용하는 규칙이 포함되어 있습니다. 그러나 네트워크 ACL을 사용하는 서브넷의 인스턴스와 이 보안 그룹을 연결하면 서브넷과 원격 컴퓨터 내의 다른 인스턴스만 인스턴스에 액세스할 수 있습니다. 네트워크 ACL 규칙이 서브넷에 대한 다른 인바운드 트래픽을 거부하기 때문입니다.

인바운드

유형	프로토콜	포트 범위	소스	설명
모든 트래픽	모두	모두	sg-123456 7890abcdef0	이 보안 그룹과 연결된 모든 인스턴스는 서로 통신할 수 있습니다.
SSH	TCP	22	0.0.0.0/0	모든 IP 주소로부터의 SSH 액세스를 허용합니다.

아웃바운드

유형	프로토콜	포트 범위	대상 주소	설명
모든 트래픽	모두	모두	0.0.0.0/0	모든 아웃바운드 트래픽을 허용합니다.

연결 문제 해결

Reachability Analyzer는 정적 구성 분석 도구입니다. Reachability Analyzer를 사용하여 VPC의 두 리소스 간 네트워크 연결성을 분석하고 디버깅할 수 있습니다. Reachability Analyzer에서는 연결할 수 있는 경우 이러한 리소스 간 가상 경로에 대한 흠별 세부 정보가 생성되고, 그렇지 않다면 차단 구성 요소가 식별됩니다. 예를 들면 누락되거나 잘못 구성된 네트워크 ACL 규칙이 식별될 수 있습니다.

자세한 내용은 [Reachability Analyzer 사용 설명서](#)를 참조하십시오.

Amazon Virtual Private Cloud에서의 복원성

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 중심으로 구축됩니다. AWS 리전에서는 짧은 대기 시간, 높은 처리량, 높은 중복성을 갖춘 네트워크를 사용하여 연결된 물리적으로 분리되고 격리된 다수의 가용 영역을 제공합니다. 가용 영역을 사용하면 중단 없이 영역 간에 자동으로 장애 극복 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 다중 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

AWS 리전 및 가용 영역에 대한 자세한 정보는 [AWS 글로벌 인프라](#)를 참조하십시오.

워크로드의 복원력 요구 사항을 충족하도록 VPC를 구성할 수 있습니다. 자세한 내용은 다음 자료를 참조하세요.

- [복원력 패턴 및 장단점 이해](#)(AWS 아키텍처 블로그)
- [네트워크 토폴로지 계획](#)(AWS Well-Architected 프레임워크)
- [Amazon Virtual Private Cloud 연결 옵션](#)(AWS 백서)

Amazon Virtual Private Cloud에 대한 규정 준수 확인

AWS 서비스가 특정 규정 준수 프로그램의 범위에 포함되는지 알아보려면 [규정 준수 프로그램 제공 범위 내 AWS 서비스](#)를 참조하고 관심 있는 규정 준수 프로그램을 선택하십시오. 일반적인 정보는 [AWS 규정 준수 프로그램](#)을 참조하십시오.

AWS Artifact(을)를 사용하여 타사 감사 보고서를 다운로드할 수 있습니다. 자세한 내용은 [AWS Artifact에서 보고서 다운로드](#)를 참조하십시오.

AWS 서비스 사용 시 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표 및 관련 법률과 규정에 따라 결정됩니다. AWS에서는 규정 준수를 지원할 다음과 같은 리소스를 제공합니다.

- [보안 및 규정 준수 빠른 시작 안내서](#) - 이 배포 안내서에서는 아키텍처 고려 사항에 대해 설명하고 보안 및 규정 준수에 중점을 둔 기본 AWS 환경을 배포하기 위한 단계를 제공합니다.
- [Amazon Web Services에서 HIPAA 보안 및 규정 준수 기술 백서 설계](#) - 이 백서는 기업에서 AWS(을)를 사용하여 HIPAA를 준수하는 애플리케이션을 만드는 방법을 설명합니다.

Note

모든 AWS 서비스에 HIPAA 자격이 있는 것은 아닙니다. 자세한 내용은 [HIPAA 적격 서비스 참조](#)를 참조하십시오.

- [AWS 규정 준수 리소스](#) - 고객 조직이 속한 산업 및 위치에 적용될 수 있는 워크북 및 가이드 컬렉션입니다.
- [AWS 고객 규정 준수 가이드](#) - 규정 준수의 관점에서 공동 책임 모델을 이해합니다. 이 가이드에서는 AWS 서비스를 보호하기 위한 모범 사례를 요약하고 여러 프레임워크(미국 표준 기술 연구소(NIST), 결제 카드 산업 보안 표준 위원회(PCI), 국제 표준화기구(ISO) 등)에서 보안 제어에 대한 지침을 매핑합니다.
- AWS Config 개발자 가이드의 [규칙을 사용하여 리소스 평가](#) - AWS Config 서비스는 내부 사례, 산업 지침 및 규제에 대한 리소스 구성의 준수 상태를 평가합니다.
- [AWS Security Hub](#) - 이 AWS 서비스(은)는 AWS 내의 보안 상태에 대한 포괄적인 보기를 제공합니다. Security Hub는 보안 제어를 사용하여 AWS 리소스를 평가하고 보안 업계 표준 및 모범 사례에 대한 규정 준수를 확인합니다. 지원되는 서비스 및 제어 목록은 [Security Hub 제어 참조](#)를 참조하십시오.
- [Amazon GuardDuty](#) - 이 AWS 서비스는 의심스럽고 악의적인 활동이 있는지 환경을 모니터링하여 AWS 계정, 워크로드, 컨테이너 및 데이터에 대한 잠재적 위협을 탐지합니다. GuardDuty는 특정 규정 준수 프레임워크에서 요구하는 침입 탐지 요구 사항을 충족하여 PCI DSS와 같은 다양한 규정 준수 요구 사항을 따르는 데 도움을 줄 수 있습니다.
- [AWS Audit Manager](#) - 이 AWS 서비스는 AWS 사용을 지속해서 감사하여 위협을 관리하고 규정 및 업계 표준을 준수하는 방법을 간소화할 수 있도록 지원합니다.

VPC에 대한 보안 모범 사례

다음 모범 사례는 일반적인 지침이며 완벽한 보안 솔루션을 나타내지는 않습니다. 이러한 모범 사례는 환경에 적절하지 않거나 충분하지 않을 수 있으므로 참고용으로만 사용해 주십시오.

- VPC에 서브넷을 추가하여 애플리케이션을 호스팅하는 경우 여러 가용 영역에 서브넷을 생성합니다. 가용 영역은 AWS 리전에 중복 전원, 네트워킹 및 연결이 있는 하나 이상의 개별 데이터 센터입니다. 여러 가용 영역을 사용하면 프로덕션 애플리케이션의 가용성, 내결함성 및 확장성이 향상됩니다. 자세한 내용은 [Amazon VPC on AWS](#)를 참조하세요.
- 보안 그룹을 사용하여 서브넷의 EC2 인스턴스에 대한 트래픽을 제어합니다. 자세한 내용은 [보안 그룹](#) 단원을 참조하십시오.
- 네트워크 ACL을 사용하여 서브넷 수준에서 인바운드 및 아웃바운드 트래픽을 제어합니다. 자세한 내용은 [네트워크 ACL을 사용하여 서브넷에 대한 트래픽 제어](#) 단원을 참조하십시오.
- (AWS Identity and Access Management)(IAM) 아이덴티티 페더레이션, 사용자, 역할을 사용하여 VPC의 AWS 리소스에 대한 액세스를 관리합니다. 자세한 내용은 [Amazon VPC용 Identity and Access Management](#) 단원을 참조하십시오.

- VPC 흐름 로그를 사용하여 VPC, 서브넷 또는 네트워킹 인터페이스에서 양쪽에서 이동하는 IP 트래픽을 모니터링합니다. 자세한 내용은 [VPC 흐름 로그](#) 단원을 참조하십시오.
- Network Access Analyzer를 사용하여 VPC에서 리소스에 대한 의도하지 않은 네트워크 액세스를 식별합니다. 자세한 내용을 알아보려면 [Network Access Analyzer Guide](#)(Network Access Analyzer 설명서)를 참조하세요.
- AWS Network Firewall를 사용하여 인바운드 및 아웃바운드 트래픽을 필터링하여 VPC를 모니터링하고 보호합니다. 자세한 정보는 [AWS Network Firewall 안내서](#)를 참조하세요.
- Amazon GuardDuty로 AWS 환경 내 계정과 컨테이너, 워크로드, 데이터에 대한 잠재적인 위협을 탐지할 수 있습니다. 기본 위협 탐지에는 Amazon EC2 인스턴스와 관련된 VPC 흐름 로그 모니터링이 포함됩니다. 자세한 내용은 Amazon GuardDuty 사용 설명서의 [VPC 흐름 로그](#)를 참조하세요.

VPC 보안 관련된 자주 하는 질문에 대한 답변은 [Amazon VPC FAQ](#)의 보안 및 필터링을 참조하십시오.

다른 AWS 서비스와 함께 Amazon VPC 사용

Amazon VPC를 다른 AWS 서비스와 함께 사용하여 요구 사항에 맞는 솔루션을 구축할 수 있습니다.

내용

- [AWS PrivateLink를 사용하여 서비스에 VPC 연결](#)
- [AWS Network Firewall을 사용하여 네트워크 트래픽 필터링](#)
- [Route 53 Resolver DNS 방화벽을 사용하여 DNS 트래픽 필터링](#)
- [Reachability Analyzer를 사용하여 연결 문제 해결](#)

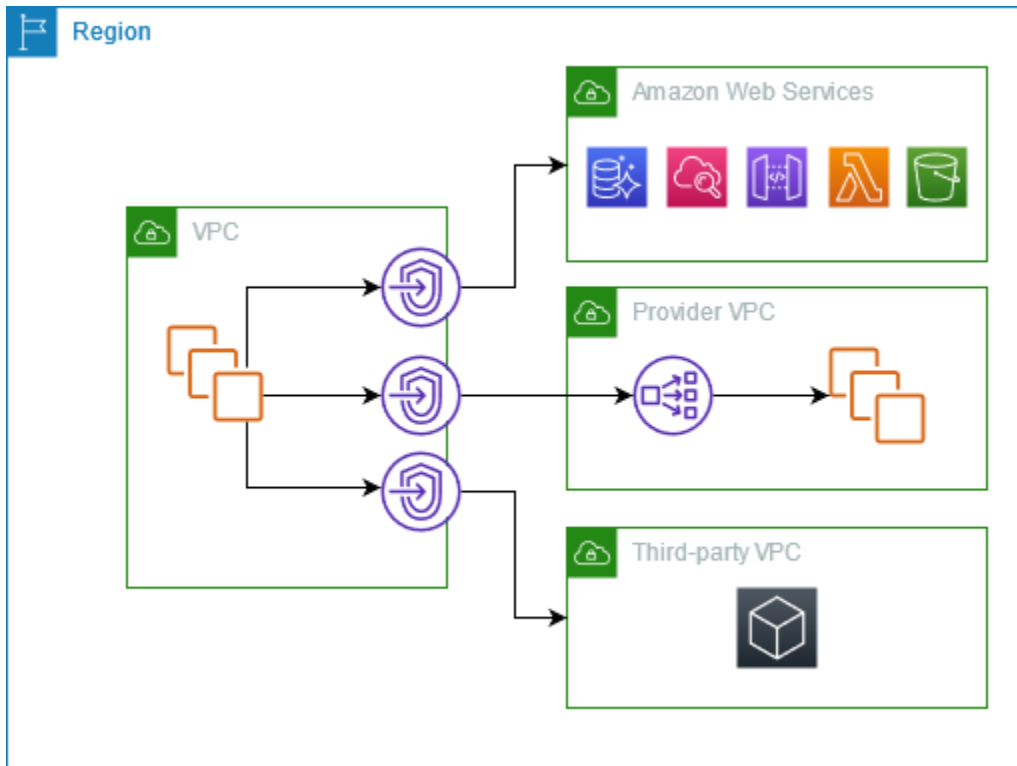
AWS PrivateLink를 사용하여 서비스에 VPC 연결

AWS PrivateLink는 Virtual Private Cloud(VPC)와 지원되는 AWS 서비스, 다른 AWS 계정에서 호스팅하는 서비스 및 지원되는 AWS Marketplace 서비스 간에 프라이빗 연결을 설정합니다. 서비스와 통신하는 데 인터넷 게이트웨이, NAT 디바이스, AWS Direct Connect 연결 또는 AWS Site-to-Site VPN 연결을 사용하지 않아도 됩니다.

AWS PrivateLink를 사용하려면 서비스 이름과 서브넷을 지정하여 VPC에 VPC 엔드포인트를 생성합니다. 그러면 서비스로 전달되는 트래픽에 대한 진입점 역할을 하는 서브넷에 탄력적 네트워크 인터페이스가 생성됩니다.

AWS PrivateLink로 구동되는 자체 VPC 엔드포인트 서비스를 생성하고 다른 AWS 고객이 해당 서비스에 액세스할 수 있도록 합니다.

다음 다이어그램에서는 AWS PrivateLink의 일반적인 사용 사례를 보여줍니다. 왼쪽의 VPC에는 프라이빗 서브넷에 여러 EC2 인스턴스와 3개의 인터페이스 VPC 엔드포인트가 있습니다. 맨 위의 VPC 엔드포인트는 AWS 서비스에 연결됩니다. 중간 VPC 엔드포인트는 다른 AWS 계정(VPC 엔드포인트 서비스)에서 호스팅하는 서비스에 연결합니다. 맨 아래의 VPC 엔드포인트는 AWS Marketplace 파트너 서비스에 연결합니다.



자세한 내용은 [AWS PrivateLink](#) 단원을 참조하십시오.

AWS Network Firewall을 사용하여 네트워크 트래픽 필터링

AWS Network Firewall을 사용하면 VPC 경계에서 네트워크 트래픽을 필터링할 수 있습니다. Network Firewall은 상태를 저장하는 관리형 네트워크 방화벽이자, 침입 탐지 및 방지 서비스입니다. 자세한 내용은 [AWS Network Firewall 개발자 안내서](#)를 참조하십시오.

다음 AWS 리소스를 사용하여 Network Firewall을 구현합니다.

Network Firewall 리소스	설명
방화벽	방화벽은 방화벽 정책의 네트워크 트래픽 필터링 동작을 보호하려는 VPC에 연결합니다. 방화벽 구성에는 방화벽 엔드포인트가 배치되는 가용 영역 및 서브넷에 대한 사양이 포함됩니다. 또한 AWS 방화벽 리소스의 방화벽 로깅 구성 및 태깅과 같은 상위 수준 설정을 정의합니다. 자세한 내용은 AWS Network Firewall의 방화벽 을 참조하세요.

Network Firewall 리소스	설명
방화벽 정책	방화벽 정책은 방화벽에 대한 모니터링 및 보호 동작을 정의합니다. 동작의 세부 정보는 정책에 추가하는 규칙 그룹과 일부 정책 기본 설정에 정의됩니다. 방화벽 정책을 사용하려면 하나 이상의 방화벽과 연결합니다. 자세한 내용은 AWS Network Firewall의 방화벽 정책 을 참조하세요.

규칙 그룹

규칙 그룹은 네트워크 트래픽을 검사하고 처리하기 위한 재사용 가능한 기준 세트입니다. 정책 구성의 일부로 방화벽 정책에 하나 이상의 규칙 그룹을 추가합니다. 상태 비저장 규칙 그룹을 정의하여 각 네트워크 패킷을 격리 상태에서 검사할 수 있습니다. 상태 비저장 규칙 그룹은 Amazon VPC 네트워크 액세스 제어 목록(ACL)과 동작 및 사용법이 유사합니다. 상태 저장 규칙 그룹을 정의하여 트래픽 흐름의 컨텍스트에서 패킷을 검사할 수도 있습니다. 상태 저장 규칙 그룹은 Amazon VPC 보안 그룹과 동작 및 사용법이 유사합니다.

자세한 내용은 [AWS Network Firewall의 규칙 그룹](#)을 참조하세요.

AWS Firewall Manager를 사용하여 AWS Organizations의 계정 및 애플리케이션에서 Network Firewall 리소스를 중앙에서 구성하고 관리할 수도 있습니다. Firewall Manager에서 단일 계정을 사용하여 여러 계정의 방화벽을 관리할 수 있습니다. 자세한 내용은 AWS WAF, AWS Firewall Manager 및 AWS Shield Advanced 개발자 안내서의 [AWS Firewall Manager](#) 섹션을 참조하세요.

Route 53 Resolver DNS 방화벽을 사용하여 DNS 트래픽 필터링

DNS 방화벽을 사용하여 VPC와 연결하는 규칙 그룹에 도메인 이름 필터링 규칙을 정의합니다. 허용하거나 차단할 도메인 이름 목록을 지정하고 차단하는 DNS 쿼리에 대한 응답을 사용자 지정할 수 있습니다. 자세한 내용은 [Route 53 Resolver DNS Firewall 설명서](#)를 참조하세요.

다음 AWS 리소스를 사용하여 DNS 방화벽을 구현합니다.

DNS 방화벽 리소스	설명
DNS 방화벽 규칙 그룹	DNS 방화벽 규칙 그룹은 DNS 쿼리를 필터링하기 위한 DNS 방화벽 규칙의 명명된 재사용 가능한 컬렉션입니다. 규칙 그룹을 필터링 규칙으로

DNS 방화벽 리소스	설명
	<p>채운 후, 규칙 그룹을 Amazon VPC의 하나 이상의 VPC와 연결합니다. 규칙 그룹을 VPC와 연결하면 VPC에 대해 DNS 방화벽 필터링을 활성화합니다. 그런 다음 Resolver가 규칙 그룹이 연결된 VPC에 대한 DNS 쿼리를 수신하면 Resolver는 필터링을 위해 쿼리를 DNS 방화벽으로 전달합니다.</p> <p>규칙 그룹 내의 각 규칙은 하나의 도메인 목록을 지정하고 도메인이 목록의 도메인 사양과 일치하는 DNS 쿼리에 대해 수행할 작업을 지정합니다. 일치하는 쿼리에 대해 허용, 차단 또는 경고를 할 수 있습니다. 차단된 쿼리에 대한 사용자 지정 응답을 정의할 수도 있습니다.</p> <p>자세한 내용은 Route 53 Resolver DNS Firewall의 규칙 그룹 및 규칙을 참조하세요.</p>
도메인 목록	<p>도메인 목록은 규칙 그룹 내부의 DNS 방화벽 규칙에서 사용하는 재사용 가능한 도메인 사양 집합입니다.</p> <p>자세한 내용은 Route 53 Resolver DNS Firewall의 도메인 목록을 참조하세요.</p>

AWS Firewall Manager를 사용하여 AWS Organizations의 계정 및 조직에서 DNS 방화벽 리소스를 중앙에서 구성하고 관리할 수도 있습니다. Firewall Manager에서 단일 계정을 사용하여 여러 계정의 방화벽을 관리할 수 있습니다. 자세한 내용은 AWS WAF, AWS Firewall Manager 및 AWS Shield Advanced 개발자 안내서의 [AWS Firewall Manager](#) 섹션을 참조하세요.

Reachability Analyzer를 사용하여 연결 문제 해결

Reachability Analyzer는 정적 구성 분석 도구입니다. Reachability Analyzer를 사용하여 VPC의 두 리소스 간 네트워크 연결성을 분석하고 디버깅할 수 있습니다. Reachability Analyzer에서는 연결할 수 있는 경우 이러한 리소스 간 가상 경로에 대한 흠벌 세부 정보가 생성되고, 그렇지 않다면 차단 구성 요소가 식별됩니다.

Reachability Analyzer를 사용하여 다음과 같은 리소스 간 연결성을 분석할 수 있습니다.

- 인스턴스
- 인터넷 게이트웨이

- 네트워크 인터페이스
- Transit Gateway
- Transit Gateway Attachment
- VPC 엔드포인트 서비스
- VPC 엔드포인트
- VPC 피어링 연결
- VPN 게이트웨이

자세한 내용은 [Reachability Analyzer 사용 설명서](#)를 참조하십시오.

VPC 예시

다음은 Virtual Private Cloud(VPC) 구성의 예시입니다.

예제

- [예시: 테스트 환경을 위한 VPC](#)
- [예시: 웹 및 데이터베이스 서버용 VPC](#)
- [예시: 프라이빗 서브넷과 NAT에 서버가 있는 VPC](#)

관련 예시

- VPC를 서로 연결하려면 Amazon VPC 피어링 설명서의 [VPC 피어링 구성](#)을 참조하세요.
- VPC를 자체 네트워크에 연결하려면 AWS Site-to-Site VPN 사용 설명서의 [Site-to-Site VPN 아키텍처](#)를 참조하세요.
- VPC를 서로 연결하거나 자체 네트워크에 연결하려면 Amazon VPC Transit Gateway의 [Transit Gateway 예](#)를 참조하세요.

추가적인 리소스

- [복원력 패턴 및 장단점 이해](#)(AWS 아키텍처 블로그)
- [네트워크 토폴로지 계획](#)(AWS Well-Architected 프레임워크)
- [Amazon Virtual Private Cloud 연결 옵션](#)(AWS 백서)

예시: 테스트 환경을 위한 VPC

이 예시에서는 개발 또는 테스트 환경으로 사용할 수 있는 VPC를 생성하는 방법을 보여줍니다. 이 VPC는 프로덕션에서 사용하기 위한 것이 아니므로 여러 가용 영역에 서버를 배포할 필요가 없습니다. 비용과 복잡성을 낮게 유지하기 위해 단일 가용 영역에 서버를 배포할 수 있습니다.

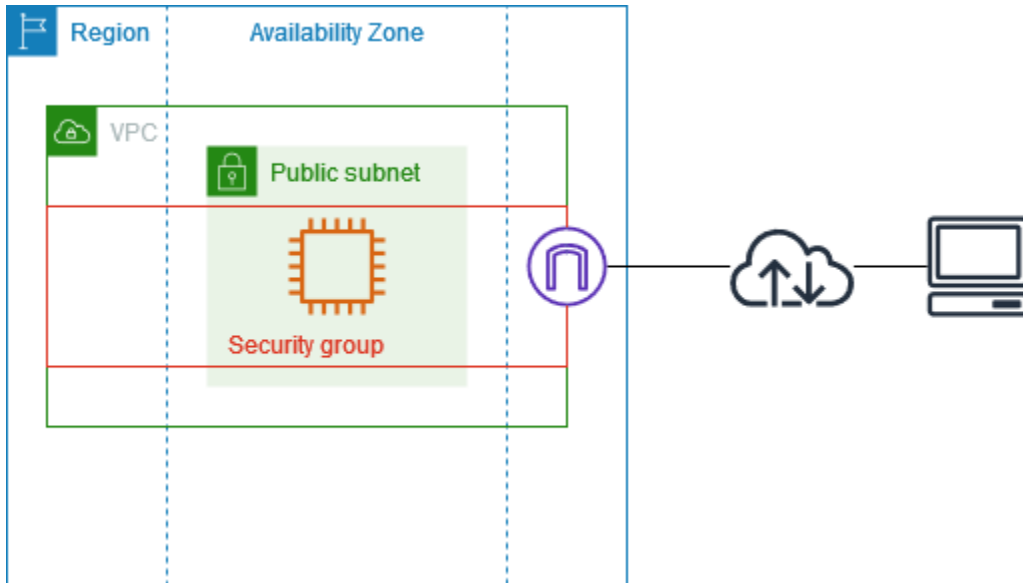
내용

- [개요](#)
- [VPC 생성](#)
- [애플리케이션 배포](#)
- [구성 테스트](#)

• 정리

개요

다음 다이어그램에서는 이 예시에 포함된 리소스의 개요를 제공합니다. VPC는 단일 가용 영역과 인터넷 게이트웨이에 퍼블릭 서브넷이 있습니다. 서버는 퍼블릭 서브넷에서 실행되는 EC2 인스턴스입니다. 인스턴스의 보안 그룹은 사용자 컴퓨터의 SSH 트래픽과 개발 또는 테스트 활동에 특별히 필요한 기타 트래픽을 허용합니다.



라우팅

Amazon VPC 콘솔을 사용하여 이 VPC를 생성하면 로컬 경로와 인터넷 게이트웨이에 대한 경로가 있는 퍼블릭 서브넷에 대한 라우팅 테이블이 생성됩니다. 다음은 IPv4와 IPv6 모두에 대한 경로가 있는 라우팅 테이블 예시입니다. 이중 스택 서브넷 대신 IPv4 전용 서브넷을 생성하는 경우 라우팅 테이블에는 IPv4 경로만 있습니다.

대상 주소	대상
<i>10.0.0.0/16</i>	로컬
<i>2001:db8:1234:1a00::/56</i>	로컬
0.0.0.0/0	<i>igw-id</i>
::/0	<i>igw-id</i>

보안

이 예시 구성에서는 애플리케이션에 필요한 트래픽을 허용하는 인스턴스에 대한 보안 그룹을 생성해야 합니다. 예를 들어 컴퓨터의 SSH 트래픽이나 네트워크의 HTTP 트래픽을 허용하는 규칙을 추가해야 할 수 있습니다.

다음은 IPv4와 IPv6 모두에 대한 규칙이 포함된 보안 그룹의 인바운드 규칙의 예시입니다. 이중 스택 서브넷 대신 IPv4 전용 서브넷을 생성하는 경우 IPv4에 대한 규칙만 필요합니다.

인바운드

소스	프로토콜	포트 범위	설명
0.0.0.0/0	TCP	80	모든 IPv4 주소에서 이루어지는 인바운드 HTTP 액세스 허용
::/0	TCP	80	모든 IPv6 주소에서 이루어지는 인바운드 HTTP 액세스 허용
0.0.0.0/0	TCP	443	모든 IPv4 주소에서 이루어지는 인바운드 HTTPS 액세스 허용
::/0	TCP	443	모든 IPv6 주소에서 이루어지는 인바운드 HTTPS 액세스 허용
##### ### IPv4 ## ##	TCP	22	(선택 사항) 네트워크에서 IPv4 IP 주소의 인바운드 SSH 액세스 허용
##### IPv6 ## ##	TCP	22	(선택 사항) 네트워크에서 IPv6 IP 주소의 인바운드 SSH 액세스 허용
##### ### IPv4 ## ##	TCP	3389	(선택 사항) 네트워크에서 IPv4 IP 주소의 인바운드 RDP 액세스 허용
##### IPv6 ## ##	TCP	3389	(선택 사항) 네트워크에서 IPv6 IP 주소의 인바운드 RDP 액세스 허용

VPC 생성

다음 절차에 따라 1개의 가용 영역에 퍼블릭 서브넷이 있는 VPC를 생성하세요. 이 구성은 개발 또는 테스트 환경에 적합합니다.

VPC를 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 대시보드에서 VPC 생성을 선택합니다.
3. 생성할 리소스에서 VPC 등을 선택합니다.
4. VPC 구성
 - a. Name tag auto-generation(이름 태그 자동 생성)에 VPC의 이름을 입력합니다.
 - b. IPv4 CIDR 블록에 애플리케이션 또는 네트워크에 필요한 CIDR 블록을 입력하거나 기본 사항을 유지할 수 있습니다. 자세한 내용은 [the section called “VPC CIDR 블록”](#) 단원을 참조하십시오.
 - c. (선택 사항) 애플리케이션이 IPv6 주소를 사용하여 통신하는 경우 IPv6 CIDR 블록, 즉 Amazon에서 제공한 IPv6 CIDR 블록을 선택합니다.
5. 서브넷 구성
 - a. 가용 영역 수에서 1을 선택합니다. 기본 가용 영역을 유지하거나, AZ 사용자 지정을 확장하고 가용 영역을 선택할 수 있습니다.
 - b. 퍼블릭 서브넷 수는 1을 선택합니다.
 - c. 프라이빗 서브넷 수는 0을 선택합니다.
 - d. 퍼블릭 서브넷에 대한 기본 CIDR 블록을 유지하거나 서브넷 CIDR 블록 사용자 지정을 확장하고 CIDR 블록을 입력할 수 있습니다. 자세한 내용은 [the section called “서브넷 CIDR 블록”](#) 단원을 참조하십시오.
6. NAT 게이트웨이에서 기본값인 없음을 유지합니다.
7. VPC 엔드포인트는 없음을 선택합니다. S3용 게이트웨이 VPC 엔드포인트는 프라이빗 서브넷에서 Amazon S3에 액세스하는 데만 사용됩니다.
8. DNS 옵션에서 두 옵션을 모두 선택된 상태로 둡니다. 결과적으로 인스턴스는 퍼블릭 IP 주소에 해당하는 퍼블릭 DNS 호스트 이름을 수신하게 됩니다.
9. VPC 생성을 선택합니다.

애플리케이션 배포

EC2 인스턴스를 배포하는 다양한 방법이 있습니다. 예:

- [Amazon EC2 인스턴스 시작 마법사](#)
- [Amazon EC2 Auto Scaling](#)
- [AWS CloudFormation](#)
- [Amazon Elastic Container Service\(Amazon ECS\)](#)

EC2 인스턴스를 배포한 후 인스턴스에 연결하고, 애플리케이션에 필요한 소프트웨어를 설치한 다음, 나중에 사용할 이미지를 생성할 수 있습니다. 자세한 내용은 Amazon EC2 설명서의 [Linux AMI 생성](#) 또는 [Windows AMI 생성](#)을 참조하세요. 또는 [EC2 Image Builder](#)를 사용하여 Amazon Machine Image(AMI)를 생성하고 관리할 수 있습니다.

구성 테스트

애플리케이션 배포를 완료한 후 테스트할 수 있습니다. EC2 인스턴스에 연결하거나 애플리케이션이 예상한 트래픽을 전송하거나 수신할 수 없는 경우 Reachability Analyzer를 사용하여 문제를 해결할 수 있습니다. 예를 들어, Reachability Analyzer는 라우팅 테이블 또는 보안 그룹의 구성 문제를 식별할 수 있습니다. 자세한 내용은 [Reachability Analyzer 사용 설명서](#)를 참조하십시오.

정리

이 구성을 마치면 이를 삭제할 수 있습니다. VPC를 삭제하려면 먼저 인스턴스를 종료해야 합니다. 자세한 내용은 [the section called “VPC 삭제”](#) 단원을 참조하십시오.

예시: 웹 및 데이터베이스 서버용 VPC

이 예시에서는 프로덕션 환경에서 2계층 아키텍처에 사용할 수 있는 VPC를 생성하는 방법을 보여줍니다. 복원력 향상을 위해 2개의 가용 영역에 서버를 배포합니다.

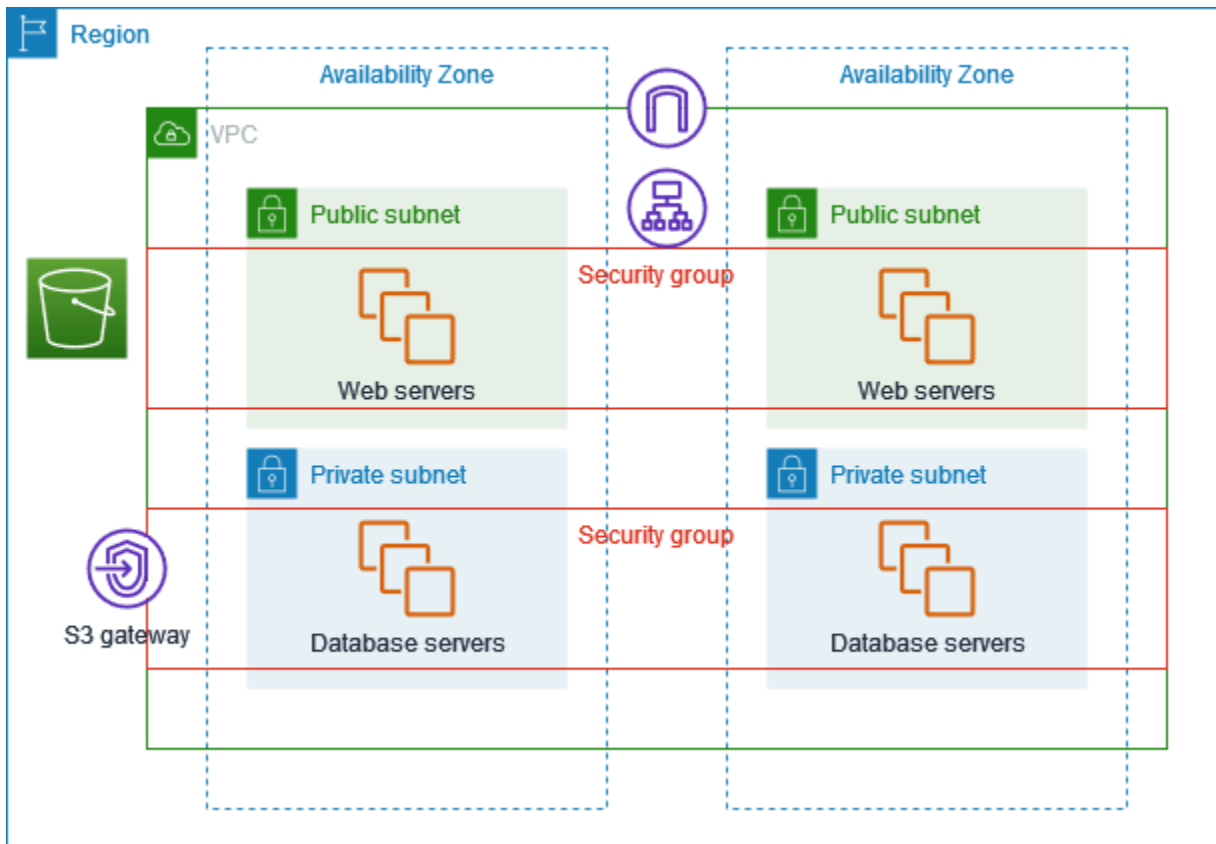
내용

- [개요](#)
- [VPC 생성](#)
- [애플리케이션 배포](#)
- [구성 테스트](#)

• 정리

개요

다음 다이어그램에서는 이 예시에 포함된 리소스의 개요를 제공합니다. VPC에는 2개의 가용 영역에 퍼블릭 서브넷과 프라이빗 서브넷이 있습니다. 웹 서버는 퍼블릭 서브넷에서 실행되며 로드 밸런서를 통해 클라이언트로부터 트래픽을 수신합니다. 웹 서버의 보안 그룹은 로드 밸런서의 트래픽을 허용합니다. 데이터베이스 서버는 프라이빗 서브넷에서 실행되며 웹 서버로부터 트래픽을 수신합니다. 데이터베이스 서버의 보안 그룹은 웹 서버의 트래픽을 허용합니다. 데이터베이스 서버는 게이트웨이 VPC 엔드포인트를 사용하여 Amazon S3에 연결할 수 있습니다.



라우팅

Amazon VPC 콘솔을 사용하여 이 VPC를 생성하면 로컬 경로와 인터넷 게이트웨이에 대한 경로가 있는 퍼블릭 서브넷의 라우팅 테이블과 로컬 경로 및 게이트웨이 VPC 엔드포인트에 대한 경로가 있는 각 프라이빗 서브넷의 경로 테이블이 생성됩니다.

다음은 IPv4와 IPv6 모두에 대한 경로가 있는 퍼블릭 서브넷의 라우팅 테이블 예시입니다. 이중 스택 서브넷 대신 IPv4 전용 서브넷을 생성하는 경우 라우팅 테이블에는 IPv4 경로만 있습니다.

대상 주소	대상
<i>10.0.0.0/16</i>	로컬
<i>2001:db8:1234:1a00::/56</i>	로컬
0.0.0.0/0	<i>igw-id</i>
::/0	<i>igw-id</i>

다음은 IPv4와 IPv6 모두에 대한 로컬 경로가 있는 프라이빗 서브넷의 라우팅 테이블 예시입니다. IPv4 전용 서브넷을 생성한 경우 라우팅 테이블에는 IPv4 경로만 있습니다. 마지막 경로는 Amazon S3로 향하는 트래픽을 게이트웨이 VPC 엔드포인트로 전송합니다.

대상 주소	대상
<i>10.0.0.0/16</i>	로컬
<i>2001:db8:1234:1a00::/56</i>	로컬
<i>s3-prefix-list-id</i>	<i>s3-gateway-id</i>

보안

이 예시 구성에서는 로드 밸런서용 보안 그룹, 웹 서버용 보안 그룹 및 데이터베이스 서버용 보안 그룹을 생성합니다.

로드 밸런서

Application Load Balancer 또는 Network Load Balancer의 보안 그룹에서는 로드 밸런서 리스너 포트의 클라이언트에서 제공되는 인바운드 트래픽이 허용되어야 합니다. 인터넷상의 위치와 관계 없이 모든 트래픽을 수락하려면 소스를 0.0.0.0/0으로 지정하세요. 로드 밸런서 보안 그룹에서는 로드 밸런서에서 출발하여 인스턴스 리스너 포트 및 상태 확인 포트의 대상 인스턴스로 향하는 아웃바운드 트래픽도 허용되어야 합니다.

웹 서버

다음 보안 그룹 규칙을 사용하면 웹 서버가 로드 밸런서로부터 HTTP 및 HTTPS 트래픽을 수신할 수 있습니다. 웹 서버가 네트워크에서 SSH 또는 RDP 트래픽을 수신하도록 허용할 수도 있습니다. 웹 서버는 데이터베이스 서버에 SQL 또는 MySQL 트래픽을 전송할 수 있습니다.

인바운드

소스	프로토콜	포트 범위	설명
<i>## ### ## ### ID</i>	TCP	80	로드 밸런서에서 접근하는 인바운드 HTTP 액세스를 허용
<i>## ### ## ### ID</i>	TCP	443	로드 밸런서에서 접근하는 인바운드 HTTPS 액세스를 허용
<i>##### ### IPv4 ## ##</i>	TCP	22	(선택 사항) 네트워크에서 IPv4 IP 주소의 인바운드 SSH 액세스 허용
<i>##### IPv6 ## ##</i>	TCP	22	(선택 사항) 네트워크에서 IPv6 IP 주소의 인바운드 SSH 액세스 허용
<i>##### ### IPv4 ## ##</i>	TCP	3389	(선택 사항) 네트워크에서 IPv4 IP 주소의 인바운드 RDP 액세스 허용
<i>##### IPv6 ## ##</i>	TCP	3389	(선택 사항) 네트워크에서 IPv6 IP 주소의 인바운드 RDP 액세스 허용

아웃바운드

대상 주소	프로토콜	포트 범위	설명
<i>Microsoft SQL Server# ##### ## ## ID</i>	TCP	1433	데이터베이스 서버에 대한 아웃바운드 Microsoft SQL Server 액세스 허용

대상 주소	프로토콜	포트 범위	설명
<i>MySQL# ##### ## ## ID</i>	TCP	3306	데이터베이스 서버에 대한 아웃바운드 MySQL 액세스 허용

데이터베이스 서버

다음 보안 그룹 규칙은 데이터베이스 서버가 웹 서버로부터 읽기 및 쓰기 요청을 수신하도록 허용합니다.

인바운드

소스	프로토콜	포트 범위	설명
<i># ## ## ## ID</i>	TCP	1433	웹 서버의 인바운드 Microsoft SQL Server 액세스 허용
<i># ## ## ## ID</i>	TCP	3306	웹 서버의 인바운드 MySQL Server 액세스 허용

아웃바운드

대상 주소	프로토콜	포트 범위	설명
0.0.0.0/0	TCP	80	IPv4를 통해 인터넷에 대한 아웃바운드 HTTP 액세스 허용
0.0.0.0/0	TCP	443	IPv4를 통해 인터넷에 대한 아웃바운드 HTTPS 액세스 허용

Amazon RDS DB 인스턴스의 보안 그룹에 대한 자세한 내용은 Amazon RDS 사용 설명서의 [보안 그룹을 통한 액세스 제어](#)를 참조하세요.

VPC 생성

다음 절차에 따라 2개의 가용 영역에 퍼블릭 서브넷과 프라이빗 서브넷이 있는 VPC를 생성하세요.

VPC를 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 대시보드에서 VPC 생성을 선택합니다.
3. 생성할 리소스에서 VPC 등을 선택합니다.
4. VPC 구성:
 - a. 이름 태그 자동 생성을 선택한 상태로 유지하여 VPC 리소스에 이름 태그를 생성하거나 선택을 취소하여 VPC 리소스에 고유한 이름 태그를 제공합니다.
 - b. IPv4 CIDR 블록에 애플리케이션 또는 네트워크에 필요한 CIDR 블록을 입력하거나 기본 사항을 유지할 수 있습니다. 자세한 내용은 [the section called “VPC CIDR 블록”](#) 단원을 참조하십시오.
 - c. (선택 사항) 애플리케이션이 IPv6 주소를 사용하여 통신하는 경우 IPv6 CIDR 블록, 즉 Amazon에서 제공한 IPv6 CIDR 블록을 선택합니다.
 - d. 테넌시 옵션을 선택합니다. 이 옵션은 VPC로 시작하는 EC2 인스턴스가 다른 AWS 계정과 공유되는 하드웨어에서 실행되는지 아니면 사용자 전용 하드웨어에서 실행되는지를 정의합니다. VPC의 테넌시를 Default로 선택하면 이 VPC로 시작된 EC2 인스턴스에서는 인스턴스를 시작할 때 지정된 테넌시 속성을 사용합니다. 자세한 내용은 Amazon EC2 사용 설명서의 [정의된 파라미터를 사용하여 인스턴스 시작](#)을 참조하세요. VPC의 테넌시를 Dedicated로 선택하면 인스턴스는 항상 전용 하드웨어에서 [전용 인스턴스](#)로 실행됩니다.
5. 서브넷 구성:
 - a. 복원 향상을 위해 2개의 가용 영역에서 인스턴스를 시작할 수 있도록 가용 영역 수에서 2를 선택합니다.
 - b. 퍼블릭 서브넷 수는 2를 선택합니다.
 - c. 프라이빗 서브넷 수는 2를 선택합니다.
 - d. 서브넷에 대한 기본 CIDR 블록을 유지하거나 서브넷 CIDR 블록 사용자 지정을 확장하고 CIDR 블록을 입력할 수 있습니다. 자세한 내용은 [the section called “서브넷 CIDR 블록”](#) 단원을 참조하십시오.
6. NAT 게이트웨이에서 기본값인 없음을 유지합니다.
7. VPC 엔드포인트에서 기본값인 S3 게이트웨이를 유지합니다. S3 버킷에 액세스하지 않으면 효과가 없지만 이 VPC 엔드포인트를 활성화하는 데 비용이 들지 않습니다.
8. DNS 옵션에서 두 옵션을 모두 선택된 상태로 둡니다. 결과적으로 웹 서버는 퍼블릭 IP 주소에 해당하는 퍼블릭 DNS 호스트 이름을 수신하게 됩니다.
9. VPC 생성을 선택합니다.

애플리케이션 배포

개발 또는 테스트 환경에서 웹 서버와 데이터베이스 서버 테스트를 완료하고 프로덕션 환경에서 애플리케이션을 배포하는 데 사용할 스크립트 또는 이미지를 생성하는 것이 가장 좋습니다.

웹 서버에 EC2 인스턴스를 사용할 수 있습니다. EC2 인스턴스를 배포하는 다양한 방법이 있습니다. 예:

- [Amazon EC2 인스턴스 시작 마법사](#)
- [AWS CloudFormation](#)
- [Amazon Elastic Container Service\(Amazon ECS\)](#)

가용성 향상을 위해 [Amazon EC2 Auto Scaling](#)을 사용하여 여러 가용 영역에 서버를 배포하고 애플리케이션에 필요한 최소 서버 용량을 유지할 수 있습니다.

[Elastic Load Balancing](#)을 사용하여 서버 전체에 트래픽을 균일하게 배포할 수 있습니다. Auto Scaling에 로드 밸런서를 연결할 수 있습니다.

데이터베이스 서버용 EC2 인스턴스를 사용하거나 목적별 데이터베이스 유형 중 하나를 사용할 수 있습니다. 자세한 내용은 [AWS의 데이터베이스: 선택 방법](#)을 참조하세요.

구성 테스트

애플리케이션 배포를 완료한 후 테스트할 수 있습니다. 애플리케이션이 예상한 트래픽을 전송하거나 수신할 수 없는 경우 Reachability Analyzer를 사용하여 문제를 해결할 수 있습니다. 예를 들어, Reachability Analyzer는 라우팅 테이블 또는 보안 그룹의 구성 문제를 식별할 수 있습니다. 자세한 내용은 [Reachability Analyzer 사용 설명서](#)를 참조하십시오.

정리

이 구성을 마치면 이를 삭제할 수 있습니다. VPC를 삭제하려면 먼저 인스턴스를 종료하고 로드 밸런서를 삭제해야 합니다. 자세한 내용은 [the section called "VPC 삭제"](#) 단원을 참조하십시오.

예시: 프라이빗 서브넷과 NAT에 서버가 있는 VPC

이 예시에서는 프로덕션 환경에서 서버에 사용할 수 있는 VPC를 생성하는 방법을 보여줍니다. 복원력 향상을 위해 Auto Scaling과 Application Load Balancer를 사용하여 2개의 가용 영역에 서버를 배포합니다. 보안 강화를 위해 프라이빗 서브넷에 서버를 배포합니다. 서버는 로드 밸런서를 통해 요청을 수

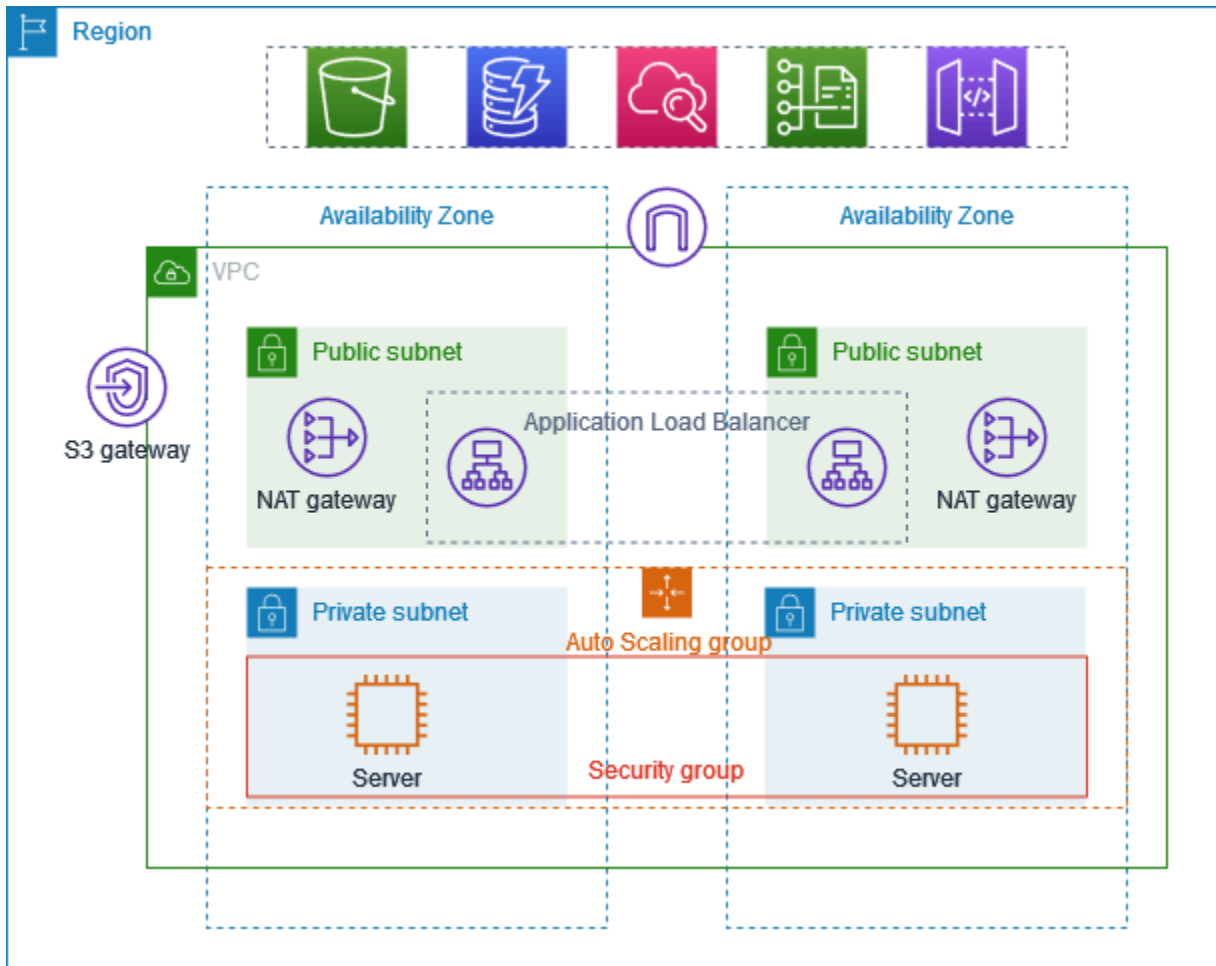
신합니다. 서버는 NAT 게이트웨이를 사용하여 인터넷에 연결할 수 있습니다. 복원력 향상을 위해 두 가용 영역 모두에 NAT 게이트웨이를 배포합니다.

내용

- [개요](#)
- [VPC 생성](#)
- [애플리케이션 배포](#)
- [구성 테스트](#)
- [정리](#)

개요

다음 다이어그램에서는 이 예시에 포함된 리소스의 개요를 제공합니다. VPC에는 2개의 가용 영역에 퍼블릭 서브넷과 프라이빗 서브넷이 있습니다. 각 퍼블릭 서브넷에는 NAT 게이트웨이와 로드 밸런서 노드가 있습니다. 서버는 프라이빗 서브넷에서 실행되고, Auto Scaling을 사용하여 시작 및 종료되고, 로드 밸런서에서 트래픽을 수신합니다. 서버는 NAT 게이트웨이를 사용하여 인터넷에 연결할 수 있습니다. 서버는 게이트웨이 VPC 엔드포인트를 사용하여 Amazon S3에 연결할 수 있습니다.



라우팅

Amazon VPC 콘솔을 사용하여 이 VPC를 생성하면 로컬 경로와 인터넷 게이트웨이에 대한 경로가 있는 퍼블릭 서브넷에 대한 라우팅 테이블이 생성됩니다. 또한 로컬 경로와 NAT 게이트웨이, 송신 전용 인터넷 게이트웨이 및 게이트웨이 VPC 엔드포인트에 대한 경로가 있는 프라이빗 서브넷의 라우팅 테이블이 생성됩니다.

다음은 IPv4와 IPv6 모두에 대한 경로가 있는 퍼블릭 서브넷의 라우팅 테이블 예시입니다. 이중 스택 서브넷 대신 IPv4 전용 서브넷을 생성하는 경우 라우팅 테이블에는 IPv4 경로만 포함됩니다.

대상 주소	대상
<i>10.0.0.0/16</i>	로컬
<i>2001:db8:1234:1a00::/56</i>	로컬
0.0.0.0/0	<i>igw-id</i>

대상 주소	대상
::/0	<i>igw-id</i>

다음은 IPv4와 IPv6 모두에 대한 경로가 있는 프라이빗 서브넷 중 하나의 라우팅 테이블 예시입니다. IPv4 전용 서브넷을 생성한 경우 라우팅 테이블에는 IPv4 경로만 포함됩니다. 마지막 경로는 Amazon S3로 향하는 트래픽을 게이트웨이 VPC 엔드포인트로 전송합니다.

대상 주소	대상
<i>10.0.0.0/16</i>	로컬
<i>2001:db8:1234:1a00::/56</i>	로컬
0.0.0.0/0	<i>nat-gateway-id</i>
::/0	<i>eigw-id</i>
<i>s3-prefix-list-id</i>	<i>s3-gateway-id</i>

보안

다음은 서버와 연결하는 보안 그룹에 대해 생성할 수 있는 규칙의 예시입니다. 보안 그룹은 리스너 포트 및 프로토콜을 통한 로드 밸런서의 트래픽을 허용해야 합니다. 또한 상태 확인 트래픽을 허용해야 합니다.

인바운드

소스	프로토콜	포트 범위	설명
<i>## ### ## ### ID</i>	<i>### #####</i>	<i>### ##</i>	리스너 포트에서 로드 밸런서의 인바운드 트래픽 허용
<i>## ### ## ### ID</i>	<i>## ## #####</i>	<i>## ## ##</i>	로드 밸런서의 인바운드 상태 확인 트래픽 허용

VPC 생성

다음 절차에 따라 2개의 가용 영역에 퍼블릭 서브넷과 프라이빗 서브넷이 있고 각 가용 영역에 NAT 게이트웨이가 있는 VPC를 생성하세요.

VPC를 생성하려면

1. <https://console.aws.amazon.com/vpc/>에서 Amazon VPC 콘솔을 엽니다.
2. 대시보드에서 VPC 생성을 선택합니다.
3. 생성할 리소스에서 VPC 등을 선택합니다.
4. VPC 구성
 - a. Name tag auto-generation(이름 태그 자동 생성)에 VPC의 이름을 입력합니다.
 - b. IPv4 CIDR 블록에 애플리케이션 또는 네트워크에 필요한 CIDR 블록을 입력하거나 기본 사항을 유지할 수 있습니다.
 - c. 애플리케이션이 IPv6 주소를 사용하여 통신하는 경우 IPv6 CIDR 블록, 즉 Amazon에서 제공한 IPv6 CIDR 블록을 선택합니다.
5. 서브넷 구성
 - a. 복원 향상을 위해 여러 가용 영역에서 인스턴스를 시작할 수 있도록 가용 영역 수에서 2를 선택합니다.
 - b. 퍼블릭 서브넷 수는 2를 선택합니다.
 - c. 프라이빗 서브넷 수는 2를 선택합니다.
 - d. 퍼블릭 서브넷에 대한 기본 CIDR 블록을 유지하거나 서브넷 CIDR 블록 사용자 지정을 확장하고 CIDR 블록을 입력할 수 있습니다. 자세한 내용은 [the section called “서브넷 CIDR 블록”](#) 단원을 참조하십시오.
6. NAT 게이트웨이에서 복원력 향상을 위해 AZ당 1개를 선택합니다.
7. 애플리케이션이 IPv6 주소를 사용하여 통신하는 경우 송신 전용 인터넷 게이트웨이에서 예를 선택합니다.
8. VPC 엔드포인트에서 인스턴스가 S3 버킷에 액세스해야 하는 경우 S3 게이트웨이를 기본값으로 유지합니다. 그렇지 않으면 프라이빗 서브넷의 인스턴스가 Amazon S3에 액세스할 수 없습니다. 이 옵션에는 비용이 들지 않으므로 나중에 S3 버킷을 사용할 경우 기본값을 유지할 수 있습니다. 없음을 선택하면 나중에 언제든지 게이트웨이 VPC 엔드포인트를 추가할 수 있습니다.
9. DNS 옵션에서 DNS 호스트 이름 활성화를 선택 취소합니다.
10. VPC 생성을 선택합니다.

애플리케이션 배포

개발 또는 테스트 환경에서 서버 테스트를 완료하고 프로덕션 환경에서 애플리케이션을 배포하는 데 사용할 스크립트 또는 이미지를 생성하는 것이 가장 좋습니다.

[Amazon EC2 Auto Scaling](#)을 사용하여 여러 가용 영역에 서버를 배포하고 애플리케이션에 필요한 최소 서버 용량을 유지할 수 있습니다.

Auto Scaling을 사용하여 인스턴스 시작

1. 시작 템플릿을 생성하여 Amazon EC2 Auto Scaling으로 EC2 인스턴스를 시작하는 데 필요한 구성 정보를 지정합니다. 단계별 지침은 Amazon EC2 Auto Scaling 사용 설명서의 [Auto Scaling을 위한 시작 템플릿 생성](#)을 참조하세요.
2. EC2 인스턴스 모음인 Auto Scaling을 최소, 최대 및 원하는 크기로 생성합니다. 단계별 지침은 Amazon EC2 Auto Scaling 사용 설명서의 [시작 템플릿을 사용하여 Auto Scaling 생성](#)을 참조하세요.
3. Auto Scaling의 인스턴스 간에 트래픽을 고르게 배포하는 로드 밸런서를 생성하고 Auto Scaling에 로드 밸런서를 연결합니다. 자세한 내용은 Elastic Load Balancing 사용 설명서 <https://docs.aws.amazon.com/elasticloadbalancing/latest/userguide/>와 Amazon EC2 Auto Scaling 사용 설명서의 [Elastic Load Balancing 사용](#)을 참조하세요.

구성 테스트

애플리케이션 배포를 완료한 후 테스트할 수 있습니다. 애플리케이션이 예상한 트래픽을 전송하거나 수신할 수 없는 경우 Reachability Analyzer를 사용하여 문제를 해결할 수 있습니다. 예를 들어, Reachability Analyzer는 라우팅 테이블 또는 보안 그룹의 구성 문제를 식별할 수 있습니다. 자세한 내용은 [Reachability Analyzer 사용 설명서](#)를 참조하십시오.

정리

이 구성을 마치면 이를 삭제할 수 있습니다. VPC를 삭제하려면 먼저 Auto Scaling을 삭제하고, 인스턴스를 종료하고, NAT 게이트웨이를 삭제하고, 로드 밸런서를 삭제해야 합니다. 자세한 내용은 [the section called "VPC 삭제"](#) 단원을 참조하십시오.

Amazon VPC 할당량

다음 표에는 AWS 계정의 Amazon VPC 리소스 할당량(이전 명칭은 '제한')이 나열되어 있습니다. 달리 표시되지 않는 한 이러한 할당량은 리전당 할당량입니다.

리소스별로 적용되는 할당량 증가를 요청하는 경우 리전에 있는 모든 리소스의 할당량이 증가합니다.

VPC 및 서브넷

명칭	기본값	조정 가능	설명
리전당 VPC	5	예	이 할당량을 늘리면 리전당 인터넷 게이트웨이에 대한 할당량도 같은 수량만큼 늘어 납니다. 리전당 수백 개의 VPC를 사용할 수 있도록 이 제한을 늘릴 수 있습니다.
VPC당 서브넷	200	예	
VPC당 IPv4 CIDR 블록	5	예 (최대 50 개)	이 기본 CIDR 블록 및 모든 보조 CIDR 블록은 이 할당량에 포함됩니다.
VPC당 IPv6 CIDR 블록	5	예 (최대 50 개)	단일 VPC에 할당할 수 있는 CIDR의 수입니다.

DNS

각 EC2 인스턴스는 Route 53 Resolver로(구체적으로 10.0.0.2 및 169.254.169.253과 같은 .2 주소) 네트워크 인터페이스별 초당 1024개의 패킷을 보낼 수 있습니다. 이 할당량은 늘릴 수 없습니다. Route 53 Resolver가 지원하는 초당 DNS 쿼리 수는 쿼리 유형, 응답 크기 및 사용 중인 프로토콜에 따라 다릅니다. 확장 가능한 DNS 아키텍처에 대한 자세한 내용과 권장 사항은 [Active Directory 포함 AWS 하이브리드 DNS 기술 가이드](#)를 확인하십시오.

탄력적 IP 주소

명칭	기본값	조정 가능	설명
각 리전의 탄력적 IP 주소	5	예	이 할당량은 개별 AWS 계정 VPC 및 공유 VPC에 적용됩니다.
퍼블릭 NAT 게이트웨이당 탄력적 IP 주소	2	예	8개까지 할당량 증가를 요청할 수 있습니다.

게이트웨이

명칭	기본값	조정 가능	설명
리전당 외부 전용 인터넷 게이트웨이	5	예	이 할당량을 늘리려면 리전당 VPC 할당량을 늘리세요. 하나의 VPC에는 한 번에 한 개의 외부 전용 인터넷 게이트웨이만 연결할 수 있습니다.
리전당 인터넷 게이트웨이	5	예	이 할당량을 늘리려면 리전당 VPC 할당량을 늘리세요. 한 번에 하나의 인터넷 게이트웨이만 하나의 VPC에 연결할 수 있습니다.
가용 영역당 NAT 게이트웨이	5	예	NAT 게이트웨이는 pending, active, deleting 상태의 할당량에 포함됩니다.
NAT 게이트웨이당 프라이빗 IP 주소 할당량	8	아니요	
VPC당 통신 사업자 게이트웨이	1	아니요	

고객 관리형 접두사 목록

고객 관리형 접두사 목록의 기본 할당량은 조정 가능하지만 Service Quotas 콘솔을 사용하여 증가를 요청할 수는 없습니다. AWS Support Center Console을 사용하여 [서비스 한도 증가 사례를 열어야](#) 합니다.

명칭	기본값	조정 가능	설명
리전별 접두사 목록	100	예	
접두사 목록당 버전	1,000	예	접두사 목록에 1,000개의 저장된 버전이 있고 새 버전을 추가하는 경우 새 버전이 추가되도록 가장 오래된 버전이 제거됩니다.
접두사 목록당 최대 항목 수	1,000	예	고객 관리형 접두사 목록의 크기를 최대 1000개까지 조정할 수 있습니다. 자세한 내용은 접두사 목록 크기 조정 단원을 참조하십시오. 리소스의 접두사 목록을 참조할 때 접두사 목록의 최대 항목 수는 리소스의 항목 수에 대한 할당량에 따라 계산됩니다. 예를 들어, 최대 항목이 20개인 접두사 목록을 만들고 보안 그룹 규칙에서 해당 접두사 목록을 참조하는 경우, 20개의 보안 그룹 규칙이 있는 것으로 계산됩니다.
리소스 유형별 접두사 목록 참조	5,000	예	이 할당량은 접두사 목록을 참조할 수 있는 리소스 유형별로 적용됩니다. 예를 들어, 모든 보안 그룹에서 접두사 목록에 대한 참조 5,000개와 모든 서브넷 라우팅 테이블에서 접두사 목록에 대한 참조 5,000개를 가질 수 있습니다. 접두사 목록을 다른 AWS 계정과 공유하는 경우, 접두사 목록에 대한 다른 계정의 참조가 이 할당량에 포함됩니다.

네트워크 ACL

명칭	기본값	조정 가능	설명
VPC당 네트워크 ACL	200	예	하나의 VPC에 있는 한 개 이상의 서브넷에 한 개의 네트워크 ACL을 연결할 수 있습니다.
네트워크 ACL당 규칙	20	예	이 할당량은 최대 인바운드 규칙 수와 최대 아웃바운드 규칙 수를 결정합니다. 이 할당량은 최대 40개의 인바운드 규칙과 40개의 아웃바운드 규칙(총 80개의 규칙)까지 늘릴 수 있지만 네트워크 성능에 영향을 미칠 수 있습니다.

네트워크 인터페이스

명칭	기본값	조정 가능	설명
인스턴스당 네트워크 인터페이스	인스턴스 유형에 따라 다릅니다.	아니요	자세한 내용은 인스턴스 유형당 네트워크 인터페이스 를 참조하세요.
리전당 네트워크 인터페이스	5,000	예	이 할당량은 개별 AWS 계정 VPC 및 공유 VPC에 적용됩니다. 이 제한은 가용 영역별로 적용됩니다. 예를 들어, 네트워크 인터페이스가 3개의 AZ에 있는 경우 각 AZ의 제한은 5,000이고 리전의 제한은 15,000입니다.

라우팅 테이블

명칭	기본값	조정 가능	설명
VPC당 라우팅 테이블	200	예	기본 라우팅 테이블은 이 할당량에 포함됩니다. 라우팅 테이블에 대한 할당량 증가를 요청하는 경우 서브넷에 대한 할당량 증가를 요청할 수도 있습니다. 라우팅 테이블들을 여러 개의 서브넷과 공유할 수 있지만, 한 서브넷을 한 번에 단 하나의 라우팅 테이블과 연결할 수 있습니다.
라우팅 테이블에 따른 경로 (전파되지 않는 경로)	50	예	이 할당량은 최대 1,000개까지 늘릴 수 있지만 네트워크 성능에 영향을 줄 수 있습니다. 이 할당량은 IPv4 및 IPv6 경로에 개별적으로 적용됩니다. 경로가 125개 이상 있다면 성능 향상을 위해 호출에 페이지 번호를 붙여 라우팅 테이블을 설명하는 것이 좋습니다.
라우팅 테이블에 따른 전파되는 경로	100	아니요	추가 접두사가 필요할 경우 기본 경로를 알려주세요.

보안 그룹

명칭	기본값	조정 가능	설명
리전당 VPC 보안 그룹	2,500	예	이 할당량은 개별 AWS 계정 VPC 및 공유 VPC에 적용됩니다. 리전 하나에 할당량을 보안 그룹 5,000개 이상으로 늘리는 경우, 성능 향상을 위해 호출에 페이지 매김을 하여 보안 그룹을 설명하는 것이 좋습니다.

명칭	기본값	조정 가능	설명
보안 그룹별 인바운드 또는 아웃바운드 규칙	60	예	<p>이 할당량은 인바운드 규칙과 아웃바운드 규칙에 개별적으로 적용됩니다. 규칙 60개의 기본 할당량을 가진 계정의 경우 보안 그룹은 인바운드 규칙 60개와 아웃바운드 규칙 60개를 가질 수 있습니다. 또한 이 할당량은 IPv4 및 IPv6 규칙에 개별적으로 적용됩니다. 규칙 60개의 기본 할당량을 가진 계정의 경우 보안 그룹은 IPv4 트래픽에 대한 인바운드 규칙 60개와 IPv6 트래픽에 대한 인바운드 규칙 60개를 가질 수 있습니다. 자세한 내용은 the section called “보안 그룹 크기” 단원을 참조하십시오.</p> <p>할당량 변경은 인바운드 규칙과 아웃바운드 규칙에 모두 적용됩니다. 이 할당량과 네트워크 인터페이스당 보안 그룹의 할당량을 곱한 값이 1,000을 초과하면 안 됩니다.</p>
네트워크 인터페이스당 보안 그룹	5	예 (최대 16개)	이 할당량과 보안 그룹당 규칙 할당량을 곱한 값이 1,000을 초과하면 안 됩니다.

VPC 공유

모든 표준 VPC 할당량은 공유된 VPC에 적용됩니다.

명칭	기본값	조정 가능	설명
VPC당 참가자 계정	100	예	VPC의 서브넷을 공유할 수 있는 참가자 계정의 최대 수입니다. 이는 VPC당 할당량이며, VPC에서 공유되는 모든 서브넷에 적용됩니다.

명칭	기본값	조정 가능	설명
			VPC 소유자는 참가자 리소스에 연결된 네트워크 인터페이스 및 보안 그룹을 볼 수 있습니다.
계정과 공유할 수 있는 서브넷	100	예	AWS 계정과 공유할 수 있는 최대 서브넷 수입니다.

네트워크 주소 사용량

NAU(네트워크 주소 사용량)는 IP 주소, 네트워크 인터페이스 및 관리형 접두사 목록의 CIDR로 구성됩니다. NAU는 VPC 크기를 계획하고 모니터링하는 데 도움이 되도록 VPC의 리소스에 적용되는 지표입니다. 자세한 내용은 [네트워크 주소 사용량](#) 단원을 참조하십시오.

NAU 개수를 구성하는 리소스에는 자체적인 개별 서비스 할당량이 있습니다. VPC에 사용 가능한 NAU 용량이 있더라도 리소스에서 해당 서비스 할당량을 초과하면 리소스를 VPC에서 시작할 수 없습니다.

명칭	기본값	조정 가능	설명
네트워크 주소 사용량	64,000	예 (최대 256,000 개)	VPC당 최대 NAU 단위 수입니다.
피어링된 네트워크 주소 사용량	128,000	예 (최대 512,000 개)	VPC 및 해당 VPC와 리전 내 피어링된 모든 VPC의 최대 NAU 단위 수입니다. 다른 리전 간에 피어링된 VPC는 이 수에 포함되지 않습니다.

Amazon EC2 API 조절

Amazon EC2 조절에 대한 자세한 내용은 Amazon EC2 API 참조의 [API 요청 조절](#)을 참조하십시오.

추가 할당량 리소스

자세한 내용은 다음 자료를 참조하세요.

- AWS Client VPN 관리자 가이드의 [AWS Client VPN 할당량](#)
- AWS Direct Connect 사용 설명서의 [AWS Direct Connect 할당량](#)
- Amazon VPC 피어링 설명서의 [피어링 할당량](#)
- AWS PrivateLink 설명서의 [PrivateLink 할당량](#)
- AWS Site-to-Site VPN 사용 설명서의 [Site-to-Site VPN 할당량](#)
- Amazon VPC Traffic Mirroring 설명서의 [Traffic Mirroring 할당량](#)
- Amazon VPC Transit Gateway 설명서의 [Transit Gateway 할당량](#)

문서 기록

다음 표에서는 Amazon VPC 사용 설명서의 각 릴리스에서 변경된 중요 사항에 관해 설명합니다.

변경 사항	설명	날짜
IPv6 선호 임대 시간	이제는 IPv6가 할당되어 실행 중인 인스턴스의 DHCPv6 임대 갱신 빈도를 선택할 수 있습니다.	2024년 2월 20일
AWS 관리형 정책 업데이트	Amazon VPC의 AmazonVPC FullAccess 및 AmazonVPC ReadOnlyAccess 관리형 정책이 업데이트되었습니다.	2024년 2월 8일
AWS 관리형 정책 업데이트	Amazon VPC의 AmazonVPC CrossAccountNetworkInterfaceOperations 관리형 정책이 업데이트되었습니다.	2023년 9월 25일
EC2-Classic 더 이상 사용되지 않음	EC2-Classic을 사용하여 EC2 인스턴스는 다른 고객과 공유되는 단일 플랫폼 네트워크에서 실행되었습니다. Amazon VPC가 EC2-Classic을 대체합니다. Amazon VPC 사용을 통해 AWS 계정에 속하도록 논리적으로 독립된 Virtual Private Cloud(VPC)에서 인스턴스가 실행됩니다.	2023년 7월 31일
NAT 게이트웨이에 보조 IPv4 주소 추가	퍼블릭 및 프라이빗 NAT 게이트웨이에 보조 프라이빗 IPv4 주소를 추가할 수 있습니다. 보조 IPv4 주소는 사용 가능한 포트 수를 늘리므로 워크로드가	2023년 1월 31일

	NAT 게이트웨이를 사용하여 설정할 수 있는 동시 연결 수에 대한 제한이 늘어납니다.	
IAM 모범 실무 따르기	IAM 모범 실무에 따라 가이드가 업데이트되었습니다. 자세한 내용은 IAM의 보안 모범 사례 를 참조하세요.	2023년 1월 4일
NAT 게이트웨이의 프라이빗 IP 주소 선택	NAT 게이트웨이를 생성할 때 이제 NAT 게이트웨이에 할당된 프라이빗 IP 주소를 선택할 수 있습니다. 이전에는 서브넷의 IP 주소 범위로부터 프라이빗 IP 주소가 자동으로 할당되었습니다.	2022년 11월 17일
IPv6 기본 게이트웨이 라우터 구성	이제 기본 VPC 라우터에서 사용할 수 있도록 세 개의 IPv6 주소가 예약되어 있습니다.	2022년 11월 11일
탄력적 IP 주소 전송	이제 하나의 AWS 계정에서 다른 계정으로 탄력적 IP 주소를 전송할 수 있습니다.	2022년 10월 31일
네트워크 주소 사용량 지표	VPC의 크기를 계획하고 모니터링하는 데 도움이 되도록 VPC의 네트워크 주소 사용량 지표를 활성화할 수 있습니다.	2022년 10월 4일
Amazon Data Firehose에 흐름 로그 게시	Amazon Data Firehose 전송 스트림을 흐름 로그 데이터의 대상으로 지정할 수 있습니다.	2022년 9월 8일

NAT 게이트웨이 대역폭	NAT 게이트웨이는 이제 최대 100Gbps의 대역폭(45Gbps에서 증가)을 지원하며 초당 (최소 4백만 개의 패킷에서) 최대 천만 개의 패킷을 처리할 수 있습니다.	2022년 6월 15일
다중 IPv6 CIDR 블록	IPv6 CIDR 블록은 최대 5개까지 VPC에 연결할 수 있습니다.	2022년 5월 12일
재구성	Amazon Virtual Private Cloud 사용 설명서의 일반적인 재구성입니다.	2022년 1월 2일
NAT 게이트웨이 IPv6에서 IPv4로 연결	NAT 게이트웨이는 IPv6에서 IPv4로의 네트워크 주소 변환을 지원합니다(일반적으로 NAT64라고 함).	2021년 11월 24일
VPC의 IPv6 전용 서브넷	IPv6 전용 EC2 인스턴스를 시작할 수 있는 IPv6 전용 서브넷을 생성할 수 있습니다.	2021년 11월 23일
Amazon S3로 VPC 흐름 로그 전송 옵션	Apache Parquet 로그 파일 형식, 시간별 분할 및 Hive 호환 S3 접두사를 지정할 수 있습니다.	2021년 10월 13일
Amazon EC2 Global View	Amazon EC2 Global View를 사용하면 단일 콘솔의 여러 AWS 리전에 걸쳐 VPC, 서브넷, 인스턴스, 보안 그룹 및 볼륨을 볼 수 있습니다.	2021년 9월 1일

<u>더 구체적인 경로</u>	로컬 경로보다 더 구체적인 경로를 라우팅 테이블에 추가할 수 있습니다. 보다 구체적인 경로를 사용하여 VPC 내의 서브넷 간의 트래픽(동서 트래픽)을 미들박스 어플라이언스로 리디렉션할 수 있습니다. 경로의 대상은 VPC에 있는 서브넷의 전체 IPv4 또는 IPv6 CIDR 블록과 일치하도록 설정할 수 있습니다.	2021년 8월 30일
<u>보안 그룹 규칙에 대한 리소스 ID 및 태깅 지원</u>	리소스 ID별로 보안 그룹 규칙을 참조할 수 있습니다. 보안 그룹 규칙에 태그를 추가할 수도 있습니다.	2021년 7월 7일
<u>프라이빗 NAT 게이트웨이</u>	프라이빗 NAT 게이트웨이를 VPC 간 또는 VPC와 온프레미스 네트워크 간의 아웃바운드 전용 프라이빗 통신에 사용할 수 있습니다.	2021년 6월 10일
<u>생성 시 태그</u>	VPC, DHCP 옵션, 인터넷 게이트웨이, 외부 전용 게이트웨이, 네트워크 ACL 및 보안 그룹을 생성할 때 태그를 추가할 수 있습니다.	2020년 6월 30일
<u>관리형 접두사 목록</u>	접두사 목록에서 CIDR 블록 세트를 생성 및 관리할 수 있습니다.	2020년 6월 29일
<u>흐름 로그 개선</u>	새 흐름 로그 필드를 사용할 수 있으며 CloudWatch Logs에 게시되는 흐름 로그의 사용자 지정 형식을 지정할 수 있습니다.	2020년 5월 4일

흐름 로그에 대한 태그 지정 지원	흐름 로그에 태그를 추가할 수 있습니다.	2020년 3월 16일
NAT 게이트웨이 생성 시 태그	NAT 게이트웨이를 만들 때 태그를 추가할 수 있습니다.	2020년 3월 9일
흐름 로그의 최대 집계 간격	흐름을 캡처하고 흐름 로그 레코드로 집계하는 최대 기간을 지정할 수 있습니다.	2020년 2월 4일
네트워크 경계 그룹 구성	Amazon Virtual Private Cloud Console에서 VPC에 대한 네트워크 경계 그룹을 구성할 수 있습니다.	2020년 1월 22일
프라이빗 DNS 이름	프라이빗 DNS 이름을 사용하여 VPC 내에서 AWS PrivateLink 기반 서비스에 비공개로 액세스할 수 있습니다.	2020년 1월 6일
게이트웨이 라우팅 테이블	라우팅 테이블을 게이트웨이와 연결하고 인바운드 VPC 트래픽을 VPC의 특정 네트워크 인터페이스로 라우팅할 수 있습니다.	2019년 12월 3일
흐름 로그 개선	흐름 로그의 사용자 지정 형식을 지정하고 흐름 로그 레코드에 반환할 필드를 선택할 수 있습니다.	2019년 9월 11일
VPC 공유	동일한 VPC에 있는 서브넷을 동일한 AWS 조직의 여러 계정과 공유할 수 있습니다.	2018년 11월 27일
기본 서브넷 생성	기본 서브넷이 없는 가용 영역에서 기본 서브넷을 생성할 수 있습니다.	2017년 11월 9일

NAT 게이트웨이에 태그 지정 지원	NAT 게이트웨이에 태그를 지정할 수 있습니다.	2017년 9월 7일
NAT 게이트웨이에 대한 Amazon CloudWatch 지표	NAT 게이트웨이에 대한 CloudWatch 지표를 볼 수 있습니다.	2017년 9월 7일
보안 그룹 규칙 설명	보안 그룹 규칙에 설명을 추가할 수 있습니다.	2017년 8월 31일
VPC의 보조 IPv4 CIDR 블록	VPC에 여러 개의 IPv4 CIDR 블록을 추가할 수 있습니다.	2017년 8월 29일
탄력적 IP 주소 복구	탄력적 IP 주소를 해제한 경우 복구할 수 있습니다.	2017년 8월 11일
기본 VPC 만들기	기존 기본 VPC를 삭제한 경우 새로운 기본 VPC를 만들 수 있습니다.	2017년 7월 27일
IPv6 지원	IPv6 CIDR 블록을 VPC에 연결하고 IPv6 주소를 VPC의 리소스에 할당할 수 있습니다.	2016년 1월 12일
비 RFC 1918 IP 주소 범위에 대한 DNS 확인 지원	이제 Amazon DNS 서버는 모든 주소 공간에서 프라이빗 DNS 호스트 이름을 프라이빗 IP 주소로 확인할 수 있습니다.	2016년 10월 24일
NAT 게이트웨이	퍼블릭 서브넷에서 NAT 게이트웨이를 만들고 프라이빗 서브넷의 인스턴스를 활성화하여 인터넷 또는 다른 AWS 서비스로 가는 아웃바운드 트래픽을 시작할 수 있습니다.	2015년 12월 17일

VPC 흐름 로그	흐름 로그를 생성하여 VPC의 네트워크 인터페이스에서 전송하고 수신하는 IP 트래픽에 대한 정보를 캡처할 수 있습니다.	2015년 6월 10일
ClassicLink	ClassicLink를 사용하여 EC2-Classic 인스턴스를 계정의 VPC에 연결할 수 있습니다. VPC 보안 그룹을 EC2-Classic 인스턴스에 연결할 수 있으므로 EC2-Classic 인스턴스와 VPC의 인스턴스가 프라이빗 IP 주소를 사용하여 서로 통신할 수 있습니다.	2015년 1월 7일
프라이빗 호스팅 영역 사용	Route 53의 프라이빗 호스팅 영역에서 정의한 사용자 지정 DNS 도메인 이름을 사용하여 VPC의 리소스에 액세스할 수 있습니다.	2014년 11월 5일
서브넷의 퍼블릭 IP 주소 지정 속성 변경	사용자 서브넷의 퍼블릭 IP 주소 지정 속성을 변경하여 해당 서브넷에서 시작한 인스턴스가 퍼블릭 IP 주소를 받을지 여부를 지정할 수 있습니다.	2014년 6월 21일
퍼블릭 IP 주소 할당	시작하는 과정에서 인스턴스에 퍼블릭 IP 주소를 할당할 수 있습니다.	2013년 8월 20일
DNS 호스트 이름 활성화 및 DNS 해석 비활성화	VPC 기본값을 수정하고 DNS 확인을 비활성화하고 DNS 호스트 이름을 활성화할 수 있습니다.	2013년 3월 11일

어디서나 사용 가능한 VPC

5개 AWS 리전의 VPC, 여러 가용 영역의 VPC, AWS 계정당 여러 VPC 및 VPC당 여러 VPN 연결에 대한 지원이 추가되었습니다.

2011년 8월 3일

전용 인스턴스

전용 인스턴스는 단일 고객에게 배정된 하드웨어를 실행하는 VPC 내에서 시작되는 Amazon EC2 인스턴스입니다.

2011년 3월 27일