

Guida per l'utente

# AWS CloudShell



# AWS CloudShell: Guida per l'utente

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

I marchi e il trade dress di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in qualsiasi modo che possa causare confusione tra i clienti o in qualsiasi modo che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà delle rispettive aziende, che possono o meno essere associate, collegate o sponsorizzate da Amazon.

---

# Table of Contents

Che cos'è AWS CloudShell? .....	1
AWS CloudShell features .....	1
AWS Command Line Interface .....	2
Shell e strumenti di sviluppo .....	2
Storage persistente .....	2
Sicurezza .....	3
Opzioni di personalizzazione .....	3
Ripristino della sessione .....	3
Prezzi per AWS CloudShell .....	4
Come iniziare AWS CloudShell? .....	4
AWS CloudShell Argomenti chiave .....	7
Domande frequenti .....	7
Come posso iniziare a usare AWS CloudShell? .....	8
A cosa devo accedere AWS CloudShell? .....	8
Cosa c'è AWS CloudShell sulConsole Toolbar? .....	8
Come posso eseguire il lancio AWS CloudShell suConsole Toolbar? .....	9
Quale Regioni AWS è AWS CloudShell disponibile in? .....	9
Quale Regione AWS viene assegnato se non AWS CloudShell è disponibile nella regione selezionata al momento dell'Console Toolbaravvio CloudShell su? .....	9
In quali tipi di shell posso usare? AWS CloudShell .....	9
Con quali browser web posso usare AWS CloudShell? .....	9
Come posso creare e gestire il mio AWS CloudShell ambiente? .....	9
Quali browser Web posso utilizzare all'avvio AWS CloudShell diConsole Toolbar? .....	10
Posso scaricare un file quando lo avvio AWS CloudShell su? Console Toolbar .....	10
Quale software è preinstallato nel mio ambiente shell? .....	10
Posso installare software che non è disponibile nell'ambiente shell? .....	10
Posso limitare le azioni che gli utenti possono eseguire in AWS CloudShell? .....	11
Come posso spostare i dati dalla mia home directory se voglio cambiare la posizione Regione AWS in cui sto usando AWS CloudShell? .....	11
Posso aumentare il limite che determina il AWS CloudShell timeout a causa dell'inattività dell'utente? .....	11
Posso accedere dalla schermata principale? AWS CloudShellAWS Console Mobile Application .....	11
Come posso avviare AWS CloudShell in? AWS Console Mobile Application .....	11

Posso usare i tasti modificatori sulle mie tastiere iOS e Android quando uso AWS CloudShell in? AWS Console Mobile Application .....	12
Posso dividere la visualizzazione a AWS CloudShell schede in più schede su? AWS Console Mobile Application .....	12
Posso AWS CloudShell accedervi da un dispositivo mobile? Console Toolbar .....	12
Nozioni di base .....	13
Prerequisiti .....	13
Indice .....	14
Passaggio 1: accedi a AWS Management Console .....	14
Passo 2: Seleziona una regione AWS CloudShell, avvia e scegli una shell .....	17
Passaggio 3: scarica un file da AWS CloudShell .....	20
Passaggio 4: Carica un file su AWS CloudShell .....	21
Fase 5: Rimuovere un file da AWS CloudShell .....	22
Passaggio 6: Creare un backup della home directory .....	22
Passaggio 7: riavviare una sessione di shell .....	24
Passaggio 8: Eliminare la home directory di una sessione di shell .....	25
Passaggio 9: Modifica il codice del file ed esegilo utilizzando la riga di comando .....	26
Passaggio 10: utilizzare AWS CLI per aggiungere il file come oggetto in un bucket Amazon S3 .....	28
Argomenti correlati .....	29
Tutorial .....	30
Tutorial: Copiare più file .....	30
Caricamento e download di più file con Amazon S3 .....	31
Caricamento e download di più file utilizzando cartelle zippate .....	34
Tutorial: Utilizzo CodeCommit .....	35
Prerequisiti .....	36
Fase 1: creazione e clonazione di un CodeCommit repository .....	36
Fase 2: Preparazione e salvataggio di un file prima di inviarlo al CodeCommit repository .....	37
Tutorial: creazione di URL predefiniti .....	38
Prerequisiti .....	38
Passaggio 1: creazione di un ruolo IAM per concaccesso al bucket Amazon S3 .....	38
Generazione dell'URL prefirmato .....	40
Tutorial: creazione di un contenitore Docker all'interno AWS CloudShell e trasferimento ad Amazon ECR .....	41
Prerequisiti .....	42
Procedura tutorial .....	42

Eliminazione .....	44
Tutorial: distribuzione di una funzione Lambda utilizzando AWS CDK .....	44
Prerequisiti .....	44
Procedura del tutorial .....	44
Eliminazione .....	47
Utilizzo di AWS CloudShell .....	48
Navigazione nell'interfaccia AWS CloudShell .....	48
.....	48
Lavorare in Regioni AWS .....	50
Specificare l'impostazione predefinita per Regione AWSAWS CLI .....	50
Utilizzo dei file e dello spazio di archiviazione .....	52
Utilizzo di Docker .....	52
Funzionalità di accessibilità .....	54
Navigazione tramite tastiera inCloudShell .....	54
CloudShellfunzionalità di accessibilità dei terminali .....	54
Scelta delle dimensioni dei caratteri e dei temi dell'interfaccia inCloudShell .....	54
Lavorare conAWS i servizi .....	56
AWS CLlesempi da riga di comando perAWS servizi selezionati .....	56
DynamoDB .....	57
AWS Cloud9 .....	57
Amazon EC2 .....	58
S3 Glacier .....	58
AWSCLI di Elastic Beanstalk .....	58
CLI di Amazon ECS .....	59
CLI AWS SAM .....	59
Personalizzazione di AWS CloudShell .....	60
Suddivisione della visualizzazione della riga di comando in più schede .....	60
Modifica della dimensione del carattere .....	61
Modifica del tema dell'interfaccia .....	61
Utilizzo di Safe Paste per testo su più righe .....	61
Usandotmuxper ripristinare la sessione .....	62
Sicurezza .....	3
Protezione dei dati .....	64
Crittografia dei dati .....	65
Identity and Access Management .....	65
Destinatari .....	66

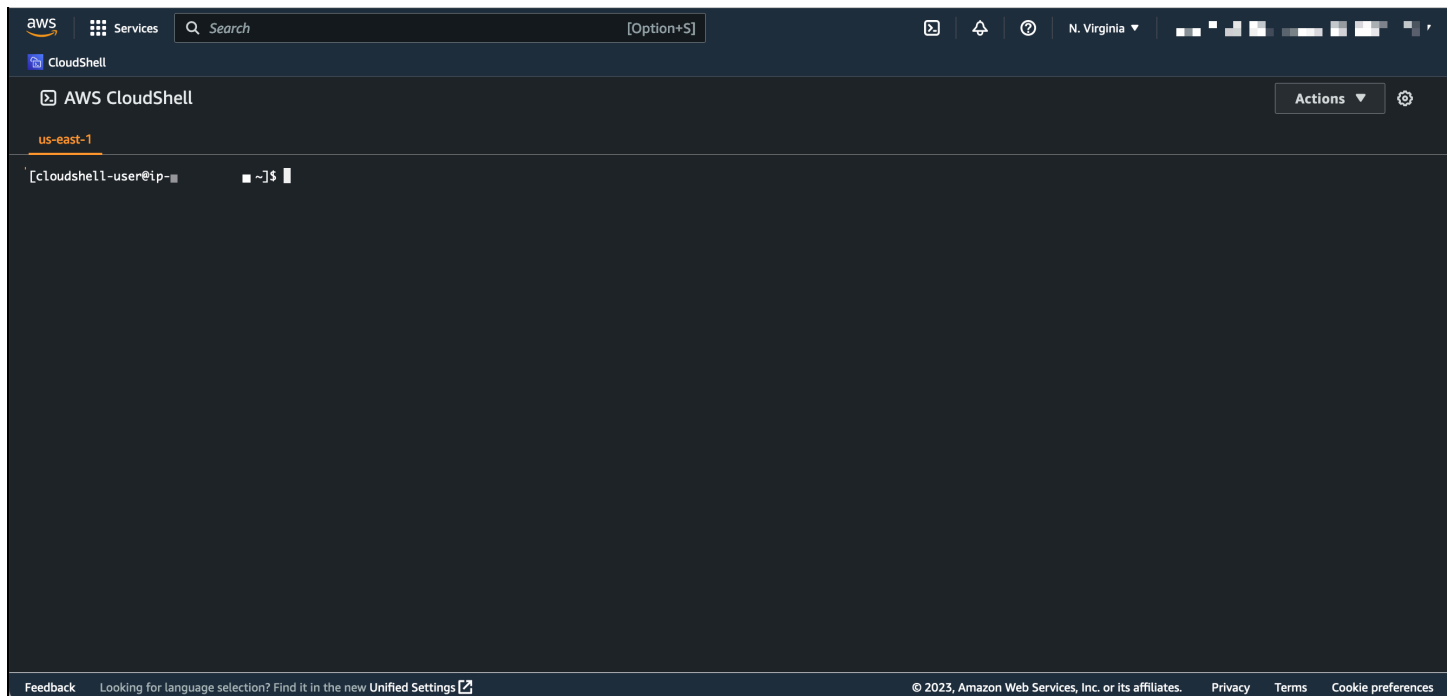
Autenticazione con identità .....	67
Gestione dell'accesso con policy .....	70
In che modo AWS CloudShell funziona con IAM .....	73
Esempi di policy basate su identità .....	80
Risoluzione dei problemi .....	83
Gestione dell' AWS CloudShell accesso e dell'utilizzo con le policy IAM .....	85
Registrazione di log e monitoraggio .....	91
Monitoraggio dell'attività con CloudTrail .....	91
AWS CloudShell in CloudTrail .....	91
Convalida della conformità .....	94
Resilienza .....	99
Sicurezza dell'infrastruttura .....	100
Analisi della configurazione e delle vulnerabilità .....	100
Best practice di sicurezza .....	100
Domande frequenti sulla sicurezza .....	101
Quali sono i AWS processi e le tecnologie utilizzati quando si avvia CloudShell e si avvia una sessione di shell? .....	102
È possibile limitare l'accesso alla rete a CloudShell? .....	102
Posso personalizzare il mio CloudShell ambiente? .....	102
Dove è effettivamente archiviata la mia \$HOME directory in? Cloud AWS .....	102
È possibile crittografare la mia cartella? \$HOME .....	102
Posso eseguire una scansione antivirus sulla mia \$HOME directory? .....	103
AWS CloudShellambiente di calcolo .....	104
Risorse dell'ambiente di calcolo .....	104
CloudShell requisiti di rete .....	104
Software preinstallato .....	105
Conchiglie .....	106
AWSinterfacce a riga di comando (CLI) .....	106
Runtime e SDK AWS: Node.js e Python 3 .....	110
Strumenti di sviluppo e utilità shell .....	113
Installazione AWS CLI nella tua home directory .....	121
Installazione di software di terze parti nell'ambiente shell .....	123
Modificare la shell con degli script .....	124
Migrazione da Amazon Linux 2 ad Amazon Linux 2023 .....	125
AWS CloudShellDomande frequenti sulla migrazione .....	125
Risoluzione dei problemi .....	127

Risoluzione degli errori .....	127
Impossibile avviare l'ambiente. Per riprovare, aggiorna il browser o riavvia selezionando Azioni, Riavvia AWS CloudShell .....	128
Impossibile avviare l'ambiente. Non disponi delle autorizzazioni necessarie. Chiedi al tuo amministratore IAM di concedere l'accesso a AWS CloudShell .....	128
Impossibile accedere alla riga di AWS CloudShell comando .....	128
Impossibile eseguire il ping di indirizzi IP esterni .....	128
Si sono verificati alcuni problemi durante la preparazione del terminale .....	129
I tasti freccia non funzionano correttamente in PowerShell .....	129
I Web Socket non supportati impediscono l'avvio delle sessioni CloudShell .....	130
Impossibile importare il AWSPowerShell.NetCore modulo .....	131
Docker non è in esecuzione quando si utilizza AWS CloudShell .....	132
Docker ha esaurito lo spazio su disco .....	133
docker push è scaduto e continua a riprovare .....	133
Browser supportati .....	134
Regioni supportate .....	135
GovCloud Regioni .....	135
Regioni con consenso esplicito .....	136
Regioni supportate per Docker .....	136
Quota di servizio e limitazione .....	137
Storage persistente .....	137
Utilizzo mensile .....	138
Dimensioni comando .....	138
Conchiglie simultanee .....	138
Sessioni Shell .....	139
Accesso alla rete e trasferimento dati .....	139
Restrizioni sui file di sistema e sul ricaricamento delle pagine .....	139
Cronologia dei documenti .....	141
.....	cxliv

# Che cos'è AWS CloudShell?

AWS CloudShell è una shell preautenticata basata su browser che è possibile avviare direttamente da AWS Management Console. È possibile accedere CloudShell da diversi modi AWS Management Console. Per ulteriori informazioni, consulta la pagina che spiega [come iniziare a utilizzare AWS CloudShell](#).

È possibile eseguire AWS CLI comandi utilizzando la shell preferita, ad esempio Bash PowerShell, o Z shell. E puoi farlo senza scaricare o installare strumenti da riga di comando.



Al momento del lancio AWS CloudShell, viene creato un [ambiente di calcolo](#) basato su Amazon Linux 2023. All'interno di questo ambiente, puoi accedere a un'[ampia gamma di strumenti di sviluppo preinstallati](#), opzioni per [caricare](#) e [scaricare](#) file e uno [storage di file che persiste](#) tra le sessioni.

(Provalo subito:) [Nozioni di base su AWS CloudShell](#)

## AWS CloudShell features

Questo argomento descrive come eseguire l'avvio CloudShell dalla console, passare facilmente da una shell a riga di comando all'altra e personalizzare in base CloudShell alle proprie preferenze. Inoltre, è possibile utilizzare fino a 1 GB di storage persistente in ciascuna Regione AWS unità e in che modo CloudShell l'ambiente è protetto da specifiche funzionalità di sicurezza.



## AWS Command Line Interface

È possibile eseguire l'avvio AWS CloudShell da AWS Management Console. Le AWS credenziali utilizzate per accedere alla console sono automaticamente disponibili in una nuova sessione di shell. Poiché AWS CloudShell gli utenti sono preautenticati, non è necessario configurare le credenziali quando si interagisce con la versione 2. Servizi AWS AWS CLI AWS CLI È preinstallato nell'ambiente di calcolo della shell.

Per ulteriori informazioni sull'interazione con i Servizi AWS uso dell'interfaccia a riga di comando, vedere. [Lavorare con AWS i servizi in AWS CloudShell](#)

## Shell e strumenti di sviluppo

Con la shell creata per AWS CloudShell le sessioni, puoi passare facilmente da una shell a riga di comando preferita all'altra. Più specificamente, puoi passare da Bash PowerShell, a Z shell È inoltre possibile accedere a strumenti e utilità preinstallati. Questi includono git, make, pip, sudo, tar, tmux vimwget, e. zip

L'ambiente shell è preconfigurato con il supporto per diversi dei principali linguaggi software, come Node.js e Python. Ciò significa che, ad esempio, è possibile eseguire Node.js Python progetti senza prima eseguire installazioni di runtime. PowerShell gli utenti possono utilizzare il .NET Core runtime.

È possibile eseguire il commit dei file creati o caricati AWS CloudShell in un repository locale prima di inviarli a un repository remoto gestito da. AWS CodeCommit

Per ulteriori informazioni, consulta [AWS CloudShell ambiente di calcolo: specifiche e software](#).

## Storage persistente

Con AWS CloudShell, puoi utilizzare fino a 1 GB di storage persistente in ciascuno senza Regione AWS costi aggiuntivi. Lo spazio di archiviazione persistente si trova nella tua home directory (\$HOME) ed è privato. A differenza delle risorse ambientali temporanee che vengono riciclate al termine di ogni sessione di shell, i dati nella home directory persistono tra una sessione e l'altra.

Per ulteriori informazioni sulla conservazione dei dati nell'archiviazione persistente, vedere. [Storage persistente](#)

## Sicurezza

L' AWS CloudShell ambiente e i relativi utenti sono protetti da specifiche funzionalità di sicurezza. Ciò include funzionalità come la gestione delle autorizzazioni IAM, le restrizioni delle sessioni di shell e Safe Paste per l'immissione di testo.

### Gestione delle autorizzazioni con IAM

In qualità di amministratore, puoi concedere e negare le autorizzazioni agli AWS CloudShell utenti utilizzando le policy IAM. Puoi anche creare policy che specificano le azioni particolari che gli utenti possono eseguire con l'ambiente shell. Per ulteriori informazioni, consulta [Gestione dell' AWS CloudShell accesso e dell'utilizzo con le policy IAM](#).

### Gestione delle sessioni Shell

Le sessioni inattive e di lunga durata vengono automaticamente interrotte e riciclate. Per ulteriori informazioni, consulta [Sessioni Shell](#).

### Safe Paste per l'immissione di testo

Safe Paste è abilitato per impostazione predefinita. Questa funzionalità di sicurezza richiede la verifica che il testo multilinea che desideri incollare nella shell non contenga script dannosi. Per ulteriori informazioni, consulta [Utilizzo di Safe Paste per testo su più righe](#).

## Opzioni di personalizzazione

Puoi personalizzare la tua AWS CloudShell esperienza in base alle tue esatte preferenze. Ad esempio, puoi modificare il layout dello schermo (più schede), le dimensioni del testo visualizzato e passare dai temi dell'interfaccia chiari a quelli scuri. Per ulteriori informazioni, consulta [Personalizzare ilAWS CloudShellesperienza](#).

[È inoltre possibile estendere l'ambiente shell installando il proprio software e modificando gli script della shell di avvio.](#)

## Ripristino della sessione

La funzionalità di ripristino della sessione ripristina le sessioni in esecuzione su una o più schede del browser nel CloudShell terminale. Se si aggiornano o riaprono le schede del browser chiuse di recente, questa funzionalità riprende la sessione fino all'arresto della shell a causa della sessione

inattiva. Per continuare a utilizzare la CloudShell sessione, premete un tasto qualsiasi nella finestra del terminale. Per ulteriori informazioni sulle sessioni Shell, consulta [Sessioni Shell](#).

Il ripristino della sessione ripristina anche l'output del terminale più recente e i processi in esecuzione in ogni scheda del terminale.

### Note

Il ripristino della sessione non è disponibile nelle applicazioni mobili.

## Prezzi per AWS CloudShell

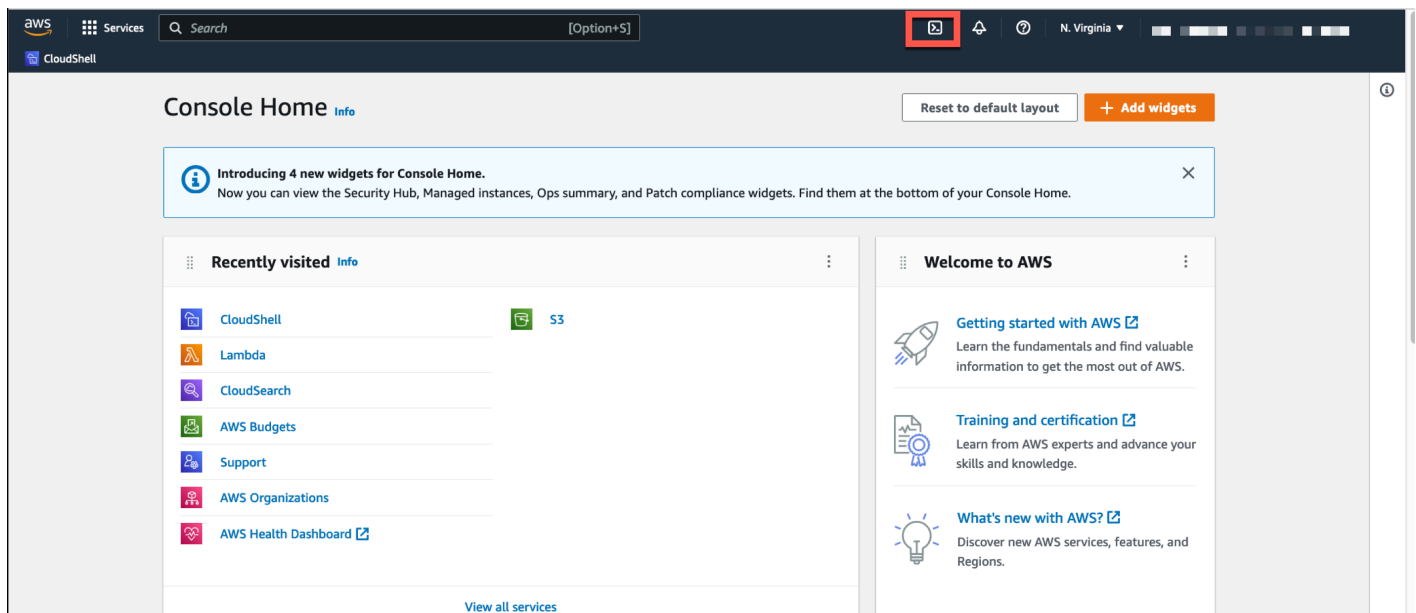
AWS CloudShell è Servizio AWS un servizio disponibile senza costi aggiuntivi. Tuttavia, paghi per AWS le altre risorse che utilizzi AWS CloudShell. Inoltre, si applicano anche le [velocità di trasferimento dati standard](#). Per ulteriori informazioni, consultare [Prezzi di AWS CloudShell](#).

Per ulteriori informazioni, consulta [Quote e restrizioni di servizio perAWS CloudShell](#).

## Come iniziare AWS CloudShell?

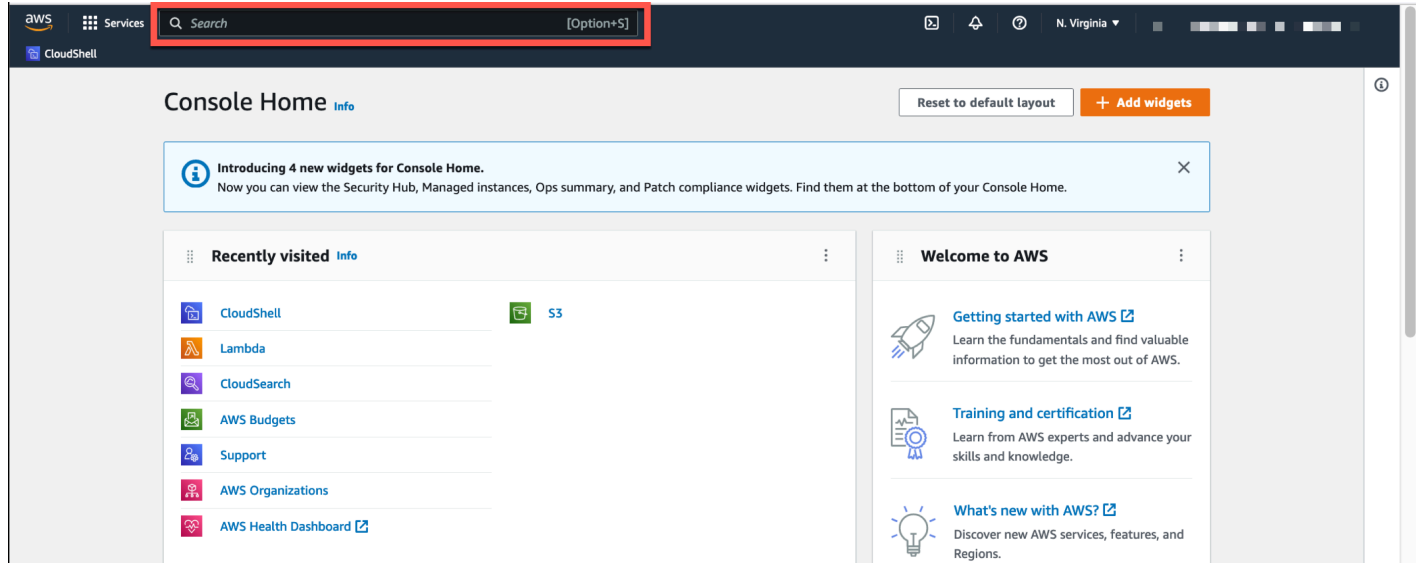
Per iniziare a lavorare con la shell, accedi a AWS Management Console e scegli una delle seguenti opzioni:

- Nella barra di navigazione, scegli l'CloudShell icona.



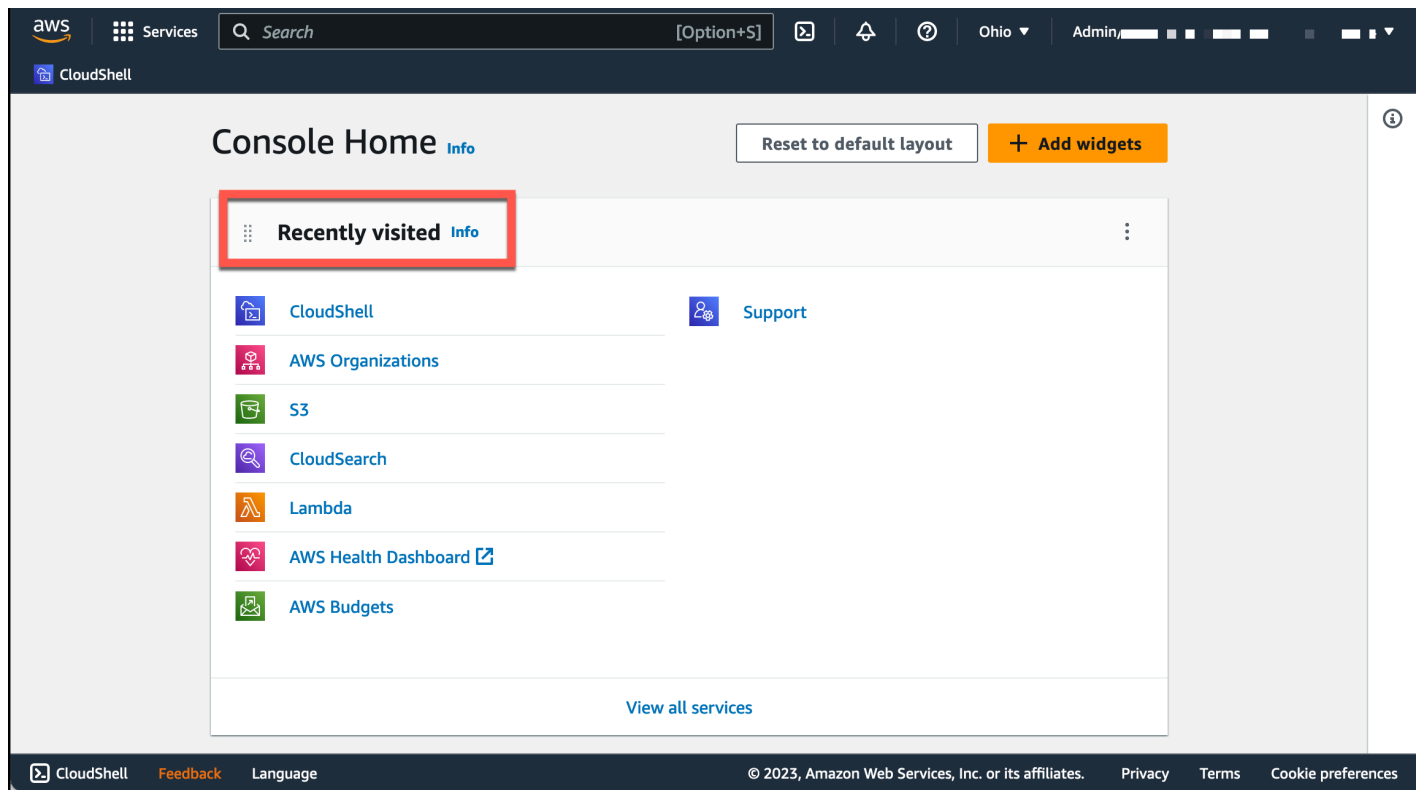
- Nella casella di ricerca, digita «CloudShell», quindi scegli CloudShell.

Questo passaggio apre la CloudShell sessione a schermo intero.

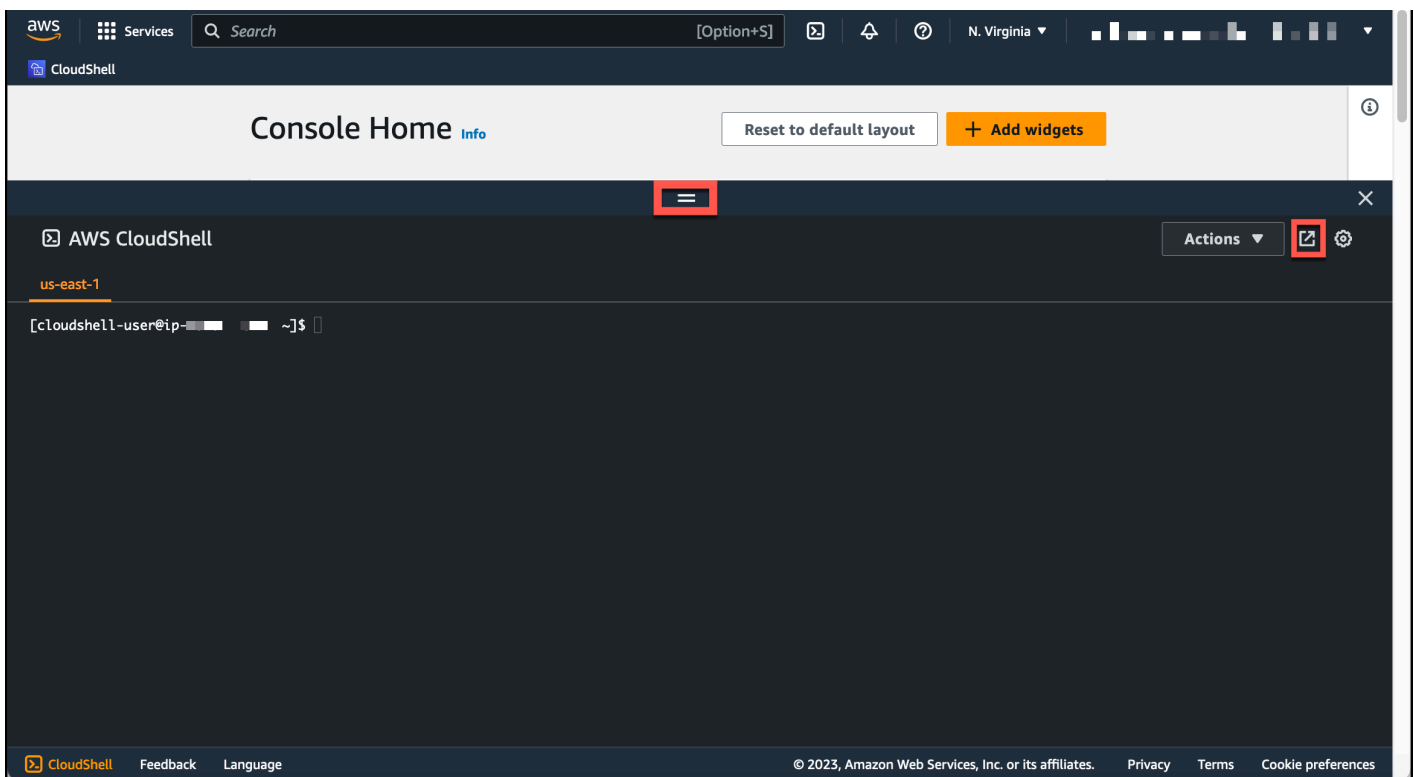
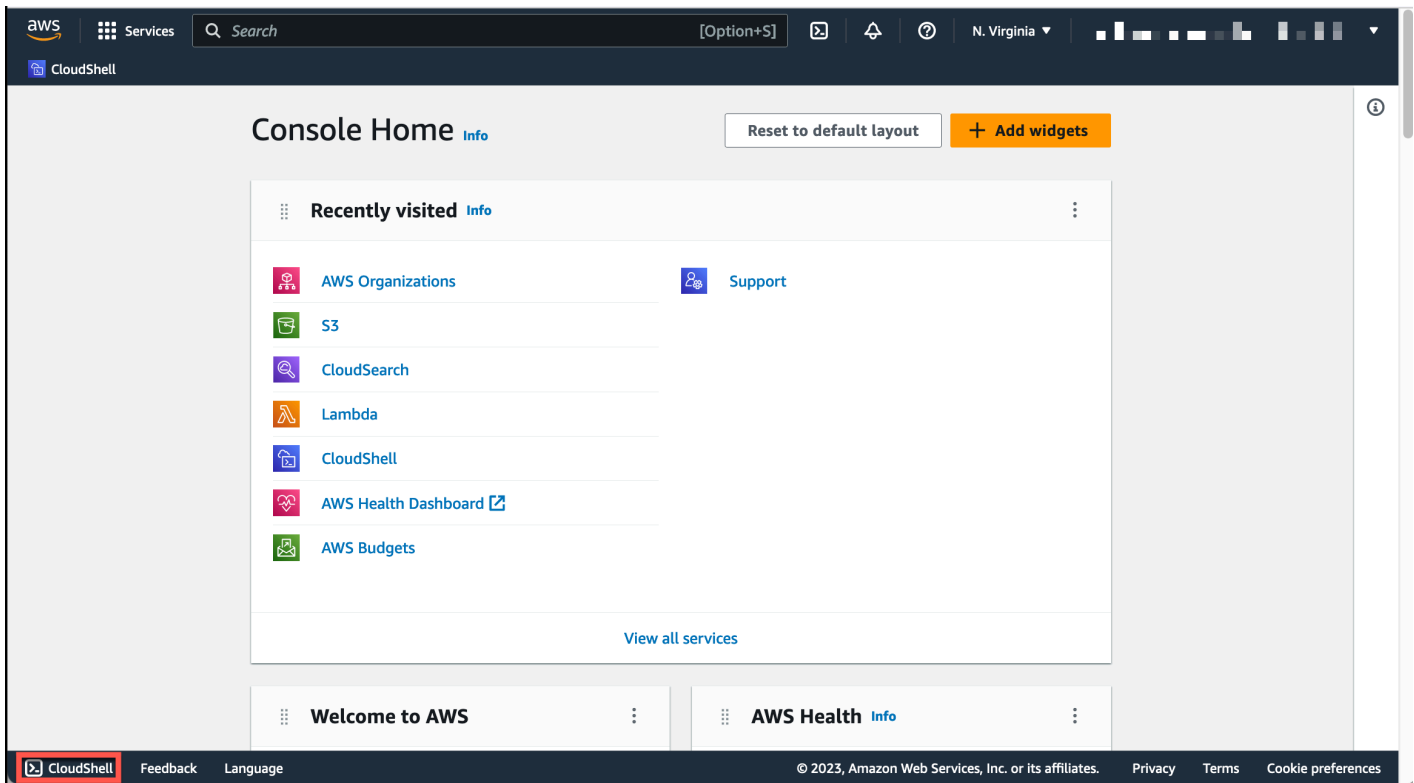


- Nel widget Visitato di recente, scegli CloudShell.

Questo passaggio apre la CloudShell sessione a schermo intero.



- Scegli CloudShell su Console Toolbar, in basso a sinistra della console. Puoi regolare l'altezza della CloudShell sessione trascinandola=.



Puoi anche passare alla CloudShell sessione a schermo intero facendo clic su Apri in una nuova scheda del browser.

Per istruzioni su come accedere a AWS Management Console ed eseguire le attività principali con AWS CloudShell, consulta [Guida introduttiva AWS CloudShell](#).

## AWS CloudShell Argomenti chiave

- [Nozioni di base su AWS CloudShell](#)
- [Utilizzo di AWS CloudShell](#)
- [Lavorare con AWS i servizi in AWS CloudShell](#)
- [Personalizzare il AWS CloudShell esperienza](#)
- [AWS CloudShell ambiente di calcolo: specifiche e software](#)

## AWS CloudShell Domande frequenti

Di seguito sono riportate le risposte ad alcune domande comuni su AWS CloudShell.

Per altre domande frequenti incentrate sulla sicurezza, consulta [AWS CloudShell Domande frequenti sulla sicurezza](#).

- [Come posso iniziare a usare AWS CloudShell?](#)
- [A cosa devo accedere AWS CloudShell?](#)
- [Cosa c'è AWS CloudShell sul Console Toolbar?](#)
- [Come posso eseguire il lancio AWS CloudShell su Console Toolbar?](#)
- [Come posso creare e gestire il mio AWS CloudShell ambiente?](#)
- [Quale Regioni AWS è AWS CloudShell disponibile in?](#)
- [Quale Regione AWS viene assegnato se non AWS CloudShell è disponibile nella regione selezionata al momento del lancio CloudShell su Console Toolbar?](#)
- [In quali tipi di shell posso usare? AWS CloudShell](#)
- [Con quali browser web posso usare AWS CloudShell?](#)
- [Quali browser Web posso utilizzare all'avvio AWS CloudShell su Console Toolbar?](#)
- [Posso scaricare un file all'avvio AWS CloudShell su Console Toolbar?](#)
- [Quale software è preinstallato nel mio ambiente shell?](#)

- [Posso installare software che non è disponibile nell'ambiente shell?](#)
- [Posso limitare le azioni che gli utenti possono eseguire in AWS CloudShell?](#)
- [Come posso spostare i dati dalla mia home directory se voglio cambiare la posizione Regione AWS in cui sto usando AWS CloudShell?](#)
- [Posso aumentare il limite che determina il AWS CloudShell timeout a causa dell'inattività dell'utente?](#)
- [Posso accedere AWS CloudShellAWS Console Mobile Application dalla schermata principale?](#)
- [Come posso avviare AWS CloudShell in AWS Console Mobile Application?](#)
- [Posso usare i tasti modificatori sulle mie tastiere IOS e Android quando uso AWS CloudShell in? AWS Console Mobile Application](#)
- [Posso dividere la visualizzazione a AWS CloudShell schede in più schede su? AWS Console Mobile Application](#)
- [Posso accedere AWS CloudShell dalla barra degli strumenti della console su un dispositivo mobile?](#)

## Come posso iniziare a usare AWS CloudShell?

Puoi iniziare avviandolo AWS CloudShell in pochi passaggi da. AWS Management Console [A tale scopo, accedi alla console utilizzando le tue credenziali Account AWS o IAM all'indirizzo <https://console.aws.amazon.com/console/home>.](#)

Per ulteriori informazioni, consulta la pagina [Nozioni di base di AWS CloudShell](#).

## A cosa devo accedere AWS CloudShell?

Poiché accedi AWS CloudShell da AWS Management Console, devi essere un utente IAM in grado di fornire un alias o un ID di account validi, un nome utente e una password.

Per l'avvio AWS CloudShell sulla console, sono necessarie le autorizzazioni IAM fornite dalla policy allegata. Per ulteriori informazioni, consulta [Gestione dell' AWS CloudShell accesso e dell'utilizzo con le policy IAM](#).

## Cosa c'è AWS CloudShell sulConsole Toolbar?

L' CloudShell icona in basso a sinistra di AWS Management Console.

## Come posso eseguire il lancio AWS CloudShell su Console Toolbar?

Puoi avviarlo AWS CloudShell su Console Toolbar scegliendo l'CloudShell icona in basso a sinistra della console.

## Quale Regioni AWS è AWS CloudShell disponibile in?

Per un elenco degli endpoint di servizio supportati Regioni AWS e associati, consulta la [AWS CloudShell pagina](#) in. Riferimenti generali di Amazon Web Services

## Quale Regione AWS viene assegnato se non AWS CloudShell è disponibile nella regione selezionata al momento dell'Console Toolbaravvio CloudShell su?

La regione predefinita è assegnata a una regione più vicina alla regione selezionata. Per maggiori informazioni, consulta [Seleziona una regione AWS CloudShell, avvia e scegli una shell](#).

È possibile eseguire il comando che fornisce le autorizzazioni per gestire le risorse in una regione diversa da quella predefinita. Per ulteriori informazioni, consulta [Working in Regioni AWS](#).

## In quali tipi di shell posso usare? AWS CloudShell

In AWS CloudShell, è possibile eseguire comandi utilizzando il Bash shell PowerShell, o il Z shell. Per cambiare shell, inserisci il nome della shell che desideri utilizzare utilizzando il seguente formato nel prompt dei comandi:

- bash: Usate il Bash shell
- pwsh: Usa PowerShell
- zsh: Usa il Z shell

## Con quali browser web posso usare AWS CloudShell?

AWS CloudShell supporta le versioni più recenti dei browser Google Chrome, Mozilla Firefox, Microsoft Edge e Apple Safari.

## Come posso creare e gestire il mio AWS CloudShell ambiente?

Il tuo AWS CloudShell ambiente viene creato e gestito in base all'ID utente IAM per regione. Puoi verificarlo UserId eseguendo `aws sts get-caller-identity`. L'ambiente è di proprietà dell'ID



utente IAM in quella regione specifica. Potrai accedere a un AWS CloudShell ambiente diverso se modifichi l'IAM UserId o la regione.

## Quali browser Web posso utilizzare all'avvio AWS CloudShell diConsole Toolbar?

Puoi avviare Console Toolbar utilizzando CloudShell le versioni più recenti dei browser Google Chrome, Microsoft Edge, Mozilla Firefox e Apple Safari.

## Posso scaricare un file quando lo avvio AWS CloudShell su? Console Toolbar

Sì, puoi scaricare un file quando lo avvii CloudShell suConsole Toolbar. Puoi scaricare un file utilizzando le versioni più recenti dei browser Google Chrome e Microsoft Edge.

Al momento, non è possibile scaricare un file utilizzando i browser Mozilla Firefox e Apple Safari.

## Quale software è preinstallato nel mio ambiente shell?

Con la shell creata per AWS CloudShell le sessioni, puoi passare facilmente da una shell a riga di comando preferita all'altra (Bash, e). PowerShell Z shell Puoi anche avere accesso a strumenti e utilità preinstallati come Make, pip, sudo e. tar tmux Vim Wget Zip

L'ambiente shell è preconfigurato con supporto per la maggior parte dei principali linguaggi software. Ad esempio, è possibile utilizzarlo per eseguire Node.js Python progetti senza dover prima eseguire installazioni in fase di esecuzione. PowerShell gli utenti possono utilizzare il .NET Core runtime.

È possibile aggiungere file creati utilizzando la shell o caricati con l'interfaccia shell a un repository controllato dalla versione gestito utilizzando una versione preinstallata di. git

Per ulteriori informazioni, consulta [Software preinstallato](#).

## Posso installare software che non è disponibile nell'ambiente shell?

Sì, AWS CloudShell gli utenti dispongono sudo dei privilegi e possono installare il software dalla riga di comando. Per ulteriori informazioni, consulta [Installazione di software di terze parti nell'ambiente shell](#).

## Posso limitare le azioni che gli utenti possono eseguire in AWS CloudShell?

Sì, puoi controllare le azioni che gli utenti possono eseguire AWS CloudShell. Ad esempio, puoi consentire agli utenti di accedere AWS CloudShell ma impedire loro di caricare o scaricare file all'interno dell'ambiente shell. Oppure, in alternativa, puoi impedire completamente l'accesso AWS CloudShell agli utenti. Per ulteriori informazioni, consulta [Gestione dell' AWS CloudShell accesso e dell'utilizzo con le policy IAM](#).

## Come posso spostare i dati dalla mia home directory se voglio cambiare la posizione Regione AWS in cui sto usando AWS CloudShell?

Per spostare i AWS CloudShell dati da una regione Regione AWS all'altra, è necessario innanzitutto scaricare il contenuto della home directory di una regione sul computer locale, quindi caricarlo nella home directory di un'altra regione. Per ulteriori informazioni, consulta [Tutorial: copia di più file tra il computer locale eAWS CloudShell](#).

## Posso aumentare il limite che determina il AWS CloudShell timeout a causa dell'inattività dell'utente?

La sessione di shell termina automaticamente dopo circa 20-30 minuti se non interagisci AWS CloudShell utilizzando la tastiera o il puntatore. I processi in esecuzione non contano come interazioni. Poiché CloudShell è progettato per attività mirate e basate sulle attività, al momento non è previsto un aumento di questo [limite di timeout](#).

[Se desideri eseguire attività basate su terminali utilizzando un Servizio AWS timeout più flessibile, ti consigliamo di utilizzare il nostro IDE basato sul cloud o di avviare e connetterti a un'AWS Cloud9istanza Amazon EC2.](#)

## Posso accedere dalla schermata principale? AWS CloudShellAWS Console Mobile Application

Sì, puoi accedere AWS CloudShell AWS Console Mobile Application accedendo alla Console Mobile Application. Per ulteriori informazioni, consulta la [Guida per l'utente di AWS Console Mobile Application](#).

## Come posso avviare AWS CloudShell in? AWS Console Mobile Application

È possibile eseguire AWS CloudShell l'avvio utilizzando uno dei seguenti metodi:

1. Seleziona l'AWS CloudShell icona nella parte inferiore della barra di navigazione.
2. Seleziona AWS CloudShell nel menu Servizi.

## Posso usare i tasti modificatori sulle mie tastiere iOS e Android quando uso AWS CloudShell in? AWS Console Mobile Application

Sì, puoi usare i tasti modificatori sulle tastiere iOS e Android. Per ulteriori informazioni, consulta la [Guida per l'utente di AWS Console Mobile Application](#).

## Posso dividere la visualizzazione a AWS CloudShell schede in più schede su? AWS Console Mobile Application

No, al momento non è possibile eseguire più AWS CloudShell schede sulla propria applicazione mobile.

## Posso AWS CloudShell accedervi da un dispositivo mobile? Console Toolbar

No, al momento non puoi AWS CloudShell accedervi dal Console Toolbar tuo dispositivo mobile.

# Nozioni di base su AWS CloudShell

Questo tutorial introduttivo mostra come avviare AWS CloudShell ed eseguire attività chiave utilizzando l'interfaccia a riga di comando della shell.

Innanzitutto, accedi a AWS Management Console e seleziona un. Regione AWS Quindi si avvia CloudShell in una nuova finestra del browser e in un tipo di shell con cui lavorare.

Successivamente, create una nuova cartella nella vostra home directory e caricate un file in essa dal computer locale. Si lavora su quel file utilizzando un editor preinstallato prima di eseguirlo come programma dalla riga di comando. Infine, richiami AWS CLI i comandi per creare un bucket Amazon S3 e aggiungi il file come oggetto al bucket.

## Prerequisiti

### Autorizzazioni IAM

Puoi ottenere le autorizzazioni per AWS CloudShell allegando la seguente policy AWS gestita alla tua identità IAM (ad esempio un utente, un ruolo o un gruppo):

- `AWSCloudShellFullAccess`: Fornisce agli utenti l'accesso completo alle AWS CloudShell relative funzionalità.

In questo tutorial, interagisci anche con Servizi AWS. Più specificamente, interagisci con Amazon S3 creando un bucket S3 e aggiungendo un oggetto a quel bucket. La tua identità IAM richiede una policy che conceda, come minimo, le autorizzazioni `s3:CreateBucket` `s3:PutObject`

Per ulteriori informazioni, consulta [Amazon S3 Actions nella Amazon Simple Storage Service User Guide](#).

### File di esercizi

Questo esercizio prevede anche il caricamento e la modifica di un file che viene poi eseguito come programma dall'interfaccia a riga di comando. Apri un editor di testo sul tuo computer locale e aggiungi il seguente frammento di codice.

```
import sys
```

```
x=int(sys.argv[1])
y=int(sys.argv[2])
sum=x+y
print("The sum is",sum)
```

Salva il file con il nome `add_prog.py`.

## Indice

- [Passaggio 1: accedi a AWS Management Console](#)
- [Passaggio 2: Seleziona una regione AWS CloudShell, avvia e scegli una shell](#)
- [Passaggio 3: Scarica un file da AWS CloudShell](#)
- [Passaggio 4: carica un file su AWS CloudShell](#)
- [Fase 5: Rimuovere un file da AWS CloudShell](#)
- [Passaggio 6: Creare un backup della home directory](#)
- [Fase 7: Riavviare una sessione di shell](#)
- [Passo 8: Eliminare la home directory di una sessione di shell](#)
- [Passo 9: Modifica il codice del file ed eseguilò dalla riga di comando](#)
- [Passaggio 10: utilizzare AWS CLI per aggiungere il file come oggetto in un bucket Amazon S3](#)

## Passaggio 1: accedi a AWS Management Console

Questo passaggio prevede l'immissione delle informazioni utente IAM per accedere a AWS Management Console. Se sei già nella console, vai al [passaggio 2](#).

- Puoi accedere a AWS Management Console utilizzando un URL di accesso per gli utenti IAM o accedendo alla pagina di accesso principale.

### IAM user sign-in URL

- Apri un browser e inserisci il seguente URL di accesso. Sostituiscilo `account_alias_or_id` con l'alias o l'ID dell'account fornito dall'amministratore.

```
https://account_alias_or_id.signin.aws.amazon.com/console/
```

- Inserisci le tue credenziali di accesso IAM e scegli **Accedi**.

## Sign in as IAM user

Account ID (12 digits) or account alias

IAM user name

Password

Sign in

[Sign in using root user email](#)

[Forgot password?](#)

### Main sign-in page

- [Apri https://aws.amazon.com/console/.](https://aws.amazon.com/console/)
- Se non hai effettuato l'accesso in precedenza utilizzando questo browser, viene visualizzata la pagina di accesso principale. Scegli utente IAM, inserisci l'alias o l'ID dell'account e scegli Avanti.

## Sign in

**Root user**

Account owner that performs tasks requiring unrestricted access. [Learn more](#)

**IAM user**

User within an account that performs daily tasks. [Learn more](#)

**Account ID (12 digits) or account alias**

**Next**

- Se hai già effettuato l'accesso come utente IAM in precedenza. Il tuo browser potrebbe ricordare l'alias o l'ID dell'account per. Account AWS In tal caso, inserisci le tue credenziali di accesso IAM e scegli Accedi.

## Sign in as IAM user

**Account ID (12 digits) or account alias**

**IAM user name**

**Password**

**Sign in**

[Sign in using root user email](#)

[Forgot password?](#)


 Note

Puoi anche accedere come utente [root](#). Questa identità ha accesso completo a tutte Servizi AWS le risorse dell'account. Ti consigliamo vivamente di non utilizzare l'utente root per le attività quotidiane, nemmeno quelle amministrative. Rispettare piuttosto la best practice di utilizzare l'utente root soltanto per creare il tuo primo utente IAM.


## Passo 2: Seleziona una regioneAWS CloudShell, avvia e scegli una shell

In questo passaggio, si avvia AWS CloudShell dall'interfaccia della console, si sceglie una shell disponibile Regione AWS e si passa alla shell preferita, ad esempio Bash PowerShell, oZ shell.

1. Per scegliere una Regione AWS regione in cui lavorare, vai al menu Seleziona una regione e seleziona una [AWSregione supportata](#) in cui lavorare. (Le regioni disponibili sono evidenziate.)

 Important

Se si cambia regione, l'interfaccia si aggiorna e il nome della persona selezionata Regione AWS viene visualizzato sopra il testo della riga di comando. Tutti i file aggiunti alla memoria persistente sono disponibili solo in questo spazio. Regione AWS Se si modificano le regioni, sono accessibili file e archivi diversi.

 Important

Se non CloudShell è disponibile nella regione selezionata all'avvio CloudShell suConsole Toolbar, nella parte inferiore sinistra della console, la regione predefinita è impostata sulla regione più vicina alla regione selezionata. Puoi eseguire il comando che fornisce le autorizzazioni per gestire le risorse in una regione diversa da quella predefinita. Per ulteriori informazioni, consulta [Working in Regioni AWS](#).

Example

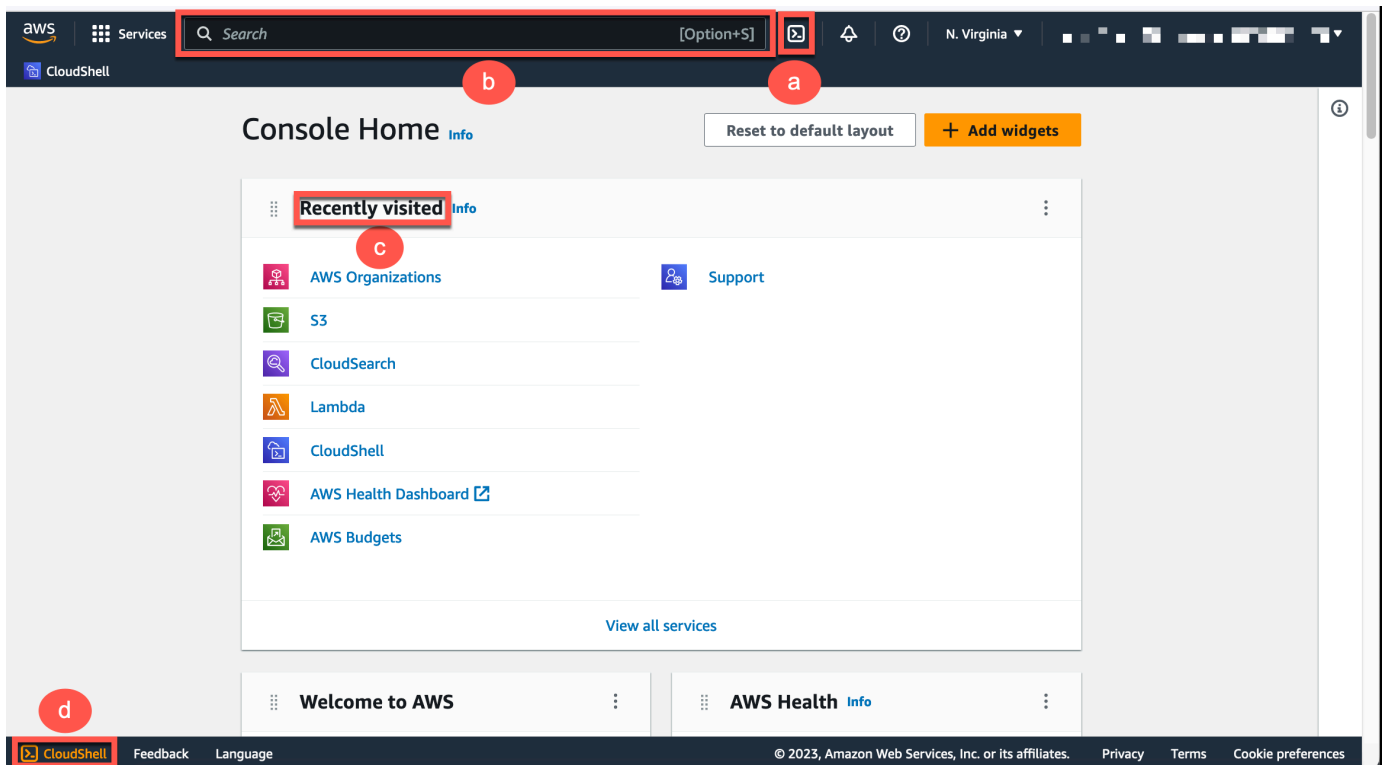
Esempio



Se scegli Europa (Spagna) eu-south-2 ma non CloudShell è disponibile in Europa (Spagna)eu-south-2, la regione predefinita è impostata su Europa (Irlanda)eu-west-1, che è la più vicina all'Europa (Spagna)eu-south-2.

Utilizzerai le quote di servizio per la regione predefinita, Europa (Irlanda) eu-west-1 e la stessa CloudShell sessione verrà ripristinata in tutte le regioni. La regione predefinita potrebbe essere modificata e riceverai una notifica nella finestra del CloudShell browser.

2. DaAWS Management Console, puoi avviarlo CloudShell scegliendo una delle seguenti opzioni:
  1. Nella barra di navigazione, scegli l'CloudShellicona.
  2. Nella casella di ricerca, digita «CloudShell», quindi scegli CloudShell.
  3. Nel widget Visitato di recente, scegli CloudShell.
  4. Scegli CloudShellsuConsole Toolbar, in basso a sinistra della console.
    - Per regolare l'altezza della CloudShell sessione, trascina=.
    - Per passare alla CloudShell sessione a schermo intero, fai clic sull'icona Apri in una nuova scheda del browser.



Quando viene visualizzato il prompt dei comandi, la shell è pronta per l'interazione.

 Note

Se riscontri problemi che ti impediscono di avviare o interagire con successo AWS CloudShell, consulta le informazioni necessarie per identificare e risolvere tali problemi.


[Risoluzione dei problemi relativi a AWS CloudShell](#)

3. Per scegliere una shell preinstallata con cui lavorare, inserisci il nome del programma al prompt della riga di comando.

### Bash

```
bash
```

Se passate a Bash, il simbolo nel prompt dei comandi si aggiorna a. \$

 Note

Bash è la shell predefinita che viene eseguita all'avvio AWS CloudShell.

### PowerShell

```
pwsh
```

Se si passa a PowerShell, il simbolo nella riga di comando viene aggiornato a PS>.

### Z shell

```
zsh
```

Se si passa a Z shell, il simbolo visualizzato nel prompt dei comandi viene aggiornato a. %

Per informazioni sulle versioni preinstallate nel tuo ambiente shell, consulta la [tabella shells](#) nella sezione Ambiente di [CloudShell calcolo AWS](#).

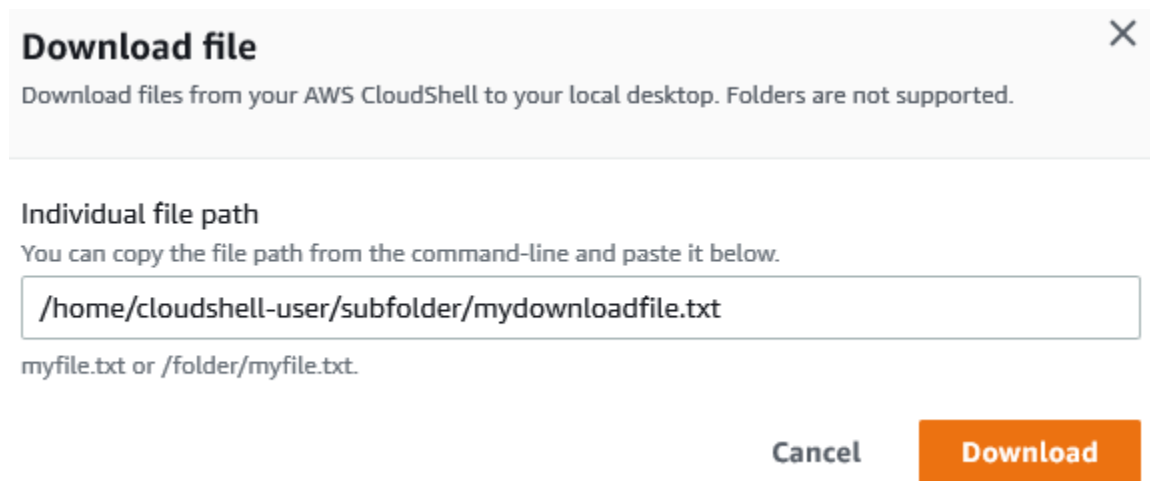
## Passaggio 3: scarica un file da AWS CloudShell

Questo passaggio illustra il processo di download di un file.

1. Per scaricare un file, vai su Azioni e scegli Scarica file dal menu.

Viene visualizzata la finestra di dialogo Scarica file.

2. Nella finestra di dialogo Scarica file, inserite il percorso del file da scaricare.



### Note

È possibile utilizzare percorsi assoluti o relativi quando si specifica un file da scaricare. Con i nomi di percorso relativi, `/home/cloudshell-user/` viene aggiunto automaticamente all'inizio per impostazione predefinita. Quindi, per scaricare un file chiamato `mydownload-file`, entrambi i seguenti sono percorsi validi:

- Percorso assoluto: `/home/cloudshell-user/subfolder/mydownloadfile.txt`
- Percorso relativo: `subfolder/mydownloadfile.txt`

3. Scegli Download (Scarica).

Se il percorso del file è corretto, viene visualizzata una finestra di dialogo. È possibile utilizzare questa finestra di dialogo per aprire il file con l'applicazione predefinita. In alternativa, è possibile salvare il file in una cartella sul computer locale.

**Note**

L'opzione Download non è disponibile all'avvio CloudShell su Console Toolbar. Puoi scaricare un file dalla CloudShell console o utilizzando il browser web Chrome. Per ulteriori informazioni su come scaricare un file, consulta [Passaggio 3: Scaricare un file da AWS CloudShell](#).

## Passaggio 4: Carica un file su AWS CloudShell

Questo passaggio descrive come caricare un file e poi spostarlo in una nuova directory nella home directory.

1. Per controllare la directory di lavoro corrente, al prompt immettete il seguente comando:

```
pwd
```

Quando premete Invio, la shell restituisce la directory di lavoro corrente (ad esempio, /home/cloudshell-user).

2. Per caricare un file in questa directory, andate su Azioni e scegliete Carica file dal menu.

Viene visualizzata la finestra di dialogo Carica file.

3. Scegliere Browse (Sfogliare).
4. Nella finestra di dialogo di caricamento dei file del sistema, selezionate il file di testo creato per questo tutorial (add\_prog.py) e scegliete Apri.
5. Nella finestra di dialogo Carica file, scegliete Carica.

Una barra di avanzamento tiene traccia del caricamento. Se il caricamento ha esito positivo, un messaggio conferma che add\_prog.py è stato aggiunto alla cartella principale della home directory.

6. Per creare una directory per il file, inserisci il comando `make directories:mkdir mysub_dir`.
7. Per spostare il file caricato dalla radice della tua home directory alla nuova directory, usa il `mv` comando:

```
mv add_prog.py mysub_dir.
```

8. Per cambiare la tua directory di lavoro nella nuova directory, inserisci `cd mysub_dir`.

Il prompt dei comandi si aggiorna per indicare che avete cambiato la directory di lavoro.

9. Per visualizzare il contenuto della directory `currentmysub_dir`, immettete il `ls` comando.

Viene elencato il contenuto della directory di lavoro. Questo include il file che hai appena caricato.

## Fase 5: Rimuovere un file da AWS CloudShell

Questo passaggio descrive come rimuovere un file da AWS CloudShell.

1. Per rimuovere un file da AWS CloudShell, utilizzate i comandi di shell standard come `rm` (remove).

```
rm my-file-for-removal
```

2. Per rimuovere più file che soddisfano i criteri specificati, esegui il `find` comando.

L'esempio seguente rimuove tutti i file che includono il suffisso «.pdf» nei loro nomi.

```
find -type f -name '*.pdf' -delete
```

### Note

Supponiamo che smettiate di utilizzarlo AWS CloudShell in uno specifico. Regione AWS Quindi, i dati presenti nella memoria persistente di quella regione vengono rimossi automaticamente dopo un periodo specificato. Per ulteriori informazioni, consulta [Persistent Storage](#).

## Passaggio 6: Creare un backup della home directory

1. Creare un file di backup

Creare una cartella temporanea all'esterno della home directory.

```
HOME_BACKUP_DIR=$(mktemp --directory)
```

È possibile utilizzare una delle seguenti opzioni per creare un backup:

- a. Crea un file di backup usando tar

Per creare un file di backup usando tar, inserisci il seguente comando:

```
tar \
  --create \
  --gzip \
  --verbose \
  --file=${HOME_BACKUP_DIR}/home.tar.gz \
  [--exclude ${HOME}/.cache] \ // Optional
  ${HOME}/
echo "Home directory backed up to this file: ${HOME_BACKUP_DIR}/home.tar.gz"
```

- b. Crea un file di backup usando zip

Per creare un file di backup utilizzando zip, inserisci il seguente comando:

```
zip \
  --recurse-paths \
  ${HOME_BACKUP_DIR}/home.zip \
  ${HOME} \
  [--exclude ${HOME}/.cache/\*] // Optional
echo "Home directory backed up to this file: ${HOME_BACKUP_DIR}/home.zip"
```

## 2. Trasferisci il file di backup all'esterno CloudShell

È possibile utilizzare una delle seguenti opzioni per trasferire il file di backup all'esterno CloudShell:

- a. Scarica il file di backup sul tuo computer locale

Puoi scaricare il file creato nel passaggio precedente. Per ulteriori informazioni su come scaricare un file da CloudShell, consulta [Scaricare un file da AWS CloudShell](#).

Nella finestra di dialogo per il download del file, inserite il percorso del file da scaricare (ad esempio, /tmp/tmp.iA99tD9L98/home.tar.gz).


- b. Trasferisci il file di backup su S3

Per generare un bucket, inserisci il seguente comando:

```
aws s3 mb s3://${BUCKET_NAME}
```

Usa AWS CLI per copiare il file nel bucket S3:

```
aws s3 cp ${HOME_BACKUP_DIR}/home.tar.gz s3://${BUCKET_NAME}
```

 Note


Potrebbero essere applicati costi per il trasferimento dei dati.

### 3. Backup diretto su un bucket S3

Per eseguire il backup direttamente su un bucket S3, inserisci il seguente comando:

```
aws s3 cp \  
  ${HOME}/ \  
  s3://${BUCKET_NAME} \  
  --recursive \  
  [--exclude .cache/\*] // Optional
```

## Passaggio 7: riavviare una sessione di shell

 Note


Come misura di sicurezza, se non interagisci con la shell utilizzando la tastiera o il puntatore per un periodo prolungato, la sessione si interrompe automaticamente. Anche le sessioni di lunga durata vengono interrotte automaticamente. Per ulteriori informazioni, consulta [Sessioni Shell](#).

1. Per riavviare una sessione di shell, scegli Azioni, Riavvia AWS CloudShell.

Riceverai una notifica che il riavvio AWS CloudShell interrompe tutte le sessioni attive nella versione corrente. Regione AWS


2. Per confermare, scegli Riavvia.

Un'interfaccia visualizza un messaggio che indica che l'ambiente di CloudShell calcolo si sta arrestando. Dopo l'arresto e il riavvio dell'ambiente, è possibile iniziare a utilizzare la riga di comando in una nuova sessione.

 Note

In alcuni casi, il riavvio dell'ambiente potrebbe richiedere alcuni minuti.

## Passaggio 8: Eliminare la home directory di una sessione di shell

 Warning

L'eliminazione della home directory è un'azione irreversibile in cui tutti i dati archiviati nella directory principale vengono eliminati definitivamente. Tuttavia, potresti prendere in considerazione questa opzione nelle seguenti situazioni:

- Hai modificato erroneamente un file e non puoi accedere all'ambiente di AWS CloudShell calcolo. L'eliminazione della home directory ripristina le impostazioni AWS CloudShell predefinite.
- Vuoi rimuovere AWS CloudShell immediatamente tutti i tuoi dati. Se si interrompe l'utilizzo AWS CloudShell in una AWS regione, lo storage persistente viene [automaticamente eliminato al termine del periodo di conservazione](#), a meno che non si AWS CloudShell riavvii nella regione.

Se hai bisogno di archiviazione a lungo termine per i tuoi file, prendi in considerazione un servizio come Amazon S3 o. CodeCommit

1. Per eliminare una sessione di shell, scegli Azioni, Elimina la CloudShell home directory di AWS.

Ti viene comunicato che l'eliminazione della AWS CloudShell home directory elimina tutti i dati attualmente archiviati nel tuo AWS CloudShell ambiente.



**Note**

Questa operazione non può essere annullata.

2. Per confermare l'eliminazione, inserisci `delete` nel campo di immissione del testo, quindi scegli Elimina.

**Delete AWS CloudShell home directory**

Deleting your home directory will delete all data currently stored in your AWS CloudShell environment. This action cannot be undone. AWS CloudShell stops all active sessions in the current AWS Region and creates a new environment immediately.

To confirm deletion, enter **delete** in the text input field.

Cancel

Delete

AWS CloudShell interrompe tutte le sessioni attive nella versione corrente Regione AWS e crea immediatamente un nuovo ambiente.

Uscire manualmente dalle sessioni di shell

Con la riga di comando, è possibile lasciare una sessione di shell e disconnettersi utilizzando il `exit` comando. È quindi possibile premere un tasto qualsiasi per riconnettersi e continuare a utilizzare AWS CloudShell.

## Passaggio 9: Modifica il codice del file ed esegilo utilizzando la riga di comando

Questo passaggio dimostra come utilizzare l'Vimeditor preinstallato per lavorare con un file. Quindi esegui quel file come programma dalla riga di comando.

1. Per modificare il file che hai caricato nel passaggio precedente, inserisci il seguente comando:

```
vim add_prog.py
```

L'interfaccia della shell si aggiorna per visualizzare l'Vimeditor.

2. Per modificare il file inVim, premi il I tasto. Ora modificate il contenuto in modo che il programma sommi tre numeri anziché due.

```
import sys
x=int(sys.argv[1])
y=int(sys.argv[2])
z=int(sys.argv[3])
sum=x+y+z
print("The sum is",sum)
```

#### Note

Se incollate il testo nell'editor e avete abilitato la [funzione Safe Paste](#), viene visualizzato un avviso. Il testo su più righe copiato può contenere script dannosi. Con la funzione Safe Paste, puoi verificare il testo completo prima che venga incollato. Se ritieni che il testo sia sicuro, scegli Incolla.

3. Dopo aver modificato il programma, premete Esc per accedere alla modalità di Vim comando. Quindi, inserisci il :wq comando per salvare il file e uscire dall'editor.

#### Note

Se non conosci la modalità di Vim comando, inizialmente potresti trovare difficile passare dalla modalità di comando alla modalità di inserimento. La modalità di comando viene utilizzata quando si salvano file e si esce dall'applicazione. La modalità di inserimento viene utilizzata quando si inserisce nuovo testo. Per accedere alla modalità di inserimento, premete el, per accedere alla modalità di comando, Esc premete. Per ulteriori informazioni Vim e sugli altri strumenti disponibili inAWS CloudShell, vedere [Strumenti di sviluppo e utilità shell](#).

4. Nell'interfaccia principale della riga di comando, esegui il seguente programma e specifica tre numeri da inserire. La sintassi è esposta di seguito.

```
python3 add_prog.py 4 5 6
```

La riga di comando mostra l'output del programma:The sum is 15.

## Passaggio 10: utilizzare AWS CLI per aggiungere il file come oggetto in un bucket Amazon S3

In questo passaggio, crei un bucket Amazon S3 e poi usi il `PutObject` metodo per aggiungere il tuo file di codice come oggetto in quel bucket.

### Note

Nella maggior parte dei casi, puoi [utilizzare un servizio come CodeCommit](#) il commit di un file software in un repository controllato dalla versione. Questo tutorial mostra come utilizzare AWS CLI in AWS CloudShell per interagire con altri servizi AWS. Utilizzando questo metodo, non è necessario scaricare o installare alcuna risorsa aggiuntiva. Inoltre, poiché hai già eseguito l'autenticazione alla shell, non è necessario configurare le credenziali prima di effettuare chiamate.

1. Per creare un bucket in uno specifico Regione AWS, inserisci il seguente comando:

```
aws s3api create-bucket --bucket insert-unique-bucket-name-here --region us-east-1
```

### Note

Se stai creando un bucket al di fuori della `us-east-1` regione, aggiungilo `create-bucket-configuration` con il `LocationConstraint` parametro per specificare la regione. Di seguito è riportato un esempio di sintassi.

```
$ aws s3api create-bucket --bucket my-bucket --region eu-west-1 --create-bucket-configuration LocationConstraint=eu-west-1
```

Se la chiamata ha esito positivo, la riga di comando visualizza una risposta del servizio simile all'output seguente.

```
{
  "Location": "/insert-unique-bucket-name-here"
}
```

**Note**

Se non rispettate [le regole per la denominazione dei bucket](#), viene visualizzato il seguente errore: Si è verificato un errore (InvalidBucketName) durante la chiamata dell' CreateBucketoperazione: Il bucket specificato non è valido.

2. Per caricare un file e aggiungerlo come oggetto al bucket appena creato, chiamate il metodo PutObject

```
aws s3api put-object --bucket insert-unique-bucket-name-here --key add_prog --body add_prog.py
```

Dopo che l'oggetto è stato caricato nel bucket Amazon S3, la riga di comando visualizza una risposta dal servizio simile al seguente output:

```
{"ETag": "\"ab123c1:w:wad4a567d8bfd9a1234ebee56\""} 
```

ETagÈ l'hash dell'oggetto che è stato archiviato. Puoi utilizzare questo hash per [verificare l'integrità dell'oggetto caricato su Amazon S3](#).

## Argomenti correlati

- [Lavorare conAWS i servizi inAWS CloudShell](#)
- [Tutorial: copia di più file tra il computer locale eAWS CloudShell](#)
- [Tutorial: Utilizzo CodeCommit inAWS CloudShell](#)
- [Utilizzo di AWS CloudShell](#)
- [Personalizzare ilAWS CloudShellesperienza](#)

# Tutorial di AWS CloudShell

I seguenti tutorial ti permetteranno di sperimentare e testare diverse funzionalità e integrazioni durante l'utilizzo. AWS CloudShell

## Argomenti

- [Tutorial: copia di più file tra il computer locale e AWS CloudShell](#)
- [Tutorial: Utilizzo CodeCommit in AWS CloudShell](#)
- [Tutorial: creazione di un URL prefirmato per oggetti Amazon S3 utilizzando informazioni simili alle seguenti. AWS CloudShell](#)
- [Tutorial: creazione di un contenitore Docker all'interno AWS CloudShell e trasferimento in un repository Amazon ECR](#)
- [Tutorial: distribuzione di una funzione Lambda utilizzando AWS CDK](#)

## Tutorial: copia di più file tra il computer locale e AWS CloudShell

Utilizzando l' CloudShell interfaccia, è possibile caricare o scaricare un singolo file alla volta tra il computer locale e l'ambiente shell. Per copiare più file contemporaneamente tra CloudShell il tuo computer locale, usa una delle seguenti opzioni:

- Amazon S3: utilizza i bucket S3 come intermediario per copiare file tra il tuo computer locale e CloudShell.
- File zip: comprimi più file in un'unica cartella zippata che può essere caricata o scaricata utilizzando l' CloudShell interfaccia.

### Note

Poiché CloudShell non consente il traffico Internet in entrata, al momento non è possibile utilizzare comandi come `scp` o `rsync` per copiare più file tra computer locali e l'ambiente di CloudShell elaborazione.

# Caricamento e download di più file con Amazon S3

## Prerequisiti

Per lavorare con bucket e oggetti, è necessaria una policy IAM che conceda le autorizzazioni per eseguire le seguenti azioni API Amazon S3:

- `s3:CreateBucket`
- `s3:PutObject`
- `s3:GetObject`

Per un elenco completo delle azioni di Amazon S3, consulta [Azioni](#) nel riferimento dell'API di Amazon Simple Storage Service.

## Caricare più file AWS CloudShell utilizzando Amazon S3

1. In AWS CloudShell, crea un bucket S3 eseguendo il seguente comando:

```
aws s3api create-bucket --bucket your-bucket-name --region us-east-1
```

Se la chiamata ha esito positivo, la riga di comando visualizza una risposta dal servizio S3:

```
{
  "Location": "/your-bucket-name"
}
```

2. Carica i file in una directory dal tuo computer locale al bucket. Per caricare i file, scegli una delle seguenti opzioni:
  - AWS Management Console: utilizzato drag-and-drop per caricare file e cartelle in un bucket.
  - AWS CLI: con la versione dello strumento installata sul computer locale, utilizza la riga di comando per caricare file e cartelle nel bucket.

## Using the console

- Aprire la console Amazon S3 su <https://s3.console.aws.amazon.com/s3/>.

(Se lo stai usando AWS CloudShell, dovresti già aver effettuato l'accesso alla console.)

- Nel pannello di navigazione a sinistra, scegli Bucket, quindi scegli il nome del bucket in cui desideri caricare le cartelle o i file. Puoi anche creare un bucket a tua scelta scegliendo Crea bucket.
- Per selezionare i file e le cartelle che desideri caricare, scegli Carica. Poi trascina i file e le cartelle selezionati nella finestra della console in cui sono elencati gli oggetti nel bucket di destinazione, oppure scegli Aggiungi file o Aggiungi cartelle.

I file scelti vengono elencati nella pagina Upload (Carica).

- Seleziona le caselle di controllo per indicare i file da aggiungere.
- Per aggiungere i file selezionati al bucket, scegli Carica.

#### Note

Per informazioni sulla gamma completa di opzioni di configurazione quando si utilizza la console, consulta [Come caricare file e cartelle in un bucket S3?](#) nella Guida per l'utente di Amazon Simple Storage.

## Using AWS CLI

#### Note

Per questa opzione, è necessario che loAWS CLI strumento sia installato sul computer locale e che le credenziali siano configurate per le chiamate aiAWS servizi. Per ulteriori informazioni, consulta la [Guida per l'utente AWS Command Line Interface](#).


- Avvia loAWS CLI strumento ed esegui il seguenteaws s3 comando per sincronizzare il bucket specificato con il contenuto della directory corrente sul tuo computer locale:

```
aws s3 sync folder-path s3://your-bucket-name
```

Se la sincronizzazione ha esito positivo, vengono visualizzati i messaggi di caricamento per ogni oggetto aggiunto al bucket.

3. Tornate alla riga di CloudShell comando e immettete il seguente comando per sincronizzare la directory nell'ambiente shell con il contenuto del bucket S3:


```
aws s3 sync s3://your-bucket-name folder-path
```

 Note

È inoltre possibile aggiungere `--exclude "<value>"` `--include "<value>"` parametri al `sync` comando per eseguire la corrispondenza dei modelli per escludere o includere un particolare file o oggetto.

Per ulteriori informazioni, vedere [Utilizzo dei filtri di esclusione e inclusione](#) nel riferimento dei CLI comandi.

Se la sincronizzazione ha esito positivo, vengono visualizzati i messaggi di download per ogni file scaricato dal bucket alla directory.


 Note

Con il comando `sync`, solo i file nuovi e aggiornati vengono copiati in modo ricorsivo dalla directory di origine a quella di destinazione.

Scarica più file AWS CloudShell utilizzando Amazon S3

1. Utilizzando la riga di comando AWS CloudShell, inserisci il seguente `aws s3` comando per sincronizzare un bucket S3 con i contenuti della directory corrente nell'ambiente shell:

```
aws s3 sync folder-path s3://your-bucket-name
```

 Note

È inoltre possibile aggiungere `--exclude "<value>"` `--include "<value>"` parametri al `sync` comando per eseguire la corrispondenza dei modelli per escludere o includere un particolare file o oggetto.



Per ulteriori informazioni, vedere [Utilizzo dei filtri di esclusione e inclusione](#) nel riferimento deiAWS CLI comandi.

Se la sincronizzazione ha esito positivo, vengono visualizzati i messaggi di caricamento per ogni oggetto aggiunto al bucket.

2. Scarica il contenuto del bucket sul tuo computer locale. Poiché la console Amazon S3 non supporta il download di più oggetti, è necessario utilizzare loAWS CLI strumento installato sul computer locale.

Dalla riga di comando delloAWS CLI strumento, eseguire il comando seguente:

```
aws s3 sync s3://your-bucket-name folder-path
```

Se la sincronizzazione ha esito positivo, la riga di comando visualizza un messaggio di download per ogni file aggiornato o aggiunto nella directory di destinazione.

#### Note

Per questa opzione, è necessario che loAWS CLI strumento sia installato sul computer locale e che le credenziali siano configurate per le chiamate aiAWS servizi. Per ulteriori informazioni, consulta la [Guida per l'utente AWS Command Line Interface](#).

## Caricamento e download di più file utilizzando cartelle zippate

Con le utilità zip/unzip, è possibile comprimere più file in un archivio che può essere trattato come un singolo file. Le utilità sono preinstallate nell'ambiente di CloudShell elaborazione.

Per ulteriori informazioni sugli strumenti preinstallati, consulta [Strumenti di sviluppo e utilità shell](#).

Carica più fileAWS CloudShell utilizzando cartelle zippate

1. Sul computer locale, aggiungi i file da caricare in una cartella zippata.
2. Avvia CloudShell, quindi scegli Azioni, Carica file.
3. Nella finestra di dialogo Carica file, scegli Seleziona file, quindi scegli la cartella compressa che hai appena creato.

4. Nella finestra di dialogo Carica file, scegliete Carica per aggiungere il file selezionato all'ambiente shell.
5. Nella CloudShell riga di comando, esegui il seguente comando per decomprimere il contenuto dell'archivio zip in una directory specificata:

```
unzip zipped-files.zip -d my-unzipped-folder
```

### Scarica più fileAWS CloudShell utilizzando cartelle zippate

1. Nella CloudShell riga di comando, esegui il seguente comando per aggiungere tutti i file nella directory corrente a una cartella compressa:

```
zip -r zipped-archive.zip *
```

2. Scegli Azioni, Scarica file.
3. Nella finestra di dialogo Scarica file, inserisci il percorso della cartella compressa (/home/cloudshell-user/zip-folder/zipped-archive.zipad esempio), quindi scegli Scarica.

Se il percorso è corretto, una finestra di dialogo del browser offre la possibilità di aprire la cartella compressa o salvarla sul computer locale.

4. Sul computer locale, ora puoi decomprimere il contenuto della cartella zippata scaricata.

## Tutorial: Utilizzo CodeCommit inAWS CloudShell

CodeCommit è un servizio sicuro, altamente scalabile e gestito che ospita i repository Git privati. UtilizzandoAWS CloudShell, è possibile lavorare CodeCommit sulla riga di comando utilizzando l'`git-remote-codecommit` utilità. Questa utilità è preinstallata nell'ambiente diAWS CloudShell elaborazione e fornisce un metodo semplice per inviare ed estrarre codice dai CodeCommit repository. Questa utilità esegue questa operazione estendendo Git. Per ulteriori informazioni, consulta la [Guida per l'utente AWS CodeCommit](#).

Questo tutorial descrive come creare un CodeCommit repository e clonarlo nel tuo ambiente diAWS CloudShell elaborazione. Imparerai anche come eseguire lo stage e il commit di un file nel tuo repository clonato prima di trasferirlo nel repository remoto gestito inAWS Cloud.

## Prerequisiti

Per informazioni sulle autorizzazioni che un utente IAM deve utilizzare AWS CloudShell, consulta la [sezione dei prerequisiti nel tutorial introduttivo](#). Sono inoltre necessarie [le autorizzazioni IAM](#) per lavorare CodeCommit.

Inoltre, prima di iniziare, assicurati di avere quanto segue:

- Una conoscenza di base dei comandi Git e dei concetti di controllo delle versioni
- Un file nella home directory della shell che può essere salvato nei repository locali e remoti. In questo tutorial, viene denominato `my-git-file`.

## Fase 1: creazione e clonazione di un CodeCommit repository

1. Nell'interfaccia a riga di CloudShell comando, inserisci il seguente `codecommit` comando per creare un CodeCommit repository chiamato `MyDemoRepo`.

```
aws codecommit create-repository --repository-name MyDemoRepo --repository-  
description "My demonstration repository"
```

Se il repository viene creato correttamente, la riga di comando visualizza la risposta del servizio.

```
{  
  "repositoryMetadata": {  
    "accountId": "111122223333",  
    "repositoryId": "0dcd29a8-941a-1111-1111-11111111111a",  
    "repositoryName": "MyDemoRepo",  
    "repositoryDescription": "My demonstration repository",  
    "lastModifiedDate": "2020-11-23T20:38:23.068000+00:00",  
    "creationDate": "2020-11-23T20:38:23.068000+00:00",  
    "cloneUrlHttp": "https://git-codecommit.eu-west-1.amazonaws.com/v1/repos/  
MyDemoRepo",  
    "cloneUrlSsh": "ssh://git-codecommit.eu-west-1.amazonaws.com/v1/repos/  
MyDemoRepo",  
    "Arn": "arn:aws:codecommit:eu-west-1:111111111111:MyDemoRepo"  
  }  
}
```

2. Aprire la directory dalla riga di comando per il repository locale e andare alla directory di lavoro.

```
mkdir my-shell-repo
cd my-shell-repo
```

3. Per clonare il repository remoto, utilizzare il `git clone` comando. (Mentre lavori `git-remote-codecommit`, usa lo stile URL HTTPS (GRC)).

```
git clone codecommit::eu-west-1://MyDemoRepo
```

Se il repository viene clonato correttamente, la riga di comando visualizza la risposta del servizio.

```
Cloning into 'MyDemoRepo'...
warning: You appear to have cloned an empty repository.
```

4. Per accedere al repository clonato, utilizzare il `cd` comando.

```
cd MyDemoRepo
```

## Fase 2: Preparazione e salvataggio di un file prima di inviarlo al CodeCommit repository

1. Aggiungi un file chiamato `my-git-file` alla `MyDemoRepo` cartella usando un editor Vim o la funzione di caricamento dei file di AWS CloudShell. Per informazioni sull'utilizzo di entrambi, consulta il [tutorial introduttivo](#).
2. Per mettere il file all'interno del repository, esegui il `git add` comando.

```
git add my-git-file
```

3. Per verificare che il file sia stato messo in fase e sia pronto per essere salvato, esegui il `git status` comando.

```
git status
```

`my-git-file` viene elencato come nuovo file e viene visualizzato in verde, a indicare che è pronto per essere salvato.

4. Salva questa versione del file staged nel repository.

```
git commit -m "first commit to repo"
```

### Note

Se ti vengono richieste informazioni di configurazione per completare il commit, usa il seguente formato.

```
$ git config --global user.name "Jane Doe"  
$ git config --global user.email janedoe@example.com
```

5. Per sincronizzare il tuo repository remoto con le modifiche apportate in quello locale, invia le modifiche al ramo upstream.

```
git push
```

## Tutorial: creazione di un URL prefirmato per oggetti Amazon S3 utilizzando informazioni simili alle seguenti.

Questo tutorial mostra come creare un URL prefirmato per condividere un oggetto Amazon S3 con altri. Poiché i proprietari degli oggetti specificano le proprie credenziali di sicurezza durante la condivisione, chiunque riceva l'URL predefinito può accedere all'oggetto per un periodo di tempo limitato.

### Prerequisiti

- Un utente IAM con autorizzazioni di accesso fornite dalla `AWSCloudShellFullAccesspolicy`.
- Per le autorizzazioni IAM necessarie per creare un URL predefinito, consulta [Condividere un oggetto con altri](#) nella Guida per l'utente di Amazon Simple Storage Service.

## Passaggio 1: creazione di un ruolo IAM per concaccesso al bucket Amazon S3

1. Per ottenere i dettagli IAM che possono essere condivisi, chiama il `get-caller-identity` comando da AWS CloudShell.



```
s3:prefix:
  - "myfolder/*"
PolicyName: S3ReadSpecificFolder
Roles:
  - !Ref CollaboratorRole
Outputs:
  CollaboratorRoleArn:
    Description: Arn for the Collaborator's Role
    Value: !GetAtt CollaboratorRole.Arn
```

3. Salva ilAWS CloudFormation modello in un file denominatotemplate.yaml.
4. Usa il modello per distribuire lo stack e creare il ruolo IAM chiamando ildeploy comando.

```
aws cloudformation deploy --template-file ./template.yaml --stack-name
  CollaboratorRole --capabilities CAPABILITY_IAM
```

## Generazione dell'URL prefirmato

1. Usando il tuo editorAWS CloudShell, aggiungi il seguente codice. Questo codice crea un URL che fornisce agli utenti federati l'accesso diretto aAWS Management Console.

```
import urllib, json, sys
import requests
import boto3
import os

def main():
    sts_client = boto3.client('sts')
    assume_role_response = sts_client.assume_role(
        RoleArn=os.environ.get(ROLE_ARN),
        RoleSessionName="collaborator-session"
    )
    credentials = assume_role_response['Credentials']
    url_credentials = {}
    url_credentials['sessionId'] = credentials.get('AccessKeyId')
    url_credentials['sessionKey'] = credentials.get('SecretAccessKey')
    url_credentials['sessionToken'] = credentials.get('SessionToken')
    json_string_with_temp_credentials = json.dumps(url_credentials)
    print(f"json string {json_string_with_temp_credentials}")
```

```

request_parameters = f"?
Action=getSignInToken&Session={urllib.parse.quote(json_string_with_temp_credentials)}"
request_url = "https://signin.aws.amazon.com/federation" + request_parameters
r = requests.get(request_url)
signin_token = json.loads(r.text)
request_parameters = "?Action=login"
request_parameters += "&Issuer=Example.org"
request_parameters += "&Destination=" + urllib.parse.quote("https://us-
west-2.console.aws.amazon.com/cloudshell")
request_parameters += "&SignInToken=" + signin_token["SignInToken"]
request_url = "https://signin.aws.amazon.com/federation" + request_parameters

# Send final URL to stdout
print (request_url)

if __name__ == "__main__":
    main()

```

2. Salva il codice in un file chiamato `share.py`.
3. Eda l'Amazon Resource Name (ARN) del ruolo IAM da AWS CloudFormation. Quindi, le Python credenziali di sicurezza provvisorie.

```

ROLE_ARN=$(aws cloudformation describe-stacks --stack-name CollaboratorRole --query
"Stacks[*].Outputs[?OutputKey=='CollaboratorRoleArn'].OutputValue" --output text)
python3 ./share.py

```

Lo script restituisce un URL a cui un collaboratore può fare clic per AWS CloudShell inserirlo AWS Management Console. Il collaboratore ha il pieno controllo della `my folder/` cartella nel bucket Amazon S3 per i successivi 3.600 secondi (1 ora). Le credenziali scadono dopo un'ora. Trascorso questo tempo, il collaboratore non può più accedere al bucket.

## Tutorial: creazione di un contenitore Docker all'interno AWS CloudShell e trasferimento in un repository Amazon ECR

Questo tutorial mostra come definire e creare un container Docker AWS CloudShell e inviarlo a un repository Amazon ECR.



## Prerequisiti

- È necessario disporre delle autorizzazioni necessarie per creare e inviare un repository Amazon ECR. Per ulteriori informazioni sui repository con Amazon ECR, consulta gli [archivi privati di Amazon ECR nella Amazon ECR User Guide](#). Per ulteriori informazioni sulle autorizzazioni necessarie per il push di immagini con Amazon ECR, consulta [Autorizzazioni IAM richieste per il push di un'immagine nella Amazon ECR User Guide](#).

## Procedura tutorial

Il seguente tutorial illustra come utilizzare l' CloudShell interfaccia per creare un contenitore Docker e inviarlo a un repository Amazon ECR.

1. Crea una nuova cartella nella tua home directory.

```
mkdir ~/docker-cli-tutorial
```

2. Vai alla cartella che hai creato.

```
cd ~/docker-cli-tutorial
```

3. Crea un Dockerfile vuoto.

```
touch Dockerfile
```

4. Utilizzando un editor di testo, ad esempionano Dockerfile, apri il file e incolla il seguente contenuto al suo interno.

```
# Dockerfile

# Base this container on the latest Amazon Linux version
FROM public.ecr.aws/amazonlinux/amazonlinux:latest

# Install the cowsay binary
RUN dnf install --assumeyes cowsay

# Default entrypoint binary
ENTRYPOINT [ "cowsay" ]

# Default argument for the cowsay entrypoint
```

```
CMD [ "Hello, World!" ]
```

- Il Dockerfile è ora pronto per essere creato. Costruisci il contenitore eseguendo `docker build`. Etichetta il contenitore con un easy-to-type nome da utilizzare nei comandi futuri.

```
docker build --tag test-container .
```

Assicurati di includere il punto finale (.).

- Ora puoi testare il contenitore per verificare che funzioni correttamente in AWS CloudShell.

```
docker container run test-container
```

- Ora che hai un contenitore Docker funzionante, devi inviarlo a un repository Amazon ECR. Se disponi di un repository Amazon ECR esistente, puoi saltare questo passaggio.

Esegui il seguente comando per creare un repository Amazon ECR per questo tutorial.

```
ECR_REPO_NAME=docker-tutorial-repo  
aws ecr create-repository --repository-name ${ECR_REPO_NAME}
```

- Dopo aver creato il repository Amazon ECR, puoi inviare il contenitore Docker al suo interno.

Esegui il comando seguente per ottenere le credenziali di accesso Amazon ECR per Docker.

```
AWS_ACCOUNT_ID=$(aws sts get-caller-identity --query "Account" --output text)  
ECR_URL=${AWS_ACCOUNT_ID}.dkr.ecr.${AWS_REGION}.amazonaws.com  
aws ecr get-login-password | docker login --username AWS --password-stdin  
${ECR_URL}
```

- Etichetta l'immagine con il repository Amazon ECR di destinazione e poi inviala a tale repository.

```
docker tag test-container ${ECR_URL}/${ECR_REPO_NAME}  
docker push ${ECR_URL}/${ECR_REPO_NAME}
```

Se riscontri errori o riscontri problemi durante il tentativo di completare questo tutorial, consulta la sezione [Risoluzione dei problemi](#) di questa guida per ricevere assistenza.

## Eliminazione

Ora hai distribuito con successo il tuo contenitore Docker nel tuo repository Amazon ECR. Per rimuovere i file creati in questo tutorial dal tuo AWS CloudShell ambiente, esegui il comando seguente.

- ```
cd ~  
rm -rf ~/docker-cli-tutorial
```

- Elimina il repository Amazon ECR.

```
aws ecr delete-repository --force --repository-name ${ECR_REPO_NAME}
```

## Tutorial: distribuzione di una funzione Lambda utilizzando AWS CDK

Questo tutorial mostra come distribuire una funzione Lambda sul tuo account utilizzando il. AWS Cloud Development Kit (AWS CDK)

### Prerequisiti

- Avvia il tuo account per utilizzarlo con. AWS CDK Per informazioni sul bootstrap conAWS CDK, consulta [Bootstrapping](#) nella Guida per sviluppatori v2. AWS CDK Se non hai ancora avviato l'account, puoi iniziare. `cdk bootstrap CloudShell`
- Assicurati di disporre delle autorizzazioni appropriate per distribuire risorse sul tuo account. Le autorizzazioni di amministratore sono consigliate.

### Procedura del tutorial

Il seguente tutorial illustra come distribuire una funzione Lambda basata su container Docker utilizzando. AWS CDK

1. Crea una nuova cartella nella tua home directory.

```
mkdir ~/docker-cdk-tutorial
```

- Vai alla cartella che hai creato.

```
cd ~/docker-cdk-tutorial
```

- Installa le AWS CDK dipendenze localmente.

```
npm install aws-cdk aws-cdk-lib
```

- Crea un AWS CDK progetto scheletrico nella cartella che hai creato.

```
touch cdk.json
mkdir lib
touch lib/docker-tutorial.js lib/Dockerfile lib/hello.js
```

- Utilizzando un editor di testo, ad esempionano `cdk.json`, apri il file e incollate il seguente contenuto al suo interno.

```
{
  "app": "node lib/docker-tutorial.js"
}
```

- Apri il `lib/docker-tutorial.js` file e incolla il seguente contenuto al suo interno.

```
// this file defines the CDK constructs we want to deploy

const { App, Stack } = require('aws-cdk-lib');
const { DockerImageFunction, DockerImageCode } = require('aws-cdk-lib/aws-lambda');
const path = require('path');

// create an application
const app = new App();

// define stack
class DockerTutorialStack extends Stack {
  constructor(scope, id, props) {
    super(scope, id, props);

    // define lambda that uses a Docker container
    const dockerfileDir = path.join(__dirname);
    new DockerImageFunction(this, 'DockerTutorialFunction', {
      code: DockerImageCode.fromImageAsset(dockerfileDir),
      functionName: 'DockerTutorialFunction',
```

```
    });  
  }  
}  
  
// instantiate stack  
new DockerTutorialStack(app, 'DockerTutorialStack');
```

7. Apri `lib/Dockerfile` e incolla il seguente contenuto al suo interno.

```
# Use a NodeJS 20.x runtime  
FROM public.ecr.aws/lambda/nodejs:20  
  
# Copy the function code to the LAMBDA_TASK_ROOT directory  
# This environment variable is provided by the lambda base image  
COPY hello.js ${LAMBDA_TASK_ROOT}  
  
# Set the CMD to the function handler  
CMD [ "hello.handler" ]
```

8. Apri il `lib/hello.js` file e incolla il seguente contenuto al suo interno.

```
// define the handler  
exports.handler = async (event) => {  
  // simply return a friendly success response  
  const response = {  
    statusCode: 200,  
    body: JSON.stringify('Hello, World!'),  
  };  
  return response;  
};
```

9. Utilizza la AWS CDK CLI per sintetizzare il progetto e distribuire le risorse. Devi avviare il tuo account.

```
npx cdk synth  
npx cdk deploy --require-approval never
```

10. Invoca la funzione Lambda per confermarla e verificarla.

```
aws lambda invoke --function-name DockerTutorialFunction out.json  
jq . out.json
```

Ora hai distribuito con successo una funzione Lambda basata su container Docker utilizzando AWS CDK. [Per ulteriori informazioni su AWS CDK, consulta la Guida per sviluppatori v2. AWS CDK](#) Se riscontri errori o riscontri problemi durante il tentativo di completare questo tutorial, consulta la sezione [Risoluzione dei problemi](#) di questa guida per ricevere assistenza.

## Eliminazione

Ora hai distribuito con successo una funzione Lambda basata su container Docker utilizzando AWS CDK. All'interno del AWS CDK progetto, esegui il comando seguente per eliminare le risorse associate. Ti verrà richiesto di confermare l'eliminazione.

- ```
npx cdk destroy DockerTutorialStack
```
- Per rimuovere i file e le risorse che hai creato in questo tutorial dal tuo AWS CloudShell ambiente, esegui il comando seguente.

```
cd ~  
rm -rf ~/docker-cli-tutorial
```

# Utilizzo di AWS CloudShell

Questa sezione descrive come interagire AWS CloudShell ed eseguire azioni specifiche con le applicazioni supportate.

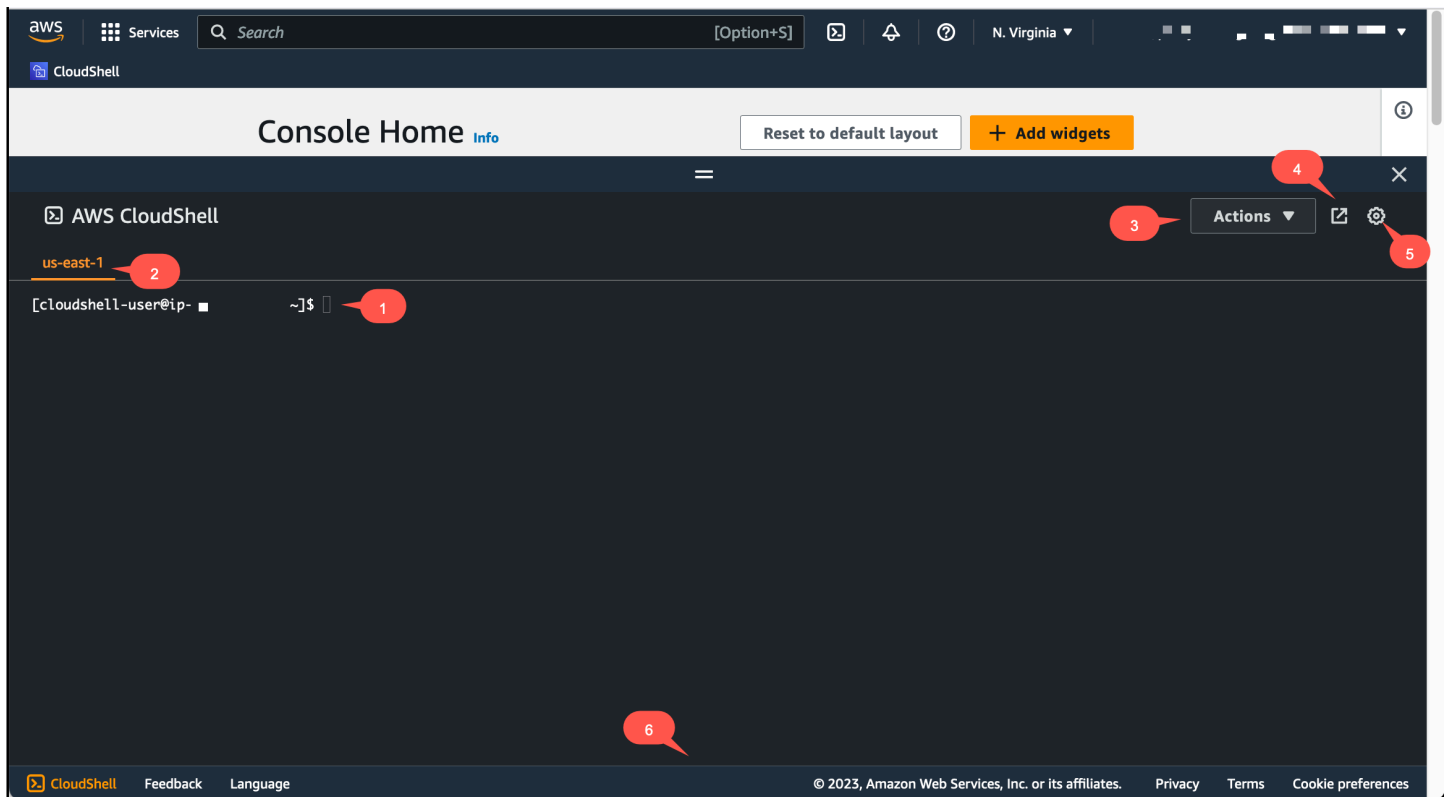
## Argomenti

- [Navigazione nell'interfaccia AWS CloudShell](#)
- [Lavorare in Regioni AWS](#)
- [Utilizzo dei file e dello spazio di archiviazione](#)
- [Utilizzo di Docker](#)


## Navigazione nell'interfaccia AWS CloudShell

È possibile navigare tra le funzionalità CloudShell dell'interfaccia da AWS Management Console e Console Toolbar.

La schermata seguente indica diverse funzionalità chiave AWS CloudShell dell'interfaccia.




1. AWS CloudShellinterfaccia a riga di comando utilizzata per eseguire comandi utilizzando [la shell preferita](#). Il tipo di shell corrente è indicato dal prompt dei comandi.
2. La scheda del terminale, che utilizza Regione AWS where, AWS CloudShell è attualmente in esecuzione.
3. Il menu Azioni, che fornisce opzioni per [modificare il layout dello schermo](#), [scaricare](#) e [caricare](#) file, [riavviare ed eliminare la AWS CloudShell](#) home directory. AWS CloudShell

 Note

L'opzione Download non è disponibile all'avvio di. CloudShell Console Toolbar

4. La scheda Apri in un nuovo browser, che offre la possibilità di accedere alla CloudShell sessione a schermo intero.
5. L'opzione Preferenze, che puoi usare per [personalizzare la tua esperienza con la shell](#).
6. La barra inferiore, che fornisce le seguenti opzioni per:
  - Avvia CloudShell dall'CloudShellicona.
  - Fornisci feedback dall'icona Feedback. Scegli il tipo di feedback che desideri inviare, aggiungi i commenti e quindi scegli Invia.
  - Per inviare un feedback CloudShell, scegli una delle seguenti opzioni:
    - Dalla console CloudShell, avvia e scegli Feedback. Aggiungi i commenti, quindi scegli Invia.
    - Scegli CloudShellnella parte inferiore sinistra della consoleConsole Toolbar, quindi scegli l'icona Apri in una nuova scheda del browser, Feedback. Aggiungi i commenti, quindi scegli Invia.

 Note

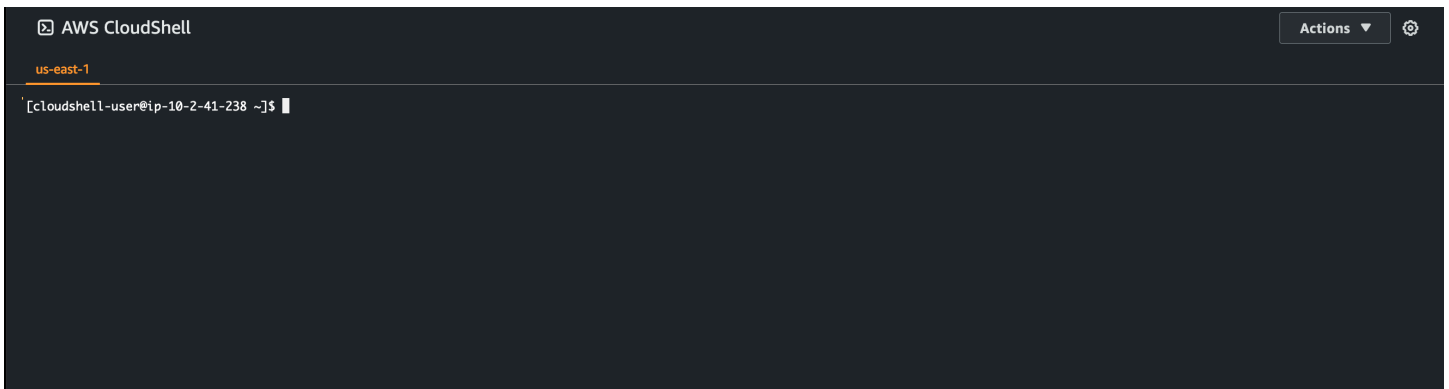
L'opzione Feedback non è disponibile CloudShell all'avvio diConsole Toolbar.

- Scopri la nostra politica sulla privacy e i termini di utilizzo e personalizza le preferenze sui cookie.



# Lavorare in Regioni AWS

La corrente in Regione AWS cui stai eseguendo viene visualizzata sopra l'interfaccia della riga di comando.



```
AWS CloudShell
us-east-1
[cloudshell-user@ip-10-2-41-238 ~]$
```

Puoi scegliere un ambiente in cui Regione AWS lavorare selezionando una regione specifica utilizzando il selettore della regione. Dopo aver modificato le regioni, l'interfaccia si aggiorna man mano che la sessione di shell si connette a un ambiente di calcolo diverso in esecuzione nella regione selezionata.

## Important

Puoi utilizzare fino a 1 GB di spazio di archiviazione persistente in ciascuna. Regione AWS L'archiviazione persistente è memorizzata nella tua home directory (\$HOME). Ciò significa che tutti i file, le directory, i programmi o gli script personali archiviati nella home directory si trovano tutti in un'unica cartella. Regione AWS Inoltre, sono diversi da quelli che si trovano nella home directory e sono archiviati in una regione diversa.

La conservazione a lungo termine dei file nell'archiviazione persistente viene gestita anche per regione. Per ulteriori informazioni, consulta [Storage persistente](#).

## Specificare l'impostazione predefinita per Regione AWS AWS CLI

È possibile utilizzare [le variabili di ambiente](#) per specificare le opzioni di configurazione e le credenziali necessarie per l'accesso tramite Servizi AWS. AWS CLI La variabile di ambiente che specifica l'impostazione predefinita Regione AWS per la sessione di shell viene impostata quando si avvia AWS CloudShell da una regione specifica di AWS Management Console o quando si sceglie un'opzione nel selettore di regione.

[Le variabili di ambiente hanno la precedenza sui file di AWS CLI credenziali che vengono aggiornati da. aws configure](#) Pertanto, non è possibile eseguire il `aws configure` comando per modificare la regione specificata dalla variabile di ambiente. Invece, per modificare la regione predefinita per AWS CLI i comandi, assegnate un valore alla variabile di `AWS_REGION` ambiente. Negli esempi che seguono, sostituiscila `us-east-1` con la regione in cui ti trovi.

## Bash or Zsh

```
$ export AWS_REGION=us-east-1
```

L'impostazione della variabile di ambiente modifica il valore utilizzato fino alla fine della sessione di shell o all'impostazione della variabile su un valore diverso. Puoi impostare variabili nello script di avvio della shell per rendere le variabili persistenti nelle sessioni future.

## PowerShell

```
PS C:\> $Env:AWS_REGION="us-east-1"
```

Se impostate una variabile di ambiente al PowerShell prompt, la variabile di ambiente salva il valore solo per la durata della sessione corrente. In alternativa, puoi impostare la variabile per tutte le PowerShell sessioni future aggiungendo la variabile al tuo PowerShell profilo. Per ulteriori informazioni sulla memorizzazione delle variabili di ambiente, consulta la [PowerShell documentazione](#).

Per confermare di aver modificato la regione predefinita, esegui il `aws configure list` comando per visualizzare i dati di AWS CLI configurazione correnti.

### Note

Per AWS CLI comandi specifici, puoi sovrascrivere la regione predefinita utilizzando l'opzione `--region` della riga di comando. Per ulteriori informazioni, consulta [Opzioni della riga di comando](#) nella Guida per l'AWS Command Line Interface.

## Utilizzo dei file e dello spazio di archiviazione

Utilizzando AWS CloudShell l'interfaccia, è possibile caricare e scaricare file dall'ambiente shell. Per ulteriori informazioni sul download e il caricamento di file, consulta [Guida introduttiva AWS CloudShell](#).

Per garantire che tutti i file aggiunti siano disponibili al termine della sessione, è necessario conoscere la differenza tra l'archiviazione persistente e quella temporanea.

- Archiviazione persistente: hai a disposizione 1 GB di spazio di archiviazione persistente per ciascuno Regione AWS. L'archiviazione persistente si trova nella tua home directory.
- Archiviazione temporanea: l'archiviazione temporanea viene riciclata al termine di una sessione. L'archiviazione temporanea si trova nelle directory esterne alla home directory.

### Important

Assicurati di lasciare i file che desideri conservare e utilizzare per le future sessioni di shell nella tua home directory. Ad esempio, supponete di spostare un file fuori dalla vostra home directory eseguendo il mv comando. Quindi, quel file viene riciclato al termine della sessione di shell corrente.

## Utilizzo di Docker

AWS CloudShell supporta completamente Docker senza installazione o configurazione. Puoi definire, creare ed eseguire contenitori Docker all'interno. AWS CloudShell Puoi distribuire risorse basate su Docker, come le funzioni Lambda basate su contenitori Docker, tramite il AWS CDK Toolkit, nonché creare contenitori Docker e inviarli ai repository Amazon ECR tramite la CLI Docker. Per i passaggi dettagliati su come eseguire entrambe queste distribuzioni, consulta i seguenti tutorial:

- [Tutorial: distribuzione di una funzione Lambda utilizzando AWS CDK](#)
- [Tutorial: creazione di un contenitore Docker all'interno AWS CloudShell e trasferimento in un repository Amazon ECR](#)

Esistono alcune restrizioni e limitazioni nell'uso di Docker con: AWS CloudShell

- Docker ha uno spazio limitato in un ambiente. Se hai immagini singole di grandi dimensioni o troppe immagini Docker preesistenti, ciò può causare problemi che potrebbero impedirti di estrarre, creare o eseguire immagini aggiuntive. [Per ulteriori informazioni su Docker, consulta la guida alla documentazione Docker.](#)
- Docker è supportato solo in alcune regioni. Per informazioni su quali regioni sono supportate con Docker, consulta [Docker](#) Regions.
- Se riscontri problemi durante l'utilizzo di Docker con AWS CloudShell, consulta la sezione [Risoluzione dei problemi](#) di questa guida per informazioni su come risolvere potenzialmente questi problemi.

# Utilizzo delle funzionalità di accessibilità per AWS CloudShell

Questo argomento descrive come utilizzare le funzionalità di accessibilità per CloudShell. È possibile utilizzare una tastiera per navigare tra gli elementi focalizzabili della pagina. È inoltre possibile personalizzare l'aspetto di CloudShell, comprese le dimensioni dei caratteri e i temi dell'interfaccia.

## Navigazione tramite tastiera in CloudShell

Per navigare tra gli elementi focalizzabili della pagina, premi **Tab**.

## CloudShell funzionalità di accessibilità dei terminali

È possibile utilizzare il **Tab** tasto nelle seguenti modalità:

- modalità terminale (impostazione predefinita)— In questa modalità, il terminale acquisisce il **Tab** inserimento chiave. Dopo che l'attenzione si è concentrata sul terminale, premi **Tab** per accedere solo alle funzionalità del terminale.
- modalità di navigazione— In questa modalità, il terminale non acquisisce il tuo **Tab** inserimento chiave. Premere **Tab** per navigare tra gli elementi focalizzabili della pagina.

Per passare dalla modalità terminale alla modalità navigazione, premere **Ctrl+M**. Dopo essere tornato indietro, **Scheda: navigazione** appare nell'intestazione ed è possibile utilizzare il **Tab** tasto per navigare nella pagina.

Per tornare alla modalità terminale, premere **Ctrl+M**. Oppure scegli **X** accanto a **Scheda: navigazione**.

### Note

Attualmente, CloudShell le funzionalità di accessibilità dei terminali non sono disponibili sui dispositivi mobili.

## Scelta delle dimensioni dei caratteri e dei temi dell'interfaccia in CloudShell

È possibile personalizzare l'aspetto di CloudShell per soddisfare le tue preferenze visive.

- Dimensione del carattere— Scegli tra il più piccolo, Piccolo, Medio, Grande, e il più grande dimensioni dei caratteri nel terminale. Per ulteriori informazioni sulla modifica della dimensione dei caratteri, vedere [the section called “Modifica della dimensione del carattere”](#).
- Tema— Scegli tra Leggero e Scuro temi di interfaccia. Per ulteriori informazioni sulla modifica del tema dell'interfaccia, vedere [the section called “Modifica del tema dell'interfaccia”](#).

# Lavorare con AWS i servizi in AWS CloudShell

Uno dei principali vantaggi AWS CloudShell è che puoi usarlo per gestire i tuoi AWS servizi dall'interfaccia a riga di comando. Ciò significa che non devi prima scaricare e installare strumenti o configurare le credenziali localmente. All'avvio AWS CloudShell, viene creato un ambiente di elaborazione in cui sono già installati i seguenti strumenti da riga di comando:

- [AWS CLI](#)
- [AWS Elastic Beanstalk CLI](#)
- [CLI di Amazon ECS](#)
- [AWS SAM](#)

E poiché hai già effettuato l'accesso AWS, non è necessario configurare le credenziali localmente prima di utilizzare i servizi. Le credenziali utilizzate per accedere a AWS Management Console vengono inoltrate a AWS CloudShell.

Se si desidera modificare la AWS regione predefinita utilizzata per AWS CLI, è possibile modificare il valore assegnato alla variabile di ambiente `AWS_REGION`. Per ulteriori informazioni, consult [Specificare l'impostazione predefinita per Regione AWS AWS CLI.](#))

Il resto di questo argomento dimostra come iniziare AWS CloudShell a utilizzare per interagire con AWS servizi selezionati dalla riga di comando.

## AWS CLI esempi da riga di comando per AWS servizi selezionati

Gli esempi seguenti rappresentano solo alcuni dei numerosi AWS servizi con cui è possibile lavorare utilizzando i comandi disponibili AWS CLI nella versione 2. Per un elenco completo, consulta l'[AWS CLI Command Reference](#).

- [DynamoDB](#)
- [AWS Cloud9](#)
- [Amazon EC2](#)
- [Ghiacciaio S3](#)

## DynamoDB

DynamoDB è un servizio di database NoSQL interamente gestito che combina prestazioni elevate e prevedibili con una scalabilità continua. L'implementazione della modalità NoSQL da parte di questo servizio supporta strutture di dati chiave-valori e documentali.

Il `create-table` comando seguente crea una tabella in stile NoSQL denominata `MusicCollection` nel tuo AWS account.

```
aws dynamodb create-table \  
  --table-name MusicCollection \  
  --attribute-definitions AttributeName=Artist,AttributeType=S \  
  AttributeName=SongTitle,AttributeType=S \  
  --key-schema AttributeName=Artist,KeyType=HASH \  
  AttributeName=SongTitle,KeyType=RANGE \  
  --provisioned-throughput ReadCapacityUnits=5,WriteCapacityUnits=5 \  
  --tags Key=Owner,Value=blueTeam
```

Per ulteriori informazioni, vedere [Uso di DynamoDB con la AWS CLI](#) nella Guida per l'AWS Command Line Interface utente.

## AWS Cloud9

AWS Cloud9 è un ambiente di sviluppo integrato (IDE) basato su cloud che consente di scrivere, eseguire ed effettuare il debug del proprio codice in una finestra del browser. L'ambiente include un editor di codice, un debugger e un terminale.

Il `create-environment-ec2` comando seguente crea un ambiente di sviluppo AWS Cloud9 EC2 con le impostazioni specificate. Avvia un'istanza Amazon EC2, quindi si connette dall'istanza all'ambiente.

```
aws cloud9 create-environment-ec2 --name my-demo-env --description "My demonstration development environment." --instance-type t2.micro --subnet-id subnet-1fab8aEX --automatic-stop-time-minutes 60 --owner-arn arn:aws:iam::123456789012:user/MyDemoUser
```

Per ulteriori informazioni, consulta la sezione [AWS Cloud9 relativa alle informazioni di comando](#).



## Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) è un servizio Web che offre capacità di calcolo sicure e ridimensionabili nel cloud. È concepito per rendere più accessibili e accessibili cloud computing su grande scala per il Web.

Il `run-instances` comando seguente avvia un'istanza `t2.micro` nella sottorete specificata di un VPC:

```
aws ec2 run-instances --image-id ami-xxxxxxx --count 1 --instance-type t2.micro --key-name MyKeyPair --security-group-ids sg-903004f8 --subnet-id subnet-6e7f829e
```

Per ulteriori informazioni, consulta [Utilizzo di Amazon EC2 con AWS CLI nella Guida per l'AWS Command Line Interface](#).

## S3 Glacier

S3 Glacier e S3 Glacier Deep Archive sono classi di cloud storage Amazon S3

Il `create-vault` comando seguente crea un archivio, un contenitore per l'archiviazione degli archivi:

```
aws glacier create-vault --vault-name my-vault --account-id -
```

Per ulteriori informazioni, consulta [Utilizzo di Amazon S3 Glacier con AWS CLI nella Guida per l'AWS Command Line Interface](#).

## AWSCLI di Elastic Beanstalk

La AWS Elastic Beanstalk CLI offre un'interfaccia a riga di comando per semplificare la creazione, l'aggiornamento e il monitoraggio di ambienti da un repository locale. In questo contesto, un ambiente si riferisce a una raccolta di AWS risorse che eseguono una versione dell'applicazione.

Il `create` comando seguente crea un nuovo ambiente in un Amazon Virtual Private Cloud (VPC) personalizzato.

```
$ eb create dev-vpc --vpc.id vpc-0ce8dd99 --vpc.elbsubnets subnet-b356d7c6,subnet-02f74b0c --vpc.ec2subnets subnet-0bb7f0cd,subnet-3b6697c1 --vpc.securitygroup sg-70cff265
```

Per ulteriori informazioni, consulta il [riferimento al comando EB CLI](#) nella Guida per gli AWS Elastic Beanstalk sviluppatori.

## CLI di Amazon ECS

Elastic Container Service (Amazon ECS) riga di comando (CLI) Sono progettati per semplificare i processi di creazione, aggiornamento e monitoraggio di cluster e processi in un ambiente di sviluppo locale. (Un cluster Amazon ECS è un raggruppamento logico di processi o servizi.)

Il `configure` comando seguente configura l'interfaccia a riga di comando di Amazon ECS per creare una configurazione del cluster denominata `ecs-cli-demo`. Questa configurazione del cluster utilizza `FARGATE` come tipo di avvio predefinito per il `ecs-cli-demo` cluster in `us-east-1` region.

```
ecs-cli configure --region us-east-1 --cluster ecs-cli-demo --default-launch-type
FARGATE --config-name ecs-cli-demo
```

Per ulteriori informazioni, consulta [Guida di riferimento alla riga di comando di Amazon ECS](#) nella Guida per gli sviluppatori di Amazon Elastic Container Service.

## CLI AWS SAM

AWS SAM La CLI è uno strumento a riga di comando che funziona su un AWS Serverless Application Model modello e su un codice di applicazione. È possibile eseguire diverse attività utilizzandolo. Questi includono l'invocazione locale delle funzioni Lambda, la creazione di un pacchetto di distribuzione per l'applicazione serverless e la distribuzione dell'applicazione serverless nel AWS cloud.

Il `init` comando seguente inizializza un nuovo progetto SAM con i parametri richiesti passati come parametri:

```
sam init --runtime python3.7 --dependency-manager pip --app-template hello-world --name
sam-app
```

Per ulteriori informazioni, consulta il [riferimento al comando AWS SAM CLI](#) nella Guida per gli AWS Serverless Application Model sviluppatori.

# Personalizzare ilAWS CloudShellesperienza

Puoi personalizzare i seguenti aspetti del tuoAWS CloudShellesperienza:


- [Layout delle schede](#): divide l'interfaccia della riga di comando in più colonne e righe.
- [Dimensione del carattere](#): regola la dimensione del testo della riga di comando.
- [Tema a colori](#): Passa dal tema chiaro a quello scuro.
- [Incolla in modo sicuro](#): attiva o disattiva una funzionalità che richiede la verifica del testo su più righe prima di incollarlo.
- [Da Tmux al ripristino della sessione](#): L'uso di tmux ripristina la sessione fino a quando la sessione non diventa inattiva.

Puoi anche estendere il tuo ambiente shell tramite[installando il proprio software](#)e[modifica degli script della shell di avvio](#).

## Suddivisione della visualizzazione della riga di comando in più schede

Esegui più comandi suddividendo l'interfaccia della riga di comando in più riquadri.

### Note

Dopo aver aperto più schede, puoi selezionarne una su cui desideri lavorare facendo clic in un punto qualsiasi del riquadro di tua scelta. Puoi chiudere una scheda scegliendo il  simbolo, che si trova accanto al nome della regione.

- Scegli **Azione** una delle seguenti opzioni di **Layout delle schede**:
  - Nuova scheda: Aggiungi una nuova scheda accanto a quella attualmente attiva.
  - Dividi in righe: Aggiungi una nuova scheda in una riga inferiore a quella attualmente attiva.
  - Dividere in colonne: Aggiungi una nuova scheda in una colonna accanto a quella attualmente attiva.

Se non c'è abbastanza spazio per visualizzare completamente ogni scheda, scorri per visualizzare l'intera scheda. Puoi anche selezionare le barre di divisione che separano i riquadri e trascinarle utilizzando il puntatore per aumentare o ridurre le dimensioni del riquadro.

## Modifica della dimensione del carattere

Aumenta o diminuisci la dimensione del testo visualizzato nell'interfaccia della riga di comando.

1. Per modificare ilAWS CloudShellimpostazioni del terminale, vai aimpostazioni,Preferenze.
2. Scegli la dimensione del testo. Le tue opzioni sonoIl più piccolo,Piccolo,Medio,Grande, ell più grande.

## Modifica del tema dell'interfaccia

Passa dal tema chiaro a quello scuro per l'interfaccia della riga di comando.

1. Per modificare ilAWS CloudShelltema, vai almpostazioni,Preferenze.
2. ScegliLeggerooScuro.

## Utilizzo di Safe Paste per testo su più righe

Safe Paste è una funzionalità di sicurezza che richiede di verificare che il testo multilinea che state per incollare nella shell non contenga script dannosi. Il testo copiato da siti di terze parti può contenere codice nascosto che attiva comportamenti imprevisti nell'ambiente della shell.

La finestra di dialogo Safe Paste mostra il testo completo che hai copiato negli appunti. Se ritieni che non vi siano rischi per la sicurezza, scegliIncolla.

**Warning: Pasting multiline text into AWS CloudShell**

Text that's copied from external sources can contain malicious scripts. Verify the text below before pasting.

```
import sys
x=int(sys.argv[1])
y=int(sys.argv[2])
z=int(sys.argv[3])
total=x+y+z
print("The total is",total)
```

Always ask before pasting multiline code

Cancel

Paste

Ti consigliamo di abilitare Safe Paste per rilevare potenziali rischi per la sicurezza negli script. Puoi attivare o disattivare questa funzionalità scegliendo **Preferenze**, **Abilita Safe Paste** / **Disattiva Safe Paste**.

## Usando tmux per ripristinare la sessione

AWS CloudShell utilizza tmux per ripristinare le sessioni su una o più schede del browser. Se aggiorni le schede del browser, la sessione riprende fino a quando la sessione non diventa inattiva. Per ulteriori informazioni, vedere [Ripristino della sessione](#).

# Sicurezza per AWS CloudShell

La sicurezza cloud di Amazon Web Services (AWS) è la priorità più alta. In qualità di AWS cliente, puoi trarre vantaggio da un data center e da un'architettura di rete progettati per soddisfare i requisiti delle organizzazioni più sensibili alla sicurezza. La sicurezza è una responsabilità condivisa tra AWS e te. Il [modello di responsabilità condivisa](#) descrive questo come sicurezza del cloud e sicurezza nel cloud.

Security of the Cloud: AWS è responsabile della protezione dell'infrastruttura che gestisce tutti i servizi offerti nel AWS Cloud e della fornitura di servizi che è possibile utilizzare in modo sicuro. La nostra responsabilità in AWS materia di sicurezza è la massima priorità e l'efficacia della nostra sicurezza viene regolarmente testata e verificata da revisori di terze parti nell'ambito dei Programmi di [AWS conformità](#).

Sicurezza nel cloud: la responsabilità dell'utente è determinata dal AWS servizio utilizzato e da altri fattori, tra cui la sensibilità dei dati, i requisiti dell'organizzazione e le leggi e i regolamenti applicabili.

AWS CloudShell segue il [modello di responsabilità condivisa](#) attraverso i AWS servizi specifici che supporta. Per informazioni sulla sicurezza dei AWS servizi, consulta la [pagina della documentazione sulla sicurezza del AWS servizio](#) e [AWS i servizi che rientrano nell'ambito delle iniziative di AWS conformità previste dal programma di conformità](#).

Negli argomenti seguenti viene illustrato come eseguire la configurazione AWS CloudShell per soddisfare gli obiettivi di sicurezza e conformità.

## Argomenti

- [Protezione dei dati in AWS CloudShell](#)
- [Identity and Access Management per AWS CloudShell](#)
- [Registrazione e monitoraggio AWS CloudShell](#)
- [Convalida della conformità per AWS CloudShell](#)
- [Resilienza in AWS CloudShell](#)
- [Sicurezza dell'infrastruttura in AWS CloudShell](#)
- [Analisi della configurazione e delle vulnerabilità in AWS CloudShell](#)
- [Best practice di sicurezza per AWS CloudShell](#)
- [AWS CloudShell Domande frequenti sulla sicurezza](#)

# Protezione dei dati in AWS CloudShell

Il modello di [responsabilità AWS condivisa modello](#) di si applica alla protezione dei dati in AWS CloudShell. Come descritto in questo modello, AWS è responsabile della protezione dell'infrastruttura globale che gestisce tutti i Cloud AWS. L'utente è responsabile del controllo dei contenuti ospitati su questa infrastruttura. Inoltre, sei responsabile della configurazione della protezione e delle attività di gestione per i Servizi AWS che utilizzi. Per ulteriori informazioni sulla privacy dei dati, vedi [Domande frequenti sulla privacy dei dati](#). Per informazioni sulla protezione dei dati in Europa, consulta il post del blog [AWS Shared Responsibility Model and GDPR](#) nel Blog sulla sicurezza AWS .

Ai fini della protezione dei dati, consigliamo di proteggere Account AWS le credenziali e configurare i singoli utenti con AWS IAM Identity Center or AWS Identity and Access Management (IAM). In tal modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere i suoi compiti. Ti suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- Usa SSL/TLS per comunicare con le risorse. AWS È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Configura l'API e la registrazione delle attività degli utenti con. AWS CloudTrail
- Utilizza soluzioni di AWS crittografia, insieme a tutti i controlli di sicurezza predefiniti all'interno Servizi AWS.
- Utilizza i servizi di sicurezza gestiti avanzati, come Amazon Macie, che aiutano a individuare e proteggere i dati sensibili archiviati in Amazon S3.
- Se hai bisogno di moduli crittografici convalidati FIPS 140-2 per l'accesso AWS tramite un'interfaccia a riga di comando o un'API, utilizza un endpoint FIPS. Per ulteriori informazioni sugli endpoint FIPS disponibili, consulta il [Federal Information Processing Standard \(FIPS\) 140-2](#).

Ti consigliamo vivamente di non inserire mai informazioni riservate o sensibili, ad esempio gli indirizzi e-mail dei clienti, nei tag o nei campi di testo in formato libero, ad esempio nel campo Nome. Ciò include quando lavori o Servizi AWS utilizzi la console, l'API AWS CloudShell o gli SDK. AWS CLI AWS I dati inseriti nei tag o nei campi di testo in formato libero utilizzati per i nomi possono essere utilizzati per i la fatturazione o i log di diagnostica. Quando fornisci un URL a un server esterno, ti suggeriamo vivamente di non includere informazioni sulle credenziali nell'URL per convalidare la tua richiesta al server.

## Crittografia dei dati

La crittografia dei dati si riferisce alla protezione dei dati quando sono inattivi (mentre sono archiviati AWS CloudShell) e quando sono in transito (mentre viaggiano tra gli endpoint di servizio AWS CloudShell e gli endpoint).

### Utilizzo della crittografia a riposo AWS KMS

La crittografia dei dati inattivi si riferisce alla protezione dei dati da accessi non autorizzati crittografando i dati durante l'archiviazione. Quando si utilizza AWS CloudShell, si dispone di uno spazio di archiviazione persistente di 1 GB per AWS regione senza alcun costo. Lo spazio di archiviazione persistente si trova nella tua home directory (\$HOME) ed è privato. A differenza delle risorse ambientali temporanee che vengono riciclate al termine di ogni sessione di shell, i dati nella home directory persistono.

La crittografia dei dati archiviati in AWS CloudShell viene implementata utilizzando le chiavi crittografiche fornite da (). AWS Key Management Service AWS KMS Si tratta di un AWS servizio gestito per la creazione e il controllo delle chiavi master dei clienti (CMK), le chiavi di crittografia utilizzate per crittografare i dati dei clienti archiviati nell'ambiente. AWS CloudShell AWS CloudShell genera e gestisce chiavi crittografiche per crittografare i dati per conto dei clienti.

### Crittografia in transito

La crittografia in transito si riferisce alla protezione dei dati da qualsiasi intercettazione mentre si spostano tra gli endpoint di comunicazione.

Per impostazione predefinita, tutte le comunicazioni di dati tra il browser Web del client e il computer basato sul cloud sono crittografate inviando tutto tramite AWS CloudShell una connessione HTTPS/TLS.

Non è necessario fare nulla per abilitare l'uso di HTTPS/TLS per la comunicazione.

## Identity and Access Management per AWS CloudShell

AWS Identity and Access Management (IAM) è un software Servizio AWS che aiuta un amministratore a controllare in modo sicuro l'accesso alle AWS risorse. Gli amministratori IAM controllano chi può essere autenticato (effettuato l'accesso) e autorizzato (disporre delle



autorizzazioni) a utilizzare le risorse. CloudShell IAM è uno Servizio AWS strumento che puoi utilizzare senza costi aggiuntivi.

## Argomenti

- [Destinatari](#)
- [Autenticazione con identità](#)
- [Gestione dell'accesso con policy](#)
- [In che modo AWS CloudShell funziona con IAM](#)
- [Esempi di policy basate sull'identità per AWS CloudShell](#)
- [Risoluzione dei problemi relativi all' CloudShell identità e all'accesso di AWS](#)
- [Gestione dell' AWS CloudShell accesso e dell'utilizzo con le policy IAM](#)

## Destinatari

Il modo in cui usi AWS Identity and Access Management (IAM) varia a seconda del lavoro che CloudShell svolgi.

Utente del servizio: se utilizzi il CloudShell servizio per svolgere il tuo lavoro, l'amministratore ti fornisce le credenziali e le autorizzazioni necessarie. Man mano che utilizzi più CloudShell funzionalità per svolgere il tuo lavoro, potresti aver bisogno di autorizzazioni aggiuntive. La comprensione della gestione dell'accesso ti consente di richiedere le autorizzazioni corrette all'amministratore. Se non riesci ad accedere a una funzionalità di CloudShell, consulta [Risoluzione dei problemi relativi all' CloudShell identità e all'accesso di AWS](#).

Amministratore del servizio: se sei responsabile delle CloudShell risorse della tua azienda, probabilmente hai pieno accesso a CloudShell. È tuo compito determinare a quali CloudShell funzionalità e risorse devono accedere gli utenti del servizio. Devi inviare le richieste all'amministratore IAM per cambiare le autorizzazioni degli utenti del servizio. Esamina le informazioni contenute in questa pagina per comprendere i concetti di base relativi a IAM. Per saperne di più su come la tua azienda può utilizzare IAM con CloudShell, consulta [In che modo AWS CloudShell funziona con IAM](#).

Amministratore IAM: un amministratore IAM potrebbe essere interessato a ottenere dei dettagli su come scrivere policy per gestire l'accesso a CloudShell. Per visualizzare esempi di policy CloudShell basate sull'identità che puoi utilizzare in IAM, consulta. [Esempi di policy basate sull'identità per AWS CloudShell](#)

## Autenticazione con identità

L'autenticazione è il modo in cui accedi AWS utilizzando le tue credenziali di identità. Devi essere autenticato (aver effettuato l'accesso root dell'account AWS) come utente IAM o assumendo un ruolo IAM.

Puoi accedere AWS come identità federata utilizzando le credenziali fornite tramite una fonte di identità. AWS IAM Identity Center (IAM Identity Center), l'autenticazione Single Sign-On della tua azienda e le tue credenziali di Google o Facebook sono esempi di identità federate. Se accedi come identità federata, l'amministratore ha configurato in precedenza la federazione delle identità utilizzando i ruoli IAM. Quando accedi AWS utilizzando la federazione, assumi indirettamente un ruolo.

A seconda del tipo di utente, puoi accedere al AWS Management Console o al portale di AWS. Per ulteriori informazioni sull'accesso a AWS, vedi [Come accedere al tuo Account AWS](#) nella Guida per l'Accedi ad AWS utente.

Se accedi a AWS livello di codice, AWS fornisce un kit di sviluppo software (SDK) e un'interfaccia a riga di comando (CLI) per firmare crittograficamente le tue richieste utilizzando le tue credenziali. Se non utilizzi AWS strumenti, devi firmare tu stesso le richieste. Per ulteriori informazioni sull'utilizzo del metodo consigliato per firmare autonomamente le richieste, consulta [Signing AWS API request](#) nella IAM User Guide.

A prescindere dal metodo di autenticazione utilizzato, potrebbe essere necessario specificare ulteriori informazioni sulla sicurezza. Ad esempio, ti AWS consiglia di utilizzare l'autenticazione a più fattori (MFA) per aumentare la sicurezza del tuo account. Per ulteriori informazioni, consulta [Autenticazione a più fattori](#) nella Guida per l'utente di AWS IAM Identity Center e [Utilizzo dell'autenticazione a più fattori \(MFA\) in AWS](#) nella Guida per l'utente di IAM.

### Account AWS utente root

Quando si crea un account AWS, si inizia con un'identità di accesso che ha accesso completo a tutti i Servizi AWS e le risorse dell'account. Questa identità è denominata utente Account AWS root ed è accessibile effettuando l'accesso con l'indirizzo e-mail e la password utilizzati per creare l'account. Si consiglia vivamente di non utilizzare l'utente root per le attività quotidiane. Conserva le credenziali dell'utente root e utilizzarle per eseguire le operazioni che solo l'utente root può eseguire. Per un elenco completo delle attività che richiedono l'accesso come utente root, consulta la sezione [Attività che richiedono le credenziali dell'utente root](#) nella Guida per l'utente di IAM.

## Identità federata

Come procedura consigliata, richiedi agli utenti umani, compresi gli utenti che richiedono l'accesso come amministratore, di utilizzare la federazione con un provider di identità per accedere Servizi AWS utilizzando credenziali temporanee.

Un'identità federata è un utente dell'elenco utenti aziendale, di un provider di identità Web AWS Directory Service, della directory Identity Center o di qualsiasi utente che accede utilizzando le Servizi AWS credenziali fornite tramite un'origine di identità. Quando le identità federate accedono Account AWS, assumono ruoli e i ruoli forniscono credenziali temporanee.

Per la gestione centralizzata degli accessi, consigliamo di utilizzare AWS IAM Identity Center. Puoi creare utenti e gruppi in IAM Identity Center oppure puoi connetterti e sincronizzarti con un set di utenti e gruppi nella tua fonte di identità per utilizzarli su tutte le tue applicazioni. Account AWS Per ulteriori informazioni sul Centro identità IAM, consulta [Cos'è Centro identità IAM?](#) nella Guida per l'utente di AWS IAM Identity Center .

## Utenti e gruppi IAM

Un [utente IAM](#) è un'identità interna Account AWS che dispone di autorizzazioni specifiche per una singola persona o applicazione. Ove possibile, consigliamo di fare affidamento a credenziali temporanee invece di creare utenti IAM con credenziali a lungo termine come le password e le chiavi di accesso. Tuttavia, per casi d'uso specifici che richiedono credenziali a lungo termine con utenti IAM, si consiglia di ruotare le chiavi di accesso. Per ulteriori informazioni, consulta la pagina [Rotazione periodica delle chiavi di accesso per casi d'uso che richiedono credenziali a lungo termine](#) nella Guida per l'utente di IAM.

Un [gruppo IAM](#) è un'identità che specifica un insieme di utenti IAM. Non è possibile eseguire l'accesso come gruppo. È possibile utilizzare gruppi per specificare le autorizzazioni per più utenti alla volta. I gruppi semplificano la gestione delle autorizzazioni per set di utenti di grandi dimensioni. Ad esempio, è possibile avere un gruppo denominato Amministratori IAM e concedere a tale gruppo le autorizzazioni per amministrare le risorse IAM.

Gli utenti sono diversi dai ruoli. Un utente è associato in modo univoco a una persona o un'applicazione, mentre un ruolo è destinato a essere assunto da chiunque ne abbia bisogno. Gli utenti dispongono di credenziali a lungo termine permanenti, mentre i ruoli forniscono credenziali temporanee. Per ulteriori informazioni, consulta [Quando creare un utente IAM \(invece di un ruolo\)](#) nella Guida per l'utente di IAM.

## Ruoli IAM

Un [ruolo IAM](#) è un'identità interna all'utente Account AWS che dispone di autorizzazioni specifiche. È simile a un utente IAM, ma non è associato a una persona specifica. Puoi assumere temporaneamente un ruolo IAM in AWS Management Console [cambiando ruolo](#). Puoi assumere un ruolo chiamando un'operazione AWS CLI o AWS API o utilizzando un URL personalizzato. Per ulteriori informazioni sui metodi per l'utilizzo dei ruoli, consulta [Utilizzo di ruoli IAM](#) nella Guida per l'utente di IAM.

I ruoli IAM con credenziali temporanee sono utili nelle seguenti situazioni:

- **Accesso utente federato:** per assegnare le autorizzazioni a una identità federata, è possibile creare un ruolo e definire le autorizzazioni per il ruolo. Quando un'identità federata viene autenticata, l'identità viene associata al ruolo e ottiene le autorizzazioni da esso definite. Per ulteriori informazioni sulla federazione dei ruoli, consulta [Creazione di un ruolo per un provider di identità di terza parte](#) nella Guida per l'utente di IAM. Se utilizzi IAM Identity Center, configura un set di autorizzazioni. IAM Identity Center mette in correlazione il set di autorizzazioni con un ruolo in IAM per controllare a cosa possono accedere le identità dopo l'autenticazione. Per ulteriori informazioni sui set di autorizzazioni, consulta [Set di autorizzazioni](#) nella Guida per l'utente di AWS IAM Identity Center .
- **Autorizzazioni utente IAM temporanee:** un utente IAM o un ruolo può assumere un ruolo IAM per ottenere temporaneamente autorizzazioni diverse per un'attività specifica.
- **Accesso multi-account:** è possibile utilizzare un ruolo IAM per permettere a un utente (un principale affidabile) con un account diverso di accedere alle risorse nell'account. I ruoli sono lo strumento principale per concedere l'accesso multi-account. Tuttavia, con alcuni Servizi AWS, è possibile allegare una policy direttamente a una risorsa (anziché utilizzare un ruolo come proxy). Per informazioni sulle differenze tra ruoli e policy basate su risorse per l'accesso multi-account, consulta [Differenza tra i ruoli IAM e le policy basate su risorse](#) nella Guida per l'utente di IAM.
- **Accesso a più servizi:** alcuni Servizi AWS utilizzano le funzionalità di altri Servizi AWS. Ad esempio, quando effettui una chiamata in un servizio, è comune che tale servizio esegua applicazioni in Amazon EC2 o archivi oggetti in Amazon S3. Un servizio può eseguire questa operazione utilizzando le autorizzazioni dell'entità chiamante, utilizzando un ruolo di servizio o utilizzando un ruolo collegato al servizio.
  - **Sessioni di accesso diretto (FAS):** quando utilizzi un utente o un ruolo IAM per eseguire azioni AWS, sei considerato un preside. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra azione in un servizio diverso. FAS utilizza le autorizzazioni del principale che chiama un Servizio AWS, combinate con la richiesta Servizio AWS per

effettuare richieste ai servizi downstream. Le richieste FAS vengono effettuate solo quando un servizio riceve una richiesta che richiede interazioni con altri Servizi AWS o risorse per essere completata. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le operazioni. Per i dettagli delle policy relative alle richieste FAS, consulta la pagina [Forward access sessions](#).

- **Ruolo di servizio:** un ruolo di servizio è un [ruolo IAM](#) assunto da un servizio per eseguire operazioni per conto dell'utente. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione [Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS](#) nella Guida per l'utente di IAM.
- **Ruolo collegato al servizio:** un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un Servizio AWS. Il servizio può assumere il ruolo per eseguire un'azione per tuo conto. I ruoli collegati al servizio vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non modificarle.
- **Applicazioni in esecuzione su Amazon EC2:** puoi utilizzare un ruolo IAM per gestire le credenziali temporanee per le applicazioni in esecuzione su un'istanza EC2 e che AWS CLI effettuano richieste API. AWS CLI è preferibile all'archiviazione delle chiavi di accesso nell'istanza EC2. Per assegnare un AWS ruolo a un'istanza EC2 e renderlo disponibile per tutte le sue applicazioni, crei un profilo di istanza collegato all'istanza. Un profilo dell'istanza contiene il ruolo e consente ai programmi in esecuzione sull'istanza EC2 di ottenere le credenziali temporanee. Per ulteriori informazioni, consulta [Utilizzo di un ruolo IAM per concedere autorizzazioni ad applicazioni in esecuzione su istanze di Amazon EC2](#) nella Guida per l'utente di IAM.

Per informazioni sull'utilizzo dei ruoli IAM, consulta [Quando creare un ruolo IAM \(invece di un utente\)](#) nella Guida per l'utente di IAM.

## Gestione dell'accesso con policy

Puoi controllare l'accesso AWS creando policy e collegandole a AWS identità o risorse. Una policy è un oggetto AWS che, se associato a un'identità o a una risorsa, ne definisce le autorizzazioni. AWS valuta queste politiche quando un principale (utente, utente root o sessione di ruolo) effettua una richiesta. Le autorizzazioni nelle policy determinano l'approvazione o il rifiuto della richiesta. La maggior parte delle politiche viene archiviata AWS come documenti JSON. Per ulteriori informazioni sulla struttura e sui contenuti dei documenti delle policy JSON, consulta [Panoramica delle policy JSON](#) nella Guida per l'utente di IAM.

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Per concedere agli utenti l'autorizzazione a eseguire azioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM. Successivamente l'amministratore può aggiungere le policy IAM ai ruoli e gli utenti possono assumere i ruoli.

Le policy IAM definiscono le autorizzazioni relative a un'operazione, a prescindere dal metodo utilizzato per eseguirla. Ad esempio, supponiamo di disporre di una policy che consente l'azione `iam:GetRole`. Un utente con tale policy può ottenere informazioni sul ruolo dall' AWS Management Console AWS CLI, dall' AWS CLI o dall' AWS API.

## Policy basate su identità

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le azioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta [Creazione di policy IAM](#) nella Guida per l'utente di IAM.

Le policy basate su identità possono essere ulteriormente classificate come policy inline o policy gestite. Le policy inline sono incorporate direttamente in un singolo utente, gruppo o ruolo. Le politiche gestite sono politiche autonome che puoi allegare a più utenti, gruppi e ruoli nel tuo Account AWS. Le politiche gestite includono politiche AWS gestite e politiche gestite dai clienti. Per informazioni su come scegliere tra una policy gestita o una policy inline, consulta [Scelta fra policy gestite e policy inline](#) nella Guida per l'utente di IAM.

## Policy basate su risorse

Le policy basate su risorse sono documenti di policy JSON che è possibile allegare a una risorsa. Gli esempi più comuni di policy basate su risorse sono le policy di attendibilità dei ruoli IAM e le policy dei bucket Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarle per controllare l'accesso a una risorsa specifica. Quando è allegata a una risorsa, una policy definisce le azioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario [specificare un principale](#) in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o Servizi AWS.

Le policy basate sulle risorse sono policy inline che si trovano in tale servizio. Non puoi utilizzare le policy AWS gestite di IAM in una policy basata sulle risorse.

## Liste di controllo degli accessi (ACL)

Le liste di controllo degli accessi (ACL) controllano quali principali (membri, utenti o ruoli dell'account) hanno le autorizzazioni per accedere a una risorsa. Le ACL sono simili alle policy basate su risorse, sebbene non utilizzino il formato del documento di policy JSON.

Amazon S3 e Amazon VPC sono esempi di servizi che supportano gli ACL. AWS WAF Per maggiori informazioni sulle ACL, consulta [Panoramica delle liste di controllo degli accessi \(ACL\)](#) nella Guida per gli sviluppatori di Amazon Simple Storage Service.

## Altri tipi di policy

AWS supporta tipi di policy aggiuntivi e meno comuni. Questi tipi di policy possono impostare il numero massimo di autorizzazioni concesse dai tipi di policy più comuni.

- **Limiti delle autorizzazioni:** un limite delle autorizzazioni è una funzione avanzata nella quale si imposta il numero massimo di autorizzazioni che una policy basata su identità può concedere a un'entità IAM (utente o ruolo IAM). È possibile impostare un limite delle autorizzazioni per un'entità. Le autorizzazioni risultanti sono l'intersezione delle policy basate su identità dell'entità e i relativi limiti delle autorizzazioni. Le policy basate su risorse che specificano l'utente o il ruolo nel campo `Principal` sono condizionate dal limite delle autorizzazioni. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni sui limiti delle autorizzazioni, consulta [Limiti delle autorizzazioni per le entità IAM](#) nella Guida per l'utente di IAM.
- **Politiche di controllo dei servizi (SCP):** le SCP sono politiche JSON che specificano le autorizzazioni massime per un'organizzazione o un'unità organizzativa (OU) in AWS Organizations. AWS Organizations è un servizio per il raggruppamento e la gestione centralizzata di più Account AWS di proprietà dell'azienda. Se abiliti tutte le funzionalità in un'organizzazione, puoi applicare le policy di controllo dei servizi (SCP) a uno o tutti i tuoi account. L'SCP limita le autorizzazioni per le entità negli account dei membri, inclusa ciascuna. Utente root dell'account AWS Per ulteriori informazioni su organizzazioni e policy SCP, consulta la pagina sulle [Policy di controllo dei servizi](#) nella Guida per l'utente di AWS Organizations .
- **Policy di sessione:** le policy di sessione sono policy avanzate che vengono trasmesse come parametro quando si crea in modo programmatico una sessione temporanea per un ruolo o un utente federato. Le autorizzazioni della sessione risultante sono l'intersezione delle policy basate su identità del ruolo o dell'utente e le policy di sessione. Le autorizzazioni possono anche provenire da una policy basata su risorse. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni, consulta [Policy di sessione](#) nella Guida per l'utente di IAM.

## Più tipi di policy

Quando più tipi di policy si applicano a una richiesta, le autorizzazioni risultanti sono più complicate da comprendere. Per scoprire come si AWS determina se consentire una richiesta quando sono coinvolti più tipi di policy, consulta [Logica di valutazione delle policy](#) nella IAM User Guide.

## In che modo AWS CloudShell funziona con IAM

Prima di utilizzare IAM per gestire l'accesso a CloudShell, scopri con quali funzionalità IAM è disponibile l'uso CloudShell.

Funzionalità IAM che puoi usare con AWS CloudShell

Funzionalità IAM	CloudShell supporto
<a href="#">Policy basate su identità</a>	Sì
<a href="#">Policy basate su risorse</a>	No
<a href="#">Azioni di policy</a>	Sì
<a href="#">Risorse relative alle policy</a>	Sì
<a href="#">Chiavi di condizione della policy (specifica del servizio)</a>	Sì
<a href="#">Liste di controllo degli accessi (ACL)</a>	No
<a href="#">ABAC (tag nelle policy)</a>	No
<a href="#">Credenziali temporanee</a>	Sì
<a href="#">Inoltro delle sessioni di accesso (FAS)</a>	No
<a href="#">Ruoli di servizio</a>	No
<a href="#">Ruoli collegati al servizio</a>	No



Per avere una panoramica di alto livello su come CloudShell e altri AWS servizi funzionano con la maggior parte delle funzionalità IAM, consulta [AWS i servizi che funzionano con IAM nella IAM User Guide](#).

## Politiche basate sull'identità per CloudShell

Supporta le policy basate su identità Sì

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le azioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta [Creazione di policy IAM](#) nella Guida per l'utente di IAM.

Con le policy basate su identità di IAM, è possibile specificare quali operazioni e risorse sono consentite o respinte, nonché le condizioni in base alle quali le operazioni sono consentite o respinte. Non è possibile specificare l'entità principale in una policy basata sull'identità perché si applica all'utente o al ruolo a cui è associato. Per informazioni su tutti gli elementi utilizzabili in una policy JSON, consulta [Guida di riferimento agli elementi delle policy JSON IAM](#) nella Guida per l'utente di IAM.

## Esempi di politiche basate sull'identità per CloudShell

Per visualizzare esempi di politiche basate sull' CloudShell identità, vedere. [Esempi di policy basate sull'identità per AWS CloudShell](#)

## Politiche basate sulle risorse all'interno CloudShell

Supporta le policy basate su risorse No

Le policy basate su risorse sono documenti di policy JSON che è possibile allegare a una risorsa. Gli esempi più comuni di policy basate su risorse sono le policy di attendibilità dei ruoli IAM e le policy dei bucket Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarle per controllare l'accesso a una risorsa specifica. Quando è allegata a una risorsa, una policy definisce le azioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario [specificare un principale](#) in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o. Servizi AWS

Per consentire l'accesso multi-account, puoi specificare un intero account o entità IAM in un altro account come principale in una policy basata sulle risorse. L'aggiunta di un principale multi-account a una policy basata sulle risorse rappresenta solo una parte della relazione di trust. Quando il principale e la risorsa sono diversi Account AWS, un amministratore IAM dell'account affidabile deve inoltre concedere all'entità principale (utente o ruolo) l'autorizzazione ad accedere alla risorsa. L'autorizzazione viene concessa collegando all'entità una policy basata sull'identità. Tuttavia, se una policy basata su risorse concede l'accesso a un principale nello stesso account, non sono richieste ulteriori policy basate su identità. Per ulteriori informazioni, consulta [Differenza tra i ruoli IAM e le policy basate su risorse](#) nella Guida per l'utente di IAM.

## Azioni politiche per CloudShell

Supporta le azioni di policy	Sì
------------------------------	----

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire azioni su quali risorse, e in quali condizioni.

L'elemento `Action` di una policy JSON descrive le azioni che è possibile utilizzare per consentire o negare l'accesso a una policy. Le azioni politiche in genere hanno lo stesso nome dell'operazione AWS API associata. Ci sono alcune eccezioni, ad esempio le azioni di sola autorizzazione che non hanno un'operazione API corrispondente. Esistono anche alcune operazioni che richiedono più operazioni in una policy. Queste operazioni aggiuntive sono denominate operazioni dipendenti.

Includi le operazioni in una policy per concedere le autorizzazioni a eseguire l'operazione associata.

Per visualizzare un elenco di CloudShell azioni, consulta [Actions defined by AWS CloudShell](#) nel Service Authorization Reference. Alcune azioni possono avere più di un'API.

Le azioni politiche in CloudShell uso utilizzano il seguente prefisso prima dell'azione:

```
cloudshell
```

Per specificare più operazioni in una sola istruzione, occorre separarle con la virgola.

```
"Action": [  
  "cloudshell:action1",  
  "cloudshell:action2"  
]
```

Per visualizzare esempi di politiche CloudShell basate sull'identità, vedere. [Esempi di policy basate sull'identità per AWS CloudShell](#)

## Risorse politiche per CloudShell

Supporta le risorse di policy	Sì
-------------------------------	----

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire operazioni su quali risorse, e in quali condizioni.

L'elemento JSON `Resource` della policy specifica l'oggetto o gli oggetti ai quali si applica l'azione. Le istruzioni devono includere un elemento `Resource` o un elemento `NotResource`. Come best practice, specifica una risorsa utilizzando il suo [nome della risorsa Amazon \(ARN\)](#). Puoi eseguire questa operazione per azioni che supportano un tipo di risorsa specifico, note come autorizzazioni a livello di risorsa.

Per le azioni che non supportano le autorizzazioni a livello di risorsa, ad esempio le operazioni di elenco, utilizza un carattere jolly (\*) per indicare che l'istruzione si applica a tutte le risorse.

```
"Resource": "*"

```

Per visualizzare un elenco dei tipi di CloudShell risorse e dei relativi ARN, consulta [Resources defined by AWS CloudShell](#) nel Service Authorization Reference. Per sapere con quali azioni è possibile specificare l'ARN di ogni risorsa, consulta [Actions defined by AWS](#). CloudShell

Per visualizzare esempi di politiche CloudShell basate sull'identità, consulta. [Esempi di policy basate sull'identità per AWS CloudShell](#)

## Chiavi relative alle condizioni delle politiche per CloudShell

Supporta le chiavi di condizione delle policy specifiche del servizio	Sì
---	----

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire azioni su quali risorse, e in quali condizioni.

L'elemento `Condition` (o blocco `Condition`) consente di specificare le condizioni in cui un'istruzione è in vigore. L'elemento `Condition` è facoltativo. Puoi compilare espressioni condizionali che utilizzano [operatori di condizione](#), ad esempio uguale a o minore di, per soddisfare la condizione nella policy con i valori nella richiesta.

Se specifichi più elementi `Condition` in un'istruzione o più chiavi in un singolo elemento `Condition`, questi vengono valutati da AWS utilizzando un'operazione AND logica. Se si specificano più valori per una singola chiave di condizione, AWS valuta la condizione utilizzando un'operazione logica. OR Tutte le condizioni devono essere soddisfatte prima che le autorizzazioni dell'istruzione vengano concesse.

Puoi anche utilizzare variabili segnaposto quando specifichi le condizioni. Ad esempio, puoi autorizzare un utente IAM ad accedere a una risorsa solo se è stata taggata con il relativo nome utente IAM. Per ulteriori informazioni, consulta [Elementi delle policy IAM: variabili e tag](#) nella Guida per l'utente di IAM.

AWS supporta chiavi di condizione globali e chiavi di condizione specifiche del servizio. Per visualizzare tutte le chiavi di condizione AWS globali, consulta le chiavi di [contesto delle condizioni AWS globali nella Guida](#) per l'utente IAM.

Per visualizzare un elenco di chiavi di CloudShell condizione, consulta [Condition keys for AWS CloudShell](#) nel Service Authorization Reference. Per sapere con quali azioni e risorse puoi utilizzare una chiave di condizione, consulta [Actions defined by AWS CloudShell](#).

Per visualizzare esempi di politiche CloudShell basate sull'identità, consulta [Esempi di policy basate sull'identità per AWS CloudShell](#)

## ACL in CloudShell

Supporta le ACL

No

Le liste di controllo degli accessi (ACL) controllano quali principali (membri, utenti o ruoli dell'account) hanno le autorizzazioni ad accedere a una risorsa. Le ACL sono simili alle policy basate su risorse, sebbene non utilizzino il formato del documento di policy JSON.

## ABAC con CloudShell

Supporta ABAC (tag nelle policy)

No

Il controllo dell'accesso basato su attributi (ABAC) è una strategia di autorizzazione che definisce le autorizzazioni in base agli attributi. In AWS, questi attributi sono chiamati tag. Puoi allegare tag a entità IAM (utenti o ruoli) e a molte AWS risorse. L'assegnazione di tag alle entità e alle risorse è il primo passaggio di ABAC. In seguito, vengono progettate policy ABAC per consentire operazioni quando il tag dell'entità principale corrisponde al tag sulla risorsa a cui si sta provando ad accedere.

La strategia ABAC è utile in ambienti soggetti a una rapida crescita e aiuta in situazioni in cui la gestione delle policy diventa impegnativa.

Per controllare l'accesso basato su tag, fornisci informazioni sui tag nell'[elemento condizione](#) di una policy utilizzando le chiavi di condizione `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`.

Se un servizio supporta tutte e tre le chiavi di condizione per ogni tipo di risorsa, il valore per il servizio è Yes (Sì). Se un servizio supporta tutte e tre le chiavi di condizione solo per alcuni tipi di risorsa, allora il valore sarà Parziale.

Per ulteriori informazioni su ABAC, consulta [Che cos'è ABAC?](#) nella Guida per l'utente di IAM. Per visualizzare un tutorial con i passaggi per l'impostazione di ABAC, consulta [Utilizzo del controllo degli accessi basato su attributi \(ABAC\)](#) nella Guida per l'utente di IAM.

## Utilizzo di credenziali temporanee con CloudShell

Supporta le credenziali temporanee	Sì
------------------------------------	----

Alcuni Servizi AWS non funzionano quando si accede utilizzando credenziali temporanee. Per ulteriori informazioni, incluse quelle che Servizi AWS funzionano con credenziali temporanee, consulta la sezione relativa alla [Servizi AWS compatibilità con IAM nella IAM User Guide](#).

Stai utilizzando credenziali temporanee se accedi AWS Management Console utilizzando qualsiasi metodo tranne nome utente e password. Ad esempio, quando accedete AWS utilizzando il link Single Sign-On (SSO) della vostra azienda, tale processo crea automaticamente credenziali temporanee. Le credenziali temporanee vengono create in automatico anche quando accedi alla console come utente e poi cambi ruolo. Per ulteriori informazioni sullo scambio dei ruoli, consulta [Cambio di un ruolo \(console\)](#) nella Guida per l'utente di IAM.

È possibile creare manualmente credenziali temporanee utilizzando l'API o AWS CLI. AWS consiglia di generare quindi possibile utilizzare tali credenziali temporanee per accedere. AWS consiglia di generare

dinamicamente credenziali temporanee anziché utilizzare chiavi di accesso a lungo termine. Per ulteriori informazioni, consulta [Credenziali di sicurezza provvisorie in IAM](#).

Quando cambi ruolo, utilizzerai un ambiente diverso. Non puoi cambiare ruolo all'interno dello stesso AWS CloudShell ambiente.

## Sessioni di accesso inoltrato per CloudShell

Supports forward access sessions (FAS)	No
--	----

Quando utilizzi un utente o un ruolo IAM per eseguire azioni AWS, sei considerato un principale. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra azione in un servizio diverso. FAS utilizza le autorizzazioni del principale che chiama un Servizio AWS, in combinazione con la richiesta Servizio AWS per effettuare richieste ai servizi downstream. Le richieste FAS vengono effettuate solo quando un servizio riceve una richiesta che richiede interazioni con altri Servizi AWS o risorse per essere completata. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le operazioni. Per i dettagli delle policy relative alle richieste FAS, consulta la pagina [Forward access sessions](#).

## Ruoli di servizio per CloudShell

Supporta i ruoli di servizio	No
------------------------------	----

Un ruolo di servizio è un [ruolo IAM](#) che un servizio assume per eseguire operazioni per tuo conto. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione [Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS](#) nella Guida per l'utente di IAM.

### Warning

La modifica delle autorizzazioni per un ruolo di servizio potrebbe compromettere la funzionalità. CloudShell Modifica i ruoli di servizio solo quando viene CloudShell fornita una guida in tal senso.

## Ruoli collegati ai servizi per CloudShell

Supporta i ruoli collegati ai servizi

No

Un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un servizio AWS. Il servizio può assumere il ruolo per eseguire un'azione per tuo conto. I ruoli collegati al servizio vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non modificarle.

## Esempi di policy basate sull'identità per AWS CloudShell

Per impostazione predefinita, gli utenti e i ruoli non dispongono dell'autorizzazione per creare o modificare risorse CloudShell. Inoltre, non possono eseguire attività utilizzando AWS Management Console, AWS Command Line Interface (AWS CLI) o API. AWS Per concedere agli utenti l'autorizzazione a eseguire azioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM. L'amministratore può quindi aggiungere le policy IAM ai ruoli e gli utenti possono assumere i ruoli.

Per informazioni su come creare una policy basata su identità IAM utilizzando questi documenti di policy JSON di esempio, consulta [Creazione di policy IAM](#) nella Guida per l'utente di IAM.

Per dettagli sulle azioni e sui tipi di risorse definiti da CloudShell, incluso il formato degli ARN per ciascun tipo di risorsa, consulta [Actions, resources and condition keys for AWS CloudShell](#) nel Service Authorization Reference.

### Argomenti

- [Best practice per le policy](#)
- [Utilizzo della console di CloudShell](#)
- [Consentire agli utenti di visualizzare le loro autorizzazioni](#)

## Best practice per le policy

Le politiche basate sull'identità determinano se qualcuno può creare, accedere o eliminare CloudShell risorse nel tuo account. Queste operazioni possono comportare costi aggiuntivi per l'Account AWS. Quando crei o modifichi policy basate su identità, segui queste linee guida e raccomandazioni:

- Inizia con le policy AWS gestite e passa alle autorizzazioni con privilegi minimi: per iniziare a concedere autorizzazioni a utenti e carichi di lavoro, utilizza le politiche gestite che concedono le autorizzazioni per molti casi d'uso comuni. AWS Sono disponibili nel tuo Account AWS. Ti consigliamo di ridurre ulteriormente le autorizzazioni definendo politiche gestite dai clienti specifiche per i tuoi casi d'uso. Per ulteriori informazioni, consulta [Policy gestite da AWS](#) o [Policy gestite da AWS per le funzioni dei processi](#) nella Guida per l'utente IAM.
- Applica le autorizzazioni con privilegi minimi: quando imposti le autorizzazioni con le policy IAM, concedi solo le autorizzazioni richieste per eseguire un'attività. Puoi farlo definendo le azioni che possono essere intraprese su risorse specifiche in condizioni specifiche, note anche come autorizzazioni con privilegi minimi. Per ulteriori informazioni sull'utilizzo di IAM per applicare le autorizzazioni, consulta [Policy e autorizzazioni in IAM](#) nella Guida per l'utente di IAM.
- Condizioni d'uso nelle policy IAM per limitare ulteriormente l'accesso: per limitare l'accesso a operazioni e risorse puoi aggiungere una condizione alle tue policy. Ad esempio, è possibile scrivere una condizione di policy per specificare che tutte le richieste devono essere inviate utilizzando SSL. Puoi anche utilizzare le condizioni per concedere l'accesso alle azioni del servizio se vengono utilizzate tramite uno specifico Servizio AWS, ad esempio AWS CloudFormation. Per ulteriori informazioni, consulta la sezione [Elementi delle policy JSON di IAM: condizione](#) nella Guida per l'utente di IAM.
- Utilizzo di IAM Access Analyzer per convalidare le policy IAM e garantire autorizzazioni sicure e funzionali: IAM Access Analyzer convalida le policy nuove ed esistenti in modo che aderiscano alla sintassi della policy IAM (JSON) e alle best practice di IAM. IAM Access Analyzer offre oltre 100 controlli delle policy e consigli utili per creare policy sicure e funzionali. Per ulteriori informazioni, consulta [Convalida delle policy per IAM Access Analyzer](#) nella Guida per l'utente di IAM.
- Richiedi l'autenticazione a più fattori (MFA): se hai uno scenario che richiede utenti IAM o un utente root nel Account AWS tuo, attiva l'MFA per una maggiore sicurezza. Per richiedere la MFA quando vengono chiamate le operazioni API, aggiungi le condizioni MFA alle policy. Per ulteriori informazioni, consulta [Configurazione dell'accesso alle API protetto con MFA](#) nella Guida per l'utente di IAM.

Per maggiori informazioni sulle best practice in IAM, consulta [Best practice di sicurezza in IAM](#) nella Guida per l'utente di IAM.

## Utilizzo della console di CloudShell

Per accedere alla CloudShell console AWS, devi disporre di un set minimo di autorizzazioni. Queste autorizzazioni devono consentirti di elencare e visualizzare i dettagli sulle CloudShell risorse del tuo.



Account AWS Se crei una policy basata sull'identità più restrittiva rispetto alle autorizzazioni minime richieste, la console non funzionerà nel modo previsto per le entità (utenti o ruoli) associate a tale policy.

Non è necessario consentire autorizzazioni minime per la console agli utenti che effettuano chiamate solo verso AWS CLI o l' AWS API. Al contrario, concedi l'accesso solo alle operazioni che corrispondono all'operazione API che stanno cercando di eseguire.

Per garantire che utenti e ruoli possano ancora utilizzare la CloudShell console, allega anche la policy CloudShell *ConsoleAccess* o la policy *ReadOnly* AWS gestita alle entità. Per ulteriori informazioni, consulta [Aggiunta di autorizzazioni a un utente](#) nella Guida per l'utente IAM.

## Consentire agli utenti di visualizzare le loro autorizzazioni

Questo esempio mostra in che modo è possibile creare una policy che consente agli utenti IAM di visualizzare le policy inline e gestite che sono allegate alla relativa identità utente. Questa politica include le autorizzazioni per completare questa azione sulla console o utilizzando l'API o a livello di codice. AWS CLI AWS

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",

```

```
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
    ],
    "Resource": "*"
}
]
```

## Risoluzione dei problemi relativi all' CloudShell identità e all'accesso di AWS

Utilizza le seguenti informazioni per aiutarti a diagnosticare e risolvere i problemi più comuni che potresti riscontrare quando lavori con CloudShell IAM.

### Argomenti

- [Non sono autorizzato a eseguire alcuna azione in CloudShell](#)
- [Non sono autorizzato a eseguire iam: PassRole](#)
- [Voglio consentire a persone esterne a me di accedere Account AWS alle mie CloudShell risorse](#)

### Non sono autorizzato a eseguire alcuna azione in CloudShell

Se ricevi un errore che indica che non sei autorizzato a eseguire un'operazione, le tue policy devono essere aggiornate per poter eseguire l'operazione.

L'errore di esempio seguente si verifica quando l'utente IAM `mateojackson` prova a utilizzare la console per visualizzare i dettagli relativi a una risorsa `my-example-widget` fittizia ma non dispone di autorizzazioni `aws:GetWidget` fittizie.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
aws:GetWidget on resource: my-example-widget
```

In questo caso, la policy per l'utente `mateojackson` deve essere aggiornata per consentire l'accesso alla risorsa `my-example-widget` utilizzando l'azione `aws:GetWidget`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

## Non sono autorizzato a eseguire iam: PassRole

Se ricevi un errore che indica che non sei autorizzato a eseguire l'operazione `iam:PassRole`, le tue policy devono essere aggiornate per poter passare un ruolo a CloudShell.

Alcuni Servizi AWS consentono di passare un ruolo esistente a quel servizio invece di creare un nuovo ruolo di servizio o un ruolo collegato al servizio. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per trasmettere il ruolo al servizio.

L'errore di esempio seguente si verifica quando un utente IAM denominato `marymajor` cerca di utilizzare la console per eseguire un'operazione in CloudShell. Tuttavia, l'azione richiede che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per passare il ruolo al servizio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

## Voglio consentire a persone esterne a me di accedere Account AWS alle mie CloudShell risorse

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle tue risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo. Per servizi che supportano policy basate su risorse o liste di controllo accessi (ACL), utilizza tali policy per concedere alle persone l'accesso alle tue risorse.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- Per sapere se CloudShell supporta queste funzionalità, consulta [In che modo AWS CloudShell funziona con IAM](#).
- Per scoprire come fornire l'accesso alle tue risorse attraverso Account AWS le risorse di tua proprietà, consulta [Fornire l'accesso a un utente IAM in un altro Account AWS di tua proprietà](#) nella IAM User Guide.

- Per scoprire come fornire l'accesso alle tue risorse a terze parti Account AWS, consulta [Fornire l'accesso a soggetti Account AWS di proprietà di terze parti](#) nella Guida per l'utente IAM.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consulta [Fornire l'accesso a utenti autenticati esternamente \(Federazione delle identità\)](#) nella Guida per l'utente di IAM.
- Per informazioni sulle differenze tra l'utilizzo di ruoli e policy basate su risorse per l'accesso multi-account, consulta [Differenza tra i ruoli IAM e le policy basate su risorse](#) nella Guida per l'utente IAM.

## Gestione dell' AWS CloudShell accesso e dell'utilizzo con le policy IAM

Con le risorse di gestione degli accessi che possono essere fornite da AWS Identity and Access Management (IAM), gli amministratori possono concedere le autorizzazioni agli utenti IAM. In questo modo, questi utenti possono accedere AWS CloudShell e utilizzare le funzionalità dell'ambiente. Gli amministratori possono anche creare policy che specificano a livello granulare le azioni che gli utenti possono eseguire con l'ambiente shell.

Il modo più rapido per un amministratore di concedere l'accesso agli utenti è tramite una AWS policy gestita. Una [policy gestita da AWS](#) è una policy autonoma creata e amministrata da AWS. La seguente policy AWS gestita per AWS CloudShell può essere allegata alle identità IAM:

- **AWS CloudShellFullAccess**: concede l'autorizzazione all'uso AWS CloudShell con accesso completo a tutte le funzionalità.

La AWS CloudShellFullAccesspolicy utilizza il carattere wildcard (\*) per fornire all'identità IAM (utente, ruolo o gruppo) l'accesso completo alle funzionalità CloudShell e alle funzionalità. Per ulteriori informazioni su questa politica, consulta [AWS CloudShellFullAccess](#) la AWS Managed Policy User Guide.

### Note

È inoltre possibile avviare CloudShell identità IAM con le seguenti politiche AWS gestite. Tuttavia, queste politiche forniscono autorizzazioni estese. Pertanto, ti consigliamo di concedere queste politiche solo se sono essenziali per il ruolo lavorativo di un utente IAM.

- [Amministratore](#): fornisce agli utenti IAM l'accesso completo e consente loro di delegare le autorizzazioni a tutti i servizi e le risorse in uso. AWS

- [Developer power user](#): consente agli utenti IAM di eseguire attività di sviluppo di applicazioni e di creare e configurare risorse e servizi che supportano lo sviluppo di applicazioni AWS consapevoli.

Per ulteriori informazioni su come allegare policy gestite, consulta [Adding IAM identity permissions \(console\)](#) nella IAM User Guide.

## Gestione delle azioni consentite nell' AWS CloudShell utilizzo di policy personalizzate

Per gestire le azioni che un utente IAM può eseguire CloudShell, crea una policy personalizzata che utilizzi la policy CloudShellPolicy gestita come modello. In alternativa, modifica una [policy in linea](#) incorporata nell'identità IAM pertinente (utente, gruppo o ruolo).

Ad esempio, puoi consentire agli utenti IAM di accedere CloudShell, ma impedire loro di inoltrare le credenziali di CloudShell ambiente utilizzate per accedere. AWS Management Console

### Important

Per eseguire l'avvio AWS CloudShell da AWS Management Console, un utente IAM necessita delle autorizzazioni per le seguenti azioni:

- `CreateEnvironment`
- `CreateSession`
- `GetEnvironmentStatus`
- `StartEnvironment`

Se una di queste azioni non è esplicitamente consentita da una policy allegata, viene restituito un errore di autorizzazione IAM quando si tenta di avviare. CloudShell

## AWS CloudShell autorizzazioni

Nome	Descrizione dell'auto rizzazione concessa	Necessario per il lancio CloudShell?
<code>cloudshell:CreateEnvironment</code>	Crea un CloudShell ambiente, recupera il layout all'inizio della CloudShell sessione e salva il layout corrente dall'applicazione web nel backend. Questa autorizzazione è prevista solo * come valore per Resource come descritto in <a href="#">the section called "Esempi di politiche IAM per CloudShell"</a>	Sì
<code>cloudshell:CreateSession</code>	Si connette a un CloudShell ambiente da. AWS Management Console	Sì
<code>cloudshell:GetEnvironmentStatus</code>	Leggi lo stato di un CloudShell ambiente.	Sì
<code>cloudshell&gt;DeleteEnvironment</code>	Elimina un CloudShell ambiente.	No
<code>cloudshell:GetFileDownloadUrls</code>	Genera URL Amazon S3 prefirmati che vengono utilizzati per scaricare file CloudShell tramite l' CloudShell interfaccia Web.	No
<code>cloudshell:GetFileUploadUrls</code>	Genera URL Amazon S3 prefirmati che vengono	No

Nome	Descrizione dell'auto rizzazione concessa	Necessario per il lancio CloudShell?
	utilizzati per caricare file CloudShell tramite l' CloudShell interfaccia Web.	
<code>cloudshell:PutCredentials</code>	Inoltra le credenziali utilizzate per accedere a. AWS Management Console CloudShell	No
<code>cloudshell:StartEnvironment</code>	Avvia un CloudShell ambiente che viene interrotto.	Sì
<code>cloudshell:StopEnvironment</code>	Arresta un CloudShell ambiente in esecuzione.	No

## Esempi di politiche IAM per CloudShell

Gli esempi seguenti mostrano come è possibile creare politiche per limitare chi può accedere CloudShell. Gli esempi mostrano anche le azioni che possono essere eseguite nell'ambiente shell.

La seguente politica impone una negazione totale dell'accesso a CloudShell e alle relative funzionalità.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "DenyCloudShell",
    "Effect": "Deny",
    "Action": [
      "cloudshell:*"
    ],
    "Resource": "*"
  }]
}
```

La seguente policy consente agli utenti IAM di accedere, CloudShell ma impedisce loro di generare URL prefirmati per il caricamento e il download di file. Gli utenti possono comunque trasferire file da e verso l'ambiente, utilizzando client come ad wget esempio.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowUsingCloudshell",
      "Effect": "Allow",
      "Action": [
        "cloudshell:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyUploadDownload",
      "Effect": "Deny",
      "Action": [
        "cloudshell:GetFileDownloadUrls",
        "cloudshell:GetFileUploadUrls"
      ],
      "Resource": "*"
    }
  ]
}
```

La seguente policy consente agli utenti IAM di accedere CloudShell. Tuttavia, la policy impedisce che le credenziali utilizzate per accedere AWS Management Console vengano inoltrate all'ambiente. CloudShell Gli utenti IAM con questa policy devono configurare manualmente le proprie credenziali all'interno. CloudShell

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowUsingCloudshell",
      "Effect": "Allow",
      "Action": [
        "cloudshell:*"
      ],
      "Resource": "*"
    },
  ],
}
```



```
{
  "Sid": "DenyCredentialForwarding",
  "Effect": "Deny",
  "Action": [
    "cloudshell:PutCredentials"
  ],
  "Resource": "*"
}]
}
```

La seguente policy consente agli utenti IAM di creare AWS CloudShell ambienti.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "CloudShellUser",
    "Effect": "Allow",
    "Action": [
      "cloudshell:CreateEnvironment",
      "cloudshell:CreateSession",
      "cloudshell:GetEnvironmentStatus",
      "cloudshell:StartEnvironment"
    ],
    "Resource": "*"
  }]
}
```

## Autorizzazioni per l'accesso Servizi AWS

CloudShell utilizza le credenziali IAM utilizzate per accedere a. AWS Management Console

### Note

Per utilizzare le credenziali IAM utilizzate per accedere a AWS Management Console, è necessario disporre `cloudshell:PutCredentials` dell'autorizzazione.

Questa funzionalità di preautenticazione CloudShell lo rende comodo da usare. AWS CLI Tuttavia, un utente IAM richiede comunque autorizzazioni esplicite per Servizi AWS le chiamate dalla riga di comando.

Ad esempio, supponiamo che agli utenti IAM venga richiesto di creare bucket Amazon S3 e di caricare file come oggetti su di essi. Puoi creare una policy che consenta esplicitamente tali azioni. La console IAM fornisce un [editor visivo](#) interattivo che guida attraverso il processo di creazione di un documento di policy in formato JSON. Dopo aver creato la policy, puoi collegarla all'identità IAM pertinente (utente, gruppo o ruolo).

Per ulteriori informazioni sull'allegazione di policy gestite, consulta [Aggiungere i permessi di identità IAM \(console\) nella Guida](#) per l'utente IAM.

## Registrazione e monitoraggio AWS CloudShell

Questo argomento descrive come registrare e monitorare AWS CloudShell attività e prestazioni con CloudTrail.

### Monitoraggio dell'attività con CloudTrail

AWS CloudShell è integrato con AWS CloudTrail, un servizio che fornisce un registro delle azioni intraprese da un utente, ruolo o Servizio AWS responsabile AWS CloudShell. CloudTrail acquisisce tutte le chiamate API AWS CloudShell come eventi. Le chiamate acquisite includono chiamate dalla AWS CloudShell console e chiamate di codice all' AWS CloudShell API.

Se crei un trail, puoi abilitare la distribuzione continua di CloudTrail eventi a un bucket Amazon Simple Storage Service (Amazon S3). Sono inclusi eventi per. AWS CloudShell

Se non configuri un trail, è comunque possibile visualizzare gli eventi più recenti nella console di CloudTrail in Event history (Cronologia eventi). Utilizzando le informazioni raccolte da CloudTrail, puoi scoprire una serie di informazioni su una richiesta. Ad esempio, puoi determinare la richiesta che è stata fatta ad AWS CloudShell, puoi conoscere l'indirizzo IP da cui è stata effettuata la richiesta, chi ha effettuato la richiesta e quando è stata effettuata.

### AWS CloudShell in CloudTrail

La tabella seguente elenca gli AWS CloudShell eventi salvati nel file di CloudTrail registro.

#### Note

AWS CloudShell evento che include:

- \*indica che si tratta di una chiamata API non mutante (di sola lettura).

- La parola `Environment` si riferisce al ciclo di vita dell'ambiente di calcolo che ospita l'esperienza shell.
- La parola `Layout` ripristina tutte le schede del browser nel terminale. CloudShell

## CloudShell Eventi in CloudTrail

Nome evento	Descrizione
<code>createEnvironment</code>	Si verifica quando viene creato un CloudShell ambiente.
<code>createSession</code>	Si verifica quando un CloudShell ambiente è connesso da AWS Management Console.
<code>deleteEnvironment</code>	Si verifica quando un CloudShell ambiente viene eliminato.
<code>deleteSession</code>	Si verifica quando la sessione nella CloudShell scheda in esecuzione nella scheda corrente del browser viene eliminata.
<code>getEnvironmentStatus*</code>	Si verifica quando viene recuperato lo stato di un CloudShell ambiente.
<code>getFileDownloadUrls*</code>	Si verifica quando vengono generati URL Amazon S3 prefirmati utilizzati per scaricare file CloudShell tramite CloudShell l'interfaccia Web.
<code>getFileUploadUrls*</code>	Si verifica quando vengono generati URL Amazon S3 prefirmati utilizzati per caricare file CloudShell tramite CloudShell l'interfaccia Web.
<code>getLayout*</code>	Si verifica quando viene CloudShell recuperato il layout all'inizio della sessione.

Nome evento	Descrizione
<code>putCredentials</code>	Si verifica quando le credenziali utilizzate per accedere a AWS Management Console to CloudShell vengono inoltrate.
<code>redeemCode*</code>	Si verifica quando inizia il flusso di lavoro per recuperare il token di aggiornamento nell'ambiente. CloudShell È possibile utilizzare successivamente questo token nel <code>putCredentials</code> comando per accedere all' CloudShell ambiente.
<code>sendHeartBeat</code>	Si verifica per confermare che la CloudShell sessione è attiva.
<code>startEnvironment</code>	Si verifica all'avvio di un CloudShell ambiente.
<code>stopEnvironment</code>	Si verifica quando un CloudShell ambiente in esecuzione viene interrotto.
<code>updateLayout</code>	Si verifica quando viene salvato il layout corrente dall'applicazione Web nel backend.

Gli eventi che includono la parola «Layout» ripristinano tutte le schede del browser nel CloudShell terminale.

### EventBridge regole per le azioni AWS CloudShell

Con EventBridge le regole, si specifica un'azione mirata da intraprendere quando EventBridge riceve un evento che corrisponde alla regola. È possibile definire una regola che specifica un'azione mirata da intraprendere in base a un' AWS CloudShell azione registrata come evento in un file di CloudTrail registro.

Ad esempio, è possibile [creare EventBridge regole AWS CLI](#) utilizzando il `put-rule` comando. Una `put-rule` chiamata deve contenere almeno un EventPattern o ScheduleExpression. Le regole con EventPatterns vengono attivate quando viene osservato un evento corrispondente. I EventPattern per AWS CloudShell gli eventi:

```
{ "source": [ "aws.cloudshell" ], "detail-type": [ "AWS API Call via CloudTrail" ],
  "detail": { "eventSource": [ "cloudshell.amazonaws.com" ] } }
```

Per ulteriori informazioni, consulta [Events and Event Patterns EventBridge nella Amazon EventBridge User Guide](#).

## Convalida della conformità per AWS CloudShell

I revisori esterni valutano la sicurezza e la conformità dei AWS servizi nell'ambito di più programmi di AWS conformità.

AWS CloudShell è conforme ai seguenti programmi di conformità:

### SOC

AWS I rapporti SOC (System and Organization Controls) sono rapporti di esame indipendenti di terze parti che dimostrano come AWS raggiungere i controlli e gli obiettivi chiave di conformità.

Servizio	SDK	<a href="#">SOC 1,2,3</a>
AWS CloudShell	CloudShell	✓

### PCI

Il Payment Card Industry Data Security Standard (PCI DSS) è uno standard di sicurezza delle informazioni proprietario amministrato dal PCI Security Standards Council, fondato da American Express, Discover Financial Services, JCB International, Worldwide e Visa Inc. MasterCard

Servizio	SDK	<a href="#">PCI</a>
AWS CloudShell	CloudShell	✓

### Certificazioni e servizi ISO e CSA STAR

AWS dispone della certificazione di conformità agli standard ISO/IEC 27001:2013, 27017:2015, 27018:2019, 27701:2019, 22301:2019, 9001:2015 e CSA STAR CCM v4.0.

Servizio	SDK	<a href="#">Certificazioni e servizi ISO e CSA STAR</a>
AWS CloudShell	CloudShell	✓

## FedRamp

FedRAMP sta per Federal Risk and Authorization Management Program; si tratta di un programma federale statunitense per la gestione di rischio e autorizzazioni applicato a livello di pubblica amministrazione che fornisce un approccio standard a valutazioni di sicurezza, assegnazione di autorizzazioni e monitoraggio continuo nell'ambito di servizi e prodotti cloud.

Servizio	SDK	<a href="#">FedRAMP Moderate (Est/Ovest)</a>	<a href="#">FedRAMP High () GovCloud</a>
AWS CloudShell	CloudShell	✓	✓

## DoD CC SRG

La Security Requirements Guide (SRG) al cloud computing del Dipartimento della Difesa (DoD) fornisce un processo di valutazione e autorizzazione standardizzato per i provider di servizi cloud (CSP) per ottenere un'autorizzazione provvisoria DoD, in modo che possano servire i clienti DoD.

I servizi che passano attraverso la valutazione e l'autorizzazione DoD CC SRG avranno il seguente stato:

- Valutazione dell'organizzazione di valutazione di terze parti (3PAO): Questo servizio è attualmente sottoposto a una valutazione da parte del nostro valutatore esterno.
- Revisione del Joint Authorization Board (JAB): questo servizio è attualmente in fase di revisione da parte del JAB.
- Revisione della Defense Information Systems Agency (DISA): Questo servizio è attualmente sottoposto a revisione DISA.

Servizio	SDK	<a href="#">DoD CC SRG IL2 (Est/Ovest)</a>	<a href="#">DoD CC SRG IL2 () GovCloud</a>	<a href="#">DoD CC SRG IL4 () GovCloud</a>	<a href="#">DoD CC SRG IL5 () GovCloud</a>	<a href="#">DoD CC SRG IL6 (Regione segreta)A WS</a>
AWS CloudShell	CloudShell	Valutazione 3PAO	N/D	N/D	N/D	N/D

## HIPAA BAA

L'Health Insurance Portability and Accountability Act del 1996 (HIPAA) è una legge federale che richiedeva la creazione di standard nazionali per proteggere le informazioni sanitarie sensibili dei pazienti dalla divulgazione senza il consenso o la conoscenza da parte dei pazienti.

AWS consente alle entità coperte e ai loro partner commerciali soggetti all'HIPAA di elaborare, archiviare e trasmettere in modo sicuro informazioni sanitarie protette (PHI). Inoltre, a partire da luglio 2013, AWS offre un Business Associate Addendum (BAA) standardizzato per tali clienti.

Servizio	SDK	<a href="#">HIPAA BAA</a>
AWS CloudShell	CloudShell	✓

## IRAP

L'Information Security Registered Assessors Program (IRAP) consente ai clienti governativi australiani di convalidare l'esistenza di controlli appropriati e di determinare il modello di responsabilità appropriato per soddisfare i requisiti del Manuale per la sicurezza delle informazioni del governo australiano (ISM) prodotto dal Centro australiano per la sicurezza informatica (ACSC).

Servizio	Spazio dei nomi*	<a href="#">Protezione IRAP</a>
AWS CloudShell	N/D	✓

\*I namespace consentono di identificare i servizi in tutto l'ambiente. AWS Ad esempio, quando crei policy IAM, lavori con Amazon Resource Names (ARN) e leggi i AWS CloudTrail log.

## MTCS

Il Multi-Tier Cloud Security (MTCS) è uno standard operativo di gestione della sicurezza di Singapore (SPRING SS 584), basato sugli standard ISO 27001/02 del sistema di gestione della sicurezza delle informazioni (ISMS).

Servizio	SDK	Stati Uniti orientali (Ohio)	Stati Uniti orientali (Virginia settentrionale)	Stati Uniti occidentali (Oregon)	Stati Uniti occidentali (California settentrionale)	Singapore	Seul
AWS CloudShell	CloudShell	✓	✓	✓	N/D	N/D	N/D

## C5

Cloud Computing Compliance Controls Catalog (C5) è uno schema di attestazione del governo tedesco introdotto in Germania dal Federal Office for Information Security (BSI) per aiutare le organizzazioni a dimostrare la sicurezza operativa contro gli attacchi informatici comuni quando utilizzano i servizi cloud nel contesto delle Security Recommendations for Cloud Providers “Raccomandazioni sulla sicurezza per fornitori di servizi cloud” del governo tedesco.

Servizio	SDK	<a href="#">C5</a>
AWS CloudShell	CloudShell	✓

## ENS High

Lo schema di accreditamento ENS (Esquema Nacional de Seguridad) è stato sviluppato dal Ministero delle Finanze e della Pubblica Amministrazione e dal CCN (National Cryptologic Centre). Ciò comprende i principi di base e i requisiti minimi necessari per un'adeguata protezione delle informazioni.



Servizio	SDK	<a href="#">ENS Alto</a>
AWS CloudShell	CloudShell	✓

## FINMA

L'Autorità federale di vigilanza sui mercati finanziari (FINMA) è l'autorità di regolamentazione indipendente dei mercati finanziari della Svizzera. AWS l'allineamento ai requisiti della FINMA dimostra il nostro costante impegno a soddisfare le crescenti aspettative per i fornitori di servizi cloud stabilite dalle autorità di regolamentazione e dai clienti svizzeri dei servizi finanziari.

Servizio	SDK	<a href="#">FINMA</a>
AWS CloudShell	CloudShell	✓

## PiTuKri

AWS l'allineamento ai PiTuKri requisiti dimostra il nostro costante impegno a soddisfare le crescenti aspettative per i fornitori di servizi cloud stabilite dall'Agenzia finlandese per i trasporti e le comunicazioni Traficom.

Servizio	SDK	<a href="#">PiTuKri</a>
AWS CloudShell	CloudShell	✓

Per un elenco di AWS servizi che rientrano nell'ambito di programmi di conformità specifici, consulta [AWS Services in Scope by Compliance Program](#) . Per informazioni generali, consulta [Programmi di AWS conformità Programmi](#) di di .

È possibile scaricare report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta [Download dei report in AWS Artifact](#).

La vostra responsabilità di conformità durante l'utilizzo AWS CloudShell è determinata dalla sensibilità dei dati, dagli obiettivi di conformità dell'azienda e dalle leggi e dai regolamenti applicabili. AWS fornisce le seguenti risorse per contribuire alla conformità:

- [Guide rapide su sicurezza e conformità](#) [Guide introduttive](#) implementazione illustrano considerazioni sull'architettura e forniscono passaggi per implementare ambienti di base incentrati sulla sicurezza e sulla conformità. AWS
- [Whitepaper sull'architettura per la sicurezza e la conformità HIPAA: questo white paper descrive in che modo le aziende possono utilizzare per creare applicazioni conformi allo standard HIPAA.](#) AWS
- AWS Risorse per [la conformità](#) [Risorse per la conformità](#): questa raccolta di potrebbe riguardare il settore e la località in cui operate.
- [Valutazione delle risorse con le regole](#) nella Guida per gli AWS Config sviluppatori: il AWS Config servizio valuta la conformità delle configurazioni delle risorse alle pratiche interne, alle linee guida del settore e alle normative.
- [AWS Security Hub](#)— Questo AWS servizio offre una visione completa dello stato di sicurezza dell'utente, AWS che consente di verificare la conformità agli standard e alle best practice del settore della sicurezza.

## Resilienza in AWS CloudShell

L'infrastruttura AWS globale è costruita attorno a AWS regioni e zone di disponibilità. AWS Le regioni forniscono più zone di disponibilità fisicamente separate e isolate, collegate con reti a bassa latenza, ad alto throughput e altamente ridondanti. Con le zone di disponibilità, puoi progettare e gestire applicazioni e database che eseguono automaticamente il failover tra zone di disponibilità senza interruzioni. Le zone di disponibilità sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture a data center singolo o multiplo tradizionali.

[Per ulteriori informazioni su AWS regioni e zone di disponibilità, consulta Global Infrastructure.AWS](#)

Oltre all'infrastruttura AWS globale, AWS CloudShell supporta funzionalità specifiche per supportare le esigenze di resilienza e backup dei dati.

- Esegui il commit dei file che crei e a cui aggiungi. AWS CodeCommit Si tratta di un servizio di controllo delle versioni ospitato da Amazon Web Services che puoi utilizzare per archiviare e gestire in modo privato le risorse nel cloud. Queste risorse possono essere costituite da documenti, codice sorgente e file binari. Per ulteriori informazioni, consulta [Tutorial: Utilizzo CodeCommit inAWS CloudShell](#).
- Usa AWS CLI le chiamate per specificare i file nella tua home directory AWS CloudShell e aggiungerli come oggetti nei bucket Amazon S3. Per un esempio, consulta il tutorial [introduttivo](#).

## Sicurezza dell'infrastruttura in AWS CloudShell

In quanto servizio gestito, AWS CloudShell è protetto dalla sicurezza di rete AWS globale. Per informazioni sui servizi AWS di sicurezza e su come AWS protegge l'infrastruttura, consulta [AWS Cloud Security](#). Per progettare il tuo AWS ambiente utilizzando le migliori pratiche per la sicurezza dell'infrastruttura, vedi [Infrastructure Protection](#) in Security Pillar AWS Well-Architected Framework.

Utilizzate chiamate API AWS pubblicate per accedere AWS CloudShell attraverso la rete. I client devono supportare quanto segue:

- Transport Layer Security (TLS). È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Suite di cifratura con Perfect Forward Secrecy (PFS), ad esempio Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Inoltre, le richieste devono essere firmate utilizzando un ID chiave di accesso e una chiave di accesso segreta associata a un principale IAM. O puoi utilizzare [AWS Security Token Service](#) (AWS STS) per generare credenziali di sicurezza temporanee per sottoscrivere le richieste.

### Note

Per impostazione predefinita, installa AWS CloudShell automaticamente le patch di sicurezza per i pacchetti di sistema dei tuoi ambienti di elaborazione.

## Analisi della configurazione e delle vulnerabilità in AWS CloudShell

È responsabilità dell' AWS CloudShell utente assicurarsi che qualsiasi software installato nell'ambiente di elaborazione sia corredato di patch e aggiornato.

## Best practice di sicurezza per AWS CloudShell

Le seguenti best practice sono linee guida generali e non rappresentano una soluzione di sicurezza completa. Poiché queste best practice potrebbero non essere appropriate o sufficienti per il tuo ambiente, gestiscile come considerazioni utili anziché prescrizioni.

## Alcune best practice di sicurezza per AWS CloudShell

- Utilizza le autorizzazioni e le policy IAM per controllare l'accesso AWS CloudShell e garantire che gli utenti possano eseguire solo le azioni (ad esempio il download e il caricamento di file) richieste dal loro ruolo. Per ulteriori informazioni, consulta [Gestione dell' AWS CloudShell accesso e dell'utilizzo con le policy IAM](#).
- Non includere dati sensibili nelle entità IAM come utenti, ruoli o nomi di sessione.
- La funzione Keep Safe Paste è abilitata per catturare potenziali rischi per la sicurezza nel testo che hai copiato da fonti esterne. Safe Paste è abilitato per impostazione predefinita. Per ulteriori informazioni, consulta [Utilizzo di Safe Paste per testo su più righe](#).
- Acquisisci familiarità con lo [Shared Security Responsibility Model](#) se hai installato applicazioni di terze parti nell'ambiente di calcolo di AWS CloudShell.
- Prepara i meccanismi di rollback prima di modificare gli script di shell che influiscono sull'esperienza dell'utente con la shell. Per ulteriori informazioni, consulta [Modificare la shell con degli script](#).
- Memorizza il tuo codice in modo sicuro in un sistema di controllo della versione, ad esempio [AWS CodeCommit](#).

## AWS CloudShell Domande frequenti sulla sicurezza

Risposte alle domande frequenti sulla sicurezza a questo Servizio AWS proposito.

- [Quali sono i AWS processi e le tecnologie utilizzati quando si avvia CloudShell e si avvia una sessione di shell?](#)
- [È possibile limitare l'accesso alla rete a CloudShell?](#)
- [Posso personalizzare il mio CloudShell ambiente?](#)
- [Dove è effettivamente archiviata la mia \\$HOME directory in? Cloud AWS](#)
- [È possibile crittografare la mia cartella? \\$HOME](#)
- [Posso eseguire una scansione antivirus sulla mia \\$HOME directory?](#)

## Quali sono i AWS processi e le tecnologie utilizzati quando si avvia CloudShell e si avvia una sessione di shell?

Quando accedi AWS Management Console, inserisci le tue credenziali utente IAM. Inoltre, quando esegui l'avvio CloudShell dall'interfaccia della console, queste credenziali vengono utilizzate nelle chiamate all' CloudShell API che creano un ambiente di calcolo per il servizio. Viene quindi creata una AWS Systems Manager sessione per l'ambiente di calcolo e CloudShell invia comandi a tale sessione.

[Torna all'elenco delle domande frequenti sulla sicurezza](#)

## È possibile limitare l'accesso alla rete a CloudShell?

Potrebbe essere possibile limitare l'accesso alla rete CloudShell collegandosi al proprio provider di rete. In alternativa, puoi utilizzare le autorizzazioni IAM, negare esplicitamente l'accesso CloudShell o non fornire alcuna autorizzazione IAM e utilizzare la funzionalità di negazione implicita di IAM. Per ulteriori informazioni, consulta [Gestire CloudShell l'accesso e l'utilizzo di AWS con le politiche IAM](#).

[Torna all'elenco delle domande frequenti sulla sicurezza](#)

## Posso personalizzare il mio CloudShell ambiente?

Puoi scaricare e installare utilità e altri software di terze parti per il tuo CloudShell ambiente. Solo il software installato nella \$HOME directory viene mantenuto tra le sessioni.

Come definito dal [modello di responsabilitàAWS condivisa](#), l'utente è responsabile della configurazione e della gestione necessarie delle applicazioni installate.

[Torna all'elenco delle domande frequenti sulla sicurezza](#)

## Dove è effettivamente archiviata la mia \$HOME directory in? Cloud AWS

L'infrastruttura per l'archiviazione dei dati nel tuo \$HOME computer è fornita da Amazon S3.

[Torna all'elenco delle domande frequenti sulla sicurezza](#)

## È possibile crittografare la mia cartella? \$HOME

I tuoi dati nella tua \$HOME directory sono già crittografati utilizzando Amazon S3 Encryption.

[Torna all'elenco delle domande frequenti sulla sicurezza](#)

## Posso eseguire una scansione antivirus sulla mia **\$HOME** directory?

Al momento non è possibile eseguire una scansione antivirus della \$HOME directory. Il supporto per questa funzionalità è in fase di revisione.

[Torna all'elenco delle domande frequenti sulla sicurezza](#)

# AWS CloudShell ambiente di calcolo: specifiche e software

Al momento del lancio AWS CloudShell, viene creato un ambiente di elaborazione basato su [Amazon Linux 2023](#) per ospitare l'esperienza shell. L'ambiente è configurato con [risorse di calcolo \(vCPU e memoria\)](#) e fornisce un'ampia gamma [di software preinstallato](#) a cui è possibile accedere dall'interfaccia a riga di comando. È inoltre possibile configurare l'ambiente predefinito installando software e modificando gli script di shell.

## Risorse dell'ambiente di calcolo

A ogni ambiente di AWS CloudShell calcolo vengono assegnate le seguenti risorse di CPU e memoria:

- 1 vCPU (unità di elaborazione centrale virtuale)
- RAM da 2 GiB

Inoltre, l'ambiente viene fornito con la seguente configurazione di archiviazione:

- Storage persistente da 1 GB (lo storage persiste dopo la fine della sessione)

Per ulteriori informazioni, consulta [Storage persistente](#).

## CloudShell requisiti di rete

### WebSockets

CloudShell dipende dal WebSocket protocollo, che consente la comunicazione interattiva bidirezionale tra il browser Web dell'utente e il CloudShell servizio nel AWS Cloud. Se utilizzi un browser in una rete privata, l'accesso sicuro a Internet è probabilmente facilitato da server proxy e firewall. WebSocket la comunicazione in genere può attraversare i server proxy senza problemi. Ma in alcuni casi, i server proxy WebSockets impediscono di funzionare correttamente. Se si verifica questo problema, la tua CloudShell interfaccia riporta il seguente errore: `Failed to open sessions : Timed out while opening the session.`

Se questo errore si verifica ripetutamente, consulta la documentazione del tuo server proxy per assicurarti che sia configurato per consentire WebSockets. In alternativa, puoi contattare l'amministratore di sistema della tua rete.

### Note

Se desideri definire autorizzazioni granulari autorizzando l'elenco di URL specifici, puoi aggiungere parte dell'URL utilizzato dalla AWS Systems Manager sessione per aprire una WebSocket connessione per l'invio di input e la ricezione di output. (I AWS CloudShell comandi vengono inviati a quella sessione di Systems Manager.)

Il formato StreamUrl utilizzato per questo scopo da Systems Manager è `wss://ssmmessages.region.amazonaws.com/v1/data-channel/session-id?stream=(input|output)`.

La regione rappresenta l'identificatore di regione per una AWS regione supportata da AWS Systems Manager, ad esempio `us-east-2` per la regione degli Stati Uniti orientali (Ohio).

Poiché l'id di sessione viene creato dopo l'avvio corretto di una determinata

sessione di Systems Manager, è possibile specificare solo `wss://`

`ssmmessages.region.amazonaws.com` quando si aggiorna l'elenco di indirizzi URL consentiti. Per ulteriori informazioni, vedete l'[StartSession](#) operazione nell'API Reference.

AWS Systems Manager

## Software preinstallato

### Note

Poiché l'ambiente di AWS CloudShell sviluppo viene aggiornato regolarmente per consentire l'accesso al software più recente, in questa documentazione non forniamo numeri di versione specifici. Descriviamo invece come verificare quale versione è installata. Per verificare la versione installata, inserite il nome del programma seguito dall'`--version` opzione (ad esempio, `git --version`).



## Conchiglie

### Conchiglie preinstallate

Nome	Descrizione	Version information (Informazioni relative alla versione)
Bash	La shell Bash è l'applicazione shell predefinita per. AWS CloudShell	<code>bash --version</code>
PowerShell (pwsh)	Offre un'interfaccia a riga di comando e supporto per il linguaggio di scripting , PowerShell è basato su Microsoft.NET Command Language Runtime. PowerShell utilizza comandi leggeri chiamati cmdlets che accettano e restituiscono oggetti.NET.	<code>pwsh --version</code>
Shell Z (zsh)	La Z Shell, nota anche come zsh, è una versione estesa di Bourne Shell che offre un supporto avanzato per la personalizzazione di temi e plugin.	<code>zsh --version</code>

## AWSinterfacce a riga di comando (CLI)

### CLI

Nome	Descrizione	Version information (Informazioni relative alla versione)
AWS CDKCLI del toolkit	Il AWS CDK Toolkit, il comando <code>CLLcdk</code> , è lo	<code>cdk --version</code>

Nome	Descrizione	Version information (Informazioni relative alla versione)
	<p>strumento principale che interagisce con l'app. AWS CDK Esegue l'app, interroga il modello di applicazione definito e produce e distribuisce i modelli generati da. AWS CloudFormation AWS CDK</p> <p><a href="#">Per ulteriori informazioni, consulta Toolkit. AWS CDK</a></p>	
AWS CLI	<p>AWS CLI È un'interfaccia a riga di comando che è possibile utilizzare per gestire più AWS servizi dalla riga di comando e automatizzarli tramite script. Per ulteriori informazioni, consulta <a href="#">Lavorare conAWS i servizi inAWS CloudShell</a>.</p> <p>Per informazioni su come assicurarti di utilizzare la maggior parte della up-to-date AWS CLI versione 2, consulta. <a href="#">Installazione AWS CLI nella tua home directory</a></p>	<pre>aws --version</pre>

Nome	Descrizione	Version information (Informazioni relative alla versione)
CLI EB	<p>La AWS Elastic Beanstalk CLI fornisce un'interfaccia a riga di comando per semplificare la creazione, l'aggiornamento e il monitoraggio degli ambienti da un repository locale.</p> <p>Per ulteriori informazioni, consulta <a href="#">Using the Elastic Beanstalk command line interface (EB CLI)</a> nella Developer Guide. AWS Elastic Beanstalk</p>	<pre>eb --version</pre>
CLI di Amazon ECS	<p>L'interfaccia a riga di comando (CLI) di Amazon Elastic Container Service (Amazon ECS) fornisce comandi di alto livello per semplificare la creazione, l'aggiornamento e il monitoraggio di cluster e attività.</p> <p>Per ulteriori informazioni, consulta <a href="#">Using the Command Line Interface di Amazon ECS</a> nella Amazon Elastic Container Service Developer Guide.</p>	<pre>ecs-cli --version</pre>

Nome	Descrizione	Version information (Informazioni relative alla versione)
CLI AWS SAM	<p>AWS SAMCLI è uno strumento da riga di comando che funziona su un AWS Serverless Application Model modello e un codice applicativo. È possibile eseguire diverse attività. Questi includono l'invocazione delle funzioni Lambda localmente, la creazione di un pacchetto di distribuzione per l'applicazione serverless e la distribuzione dell'applicazione serverless nel cloud. AWS</p> <p>Per ulteriori informazioni, consulta il <a href="#">riferimento ai comandi AWS SAM CLI</a> nella AWS Serverless Application Model Developer Guide.</p>	<pre>sam --version</pre>

Nome	Descrizione	Version information (Informazioni relative alla versione)
AWS Tools for PowerShell	<p>AWS Tools for PowerShell sono PowerShell moduli basati sulle funzionalità esposte da AWS SDK for .NET. Con AWS Tools for PowerShell, puoi eseguire operazioni di script sulle tue AWS risorse dalla PowerShell riga di comando.</p> <p>AWS CloudShell preinstalla la versione modularizzata (AWS.Tools) di AWS Tools for PowerShell. Per ulteriori informazioni, consulta <a href="#">Using the AWS Tools for PowerShell</a> nella AWS Tools for PowerShell User Guide.</p>	<pre>powershell --Command 'Get-Module -ListAvailable -Name AWS.Tools.Common'</pre>

## Runtime e SDK AWS: Node.js e Python 3

### Runtime e SDK AWS

Nome	Descrizione	Version information (Informazioni relative alla versione)
Node.js (con npm)	<p>Node.js è un JavaScript runtime progettato per semplificare l'applicazione di tecniche di programmazione asincrona. Per ulteriori informazioni, consulta la</p>	<ul style="list-style-type: none"> <li>Node.js: <code>node --version</code></li> <li>npm: <code>npm --version</code></li> </ul>

Nome	Descrizione	Version information (Informazioni relative alla versione)
	<p><a href="#">documentazione sul sito ufficiale Node.js</a>.</p> <p>npm è un gestore di pacchetti che fornisce l'accesso a un registro di JavaScript moduli online. Per ulteriori informazioni, consulta la <a href="#">documentazione sul sito ufficiale di npm</a>.</p>	
SDK per Node.js JavaScript	<p>Il kit di sviluppo software (SDK) aiuta a semplificare la codifica fornendo JavaScript oggetti per i servizi AWS tra cui Amazon S3, Amazon EC2, DynamoDB e Amazon SWF. Per ulteriori informazioni, consulta la <a href="#">Guida per gli sviluppatori di AWS SDK for JavaScript</a>.</p>	<pre>npm -g ls --depth 0 2&gt;/dev/null   grep aws-sdk</pre>

Nome	Descrizione	Version information (Informazioni relative alla versione)
Python	<p>Python 3 è pronto per l'uso nell'ambiente shell. Python 3 è ora considerato la versione predefinita del linguaggio di programmazione (il supporto per Python 2 è terminato a gennaio 2020). Per ulteriori informazioni, consulta la <a href="#">documentazione sul sito ufficiale di Python</a>.</p> <p>Inoltre, è preinstallato pip, l'installatore di pacchetti per Python. È possibile utilizzare questo programma da riga di comando per installare pacchetti Python dagli indici online come Python Package Index. Per ulteriori informazioni, consulta la <a href="#">documentazione fornita dalla Python Packaging Authority</a>.</p>	<ul style="list-style-type: none"><li>• Python 3: <code>python3 --version</code></li><li>• pip: <code>pip3 --version</code></li></ul>

Nome	Descrizione	Version information (Informazioni relative alla versione)
SDK per Python (Boto3)	<p>Boto è il kit di sviluppo software (SDK) utilizzato dagli sviluppatori Python per creare, configurare e gestire Servizi AWS, come Amazon EC2 e Amazon S3. L'SDK fornisce un'API orientata agli oggetti easy-to-use e un accesso a basso livello a Servizi AWS</p> <p><a href="#">Per ulteriori informazioni, consulta la documentazione di Boto3.</a></p>	<code>pip3 list   grep boto3</code>

## Strumenti di sviluppo e utilità shell

### Strumenti di sviluppo e utilità shell

Nome	Descrizione	Version information (Informazioni relative alla versione)
bash-completion	<p>bash-completion è una raccolta di funzioni di shell che consentono il completamento automatico di comandi o argomenti parzialmente digitati premendo il tasto Tab. Puoi trovare i pacchetti supportati da bash-completion in <code>/usr/share/bash-completion/completions</code></p>	<code>dnf info bash-completion</code>



Nome	Descrizione	Version information (Informazioni relative alla versione)
	<p>Per impostare il completamento automatico dei comandi di un pacchetto, è necessario reperire il file di programma . Ad esempio, per impostare il completamento automatico per i comandi Git, aggiungi la seguente riga <code>.bashrc</code> in modo che la funzionalità sia disponibile ogni volta che inizia la AWS CloudShell sessione:</p> <pre>source /usr/share/ bash-completion/ completions/git</pre> <p>Se desideri utilizzare script di completamento personalizzati, aggiungili alla tua home directory persistente (\$HOME) e inseriscili direttamente da essa. <code>.bashrc</code></p> <p>Per ulteriori informazioni, consultate la pagina <a href="#">README</a> del progetto su. GitHub</p>	

Nome	Descrizione	Version information (Informazioni relative alla versione)
CodeCommit utilità per Git	<p>git-remote-codecommit è un'utilità che fornisce un metodo semplice per inviare ed estrarre codice dai CodeCommit repository estendendo Git. È il metodo consigliato per supportare connessioni effettuate con accesso federato, provider di identità e credenziali temporanee.</p> <p>Per ulteriori informazioni, consulta <a href="#">i passaggi di configurazione per le connessioni HTTPS a AWS CodeCommit with git-remote-codecommit nella Guida per l'AWS CodeCommit utente</a>.</p>	<pre>pip3 list   grep git-remote-codecommit</pre>
Git	<p>Git è un sistema di controllo delle versioni distribuito che supporta le moderne pratiche di sviluppo software attraverso flussi di lavoro delle filiali e gestione dei contenuti. Per ulteriori informazioni, consulta la <a href="#">pagina di documentazione sul sito ufficiale di Git</a>.</p>	<pre>git --version</pre>

Nome	Descrizione	Version information (Informazioni relative alla versione)
iputils	Il pacchetto iputils contiene utilità per le reti Linux. <a href="#">Per ulteriori informazioni sulle utilità fornite, consultate il repository iputils su. GitHub</a>	Esempi di uno strumento iputils: <code>arping -V</code>
jq	L'utilità jq analizza i dati in formato JSON per produrre un output modificato dai filtri della riga di comando. <a href="#">Per ulteriori informazioni, consulta il manuale jq ospitato su. GitHub</a>	<code>jq --version</code>
kubectl	kubectl è uno strumento a riga di comando per comunicare con il piano di controllo di un cluster Kubernetes, utilizzando l'API Kubernetes.	<code>kubectl --version</code>
make	L'utilità make viene utilizzata <code>makefiles</code> per automatizzare set di attività e organizzare la compilazione del codice. Per ulteriori informazioni, consultate la documentazione di <a href="#">GNU Make</a> .	<code>make --version</code>

Nome	Descrizione	Version information (Informazioni relative alla versione)
man	Il comando man fornisce pagine di manuale per le utilità e gli strumenti da riga di comando. Ad esempio, <code>man ls</code> restituisce la pagina di manuale del <code>ls</code> comando che elenca il contenuto delle directory. Per ulteriori informazioni, vedere la <a href="#">voce di Wikipedia nella pagina man</a> .	<code>man --version</code>
nano	nano è un editor piccolo e intuitivo per interfacce testuali. Per ulteriori informazioni, consultate la documentazione di <a href="#">GNU nano</a> .	<code>nano --version</code>
procps	procps è un'utilità di amministrazione del sistema che è possibile utilizzare per monitorare e arrestare i processi attualmente in esecuzione. Per ulteriori informazioni, consultate <a href="#">il file README che elenca i programmi che possono essere eseguiti</a> con procps.	<code>ps --version</code>

Nome	Descrizione	Version information (Informazioni relative alla versione)
Client SSH	I client SSH utilizzano il protocollo Secure Shell per le comunicazioni crittografate con un computer remoto. OpenSSH è il client SSH preinstallato. Per ulteriori informazioni, consultate il sito <a href="#">OpenSSH gestito da OpenBSD</a> .	<code>ssh -V</code>
sudo	Con l'utilità sudo, gli utenti possono eseguire un programma con le autorizzazioni di sicurezza di un altro utente, in genere il superutente. Sudo è utile quando è necessario installare applicazioni come amministratore di sistema. Per ulteriori informazioni, consulta il manuale <a href="#">Sudo</a> .	<code>sudo --version</code>
tar	tar è un'utilità da riga di comando che puoi usare per raggruppare più file in un unico file di archivio (spesso chiamato tarball). Per ulteriori informazioni, consultate la documentazione di <a href="#">GNU tar</a> .	<code>tar --version</code>

Nome	Descrizione	Version information (Informazioni relative alla versione)
tmux	tmux è un multiplexer di terminale che puoi usare per eseguire diversi programmi contemporaneamente in più finestre. Per maggiori informazioni, consulta <a href="#">un blog che fornisce una breve introduzione a tmux</a> .	tmux -V
unzip	<a href="#">Per ulteriori informazioni, vedere zip/unzip.</a>	
vim	vim è un editor personalizzabile con cui puoi interagire e tramite un'interfaccia testuale. Per ulteriori informazioni, consulta le <a href="#">risorse di documentazione disponibili</a> su vim.org.	vim --version
wget	wget è un programma informatico utilizzato per recuperare contenuti dai server Web specificati dagli endpoint nella riga di comando. Per ulteriori informazioni, vedere la documentazione di <a href="#">GNU Wget</a> .	wget --version

Nome	Descrizione	Version information (Informazioni relative alla versione)
zip/unzip	<p>Le utilità zip/unzip utilizzano un formato di file di archivio che offre una compressione dei dati senza perdita di dati. Richiamate il comando zip per raggruppare e comprimere i file in un unico archivio. Usa unzip per estrarre i file da un archivio in una directory specificata.</p>	<pre>unzip --version  zip --version</pre>

Nome	Descrizione	Version information (Informazioni relative alla versione)
Docker	<p><a href="#">Docker</a> è una piattaforma aperta per lo sviluppo, la spedizione e l'esecuzione di applicazioni. Docker ti consente di separare le applicazioni dall'infrastruttura in modo da poter distribuire il software rapidamente. Ti consente di creare file Dockerfile all'interno e di creare AWS CloudShell risorse Docker con CDK. <a href="#">Per informazioni su quali regioni sono supportate con Docker, consulta Docker Regions.</a></p> <p>Tieni presente che Docker ha uno spazio limitato nell'ambiente. Se hai immagini singole di grandi dimensioni o troppe immagini Docker preesistenti, ciò può causare problemi. Per ulteriori informazioni su Docker, consulta la guida alla <a href="#">documentazione Docker</a>.</p>	<code>docker --version</code>

## Installazione AWS CLI nella tua home directory

Come il resto del software preinstallato nell' CloudShell ambiente, lo AWS CLI strumento viene aggiornato automaticamente con aggiornamenti pianificati e patch di sicurezza. Se vuoi assicurarti di avere la maggior parte delle up-to-date versioni di AWS CLI, puoi scegliere di installare manualmente lo strumento nella home directory della shell.



**⚠ Important**

È necessario installare manualmente la copia AWS CLI nella home directory in modo che sia disponibile al successivo avvio di una CloudShell sessione. Questa installazione è necessaria perché i file aggiunti alle directory esterne \$HOME vengono eliminati al termine di una sessione di shell. Inoltre, dopo l'installazione AWS CLI, questa copia di non viene aggiornata automaticamente. In altre parole, è tua responsabilità gestire gli aggiornamenti e le patch di sicurezza.

Per ulteriori informazioni sul modello di responsabilità AWS condivisa, vedere [Protezione dei dati in AWS CloudShell](#).

**Per installare AWS CLI**

1. Nella CloudShell riga di comando, utilizzate il `curl` comando per trasferire una copia zippata del file AWS CLI installato nella shell:

```
curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"
```

2. Decomprimi la cartella zippata:

```
unzip awscliv2.zip
```

3. Per aggiungere lo strumento a una cartella specificata, esegui il AWS CLI programma di installazione:

```
sudo ./aws/install --install-dir /home/cloudshell-user/usr/local/aws-cli --bin-dir /home/cloudshell-user/usr/local/bin
```

Se è stato installato correttamente, nella riga di comando viene visualizzato il seguente messaggio:

```
You can now run: /home/cloudshell-user/usr/local/bin/aws --version
```

4. Per comodità, consigliamo di aggiornare anche la variabile di PATH ambiente in modo da non dover specificare il percorso di installazione dello strumento durante l'esecuzione dei `aws` comandi:

```
export PATH=/home/cloudshell-user/usr/local/bin:$PATH
```

### Note

Se annulli questa modifica aPATH, aws i comandi che non presentano un percorso specificato utilizzano per impostazione predefinita la versione preinstallata diAWS CLI.

## Installazione di software di terze parti nell'ambiente shell

### Note

Ti consigliamo di esaminare il [modello di responsabilità di sicurezza condivisa](#) prima di installare applicazioni di terze parti nell'ambiente AWS CloudShell di calcolo in uso.

Per impostazione predefinita, tutti AWS CloudShell gli utenti dispongono delle autorizzazioni sudo. Pertanto, è possibile utilizzare il sudo comando per installare software che non è già disponibile nell'ambiente di calcolo della shell. Ad esempio, è possibile utilizzare l'utilità di gestione dei pacchetti DNF to installcowsay, che genera immagini artistiche in formato ASCII di una mucca sudo con un messaggio:

```
sudo dnf install cowsay
```

È quindi possibile avviare il programma appena installato digitando. echo "Welcome to AWS CloudShell" | cowsay

### Important

Package gestisce utilità come dnf install programs in directory /usr/bin (ad esempio), che vengono riciclate al termine della sessione di shell. Ciò significa che software aggiuntivo viene installato e utilizzato per sessione.

## Modificare la shell con degli script

Se si desidera modificare l'ambiente shell predefinito, è possibile modificare uno script di shell che viene eseguito ogni volta che l'ambiente shell viene avviato. Lo `.bashrc` script viene eseguito ogni volta che viene avviata la shell bash predefinita.

### Warning

Se modificate erroneamente il `.bashrc` file, potreste non essere in grado di accedere al vostro ambiente shell in seguito. È buona norma fare una copia del file prima di modificarlo. È inoltre possibile ridurre il rischio aprendo due shell durante la modifica. `.bashrc` Se perdi l'accesso a una shell, sei ancora connesso all'altra shell e puoi annullare eventuali modifiche. Se perdi l'accesso dopo aver modificato erroneamente `.bashrc` qualsiasi altro file, puoi tornare AWS CloudShell alle impostazioni predefinite [eliminando](#) la tua home directory.

Nella procedura, modificherete `.bashrc` lo script in modo che l'ambiente della shell passi automaticamente all'esecuzione della shell Z.

1. Apri il file `.bashrc` usando un editor di testo (Vim, per esempio):

```
vim .bashrc
```

2. Nell'interfaccia dell'editor, premi il tasto `I` per iniziare a modificare, quindi aggiungi quanto segue:

```
zsh
```

3. Per uscire e salvare il `.bashrc` file modificato, premi `Esc` per accedere alla modalità di comando Vim e inserisci quanto segue:

```
:wq
```

4. Usa il `source` comando per ricaricare il `.bashrc` file:

```
source .bashrc
```

Quando l'interfaccia della riga di comando diventa nuovamente disponibile, il simbolo del prompt è cambiato `%` per indicare che ora stai usando la shell Z.

# AWS CloudShell migrazione da Amazon Linux 2 ad Amazon Linux 2023

AWS CloudShell, che era basato su Amazon Linux 2 (AL2), è migrato ad Amazon Linux 2023 (AL2023). Per ulteriori informazioni su AL2023, consulta [What is Amazon Linux 2023 \(AL2023\)](#) nella Amazon Linux 2023 User Guide.

Con AL2023, puoi continuare ad accedere al tuo CloudShell ambiente esistente con tutti gli strumenti forniti da CloudShell. Per ulteriori informazioni sugli strumenti disponibili, consulta [Software preinstallato](#).

AL2023 offre diversi miglioramenti agli strumenti di sviluppo, incluse le versioni più recenti di pacchetti Node come.js 18 e 3.9. Python

## Note

In AL2023, Python 2 non viene più fornito con l'ambiente in uso. CloudShell

Per ulteriori informazioni sulle principali differenze tra AL2 e AL2023, consulta [Comparazione tra Amazon Linux 2 e Amazon Linux 2023](#) nella Amazon Linux 2023 User Guide.

Se hai domande, contatta [AWS Support](#). Puoi anche cercare risposte e pubblicare domande in [AWS re:Post](#). Quando entri AWS re:Post, ti potrebbe essere richiesto di accedere a AWS.

## AWS CloudShell Domande frequenti sulla migrazione

Di seguito sono riportate le risposte ad alcune domande comuni sulla migrazione da AL2 a AL2023 con AWS CloudShell

- [Questa migrazione influirà su altre mie AWS risorse, come le istanze Amazon EC2 in esecuzione su AL2?](#)
- [Quali sono i pacchetti che verranno modificati con la migrazione ad AL2023?](#)
- [Posso rinunciare alla migrazione?](#)
- [Posso creare un backup del mio AWS CloudShell ambiente?](#)

Questa migrazione influirà su altre mie AWS risorse, come le istanze Amazon EC2 in esecuzione su AL2?

Nessun servizio o risorsa diverso dal tuo AWS CloudShell ambiente è interessato da questa migrazione. Ciò include le risorse che potresti aver creato o a cui hai avuto accesso dall'internoAWS CloudShell. Ad esempio, se hai creato un'istanza Amazon EC2 in esecuzione su AL2, questa non verrà migrata su AL2023.

Quali sono i pacchetti che sono stati modificati con la migrazione ad AL2023?

AWS CloudShell gli ambienti attualmente includono software preinstallato. Per maggiori informazioni sull'elenco completo dei software preinstallati, consulta Software [preinstallato](#). AWS CloudShell continuerà a fornire questi pacchetti, ad eccezione di Python 2. Per la differenza completa tra i pacchetti forniti da AL2 e AL2023, vedi [Confronto tra AL2 e AL2023](#). Per i clienti con requisiti specifici di pacchetto e versione che non saranno più soddisfatti dopo la migrazione ad AL2023, consigliamo di contattare l'AWS assistenza per inviare una richiesta.

Posso rinunciare alla migrazione

La risposta è NO. AWS CloudShell gli ambienti sono gestiti da AWS, pertanto, tutti gli ambienti sono stati aggiornati ad AL2023.

Posso creare un backup del mio ambiente? AWS CloudShell

AWS CloudShell continuerà a mantenere la home directory dell'utente. Per ulteriori informazioni, consulta [Quote e restrizioni del servizio](#) per. AWS CloudShell Se hai file o configurazioni archiviati nella tua cartella home e desideri creare un backup per la stessa, completa il [Passaggio 6: Crea un backup della home directory](#).

# Risoluzione dei problemi relativi a AWS CloudShell

Durante l'utilizzo di AWS CloudShell, è possibile che si verifichino problemi, ad esempio quando si avvia CloudShell o si eseguono attività chiave utilizzando l'interfaccia a riga di comando della shell. Le informazioni contenute in questo capitolo spiegano come risolvere alcuni dei problemi più comuni che potresti riscontrare.

[Per le risposte a una serie di domande in merito CloudShell, consulta le AWS CloudShell domande frequenti.](#) Puoi anche cercare risposte e pubblicare domande nel [Forum di AWS CloudShell discussione](#). Quando accedi a questo forum, ti potrebbe essere richiesto di accedere a AWS. È inoltre possibile [contattarci](#) direttamente.

## Risoluzione degli errori

Quando riscontri uno dei seguenti errori indicizzati, puoi utilizzare le seguenti soluzioni per risolvere questi errori.

### Argomenti

- [Impossibile avviare l'ambiente. Per riprovare, aggiorna il browser o riavvia selezionando Azioni, Riavvia AWS CloudShell](#)
- [Impossibile avviare l'ambiente. Non disponi delle autorizzazioni necessarie. Chiedi al tuo amministratore IAM di concedere l'accesso a AWS CloudShell](#)
- [Impossibile accedere alla riga di AWS CloudShell comando](#)
- [Impossibile eseguire il ping di indirizzi IP esterni](#)
- [Si sono verificati alcuni problemi durante la preparazione del terminale](#)
- [I tasti freccia non funzionano correttamente in PowerShell](#)
- [I Web Socket non supportati impediscono l'avvio delle sessioni CloudShell](#)
- [Impossibile importare il AWSPowerShell.NetCore modulo](#)
- [Docker non è in esecuzione quando si utilizza AWS CloudShell](#)
- [Docker ha esaurito lo spazio su disco](#)
- [docker push è scaduto e continua a riprovare](#)

## Impossibile avviare l'ambiente. Per riprovare, aggiorna il browser o riavvia selezionando Azioni, Riavvia AWS CloudShell

Problema: quando tenti di eseguire l'avvio AWS CloudShell da AWS Management Console, ti viene negato l'accesso anche dopo aver richiesto le autorizzazioni all'amministratore IAM e aver aggiornato il browser o riavviato. CloudShell

Soluzione: contatta l'[AWS Assistenza](#).

[\(Torna all'inizio\)](#)

## Impossibile avviare l'ambiente. Non disponi delle autorizzazioni necessarie. Chiedi al tuo amministratore IAM di concedere l'accesso a AWS CloudShell

Problema: quando tenti di eseguire l'avvio AWS CloudShell da AWS Management Console, ti viene negato l'accesso e ti viene comunicato che non disponi delle autorizzazioni necessarie.

Causa: l'identità IAM che stai utilizzando per accedere AWS CloudShell non dispone delle autorizzazioni IAM necessarie.

Soluzione: richiedi all'amministratore IAM di fornirti le autorizzazioni necessarie. Possono farlo aggiungendo una policy AWS gestita allegata (AWSCloudShellFullAccess) o una policy in linea incorporata. Per ulteriori informazioni, consulta [Gestione dell' AWS CloudShell accesso e dell'utilizzo con le policy IAM](#).

[\(Torna all'inizio\)](#)

## Impossibile accedere alla riga di AWS CloudShell comando

Problema: dopo aver modificato un file utilizzato dall'ambiente di calcolo, non è possibile accedere alla riga di comando in. AWS CloudShell

Soluzione: se perdi l'accesso dopo aver modificato erroneamente `.bashrc` qualsiasi altro file, puoi tornare AWS CloudShell alle impostazioni predefinite [eliminando](#) la tua home directory.

[\(Torna all'inizio\)](#)

## Impossibile eseguire il ping di indirizzi IP esterni

Problema: quando si esegue un comando ping dalla riga di comando (ad esempio, `ping amazon.com`), viene visualizzato il seguente messaggio.

```
ping: socket: Operation not permitted
```

**Causa:** l'utilità ping utilizza Internet Control Message Protocol (ICMP) per inviare pacchetti di richieste echo a un host di destinazione. Attende la risposta di un'eco dalla destinazione. Poiché il protocollo ICMP non è abilitato in AWS CloudShell, l'utilità ping non funziona nell'ambiente di calcolo della shell.

**Soluzione:** poiché ICMP non è supportato in AWS CloudShell, è possibile eseguire il seguente comando per installare Netcat. Netcat è un'utilità di rete informatica per la lettura e la scrittura su connessioni di rete tramite TCP o UDP.

```
sudo yum install nc
nc -zv www.amazon.com 443
```

[\(Torna all'inizio\)](#)

## Si sono verificati alcuni problemi durante la preparazione del terminale

**Problema:** quando si tenta di accedere AWS CloudShell utilizzando il browser Microsoft Edge, non è possibile avviare una sessione di shell e il browser visualizza un messaggio di errore.

**Causa:** AWS CloudShell non è compatibile con le versioni precedenti di Microsoft Edge. Puoi accedere AWS CloudShell utilizzando le ultime quattro versioni principali dei [browser supportati](#).

**Soluzione:** installa una versione aggiornata del browser Edge dal [sito Microsoft](#).

[\(Torna all'inizio\)](#)

## I tasti freccia non funzionano correttamente in PowerShell

**Problema:** durante il normale funzionamento, puoi usare i tasti freccia per navigare nell'interfaccia della riga di comando ed eseguire la scansione avanti e indietro nella cronologia dei comandi. Tuttavia, quando si premono i tasti freccia in alcune versioni di PowerShell on AWS CloudShell, le lettere potrebbero essere emesse in modo errato.

**Causa:** la situazione in cui i tasti freccia emettono erroneamente le lettere è un problema noto delle versioni PowerShell 7.2.x in esecuzione su Linux.

**Soluzione:** per eliminare le sequenze di escape che modificano il comportamento dei tasti freccia, modificate il file di PowerShell profilo e impostate la variabile su. `$PSStyle PlainText`



1. Nella AWS CloudShell riga di comando, immettete il seguente comando per aprire il file di profilo.

```
vim ~/.config/powershell/Microsoft.PowerShell_profile.ps1
```

#### Note

Se l'hai già fatto PowerShell, puoi anche aprire il file del profilo nell'editor con il seguente comando.

```
vim $PROFILE
```

2. Nell'editor, vai alla fine del testo esistente del file, premi i per accedere alla modalità di inserimento, quindi aggiungi la seguente istruzione.

```
$PSStyle.OutputRendering = 'PlainText'
```

3. Dopo aver apportato la modifica, premi Esc per accedere alla modalità di comando. Quindi, inserisci il seguente comando per salvare il file e uscire dall'editor.

```
:wq
```

#### Note

Le modifiche avranno effetto al prossimo avvio PowerShell.

[\(Torna all'inizio\)](#)

## I Web Socket non supportati impediscono l'avvio delle sessioni CloudShell

**Problema:** Quando si tenta di avviare AWS CloudShell, si riceve ripetutamente il seguente messaggio: `Failed to open sessions : Timed out while opening the session`

**Causa:** CloudShell dipende dal WebSocket protocollo, che consente la comunicazione interattiva bidirezionale tra il browser web e AWS CloudShell. Se utilizzi un browser in una rete privata, l'accesso sicuro a Internet è probabilmente facilitato da server proxy e firewall. WebSocket la comunicazione in genere può attraversare i server proxy senza problemi. Tuttavia, in alcuni casi, i server proxy

WebSockets impediscono di funzionare correttamente. Se si verifica questo problema, non è possibile avviare una sessione di shell e il tentativo di connessione alla fine scade.

Soluzione: un timeout di connessione potrebbe essere causato da un problema diverso da quello non WebSockets supportato. In tal caso, aggiorna innanzitutto la finestra del browser in cui si trova l'interfaccia della CloudShell riga di comando.

Se continui a ricevere errori di timeout dopo l'aggiornamento, consulta la documentazione del tuo server proxy. Inoltre, assicurati che il tuo server proxy sia configurato per consentire i Web Socket. In alternativa, contattate l'amministratore di sistema della rete.

#### Note

Supponiamo che tu voglia definire autorizzazioni granulari consentendo l'elenco di URL specifici. Puoi aggiungere parte dell'URL utilizzato dalla AWS Systems Manager sessione per aprire una WebSocket connessione per l'invio di input e la ricezione di output. I AWS CloudShell comandi vengono inviati a quella sessione di Systems Manager.

Il formato utilizzato per StreamUrl questo scopo da Systems Manager è `wss://ssmmessages.region.amazonaws.com/v1/data-channel/session-id?stream=(input|output)`.

La regione rappresenta l'identificatore di regione per un Regione AWS uomo supportato da AWS Systems Manager. Ad esempio, `us-east-2` è l'identificatore della regione degli Stati Uniti orientali (Ohio).

Poiché l'id di sessione viene creato dopo l'avvio corretto di una determinata sessione di Systems Manager, è possibile specificare solo `wss://ssmmessages.region.amazonaws.com` quando si aggiorna l'elenco degli URL consentiti. Per ulteriori informazioni, vedete l'[StartSession](#) operazione nell'API Reference. AWS Systems Manager

[\(Torna all'inizio\)](#)

## Impossibile importare il **AWSPowerShell.NetCore** modulo

Problema: quando si importa il `AWSPowerShell.NetCore` modulo in PowerShell con `Import-Module -Name AWSPowerShell.NetCore`, viene visualizzato il seguente messaggio di errore:

Import-Module: il modulo specificato 'AWSPowerShell.NetCore' non è stato caricato perché non è stato trovato alcun file di modulo valido in nessuna directory del modulo.

**Causa:** il `AWSPowerShell.NetCore` modulo viene sostituito dai moduli `AWS.Tools` per servizio in `AWS CloudShell`

**Soluzione:** è possibile che eventuali istruzioni di importazione esplicite non siano più necessarie o debbano essere modificate nel modulo `.Tools` correlato per servizio. `AWS`

Example

Example

- Nella maggior parte dei casi, finché non vengono utilizzati tipi `.Net`, non è necessaria alcuna dichiarazione di importazione esplicita. Di seguito sono riportati alcuni esempi di istruzioni di importazione.
  - `Get-S3Bucket`
  - `(Get-EC2Instance).Instances`
- Se vengono utilizzati tipi `.Net`, importate il modulo a livello di servizio `()AWS.Tools.<Service>`. Di seguito è riportato un esempio di sintassi.

```
Import-Module -Name AWS.Tools.EC2
$instanceTag = [Amazon.EC2.Model.Tag]::new("Environment", "Dev")
```

```
Import-Module -Name AWS.Tools.S3
$lifecycleRule = [Amazon.S3.Model.LifecycleRule]::new()
```

Per ulteriori informazioni, vedi l'[annuncio della versione 4](#) per `AWS Tools for PowerShell`

[\(Torna all'inizio\)](#)

## Docker non è in esecuzione quando si utilizza `AWS CloudShell`

**Problema:** Docker non funziona correttamente durante l'utilizzo. `AWS CloudShell` Viene visualizzato il seguente messaggio di errore:`docker: Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?.`

**Soluzione:** prova a riavviare l'ambiente. Questo messaggio di errore può verificarsi quando esegui `Docker AWS CloudShell` in una regione che non lo supporta. [Assicurati di utilizzare Docker in una](#)

[regione supportata. Per informazioni su quali regioni supportano l'uso dei contenitori DockerAWS CloudShell, consulta Docker Regions.](#)

## Docker ha esaurito lo spazio su disco

Problema: Stai ricevendo il seguente messaggio di errore: `ERROR: failed to solve: failed to register layer: write [...]: no space left on device.`

Causa: Il Dockerfile sta superando lo spazio disponibile su disco in AWS CloudShell. Ciò può essere causato da singole immagini di grandi dimensioni o da troppe immagini Docker preesistenti.

Soluzione: esegui `df -h` per trovare l'utilizzo del disco. Esegui `sudo du -sh /folder/folder1` per valutare le dimensioni di alcune cartelle che ritieni possano essere grandi e valuta la possibilità di eliminare altri file per liberare spazio. Un'opzione potrebbe essere quella di prendere in considerazione la rimozione delle immagini Docker inutilizzate eseguendole. `docker rmi` [Tieni presente che Docker ha uno spazio limitato nell'ambiente, per ulteriori informazioni su Docker, consulta la guida alla documentazione di Docker.](#)

## **docker push** è scaduto e continua a riprovare

Problema: quando si esegue `docker push`, il timeout è scaduto e si continua a riprovare senza successo.

Causa: ciò può essere causato dalla mancanza di autorizzazioni, dal trasferimento al repository sbagliato o dalla mancanza di autenticazione.

Soluzione: per provare a risolvere il problema, assicurati di eseguire il push nel repository corretto. Esegui `docker login` per autenticarti correttamente. Assicurati di disporre di tutte le autorizzazioni necessarie per il trasferimento a un repository Amazon ECR.

# Browser supportati per AWS CloudShell

La tabella seguente elenca i browser supportati per AWS CloudShell.

## Supporto browser Web

Browser	Versione
Google Chrome	Le ultime tre versioni principali
Mozilla Firefox	Le ultime tre versioni principali
Microsoft Edge	Le ultime tre versioni principali
Apple Safari per macOS	Le ultime due versioni principali

# AWS Regioni supportate per AWS CloudShell

Questa sezione riporta l'elenco delle AWS regioni supportate e delle regioni opt-in per AWS CloudShell. Per un elenco degli endpoint e delle quote di AWS servizio per CloudShell, consulta la [AWS CloudShell pagina](#) in. Riferimenti generali di Amazon Web Services

Di seguito sono elencate le AWS regioni supportate per: AWS CloudShell

- Stati Uniti orientali (Ohio)
- Stati Uniti orientali (Virginia settentrionale)
- Stati Uniti occidentali (California settentrionale)
- Stati Uniti occidentali (Oregon)
- Asia Pacifico (Mumbai)
- Asia Pacifico (Osaka)
- Asia Pacifico (Seul)
- Asia Pacifico (Sydney)
- Asia Pacifico (Singapore)
- Asia Pacifico (Tokyo)
- Canada (Centrale)
- Europa (Francoforte)
- Europa (Irlanda)
- Europa (London)
- Europa (Paris)
- Europa (Stoccolma)
- Sud America (San Paolo)

## GovCloud Regioni

Le seguenti sono le GovCloud regioni supportate per CloudShell:

- AWS GovCloud (Stati Uniti orientali)
- AWS GovCloud (Stati Uniti occidentali)

## Regioni con consenso esplicito

Il consenso per l'utilizzo delle regioni non è attivato per impostazione predefinita. È necessario abilitare manualmente queste regioni per utilizzarle. Per ulteriori informazioni, consulta [Managing AWS Regions](#). Le seguenti sono le regioni Opt-in supportate per CloudShell:

- Africa (Città del Capo)
- Asia Pacifico (Hong Kong)
- Asia Pacifico (Giacarta)
- Europa (Milano)
- Medio Oriente (Bahrein)
- Medio Oriente (Emirati Arabi Uniti)

## Regioni supportate per Docker

L'ambiente di AWS CloudShell elaborazione supporta solo contenitori Docker nelle seguenti regioni:

- Stati Uniti orientali (Ohio)
- Stati Uniti orientali (Virginia settentrionale)
- Stati Uniti occidentali (Oregon)
- Asia Pacifico (Mumbai)
- Asia Pacifico (Sydney)
- Asia Pacifico (Singapore)
- Asia Pacifico (Tokyo)
- Canada (Centrale)
- Europa (Francoforte)
- Europa (Irlanda)
- Europe (London)
- Europa (Parigi)
- Sud America (San Paolo)

# Quote e restrizioni di servizio perAWS CloudShell

Questa pagina descrive le quote e le restrizioni del Servizio che si applicano alle seguenti aree:

- [Archiviazione permanente](#)
- [Utilizzo mensile](#)
- [Dimensione del comando](#)
- [Conchiglie simultanee](#)
- [Sessioni Shell](#)
- [Accesso alla rete e trasferimento dati](#)
- [Ricaricamento di file di sistema e pagine](#)

## Storage persistente

ConAWS CloudShell, hai a disposizione uno spazio di archiviazione persistente di 1 GB per ciascuno senza alcunRegione AWS costo. L'archiviazione persistente si trova nella tua home directory (\$HOME) ed è privata per te. A differenza delle risorse ambientali effimere che vengono riciclate al termine di ogni sessione shell, i dati nella home directory persistono tra le sessioni.

Se interrompi l'utilizzoAWS CloudShell in unRegione AWS, i dati vengono conservati nella memoria persistente di quella Regione per 120 giorni dopo la fine dell'ultima sessione. Dopo 120 giorni, a meno che tu non intervenga, i tuoi dati vengono automaticamente eliminati dalla memoria persistente di quella regione. Puoi impedirne la rimozione avviandoloAWS CloudShell nuovamente in quella finestraRegione AWS. Per ulteriori informazioni, consulta la [Fase 2: Seleziona una regione, avvia AWS CloudShell e scegli una shell](#).

### Note

#### Scenario di utilizzo

Márcia è solitaAWS CloudShell archiviare i file nelle sue directory di casa e in EuropaRegioni AWS (Irlanda) e in Europa (Irlanda) e in Europa (Irlanda). Ha quindi iniziato a utilizzareAWS CloudShell esclusivamente in Europa (Irlanda) e ha smesso di lanciare sessioni di shell negli Stati Uniti e in altri Stati Uniti (Virginia settentrionale) e ha smesso di lanciare sessioni di shell negli Stati Uniti e in Europa (Virginia).



Prima della scadenza per l'eliminazione dei dati negli Stati Uniti orientali (Virginia settentrionale), Márcia decide di impedire che la sua home directory venga riciclata avviando AWS CloudShell e selezionando nuovamente la regione Stati Uniti orientali (Virginia settentrionale). Poiché ha utilizzato continuamente l'Europa (Irlanda) per le sessioni di shell, la sua archiviazione persistente in quella regione non ne risente.

## Utilizzo mensile

Sono previste quote di utilizzo mensili AWS CloudShell per ciascuno Regione AWS dei tuoi Account AWS. Se si tenta di accedere AWS CloudShell dopo aver raggiunto la quota mensile per quella regione, viene visualizzato un messaggio che spiega perché l'ambiente shell non può essere avviato.

### Note

Se devi aumentare le quote di utilizzo mensili, contatta [AWS Support](#).

## Dimensioni comando

La dimensione del comando non può superare i 65412 caratteri.

### Note

Se intendi eseguire il comando che supera i 65412 caratteri, crea uno script con la lingua che preferisci, quindi esegilo dall'interfaccia della riga di comando. Per ulteriori informazioni sulla gamma di software preinstallati a cui è possibile accedere dall'interfaccia a riga di comando, vedere [Software preinstallato](#).

Per vedere un esempio di come creare uno script e quindi eseguirlo dall'interfaccia a riga di comando, vedi [Tutorial: Getting started with AWS CloudShell](#).

## Conchiglie simultanee

- Conchiglie simultanee: puoi utilizzare fino a 10 shell contemporaneamente in ciascuna di esse Regione AWS per il tuo account.

## Sessioni Shell

- Sessione di shell: AWS CloudShell è un ambiente di shell interattivo: se non si interagisce con la tastiera o il puntatore per 20-30 minuti, la sessione di shell termina. I processi in esecuzione non sono considerati interazioni.
- Sessioni di lunga durata: una sessione shell che viene eseguita ininterrottamente per circa 12 ore termina automaticamente anche se l'utente interagisce regolarmente con essa durante quel periodo.

## Accesso alla rete e trasferimento dati

Le seguenti restrizioni si applicano sia al traffico in entrata che in uscita del tuo AWS CloudShell ambiente:

- In uscita: è possibile accedere alla rete Internet pubblica: è possibile accedere alla rete Internet pubblica.
- In entrata: non puoi accedere alle porte in entrata. Non è disponibile alcun indirizzo IP pubblico.

### Warning

Con l'accesso alla rete Internet pubblica, esiste il rischio che alcuni utenti possano esportare dati dall'AWS CloudShell ambiente. Consigliamo agli amministratori IAM di gestire l'elenco degli AWS CloudShell utenti autorizzati tramite gli strumenti IAM. Per informazioni su come negare esplicitamente l'accesso a determinati utenti, vedere [Gestione delle azioni consentite nell' AWS CloudShell utilizzo di policy personalizzate](#).

Trasferimento dati: il caricamento e il download di file da e verso file di grandi dimensioni AWS CloudShell potrebbero essere lenti. In alternativa, puoi trasferire file nel tuo ambiente da un bucket Amazon S3 utilizzando l'interfaccia a riga di comando della shell.

## Restrizioni sui file di sistema e sul ricaricamento delle pagine

- File di sistema: se si modificano erroneamente i file richiesti dall'ambiente di elaborazione, potrebbero verificarsi problemi durante l'accesso o l'utilizzo dell'AWS CloudShell ambiente. In tal caso, potrebbe essere necessario [eliminare la home directory](#) per riottenere l'accesso.

- Ricaricamento delle pagine: per ricaricare l'AWS CloudShellinterfaccia, utilizza il pulsante di aggiornamento nel browser anziché la sequenza di tasti di scelta rapida predefinita per il tuo sistema operativo.

# Cronologia dei documenti per la Guida per l'utente di AWS CloudShell

## Aggiornamenti recenti

Nella tabella seguente sono descritte le modifiche importanti apportate alla Guida per l'utente di AWS CloudShell.

Modifica	Descrizione	Data
<a href="#">Sono stati aggiunti nuovi tutorial alla Guida per l'utente AWS CloudShell</a>	Sono stati aggiunti due nuovi tutorial che spiegano come creare un contenitore Docker all'interno AWS CloudShell e inviarlo a un repository Amazon ECR e come implementare una funzione Lambda tramite AWS CDK	27 dicembre 2023
<a href="#">Contenitori Docker supportati in alcune regioni AWS CloudShell</a>	Il supporto per i contenitori Docker con AWS CloudShell è stato aggiunto in alcune regioni.	27 dicembre 2023
<a href="#">AWS CloudShell è migrato per utilizzare ora Amazon Linux 2023 (AL2023)</a>	AWS CloudShell ora utilizza AL2023 ed è migrato da Amazon Linux 2.	4 dicembre 2023
<a href="#">Nuove regioni AWS per AWS CloudShell</a>	AWS CloudShell è ora disponibile a livello generale nelle seguenti AWS regioni: <ul style="list-style-type: none"><li>• Stati Uniti occidentali (California settentrionale)</li><li>• Africa (Città del Capo)</li><li>• Asia Pacifico (Hong Kong)</li></ul>	16 giugno 2023

- Asia Pacifico (Osaka-Lo cale)
- Asia Pacific (Seul)
- Asia Pacifico (Giacarta)
- Asia Pacifico (Singapore)
- Europa (Parigi)
- Europe (Stockholm)
- Europe (Milan)
- Medio Oriente (Bahrein)
- Medio Oriente (Emirati Arabi Uniti)

[AWS CloudShellAvvia su Console Toolbar](#)

Avvia CloudShell suConsole Toolbar, nella parte inferiore sinistra della console scegliendo CloudShell.

28 marzo 2023

[Nuove AWS regioni per AWS CloudShell](#)

AWS CloudShellè ora disponibile nelle seguenti AWS regioni:

6 ottobre 2022

- Canada (Centrale)
- Europa (Londra)
- Sud America (San Paolo)

[AWS CloudShellsupportato negli Stati Uniti in AWS GovCloud](#)

AWS CloudShellè ora supportato nella regione AWS GovCloud (Stati Uniti).

29 giugno 2022

[Domande frequenti sulla sicurezza](#)

Domande frequenti aggiuntive e incentrate sui problemi di sicurezza.

14 aprile 2022

---

<a href="#">Web Socket</a>	È stata aggiunta una sezione ai requisiti di rete che spiega l'uso CloudShell del WebSocket protocollo.	21 marzo 2022
<a href="#">Risoluzione dei problemi dei tasti freccia in PowerShell</a>	Segui i passaggi per corregger e i tasti freccia che emettono in modo errato le lettere quando vengono premuti.	7 febbraio 2022
<a href="#">Completamento automatico del tasto Tab</a>	Nuova documentazione che spiega come usare bash-completion, che consente il completamento automatico di comandi o argomenti parzialmente digitati premendo il tasto Tab.	24 settembre 2021
<a href="#">Specificare le regioni AWS</a>	Documentazione sulla specificazione dei valori predefiniti Regione AWS per i comandi AWS CLI.	11 maggio 2021
<a href="#">Formattazione nelle versioni PDF e Kindle</a>	Dimensioni fisse dell'immagine e testo nelle celle della tabella.	10 marzo 2021

[Versione a disponibilità generale \(GA\) di AWS CloudShell in AWS regioni selezionate](#)

AWS CloudShell è ora disponibile a livello generale nelle seguenti AWS regioni:

15 dicembre 2020

- Stati Uniti orientali (Ohio)
- Stati Uniti orientali (Virginia settentrionale)
- Stati Uniti occidentali (Oregon)
- Asia Pacifico (Tokyo)
- Europa (Irlanda)
- Asia Pacifico (Mumbai)
- Asia Pacifico (Sydney)
- Europa (Francoforte)

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.