



Guida per l'utente

# AWS CloudShell



# AWS CloudShell: Guida per l'utente

Copyright © 2026 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 dei rispettivi proprietari, che possono o meno essere affiliati, collegati o sponsorizzati da Amazon.

---

# Table of Contents

Che cos'è AWS CloudShell? .....	1
Caratteristiche di AWS CloudShell .....	1
AWS Command Line Interface .....	2
Shell e strumenti di sviluppo .....	2
Storage persistente .....	2
CloudShell Ambienti VPC .....	3
Sicurezza .....	3
Opzioni di personalizzazione .....	3
Ripristino della sessione .....	4
Prezzi per AWS CloudShell .....	4
AWS CloudShell Argomenti chiave .....	4
Nozioni di base .....	5
Prerequisiti .....	5
Indice .....	6
Passaggio 1: accedi a Console di gestione AWS .....	6
Passo 2: Seleziona una regione AWS CloudShell, avvia e scegli una shell .....	7
Passaggio 3: scarica un file da AWS CloudShell .....	10
Passaggio 4: carica un file su AWS CloudShell .....	11
Fase 5: Rimuovere un file da AWS CloudShell .....	12
Passaggio 6: Creare un backup della home directory .....	12
Passaggio 7: riavviare una sessione di shell .....	14
Passaggio 8: Eliminare la home directory di una sessione di shell .....	15
Passaggio 9: Modifica il codice del file ed eseguilo utilizzando la riga di comando .....	16
Passaggio 10: utilizzare AWS CLI per aggiungere il file come oggetto in un bucket Amazon S3 .....	18
Argomenti correlati .....	19
Tutorial .....	20
Tutorial: Copiare più file .....	20
Caricamento e download di più file con Amazon S3 .....	21
Caricamento e download di più file utilizzando cartelle zippate .....	24
Tutorial: creazione di predefiniti URLs .....	26
Prerequisiti .....	26
Fase 1: creare un ruolo IAM per concedere l'accesso al bucket Amazon S3 .....	26
Genera l'URL predefinito .....	27

Tutorial: creazione di un contenitore Docker all'interno CloudShell e trasferimento ad Amazon ECR .....	29
Prerequisiti .....	29
Procedura tutorial .....	29
Eliminazione .....	32
Tutorial: distribuzione di una funzione Lambda utilizzando AWS CDK .....	32
Prerequisiti .....	32
Procedura del tutorial .....	32
Eliminazione .....	35
AWS CloudShell Concetti .....	36
Navigazione nell'interfaccia AWS CloudShell .....	36
Lavorare in Regioni AWS .....	37
Specificare l'impostazione predefinita per Regione AWS AWS CLI .....	38
Utilizzo dei file e dello spazio di archiviazione .....	39
Accesso CloudShell nell'applicazione Console Mobile .....	40
Utilizzo di Docker .....	40
Funzionalità di accessibilità .....	42
Navigazione tramite tastiera in CloudShell .....	42
CloudShell funzionalità di accessibilità del terminale .....	42
Scelta delle dimensioni dei caratteri e dei temi dell'interfaccia in CloudShell .....	42
Gestisci i servizi AWS .....	44
AWS CLI esempi di riga di comando per servizi selezionati AWS .....	44
DynamoDB .....	45
Amazon EC2 .....	45
Amazon Glacier .....	45
AWS CLI Elastic Beanstalk .....	46
CLI di Amazon ECS .....	46
AWS SAM CLI .....	47
Ingresso CLI Kiro CloudShell .....	48
Usare il comando chat di Kiro in CloudShell .....	48
Usare il comando Kiro translate in CloudShell .....	48
Completamento del comando CLI in CloudShell .....	48
I suggerimenti in linea di Kiro in CloudShell .....	49
Policy basata sull'identità per Kiro CLI in CloudShell .....	49
Esecuzione di un comando CloudShell dalle console AWS di servizio .....	50
Personalizzazione AWS CloudShell .....	52

Suddivisione della visualizzazione della riga di comando in più schede .....	52
Modifica della dimensione del carattere .....	53
Modifica del tema dell'interfaccia .....	53
Utilizzo di Safe Paste per il testo su più righe .....	53
Utilizzo tmux per ripristinare la sessione .....	54
.....	54
Utilizzo AWS CloudShell in Amazon Virtual Private Cloud (Amazon VPC) .....	55
Vincoli operativi .....	55
Creazione di un CloudShell ambiente VPC .....	56
Autorizzazioni IAM richieste per la creazione e l'utilizzo di ambienti CloudShell VPC .....	57
Policy IAM che garantisce CloudShell l'accesso completo, incluso l'accesso al VPC .....	58
Utilizzo delle chiavi di condizione IAM per ambienti VPC .....	61
Policy di esempio con chiavi di condizione per le impostazioni VPC .....	62
Sicurezza .....	3
Protezione dei dati .....	67
Crittografia dei dati .....	68
Identity and Access Management .....	68
Destinatari .....	69
Autenticazione con identità .....	69
Gestione dell'accesso tramite policy .....	71
In che modo AWS CloudShell funziona con IAM .....	72
Identity-based esempi di policy .....	78
Risoluzione dei problemi .....	81
Gestione AWS CloudShell accesso e utilizzo con le policy IAM .....	83
Registrazione di log e monitoraggio .....	97
Monitoraggio dell'attività con CloudTrail .....	97
AWS CloudShell in CloudTrail .....	97
Convalida della conformità .....	100
Resilienza .....	105
Sicurezza dell'infrastruttura .....	105
Best practice di sicurezza .....	106
Domande frequenti sulla sicurezza .....	107
What AWS i processi e le tecnologie vengono utilizzati quando si avvia CloudShell e si avvia una sessione di shell? .....	107
È possibile limitare l'accesso alla rete a CloudShell? .....	107
Posso personalizzare il mio CloudShell ambiente? .....	108

Dov'è il mio \$CASA directory effettivamente memorizzata nel Cloud AWS? .....	108
È possibile crittografare il mio \$CASA elenco? .....	108
Posso eseguire una scansione antivirus sul mio \$CASA elenco? .....	108
Posso limitare l'ingresso o l'uscita dei dati per me? CloudShell .....	109
Posso limitare l'accesso root a un CloudShell utente all'interno del contenitore? .....	109
AWS CloudShell ambiente di calcolo .....	110
Risorse dell'ambiente di calcolo .....	110
CloudShell requisiti di rete .....	110
Software preinstallato .....	111
Conchiglie .....	112
AWS interfacce a riga di comando (CLI) .....	112
Runtimes e AWS SDKs: Node.js e Python 3 .....	116
Strumenti di sviluppo e utilità shell .....	119
Installazione AWS CLI nella tua home directory .....	128
Installazione di software di terze parti nell'ambiente shell .....	129
Modificare la shell con degli script .....	130
Migrazione da Amazon Linux 2 ad Amazon Linux 2023 .....	131
AWS CloudShell Migrazione FAQs .....	132
Risoluzione dei problemi .....	134
Risoluzione degli errori .....	134
Accesso negato .....	135
Autorizzazioni insufficienti .....	135
Impossibile accedere AWS CloudShell riga di comando .....	135
Impossibile eseguire il ping di indirizzi IP esterni .....	135
Si sono verificati alcuni problemi durante la preparazione del terminale .....	136
I tasti freccia non funzionano correttamente in PowerShell .....	136
I Web Socket non supportati impediscono l'avvio delle sessioni CloudShell .....	137
Impossibile importare il AWSPowerShell.NetCore modulo .....	138
Docker non è in esecuzione quando si utilizza AWS CloudShell .....	139
Docker ha esaurito lo spazio su disco .....	140
docker push sta scadendo e continua a riprovare .....	140
Impossibile accedere alle risorse all'interno del VPC dal mio AWS CloudShell Ambiente VPC .....	140
L'ENI utilizzato da AWS CloudShell perché il mio ambiente VPC non è stato pulito .....	141
L'utente con CreateEnvironment autorizzazione solo per gli ambienti VPC ha anche accesso al pubblico AWS CloudShell ambienti .....	141

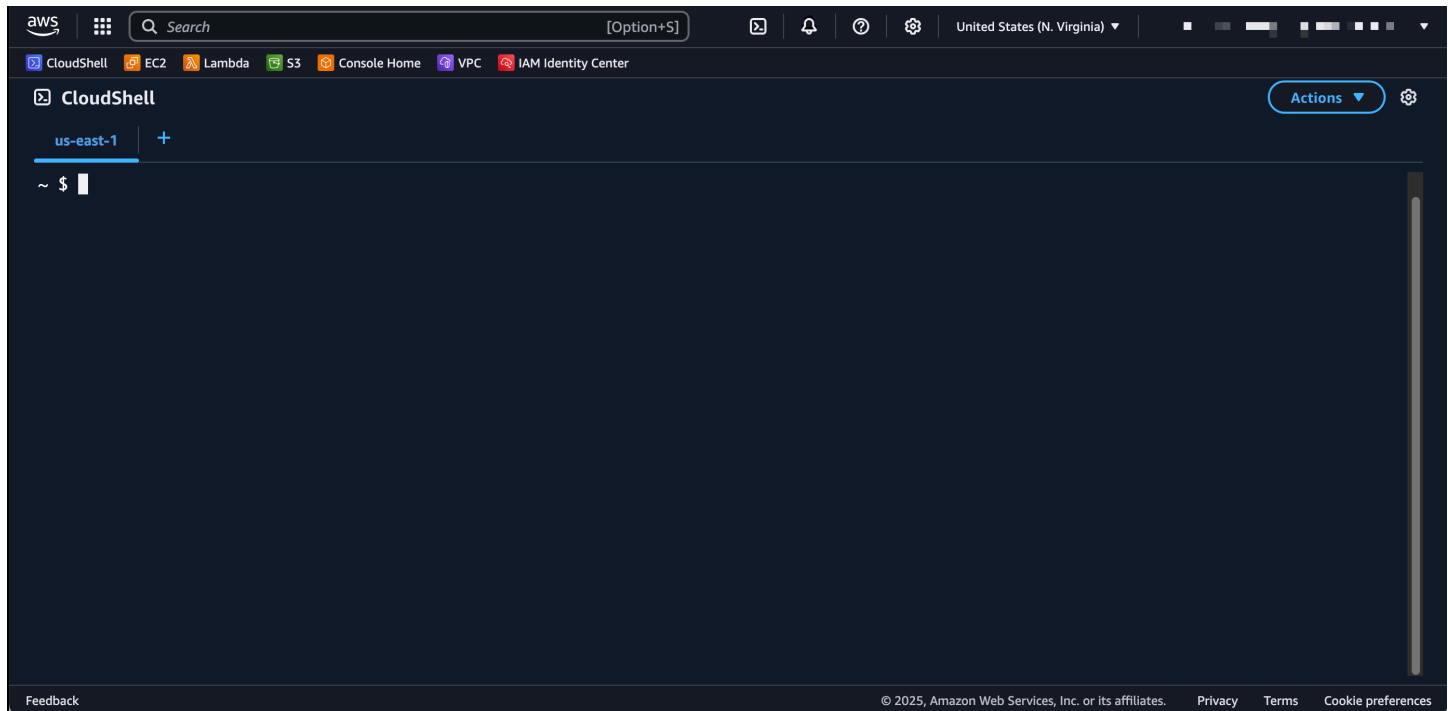
---

Regioni supportate .....	143
Quote e restrizioni del servizio .....	144
Storage persistente .....	144
Utilizzo mensile .....	145
Shell simultanee .....	145
Dimensione del comando .....	146
Sessioni di shell .....	146
Ambienti VPC .....	146
Accesso alla rete e trasferimento dei dati .....	147
Restrizioni sui file di sistema e sui ricaricamenti delle pagine .....	147
Cronologia dei documenti .....	148
.....	cli

# Che cos'è AWS CloudShell?

AWS CloudShell è una shell preautenticata basata su browser che è possibile avviare direttamente da Console di gestione AWS. È possibile accedere CloudShell da diversi modi Console di gestione AWS. Per ulteriori informazioni, consulta [Guida introduttiva a AWS CloudShell](#)

È possibile eseguire AWS CLI comandi utilizzando la shell preferita, ad esempio Bash PowerShell, oZ 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. Puoi utilizzarlo CloudShell nelle versioni più recenti dei browser Google Chrome, Mozilla Firefox, Microsoft Edge e Apple Safari.

(Provalo subito: [Guida introduttiva con AWS CloudShell](#))

## Caratteristiche di AWS CloudShell

AWS CloudShell offre le seguenti caratteristiche:

## AWS Command Line Interface

È possibile eseguire il lancio AWS CloudShell da Console di gestione AWS. 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 l' Servizi AWS uso dell'interfaccia a riga di comando, vedere. [Gestisci AWS i servizi dalla CLI in 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.

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 Regione AWS in ciascuna unità senza 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](#)

### Note

CloudShell Gli ambienti VPC non dispongono di storage persistente. La directory \$HOME viene eliminata quando l'ambiente VPC scade (dopo 20-30 minuti di inattività) o quando si elimina o si riavvia l'ambiente.

## CloudShell Ambienti VPC

AWS CloudShell il cloud privato virtuale (VPC) ti consente di creare un CloudShell ambiente nel tuo VPC. Per ogni ambiente VPC, puoi assegnare un VPC, aggiungere una sottorete e associare uno o più gruppi di sicurezza. AWS CloudShell eredita la configurazione di rete del VPC e consente di AWS CloudShell utilizzarlo in modo sicuro all'interno della stessa sottorete delle altre risorse del VPC.

## Sicurezza

L' AWS CloudShell ambiente e i suoi utenti sono protetti da funzionalità di sicurezza specifiche. 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 AWS CloudShell accesso e 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 di 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 il 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 il AWS CloudShell esperienza](#).

È inoltre possibile estendere l'ambiente shell [installando il proprio software](#) e [modificando](#) la shell con degli script.

## 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 di servizio e restrizioni per AWS CloudShell](#).

## AWS CloudShell Argomenti chiave

- [Guida introduttiva con AWS CloudShell](#)
- [AWS CloudShell Concetti](#)
- [Gestisci AWS i servizi dalla CLI in CloudShell](#)
- [Personalizzare il AWS CloudShell esperienza](#)
- [AWS CloudShell ambiente di calcolo: specifiche e software](#)

# Guida introduttiva con 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 Console di gestione AWS 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 Guida per l'utente di Amazon Simple Storage Service.

### 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 Console di gestione AWS](#)
- [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 Console di gestione AWS

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

- Puoi accedervi Console di gestione AWS 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.

## Main sign-in page

- Aprire <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.
- 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 AWSIn tal caso, inserisci le tue credenziali di accesso IAM e scegli Accedi.

### 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 regione AWS CloudShell, avvia e scegli una shell

In questo passaggio, si avvia 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 [AWS regione supportata](#) in cui lavorare. (Le Regioni disponibili sono evidenziate.)

### Important

Se cambi regione, l'interfaccia si aggiorna e il nome della regione 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 su Console 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, vedere [Working in Regioni AWS](#).

### Example

### Esempio

Se scegli Europa (Spagna eu-south-2) CloudShell ma non 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. Da Console di gestione AWS, puoi avviarlo CloudShell scegliendo una delle seguenti opzioni:
  1. Nella barra di navigazione seleziona l'icona CloudShell.
  2. Nella casella di ricerca, digita «CloudShell», quindi scegli CloudShell.
  3. Nel widget Visitato di recente, scegli CloudShell.
  4. Scegli CloudShell su Console Toolbar, in basso a sinistra della console.
    - Puoi regolare l'altezza della CloudShell sessione trascinandola=.
    - Puoi passare alla CloudShell sessione a schermo intero facendo clic su 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 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

### Note

Questa opzione non è disponibile per gli ambienti VPC.

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 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.

## Passaggio 4: carica un file su AWS CloudShell

### Note

Questa opzione non è disponibile per gli ambienti VPC.

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 (Sfogliala).
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 monitora il 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 corrente `mysub_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

Questo passaggio descrive come creare un backup della home directory.

1. Creare un file di backup

Crea 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

Questo passaggio descrive come 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 di shell](#).

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

Ti viene comunicato 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

Questo passaggio descrive come eliminare una sessione di shell.

#### Note

Questa opzione non è disponibile per gli ambienti VPC. Quando riavvii un ambiente VPC, la sua home directory viene eliminata.

#### 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.

1. Per eliminare una sessione di shell, scegli Azioni, Elimina.

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.

AWS CloudShell interrompe tutte le sessioni attive nella sessione corrente Regione AWS. Puoi creare un nuovo ambiente o configurare un ambiente CloudShell VPC.

3. Per creare un nuovo ambiente, scegli Apri una scheda.
4. Per creare un ambiente CloudShell VPC, scegli Crea un ambiente VPC.

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 eseguilo 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 sugli Vim altri strumenti disponibili in AWS 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

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

- [Gestisci AWS i servizi dalla CLI in CloudShell](#)
- [Copiare più file tra il computer locale e CloudShell](#)
- [AWS CloudShell Concetti](#)
- [Personalizzare il AWS CloudShell esperienza](#)

# AWS CloudShell tutorial

I seguenti tutorial mostrano come sperimentare e testare diverse funzionalità e integrazioni durante l'utilizzo. AWS CloudShell

Panoramica del tutorial	Ulteriori informazioni
Copiare più file	<a href="#">the section called “Tutorial: Copiare più file”</a>
Creazione di predefiniti URLs	<a href="#">???</a>
Creazione di un contenitore Docker all'interno di AWS CloudShell e trasferimento ad Amazon ECR	<a href="#">???</a>
Implementazione di una funzione Lambda utilizzando AWS CDK	<a href="#">???</a>

## Copiare più file tra il computer locale e CloudShell

Questo tutorial mostra come copiare più file tra il computer locale e. CloudShell

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

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

### Note

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

## Caricamento e download di più file con Amazon S3

Questo passaggio descrive come caricare e scaricare più file utilizzando Amazon S3.

### Prerequisiti

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

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

Per un elenco completo delle azioni di Amazon S3, consulta [Azioni](#) in Riferimento API di Amazon Simple Storage Service.

### Caricare più file su AWS CloudShell Amazon S3

Questo passaggio descrive come caricare più file utilizzando Amazon S3.

1. In AWS CloudShell, crea un bucket S3 eseguendo il seguente comando: `s3`

```
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. Scegliete una delle seguenti opzioni per caricare i file:
  - **Console di gestione AWS:** consente drag-and-drop di 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

- Apri la console Amazon S3 all'indirizzo. <https://s3.console.aws.amazon.com/s3/>

(Se la utilizzi AWS CloudShell, dovresti aver già effettuato l'accesso alla console.)

- Nel riquadro di navigazione a sinistra, scegli Bucket, quindi scegli il nome del bucket in cui vuoi caricare le cartelle o i file. Puoi anche creare un bucket a tua scelta scegliendo Crea bucket.
- Per selezionare i file e le cartelle da caricare, scegli Carica. Quindi, trascina i file e le cartelle selezionati nella finestra della console che elenca 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 sull'intera gamma di opzioni di configurazione quando si utilizza la console, vedi [Come si caricano file e cartelle in un bucket S3?](#) nella Guida per l'utente di Amazon Simple Storage Service.

## Using AWS CLI

### Note

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

- Avvia lo AWS CLI strumento ed esegui il `aws s3` comando seguente 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 CloudShell riga di 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

Puoi anche 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, vedete [Uso dei filtri di esclusione e inclusione](#) nel riferimento ai AWS 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 alla destinazione.

## Scarica più file AWS CloudShell utilizzando Amazon S3

Questo passaggio descrive come scaricare più file utilizzando Amazon S3.

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

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

**Note**

Puoi anche 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, vedete [Uso dei filtri di esclusione e inclusione](#) nel riferimento ai AWS 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 lo AWS CLI strumento installato sul computer locale.

Dalla riga di comando dello AWS CLI strumento, esegui il seguente comando:

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

## Caricamento e download di più file utilizzando cartelle zippate

Questo passaggio descrive come caricare e scaricare più file utilizzando cartelle compresse.

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

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

## Caricare più file AWS CloudShell utilizzando cartelle zippate

Questo passaggio descrive come caricare più file utilizzando cartelle compresse.

1. Sul computer locale, aggiungi i file da caricare in una cartella compressa.
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 comando seguente per decomprimere il contenuto dell'archivio zip in una directory specificata:

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

## Scarica più file AWS CloudShell utilizzando cartelle zippate

Questo passaggio descrive come scaricare più file utilizzando cartelle compresse.

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 compressa scaricata.

# Creazione di un URL predefinito per oggetti Amazon S3 utilizzando CloudShell

Questo tutorial mostra come creare un URL predefinito 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 policy. `AWSCloudShellFullAccess`
- Per le autorizzazioni IAM necessarie per creare un URL predefinito, consulta [Share an object with others](#) nella Amazon Simple Storage Service User Guide.

## Fase 1: creare un ruolo IAM per concedere l'accesso al bucket Amazon S3

Questo passaggio descrive come creare un ruolo IAM per concedere l'accesso al bucket Amazon S3.

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

```
aws sts get-caller-identity
```

Se la chiamata ha esito positivo, la riga di comando visualizza una risposta simile alla seguente.

```
{
  "Account": "123456789012",
  "UserId": "AROAXX0ZUU0TTWDCVIDZ2:redirect_session",
  "Arn": "arn:aws:sts::531421766567:assumed-role/Feder08/redirect_session"
}
```

2. Prendi le informazioni sull'utente ottenute nel passaggio precedente e aggiungile a un CloudFormation modello. Questo modello crea un ruolo IAM. Questo ruolo concede al tuo collaboratore le autorizzazioni con il minimo privilegio per le risorse condivise.

```
Resources:
  CollaboratorRole:
    Type: AWS::IAM::Role
```

```
Properties:
  AssumeRolePolicyDocument:
    Version: 2012-10-17
    Statement:
      - Effect: Allow
        Principal:
          AWS: "arn:aws:iam::531421766567:role/Feder08"
        Action: "sts:AssumeRole"
    Description: Role used by my collaborators
    MaxSessionDuration: 7200
  CollaboratorPolicy:
    Type: AWS::IAM::Policy
    Properties:
      PolicyDocument:
        Version: 2012-10-17
        Statement:
          - Effect: Allow
            Action:
              - 's3:*'
            Resource: 'arn:aws:s3:::<YOUR_BUCKET_FOR_FILE_TRANSFER>'
            Condition:
              StringEquals:
                s3:prefix:
                  - "myfolder/*"
      PolicyName: S3ReadSpecificFolder
    Roles:
      - !Ref CollaboratorRole
Outputs:
  CollaboratorRoleArn:
    Description: Arn for the Collaborator's Role
    Value: !GetAtt CollaboratorRole.Arn
```

3. Salva il CloudFormation modello in un file denominato `template.yaml`
4. Usa il modello per distribuire lo stack e creare il ruolo IAM chiamando il `deploy` comando.

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

## Genera l'URL predefinito

Questo passaggio descrive come generare l'URL predefinito.

1. Utilizzando il tuo editor AWS CloudShell, aggiungi il codice seguente. Questo codice crea un URL che fornisce agli utenti federati l'accesso diretto a. Console di gestione AWS

```
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. Esegui quanto segue dalla riga di comando per recuperare l'Amazon Resource Name (ARN) del ruolo IAM da CloudFormation. Quindi, usalo nello Python script per ottenere credenziali di sicurezza temporanee.

```
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 su cui un collaboratore può fare clic per inserirlo AWS CloudShell . Console di gestione AWS. Il collaboratore ha il pieno controllo della `myfolder/` cartella nel bucket Amazon S3 per i prossimi 3.600 secondi (1 ora). Le credenziali scadono dopo un'ora. Dopo questo periodo, il collaboratore non può più accedere al bucket.

## Creazione di un container Docker all'interno 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 directory home.

```
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 --assumeeyes cowsay

# Default entrypoint binary
ENTRYPOINT [ "cowsay" ]

# Default argument for the cowsay entrypoint
CMD [ "Hello, World!" ]
```

5. 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 (.).

Immagine del comando `docker build` eseguito all'interno. AWS CloudShell

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

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

Immagine del comando `run` del contenitore `docker` all'interno AWS CloudShell

7. 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}
```

Immagine del comando utilizzato per creare un repository Amazon ECR all'interno AWS CloudShell

8. 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}
```

Immagine del comando usato per ottenere le credenziali di accesso Amazon ECR per Docker.

#### Note

Se la variabile di `AWS_REGION` ambiente non è impostata nel tuo CloudShell o desideri interagire con le risorse in altre Regioni AWS, esegui il seguente comando:

```
AWS_REGION=<your-desired-region>
```

9. 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}
```

Immagine del comando usato per etichettare l'immagine con il repository Amazon ECR di destinazione.

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}
```

## Implementazione di una funzione Lambda utilizzando in AWS CDK CloudShell

Questo tutorial mostra come implementare una funzione Lambda sul tuo account utilizzando AWS Cloud Development Kit (AWS CDK) in CloudShell

### Prerequisiti

- Avvia il tuo account per utilizzarlo con AWS CDK. Per informazioni sul bootstrap con AWS CDK, consulta [Bootstrapping](#) nella Guida per sviluppatori v2. AWS CDK. Se non hai ancora avviato l'account, puoi iniziare con `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 in AWS CDK CloudShell

1. Crea una nuova cartella nella directory home.

```
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
```

Immagine del comando usato per installare le AWS CDK dipendenze.

- 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
```

Immagine del comando per utilizzare la AWS CDK CLI per sintetizzare il progetto e distribuire le risorse.

10. Invoca la funzione Lambda per confermarla e verificarla.

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

Immagine del comando usato per richiamare la funzione Lambda.

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
```

# AWS CloudShell Concetti

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

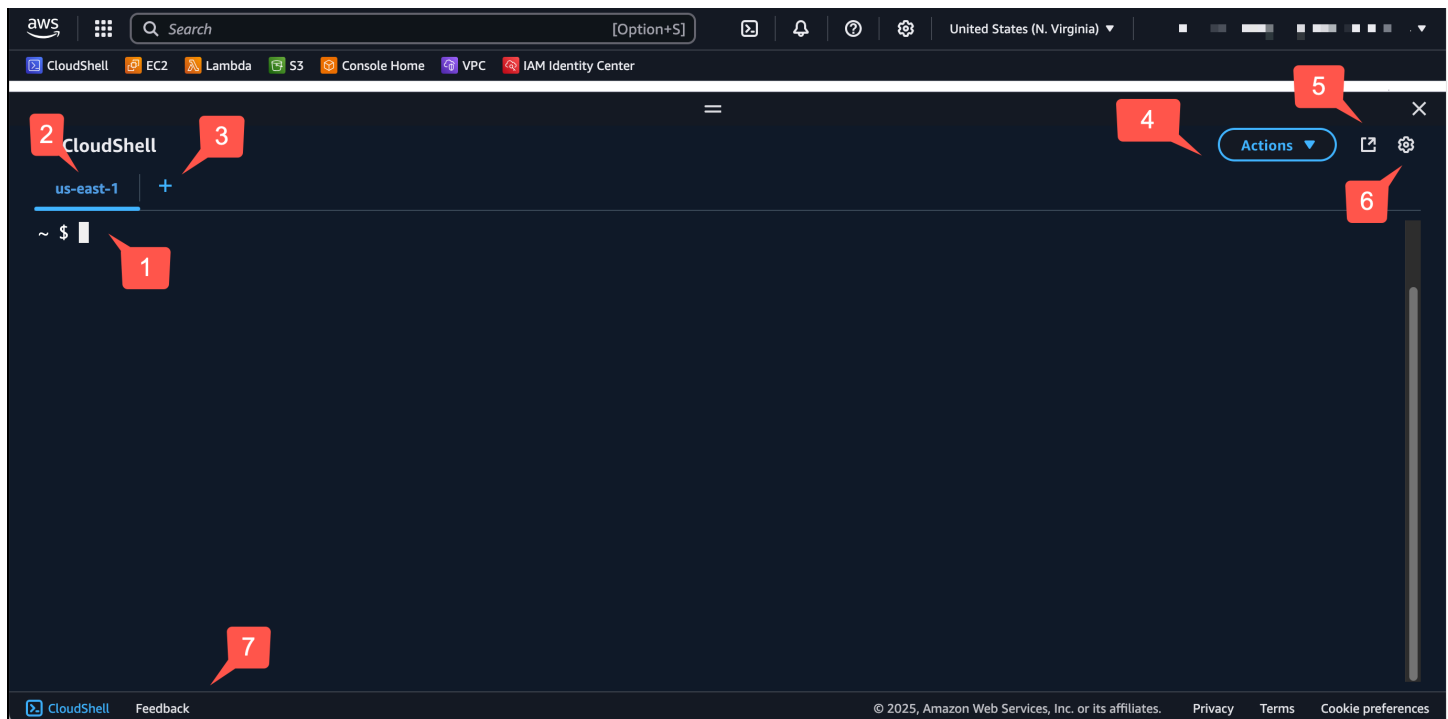
## Argomenti

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

## Navigazione nell'interfaccia AWS CloudShell

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

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



1. AWS CloudShell interfaccia 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. L'icona + è un menu a discesa che include opzioni per creare, riavviare ed eliminare ambienti.
4. Il menu Azioni, che offre 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

5. La scheda Apri in un nuovo browser, che offre la possibilità di accedere alla CloudShell sessione a schermo intero.
6. L'opzione Preferenze, che puoi usare per [personalizzare la tua esperienza con la shell](#).
7. La barra inferiore, che fornisce le seguenti opzioni per:
  - Avvia CloudShell dall'CloudShell icona.
  - 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 CloudShellsuConsole Toolbar, in basso a sinistra della console, 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 correndo viene visualizzata come una scheda.

Puoi scegliere un ambiente in cui Regione AWS lavorare selezionando una regione specifica usando 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](#).

- Lo storage persistente non è disponibile per gli AWS CloudShell ambienti VPC.

## 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 Console di gestione AWS 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 le 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 utente.

## 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 che si trovano all'esterno della 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.

## Accesso CloudShell nell'applicazione Console Mobile

È possibile accedere CloudShell AWS Console Mobile Application dalla schermata principale. Dalla schermata principale, è possibile visualizzare informazioni su CloudShell e altri AWS servizi. Per ulteriori informazioni, consulta [Nozioni di base di AWS Console Mobile Application](#). Per avviarlo CloudShell in AWS Console Mobile Application, scegli una delle seguenti opzioni:

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

Puoi uscire CloudShell in qualsiasi momento selezionando X.

Per ulteriori informazioni sull'accesso CloudShell nell'applicazione Console Mobile, vedere [Access AWS CloudShell](#).

### Note

Al momento, non è possibile creare o avviare ambienti VPC in. AWS Console Mobile Application

## 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: Implementazione 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 disponibile in tutte le AWS regioni, ad eccezione delle regioni AWS GovCloud (Stati Uniti). Per un elenco delle regioni in cui è disponibile Docker, consulta [AWS Regioni supportate](#) per. AWS CloudShell
- 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.

# 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. Puoi anche personalizzare l'aspetto di CloudShell, incluse 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 del terminale

È possibile utilizzare il Tab tasto nelle seguenti modalità:

- Modalità terminale (impostazione predefinita): in questa modalità, il terminale acquisisce la Tab chiave immessa. Dopo aver concentrato l'attenzione sul terminale, premi Tab per accedere solo alle funzionalità del terminale.
- Modalità di navigazione: in questa modalità, il terminale non registra la Tab chiave immessa. Premi Tab per navigare tra gli elementi focalizzabili della pagina.

Per passare dalla modalità terminale alla modalità navigazione, premi Ctrl +M. Dopo essere tornati indietro, nell'intestazione viene visualizzato il tasto di navigazione Tab: ed è possibile utilizzare il Tab tasto per navigare all'interno della pagina.

Per tornare alla modalità terminale, premi Ctrl +M. In alternativa, scegli X accanto a Tab: navigazione.

### Note

Attualmente, le funzionalità di accessibilità dei CloudShell 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 adattarlo alle proprie preferenze visive.

- Dimensione del carattere: scegli tra le dimensioni dei caratteri più piccole, piccole, medie, grandi e più grandi nel terminale. Per ulteriori informazioni sulla modifica della dimensione del carattere, consulta [the section called “Modifica della dimensione del carattere”](#).
- Tema: scegli tra i temi dell'interfaccia Chiaro e Scuro. Per ulteriori informazioni sulla modifica del tema dell'interfaccia, consulta [the section called “Modifica del tema dell'interfaccia”](#).

# Gestisci AWS i servizi dalla CLI in CloudShell

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

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

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

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

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 [AWS CLI Command Reference](#).

- [DynamoDB](#)
- [Amazon EC2](#)
- [Amazon Glacier](#)

## 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 di questo servizio supporta strutture di dati chiave-valore e documento.

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

```
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, consulta [Using DynamoDB with AWS CLI](#) nella AWS Command Line Interface Guida per l'utente.

## Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) Elastic Compute Cloud (Amazon EC2) è un servizio Web che fornisce capacità di calcolo sicura e ridimensionabile nel cloud. È progettato per rendere il cloud computing su scala web più semplice e accessibile.

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 [Using Amazon EC2 with the AWS CLI](#) nella AWS Command Line Interface User Guide.

## Amazon Glacier

Amazon Glacier e Amazon Glacier Deep Archive sono classi di storage cloud Amazon S3 sicure, durevoli ed estremamente economiche per l'archiviazione dei dati e il backup a lungo termine.

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

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

Per ulteriori informazioni, consulta [Using Amazon Glacier with AWS CLI](#) nella AWS Command Line Interface Guida per l'utente.

## AWS CLI Elastic Beanstalk

La AWS Elastic Beanstalk CLI fornisce un'interfaccia a riga di comando creata 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 ai comandi EB CLI](#) nella AWS Elastic Beanstalk Developer Guide.

## CLI di Amazon ECS

L'interfaccia a riga di comando (CLI) di Amazon Elastic Container Service (Amazon ECS) fornisce diversi comandi di alto livello. Sono progettati per semplificare i processi di creazione, aggiornamento e monitoraggio di cluster e attività da un ambiente di sviluppo locale. (Un cluster Amazon ECS è un raggruppamento logico di attività o servizi.)

Il `configure` comando seguente configura la CLI di Amazon ECS per creare una configurazione 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.

# AWS SAM CLI

AWS SAM CLI 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 utilizzandolo. 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

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

```
sam init --runtime python3.9 --dependency-manager pip --app-template hello-world --name sam-app
```

Per ulteriori informazioni, consulta il [riferimento ai comandi AWS SAM CLI](#) nella AWS Serverless Application Model Developer Guide.

## Utilizzo della CLI di Kiro in CloudShell

La CLI di Kiro è un'interfaccia a riga di comando che consente di interagire con Kiro. Per ulteriori informazioni, consulta le [funzionalità principali di Kiro CLI](#) nella Guida per l'utente di Kiro.

Kiro CLI CloudShell in ti consente di interagire in conversazioni in linguaggio naturale, porre domande e ricevere risposte da Kiro, il tutto dal tuo terminale. È possibile ottenere il relativo comando shell che riduce la necessità di cercare, ricordare la sintassi e ricevere suggerimenti sui comandi durante la digitazione nel terminale.

Se non vedi le funzionalità CloudShell della CLI di Kiro, contatta il tuo amministratore per fornirti le autorizzazioni IAM. Per ulteriori informazioni, consulta [Esempi di policy basate sull'identità per Kiro Developer nella Guida per l'utente di Kiro](#).

Questo capitolo spiega come utilizzare le funzionalità della CLI di Kiro in CloudShell

Alcune funzionalità della CLI di Kiro richiedono l'autenticazione. Per ulteriori informazioni, consulta [Authentication](#) nella Kiro User Guide.

## Usare il comando chat di Kiro in CloudShell

Il `kiro-cli` comando ti consente di porre domande e ricevere risposte da Kiro, il tutto dal tuo terminale. Per iniziare una conversazione con Kiro, esegui il `kiro-cli` comando nel terminale. CloudShell Per ulteriori informazioni, consulta [Chattare con Kiro nella CLI nella Guida](#) per l'utente di Kiro.

## Usare il comando Kiro translate in CloudShell

Il `kiro-cli translate` comando consente di scrivere istruzioni in linguaggio naturale. Per tradurre con Kiro, esegui `kiro-cli translate` il comando nel CloudShell terminale. Per maggiori informazioni, consulta [Tradurre dal linguaggio naturale a bash nella Guida per](#) l'utente di Kiro.

## Completamento del comando CLI in CloudShell

Il completamento della CLI CloudShell fornisce suggerimenti per comandi e opzioni durante la digitazione nel terminale. Per ulteriori informazioni, consulta [Generazione del completamento della riga di comando nella Guida](#) per l'utente di Kiro.

## Utilizzo dei suggerimenti in linea di Kiro in CloudShell

I suggerimenti in linea di Kiro CloudShell forniscono suggerimenti sui comandi durante la digitazione nel terminale. Per ulteriori informazioni, consulta [Kiro inline sulla riga di comando nella Guida per l'utente di Kiro](#).

Per usare i suggerimenti in linea di Kiro in CloudShell

1. Da Console di gestione AWS, Scegli. CloudShell
2. Sul CloudShell terminale, passa alla shell Z e inizia a digitare. Per passare alla shell Z, digita `zsh` nel terminale, quindi premi Invio.

### Note

Attualmente, Kiro inline è supportato solo nella shell Z.

Quando inizi a digitare il comando, Kiro darà suggerimenti basati sull'input corrente e sui comandi precedenti. I suggerimenti in linea vengono abilitati automaticamente.

Per disabilitare i suggerimenti in linea, esegui il seguente comando:

```
kiro-cli inline disable
```

Per abilitare i suggerimenti in linea, esegui il seguente comando:

```
kiro-cli inline enable
```

## Policy basata sull'identità per Kiro CLI in CloudShell

Per utilizzare la CLI di Kiro, assicurati CloudShell di disporre delle autorizzazioni IAM richieste. Per ulteriori informazioni, consulta [Esempi di policy basate sull'identità per Kiro Developer nella Kiro User Guide](#).

# Esecuzione di un comando CloudShell dalle console AWS di servizio

Puoi eseguire un comando nel CloudShell terminale tramite le console [Amazon ElastiCache e Amazon DocumentDB \(con compatibilità con MongoDB\)](#) in. Console di gestione AWS

Per eseguire un comando CloudShell da altre console di AWS servizio, la policy IAM assegnata al tuo ruolo deve includere le autorizzazioni. `cloudshell:approveCommand`

CloudShell si apre sulla barra degli strumenti della console e viene visualizzato il popup del comando Esegui in. CloudShell Nel pop-up del comando Esegui, il comando viene visualizzato nella casella di comando.

Per eseguire un comando nel CloudShell terminale, scegli uno dei seguenti passaggi:

1. Inserisci un nome nella casella Nuovo nome ambiente se non hai creato un ambiente VPC in. CloudShell

Puoi visualizzare i dettagli dell'ambiente VPC basati sui dettagli VPC della tua risorsa.

- a. Scegli Create and run (Crea ed esegui).

Questo passaggio creerà un nuovo ambiente CloudShell VPC ed eseguirà il comando nel CloudShell terminale.


2. È possibile visualizzare il nome CloudShell dell'ambiente se è già stato creato un ambiente CloudShell VPC.

## Note

Se disponi già di un ambiente CloudShell VPC, non puoi creare un nuovo ambiente VPC.

- a. Scegli Esegui.

Questo passaggio eseguirà il comando nel CloudShell terminale nell'ambiente CloudShell VPC selezionato.

 Note

Se non disponi dell'autorizzazione per visualizzare gli ambienti VPC creati, contatta l'amministratore per aggiungere l'`cloudshell:describeEnvironments` autorizzazione. Per ulteriori informazioni, consulta [Gestione dell' AWS CloudShell accesso e dell'utilizzo con le politiche IAM](#).

Puoi continuare a eseguire comandi nel CloudShell terminale.

# Personalizzare il AWS CloudShell esperienza

Puoi personalizzare i seguenti aspetti della tua AWS CloudShell esperienza:

- [Layout a schede](#): suddivide 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 funzione che richiede la verifica del testo su più righe prima di incollarlo.
- [Tmux per ripristinare la sessione: L'uso di tmux ripristina](#) la sessione fino a quando la sessione non diventa inattiva.

[È inoltre possibile estendere l'ambiente shell installando il proprio software e modificando la shell con degli script.](#)

## 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 x, che si trova accanto al nome della regione.

- Scegli Azioni e una delle seguenti opzioni dal layout a 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.
  - Dividi 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 le impostazioni del AWS CloudShell terminale, vai su Impostazioni, Preferenze.
2. Scegli una dimensione del testo. Le opzioni disponibili sono la più piccola, la più piccola, la media, la più grande e la più grande.

## Modifica del tema dell'interfaccia

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

1. Per cambiare il AWS CloudShell tema, vai su Impostazioni, Preferenze.
2. Scegli Chiaro o Scuro.

## Utilizzo di Safe Paste per il 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, scegli Incolla.

**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 catturare potenziali rischi per la sicurezza negli script. Puoi attivare o disattivare questa funzionalità scegliendo Preferenze, Abilita Safe Paste e Disabilita Safe Paste.

## Utilizzo 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, consulta Ripristino della sessione.](#)

# Utilizzo AWS CloudShell in Amazon VPC

AWS CloudShell il cloud privato virtuale (VPC) ti consente di creare un CloudShell ambiente nel tuo VPC. Per ogni ambiente VPC, puoi assegnare un VPC, aggiungere una sottorete e associare fino a cinque gruppi di sicurezza. AWS CloudShell eredita la configurazione di rete del VPC e consente di AWS CloudShell utilizzare in modo sicuro all'interno della stessa sottorete delle altre risorse del VPC e di connettersi ad esse.

Con Amazon VPC, puoi avviare AWS risorse in una rete virtuale logicamente isolata che hai definito. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS. Per ulteriori informazioni sul VPC, consulta [Amazon Virtual Private Cloud](#).

## Vincoli operativi

AWS CloudShell Gli ambienti VPC presentano i seguenti vincoli:

- Puoi creare un massimo di due ambienti VPC per principale IAM.
- È possibile assegnare un massimo di cinque gruppi di sicurezza per un ambiente VPC.
- Non è possibile utilizzare le opzioni di CloudShell caricamento e download nel menu Azioni per ambienti VPC.

### Note

È possibile caricare o scaricare file da ambienti VPC che hanno accesso a Internet ingress/egress tramite altri strumenti CLI.

- Gli ambienti VPC non supportano lo storage persistente. Lo storage è effimero. I dati e la home directory vengono eliminati al termine di una sessione di ambiente attiva.
- Il tuo AWS CloudShell ambiente può connettersi a Internet solo se si trova in una sottorete VPC privata.

### Note

Per impostazione predefinita, gli indirizzi IP pubblici non vengono assegnati agli ambienti CloudShell VPC. Gli ambienti VPC creati in sottoreti pubbliche con tabelle di routing configurate per instradare tutto il traffico verso Internet Gateway non avranno accesso alla

rete Internet pubblica, ma le sottoreti private configurate con Network Address Translation (NAT) avranno accesso alla rete Internet pubblica. Gli ambienti VPC creati in tali sottoreti private avranno accesso alla rete Internet pubblica.

- Per impostazione predefinita, il tuo AWS CloudShell ambiente non sarà configurato per utilizzare la zona DNS locale nel tuo VPC.
- Per fornire un CloudShell ambiente gestito per il tuo account, AWS potresti fornire l'accesso di rete ai seguenti servizi per l'host di elaborazione sottostante:
  - Simple Storage Service (Amazon S3)
  - Endpoint VPC
    - com.amazonaws. <region>messaggi.ssm
    - com.amazonaws. <region>.registri
    - com.amazonaws.it. <region>.kms
    - com.amazonaws.it. <region>.execute-api
    - com.amazonaws. <region>.ecs-telemetry
    - com.amazonaws. <region>.agente ecs
    - com.amazonaws. <region>.ecs
    - com.amazonaws.it. <region>.ecr.dkr
    - com.amazonaws. <region>.ecr.api
    - com.amazonaws. <region>.codecatalyst.pacchetti
    - com.amazonaws. <region>.codecatalyst.git
    - aws.api.global.codecatalyst

Non puoi limitare l'accesso a questi endpoint modificando la configurazione del VPC.

CloudShell Il VPC è disponibile in tutte le AWS regioni e GovCloud regioni. Per un elenco delle regioni in cui è disponibile il CloudShell VPC, consulta [AWS Regioni supportate](#) per. AWS CloudShell

## Creazione di un CloudShell ambiente VPC

Questo argomento illustra i passaggi per creare un ambiente VPC in. CloudShell

L'amministratore deve fornire le autorizzazioni IAM necessarie per consentirti di creare ambienti VPC. Per ulteriori informazioni sull'abilitazione delle autorizzazioni per creare ambienti CloudShell VPC, consulta [the section called “Autorizzazioni IAM richieste per la creazione e l'utilizzo di ambienti CloudShell VPC”](#)

Per creare un ambiente CloudShell VPC

1. Nella pagina della CloudShell console, scegli l'icona +, quindi scegli Crea ambiente VPC dal menu a discesa.
2. Nella pagina Crea un ambiente VPC, inserisci un nome per il tuo ambiente VPC nella casella Nome.
3. Dall'elenco a discesa Virtual private cloud (VPC), scegli un VPC.
4. Dall'elenco a discesa Subnet, scegli una sottorete.
5. Dall'elenco a discesa Gruppo di sicurezza, scegli uno o più gruppi di sicurezza che desideri assegnare al tuo ambiente VPC.

#### Note

Puoi scegliere un massimo di cinque gruppi di sicurezza.

6. Scegli Crea per creare il tuo ambiente VPC.
7. (Facoltativo) Scegliete Azioni, quindi scegliete Visualizza dettagli per esaminare i dettagli dell'ambiente VPC appena creato. L'indirizzo IP dell'ambiente VPC viene visualizzato nel prompt della riga di comando.

Per informazioni sull'utilizzo degli ambienti VPC, vedere [Nozioni di base](#)

## Autorizzazioni IAM richieste per la creazione e l'utilizzo di ambienti CloudShell VPC

Per creare e utilizzare ambienti CloudShell VPC, l'amministratore IAM deve abilitare l'accesso alle autorizzazioni Amazon EC2 specifiche per VPC. Questa sezione elenca le autorizzazioni Amazon EC2 necessarie per creare e utilizzare ambienti VPC.

Per creare ambienti VPC, la policy IAM assegnata al tuo ruolo deve includere le seguenti autorizzazioni Amazon EC2:

- `ec2:DescribeVpcs`
- `ec2:DescribeSubnets`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeDhcpOptions`
- `ec2:DescribeNetworkInterfaces`
  
- `ec2:CreateTags`
- `ec2:CreateNetworkInterface`
- `ec2:CreateNetworkInterfacePermission`

Ti consigliamo di includere:

- `ec2>DeleteNetworkInterface`

#### Note

Questa autorizzazione non è obbligatoria, ma è necessaria per CloudShell ripulire la risorsa ENI (gli ENI creati per gli ambienti CloudShell VPC sono contrassegnati `ManagedByCloudShell` con la chiave) da essa creata. Se questa autorizzazione non è abilitata, è necessario pulire manualmente la risorsa ENI dopo ogni utilizzo dell'ambiente CloudShell VPC.

## Policy IAM che garantisce CloudShell l'accesso completo, incluso l'accesso al VPC

L'esempio seguente mostra come abilitare le autorizzazioni complete, incluso l'accesso al VPC, per: CloudShell

JSON

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

```

    "Sid": "AllowCloudShellOperations",
    "Effect": "Allow",
    "Action": [
      "cloudshell:*"
    ],
    "Resource": "*"
  },
  {
    "Sid": "AllowDescribeVPC",
    "Effect": "Allow",
    "Action": [
      "ec2:DescribeSubnets",
      "ec2:DescribeSecurityGroups",
      "ec2:DescribeVpcs"
    ],
    "Resource": "*"
  },
  {
    "Sid": "AllowInspectVPCConfigurationViaCloudShell",
    "Effect": "Allow",
    "Action": [
      "ec2:DescribeDhcpOptions",
      "ec2:DescribeNetworkInterfaces"
    ],
    "Resource": "*",
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:CalledVia": "cloudshell.amazonaws.com"
      }
    }
  },
  {
    "Sid": "AllowCreateTagWithCloudShellKeyViaCloudShell",
    "Effect": "Allow",
    "Action": [
      "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*:*:network-interface/*",
    "Condition": {
      "StringEquals": {
        "ec2:CreateAction": "CreateNetworkInterface"
      },
      "ForAnyValue:StringEquals": {
        "aws:TagKeys": "ManagedByCloudShell",

```



```
    },
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "cloudshell.amazonaws.com"
    }
  }
},
{
  "Sid": "AllowDeleteNetworkInterfaceWithCloudShellTagViaCloudShell",
  "Effect": "Allow",
  "Action": [
    "ec2:DeleteNetworkInterface"
  ],
  "Resource": "arn:aws:ec2:*:*:network-interface/*",
  "Condition": {
    "StringEquals": {
      "aws:ResourceTag/ManagedByCloudShell": ""
    },
    "ForAnyValue:StringEquals": {
      "aws:CalledVia": "cloudshell.amazonaws.com"
    }
  }
}
]
}
```

## Utilizzo delle chiavi di condizione IAM per ambienti VPC

Puoi utilizzare le chiavi di CloudShell-specific condizione per le impostazioni VPC per fornire controlli di autorizzazione aggiuntivi per i tuoi ambienti VPC. Puoi anche specificare le sottoreti e i gruppi di sicurezza che l'ambiente VPC può e non può utilizzare.

CloudShell supporta le seguenti chiavi di condizione nelle politiche IAM:

- `CloudShell:VpcIds`— Consentire o negare uno o più VPC
- `CloudShell:SubnetIds`— Consentire o negare una o più sottoreti
- `CloudShell:SecurityGroupIds`— Consentire o negare uno o più gruppi di sicurezza

**Note**

Se le autorizzazioni per gli utenti con accesso agli CloudShell ambienti pubblici vengono modificate per aggiungere restrizioni all'`cloudshell:createEnvironment`, possono comunque accedere all'ambiente pubblico esistente. Tuttavia, se desideri modificare una policy IAM con questa restrizione e disabilitare il loro accesso all'ambiente pubblico esistente, devi prima aggiornare la policy IAM con la restrizione, quindi assicurarti che ogni CloudShell utente del tuo account elimini manualmente l'ambiente pubblico esistente utilizzando l'interfaccia utente CloudShell web (Azioni → Elimina ambiente). CloudShell

## Policy di esempio con chiavi di condizione per le impostazioni VPC

Negli esempi seguenti viene illustrato come utilizzare le chiavi di condizione per le impostazioni VPC. Dopo aver creato un'istruzione delle policy con le restrizioni desiderate, aggiungere l'istruzione delle policy per l'utente o il ruolo di destinazione.

Assicurati che gli utenti creino solo ambienti VPC e neghi la creazione di ambienti pubblici

Per garantire che gli utenti possano creare solo ambienti VPC, utilizza l'autorizzazione di negazione come mostrato nell'esempio seguente:

```
{
  "Statement": [
    {
      "Sid": "DenyCloudShellNonVpcEnvironments",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
      "Condition": {
        "Null": {
          "cloudshell:VpcIds": "true"
        }
      }
    }
  ]
}
```

## Negare agli utenti l'accesso a specifici VPC, sottoreti o gruppi di sicurezza

Per negare agli utenti l'accesso a specifici VPC, utilizzare `StringEquals` per verificare il valore della condizione `cloudshell:VpcIds`. L'esempio seguente nega agli utenti l'accesso a `vpc-1` e `vpc-2`

Per negare agli utenti l'accesso a specifici VPC, utilizzare `StringEquals` per verificare il valore della condizione `cloudshell:SubnetIds`. L'esempio seguente nega agli utenti l'accesso a `subnet-1` e `subnet-2`

Per negare agli utenti l'accesso a specifici VPC, utilizzare `StringEquals` per verificare il valore della condizione `cloudshell:SecurityGroupIds`. L'esempio seguente nega agli utenti l'accesso a `sg-1` e `sg-2`

### JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceOutOfSecurityGroups",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "cloudshell:SecurityGroupIds": [
            "sg-1",
            "sg-2"
          ]
        }
      }
    }
  ]
}
```

## Consenti agli utenti di creare ambienti con configurazioni VPC specifiche

Per consentire agli utenti di accedere a VPC specifici, utilizza `StringEquals` per verificare il valore della condizione. `cloudshell:VpcIds` L'esempio seguente consente agli utenti di accedere a `vpc-1` e `vpc-2`:

Per consentire agli utenti di accedere a VPC specifici, utilizza `StringEquals` per verificare il valore della `cloudshell:SubnetIds` condizione. L'esempio seguente consente agli utenti di accedere a `subnet-1` e `subnet-2`:

### JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceStayInSpecificSubnets",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "ForAllValues:StringEquals": {
          "cloudshell:SubnetIds": [
            "subnet-1",
            "subnet-2"
          ]
        }
      }
    }
  ]
}
```

Per consentire agli utenti di accedere a VPC specifici, utilizza `StringEquals` per verificare il valore della `cloudshell:SecurityGroupIds` condizione. L'esempio seguente consente agli utenti di accedere a `sg-1` e `sg-2`:

## JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceStayInSpecificSecurityGroup",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "ForAllValues:StringEquals": {
          "cloudshell:SecurityGroupIds": [
            "sg-1",
            "sg-2"
          ]
        }
      }
    }
  ]
}
```

# 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](#)
- [Best practice di sicurezza per AWS CloudShell](#)
- [AWS CloudShell Domande frequenti sulla sicurezza](#)

# Protezione dei dati in AWS CloudShell

Ulteriori informazioni su come il [modello di responsabilità condivisa](#) AWS 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. L'utente è inoltre responsabile della configurazione della protezione e delle attività di gestione per i Servizi AWS utilizzati. Per ulteriori informazioni sulla privacy dei dati, consulta [Domande frequenti sulla privacy dei dati](#). Per ulteriori informazioni sulla protezione dei dati in Europa, consulta il [Centro generale sulla protezione dei dati \(GDPR\)](#).

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

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- SSL/TLS Da utilizzare per comunicare con AWS le risorse. È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Configura l'API e la registrazione delle attività degli utenti con AWS CloudTrail. Per informazioni sull'utilizzo dei CloudTrail percorsi per acquisire AWS le attività, consulta [Lavorare con i CloudTrail percorsi](#) nella Guida per l'AWS CloudTrail utente.
- 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 140-3 per accedere AWS tramite un'interfaccia a riga di comando o un'API, usa un endpoint FIPS. Per ulteriori informazioni sugli endpoint FIPS disponibili, consulta il [Federal Information Processing Standard \(FIPS\) 140-3](#).

Ti consigliamo 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 la fatturazione o i log di diagnostica. Quando si fornisce un URL a un server esterno, suggeriamo vivamente di non includere informazioni sulle credenziali nell'URL per convalidare la 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 tra AWS CloudShell gli endpoint di servizio.

### 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 AWS KMS keys: 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 dei dati 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 AWS CloudShell vengono crittografate inviando tutto tramite 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 tramite policy](#)
- [In che modo AWS CloudShell funziona con IAM](#)
- [Identity-based esempi di policy per AWS CloudShell](#)
- [Risoluzione dei problemi di CloudShell identità e accesso AWS](#)
- [Gestione AWS CloudShell accesso e utilizzo con le policy IAM](#)

## Destinatari

Il modo in cui utilizzi AWS Identity and Access Management (IAM) varia in base al tuo ruolo:

- Utente del servizio: richiedi le autorizzazioni all'amministratore se non riesci ad accedere alle funzionalità (consulta [Risoluzione dei problemi di CloudShell identità e accesso AWS](#))
- Amministratore del servizio: determina l'accesso degli utenti e invia le richieste di autorizzazione (consulta [In che modo AWS CloudShell funziona con IAM](#))
- Amministratore IAM: scrivi policy per gestire l'accesso (consulta [Identity-based esempi di policy per AWS CloudShell](#))

## Autenticazione con identità

L'autenticazione è il modo in cui accedi AWS utilizzando le tue credenziali di identità. Devi autenticarti come utente IAM o assumendo un ruolo IAM. Utente root dell'account AWS

Puoi accedere come identità federata utilizzando credenziali provenienti da una fonte di identità come AWS IAM Identity Center (IAM Identity Center), autenticazione Single Sign-On o credenziali. Google/Facebook Per ulteriori informazioni sull'accesso, consulta [Come accedere all' Account AWS](#) nella Guida per l'utente di Accedi ad AWS .

Per l'accesso programmatico, AWS fornisce un SDK e una CLI per firmare crittograficamente le richieste. Per ulteriori informazioni, consulta [AWS Signature Version 4 per le richieste API](#) nella Guida per l'utente di IAM.

## Account AWS utente root

Quando si crea un Account AWS, si inizia con un'identità di accesso denominata utente Account AWS root che ha accesso completo a tutte Servizi AWS le risorse. Consigliamo vivamente di non utilizzare l'utente root per le attività quotidiane. Per le attività che richiedono le credenziali dell'utente root, consulta [Attività che richiedono le credenziali dell'utente root](#) nella Guida per l'utente IAM.

## Identità federata

Come procedura ottimale, richiedi agli utenti umani di utilizzare la federazione con un provider di identità per accedere Servizi AWS utilizzando credenziali temporanee.

Un'identità federata è un utente della directory aziendale, del provider di identità Web o Directory Service che accede Servizi AWS utilizzando le credenziali di una fonte di identità. Le identità federate assumono ruoli che forniscono credenziali temporanee.

Per la gestione centralizzata degli accessi, si consiglia di utilizzare AWS IAM Identity Center. Per ulteriori informazioni, consulta [Che cos'è il Centro identità IAM?](#) nella Guida per l'utente di AWS IAM Identity Center .

## Utenti e gruppi IAM

Un [utente IAM](#) è una identità che dispone di autorizzazioni specifiche per una singola persona o applicazione. Ti consigliamo di utilizzare credenziali temporanee invece di utenti IAM con credenziali a lungo termine. Per ulteriori informazioni, consulta [Richiedere agli utenti umani di utilizzare la federazione con un provider di identità per accedere AWS utilizzando credenziali temporanee](#) nella Guida per l'utente IAM.

Un [gruppo IAM](#) specifica una raccolta di utenti IAM e semplifica la gestione delle autorizzazioni per gestire gruppi di utenti di grandi dimensioni. Per ulteriori informazioni, consulta [Casi d'uso per utenti IAM](#) nella Guida per l'utente di IAM.

## Ruoli IAM

Un [ruolo IAM](#) è un'identità con autorizzazioni specifiche che fornisce credenziali temporanee. Puoi assumere un ruolo [passando da un ruolo utente a un ruolo IAM \(console\)](#) o chiamando un'operazione AWS CLI o AWS API. Per ulteriori informazioni, consulta [Metodi per assumere un ruolo](#) nella Guida per l'utente di IAM.

I ruoli IAM sono utili per l'accesso degli utenti federati, le autorizzazioni utente IAM temporanee, l'accesso multi-account, l'accesso multi-servizio e le applicazioni in esecuzione su Amazon EC2. Per

maggiori informazioni, consultare [Accesso a risorse multi-account in IAM](#) nella Guida per l'utente IAM.

## Gestione dell'accesso tramite policy

Puoi controllare l'accesso AWS creando policy e associandole a AWS identità o risorse. Una policy definisce le autorizzazioni quando è associata a un'identità o a una risorsa. AWS valuta queste politiche quando un preside effettua una richiesta. La maggior parte delle politiche viene archiviata AWS come documenti JSON. Per maggiori informazioni sui documenti delle policy JSON, consulta [Panoramica delle policy JSON](#) nella Guida per l'utente IAM.

Utilizzando le policy, gli amministratori specificano chi ha accesso a cosa definendo quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Un amministratore IAM crea le policy IAM e le aggiunge ai ruoli, che gli utenti possono quindi assumere. Le policy IAM definiscono le autorizzazioni indipendentemente dal metodo utilizzato per eseguirle.

### Identity-based politiche

Identity-based le politiche sono documenti di policy sulle autorizzazioni JSON che alleggi a un'identità (utente, gruppo o ruolo). Tali policy controllano le operazioni autorizzate per l'identità, nonché le risorse e le condizioni in cui possono essere eseguite. Per informazioni su come creare una policy basata su identità, consultare [Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente](#) nella Guida per l'utente IAM.

Identity-based le politiche possono essere politiche in linea (incorporate direttamente in una singola identità) o politiche gestite (politiche autonome collegate a più identità). Per informazioni su come scegliere tra una policy gestita o una policy inline, consulta [Scegliere tra policy gestite e policy in linea](#) nella Guida per l'utente di IAM.

### Resource-based politiche

Resource-based le politiche sono documenti di policy JSON allegati a una risorsa. Gli esempi includono le policy di trust dei ruoli IAM e le policy dei bucket di Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. In una policy basata sulle risorse è obbligatorio [specificare un'entità principale](#).

Resource-based le politiche sono politiche in linea che si trovano in quel servizio. Non è possibile utilizzare le policy AWS gestite di IAM in una policy basata sulle risorse.

## Altri tipi di policy

AWS supporta tipi di policy aggiuntivi che possono impostare le autorizzazioni massime concesse dai tipi di policy più comuni:

- **Limiti delle autorizzazioni:** imposta il numero massimo di autorizzazioni che una policy basata su identità ha la possibilità di concedere a un'entità IAM. Per ulteriori informazioni, consulta [Limiti delle autorizzazioni per le entità IAM](#) nella Guida per l'utente di IAM.
- **Policy di controllo dei servizi (SCP):** specifica il numero massimo di autorizzazioni per un'organizzazione o un'unità organizzativa (OU) in AWS Organizations. Per ulteriori informazioni, consultare [Policy di controllo dei servizi](#) nella Guida per l'utente di AWS Organizations .
- **Policy di controllo delle risorse (RCP):** imposta le autorizzazioni massime disponibili per le risorse degli account. Per ulteriori informazioni, consulta [Policy di controllo delle risorse \(RCP\)](#) nella Guida per l'utente di AWS Organizations .
- **Policy di sessione:** policy avanzate passate come parametro quando si crea una sessione temporanea per un ruolo o un utente federato. Per maggiori informazioni, consultare [Policy di sessione](#) nella Guida per l'utente IAM.

## Più tipi di policy

Quando a una richiesta si applicano più tipi di politiche, le autorizzazioni risultanti sono più complicate da comprendere. Per scoprire come si AWS determina se consentire o meno 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 possibile utilizzare CloudShell.

Funzionalità IAM che puoi usare con AWS CloudShell

Funzionalità IAM	CloudShell supporto
<a href="#">Identity-based politiche</a>	Si

Funzionalità IAM	CloudShell supporto
<a href="#">Resource-based politiche</a>	No
<a href="#">Operazioni 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="#">Service-linked ruoli</a>	No

Per avere una visione di alto livello di 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](#).

## Identity-based politiche per CloudShell

Supporta le policy basate sull'identità: sì

Identity-based le policy sono documenti di policy sulle autorizzazioni JSON che puoi allegare a un'identità, ad esempio un utente IAM, un gruppo di utenti o un ruolo. Tali policy definiscono le operazioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta [Definizione di autorizzazioni personalizzate IAM con policy gestite dal cliente](#) nella Guida per l'utente di IAM.

Con le policy basate sull'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. 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 IAM.

## Identity-based esempi di policy per CloudShell

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

## Resource-based politiche all'interno CloudShell

Supporta le policy basate su risorse: no

Resource-based le politiche sono documenti di policy JSON allegati a una risorsa. Esempi di policy basate sulle risorse sono le policy di attendibilità dei ruoli IAM e le policy di bucket Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le operazioni che un principale può eseguire su tale risorsa e a quali condizioni. In una policy basata sulle risorse è obbligatorio [specificare un'entità principale](#). I principali possono includere account, utenti, ruoli, utenti federati o. Servizi AWS

Per consentire l'accesso multi-account, è possibile specificare un intero account o entità IAM in un altro account come entità principale in una policy basata sulle risorse. Per ulteriori informazioni, consulta [Accesso a risorse multi-account in IAM](#) nella Guida per l'utente IAM.

## Azioni politiche per CloudShell

Supporta le operazioni di policy: si

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

L'elemento `Action` di una policy JSON descrive le operazioni che è possibile utilizzare per consentire o negare l'accesso in una policy. Includere le operazioni in una policy per concedere le autorizzazioni di 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. [Identity-based esempi di policy per AWS CloudShell](#)

## Risorse politiche per CloudShell

Supporta le risorse relative alle policy: sì

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale entità 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'operazione. Come best practice, specifica una risorsa utilizzando il suo [nome della risorsa Amazon \(ARN\)](#). Per le azioni che non supportano le autorizzazioni a livello di risorsa, si 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. [Identity-based esempi di policy 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. In altre parole, quale entità principale può eseguire operazioni su quali risorse e in quali condizioni.

L'elemento `Condition` specifica quando le istruzioni vengono eseguite in base a criteri definiti. È possibile 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. 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 [Identity-based esempi di policy 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 per 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

Attribute-based il controllo degli accessi (ABAC) è una strategia di autorizzazione che definisce le autorizzazioni in base ad attributi chiamati tag. È possibile allegare tag a entità e AWS risorse IAM, quindi progettare politiche ABAC per consentire le operazioni quando il tag del principale corrisponde al tag sulla risorsa.

Per controllare l'accesso basato su tag, fornire 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 Sì. Se un servizio supporta tutte e tre le chiavi di condizione solo per alcuni tipi di risorsa, allora il valore sarà Parziale.

Per maggiori informazioni su ABAC, consulta [Definizione delle autorizzazioni con autorizzazione 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

Le credenziali temporanee forniscono l'accesso a breve termine alle AWS risorse e vengono create automaticamente quando si utilizza la federazione o si cambia ruolo. AWS consiglia di generare dinamicamente credenziali temporanee anziché utilizzare chiavi di accesso a lungo termine. Per ulteriori informazioni, consulta [Credenziali di sicurezza temporanee in IAM](#) e [Servizi AWS compatibili con IAM](#) nella Guida per l'utente 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

Supporta l'inoltro delle sessioni di accesso (FAS): no

Le sessioni di accesso inoltrato (FAS) utilizzano le autorizzazioni del principale chiamante al Servizio AWS, combinate con la richiesta di effettuare richieste Servizio AWS ai servizi downstream. Per i dettagli delle policy relative alle richieste FAS, consulta [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 [Create a role to delegate permissions to an Servizio AWS](#) nella Guida per l'utente IAM.

### Warning

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

## Service-linked ruoli 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 di eseguire un'azione per conto dell'utente. Service-linked i ruoli appaiono nel tuo Account AWS account e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati al servizio, ma non modificarle.

## Identity-based esempi di policy per AWS CloudShell

Per impostazione predefinita, gli utenti e i ruoli non dispongono dell'autorizzazione per creare o modificare risorse CloudShell. Per concedere agli utenti l'autorizzazione a eseguire azioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM.

Per informazioni su come creare una policy basata su identità IAM utilizzando questi documenti di policy JSON di esempio, consulta [Creazione di policy IAM \(console\)](#) 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 CloudShell](#)
- [Consentire agli utenti di visualizzare le loro autorizzazioni](#)

## Best practice per le policy

Identity-based le politiche determinano se qualcuno può creare, accedere o eliminare CloudShell risorse nel tuo account. Queste operazioni possono comportare costi aggiuntivi per l' Account AWS. Quando si creano o modificano policy basate sull'identità, seguire 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 AWS gestite che concedono le autorizzazioni per molti casi d'uso comuni. Sono disponibili nel tuo. Account AWS Ti consigliamo di ridurre ulteriormente le autorizzazioni definendo politiche gestite dai AWS clienti

specifiche per i tuoi casi d'uso. Per maggiori informazioni, consulta [Policy gestite da AWS](#) o [Policy gestite da AWS per le funzioni dei processi](#) nella Guida per l'utente di IAM.

- Applicazione delle autorizzazioni con privilegio minimo - Quando si impostano le autorizzazioni con le policy IAM, concedere solo le autorizzazioni richieste per eseguire un'attività. È possibile farlo definendo le azioni che possono essere intraprese su risorse specifiche in condizioni specifiche, note anche come autorizzazioni con privilegio minimo. Per maggiori 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 ad azioni e risorse è possibile aggiungere una condizione alle 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 CloudFormation. Per maggiori informazioni, consultare la sezione [Elementi delle policy JSON di IAM: condizione](#) nella Guida per l'utente di IAM.
- Utilizzo dello strumento di analisi degli accessi IAM per convalidare le policy IAM e garantire autorizzazioni sicure e funzionali - Lo strumento di analisi degli accessi IAM convalida le policy nuove ed esistenti in modo che aderiscano al linguaggio (JSON) della policy IAM e alle best practice di IAM. Lo strumento di analisi degli accessi IAM offre oltre 100 controlli delle policy e consigli utili per creare policy sicure e funzionali. Per maggiori informazioni, consultare [Convalida delle policy per il Sistema di analisi degli accessi IAM](#) 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, aggiungere le condizioni MFA alle policy. Per maggiori informazioni, consultare [Protezione dell'accesso API 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 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, è opportuno concedere l'accesso solo alle azioni 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 maggiori informazioni, consulta [Aggiunta di autorizzazioni a un utente](#) nella Guida per l'utente di 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 collegate 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:ListGroupForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ]
    }
  ]
}
```

```
    ],  
    "Resource": "*"    
  }  
]  
}
```

## Risoluzione dei problemi di CloudShell identità e accesso AWS

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

### Argomenti

- [Non sono autorizzato a eseguire alcuna azione in CloudShell](#)
- [Non sono autorizzato a eseguire iam: PassRole](#)
- [Voglio consentire l'accesso a persone esterne al mio Account AWS per accedere 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 `awes:GetWidget` fittizie.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:  
awes: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 `awes: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'operazione richiede che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per trasmettere 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 l'accesso a persone esterne al mio Account AWS per accedere 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 degli accessi (ACL), utilizzare tali policy per concedere alle persone l'accesso alle proprie risorse.

Per maggiori 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 IAM.
- Per informazioni sulle differenze di utilizzo tra ruoli e policy basate su risorse per l'accesso multi-account, consulta [Accesso a risorse multi-account in IAM](#) nella Guida per l'utente di IAM.

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

Con le risorse di gestione degli accessi che possono essere fornite da AWS Identity and Access Management, 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 ogni servizio e risorsa 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 in AWS CloudShell utilizzo di politiche 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. Console di gestione AWS

### Important

Per eseguire l'avvio AWS CloudShell da Console di gestione AWS, 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. Console di gestione AWS	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. Non è disponibile per gli ambienti VPC.	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. Non è disponibile per gli ambienti VPC.	
<code>cloudshell:DescribeEnvironments</code>	Descrive gli ambienti.	No
<code>cloudshell:PutCredentials</code>	Inoltra le credenziali utilizzate per accedere a. Console di gestione AWS 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
<code>cloudshell:ApproveCommand</code>	Approva un comando inviato CloudShell da altre console AWS di servizio.	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.

## JSON

```
{
  "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.

## JSON

```
{
  "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 Console di gestione AWS vengano inoltrate all'ambiente. CloudShell Gli utenti IAM con questa policy devono configurare manualmente le proprie credenziali all'interno. CloudShell

JSON

```
{
  "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.

JSON

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

```
    "Action": [  
        "cloudshell:CreateEnvironment",  
        "cloudshell:CreateSession",  
        "cloudshell:GetEnvironmentStatus",  
        "cloudshell:StartEnvironment"  
    ],  
    "Resource": "*" ]  
}
```

## Autorizzazioni IAM richieste per la creazione e l'utilizzo di ambienti CloudShell VPC

Per creare e utilizzare ambienti CloudShell VPC, l'amministratore IAM deve abilitare l'accesso alle autorizzazioni Amazon EC2 specifiche per VPC. Questa sezione elenca le autorizzazioni Amazon EC2 necessarie per creare e utilizzare ambienti VPC.

Per creare ambienti VPC, la policy IAM assegnata al tuo ruolo deve includere le seguenti autorizzazioni Amazon EC2:

- `ec2:DescribeVpcs`
- `ec2:DescribeSubnets`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeDhcpOptions`
- `ec2:DescribeNetworkInterfaces`
  
- `ec2:CreateTags`
- `ec2:CreateNetworkInterface`
- `ec2:CreateNetworkInterfacePermission`

Ti consigliamo di includere anche:

- `ec2:DeleteNetworkInterface`

**Note**

Questa autorizzazione non è obbligatoria, ma è necessaria per CloudShell ripulire la risorsa ENI (gli ENI creati per gli ambienti CloudShell VPC sono contrassegnati `ManagedByCloudShell` con la chiave) da essa creata. Se questa autorizzazione non è abilitata, è necessario pulire manualmente la risorsa ENI dopo ogni utilizzo dell'ambiente CloudShell VPC.

Policy IAM che garantisce CloudShell l'accesso completo, incluso l'accesso al VPC

L'esempio seguente mostra come abilitare le autorizzazioni complete, incluso l'accesso al VPC, per: CloudShell

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowCloudShellOperations",
      "Effect": "Allow",
      "Action": [
        "cloudshell:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowDescribeVPC",
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeDhcpOptions",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeSubnets",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeVpcs"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AllowCreateTagWithCloudShellKey",
```

```

    "Effect": "Allow",
    "Action": [
      "ec2:CreateTags"
    ],
    "Resource": "arn:aws:ec2:*:*:network-interface/*",
    "Condition": {
      "StringEquals": {
        "ec2:CreateAction": "CreateNetworkInterface"
      },
      "ForAnyValue:StringEquals": {
        "aws:TagKeys": "ManagedByCloudShell"
      }
    }
  },
  {
    "Sid": "AllowCreateNetworkInterfaceWithSubnetsAndSG",
    "Effect": "Allow",
    "Action": [
      "ec2:CreateNetworkInterface"
    ],
    "Resource": [
      "arn:aws:ec2:*:*:subnet/*",
      "arn:aws:ec2:*:*:security-group*"
    ]
  },
  {
    "Sid": "AllowCreateNetworkInterfaceWithCloudShellTag",
    "Effect": "Allow",
    "Action": [
      "ec2:CreateNetworkInterface"
    ],
    "Resource": "arn:aws:ec2:*:*:network-interface/*",
    "Condition": {
      "ForAnyValue:StringEquals": {
        "aws:TagKeys": "ManagedByCloudShell"
      }
    }
  },
  {
    "Sid": "AllowCreateNetworkInterfacePermissionWithCloudShellTag",
    "Effect": "Allow",
    "Action": [
      "ec2:CreateNetworkInterfacePermission"
    ],
  },

```

```

    "Resource": "arn:aws:ec2:*:*:network-interface/*",
    "Condition": {
      "StringEquals": {
        "aws:ResourceTag/ManagedByCloudShell": ""
      }
    },
    {
      "Sid": "AllowDeleteNetworkInterfaceWithCloudShellTag",
      "Effect": "Allow",
      "Action": [
        "ec2:DeleteNetworkInterface"
      ],
      "Resource": "arn:aws:ec2:*:*:network-interface/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/ManagedByCloudShell": ""
        }
      }
    }
  ]
}

```

## Utilizzo delle chiavi di condizione IAM per ambienti VPC

Puoi utilizzare le chiavi di CloudShell-specific condizione per le impostazioni VPC per fornire controlli di autorizzazione aggiuntivi per i tuoi ambienti VPC. Puoi anche specificare le sottoreti e i gruppi di sicurezza che l'ambiente VPC può e non può utilizzare.

CloudShell supporta le seguenti chiavi di condizione nelle politiche IAM:

- `CloudShell:VpcIds`— Consentire o negare uno o più VPC
- `CloudShell:SubnetIds`— Consentire o negare una o più sottoreti
- `CloudShell:SecurityGroupIds`— Consentire o negare uno o più gruppi di sicurezza

### Note

Se le autorizzazioni per gli utenti con accesso agli CloudShell ambienti pubblici vengono modificate per aggiungere restrizioni all'`cloudshell:createEnvironmentazione`, possono comunque accedere all'ambiente pubblico esistente. Tuttavia, se desideri modificare

una policy IAM con questa restrizione e disabilitare il loro accesso all'ambiente pubblico esistente, devi prima aggiornare la policy IAM con la restrizione, quindi assicurarti che ogni CloudShell utente del tuo account elimini manualmente l'ambiente pubblico esistente utilizzando l'interfaccia utente CloudShell web (Azioni → Elimina ambiente). CloudShell

## Policy di esempio con chiavi di condizione per le impostazioni VPC

Negli esempi seguenti viene illustrato come utilizzare le chiavi di condizione per le impostazioni VPC. Dopo aver creato un'istruzione delle policy con le restrizioni desiderate, aggiungere l'istruzione delle policy per l'utente o il ruolo di destinazione.

Assicurati che gli utenti creino solo ambienti VPC e neghi la creazione di ambienti pubblici

Per garantire che gli utenti possano creare solo ambienti VPC, utilizza l'autorizzazione di negazione come mostrato nell'esempio seguente:

```
{
  "Statement": [
    {
      "Sid": "DenyCloudShellNonVpcEnvironments",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
      "Condition": {
        "Null": {
          "cloudshell:VpcIds": "true"
        }
      }
    }
  ]
}
```

Negare agli utenti l'accesso a specifici VPC, sottoreti o gruppi di sicurezza

Per negare agli utenti l'accesso a specifici VPC, utilizzare `StringEquals` per verificare il valore della condizione `cloudshell:VpcIds`. L'esempio seguente nega agli utenti l'accesso a `vpc-1 vpc-2`

Per negare agli utenti l'accesso a specifici VPC, utilizzare `StringEquals` per verificare il valore della condizione `cloudshell:SubnetIds`. L'esempio seguente nega agli utenti l'accesso a `subnet-1` e `subnet-2`

Per negare agli utenti l'accesso a specifici VPC, utilizzare `StringEquals` per verificare il valore della condizione `cloudshell:SecurityGroupIds`. L'esempio seguente nega agli utenti l'accesso a `sg-1` e `sg-2`

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceOutOfSecurityGroups",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Deny",
      "Resource": "*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "cloudshell:SecurityGroupIds": [
            "sg-1",
            "sg-2"
          ]
        }
      }
    }
  ]
}
```

Consenti agli utenti di creare ambienti con configurazioni VPC specifiche

Per consentire agli utenti di accedere a VPC specifici, utilizza `StringEquals` per verificare il valore della condizione. `cloudshell:VpcIds` L'esempio seguente consente agli utenti di accedere a `vpc-1` e `vpc-2`:

Per consentire agli utenti di accedere a VPC specifici, utilizza `StringEquals` per verificare il valore della `cloudshell:SubnetIds` condizione. L'esempio seguente consente agli utenti di accedere a `subnet-1` e `subnet-2`:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceStayInSpecificSubnets",
      "Action": [
        "cloudshell:CreateEnvironment"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "ForAllValues:StringEquals": {
          "cloudshell:SubnetIds": [
            "subnet-1",
            "subnet-2"
          ]
        }
      }
    }
  ]
}
```

Per consentire agli utenti di accedere a VPC specifici, utilizza `StringEquals` per verificare il valore della `cloudshell:SecurityGroupIds` condizione. L'esempio seguente consente agli utenti di accedere a `sg-1` e `sg-2`:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "EnforceStayInSpecificSecurityGroup",
      "Action": [
```

```
    "cloudshell:CreateEnvironment"
  ],
  "Effect": "Allow",
  "Resource": "*",
  "Condition": {
    "ForAllValues:StringEquals": {
      "cloudshell:SecurityGroupIds": [
        "sg-1",
        "sg-2"
      ]
    }
  }
}
```

## Autorizzazioni per l'accesso Servizi AWS

CloudShell utilizza le credenziali IAM utilizzate per accedere a Console di gestione AWS

### Note

Per utilizzare le credenziali IAM utilizzate per accedere a Console di gestione AWS, 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 JSON-formatted policy. 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.

## Autorizzazioni per l'accesso alle funzionalità della CLI di Amazon Q in CloudShell

Per utilizzare le funzionalità della CLI di Amazon Q CloudShell, come suggerimenti in linea, chat e traduzione, assicurati di disporre delle autorizzazioni IAM richieste. Se non riesci ad accedere alle funzionalità dell'interfaccia a riga di comando di Amazon Q in CloudShell, contatta il tuo amministratore per fornirti le autorizzazioni IAM necessarie. Per ulteriori informazioni, consulta [esempi di Identity-based policy per Amazon Q Developer](#) nella Amazon Q Developer User Guide.

## 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	Description
<code>createEnvironment</code>	Si verifica quando viene creato un CloudShell ambiente.
<code>createSession</code>	Si verifica quando un CloudShell ambiente è connesso da Console di gestione AWS.
<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>cloudshell:DescribeEnvironments</code>	Descrive gli ambienti.
<code>getLayout*</code>	Si verifica quando viene recuperato il CloudShell layout all'inizio della sessione.

Nome evento	Description
<code>putCredentials</code>	Si verifica quando le credenziali utilizzate per accedere a Console di gestione AWS 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

Third-party i revisori 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 alle norme ISO/IEC 27001:2013, 27017:2015, 27018:2019, 27701:2019, 22301:2019, 9001:2015e 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 Moderato ( ) East/West</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:

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

Servizio	SDK	<a href="#">DoD CC SRG IL2 () East/West</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)AWS</a>
AWS CloudShell	CloudShell	✓	✓	✓	✓	N/A

## 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/A	✓

\*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 del sistema di gestione della sicurezza delle 27001/02 informazioni (ISMS).

Servizio	SDK	US-East(O hio)	US-East(N .Virginia)	US-West(O regon)	US-West(N .California)	Singapore	Seul
AWS CloudShell	CloudShell	✓	✓	✓	N/A	N/A	N/A

## C5

Il Cloud Computing Compliance Controls Catalog (C5) è uno schema di Government-backed attestazione tedesco introdotto in Germania dall'Ufficio federale per la sicurezza delle informazioni (BSI) per aiutare le organizzazioni a dimostrare la sicurezza operativa contro i comuni attacchi informatici quando utilizzano servizi cloud nel contesto delle «Raccomandazioni di sicurezza per i 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 come le aziende possono utilizzare](#) per creare applicazioni. AWS HIPAA-compliant
- [AWS Risorse per la conformità](#) [Risorse](#) per : questa raccolta di cartelle di lavoro e guide potrebbe riguardare il settore e la località in cui operi.
- [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 CSPM](#)— 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 le seguenti funzionalità per soddisfare le esigenze di resilienza e backup dei dati:

- 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 la [Guida introduttiva a. AWS CloudShell](#)

## 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 crittografia con Perfect Forward Secrecy (PFS) come DHE (Ephemeral) o ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Diffie-Hellman La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

#### Note

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

## 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, ti consigliamo di considerarle utili anziché come 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, scaricare e caricare file) richieste dal loro ruolo. Per ulteriori informazioni, consulta [Gestione dell' AWS CloudShell accesso e dell'utilizzo con le politiche 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 sull'utilizzo di Safe Paste per il testo su più righe, vedere [Uso di Safe Paste per il testo su più righe](#).
- Acquisisci familiarità con lo [Shared Security Responsibility Model](#) se installi 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 sulla modifica dell'ambiente shell predefinito, consultate [Modificare la shell con gli script](#).
- Archivia il codice in modo sicuro in un sistema di controllo delle versioni.

## AWS CloudShell Domande frequenti sulla sicurezza

Di seguito sono riportate le risposte alle domande frequenti sulla sicurezza per CloudShell.

- [What AWS i processi e le tecnologie vengono 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?](#)
- [Dov'è il mio \\$CASA directory effettivamente memorizzata nel Cloud AWS?](#)
- [È possibile crittografare il mio \\$CASA elenco?](#)
- [Posso eseguire una scansione antivirus sul mio \\$CASA elenco?](#)
- [Posso limitare l'accesso root a un CloudShell utente all'interno del contenitore?](#)

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

Quando accedi Console di gestione AWS, 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?

Per gli ambienti pubblici, non è possibile limitare l'accesso alla rete. Se desideri limitare l'accesso alla rete, devi abilitare l'autorizzazione a creare solo ambienti VPC e negare la creazione di ambienti pubblici.

Per ulteriori informazioni, consulta [Assicurarsi che gli utenti creino solo ambienti VPC e negare la creazione di ambienti pubblici](#).

Per gli ambienti CloudShell VPC, le impostazioni di rete vengono ereditate dal tuo VPC. L'utilizzo CloudShell in un VPC consente di controllare l'accesso alla rete dell'ambiente CloudShell VPC.

[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](#)

## Dov'è il mio \$CASA directory effettivamente memorizzata nel Cloud AWS?

Per gli ambienti pubblici, l'infrastruttura per l'archiviazione dei dati \$HOME è fornita da Amazon S3.

Per gli ambienti VPC, la \$HOME directory viene eliminata quando l'ambiente VPC scade (dopo 20-30 minuti di inattività) o quando si elimina o si riavvia l'ambiente.

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

## È possibile crittografare il mio \$CASA elenco?

No, non è possibile cifrare la \$HOME cartella con una propria chiave. Ma CloudShell crittografa il contenuto della tua \$HOME directory mentre lo archivia in Amazon S3.

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

## Posso eseguire una scansione antivirus sul mio \$CASA elenco?

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](#)

## Posso limitare l'ingresso o l'uscita dei dati per me? CloudShell

Per limitare l'ingresso o l'uscita, ti consigliamo di utilizzare un ambiente VPC CloudShell . La \$HOME directory di un ambiente VPC viene eliminata quando l'ambiente VPC scade (dopo 20-30 minuti di inattività) o quando si elimina o si riavvia l'ambiente. Nel menu Azioni, le opzioni di caricamento e download non sono disponibili per gli ambienti VPC.

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

## Posso limitare l'accesso root a un CloudShell utente all'interno del contenitore?

No. AWS CloudShell fornisce l'accesso root all'interno del contenitore di elaborazione fin dalla progettazione. I contenitori in dotazione CloudShell fungono da imballaggio del codice e da praticità operativa. Non sono limiti di sicurezza.

CloudShell gestisce il controllo degli accessi tramite AWS Identity and Access Management (IAM). Ogni CloudShell sessione riceve credenziali IAM temporanee, ruotate regolarmente, nell'ambito delle autorizzazioni dell'utente. Queste credenziali sono il limite di sicurezza, non il contenitore stesso.

Poiché il contenitore e l'istanza sottostante condividono lo stesso ambito di credenziali IAM, l'accesso oltre il confine del contenitore non fornisce autorizzazioni aggiuntive. AWS

[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. Assicurati che qualsiasi software che installi nell'ambiente di calcolo sia patchato e aggiornato. Puoi anche 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 specificando l'elenco delle autorizzazioni URLs, 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	Description	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>

## AWS interfacce a riga di comando (CLI)

### CLI

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

Nome	Description	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="#">Gestisci AWS i servizi dalla CLI in 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	Description	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	Description	Version information (Informazioni relative alla versione)
AWS SAM CLI	<p>AWS SAM CLI 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	Description	Version information (Informazioni relative alla versione)
AWS Strumenti per PowerShell	<p>AWS Strumenti per PowerShell Sono PowerShell moduli basati sulle funzionalità esposte da. SDK per .NET Con AWS Strumenti per 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 Strumenti per PowerShell Per ulteriori informazioni, consulta <a href="#">Using the AWS Tools for PowerShell</a> nella AWS Strumenti per PowerShell User Guide.</p>	<pre>pwsh --Command 'Get-AWSPowerShellVersion'</pre>

## Runtimes e AWS SDKs: Node.js e Python 3

### Runtimes e AWS SDKs

Nome	Description	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	Description	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 per JavaScript</a>.</p>	<pre>npm -g ls --depth 0 2&gt;/dev/null   grep aws-sdk</pre>

Nome	Description	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	Description	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	Description	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	Description	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 in <code>.bashrc</code></p> <p>Per ulteriori informazioni, consultate la pagina <a href="#">README</a> del progetto su. GitHub</p>	

Nome	Description	Version information (Informazioni relative alla versione)
cqlsh-expansion	<p>cqlsh-expansion è un toolkit che include cqlsh e helper preconfigurati per Amazon Keyspaces pur mantenendo la piena compatibilità con Apache Cassandra. Per ulteriori informazioni, consulta <a href="#">Using cqlsh per connettersi ad Amazon Keyspaces nella Amazon Keyspaces (for Apache Cassandra) Developer Guide</a>.</p>	<pre>cqlsh-expansion --version</pre>

Nome	Description	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 AWS regioni sono supportate con Docker, consulta Regioni supportate per. AWSAWS CloudShell</a> 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>	<pre>docker --version</pre>

Nome	Description	Version information (Informazioni relative alla versione)
Git	Git è un sistema di controllo delle versioni distribuito che supporta le moderne pratiche di sviluppo software attraverso i 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> .	<code>git --version</code>
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 <code>iputils: 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 da riga di comando per comunicare con il piano di controllo di un cluster Kubernetes, utilizzando l'API Kubernetes.	<code>kubectl --version</code>

Nome	Description	Version information (Informazioni relative alla versione)
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>
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>

Nome	Description	Version information (Informazioni relative alla versione)
OpenJDK 21	Amazon Corretto 21 è una distribuzione LTS (Long-Term Supported) di <a href="#">OpenJDK 21</a> . Amazon Corretto è una distribuzione pronta per la produzione gratuita, con un ambiente multiplatforma di Open Java Development Kit (OpenJDK). Per ulteriori informazioni, consulta <a href="#">Cos'è Amazon Corretto 21?</a> nella Guida per l'utente di Corretto 21.	java -version
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.	ps --version

Nome	Description	Version information (Informazioni relative alla versione)
psql	PostgreSQL è un potente sistema di database open source che utilizza funzionalità SQL standard fornendo al contempo funzionalità robuste per la gestione e la scalabilità sicure di operazioni di dati complesse. Per ulteriori informazioni, consulta <a href="#">Cos'è PostgreSQL</a> .	<code>psql --version</code>
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>

Nome	Description	Version information (Informazioni relative alla versione)
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> .	tar --version
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
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	Description	Version information (Informazioni relative alla versione)
zip/unzip	Le zip/unzip utilità 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.	<code>unzip --version</code> <code>zip --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, usa 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 di AWS 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 a AL2 AL2023

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

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

AL2023 fornisce 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 [Confronto tra Amazon Linux 2 e Amazon Linux 2023](#) nella Guida per l'utente di Amazon Linux 2023.

Se hai domande, contatta [Supporto](#). 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 Migrazione FAQs

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

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

### La migrazione AL2023 influirà su altre mie AWS risorse, come le istanze Amazon EC2 in esecuzione? AL2

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

### A quali pacchetti sono stati modificati con la migrazione? 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 [Comparing AL2 and AL2023](#). Per i clienti con requisiti specifici di pacchetto e versione che non saranno più soddisfatti dopo la migrazione a AL2023, consigliamo di contattare l'AWS assistenza per inviare una richiesta.

### Posso rinunciare alla migrazione?

No, non puoi rinunciare alla migrazione. AWS CloudShell gli ambienti sono gestiti da AWS, pertanto, tutti gli ambienti sono stati aggiornati a AL2023.

## Posso creare un backup del mio AWS CloudShell ambiente?

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 AWS CloudShell

Durante l'utilizzo 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 potreste 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

- [Accesso negato](#)
- [Autorizzazioni insufficienti](#)
- [Impossibile accedere AWS CloudShell riga di 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.NetCoremodulo](#)
- [Docker non è in esecuzione quando si utilizza AWS CloudShell](#)
- [Docker ha esaurito lo spazio su disco](#)
- [docker push sta scadendo e continua a riprovare](#)
- [Impossibile accedere alle risorse all'interno del VPC dal mio AWS CloudShell Ambiente VPC](#)
- [L'ENI utilizzato da AWS CloudShell perché il mio ambiente VPC non è stato pulito](#)
- [L'utente con CreateEnvironmentautorizzazione solo per gli ambienti VPC ha anche accesso al pubblico AWS CloudShell ambienti](#)

## Accesso negato

**Problema:** quando si tenta di eseguire l'avvio CloudShell da Console di gestione AWS, viene visualizzato il messaggio "Unable to start the environment. Per riprovare, aggiorna il browser o riavvia selezionando Azioni, Riavvia AWS CloudShell». Ti viene negato l'accesso anche dopo aver richiesto le autorizzazioni all'amministratore IAM e dopo aver aggiornato il browser o riavviato. CloudShell

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

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

## Autorizzazioni insufficienti

**Problema:** quando si tenta di CloudShell eseguire l'avvio da Console di gestione AWS, viene visualizzato il messaggio «Impossibile avviare l'ambiente. Non disponi delle autorizzazioni necessarie. Chiedi al tuo amministratore IAM di concedere l'accesso a AWS CloudShell". 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 al tuo 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 AWS CloudShell accesso e utilizzo con le policy IAM](#).

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

## Impossibile accedere AWS CloudShell riga di comando

**Problema:** dopo aver modificato un file utilizzato dall'ambiente di calcolo, non puoi 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 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 CloudShell 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 byImport-Module -Name AWSPowerShell.NetCore`, viene visualizzato il seguente messaggio di errore:

Import-Module: Il modulo 'AWSPowerShell.NetCore' specificato 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 Strumenti per 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. `GovCloud` Assicurati di utilizzare `Docker` nelle regioni `AWS`

supportate. Per un elenco delle regioni in cui Docker è disponibile, consulta [AWS Regioni supportate](#) per. AWS CloudShell

## 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` sta scadendo e continua a riprovare

Problema: quando esegui, il timeout `docker push` è scaduto e 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.

## Impossibile accedere alle risorse all'interno del VPC dal mio AWS CloudShell Ambiente VPC

Problema: impossibile accedere alle risorse all'interno del VPC durante l'utilizzo del mio ambiente VPC AWS CloudShell .

Causa: il tuo ambiente AWS CloudShell VPC eredita le impostazioni di rete del tuo VPC.

Soluzione: per risolvere questo problema, assicurati che il tuo VPC sia configurato correttamente per accedere alle tue risorse. [Per ulteriori informazioni, consulta la documentazione VPC Connect](#)

[your VPC ad altre reti e la documentazione Network Access Analyzer Network Access Analyzer](#). Puoi trovare l'indirizzo IPv4 utilizzato dall'ambiente AWS CloudShell VPC eseguendo il comando ``ip -a`` all'interno del tuo ambiente nel prompt della riga di comando o nella pagina della console VPC.

## L'ENI utilizzato da AWS CloudShell perché il mio ambiente VPC non è stato pulito

Problema: non riesco a ripulire l'ENI utilizzato dal AWS CloudShell mio ambiente VPC.

Causa: `ec2:DeleteNetworkInterface` l'autorizzazione non è abilitata per il tuo ruolo.

Soluzione: per risolvere questo problema, assicurati che `ec2:DeleteNetworkInterface` l'autorizzazione sia abilitata per il tuo ruolo, come mostrato nel seguente script di esempio:

```
{
  "Effect": "Allow",
  "Action": [
    "ec2:DeleteNetworkInterface"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceTag/ManagedByCloudShell": ""
    }
  },
  "Resource": "arn:aws:ec2:*:*:network-interface/*"
}
```

## L'utente con **CreateEnvironment** autorizzazione solo per gli ambienti VPC ha anche accesso al pubblico AWS CloudShell ambienti

Problema: gli utenti `CreateEnvironment` autorizzati ai soli ambienti VPC possono accedere anche agli ambienti pubblici AWS CloudShell .

Causa: se limiti `CreateEnvironment` le autorizzazioni solo per la creazione di ambienti VPC e se hai già creato un ambiente pubblico, manterrai l'accesso all'ambiente CloudShell pubblico esistente fino a quando questo ambiente non verrà eliminato utilizzando l'interfaccia utente web. Ma se non l'hai mai usato CloudShell prima, non avrai accesso agli ambienti pubblici.

Soluzione: per limitare l'accesso agli AWS CloudShell ambienti pubblici, l'amministratore IAM deve prima aggiornare la policy IAM con la restrizione, quindi l'utente deve eliminare manualmente

l'ambiente pubblico esistente utilizzando l'interfaccia utente AWS CloudShell web. (Azioni → Elimina CloudShell ambiente).

# Supportata AWS Regioni per AWS CloudShell

Per informazioni sulle AWS regioni e gli endpoint di servizio AWS CloudShell supportati, consulta le seguenti risorse:

- [Regioni AWS](#)
- [Service endpoints \(Endpoint del servizio\)](#)

# Quote di servizio e restrizioni per AWS CloudShell

Questa pagina descrive le quote e le restrizioni del servizio che si applicano alle seguenti aree:

- [Archiviazione persistente](#)
- [Utilizzo mensile](#)
- [Shell concorrenti](#)
- [Dimensione del comando](#)
- [Sessioni di shell](#)
- [Ambienti VPC](#)
- [Accesso alla rete e trasferimento dati](#)
- [File di sistema e ricaricamenti delle pagine](#)

## Storage persistente

Con AWS CloudShell, hai a disposizione uno spazio di archiviazione persistente di 1 GB per ciascuno senza alcun Regione AWS costo. L'archiviazione persistente si trova nella tua home directory (\$HOME) ed è privata per te. 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.

### Note

CloudShell Gli ambienti VPC non dispongono di storage persistente. La directory \$HOME viene eliminata quando l'ambiente VPC scade (dopo 20-30 minuti di inattività) o quando si elimina l'ambiente.

Se smetti di utilizzare AWS CloudShell in un Regione 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 non si intervenga, i dati vengono automaticamente eliminati dall'archivio persistente di quella regione. È possibile impedire la rimozione AWS CloudShell riavviando l'operazione in tale Regione AWS area. Per ulteriori informazioni, consulta [Passaggio 2: Seleziona una regione AWS CloudShell, avvia e scegli una shell](#).

### Note

#### Scenario di utilizzo

Márcia era solita AWS CloudShell archiviare i file nelle sue home directory in due Regioni AWS: Stati Uniti orientali (Virginia settentrionale) ed Europa (Irlanda). Ha quindi iniziato a utilizzare AWS CloudShell esclusivamente in Europa (Irlanda) e ha smesso di lanciare sessioni di shell negli Stati Uniti orientali (Virginia settentrionale).

Prima della scadenza per l'eliminazione dei dati negli Stati Uniti orientali (Virginia settentrionale), Márcia decide di impedire il riciclaggio della sua home directory avviando AWS CloudShell e selezionando nuovamente la regione Stati Uniti orientali (Virginia settentrionale). Poiché ha sempre utilizzato l'Europa (Irlanda) per le sessioni di shell, la sua archiviazione persistente in quella regione non ne risente.

## Utilizzo mensile

Ciascuna Regione AWS di voi Account AWS ha una quota di utilizzo mensile per AWS CloudShell. Questa quota combina il tempo totale impiegato CloudShell da tutti i responsabili IAM in quella regione. Se tenti di accedere CloudShell dopo aver raggiunto la quota mensile per quella regione, viene visualizzato un messaggio che spiega perché l'ambiente shell non può essere avviato.

Per richiedere un aumento utilizzando la console Service Quotas

Puoi richiedere un aumento delle quote di utilizzo mensili aprendo la console [Service Quotas](#). Per ulteriori informazioni, consulta [Richiesta di un aumento delle quote nella](#) Guida per l'utente di Service Quotas.

## Shell simultanee

Puoi lanciare fino a 10 shell contemporaneamente in ciascuna Regione AWS per il tuo account.

Per richiedere un aumento utilizzando la console Service Quotas

Puoi richiedere un aumento della quota per ogni regione aprendo la console [Service Quotas](#). Per ulteriori informazioni, consulta [Richiesta di un aumento delle quote nella](#) Guida per l'utente di Service Quotas.

## Dimensione del 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 il linguaggio di tua scelta, quindi esegilo dall'interfaccia a riga di comando. [Per ulteriori informazioni sulla gamma di software preinstallato 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: Guida introduttiva](#). AWS CloudShell

## Sessioni di shell

- Sessioni inattive: AWS CloudShell è un ambiente shell interattivo: se non interagisci con esso utilizzando la tastiera o il puntatore per 20-30 minuti, la sessione di shell termina. I processi in esecuzione non contano come interazioni.

Se desideri eseguire attività basate su terminali utilizzando un servizio AWS con timeout più flessibili, ti consigliamo di avviare e connetterti a [un'istanza Amazon EC2](#).

- Sessioni di lunga durata: una sessione di shell che viene eseguita ininterrottamente per circa 12 ore termina automaticamente anche se l'utente interagisce regolarmente con essa durante quel periodo.

## Ambienti VPC

Puoi creare solo fino a due ambienti VPC per principale IAM.

### Note

La connessione al tuo VPC privato e l'accesso alle risorse al suo interno sono gratuite. I trasferimenti di dati all'interno del tuo VPC privato sono inclusi nella fatturazione VPC e i trasferimenti di dati tra le tue reti CloudShell vengono addebitati allo stesso costo del tuo VPCs VPC attuale. CloudShell

## Accesso alla rete e trasferimento dei dati

Le seguenti restrizioni si applicano sia al traffico in entrata che in uscita dell'ambiente in uso: AWS CloudShell

- In uscita: puoi 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, c'è il rischio che determinati 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, consulta [Gestione delle azioni consentite in AWS CloudShell utilizzo di politiche personalizzate](#)

Trasferimento dati: il caricamento e lo scaricamento 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 sui ricaricamenti 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'ambiente. AWS CloudShell In tal caso, potrebbe essere necessario [eliminare la home directory](#) per riottenere l'accesso.
- Ricaricamento delle pagine: per ricaricare l' AWS CloudShell interfaccia, utilizzate il pulsante di aggiornamento del browser anziché la sequenza di tasti di scelta rapida predefinita per il sistema operativo.

# Cronologia dei documenti per la Guida per AWS CloudShell l'utente

## Aggiornamenti recenti

Nella tabella seguente sono descritte le modifiche importanti apportate alla Guida per l'utente di AWS CloudShell .

Modifica	Descrizione	Data
<a href="#">CLI di Amazon Q in AWS CloudShell</a>	È stato aggiunto il supporto per l'utilizzo delle funzionalità CLI di Amazon Q in. AWS CloudShell	2 ottobre 2024
<a href="#">Supporto Amazon VPC per alcune AWS CloudShell regioni</a>	È stato aggiunto il supporto per la creazione e l'utilizzo di ambienti AWS CloudShell VPC in determinate regioni.	13 giugno 2024
<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.	04 dicembre 2023

[Nuove regioni AWS per AWS CloudShell](#)

AWS CloudShell è ora disponibile a livello generale nelle seguenti AWS regioni:

16 giugno 2023

- Stati Uniti occidentali (California settentrionale)
- Africa (Città del Capo)
- Asia Pacifico (Hong Kong)
- Asia Pacifico (Osaka)
- Asia Pacifico (Seul)
- Asia Pacifico (Giacarta)
- Asia Pacifico (Singapore)
- Europa (Parigi)
- Europe (Stockholm)
- Europe (Milan)
- Medio Oriente (Bahrein)
- Medio Oriente (Emirati Arabi Uniti)

[AWS CloudShell Avvia su Console Toolbar](#)

Avvia CloudShell su Console Toolbar, nella parte inferiore sinistra della console scegliendo CloudShell.

28 marzo 2023

[Nuove AWS regioni per AWS CloudShell](#)

AWS CloudShell è ora disponibile nelle seguenti AWS regioni:

6 ottobre 2022

- Canada (Centrale)
- Europa (Londra)
- Sud America (San Paolo)

[AWS CloudShell supportato negli Stati Uniti in AWS GovCloud](#)

AWS CloudShell è ora supportato nella regione AWS GovCloud (Stati Uniti).

29 giugno 2022

---

<a href="#">Sicurezza FAQs</a>	Inoltre FAQs incentrato 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 delle impostazioni predefinite 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.