TR-4921: Difesa dal ransomware

How to enable StorageGRID in your environment

NetApp
July 05, 2024

Sommario

TR-4921: Difesa dal ransomware

Proteggere gli oggetti StorageGRID S3 dal ransomware

Scopri di più sugli attacchi ransomware e su come proteggere i dati grazie alle Best practice di sicurezza di StorageGRID.

Gli attacchi ransomware sono in aumento. Questo documento fornisce alcuni consigli su come proteggere i dati degli oggetti su StorageGRID.

Il ransomware di oggi è il pericolo sempre presente nel data center. Il ransomware è progettato per crittografare i dati e renderli inutilizzabili dagli utenti e dalle applicazioni che li fanno affidamento. La protezione inizia con le solite difese di reti rafforzate e solide pratiche di sicurezza per gli utenti, e dobbiamo seguire le procedure di sicurezza per l'accesso ai dati.

Il ransomware è una delle maggiori minacce alla sicurezza odierne. Il team NetApp StorageGRID collabora con i nostri clienti per stare al passo con queste minacce. Con l'uso del blocco degli oggetti e del controllo delle versioni, è possibile proteggere da modifiche indesiderate e ripristinare da attacchi dannosi. La sicurezza dei dati è un'impresa multi-layer che considera lo storage a oggetti solo una parte del data center.

Best practice di StorageGRID

Per StorageGRID, le Best practice sulla sicurezza devono includere l'utilizzo di HTTPS con certificati firmati sia per la gestione che per l'accesso agli oggetti. Crea account utente dedicati per applicazioni e singoli utenti e non utilizza gli account root tenant per l'accesso alle applicazioni o ai dati utente. In altre parole, seguire il principio del privilegio minimo. Utilizzare i gruppi di protezione con criteri IAM (Identity and Access Management) definiti per gestire i diritti degli utenti e accedere agli account specifici per le applicazioni e gli utenti. Con queste misure in atto, devi comunque assicurarti che i tuoi dati siano protetti. Nel caso di Simple Storage Service (S3), quando gli oggetti vengono modificati per crittografarli, viene eseguita la sovrascrittura dell'oggetto originale.

Metodi di difesa

Il meccanismo di protezione dal ransomware primario nell'API S3 consiste nell'implementare il blocco degli oggetti. Non tutte le applicazioni sono compatibili con il blocco degli oggetti, pertanto sono disponibili altre due opzioni per proteggere gli oggetti descritti in questo report: La replica in un altro bucket con la versione abilitata e la versione con i criteri IAM.

Dove trovare ulteriori informazioni

Per ulteriori informazioni sulle informazioni descritte in questo documento, consultare i seguenti documenti e/o siti Web:

- Centro di documentazione NetApp StorageGRID https://docs.netapp.com/us-en/storagegrid-118/
- Abilitazione NetApp StorageGRID https://docs.netapp.com/us-en/storagegrid-enable/
- Pagina risorse documentazione StorageGRID https://www.netapp.com/data-storage/storagegrid/documentation/
- Documentazione dei prodotti NetApp https://www.netapp.com/support-and-training/documentation/

Difesa dal ransomware tramite il blocco degli oggetti

Scopri come il blocco degli oggetti in StorageGRID fornisce un modello WORM per impedire la cancellazione o la sovrascrittura dei dati e come soddisfa i requisiti normativi.

Il blocco degli oggetti fornisce un modello WORM per impedire che gli oggetti vengano eliminati o sovrascritti. L'implementazione di StorageGRID del blocco degli oggetti è "Valutazione Cohasset" per aiutare a soddisfare i requisiti normativi, supportando la conservazione a fini giudiziari, la modalità di conformità e la modalità di governance per la conservazione degli oggetti e le policy di conservazione predefinite dei bucket. È necessario abilitare il blocco degli oggetti come parte della creazione e del controllo delle versioni del bucket. Una versione specifica di un oggetto è bloccata e, se non viene definito alcun ID di versione, la conservazione viene posizionata sulla versione corrente dell'oggetto. Se la versione corrente ha la conservazione configurata e si tenta di eliminare, modificare o sovrascrivere l'oggetto, viene creata una nuova versione con un marcatore di eliminazione o la nuova revisione dell'oggetto come versione corrente, e la versione bloccata viene mantenuta come una versione non corrente. Per le applicazioni non ancora compatibili, è comunque possibile utilizzare la configurazione di blocco degli oggetti e di conservazione predefinita inserita nel bucket. Una volta definita la configurazione, viene applicata una conservazione degli oggetti a ogni nuovo oggetto inserito nel bucket. Questa operazione funziona finché l'applicazione è configurata per non eliminare o sovrascrivere gli oggetti prima che sia trascorso il tempo di conservazione.

Di seguito sono riportati alcuni esempi di utilizzo dell'API di blocco degli oggetti:

Blocco oggetto conservazione legale è un semplice stato on/off applicato a un oggetto.

```
aws s3api put-object-legal-hold --bucket mybucket --key myfile.txt --legal
-hold Status=ON --endpoint-url https://s3.company.com
```

L'impostazione dello stato di conservazione a fini giudiziari non restituisce alcun valore se l'operazione è riuscita, pertanto può essere verificata con un'operazione GET.

```
aws s3api get-object-legal-hold --bucket mybucket --key myfile.txt
--endpoint-url https://s3.company.com
{
    "LegalHold": {
        "Status": "ON"
    }
}
```

Per disattivare la sospensione legale, applicare lo stato OFF.

```
aws s3api put-object-legal-hold --bucket mybucket --key myfile.txt --legal
-hold Status=OFF --endpoint-url https://s3.company.com
aws s3api get-object-legal-hold --bucket mybucket --key myfile.txt
--endpoint-url https://s3.company.com
{
    "LegalHold": {
        "Status": "OFF"
    }
}
```

L'impostazione della conservazione dell'oggetto viene eseguita con un Retain until timestamp.

```
aws s3api put-object-retention --bucket mybucket --key myfile.txt
--retention '{"Mode":"COMPLIANCE", "RetainUntilDate": "2022-06-
10T16:00:00"}'  --endpoint-url https://s3.company.com
```

Anche in questo caso, non viene restituito alcun valore in caso di esito positivo, pertanto è possibile verificare lo stato di conservazione in modo simile con una chiamata Get.

```
aws s3api get-object-retention --bucket mybucket --key myfile.txt
--endpoint-url https://s3.company.com
{
    "Retention": {
        "Mode": "COMPLIANCE",
        "RetainUntilDate": "2022-06-10T16:00:00+00:00"
    }
```

L'inserimento di una conservazione predefinita in un bucket abilitato per il blocco degli oggetti utilizza un periodo di conservazione in giorni e anni.

```
aws s3api put-object-lock-configuration --bucket mybucket --object-lock
-configuration '{ "ObjectLockEnabled": "Enabled", "Rule": {
"DefaultRetention": { "Mode": "COMPLIANCE", "Days": 1 }}}' --endpoint-url
https://s3.company.com
```

Come per la maggior parte di queste operazioni, non viene restituita alcuna risposta in caso di esito positivo, quindi è possibile eseguire un'operazione di RECUPERO per la verifica della configurazione.

```
aws s3api get-object-lock-configuration --bucket mybucket --endpoint-url
https://s3.company.com
{
    "ObjectLockConfiguration": {
        "ObjectLockEnabled": "Enabled",
        "Rule": {
            "DefaultRetention": {
                "Mode": "COMPLIANCE",
                "Days": 1
            }
        }
    }
}
```

Successivamente, è possibile inserire un oggetto nel bucket con la configurazione di conservazione applicata.

```
aws s3 cp myfile.txt s3://mybucket --endpoint-url https://s3.company.com
```

L'operazione PUT restituisce una risposta.

```
upload: ./myfile.txt to s3://mybucket/myfile.txt
```

Nell'oggetto Retention, la durata di conservazione impostata nel bucket nell'esempio precedente viene convertita in un timestamp di conservazione sull'oggetto.

```
aws s3api get-object-retention --bucket mybucket --key myfile.txt
--endpoint-url https://s3.company.com
{
    "Retention": {
        "Mode": "COMPLIANCE",
        "RetainUntilDate": "2022-03-02T15:22:47.202000+00:00"
    }
}
```

Difesa da ransomware tramite bucket replicati con versione

Scopri come replicare gli oggetti in un bucket secondario utilizzando StorageGRID CloudMirror.

Non tutte le applicazioni e i carichi di lavoro saranno compatibili con il blocco degli oggetti. Un'altra opzione è replicare gli oggetti in un bucket secondario nella stessa griglia (preferibilmente un tenant diverso con accesso limitato) o in qualsiasi altro endpoint S3 con il servizio della piattaforma StorageGRID, CloudMirror.

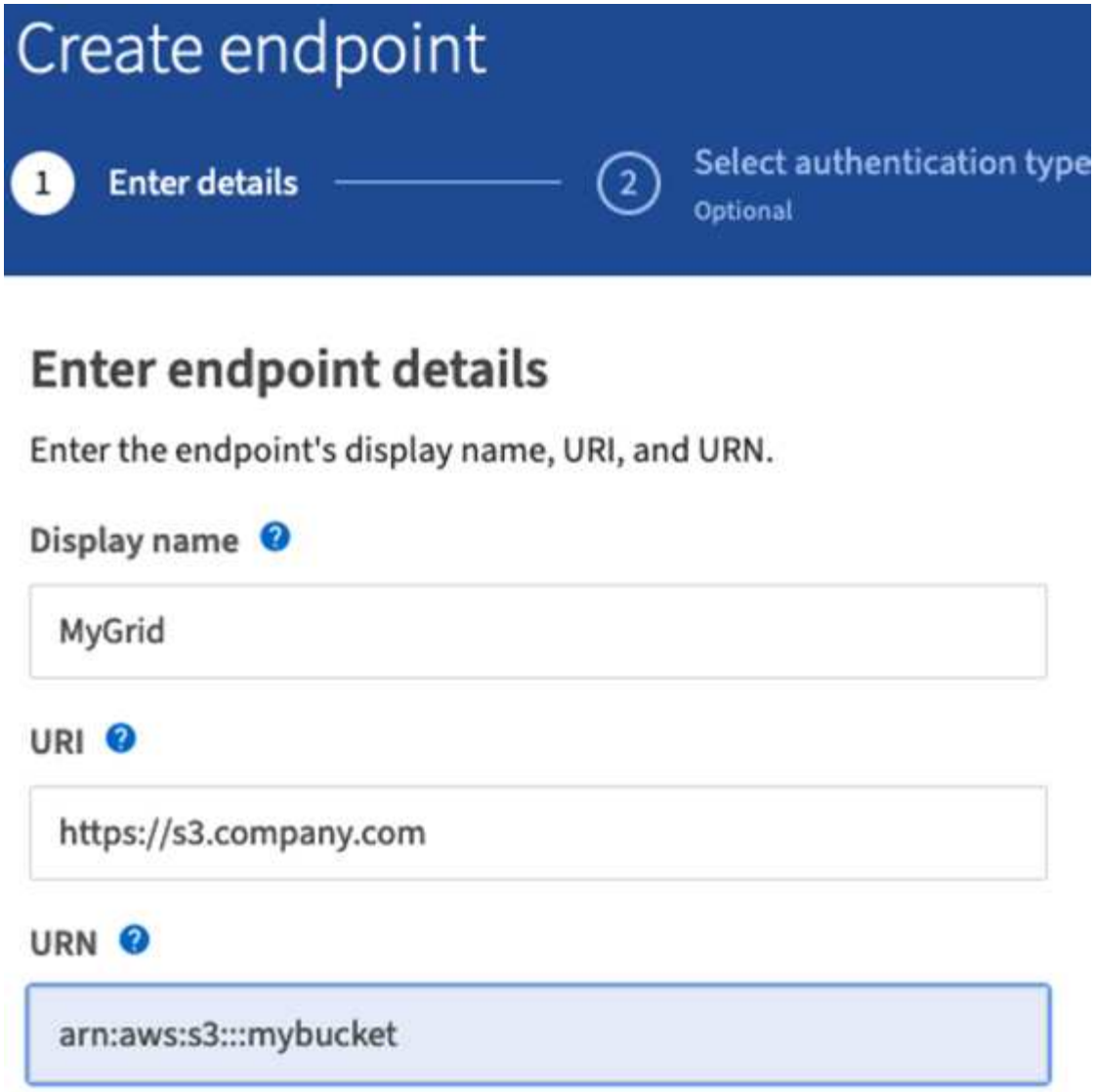
StorageGRID CloudMirror è un componente di StorageGRID che può essere configurato per replicare gli oggetti di un bucket in una destinazione definita quando vengono acquisiti nel bucket di origine e non replicano le eliminazioni. Poiché CloudMirror è un componente integrato di StorageGRID, non può essere disattivato o manipolato da un attacco basato su API S3. È possibile configurare questo bucket replicato con la versione abilitata. In questo scenario, è necessario eseguire una pulizia automatica delle vecchie versioni del bucket replicato, che possono essere eliminate in modo sicuro. A tale scopo, è possibile utilizzare il motore dei criteri ILM di StorageGRID. Creare regole per gestire il posizionamento degli oggetti in base al tempo non corrente per diversi giorni sufficienti ad identificarli e recuperarli da un attacco.

Un aspetto negativo di questo approccio è il fatto che consuma più storage disponendo di una seconda copia completa del bucket e di più versioni degli oggetti conservati per un po' di tempo. Inoltre, gli oggetti intenzionalmente eliminati dal bucket primario devono essere rimossi manualmente dal bucket replicato. Esistono altre opzioni di replica esterne al prodotto, come NetApp CloudSync, che possono replicare le eliminazioni per una soluzione simile. Un altro aspetto negativo per il bucket secondario che è abilitato per il controllo delle versioni e non per il blocco degli oggetti è la presenza di una serie di account privilegiati che potrebbero essere utilizzati per causare danni alla posizione secondaria. Il vantaggio è che dovrebbe essere un account univoco per quel bucket di endpoint o tenant e la compromissione probabilmente non include l'accesso agli account nella sede principale o viceversa.

Una volta creati i bucket di origine e destinazione e configurata la destinazione con la versione, è possibile configurare e abilitare la replica, come segue:

Fasi

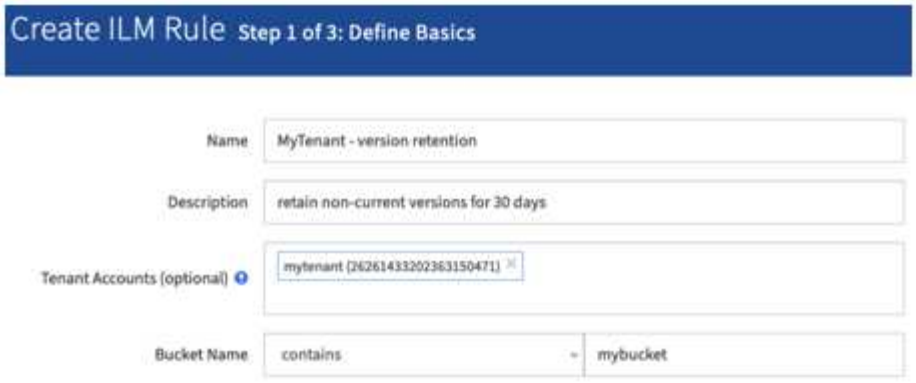
1. Per configurare CloudMirror, creare un endpoint dei servizi di piattaforma per la destinazione S3.

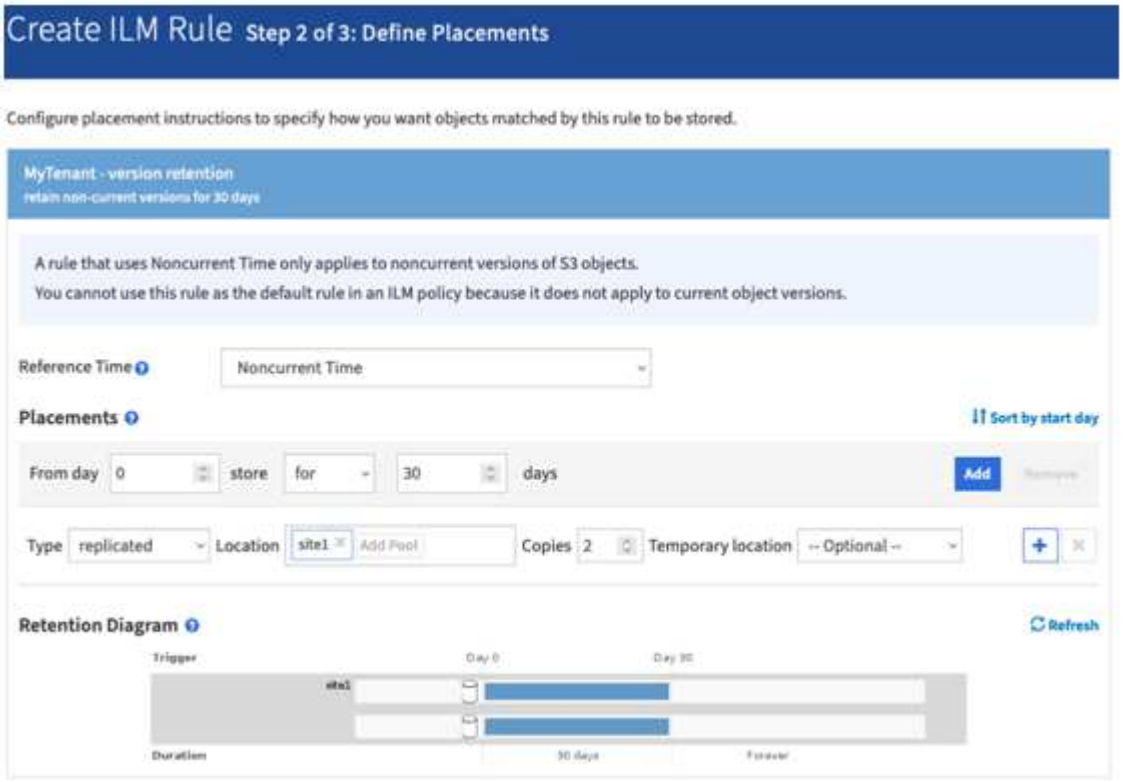


2. Nel bucket di origine, configurare la replica per utilizzare l'endpoint configurato.

```
<ReplicationConfiguration>
    <Role></Role>
    <Rule>
        <Status>Enabled</Status>
        <Prefix></Prefix>
        <Destination>
            <Bucket>arn:aws:s3:::mybucket</Bucket>
            <StorageClass>STANDARD</StorageClass>
        </Destination>
    </Rule>
</ReplicationConfiguration>
```

3. Creare regole ILM per gestire il posizionamento dello storage e la gestione della durata dello storage della versione. In questo esempio, vengono configurate le versioni non correnti degli oggetti da memorizzare.





Ci sono due copie nel sito 1 per 30 giorni. È inoltre possibile configurare le regole per la versione corrente degli oggetti in base all'utilizzo del tempo di acquisizione come tempo di riferimento nella regola ILM in modo che corrispondano alla durata di archiviazione del bucket di origine. Il posizionamento dello storage per le versioni a oggetti può essere sottoposto a erasure coding o replicato.

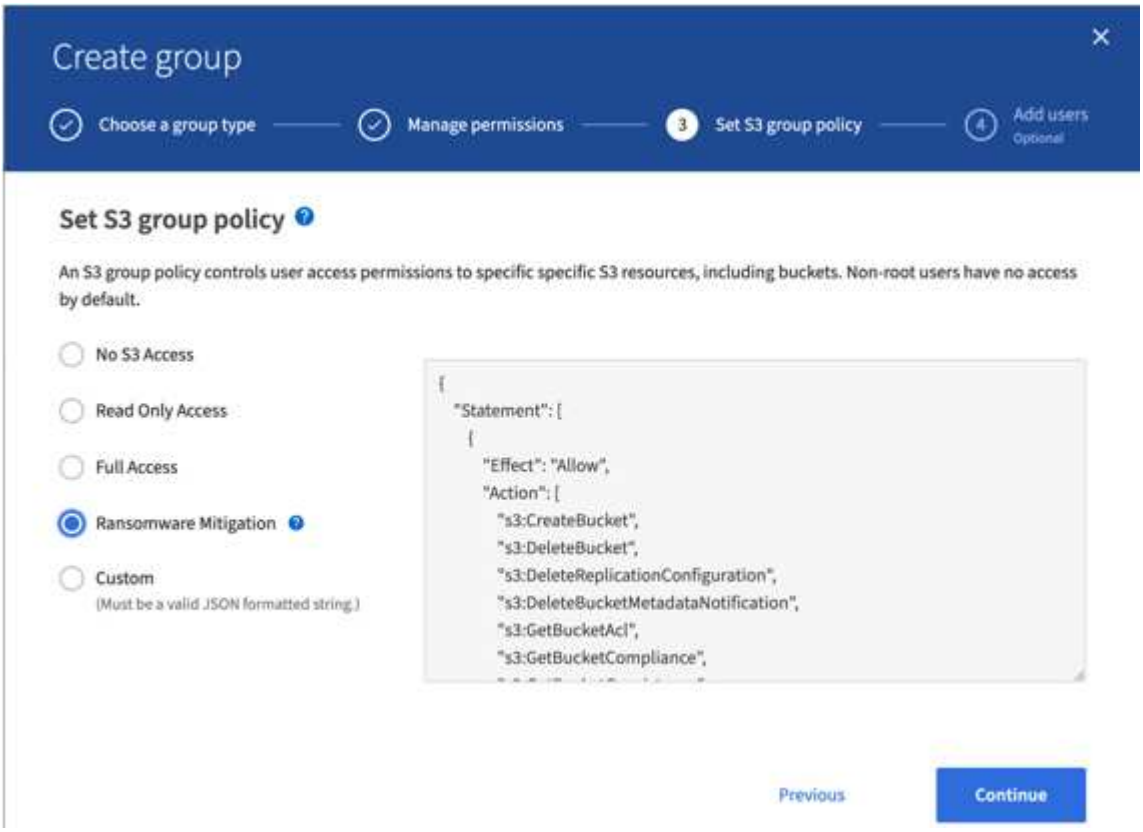
Difesa dal ransomware tramite versione con policy IAM di protezione

Scopri come proteggere i tuoi dati abilitando il controllo delle versioni nel bucket e implementando criteri IAM nei gruppi di sicurezza degli utenti in StorageGRID.

Un metodo per proteggere i dati senza utilizzare il blocco degli oggetti o la replica consiste nell'abilitare la versione nel bucket e implementare le policy IAM sui gruppi di sicurezza degli utenti per limitare la capacità degli utenti di gestire versioni degli oggetti. In caso di attacco, vengono create nuove versioni errate dei dati

come la versione corrente e la versione non corrente più recente è quella sicura. Gli account compromessi per ottenere l'accesso ai dati non hanno accesso per eliminare o modificare in altro modo la versione non corrente proteggendoli per operazioni di ripristino successive. Proprio come lo scenario precedente, le regole ILM gestiscono la conservazione delle versioni non correnti con una durata a scelta. L'aspetto negativo è che esiste ancora la possibilità di disporre di account privilegiati per un attacco di un attore non valido, ma tutti gli account del servizio applicazioni e gli utenti devono essere configurati con un accesso più restrittivo. I criteri di gruppo restrittivi devono consentire esplicitamente a ogni azione di cui si desidera che gli utenti o l'applicazione siano in grado di eseguire e negare esplicitamente le azioni di cui non si desidera che siano in grado di eseguire tali azioni. NetApp sconsiglia l'utilizzo di un'opzione con caratteri jolly, poiché in futuro potrebbe essere introdotta una nuova azione e si desidera controllare se è consentita o negata. Per questa soluzione, l'elenco di negazione deve includere DeleteObjectVersion, PutBucketPolicy, DeleteBucketPolicy, PutLifecycleConfiguration e PutBucketVersioning per proteggere la configurazione delle versioni del bucket e delle versioni dell'oggetto da modifiche dell'utente o del programma.

In StorageGRID 11,7 è stata introdotta una nuova opzione di policy di gruppo S3 "mitigazione del ransomware" per rendere più semplice l'implementazione di questa soluzione. Quando si crea un gruppo di utenti nel tenant, dopo aver selezionato le autorizzazioni del gruppo, è possibile visualizzare questo nuovo criterio opzionale.



Di seguito viene riportato il contenuto dei criteri di gruppo che includono la maggior parte delle operazioni disponibili esplicitamente consentite e il minimo richiesto negato.

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:CreateBucket",
                "s3:DeleteBucket",
```

```
                "s3:DeleteReplicationConfiguration",
"s3:DeleteBucketMetadataNotification",
                "s3:GetBucketAcl",
                "s3:GetBucketCompliance",
                "s3:GetBucketConsistency",
                "s3:GetBucketLastAccessTime",
                "s3:GetBucketLocation",
                "s3:GetBucketNotification"
"s3:GetBucketObjectLockConfiguration",
                "s3:GetBucketPolicy",
                "s3:GetBucketMetadataNotification",
                "s3:GetReplicationConfiguration",
                "s3:GetBucketCORS",
                "s3:GetBucketVersioning",
                "s3:GetBucketTagging",
                "s3:GetEncryptionConfiguration",
                "s3:GetLifecycleConfiguration",
                "s3:ListBucket",
                "s3:ListBucketVersions",
                "s3:ListAllMyBuckets",
                "s3:ListBucketMultipartUploads",
                "s3:PutBucketConsistency",
                "s3:PutBucketLastAccessTime",
                "s3:PutBucketNotification",
"s3:PutBucketObjectLockConfiguration",
                "s3:PutReplicationConfiguration",
                "s3:PutBucketCORS",
                "s3:PutBucketMetadataNotification",
                "s3:PutBucketTagging",
                "s3:PutEncryptionConfiguration",
                "s3:AbortMultipartUpload",
                "s3:DeleteObject",
                "s3:DeleteObjectTagging",
                "s3:DeleteObjectVersionTagging",
                "s3:GetObject",
                "s3:GetObjectAcl",
                "s3:GetObjectLegalHold",
                "s3:GetObjectRetention",
                "s3:GetObjectTagging",
                "s3:GetObjectVersion",
                "s3:GetObjectVersionAcl",
                "s3:GetObjectVersionTagging",
                "s3:ListMultipartUploadParts",
                "s3:PutObject",
                "s3:PutObjectAcl",
                "s3:PutObjectLegalHold",
```

```
                "s3:PutObjectRetention",
                "s3:PutObjectTagging",
                "s3:PutObjectVersionTagging",
                "s3:RestoreObject",
                "s3:ValidateObject",
                "s3:PutBucketCompliance",
                "s3:PutObjectVersionAcl"
            ],
            "Resource": "arn:aws:s3:::*"
        },
        {
            "Effect": "Deny",
            "Action": [
                "s3:DeleteObjectVersion",
                "s3:DeleteBucketPolicy",
                "s3:PutBucketPolicy",
                "s3:PutLifecycleConfiguration",
                "s3:PutBucketVersioning"
            ],
            "Resource": "arn:aws:s3:::*"
        }
    ]
}
```

Informazioni sul copyright

Informazioni sul marchio commerciale