Tr-4921 : défense contre les ransomware

How to enable StorageGRID in your environment

NetApp
July 05, 2024

Sommaire

Tr-4921 : défense contre les ransomware

Protégez les objets StorageGRID S3 contre les attaques par ransomware

Découvrez les attaques par ransomware et comment protéger vos données grâce aux bonnes pratiques de sécurité de StorageGRID.

Le nombre d'attaques par ransomware est en hausse Ce document fournit quelques recommandations sur la protection des données d'objet sur StorageGRID.

Les ransomware représentent aujourd'hui le danger omniprésent dans les data centers. Les ransomwares ont été conçus pour chiffrer les données et les rendre inutilisables par des utilisateurs et des applications qui en dépendent. La protection commence par les défenses habituelles : une mise en réseau renforcée et de solides pratiques de sécurité des utilisateurs. Nous devons ensuite appliquer les pratiques de sécurité de l'accès aux données.

Les ransomwares sont l'une des plus grandes menaces de sécurité. L'équipe NetApp StorageGRID travaille avec nos clients pour garder une longueur d'avance sur ces menaces. Le verrouillage d'objets et la gestion des versions vous permettent de vous protéger contre les modifications indésirables et de restaurer votre système suite à des attaques malveillantes. La sécurité des données est une entreprise multiniveaux, dans laquelle le stockage objet n'est qu'une partie de votre data Center.

Meilleures pratiques StorageGRID

Pour StorageGRID, les bonnes pratiques en matière de sécurité doivent inclure l'utilisation du protocole HTTPS avec des certificats signés pour la gestion et l'accès aux objets. Créez des comptes utilisateur dédiés aux applications et aux particuliers et n'utilisez pas les comptes root des locataires pour l'accès aux applications ou aux données utilisateur. En d'autres termes, suivez le principe du privilège minimum. Utilisez des groupes de sécurité avec des règles de gestion des identités et des accès (IAM) définies pour régir les droits d'utilisateur et les comptes d'accès spécifiques aux applications et aux utilisateurs. Une fois ces mesures mises en place, vous devez vous assurer que vos données sont protégées. Dans le cas de simple Storage Service (S3), lorsque les objets sont modifiés pour les chiffrer, il est remplacé par l'objet d'origine.

Méthodes de défense

Le mécanisme principal de protection contre les ransomwares dans l'API S3 consiste à mettre en œuvre le verrouillage objet. Toutes les applications ne sont pas compatibles avec le verrouillage d'objet. Il existe donc deux autres options pour protéger vos objets décrites dans ce rapport : la réplication vers un autre compartiment avec la gestion des versions activée et la gestion des versions avec les règles IAM.

Où trouver des informations complémentaires

Pour en savoir plus sur les informations données dans ce livre blanc, consultez ces documents et/ou sites web :

- Centre de documentation NetApp StorageGRID https://docs.netapp.com/us-en/storagegrid-118/
- Accompagnement NetApp StorageGRID https://docs.netapp.com/us-en/storagegrid-enable/
- Ressources de documentation StorageGRID https://www.netapp.com/data-storage/storagegrid/documentation/

- Documentation des produits NetApp https://www.netapp.com/support-and-training/documentation/

Protégez vos données contre les ransomwares à l'aide d'un verrouillage objet

Découvrez comment le verrouillage d'objets dans StorageGRID fournit un modèle WORM pour empêcher la suppression ou le remplacement des données, et comment il répond aux exigences réglementaires.

Le verrouillage des objets fournit un modèle WORM qui empêche la suppression ou l'écrasement d'objets. L'implémentation du verrouillage objet par StorageGRID "Cohasset évalué" permet de respecter les exigences réglementaires et prend en charge la conservation à des fins juridiques, le mode de conformité et le mode de gouvernance pour la conservation des objets ainsi que les règles de conservation des compartiments par défaut. Vous devez activer le verrouillage d'objet dans le cadre de la création de compartiment et de la gestion des versions. Une version spécifique d'un objet est verrouillée, et si aucun ID de version n'est défini, la rétention est placée sur la version actuelle de l'objet. Si la conservation de la version actuelle est configurée et qu'une tentative de suppression, de modification ou d'écrasement de l'objet est effectuée, une nouvelle version est créée avec un marqueur de suppression ou la nouvelle révision de l'objet comme version actuelle, et la version verrouillée est conservée comme une version non actuelle. Pour les applications qui ne sont pas encore compatibles, vous pouvez toujours utiliser le verrouillage objet et une configuration de conservation par défaut placée sur le compartiment. Une fois la configuration définie, une conservation d'objet est appliquée à chaque nouvel objet placé dans le compartiment. Cela fonctionne tant que l'application est configurée pour ne pas supprimer ou écraser les objets avant que la durée de conservation ne soit écoulée.

Voici quelques exemples d'utilisation de l'API de verrouillage d'objet :

La mise en attente légale du verrouillage d'objet est un état activé/désactivé simple appliqué à un objet.

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

La définition de l'état de mise en attente légale ne renvoie aucune valeur si elle a réussi, de sorte qu'elle peut être vérifiée à l'aide d'une opération GET.

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

Pour désactiver la mise en attente légale, appliquez le statut 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"
    }
}
```

La définition de la conservation d'objet s'effectue à l'aide d'un horodatage de conservation jusqu'à.

```
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
```

Encore une fois, il n'y a pas de valeur renvoyée en cas de réussite, vous pouvez donc vérifier l'état de conservation de la même manière avec un appel 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"
    }
```

Le fait de conserver une conservation par défaut dans un compartiment activé pour le verrouillage d'objet applique une période de conservation en jours et en années.

```
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
```

Comme pour la plupart de ces opérations, aucune réponse n'est renvoyée en cas de succès. Par conséquent, nous pouvons effectuer une OPÉRATION GET pour que la configuration puisse être vérifiée.

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

Vous pouvez ensuite placer un objet dans le compartiment avec la configuration de conservation appliquée.

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

L'opération PUT renvoie une réponse.

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

Sur l'objet de conservation, la durée de conservation définie dans le compartiment de l'exemple précédent est convertie en horodatage de conservation de l'objet.

```
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"
    }
}
```

Protection contre les ransomwares à l'aide d'un compartiment répliqué avec gestion des versions

Découvrez comment répliquer des objets vers un compartiment secondaire à l'aide de StorageGRID CloudMirror.

Les applications et les charges de travail ne seront pas toutes compatibles avec le verrouillage en mode objet. Une autre option consiste à répliquer les objets vers un compartiment secondaire dans la même grille (de

préférence un locataire différent avec accès limité) ou tout autre terminal S3 avec le service de plateforme StorageGRID, CloudMirror.

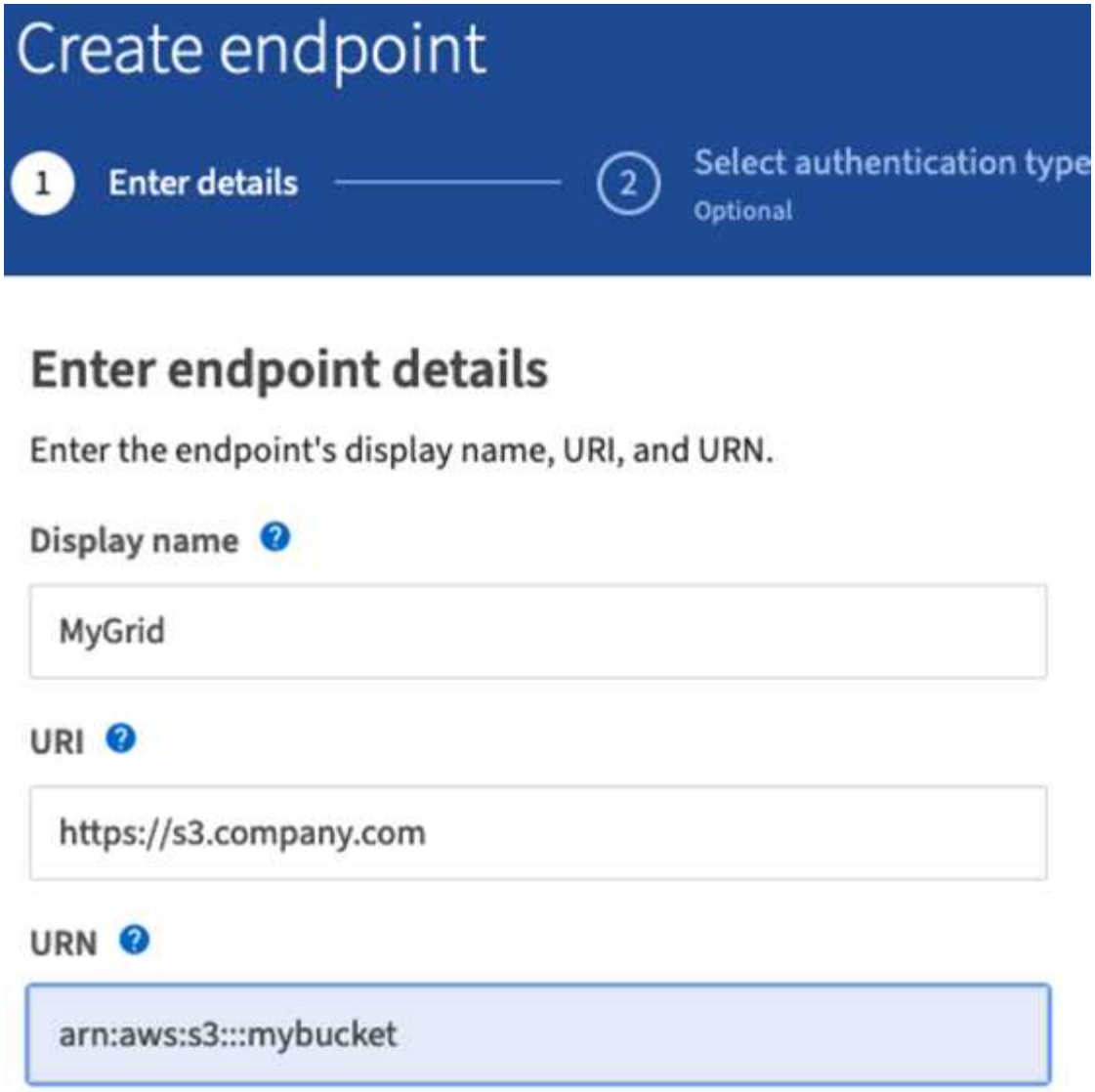
StorageGRID CloudMirror est un composant de StorageGRID qui peut être configuré pour répliquer les objets d'un compartiment vers une destination définie lors de leur ingestion dans le compartiment source et ne réplique pas les suppressions. Comme CloudMirror est un composant intégré de StorageGRID, il ne peut pas être désactivé ou manipulé par une attaque basée sur l'API S3. Vous pouvez configurer ce compartiment répliqué avec la gestion des versions activée. Dans ce scénario, vous avez besoin d'un nettoyage automatisé des anciennes versions du compartiment répliqué qui peuvent être jetées en toute sécurité. Pour cela, vous pouvez utiliser le moteur de règles ILM de StorageGRID. Créez des règles pour gérer le placement des objets en fonction d'une période non actuelle pendant plusieurs jours, suffisamment pour avoir identifié et récupéré une attaque.

L'un des inconvénients de cette approche est qu'elle consomme plus de stockage en conservant une seconde copie complète du compartiment et plusieurs versions des objets pendant un certain temps. En outre, les objets qui ont été supprimés intentionnellement du compartiment principal doivent être supprimés manuellement du compartiment répliqué. Il existe d'autres options de réplication en dehors du produit, telles que NetApp CloudSync, qui peuvent répliquer les suppressions pour une solution similaire. Un autre inconvénient est que la gestion des versions du compartiment secondaire est activée et que le verrouillage d'objet n'est pas activé, c'est qu'il existe un certain nombre de comptes privilégiés qui peuvent être utilisés pour causer des dommages à l'emplacement secondaire. L'avantage est qu'il doit s'agir d'un compte unique pour ce terminal ou ce compartiment locataire, et le compromis n'inclut probablement pas l'accès aux comptes sur l'emplacement principal, et inversement.

Une fois les compartiments source et destination créés et la destination configurée avec la gestion des versions, vous pouvez configurer et activer la réplication comme suit :

Étapes

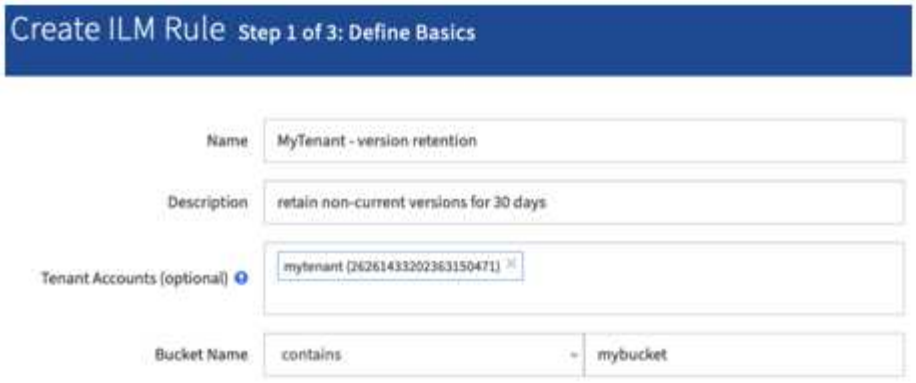
1. Pour configurer CloudMirror, créez un terminal de services de plateforme pour la destination S3.

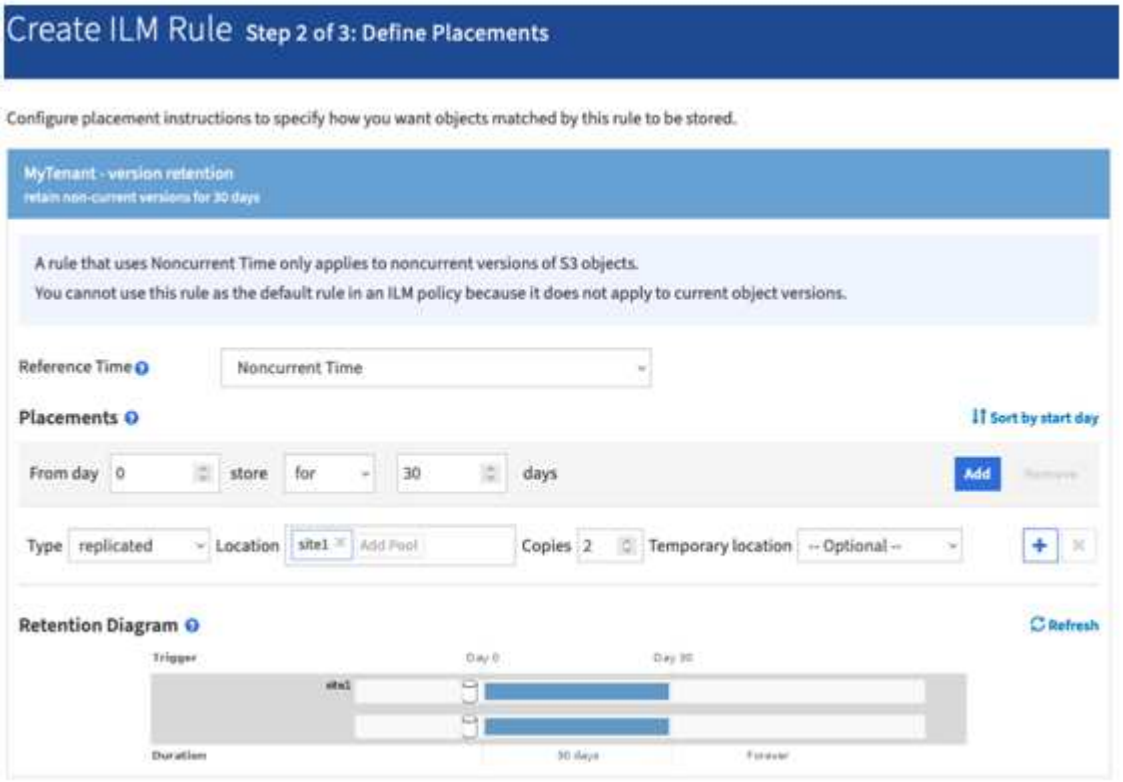


2. Sur le compartiment source, configurez la réplication pour utiliser le terminal configuré.

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

3. Créez des règles ILM pour gérer le placement du stockage et la gestion de la durée du stockage des versions. Dans cet exemple, les versions non actuelles des objets à stocker sont configurées.





Il y a deux copies sur le site 1 pendant 30 jours. Vous configurez également les règles de la version actuelle des objets en fonction de l'utilisation de l'heure d'ingestion comme heure de référence dans la règle ILM pour correspondre à la durée de stockage du compartiment source. Le placement de stockage des versions d'objets peut être codé par effacement ou répliqué.

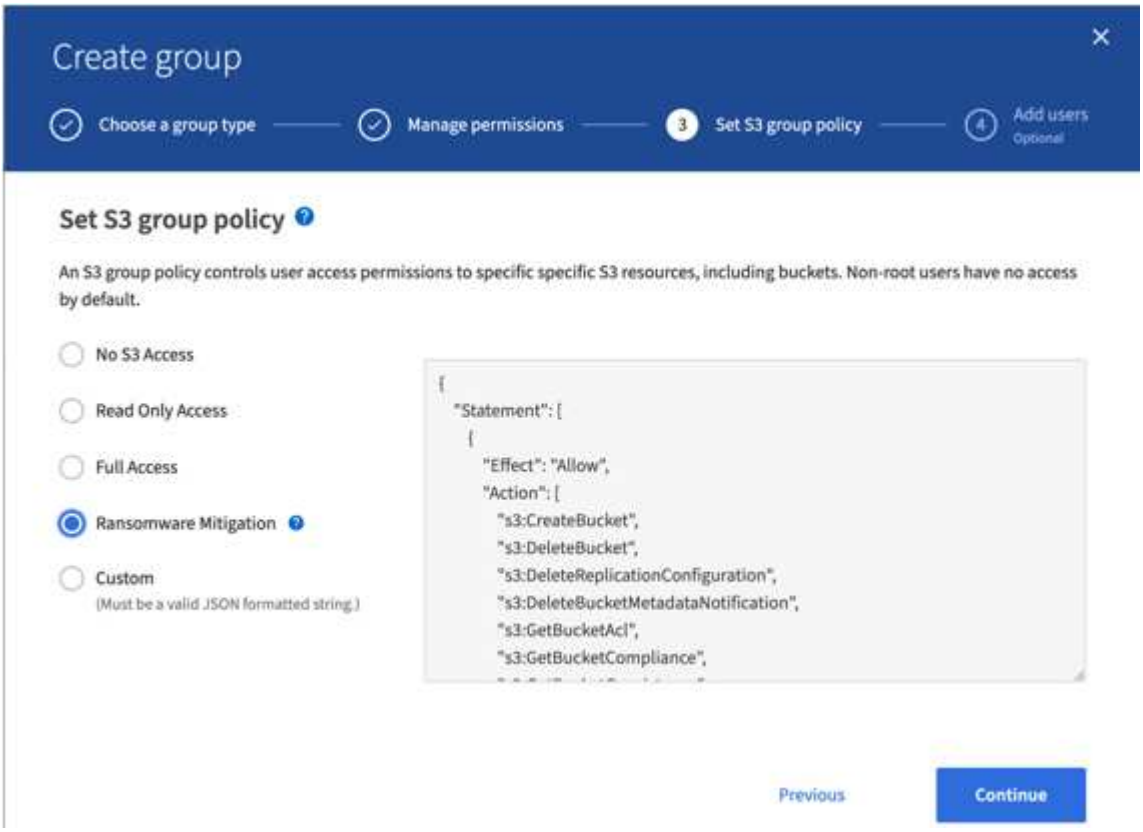
Défense anti-ransomware à l'aide du contrôle des versions avec une politique IAM de protection

Découvrez comment protéger vos données en activant la gestion des versions dans le compartiment et en implémentant les règles IAM sur les groupes de sécurité des utilisateurs dans StorageGRID.

Une méthode pour protéger vos données sans verrouillage objet ou réplication consiste à activer la gestion des versions sur le compartiment et à mettre en œuvre des règles IAM sur les groupes de sécurité utilisateur

afin de limiter la capacité des utilisateurs à gérer des versions des objets. En cas d'attaque, de nouvelles versions incorrectes des données sont créées en tant que version actuelle, et la version la plus récente non-actuelle est la sécurité des données. Les comptes compromis pour accéder aux données n'ont pas accès à supprimer ni à modifier la version non actuelle qui les protège pour des opérations de restauration ultérieures. Comme dans le scénario précédent, les règles ILM gèrent la conservation des versions non actuelles avec la durée de votre choix. L'inconvénient est qu'il existe toujours la possibilité de comptes privilégiés pour une attaque de mauvais acteurs, mais tous les comptes de service d'application et les utilisateurs doivent être configurés avec un accès plus restrictif. La stratégie de groupe restrictif doit explicitement autoriser chaque action que vous souhaitez que les utilisateurs ou l'application soient capables et refuser explicitement toute action dont vous ne voulez pas qu'ils soient capables. NetApp ne recommande pas l'utilisation d'une autorisation générique car une nouvelle action pourrait être introduite à l'avenir et vous voudrez contrôler si elle est autorisée ou refusée. Pour cette solution, la liste de refus doit inclure DeleteObjectVersion, PutBucketPolicy, DeleteBuckePolicy, PutLifecycleConfiguration et PutBucketVersioning afin de protéger la configuration de gestion des versions du compartiment et de l'objet des modifications utilisateur ou programmatiques.

Dans StorageGRID 11.7, une nouvelle option de stratégie de groupe S3 « réduction des ransomware » a été introduite pour faciliter l'implémentation de cette solution. Lorsque vous créez un groupe d'utilisateurs dans le tenant, après avoir sélectionné les autorisations de groupe, vous pouvez voir cette nouvelle stratégie facultative.



Voici le contenu de la stratégie de groupe qui inclut la plupart des opérations disponibles explicitement autorisées et le minimum requis refusé.

```
{
    "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:::*"
        }
    ]
}
```

Informations sur le copyright

Informations sur les marques commerciales