Format du fichier journal d'audit

StorageGRID software

NetApp
December 03, 2025

Sommaire

Format du fichier journal d’audit

Format du fichier journal d’audit

Format du fichier journal d’audit

Les fichiers journaux d’audit se trouvent sur chaque nœud d’administration et contiennent une collection de messages d’audit individuels.

Chaque message d’audit contient les éléments suivants :

- Le temps universel coordonné (UTC) de l’événement qui a déclenché le message d’audit (ATIM) au format ISO 8601, suivi d’un espace :

  YYYY-MM-DDTHH:MM:SS.UUUUUU, où *UUUUUU* sont des microsecondes.

- Le message d’audit lui-même, placé entre crochets et commençant par `AUDT` .

L’exemple suivant montre trois messages d’audit dans un fichier journal d’audit (sauts de ligne ajoutés pour plus de lisibilité). Ces messages ont été générés lorsqu’un locataire a créé un bucket S3 et ajouté deux objets à ce bucket.

```
2019-08-07T18:43:30.247711
[AUDT:[RSLT(FC32):SUCS][CNID(UI64):1565149504991681][TIME(UI64):73520][SAI
P(IPAD):"10.224.2.255"][S3AI(CSTR):"17530064241597054718"]
[SACC(CSTR):"s3tenant"][S3AK(CSTR):"SGKH9100SCkNB8M3MTWNt-
PhoTDwB9JOk7PtyLkQmA=="][SUSR(CSTR):"urn:sgws:identity::175300642415970547
18:root"]
[SBAI(CSTR):"17530064241597054718"][SBAC(CSTR):"s3tenant"][S3BK(CSTR):"buc
ket1"][AVER(UI32):10][ATIM(UI64):1565203410247711]
[ATYP(FC32):SPUT][ANID(UI32):12454421][AMID(FC32):S3RQ][ATID(UI64):7074142
142472611085]]

2019-08-07T18:43:30.783597
[AUDT:[RSLT(FC32):SUCS][CNID(UI64):1565149504991696][TIME(UI64):120713][SA
IP(IPAD):"10.224.2.255"][S3AI(CSTR):"17530064241597054718"]
[SACC(CSTR):"s3tenant"][S3AK(CSTR):"SGKH9100SCkNB8M3MTWNt-
PhoTDwB9JOk7PtyLkQmA=="][SUSR(CSTR):"urn:sgws:identity::175300642415970547
18:root"]
[SBAI(CSTR):"17530064241597054718"][SBAC(CSTR):"s3tenant"][S3BK(CSTR):"buc
ket1"][S3KY(CSTR):"fh-small-0"]
[CBID(UI64):0x779557A069B2C037][UUID(CSTR):"94BA6949-38E1-4B0C-BC80-
EB44FB4FCC7F"][CSIZ(UI64):1024][AVER(UI32):10]
[ATIM(UI64):1565203410783597][ATYP(FC32):SPUT][ANID(UI32):12454421][AMID(F
C32):S3RQ][ATID(UI64):8439606722108456022]]

2019-08-07T18:43:30.784558
[AUDT:[RSLT(FC32):SUCS][CNID(UI64):1565149504991693][TIME(UI64):121666][SA
IP(IPAD):"10.224.2.255"][S3AI(CSTR):"17530064241597054718"]
[SACC(CSTR):"s3tenant"][S3AK(CSTR):"SGKH9100SCkNB8M3MTWNt-
PhoTDwB9JOk7PtyLkQmA=="][SUSR(CSTR):"urn:sgws:identity::175300642415970547
18:root"]
[SBAI(CSTR):"17530064241597054718"][SBAC(CSTR):"s3tenant"][S3BK(CSTR):"buc
ket1"][S3KY(CSTR):"fh-small-2000"]
[CBID(UI64):0x180CBD8E678EED17][UUID(CSTR):"19CE06D0-D2CF-4B03-9C38-
E578D66F7ADD"][CSIZ(UI64):1024][AVER(UI32):10]
[ATIM(UI64):1565203410784558][ATYP(FC32):SPUT][ANID(UI32):12454421][AMID(F
C32):S3RQ][ATID(UI64):13489590586043706682]]
```

Dans leur format par défaut, les messages d'audit dans les fichiers journaux d'audit ne sont pas faciles à lire ou à interpréter. Vous pouvez utiliser le"outil d'audit-explication" pour obtenir des résumés simplifiés des messages d'audit dans le journal d'audit. Vous pouvez utiliser le"outil de somme d'audit" pour résumer combien d'opérations d'écriture, de lecture et de suppression ont été enregistrées et combien de temps ces opérations ont pris.

Utiliser l'outil d'audit-explication

Vous pouvez utiliser le `audit-explain` outil permettant de traduire les messages d'audit du journal d'audit dans un format facile à lire.

Avant de commencer

- Tu as"autorisations d'accès spécifiques" .
- Vous devez avoir le `Passwords.txt` déposer.
- Vous devez connaître l'adresse IP du nœud d'administration principal.

À propos de cette tâche

Le `audit-explain` L'outil, disponible sur le nœud d'administration principal, fournit des résumés simplifiés des messages d'audit dans un journal d'audit.

> Le `audit-explain` L'outil est principalement destiné à être utilisé par le support technique lors des opérations de dépannage. Traitement `audit-explain` les requêtes peuvent consommer une grande quantité de puissance CPU, ce qui peut avoir un impact sur les opérations StorageGRID .

Cet exemple montre une sortie typique de la `audit-explain` outil. Ces quatre"CRACHER" des messages d'audit ont été générés lorsque le locataire S3 avec l'ID de compte 92484777680322627870 a utilisé des requêtes S3 PUT pour créer un bucket nommé « bucket1 » et ajouter trois objets à ce bucket.

```
SPUT S3 PUT bucket bucket1 account:92484777680322627870 usec:124673
SPUT S3 PUT object bucket1/part1.txt tenant:92484777680322627870
cbid:9DCB157394F99FE5 usec:101485
SPUT S3 PUT object bucket1/part2.txt tenant:92484777680322627870
cbid:3CFBB07AB3D32CA9 usec:102804
SPUT S3 PUT object bucket1/part3.txt tenant:92484777680322627870
cbid:5373D73831ECC743 usec:93874
```

Le `audit-explain` L'outil peut effectuer les opérations suivantes :

- Traiter les journaux d'audit simples ou compressés. Par exemple:

  ```
  audit-explain audit.log

  audit-explain 2019-08-12.txt.gz
  ```

- Traiter plusieurs fichiers simultanément. Par exemple:

  ```
  audit-explain audit.log 2019-08-12.txt.gz 2019-08-13.txt.gz

  audit-explain /var/local/log/*
  ```

- Acceptez l'entrée d'un tube, ce qui vous permet de filtrer et de prétraiter l'entrée à l'aide du `grep` commandement ou autres moyens. Par exemple:

```
grep SPUT audit.log | audit-explain

grep bucket-name audit.log | audit-explain
```

Étant donné que les journaux d'audit peuvent être très volumineux et lents à analyser, vous pouvez gagner du temps en filtrant les parties que vous souhaitez consulter et en les exécutant. `audit-explain` sur les parties, au lieu du fichier entier.

> Le `audit-explain` l'outil n'accepte pas les fichiers compressés comme entrée canalisée. Pour traiter les fichiers compressés, fournissez leurs noms de fichiers comme arguments de ligne de commande ou utilisez le `zcat` outil pour décompresser les fichiers en premier. Par exemple:
>
> ```
> zcat audit.log.gz | audit-explain
> ```

Utilisez le `help (-h)` option pour voir les options disponibles. Par exemple:

```
$ audit-explain -h
```

Étapes

1. Connectez-vous au nœud d'administration principal :
   a. Entrez la commande suivante : `ssh admin@primary_Admin_Node_IP`
   b. Entrez le mot de passe indiqué dans le `Passwords.txt` déposer.
   c. Entrez la commande suivante pour passer en root : `su -`
   d. Entrez le mot de passe indiqué dans le `Passwords.txt` déposer.

      Lorsque vous êtes connecté en tant que root, l'invite passe de `$` à `#` .
2. Entrez la commande suivante, où `/var/local/log/audit.log` représente le nom et l'emplacement du ou des fichiers que vous souhaitez analyser :

   ```
   $ audit-explain /var/local/log/audit.log
   ```

   Le `audit-explain` L'outil imprime des interprétations lisibles par l'homme de tous les messages dans le ou les fichiers spécifiés.

   > Pour réduire la longueur des lignes et améliorer la lisibilité, les horodatages ne sont pas affichés par défaut. Si vous souhaitez voir les horodatages, utilisez l'horodatage(`-t`) option.

Utiliser l'outil d'audit-somme

Vous pouvez utiliser le `audit-sum` outil permettant de compter les messages d'audit d'écriture, de lecture, d'en-tête et de suppression et de voir le temps minimum, maximum et moyen (ou la taille) pour chaque type d'opération.

Avant de commencer

- Tu as"autorisations d'accès spécifiques" .

- Vous devez avoir le `Passwords.txt` déposer.
- Vous devez connaître l'adresse IP du nœud d'administration principal.

À propos de cette tâche

Le `audit-sum` L'outil, disponible sur le nœud d'administration principal, résume le nombre d'opérations d'écriture, de lecture et de suppression enregistrées et la durée de ces opérations.

> Le `audit-sum` L'outil est principalement destiné à être utilisé par le support technique lors des opérations de dépannage. Traitement `audit-sum` les requêtes peuvent consommer une grande quantité de puissance CPU, ce qui peut avoir un impact sur les opérations StorageGRID .

Cet exemple montre une sortie typique de la `audit-sum` outil. Cet exemple montre combien de temps ont pris les opérations du protocole.

```
  message group           count     min(sec)        max(sec)
average(sec)
  =============           =====     ========        ========
============
  IDEL                      274
  SDEL                   213371        0.004          20.934
0.352
  SGET                   201906        0.010        1740.290
1.132
  SHEA                    22716        0.005           2.349
0.272
  SPUT                 1771398        0.011        1770.563
0.487
```

Le `audit-sum` L'outil fournit les décomptes et les heures des messages d'audit S3, Swift et ILM suivants dans un journal d'audit.

> Les codes d'audit sont supprimés du produit et de la documentation à mesure que les fonctionnalités sont obsolètes. Si vous rencontrez un code d'audit qui n'est pas répertorié ici, vérifiez les versions précédentes de cette rubrique pour les anciennes versions de SG. Par exemple : "Documentation de l'outil d'audit de StorageGRID 11.8" .

Code	Description	Se référer à
IDEL	Suppression initiée par ILM : enregistre le moment où ILM démarre le processus de suppression d'un objet.	"IDEL : suppression initiée par ILM"
SDEL	S3 DELETE : enregistre une transaction réussie pour supprimer un objet ou un bucket.	"SDEL : SUPPRESSION S3"
SGET	S3 GET : enregistre une transaction réussie pour récupérer un objet ou répertorier les objets dans un bucket.	"SGET : S3 OBTENIR"

Code	Description	Se référer à
KARITÉ	S3 HEAD : enregistre une transaction réussie pour vérifier l'existence d'un objet ou d'un bucket.	"SHEA : TÊTE S3"
CRACHER	S3 PUT : enregistre une transaction réussie pour créer un nouvel objet ou un nouveau bucket.	"SPUT : S3 PUT"
WDEL	Swift DELETE : enregistre une transaction réussie pour supprimer un objet ou un conteneur.	"WDEL : SUPPRESSION rapide"
WGET	Swift GET : enregistre une transaction réussie pour récupérer un objet ou répertorier les objets dans un conteneur.	"WGET : Swift GET"
WHEA	Swift HEAD : enregistre une transaction réussie pour vérifier l'existence d'un objet ou d'un conteneur.	"WHEA : TÊTE RAPIDE"
WPUT	Swift PUT : enregistre une transaction réussie pour créer un nouvel objet ou conteneur.	"WPUT : Swift PUT"

Le `audit-sum` L'outil peut effectuer les opérations suivantes :

- Traiter les journaux d'audit simples ou compressés. Par exemple:

```
audit-sum audit.log

audit-sum 2019-08-12.txt.gz
```

- Traiter plusieurs fichiers simultanément. Par exemple:

```
audit-sum audit.log 2019-08-12.txt.gz 2019-08-13.txt.gz

audit-sum /var/local/log/*
```

- Acceptez l'entrée d'un tube, ce qui vous permet de filtrer et de prétraiter l'entrée à l'aide du `grep` commandement ou autres moyens. Par exemple:

```
grep WGET audit.log | audit-sum

grep bucket1 audit.log | audit-sum

grep SPUT audit.log | grep bucket1 | audit-sum
```

Cet outil n'accepte pas les fichiers compressés comme entrée canalisée. Pour traiter les fichiers compressés, fournissez leurs noms de fichiers comme arguments de ligne de commande ou utilisez le `zcat` outil pour décompresser les fichiers en premier. Par exemple:

```
audit-sum audit.log.gz

zcat audit.log.gz | audit-sum
```

Vous pouvez utiliser des options de ligne de commande pour résumer les opérations sur les buckets séparément des opérations sur les objets ou pour regrouper les résumés de messages par nom de bucket, par période ou par type de cible. Par défaut, les résumés affichent le temps de fonctionnement minimum, maximum et moyen, mais vous pouvez utiliser le `size (-s)` option permettant de regarder la taille de l'objet à la place.

Utilisez le `help (-h)` option pour voir les options disponibles. Par exemple:

```
$ audit-sum -h
```

Étapes

1. Connectez-vous au nœud d'administration principal :
   a. Entrez la commande suivante : `ssh admin@`*`primary_Admin_Node_IP`*
   b. Entrez le mot de passe indiqué dans le `Passwords.txt` déposer.
   c. Entrez la commande suivante pour passer en root : `su -`
   d. Entrez le mot de passe indiqué dans le `Passwords.txt` déposer.

      Lorsque vous êtes connecté en tant que root, l'invite passe de `$` à `#` .
2. Si vous souhaitez analyser tous les messages liés aux opérations d'écriture, de lecture, de lecture et de suppression, suivez ces étapes :
   a. Entrez la commande suivante, où `/var/local/log/audit.log` représente le nom et l'emplacement du ou des fichiers que vous souhaitez analyser :

      ```
      $ audit-sum /var/local/log/audit.log
      ```

      Cet exemple montre une sortie typique de la `audit-sum` outil. Cet exemple montre combien de temps ont pris les opérations du protocole.

```
  message group           count     min(sec)        max(sec)
average(sec)
  =============           =====     ========        ========
============
  IDEL                      274
  SDEL                   213371        0.004          20.934
0.352
  SGET                   201906        0.010        1740.290
1.132
  SHEA                    22716        0.005           2.349
0.272
  SPUT                 1771398        0.011        1770.563
0.487
```

Dans cet exemple, les opérations SGET (S3 GET) sont les plus lentes en moyenne avec 1,13 seconde, mais les opérations SGET et SPUT (S3 PUT) affichent toutes deux des temps de pire cas longs d'environ 1 770 secondes.

b. Pour afficher les 10 opérations de récupération les plus lentes, utilisez la commande grep pour sélectionner uniquement les messages SGET et ajoutez l'option de sortie longue(`-l`) pour inclure les chemins d'objet :

   ```
   grep SGET audit.log | audit-sum -l
   ```

   Les résultats incluent le type (objet ou bucket) et le chemin, ce qui vous permet de rechercher dans le journal d'audit d'autres messages relatifs à ces objets particuliers.

```
Total:          201906 operations
    Slowest:      1740.290 sec
    Average:         1.132 sec
    Fastest:         0.010 sec
    Slowest operations:
        time(usec)       source ip         type      size(B) path
        ========== ============== ============ ============ ====
        1740289662   10.96.101.125       object   5663711385
backup/r9O1OaQ8JB-1566861764-4519.iso
        1624414429   10.96.101.125       object   5375001556
backup/r9O1OaQ8JB-1566861764-6618.iso
        1533143793   10.96.101.125       object   5183661466
backup/r9O1OaQ8JB-1566861764-4518.iso
             70839   10.96.101.125       object        28338
bucket3/dat.1566861764-6619
             68487   10.96.101.125       object        27890
bucket3/dat.1566861764-6615
             67798   10.96.101.125       object        27671
bucket5/dat.1566861764-6617
             67027   10.96.101.125       object        27230
bucket5/dat.1566861764-4517
             60922   10.96.101.125       object        26118
bucket3/dat.1566861764-4520
             35588   10.96.101.125       object        11311
bucket3/dat.1566861764-6616
             23897   10.96.101.125       object        10692
bucket3/dat.1566861764-4516
```

+ À partir de cet exemple de sortie, vous pouvez voir que les trois requêtes S3 GET les plus lentes concernaient des objets d'une taille d'environ 5 Go, ce qui est beaucoup plus grand que les autres objets. La grande taille explique les temps de récupération les plus lents dans le pire des cas.

3. Si vous souhaitez déterminer les tailles des objets ingérés et récupérés dans votre grille, utilisez l'option de taille(`-s`):

   ```
   audit-sum -s audit.log
   ```

```
  message group           count       min(MB)          max(MB)
average(MB)
  =============           =====     ========        ========
============
  IDEL                      274        0.004        5000.000
1654.502
  SDEL                   213371        0.000          10.504
1.695
  SGET                   201906        0.000        5000.000
14.920
  SHEA                    22716        0.001          10.504
2.967
  SPUT                 1771398        0.000        5000.000
2.495
```

Dans cet exemple, la taille moyenne de l'objet pour SPUT est inférieure à 2,5 Mo, mais la taille moyenne pour SGET est beaucoup plus grande. Le nombre de messages SPUT est bien supérieur au nombre de messages SGET, ce qui indique que la plupart des objets ne sont jamais récupérés.

4. Si vous souhaitez déterminer si les récupérations ont été lentes hier :

   a. Émettez la commande sur le journal d'audit approprié et utilisez l'option de regroupement par heure(`-gt`), suivi de la période (par exemple, 15M, 1H, 10S) :

      ```
      grep SGET audit.log | audit-sum -gt 1H
      ```

```
  message group           count    min(sec)       max(sec)
average(sec)
  =============           =====     ========        ========
============
  2019-09-05T00            7591        0.010        1481.867
1.254
  2019-09-05T01            4173        0.011        1740.290
1.115
  2019-09-05T02           20142        0.011        1274.961
1.562
  2019-09-05T03           57591        0.010        1383.867
1.254
  2019-09-05T04          124171        0.013        1740.290
1.405
  2019-09-05T05          420182        0.021        1274.511
1.562
  2019-09-05T06         1220371        0.015        6274.961
5.562
  2019-09-05T07          527142        0.011        1974.228
2.002
  2019-09-05T08          384173        0.012        1740.290
1.105
  2019-09-05T09           27591        0.010        1481.867
1.354
```

Ces résultats montrent que le trafic S3 GET a augmenté entre 06h00 et 07h00. Les temps maximum et moyen sont également considérablement plus élevés à ces moments-là, et ils n'augmentent pas progressivement à mesure que le nombre augmente. Cela suggère que la capacité a été dépassée quelque part, peut-être dans le réseau ou dans la capacité du réseau à traiter les demandes.

b. Pour déterminer la taille des objets récupérés chaque heure hier, ajoutez l'option de taille(`-s`) à la commande :

```
grep SGET audit.log | audit-sum -gt 1H -s
```

```
  message group           count       min(B)          max(B)
average(B)
  =============           =====     ========        ========
============
  2019-09-05T00            7591        0.040        1481.867
1.976
  2019-09-05T01            4173        0.043        1740.290
2.062
  2019-09-05T02           20142        0.083        1274.961
2.303
  2019-09-05T03           57591        0.912        1383.867
1.182
  2019-09-05T04          124171        0.730        1740.290
1.528
  2019-09-05T05          420182        0.875        4274.511
2.398
  2019-09-05T06         1220371        0.691  5663711385.961
51.328
  2019-09-05T07          527142        0.130        1974.228
2.147
  2019-09-05T08          384173        0.625        1740.290
1.878
  2019-09-05T09           27591        0.689        1481.867
1.354
```

Ces résultats indiquent que certaines récupérations très importantes ont eu lieu lorsque le trafic de récupération global était à son maximum.

c. Pour voir plus de détails, utilisez le"outil d'audit-explication" pour passer en revue toutes les opérations SGET pendant cette heure :

```
grep 2019-09-05T06 audit.log | grep SGET | audit-explain | less
```

Si la sortie de la commande grep doit comporter plusieurs lignes, ajoutez le `less` commande pour afficher le contenu du fichier journal d'audit une page (un écran) à la fois.

5. Si vous souhaitez déterminer si les opérations SPUT sur les buckets sont plus lentes que les opérations SPUT pour les objets :

a. Commencez par utiliser le `-go` option, qui regroupe séparément les messages pour les opérations d'objet et de compartiment :

```
grep SPUT sample.log | audit-sum -go
```

```
  message group           count     min(sec)        max(sec)
average(sec)
  =============           =====     ========        ========
============
  SPUT.bucket                 1        0.125           0.125
0.125
  SPUT.object                12        0.025           1.019
0.236
```

Les résultats montrent que les opérations SPUT pour les buckets ont des caractéristiques de performances différentes des opérations SPUT pour les objets.

b. Pour déterminer quels buckets ont les opérations SPUT les plus lentes, utilisez le `-gb` option, qui regroupe les messages par bucket :

```
grep SPUT audit.log | audit-sum -gb
```

```
  message group                 count     min(sec)        max(sec)
average(sec)
  =============                 =====     ========        ========
============
  SPUT.cho-non-versioning       71943        0.046        1770.563
1.571
  SPUT.cho-versioning           54277        0.047        1736.633
1.415
  SPUT.cho-west-region          80615        0.040          55.557
1.329
  SPUT.ldt002                 1564563        0.011          51.569
0.361
```

c. Pour déterminer quels buckets ont la plus grande taille d'objet SPUT, utilisez à la fois le `-gb` et le `-s` options:

```
grep SPUT audit.log | audit-sum -gb -s
```

```
  message group                    count       min(B)          max(B)
average(B)
  =============                    =====     ========        ========
============
  SPUT.cho-non-versioning          71943        2.097        5000.000
21.672
  SPUT.cho-versioning              54277        2.097        5000.000
21.120
  SPUT.cho-west-region             80615        2.097         800.000
14.433
  SPUT.ldt002                    1564563        0.000         999.972
0.352
```