Table des matières

Rechercher, Filter, Formater 2
EXEMPLE 2
EXPLICATION 3
Format Date 3
apache2/error.log 3
apache2/acces.log 4
icecast2/access.log 5
Fail2ban.log 5

Rechercher, Filter, Formater

EXEMPLE

Je souhaite rechercher dans mes logs apache les adresse ip qui se connectent sur mon site Web.

- Les logs se situe dans **/var/log/apache2/access.log**
- Les commande dont j'ai besoin sont: **GREP**, **SED**, **CUT** et **TR**

commençons dans l'ordre:
La première commande pour rechercher:

```
grep -i "POST" /var/log/apache2/access.log
```

RETOURNE

```
27.50.160.35 - - [01/Mar/2020:10:55:12 +0100] "POST /tomcat.php HTTP/1.1"
404 495 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:48.0) Gecko/20100101
Firefox/48.0"
192.168.0.3 - - [01/Mar/2020:10:55:12 +0100] "POST /tomcat.php HTTP/1.1" 404
495 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:48.0) Gecko/20100101
Firefox/48.0"
```

Améliorons la avec **sed** qui supprimera l'information dont je n'est pas besoin.
Ici je souhaite enlever les ip de mon réseaux local. (192.168.xxx.xxx)
La commande sera donc:

```
grep -i "POST" /var/log/apache2/access.log | sed '/192.168./d'
```

RETOURNE

```
27.50.160.35 - - [01/Mar/2020:10:55:12 +0100] "POST /tomcat.php HTTP/1.1"
404 495 "-" "Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:48.0) Gecko/20100101
Firefox/48.0"
```

Maintenant je veux tronquer ce résultat que sur les 16 premier caractère afin d'obtenir que l'ip.

```
grep -i "POST" /var/log/apache2/access.log | sed '/192.168./d' | cut -c -16
```

RETOURNE

```
27.50.160.35 - - [
```

Le résultat est pas mal mais je ne veut pas de caractère de type (-) ou ([).

```
grep -i "POST" /var/log/apache2/access.log | sed '/192.168./d' | cut -c -16
| tr -d '-' | tr -d '[]'
```

ou

```
grep -i "POST" /var/log/apache2/access.log | sed '/192.168./d' | cut -d "-"
```

RETOURNE

```
27.50.160.35
```

EXPLICATION

“grep -i” va rechercher dans le fichier /var/log/apache2/access.log tout les ligne avec le mot “POST”
“sed '/xxx/d'” va supprimer toute les lignes contenant l'expression régulière '/192.168./d'
“cut -c -c16” va couper chaque ligne en partant du caractèrer 0 à 16 afin d'avoir les 16 premiers seulement par ligne.
cut -d “-” va couper chaque ligne dès qu'il trouvera le caractère “-”.
“tr -d 'x'” va tronquer les lignes en enlevant tout les caractères 'x' ou 'y' dans les lignes.
“ | ” quant à lui sert juste de re-directeur de flux pour le filtrage de donnée.

Format Date

Dans certains cas l'horodatage des logs est au format US, ce qui est bloquant quand votre système est en FR !!
Voici une solution avec différents exemple en fonction des formats de log ! :

apache2/error.log

Si je check la date :

```
date
dim. 10 août 2025 13:20:46 CEST
```

Dans mon log avec la commande suivante :

```
grep "invalid URI path" /var/log/apache2/error.log | sed 's/\[client/\ /' |
```

```
awk {'print $1,$2,$3,$4,$5"\n",$10"\n",$15'} | tr -d '[]'
```

J'ai :

```
Fri Aug 09 05:47:26.035928 2025
 119.18.55.217:48930
 (/cgi-bin/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/bin/sh)
Sun Aug 10 10:28:27.075533 2025
 167.86.109.190:55428
 (/cgi-bin/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/bin/sh)
```

Si je veux que le log de la date du jour, voici comment modifier ma commande:

```
grep "invalid URI path" /var/log/apache2/error.log | grep "$(env TZ=UTC
LC_ALL=C date +%a\ %b\ %d)" | sed 's/\[client/\ /' | awk {'print
$1,$2,$3,$4,$5"\n",$10"\n",$15'} | tr -d '[]'
```

Elle me retourne :

```
Sun Aug 10 10:28:27.075533 2025
 167.86.109.190:55428
 (/cgi-bin/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/.%2e/bin/sh)
```

apache2/acces.log

```
grep -i "SCAN\|Zgrab\|Censysinspect" /var/log/apache2/access.log | sed
'/10.10./d' | sed '/mazinger/d' | tr -d '[]();"'
```

Me renvoie :

```
164.92.251.237 - - 07/Aug/2025:01:11:31 +0200 GET /ab2g HTTP/1.1 404 2721 -
Mozilla/5.0 zgrab/0.x
164.92.251.237 - - 08/Aug/2025:01:11:32 +0200 GET /ab2h HTTP/1.1 404 2721 -
Mozilla/5.0 zgrab/0.x
104.131.21.188 - - 09/Aug/2025:01:42:09 +0200 GET /ab2g HTTP/1.1 404 2721 -
Mozilla/5.0 zgrab/0.x
104.131.21.188 - - 10/Aug/2025:01:42:09 +0200 GET /ab2h HTTP/1.1 404 2720 -
Mozilla/5.0 zgrab/0.x
137.184.77.127 - - 10/Aug/2025:02:09:05 +0200 GET /.env HTTP/1.1 404 341 -
Mozilla/5.0 Keydrop.io/1.0onlyscans.com/about
```

Avec filtrage :

```
grep -i "SCAN\|Zgrab\|Censysinspect" /var/log/apache2/access.log | sed
'/10.200./d' | sed '/mazinger/d' | tr -d '[]();"' | grep "$(env TZ=UTC
LC_ALL=C date +%d/%b/%Y)" | awk {'print $1,$2,$11,$12,$13,$14,$15,$16'}
```

Me renvoie :

```
104.131.21.188 - - 10/Aug/2025:01:42:09 +0200 GET /ab2h HTTP/1.1 404 2720 -
Mozilla/5.0 zgrab/0.x
137.184.77.127 - - 10/Aug/2025:02:09:05 +0200 GET /.env HTTP/1.1 404 341 -
Mozilla/5.0 Keydrop.io/1.0onlyscans.com/about
```

icecast2/access.log

```
grep -i "GET" /var/log/icecast2/access.log | sed '/10.200./d' | sed
'/192.168./d' | sed '/mazinger/d' | tr -d '[]' | cut -d "+" -f 1 | sed
's/....$//' | uniq -c
```

Me renvoie :

```
7 149.50.96.114 - - 08/Aug/2025:01:02
8 205.210.31.209 - - 09/Aug/2025:04:39
4 149.50.96.114 - - 10/Aug/2025:06:14
22 89.248.168.227 - - 10/Aug/2025:07:53
```

Avec Filtrage :

```
grep -i "GET" /var/log/icecast2/access.log | grep "$(env TZ=UTC LC_ALL=C
date +%d/%b/%Y)" | sed '/10.200./d' | sed '/192.168./d' | sed '/mazinger/d'
| tr -d '[]' | cut -d "+" -f 1 | sed 's/....$//' | uniq -c
```

Me renvoie :

```
4 149.50.96.114 - - 10/Aug/2025:06:14
22 89.248.168.227 - - 10/Aug/2025:07:53
```

Fail2ban.log

```
grep -w "Ban" /var/log/fail2ban.log | grep -v "Restore Ban" | awk {'print
$1,$2,$6,$8'}
```

Me renvoie :

```
2025-08-07 07:34:43,011 [apache-404] 199.45.154.141
2025-08-08 08:29:33,566 [apache-404] 34.76.2.252
2025-08-09 08:29:33,967 [icecast2] 34.76.2.252
2025-08-10 08:51:51,014 [apache-404] 104.234.115.80
2025-08-10 10:06:40,774 [apache-400] 137.184.77.127
```

```
grep -w "Ban" /var/log/fail2ban.log | grep -v "Restore Ban" | grep "$(date
+%Y-%m-%d)" | awk {'print $1,$2,$6,$8'}
```

Me renvoie :

```
2025-08-10 08:51:51,014 [apache-404] 104.234.115.80
2025-08-10 10:06:40,774 [apache-400] 137.184.77.127
```

— *sylvain 2020/03/01 17:24*