

Veille notion de Syslog sous Linux.

Qu'est ce que c'est que le syslog ?

Le syslog est le processus de journalisation qui enregistre les événements afin de permettre de localiser rapidement les défaillances d'un système. Chaque service possède son propre fichier de log qui sont tous stockés dans la partition `/var/log`.

A quoi ressemble une ligne d'évènement ?

Dans chaque ligne d'évènement on distingue :

- La date à laquelle l'évènement a été déclenché
- Le processus déclencheur de l'évènement
- Le processus ayant demandé l'ajout du message correspondant au log
- Le niveau de gravité du message (priority)

Les types de messages :

Il existe 5 types de messages qui sont les suivantes :

<code>/var/log/secure</code>	Syslog stocke dans le fichier de log « secure » tous les messages liés à la sécurité y compris ceux de l'authentification.
<code>/var/log/maillog</code>	Syslog stocke les messages liés aux messageries.
<code>/var/log/cron</code>	Contient tous les messages liés au démarrage du système.
<code>/var/log/boot.log</code>	Contient tous les messages liés au démarrage du système.
<code>/var/log/messages</code>	La plupart des messages log sont enregistré dans le fichier <code>/var/log/messages</code> , sauf les types de messages qu'on a vus précédemment.

Fonctionnement du syslog

Syslog possède un fichier de configuration « `syslog.conf` » qui est stocké dans le répertoire `/etc`. Ce fichier est sous la forme « `facility.priorité /var/log/fichierlog.log` » :

- La Priorité indique la criticité du message généré par un programme.

Code	Priorité	Description
0	Emergency	Le système est inutilisable
1	Alert	Une action immédiate est requise
2	Critical	Condition critique
3	Errors	Erreurs détectés
4	Warning	Avertissement
5	Notice	Événement normal mais significatif.
6	Info	Message d'information
7	Debug	débogage

- La facility indique le type de message généré par un programme

Type	Description
Kern	Utilisé pour les messages du noyau
user	Utilisateur
mail	Messagerie
cron	le planificateur des tâches
auth	Utilisé pour plusieurs événements de sécurité.
authpriv	Utilisé pour les contrôles d'accès
dæmon	Utilisé par les process système et les daemon.
mark	Pour les messages générés par syslog lui-même contenant un horodatage et la chaîne de caractère "--MARK--"

Ressources :

https://fr.wikibooks.org/wiki/Le_syst%C3%A8me_d%27exploitation_GNU-Linux/Les_fichiers_journaux_syslog
<https://www.linuxtricks.fr/wiki/syslog-les-journaux-système-sous-linux>
<https://sysreseau.net/syslog-la-journalisation-sous-linux/>
<https://doc.ubuntu-fr.org/syslog-ng>

Activité

Notre application Laravel doit permettre d'inscrire ses évènements dans le syslog.

Autrement dit, nous devons pouvoir référencer des fichiers de logs concernant :

- Nombre de tentative de connexion
- Quand le compte est bloqué

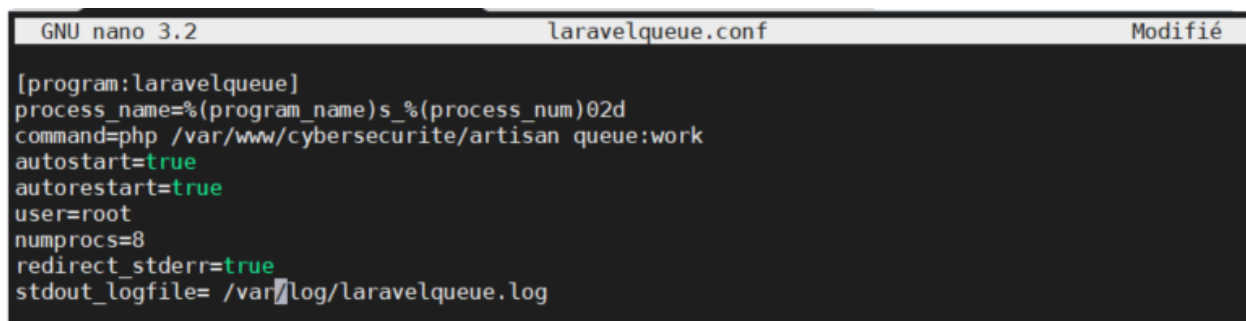
Ajustement :

Avant de mettre en place l'écriture dans le syslog du système, je dois faire persister la commande « php artisan queue:work » afin que les tâches cron se déclenchent correctement. Pour ce faire, j'utilise un package permettant de démarrer automatiquement la commande.

J'installe le package « supervisor » grâce à la commande :

- `sudo apt-get install supervisor`

Ensuite j'ajoute un `laravelqueue.conf` dans le dossier `/etc/supervisor/conf.d` afin de lui indiquer le processus qu'il doit automatiquement lancer.



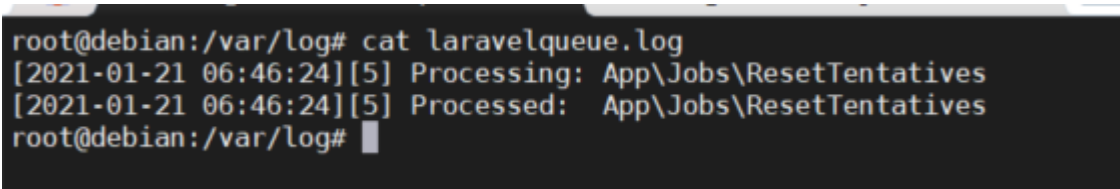
```
GNU nano 3.2                                laravelqueue.conf                                Modifié

[program:laravelqueue]
process_name=%(program_name)s_%(process_num)02d
command=php /var/www/cybersecurite/artisan queue:work
autostart=true
autorestart=true
user=root
numprocs=8
redirect_stderr=true
stdout_logfile= /var/log/laravelqueue.log
```

Je redémarre le service afin que mon fichier de conf soit pris en compte :

- `sudo /etc/init.d/supervisor restart`

En faisant le test, je peux voir que cela fonctionne correctement :



```
root@debian:/var/log# cat laravelqueue.log
[2021-01-21 06:46:24][5] Processing: App\Jobs\ResetTentatives
[2021-01-21 06:46:24][5] Processed: App\Jobs\ResetTentatives
root@debian:/var/log#
```

Ressources :

<https://blog.elao.com/fr/infra/utiliser-supervisor-pour-controler-ses-services-applicatifs/>

Mise en place de l'écriture dans le syslog du système et non dans laravel

Modification du fichier rsyslog.conf pour l'écriture dans un fichier de log spécifique :

```
GNU nano 3.2 /etc/rsyslog.conf

# Some "catch-all" log files.
#
*,debug;\
auth,authpriv.none;\
news,news.none;mail.none     -/var/log/debug
*,info;*,notice;*,warn;\
auth,authpriv.none;\
cron,daemon.none;\
mail,news.none               -/var/log/messages
#
# Emergencies are sent to everybody logged in.
#
*,emerg                       :omusrmsg:*
#
local0.*info /var/log/securityinfo.log
```

Modification de la manière de faire les logs :

```
AuthController.php x README.md app.php .env
app > Http > Controllers > AuthController.php
43 it($user) {
44     $user->tentatives = $user->tentatives + 1;
45     $user->save();
46     if($user->tentatives > 3) {
47         $user->tentatives = 3;
48         $user->save();
49         $resetJob = (new ResetTentatives($user->id))->delay(Carbon::now()->addSeconds(30));
50         dispatch($resetJob);
51
52         openlog('cybersecurite_app', LOG_NDELAY, LOG_USER);
53         syslog(LOG_INFO|LOG_LOCAL0, "L'utilisateur {$user->email} à atteint son nombre maximal de tentative de connexion ! ");
54         // Log::channel('abuse')->info("L'utilisateur {$user->email} à atteint son nombre maximal de tentative de connexion ! ");
55
56         return response()->json([
57             'success' => false,
58             'type' => 'info',
59             'message' => "Veuillez réessayer dans 30 secondes",
60         ]);
61     }
62 }
63 }
```

Test et vérification après 3 tentatives d'échecs :

```
3. secu@debian: /var/www/cybersecurite x 4. secu@debian: /var/log
root@debian:/var/log/web# cd ..
root@debian:/var/log# ls
alternatives.log      faillog               messages              syslog.2.gz
alternatives.log.1    cups                 fontconfig.log        syslog.3.gz
alternatives.log.2.gz daemon.log            gdm3                 mysql                 unattended-upgrades
apache2               daemon.log.1         hp                    private              user.log
apt                   debug               installer             securityinfo.log     user.log.1
auth.log              debug.1             kern.log              speech-dispatcher    web
auth.log.1            dpkg.log            kern.log.1            supervisor           wtmp
boot.log              dpkg.log.1          laravelqueue.log      syslog
btm                   dpkg.log.2.gz       lastlog               syslog.1
root@debian:/var/log# cat securityinfo.log
Jan 21 08:44:15 debian cybersecurite_app: L'utilisateur admin@gmail.com à atteint son nombre maximal de tentative de connexion !
root@debian:/var/log#
```

```
3. secu@debian: /var/www/cybersecurite x 4. secu@debian: /var/log
root@debian:/var/log# cat securityinfo.log
Jan 21 08:44:15 debian cybersecurite_app: L'utilisateur admin@gmail.com à atteint son nombre maximal de tentative de connexion !
Jan 21 08:46:57 debian cybersecurite_app: L'utilisateur admin@gmail.com à atteint son nombre maximal de tentative de connexion !
root@debian:/var/log#
```

```
rd 37 from systemd. [v8.1901.0]
Jan 21 10:15:58 debian rsyslogd: [origin software="rsyslogd" swVersion="8.1901.0" x-pid="30279" x-info="https://www.rsyslog.com"] start
Jan 21 10:15:58 debian systemd[1]: Started System Logging Service.
Jan 21 10:16:52 debian cybersecurite_app: L'utilisateur admin@gmail.com à atteint son nombre maximal de tentative de connexion !
Jan 21 10:17:01 debian CRON[30385]: (root) CMD ( cd / && run-parts --report /etc/cron.hourly)
Jan 21 10:20:11 debian systemd[1]: Started Session 37 of user secu.
Jan 21 10:20:12 debian systemd[1]: session-37.scope: Succeeded.
root@debian: /var/log#
```