REALISATION PERSONNELLE

Gestion des logs avec Graylog sous Debian 11

Auteur : Winness RAKOTOZAFY

Version de février 2023

AVANT – PROPOS

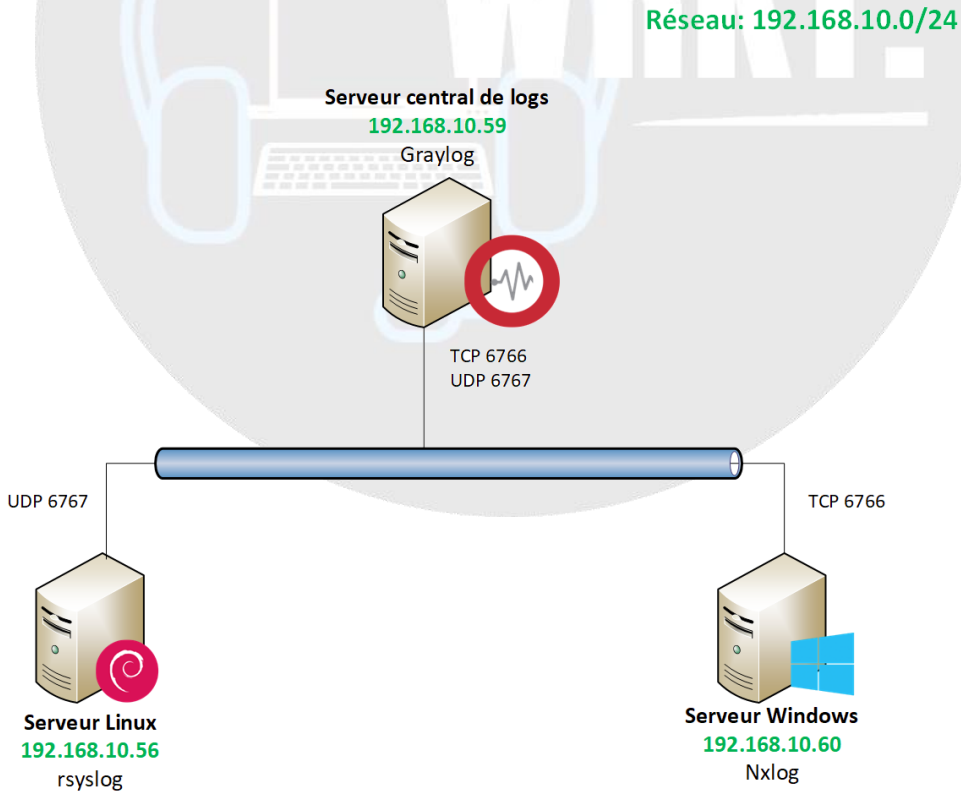
La gestion des journaux d'évènement est une opération cruciale pour la sécurité informatique d'une organisation. En effet, dans un scénario d'attaque, la gestion des logs externalisée et centralisée sur un serveur externe permet aux administrateurs du parc de garder une trace des actions effectués sur le système d'information.

Entre autre, la centralisation de logs est un garant du principe de Traçabilité dans la sécurité informatique.

Dans le cadre de cette documentation, nous allons mettre en place cette gestion des logs en utilisation la solution **Graylog** sous un serveur Debian.

Le choix de Graylog s'est effectué du fait que c'est une solution gratuite et considéré comme un des **leaders du marché,** au côté de Splunk, dans les fonctionnalités de centralisation, de visualisation, mais aussi dans la surveillance d'une infrastructure. Son côté graphique via une interface web motive également le choix de la solution, qui nous permettra de visualiser en temps réel les logs des différents serveurs, applications, bases de données et équipements de l'infrastructure.

Pour mieux comprendre les opérations que nous allons mener au cours de cette documentation, veuillez-vous référer au schéma ci-dessous :



Enfin, cette expérimentation s'effectuera sous forme de machine virtuelle sous VMWare pour nous permettre de découvrir la solution et apprendre à administrer une solution demandée par les entreprises.

SOMMAIRE

I- Installation des prérequis

Selon la documentation officielle de Graylog, pour fonctionner correctement, il est nécessaire d'installer plusieurs utilitaires, à savoir :

- Une plateforme Java sous **OpenJDK** 17 (qui est préalablement installé nativement avec Graylog depuis la version 5.0) ;
- Une base de données **MongoDB**
- Un moteur de recherche sous **OpenSearch** ou **ElasticSearch**, qui permettra également l'implémentation des tableaux de bord pour l'exploitation des logs.

Ainsi, comme nous venons de découvrir la solution, comme toute première découverte, procéder à l'installation de la solution en suivant la documentation officielle de Graylog disponible sur le lien suivant : https://go2docs.graylog.org/5-0/home.htm

1- Installation de MongoDB

Pour installer MongoDB, nous allons rajouter les dépôts officiels de MongoDB sur notre serveur.

Pour ce faire, nous allons tout d'abord installer **gnupg** pour le déchiffrement de la clé publique de MongoDB :

```
sudo apt install gnupg -y
```

Nous pouvons ainsi télécharger la clé publique de MongoDB, et l'ajouter dans les clés de confiance d'apt. Pour ce faire, lancez donc les commandes suivantes :

```
wget -qO - https://www.mongodb.org/static/pgp/server-6.0.asc | sudo apt-key add -
```

Puis, rajoutez le dépôt official de MongoDB dans la liste des sources du gestionnaire de paquets apt :

```
echo "deb http://repo.mongodb.org/apt/debian bullseye/mongodb-org/6.0 main" | sudo tee /etc/apt/sources.list.d/mongodb-org-6.0.list
```

Mettez à jour les dépôts par un **apt update**, et lancez la commande d'installation de MongoDB :

```
sudo apt update && sudo apt install mongodb-org -y
```

Enfin, démarrez les services de mongodb, et les paramétrer de sorte qu'ils démarrent également au démarrage de notre système :

```
sudo systemctl enable mongod.service
sudo systemctl restart mongod.service
```

Vérifiez que le service est bien installé et opérationnel :

```
sudo systemctl status mongod.service
```

```
sysadmin@graylog:~$ sudo systemctl status mongod.service
● mongod.service - MongoDB Database Server
     Loaded: loaded (/lib/systemd/system/mongod.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2023-02-25 20:41:34 CET; 12s ago
       Docs: https://docs.mongodb.org/manual
   Main PID: 4165 (mongod)
     Memory: 75.5M
        CPU: 561ms
     CGroup: /system.slice/mongod.service
             └─4165 /usr/bin/mongod --config /etc/mongod.conf

févr. 25 20:41:34 graylog systemd[1]: Started MongoDB Database Server.
```

MongoDB est bien installé et opérationnel. A présent, nous allons installer le moteur de recherche **Opensearch**

2- Installation et configuration d'Opensearch

Comme moteur de recherche, nous installerons Opensearch, du fait que les nouvelles versions d'ElasticSearch ne sont plus supportées par Graylog. Les étapes d'installation d'Openseach sont détaillé dans la documentation officielle d'installation.

Tout d'abord, désactivez les Transparent Hugepages pour de meilleures performances au niveau de l'exploitation de la base de données. Pour ce faire, créez un service nommé disable-transparent-huge-pages en renseignant les commandes suivantes :

```
echo "Description=Disable Transparent Huge Pages (THP)

DefaultDependencies=no

After=sysinit.target local-fs.target

[Service]

Type=oneshot

ExecStart=/bin/sh -c 'echo never | tee
/sys/kernel/mm/transparent_hugepage/enabled > /dev/null'

[Install]

WantedBy=basic.target" | sudo tee /etc/systemd/system/disable-
transparent-huge-pages.service
```

Ensuite, démarrez les services de désactivation de transparent-huge-pages :

```
sudo systemctl enable disable-transparent-huge-pages.service

sudo systemctl start disable-transparent-huge-pages.service
```

Ajoutez un utilisateur pour Opensearch :

```
sudo adduser --system --disabled-password --disabled-login --home
/var/empty --no-create-home --quiet --force-badname --group opensearch
```

Procédez ensuite au téléchargement de l'archive d'installation d'Opensearch :

```
wget https://artifacts.opensearch.org/releases/bundle/opensearch/2.4.1/opensearch-2.4.1-linux-x64.tar.gz
```

Créez les dossiers suivants pour y stocker les données et les fichiers journaux d'Opensearch :

```
sudo mkdir -p /graylog/opensearch/data
sudo mkdir /var/log/opensearch
```

Extraire le contenu de l'archive et déplacez les contenus dans le premier dossier créé précédemment :

```
sudo tar -zxf opensearch-2.4.1-linux-x64.tar.gz
sudo mv opensearch-2.4.1/* /graylog/opensearch/
```

Puis, définissez les bons droits sur les fichiers/dossiers d'opensearch et rajoutez l'utilisateur opensearch créé précédemment comme propriétaire du fichier :

```
sudo chown -R opensearch:opensearch /graylog/opensearch/
sudo chown -R opensearch:opensearch /var/log/opensearch
sudo chmod -R 2750 /graylog/opensearch/
sudo chmod -R 2750 /var/log/opensearch
```

- Le droit **2750**, permet entre autres d'assigner les droits de lecture, écriture, exécution au propriétaire, lecture, écriture (spéciale (2)), exécution au groupe, et aucun droit pour les autres.

```
sysadmin@graylog:~$ ls -l /graylog/
total 4
drwxr-s--- 11 opensearch opensearch 4096 26 févr. 11:06 opensearch
sysadmin@graylog:~$
```

Créez ensuite un fichier log vide pour graylog

```
sudo -u opensearch touch /var/log/opensearch/graylog.log
```

Créez le service opensearch.service

```
echo "[Unit]
Description=Opensearch
Documentation=https://opensearch.org/docs/latest
Requires=network.target remote-fs.target
After=network.target remote-fs.target
ConditionPathExists=/graylog/opensearch
ConditionPathExists=/graylog/opensearch/data
[Service]
Environment=OPENSEARCH_HOME=/graylog/opensearch
Environment=OPENSEARCH_PATH_CONF=/graylog/opensearch/config
```

```
ReadWritePaths=/var/log/opensearch
User=opensearch
Group=opensearch
WorkingDirectory=/graylog/opensearch
ExecStart=/graylog/opensearch/bin/opensearch
# Specifies the maximum file descriptor number that can be opened by this process
LimitNOFILE=65535
# Specifies the maximum number of processes
LimitNPROC=4096
# Specifies the maximum size of virtual memory
LimitAS=infinity
# Specifies the maximum file size
LimitFSIZE=infinity
# Disable timeout logic and wait until process is stopped
TimeoutStopSec=0
# SIGTERM signal is used to stop the Java process
KillSignal=SIGTERM
# Send the signal only to the JVM rather than its control group
KillMode=process
# Java process is never killed
SendSIGKILL=no
# When a JVM receives a SIGTERM signal it exits with code 143
SuccessExitStatus=143
# Allow a slow startup before the systemd notifier module kicks in to extend the timeout
TimeoutStartSec=180
[Install]
WantedBy=multi-user.target" | sudo tee /etc/systemd/system/opensearch.service
```

Maintenant que OpenSearch est installé, il nous faut configurer les fichiers d'Opensearch pour se diriger vers Graylog. Pour ce faire, éditez le fichier **`/graylog/opensearch/config/opensearch.yml`**

```
sudo vim /graylog/opensearch/config/opensearch.yml
```

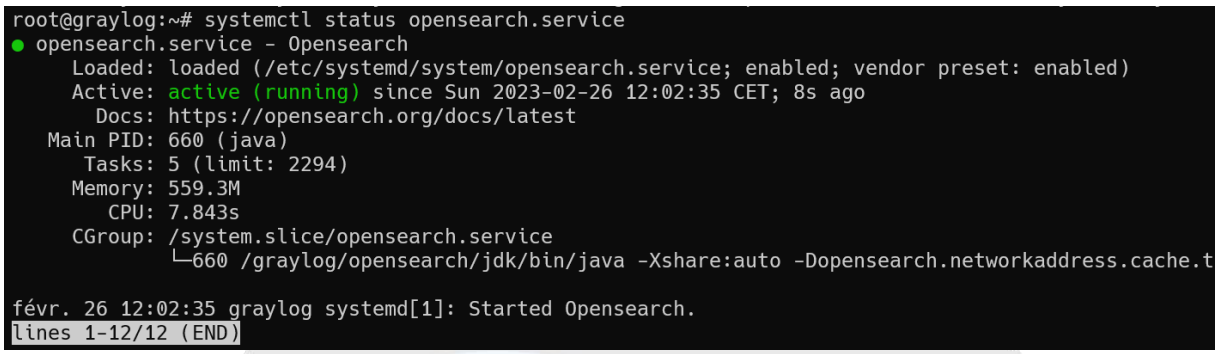
Et modifiez les lignes suivantes (le reste de la configuration est à adapter selon vos besoins de sécurité) :

```
cluster.name: graylog
path.data: /graylog/opensearch/data
path.logs: /var/log/opensearch
network.host: 0.0.0.0
discovery.type: single-node
action.auto_create_index: false
plugins.security.disabled: true
```

Démarrez enfin les service opensearch.service :

```
sudo systemctl daemon-reload
sudo systemctl enable opensearch.service
sudo systemctl start opensearch.service
```

Vérifiez que le service est bien opérationnel :



A présent, les prérequis sont installés et opérationnels, nous allons passer à l'installation de Graylog en lui-même.

II- Installation et configuration de Graylog

1- Installation des paquets de Graylog

Tout d'abord, téléchargez le paquet Graylog depuis le dépôt officiel :

```
curl -OSL https://packages.graylog2.org/repo/packages/graylog-5.0-
repository_latest.deb
```

Puis, installez les dépôts sur le serveur en utilisant la commande **dpkg**

```
sudo dpkg -I graylog-5.0-repository_latest.deb
```

Enfin, installez le serveur graylog

```
sudo apt-get update && sudo apt-get install graylog-server-y
```

Patientez l'installation et passez à la configuration des fichiers de configuration

2- Configuration de Graylog

Selon la documentation officielle, il est important de définir les mots de passe du compte administrateur en version hashé 256sum, et un mot de passe fort dans le fichier de configuration **`/etc/graylog/server/server.conf`**.

Pour ce faire, générez le mot de passe hashé en lançant la commande suivante :

```
echo -n “Enter Password: “ && head -1 </dev/stdin | tr -d ‘\n’ |
sha256sum | cut -d” “ -f1
```

```
sysadmin@graylog:~$ echo -n "Enter Password: " && head -1 </dev/stdin | tr -d '\n' | sha256sum | cut -d " " -f1
Enter Password:
b03ddf3ca2e714a6548e7495e2a
sysadmin@graylog:~$
```

Conservez ensuite la sortie de la commande, nous l’utiliserons plus tard.

Générer un mot de passe fort avec pwgen

```
pwgen -N 1 -s 96
```

```
sysadmin@graylog:~$ pwgen -N 1 -s 96
sysadmin@graylog:~$
```

Copiez ensuite ces mots de passe sur les lignes **password_secret** pour le mot de passe généré par pwgen, et **root_password_sha2** pour le mot de passe hashé 256 généré.

Modifier la ligne sur **http_bind** avec l’adresse IP de votre serveur pour permettre aux ordinateurs des administrateurs de votre parc d’accéder à l’interface web de Graylog. Cependant, pour des raisons de sécurité, il serait judiciable de n’autoriser que les postes administrateurs pour accéder à l’interface web (par des règles de firewalling par exemple).

```
http_bind = 192.168.10.59:9000
```

Enfin démarrez les services du serveur Graylog :

```
sudo systemctl daemon-reload

sudo systemctl enable graylog-server

sudo systemctl start graylog-server
```

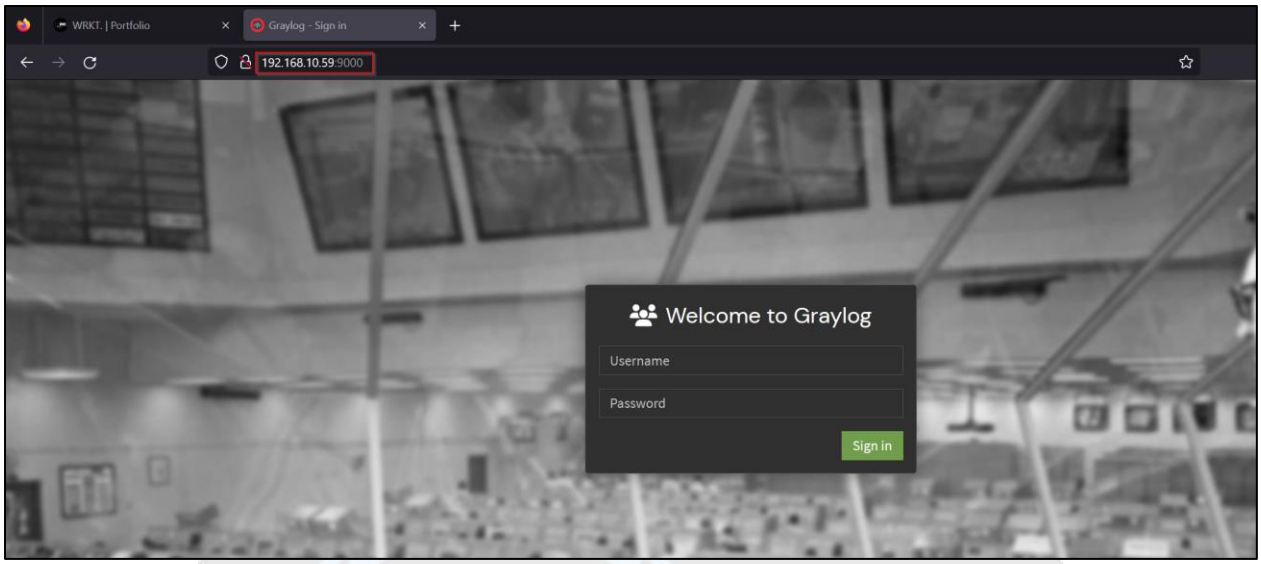
Vérifiez le bon fonctionnement du service :

```
sysadmin@graylog:~$ sudo systemctl status graylog-server
● graylog-server.service - Graylog server
     Loaded: loaded (/lib/systemd/system/graylog-server.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2023-02-26 13:29:34 CET; 9s ago
       Docs: http://docs.graylog.org/
   Main PID: 2100 (graylog-server)
      Tasks: 22 (limit: 2294)
     Memory: 243.9M
        CPU: 8.638s
     CGroup: /system.slice/graylog-server.service
             ├─2100 /bin/sh /usr/share/graylog-server/bin/graylog-server
             └─2101 /usr/share/graylog-server/jvm/bin/java -Xms1g -Xmx1g -server -XX:+UseG1GC -XX:-Omi

févr. 26 13:29:34 graylog systemd[1]: Started Graylog server.
févr. 26 13:29:36 graylog graylog-server[2101]: WARNING: sun.reflect.Reflection.getCallerClass is not
lines 1-14/14 (END)
```

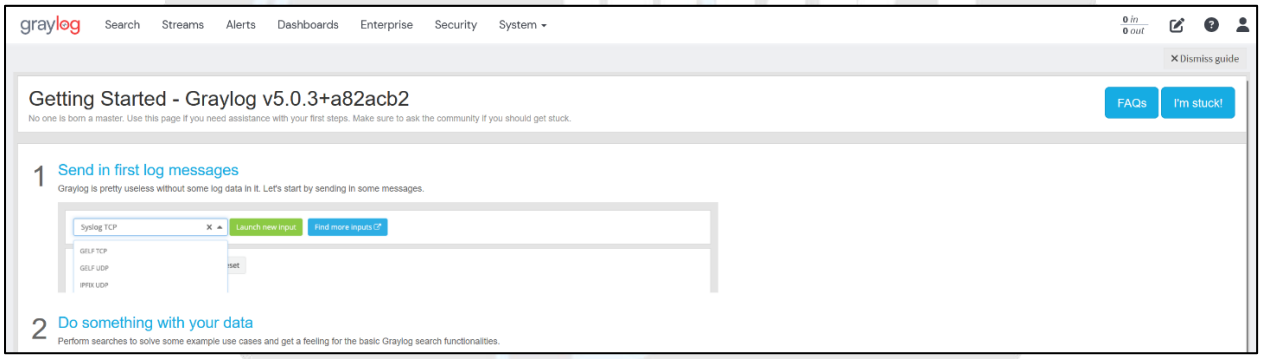
A présent, nous pouvons accéder à l’interface web de Graylog pour l’exploitation et la gestion des logs des équipements de notre parc.

3- Exploitation du serveur par l'interface web

Pour accéder à l'interface web de Graylog, allez sur un navigateur web, et renseignez l'adresse suivante : http:/192.168.10.59:9000



Renseignez ensuite le mot de passe que vous avez hashé précédemment, avec comme username **`admin`.** Par la suite, vous arriverez à la page d'accueil comme ci-dessous :

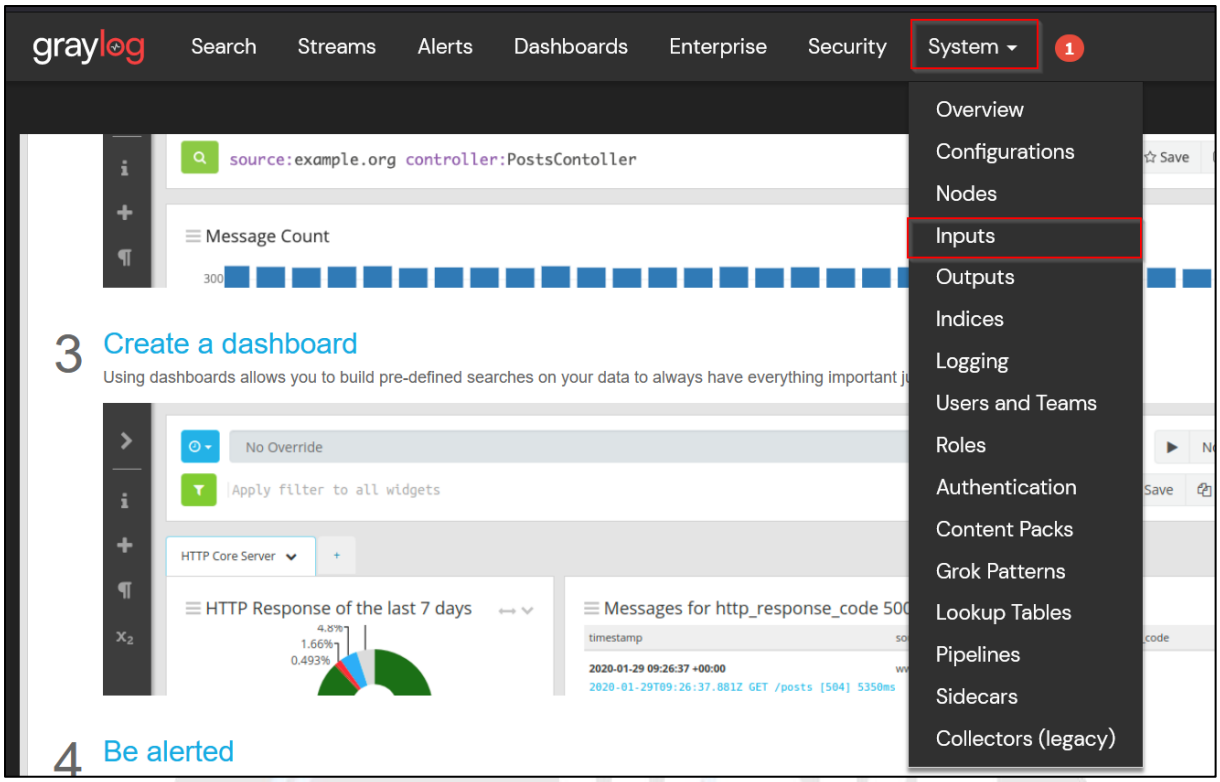


Excellent, vous êtes sur la page d'accueil de Graylog, ce qui prouve que toute l'installation effectué précédemment est réussie et fonctionnelle. Nous allons passer à présent à la configuration des **inputs**, ce qui permettra au serveur Graylog de récupérer les logs des serveurs distants.

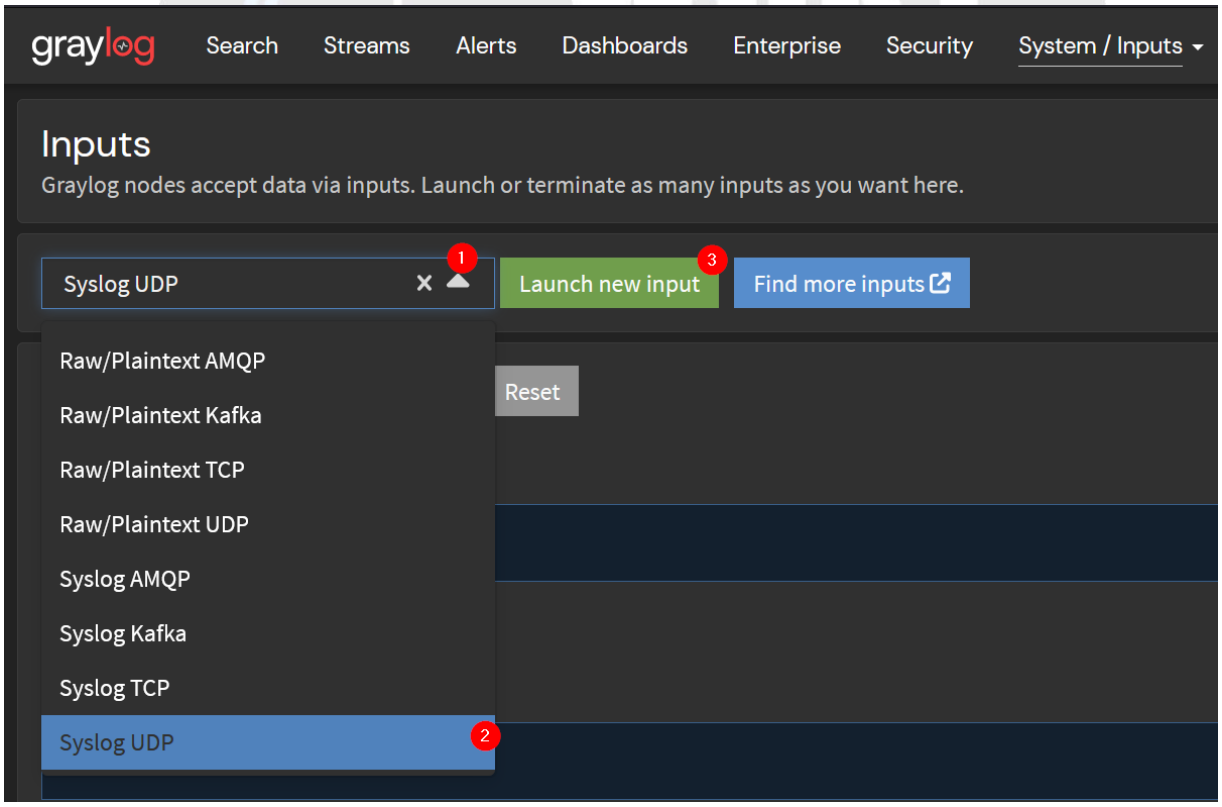
3.1- Configuration des inputs

Pour la configuration des inputs, vous avez le choix de configurer un input pour chaque serveur (mais ce qui ouvrira beaucoup trop de ports sur le serveur), ou créer un input pour chaque type de serveur.

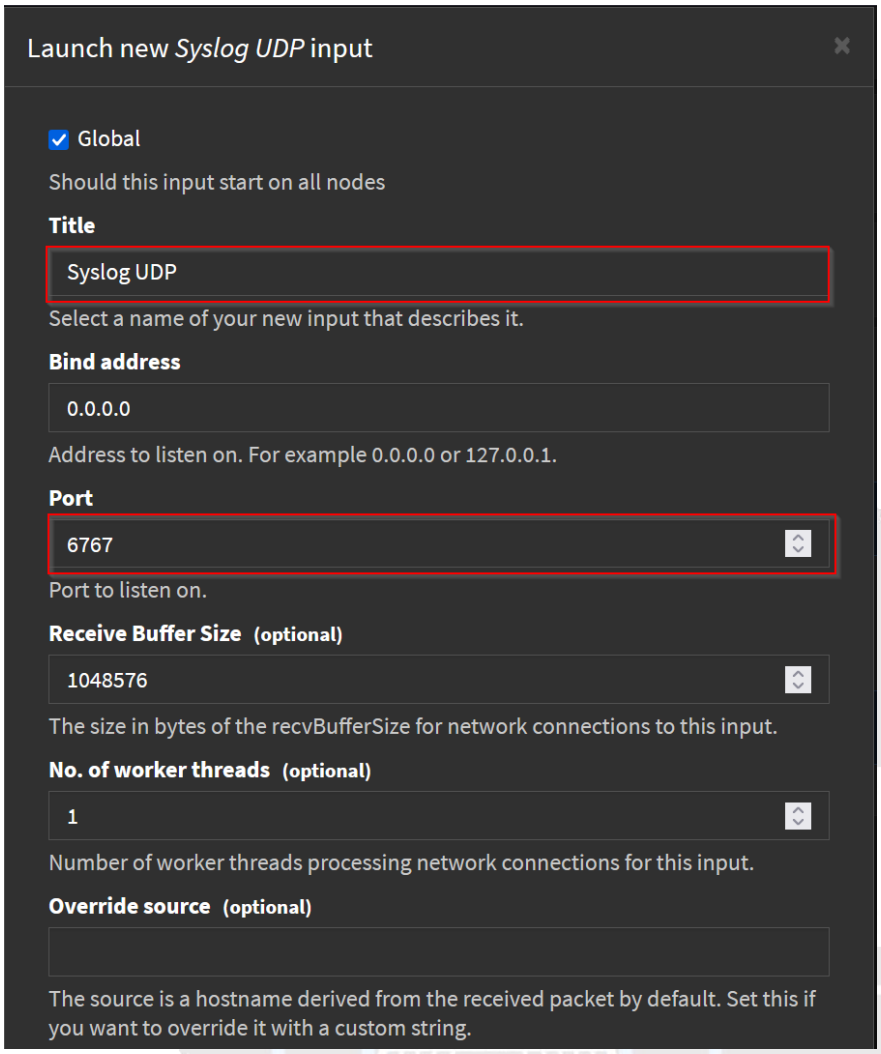
Sans plus attendre, créez un **input** pour les serveurs Linux du parc en cliquant sur **System → Inputs**



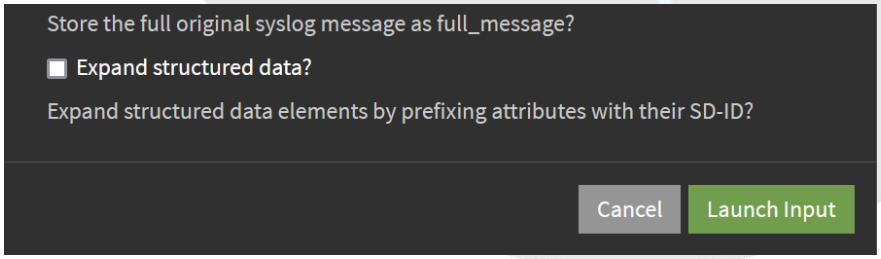
Ensuite, faites défiler la liste déroulante, sélectionnez **Syslog UDP** (ou TCP)**,** et cliquez sur **Launch new input**.



Configurez ensuite le nom de l'input et le port pour récupérer les logs depuis les serveurs distants (pour le reste, vous pouvez laisser les paramétrages par défaut):

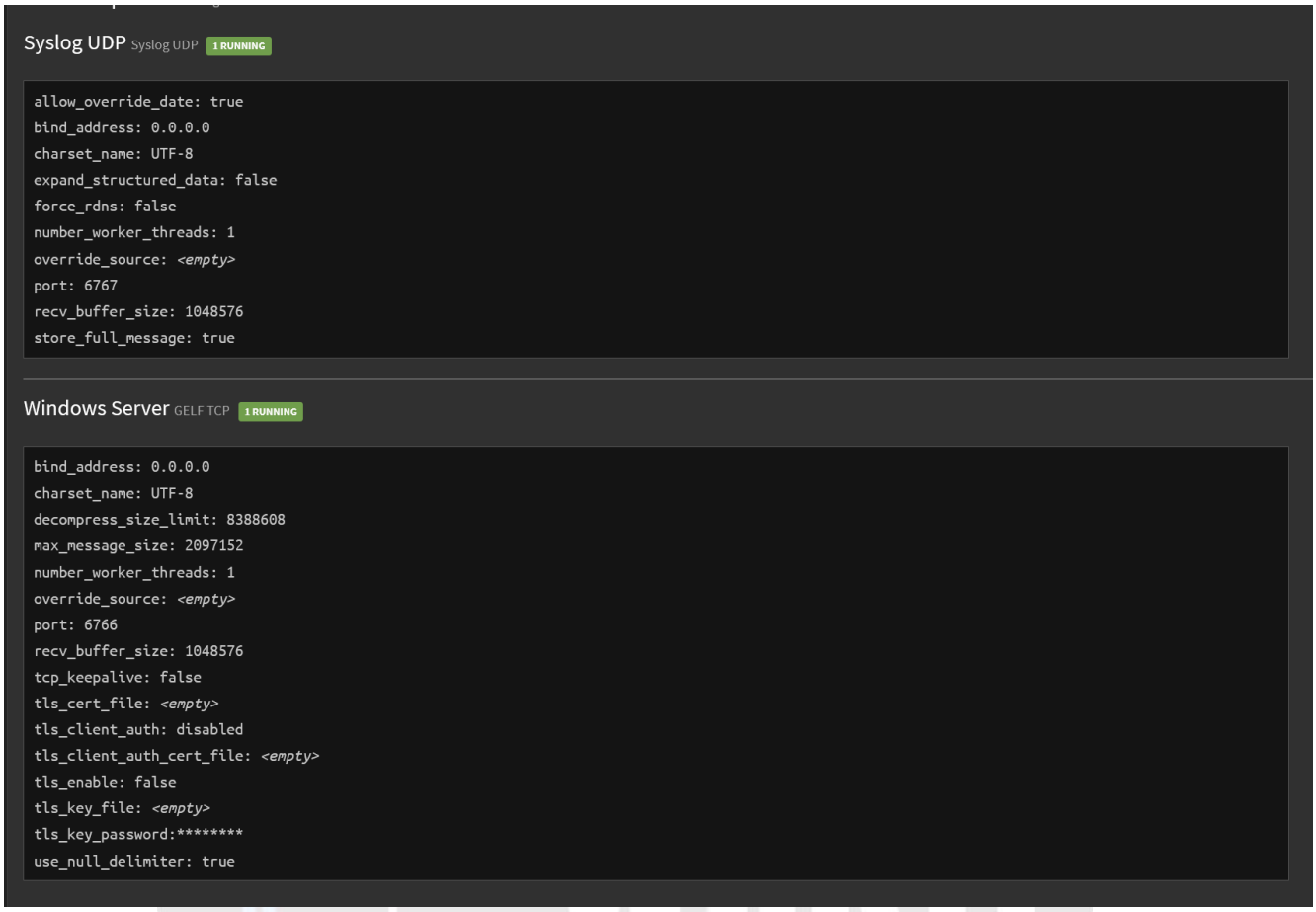


Quand ces paramétrages sont effectués, cliquez sur **Launch input**.



Nous créerons également un input pour les clients Windows. Bien que le format syslog est supporté, il est recommandé d'utiliser le format GELF (Graylog Extended Log Format) pour les logs de Windows afin de les rendre plus lisible.

Ainsi, nous aurons un input **Syslog UDP** pour les machines sous Linux, et un input **GELF TCP** pour les machines Windows.



A présent, nous allons configurer les machines clientes pour envoyer leur fichier journaux vers le serveur Graylog.

3.2- Configuration des machines clientes

Tout d'abord, concernant les machines clientes du parc, nous avons de l'un, les serveurs sous Linux, et de l'autre, les serveurs sous Windows. Sous Linux, l'envoi des fichiers de journaux vers un site distant exploite le service rsyslog, et pour les serveurs Windows, il nous faudra télécharger **NXLog**, pour configurer l'envoi des informations.

3.2.1- Serveur Linux

Sur le serveur Linux, éditez le fichier **/etc/rsyslog.conf**, et rajoutez les lignes suivantes sous RULES :

```
*.*                                     @192.168.10.59:6767
```

- *.* correspond à tous les logs du système (équivalent de syslog sur les machines Linux)
- **192.168.10.59** correspond à l'adresse IP de votre serveur Graylog (à adapter selon votre adresse IP)
- **6767**, correspond au port configuré à l'input de Graylog sur Syslog UDP

```
###############
#### RULES ####
###############

#
# First some standard log files.  Log by facility.
#
auth,authpriv.*                 /var/log/auth.log
*.*;auth,authpriv.none          -/var/log/syslog
#cron.*                         /var/log/cron.log
daemon.*                        -/var/log/daemon.log
kern.*                          -/var/log/kern.log
lpr.*                           -/var/log/lpr.log
mail.*                          -/var/log/mail.log
user.*                          -/var/log/user.log
*.*                             @192.168.10.59:6767
```

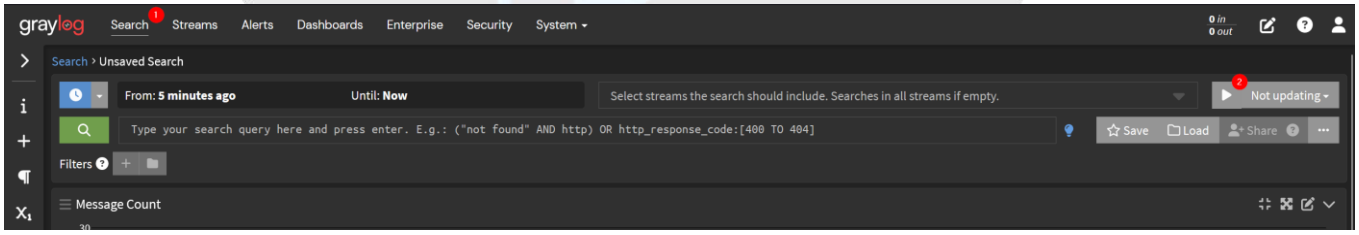
Redémarrez ensuite le service rsyslog

```
sudo systemctl restart rsyslog.service
```

Cette opération effectué, sur une machine du réseau, effectuons une connexion ssh sur la machine cliente Linux dont l'adresse est **192.168.10.56**

```
PowerShell 7.3.3
Loading personal and system profiles took 580ms.
 0s
↳  ssh sysadmin@192.168.10.56
sysadmin@192.168.10.56's password:
```

Puis vérifiez que le log d'authentification SSH sur le client Linux est bien remonté sur le serveur Graylog. Pour cela, cliquez sur l'onglet **Search,** puis cliquez sur le bouton **Play** pour récupérer les derniers logs des machines clientes



Les logs sont bien remontés sur le serveur Graylog, et mieux encore, pour faciliter la lecture, Graylog nous met en source directement le hostname de la machine distante.

```
sysadmin@asterisk:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:98:d8:50 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.10.56/24 brd 192.168.10.255 scope global dynamic ens33
       valid_lft 1698sec preferred_lft 1698sec
    inet6 fe80::20c:29ff:fe98:d850/64 scope link
       valid_lft forever preferred_lft forever
```

 # Centralisation des logs des machines Linux OK

A présent, passons à la configuration de l'envoi des logs sur les serveurs Windows.

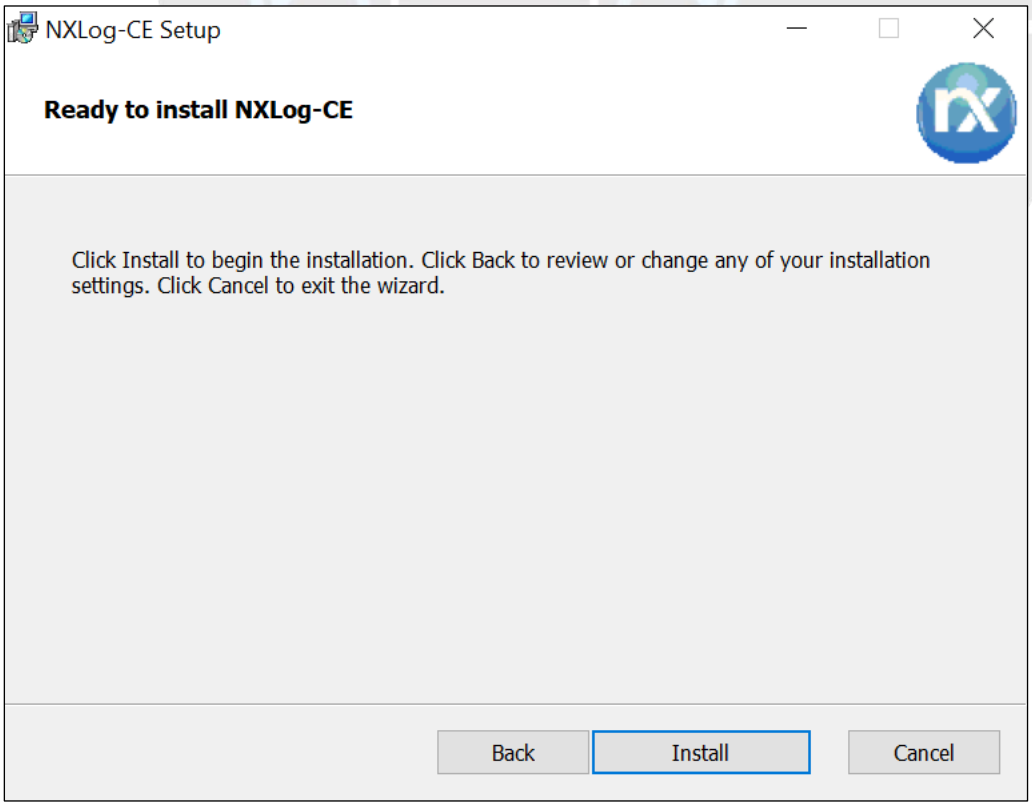
3.2.2- Serveur Windows

Sous Windows, afin de faire remonter les logs vers un serveur de centralisation de logs, nous allons installer **NxLog Community Edition**, qui est un logiciel de gestion de logs open source conçu pour collecter, transformer et acheminer des données de journalisation de manière fiable et sécurisée.

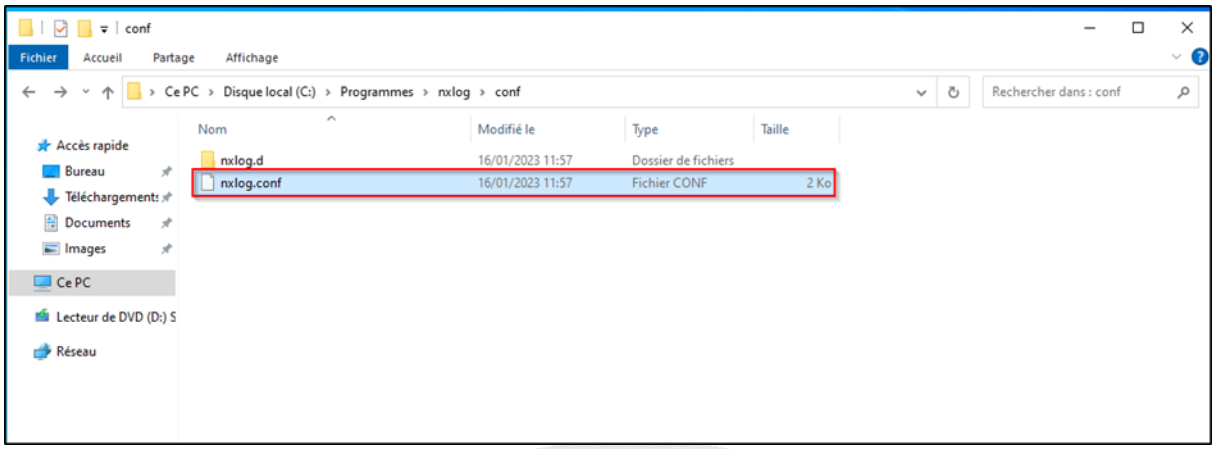
Contrairement à Linux, Windows n'a pas rsyslog installé nativement sur son système, d'où notre recours à cette solution.

Le logiciel est téléchargeable via le lien suivant : https://nxlog.co/downloads/nxlog-ce#nxlog-community-edition

Sur le serveur Windows, installez Nxlog :



L'installation terminé, modifiez le fichier de configuration situé à **`C:\Programmes\nxlog\conf\nxlog.conf`**

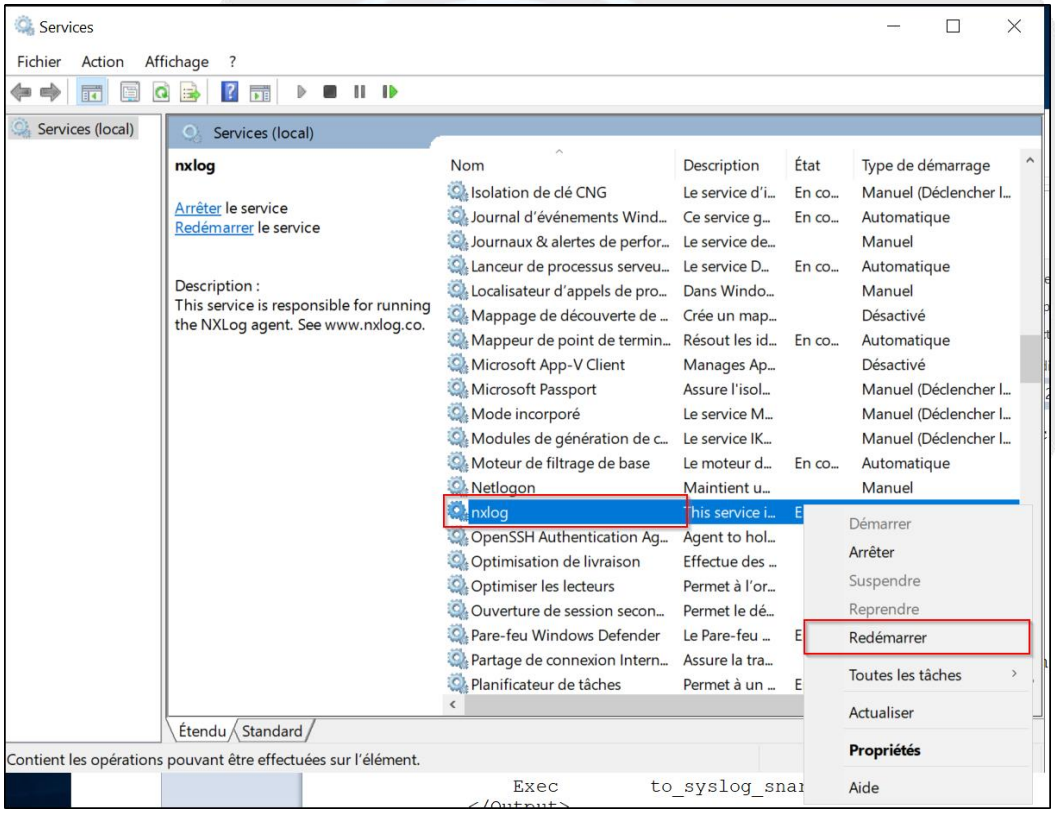


Rajoutez les extensions **xm_gelf**, décommentez les lignes d'input et configurer l'adresse IP du serveur central de logs sur l'Output. Mon fichier de configuration est comme ci-dessous :

```
Panic Soft
#NoFreeOnExit TRUE

define ROOT     C:\Program Files\nxlog
define CERTDIR  %ROOT%\cert
define CONFDIR  %ROOT%\conf\nxlog.d
define LOGDIR   %ROOT%\data

include %CONFDIR%\\*.conf
define LOGFILE  %LOGDIR%\nxlog.log
LogFile %LOGFILE%

Moduledir %ROOT%\modules
CacheDir  %ROOT%\data
Pidfile   %ROOT%\data\nxlog.pid
SpoolDir  %ROOT%\data

<Extension _syslog>
    Module      xm_syslog
</Extension>

<Extension _charconv>
    Module      xm_charconv
    AutodetectCharsets iso8859-2, utf-8, utf-16, utf-32
</Extension>

<Extension _exec>
    Module      xm_exec
</Extension>

<Extension _fileop>
    Module      xm_fileop

    # Check the size of our log file hourly, rotate if larger than 5MB
    <Schedule>
        Every   1 hour
        Exec    if (file_exists('%LOGFILE%') and \
                   (file_size('%LOGFILE%') >= 5M)) \
                    file_cycle('%LOGFILE%', 8);
    </Schedule>

    # Rotate our log file every week on Sunday at midnight
    <Schedule>
        When    @weekly
        Exec    if file_exists('%LOGFILE%') file_cycle('%LOGFILE%', 8);
    </Schedule>
</Extension>
```

```
<Extension gelf>
    Module      xm_gelf
</Extension>

# Snare compatible example configuration
# Collecting event log
 <Input in>
     Module      im_msvistalog
 </Input>
#
# Converting events to Snare format and sending them out over TCP syslog
 <Output out>
     Module      om_tcp
     Host        192.168.10.59
     Port        6766
     OutputType  GELF_TCP
 </Output>
#
# Connect input 'in' to output 'out'
 <Route 1>
     Path        in => out
 </Route>
```

Redémarrez ensuite le service nxlog sur Windows



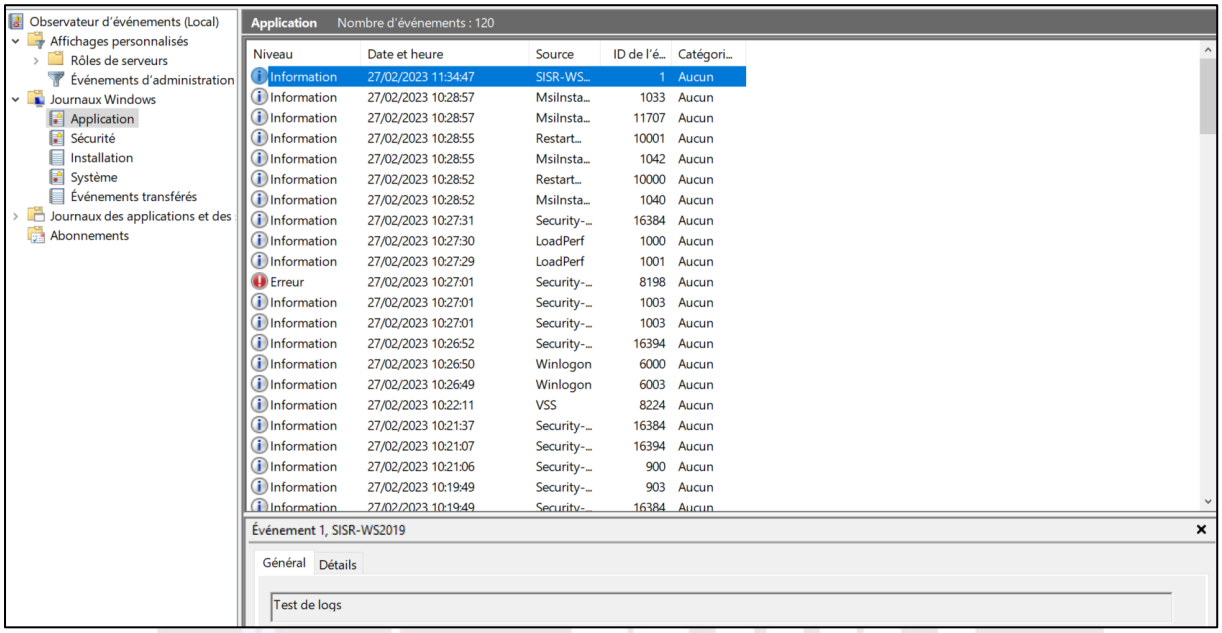
Vérifiez que le service est bien opérationnel en regardant les logs du service nxlog sous

`C:\Program Files\nxlog\data`

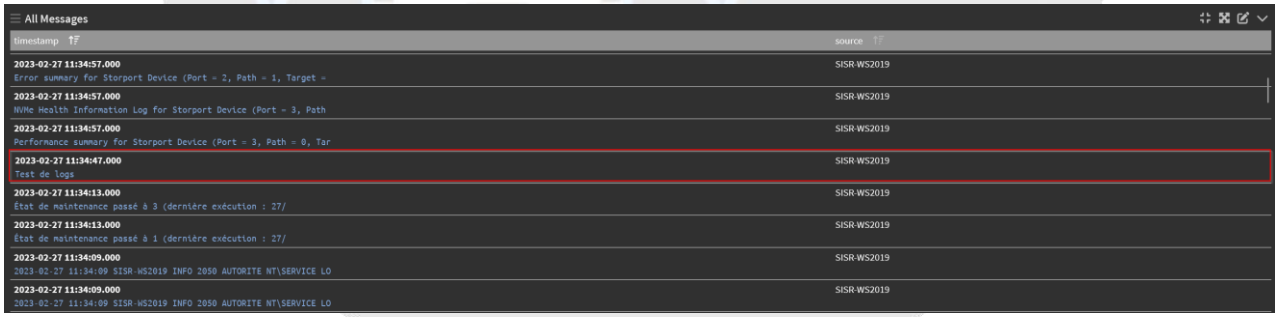
```
2023-02-27 11:07:14 WARNING nxlog-ce received a termination request signal, exiting...
2023-02-27 11:07:16 INFO connecting to 192.168.10.59:6766
2023-02-27 11:07:16 INFO nxlog-ce-3.1.2319 started
2023-02-27 11:14:36 WARNING stopping nxlog service
2023-02-27 11:14:36 WARNING nxlog-ce received a termination request signal, exiting...
2023-02-27 11:14:39 INFO connecting to 192.168.10.59:6766
2023-02-27 11:14:39 INFO nxlog-ce-3.1.2319 started
```

Comme les logs de Windows risque d'être surchargé, nous allons tester le bon fonctionnement du service en créant un log personnalisé sur le serveur Windows avec la commande suivante :

```
eventcreate /ID 1 /L APPLICATION /T INFORMATION /SO SISR-WS2019 /D "Test de logs"
```



Vérifions enfin que les logs et le log personnalisé soit remonté sur le serveur Graylog :



 # Centralisation des logs du serveur Windows OK

Maintenant que la centralisation des logs est opérationnel, nous allons voir ensemble comment créer un tableau de bord pour les graphes avec Graylog.

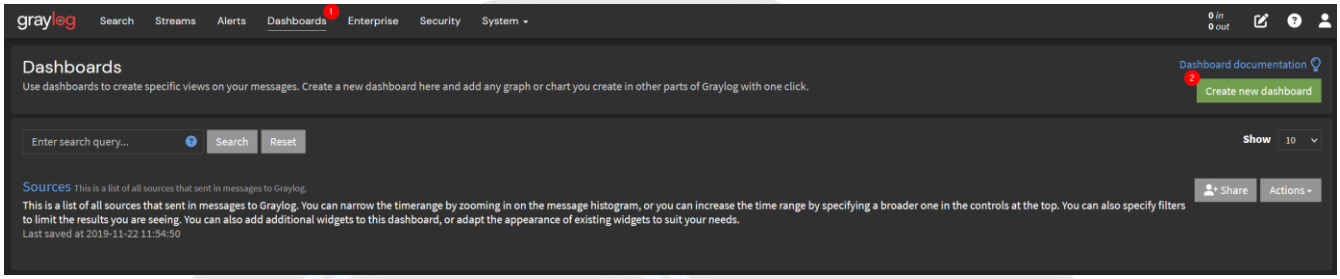
III- Ajout des tableaux de bord sous Graylog

Pour mieux visualiser la remontée d'information, il est possible sous Graylog de configurer des dashboard pour avoir un statistique des logs remontées par service (ssh, taux d'erreur, etc …).

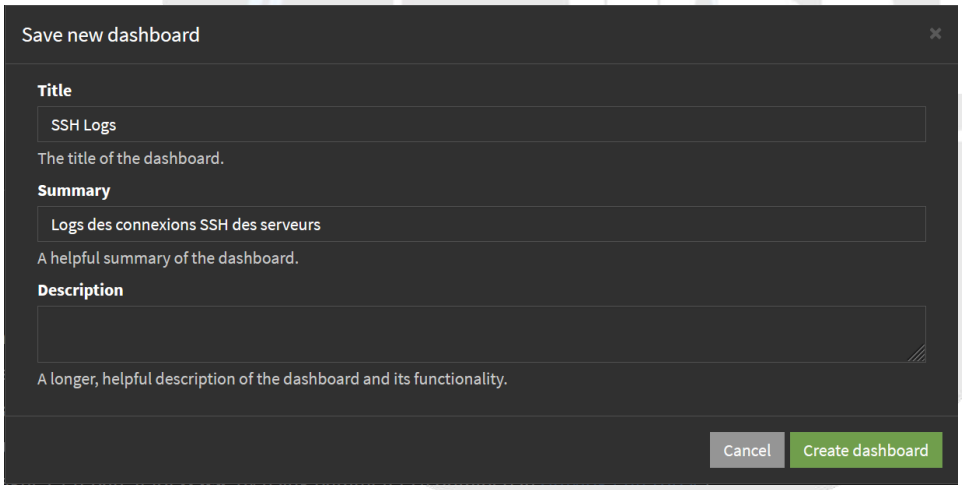
Dans la suite, nous ne détaillerons pas la mise en place d'un dashboard complet, du fait que le principal objectif était de mettre en place la centralisation des logs.

Toutefois, nous profiterons de cette fonctionnalité de Graylog pour en découvrir ses avantages, et donc nous créerons un dashboard pour les connexions SSH des serveurs.

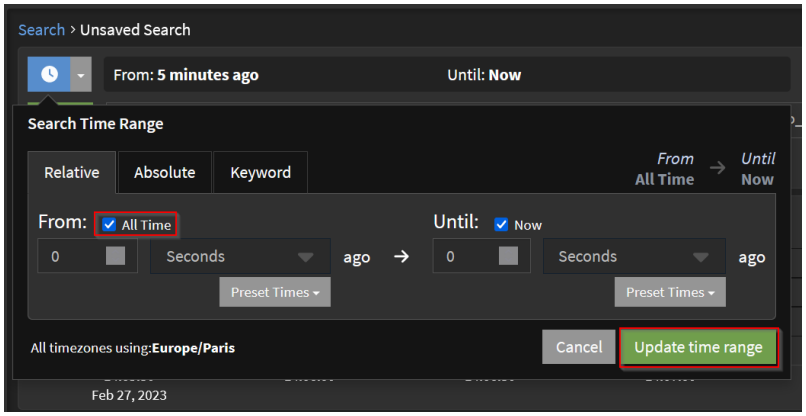
Sur l'interface web de Graylog, cliquez sur **Dashboards → Create new dashboard**

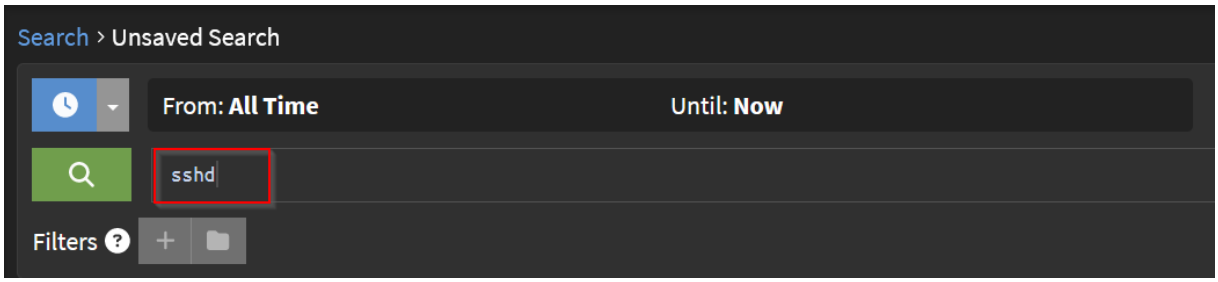


Ensuite, nommez votre dashboard, par exemple « **SSH Logs** » puis valider en cliquant sur **Create dashbord**

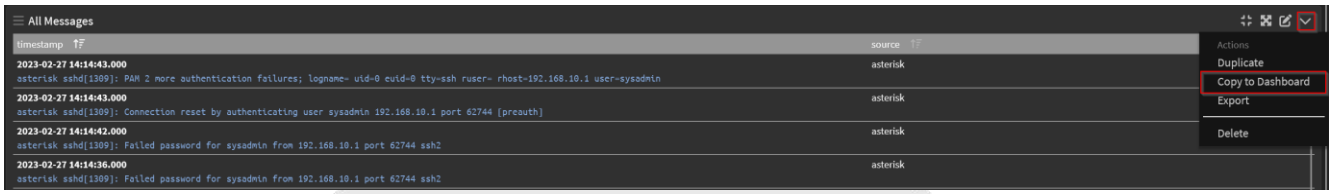


Puis, sur Search, effectuez une recherche des logs depuis la synchronisation, et rechercher le service sshd :

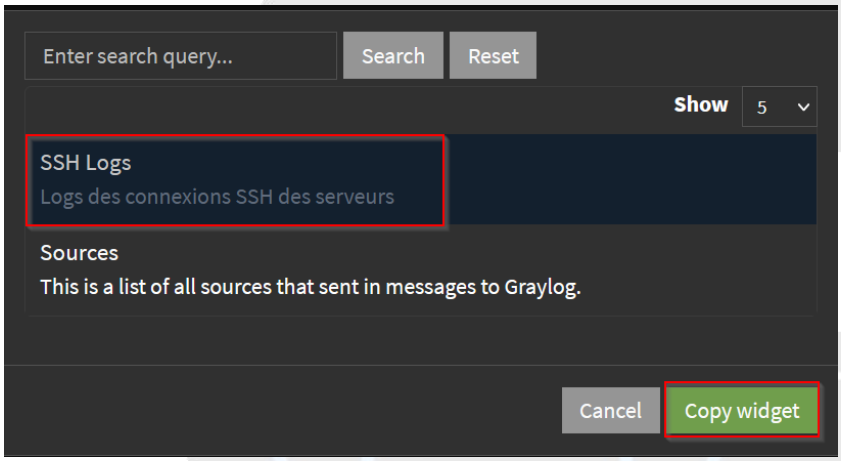




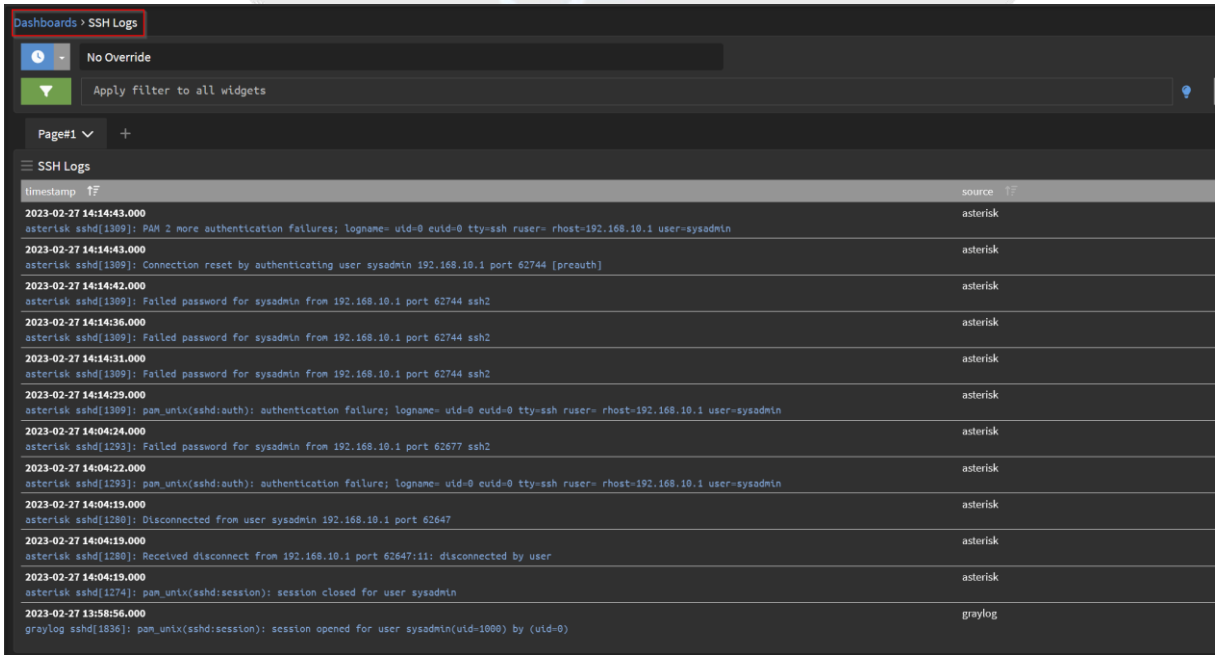
Copiez ensuite les données en cliquant sur la flèche puis **Copy to Dashboard** :



Sélectionnez **SSH Logs**, puis cliquez **Copy widget**



Maintenant lorsque vous voudriez savoir les logs sur le service SSH, vous pouvez directement allez sur **Dashboard → SSH Logs**



CONCLUSION

Pour conclure, Graylog est une solution complète, intuitif et lisible par les utilisateurs en terme de gestion des évènements sur un parc. Aussi, grâce à ses différentes fonctionnalités, l'administration des fichiers journaux se retrouve plus agréable que les solutions effectuées que par ligne de commande.

Dans cette documentation, nous nous sommes limités sur les logs des connexions SSH et Windows, mais pour avoir plus de données à traiter, il serait plus intéressant d'avoir un large parc avec différents services (des pares-feux, un serveur web, …) ou même simuler une attaque pirate sur l'un des serveurs, et d'en garder les traces des actes malveillantes.

Cela dit, l'objectif était de mettre en place une solution de centralisation de logs, avec une interface graphique pour faciliter son exploitation. Pour découvrir plus de fonctionnalités sur Graylog (messages d'alertes, analyse des anomalies, …), je vous invite à consulter leur documentation officielle, qui me semble très complète et dont nous nous sommes servi pour la mise en service du serveur ci-présent.