Cosnefroy Thibaut
BTS 2 SIO
30/03/2015

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

Conditions requises :

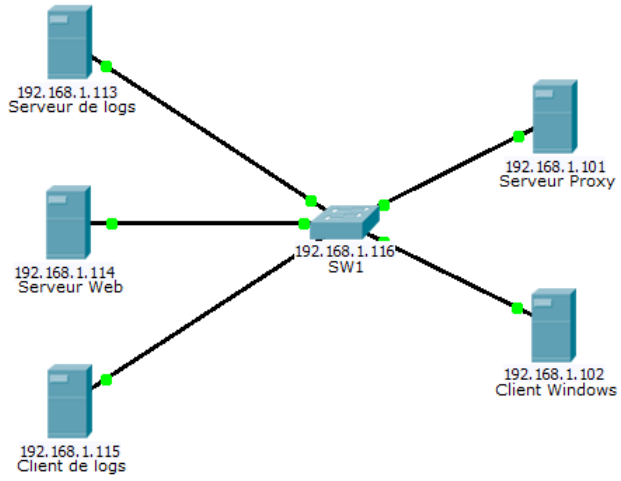
Dans ce TP, il est nécessaire d'avoir 4 machines Debian. Une pour le serveur de logs qui aura pour adresses IP 192.168.1.113, une pour le serveur Web qui aura pour adresse IP 192.168.1.114, une pour le client de logs qui aura pour adresse IP 192.168.1.115 et une pour le serveur Proxy qui aura pour adresse IP 192.168.1.101. Ensuite, il est nécessaire d'avoir une machine cliente Windows qui aura pour adresse IP 192.168.1.102. Pour finir, il est nécessaire d'avoir un Switch Cisco qui aura pour adresse IP 192.168.1.116.

Caractéristiques de bases :

La centralisation des logs de plusieurs serveurs sur un seul peut présenter un grand intérêt au niveau de la sécurité au sein d'un système d'information. En effet, il est plus facile pour des outils d'analyse de logs de comparer, lire et scanner des fichiers se situant sur un seul et unique serveur plutôt que de le faire à distance ou via des agents distants. De plus, en cas de crash serveur, il sera possible de récupérer les erreurs et actions menées sur notre serveur avant que celui-ci ne crash, facilitant ainsi la remise en activité de celui-ci et sa sécurisation future.
Syslog-ng est un gestionnaire de journaux systèmes de nouvelle génération (ng = new generation), capable de filtrer les messages en utilisant les expressions régulières, d'envoyer/recevoir les logs sur le réseau via UDP ou TCP et qui est compatible avec l'IPV6.
On utilisera l'interface graphique ELSA (Entreprise Log Search and Archive) qui utilise le gestionnaire de logs syslog-ng. Il n'est pas compatible avec rsyslog, le gestionnaire de logs installé automatiquement sur les nouvelles versions d'Ubuntu.

Contexte :

Mettre en place une centralisation de logs de différents serveurs.



Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

Pour commencer, on se rend sur le serveur de logs et on modifie les paramètres de la carte réseau :

```
root@debian:~# nano /etc/network/interfaces
```

```
  GNU nano 2.2.6        Fichier : /etc/network/interfaces                Modifié

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug eth0
iface eth0 inet static

address 192.168.1.113
netmask 255.255.255.0
gateway 192.168.1.254_


# This is an autoconfigured IPv6 interface
iface eth0 inet6 auto
```

Et on redémarre la carte réseau ensuite :

```
root@debian:~# ifdown eth0
root@debian:~# ifup eth0
```

Et on vérifie que l'adresse IP ait bien été affectée :

```
root@debian:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:da:79:70
          inet adr:192.168.1.113  Bcast:192.168.1.255  Masque:255.255.255.0
          adr inet6: fd23:6507:b29b:1:a00:27ff:feda:7970/64 Scope:Global
          adr inet6: fe80::a00:27ff:feda:7970/64 Scope:Lien
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:571 errors:0 dropped:1 overruns:0 frame:0
          TX packets:23 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:50651 (49.4 KiB)  TX bytes:2633 (2.5 KiB)

lo        Link encap:Boucle locale
          inet adr:127.0.0.1  Masque:255.0.0.0
          adr inet6: ::1/128 Scope:Hôte
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

root@debian:~# _
```

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

Ensuite, on passe à l'installation de syslog-ng et ELSA sur le serveur de logs.

ELSA est une solution de serveur de logs qui gère automatiquement l'installation du gestionnaire de logs « syslog-ng », de la base de données « MySQL » et l'interface web via un script.

Faire ces deux commandes :

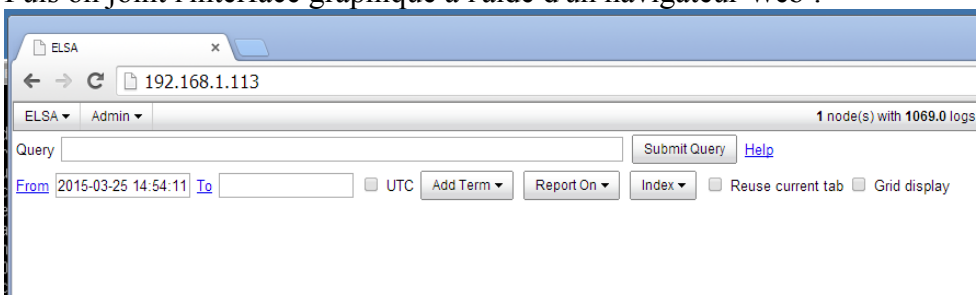
```
root@debian:~# wget http://enterprise-log-search-and-archive.googlecode.com/svn/trunk/elsa/contrib/install.sh
```

```
root@debian:~# sh -c "sh install.sh node" && sh -c "sh install.sh web"
```

Afin d'accéder à l'interface graphique d'ELSA, on installe d'abord Apache2 :

```
root@logs:~# apt-get install apache2
```

Puis on joint l'interface graphique à l'aide d'un navigateur Web :



ELSA est configurée nativement pour recevoir les logs par les protocoles TCP ou UDP. Sa configuration s'effectue à l'aide du fichier /usr/local/elsa/node/conf/syslog-ng.conf.

Ensuite, on configure l'envoi et la réception des logs. Pour ce faire, on configure un client Linux pour les logs système (client de logs).

On installe Apache2 dans un premier temps :

```
root@debian:~# apt-get install apache2
```

Puis on configure les paramètres de la carte réseau :

```
root@debian:~# nano /etc/network/interfaces
```

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

```
  GNU nano 2.2.6        Fichier : /etc/network/interfaces              Modifié

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug eth0

iface eth0 inet static

address 192.168.1.115
netmask 255.255.255.0
gateway 192.168.1.254

# This is an autoconfigured IPv6 interface
iface eth0 inet6 auto
```

On redémarre la carte réseau ensuite :

```
root@debian:~# ifdown eth0
root@debian:~# ifup eth0
```

On vérifie que l'adresse IP ait bien été affectée à la machine :

```
root@debian:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:f2:e1:69
          inet adr:192.168.1.115  Bcast:192.168.1.255  Masque:255.255.255.0
          adr inet6: fe80::a00:27ff:fef2:e169/64 Scope:Lien
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4191 errors:0 dropped:4 overruns:0 frame:0
          TX packets:2591 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:5046452 (4.8 MiB)  TX bytes:188149 (183.7 KiB)

lo        Link encap:Boucle locale
          inet adr:127.0.0.1  Masque:255.0.0.0
          adr inet6: ::1/128 Scope:Hôte
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:0
          RX bytes:240 (240.0 B)  TX bytes:240 (240.0 B)

root@debian:~# _
```

Puis on installe syslog-ng :

```
root@debian:~# apt-get install syslog-ng
```

On répond « o » pour continuer l'installation des paquets :

```
Souhaitez-vous continuer [O/n] ? o
```

On édite le fichier de configuration de syslog-ng :

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

```
root@debian:~# nano /etc/syslog-ng/syslog-ng.conf
```

Et on le modifie de la façon suivante :

```
  GNU nano 2.2.6      Fichier : /etc/syslog-ng/syslog-ng.conf          Modifié

};

########################
# Sources
########################
# This is the default behavior of sysklogd package
# Logs may come from unix stream, but not from another machine.
#
source s_src {
        internal();
        unix-stream("/dev/log");
        udp();_
};
```

* Seul l'onglet « source s_src » a été modifié sur la capture d'écran ci-dessus.

```
  GNU nano 2.2.6      Fichier : /etc/syslog-ng/syslog-ng.conf          Modifié
  GNU nano 2.2.6      Fichier : /etc/syslog-ng/syslog-ng.conf          Modifié

log { source(s_src); filter(f_console); destination(d_console_all);
                                        destination(d_xconsole); };
log { source(s_src); filter(f_crit); destination(d_console); };

# All messages send to a remote site
#
#log { source(s_src); destination(d_net); };

log { source(s_src); destination(d_elsa); };_

###
# Include all config files in /etc/syslog-ng/conf.d/
###
@include "/etc/syslog-ng/conf.d/"


destination d_daemon { file("/var/log/daemon.log"); };
destination d_kern { file("/var/log/kern.log"); };
```

* C'est la première ligne de destination qui a été rajoutée sur la prise d'écran ci-dessus.
* Ici, seul la ligne log soulignée d'une couleur a été rajoutée.

On redémarre ensuite le service syslog-ng afin de vérifier que le fichier ne renvoie aucune erreur et que les modifications que l'on vient d'apporter soient prises en compte :

```
root@debian:~# service syslog-ng restart
[ ok ] Stopping system logging: syslog-ng.
[ ok ] Starting system logging: syslog-ng.
root@debian:~# _
```

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

Désormais, on se dirige sur le serveur Web.
Tout d'abord, on installe Apache2 sur la machine :

```
root@debian:~# apt-get install apache2
```

Puis on configure les paramètres de la carte réseau :

```
root@debian:~# nano /etc/network/interfaces
```

```
  GNU nano 2.2.6        Fichier : /etc/network/interfaces              Modifié

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug eth0
iface eth0 inet static

address 192.168.1.114
netmask 255.255.255.0
gateway 192.168.1.254

# This is an autoconfigured IPv6 interface
iface eth0 inet6 auto
```

On redémarre la carte réseau ensuite :

```
root@debian:~# ifdown eth0
root@debian:~# ifup eth0
```

On vérifie que l'adresse IP ait bien été affectée à la machine :

```
root@debian:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:f2:e1:69
          inet adr:192.168.1.114  Bcast:192.168.1.255  Masque:255.255.255.0
          adr inet6: fe80::a00:27ff:fef2:e169/64 Scope:Lien
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:4414 errors:0 dropped:4 overruns:0 frame:0
          TX packets:55 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:5778807 (5.5 MiB)  TX bytes:5487 (5.3 KiB)

lo        Link encap:Boucle locale
          inet adr:127.0.0.1  Masque:255.0.0.0
          adr inet6: ::1/128 Scope:Hôte
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:0
          RX bytes:240 (240.0 B)  TX bytes:240 (240.0 B)

root@debian:~# _
```

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

Puis on installe syslog-ng :

```
root@debian:~# apt-get install syslog-ng
```

On édite le fichier de configuration de syslog-ng :

```
root@debian:~# nano /etc/syslog-ng/syslog-ng.conf
```

Et on le modifie de la façon suivante :

```
  GNU nano 2.2.6      Fichier : /etc/syslog-ng/syslog-ng.conf          Modifié

};

# If you wish to get logs from remote machine you should uncomment
# this and comment the above source line.
#
#source s_net { tcp(ip(127.0.0.1) port(1000) authentication(required) encrypt(a$

source s_apache2_error.log {file("/var/log/apache2/error.log" program_override $
source s_apache2_access.log {file("/var/log/apache2/access.log" program_overrid$
```

* Ce sont les deux sources qui ont été rajoutées ici.

```
  GNU nano 2.2.6      Fichier : /etc/syslog-ng/syslog-ng.conf          Modifié

};

# If you wish to get logs from remote machine you should uncomment
# this and comment the above source line.
#
#source s_net { tcp(ip(127.0.0.1) port(1000) authentication(required) encrypt(a$

$rride ("url"));};
source s_apache2_access.log {file("/var/log/apache2/access.log" program_overrid$
```

* Ceci correspond à la fin de la première ligne des sources ajoutées.

```
  GNU nano 2.2.6      Fichier : /etc/syslog-ng/syslog-ng.conf          Modifié

};

# If you wish to get logs from remote machine you should uncomment
# this and comment the above source line.
#
#source s_net { tcp(ip(127.0.0.1) port(1000) authentication(required) encrypt(a$

source s_apache2_error.log {file("/var/log/apache2/error.log" program_override $
$verride ("url"));};
```

* Ceci correspond à la fin de la deuxième ligne des sources ajoutées.

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

```
  GNU nano 2.2.6      Fichier : /etc/syslog-ng/syslog-ng.conf

};

# If you wish to get logs from remote machine you should uncomment
# this and comment the above source line.
#
#source s_net { tcp(ip(127.0.0.1) port(1000) authentication(required) encrypt(a$

source s_apache2_error.log {file("/var/log/apache2/error.log" program_override $
source s_apache2_access.log {file("/var/log/apache2/access.log" program_overrid$

########################
# Destinations
########################
# First some standard logfile
#

destination d_elsa {udp("192.168.1.113");};

destination d_auth { file("/var/log/auth.log"); };
destination d_cron { file("/var/log/cron.log"); };
```

* Seul la ligne destination de couleur a été rajoutée sur la prise d'écran ci-dessus.

```
  GNU nano 2.2.6      Fichier : /etc/syslog-ng/syslog-ng.conf          Modifié

log { source(s_src); filter(f_messages); destination(d_messages); };

log { source(s_src); filter(f_console); destination(d_console_all);
                                        destination(d_xconsole); };
log { source(s_src); filter(f_crit); destination(d_console); };

# All messages send to a remote site
#
#log { source(s_src); destination(d_net); };

log { source(s_apache2_error.log); destination(d_elsa);};
log { source(s_apache2_access.log); destination(d_elsa);};_

###
# Include all config files in /etc/syslog-ng/conf.d/
###
@include "/etc/syslog-ng/conf.d/"
```

* Seul les deux lignes log de couleur ont été rajoutées ici.

On redémarre ensuite le service syslog-ng afin de vérifier que le fichier ne renvoie aucune erreur et que les modifications que l'on vient d'apporter soient prises en compte :

```
root@debian:~# service syslog-ng restart
[ ok ] Stopping system logging: syslog-ng.
[ ok ] Starting system logging: syslog-ng.
root@debian:~# _
```

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

A présent, on se rend sur le serveur Proxy. Tout d'abord, on installe Apache2 sur la machine :

```
root@debian:~# apt-get install apache2
```

Puis on configure les paramètres de la carte réseau :

```
root@debian:~# nano /etc/network/interfaces
```

```
  GNU nano 2.2.6        Fichier : /etc/network/interfaces                Modifié

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug eth0
iface eth0 inet static
address 192.168.1.101
netmask 255.255.255.0
gateway 192.168.1.254

_

# This is an autoconfigured IPv6 interface
iface eth0 inet6 auto
```

On redémarre la carte réseau ensuite :

```
root@debian:~# ifdown eth0
root@debian:~# ifup eth0
```

On vérifie que l'adresse IP ait bien été affectée à la machine :

```
root@debian:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:42:56:8c
          inet adr:192.168.1.101  Bcast:192.168.1.255  Masque:255.255.255.0
          adr inet6: fe80::a00:27ff:fe42:568c/64 Scope:Lien
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:18 errors:0 dropped:0 overruns:0 frame:0
          TX packets:204 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:1000
          RX bytes:1318 (1.2 KiB)  TX bytes:13776 (13.4 KiB)

lo        Link encap:Boucle locale
          inet adr:127.0.0.1  Masque:255.0.0.0
          adr inet6: ::1/128 Scope:Hôte
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:52 errors:0 dropped:0 overruns:0 frame:0
          TX packets:52 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 lg file transmission:0
          RX bytes:4176 (4.0 KiB)  TX bytes:4176 (4.0 KiB)

root@debian:~# _
```

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

Puis on installe syslog-ng :

```
root@debian:~# apt-get install syslog-ng
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  libdbi1 libevtlog0 libglib2.0-0 libglib2.0-data libjson0 libmongo-client0
  libnet1 libsyslog-ng-3.3.5 libsystemd-daemon0 shared-mime-info
  syslog-ng-core syslog-ng-mod-json syslog-ng-mod-mongodb syslog-ng-mod-sql
Paquets suggérés :
  mongodb-server libdbd-mysql libdbd-pgsql libdbd-sqlite3
Les paquets suivants seront ENLEVÉS :
  rsyslog
Les NOUVEAUX paquets suivants seront installés :
  libdbi1 libevtlog0 libglib2.0-0 libglib2.0-data libjson0 libmongo-client0
  libnet1 libsyslog-ng-3.3.5 libsystemd-daemon0 shared-mime-info syslog-ng
  syslog-ng-core syslog-ng-mod-json syslog-ng-mod-mongodb syslog-ng-mod-sql
0 mis à jour, 15 nouvellement installés, 1 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 4 749 ko dans les archives.
Après cette opération, 17,3 Mo d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer [O/n] ? o_
```

On édite le fichier de configuration de syslog-ng :

```
root@debian:~# nano /etc/syslog-ng/syslog-ng.conf_
```

Et on le modifie de la façon suivante :

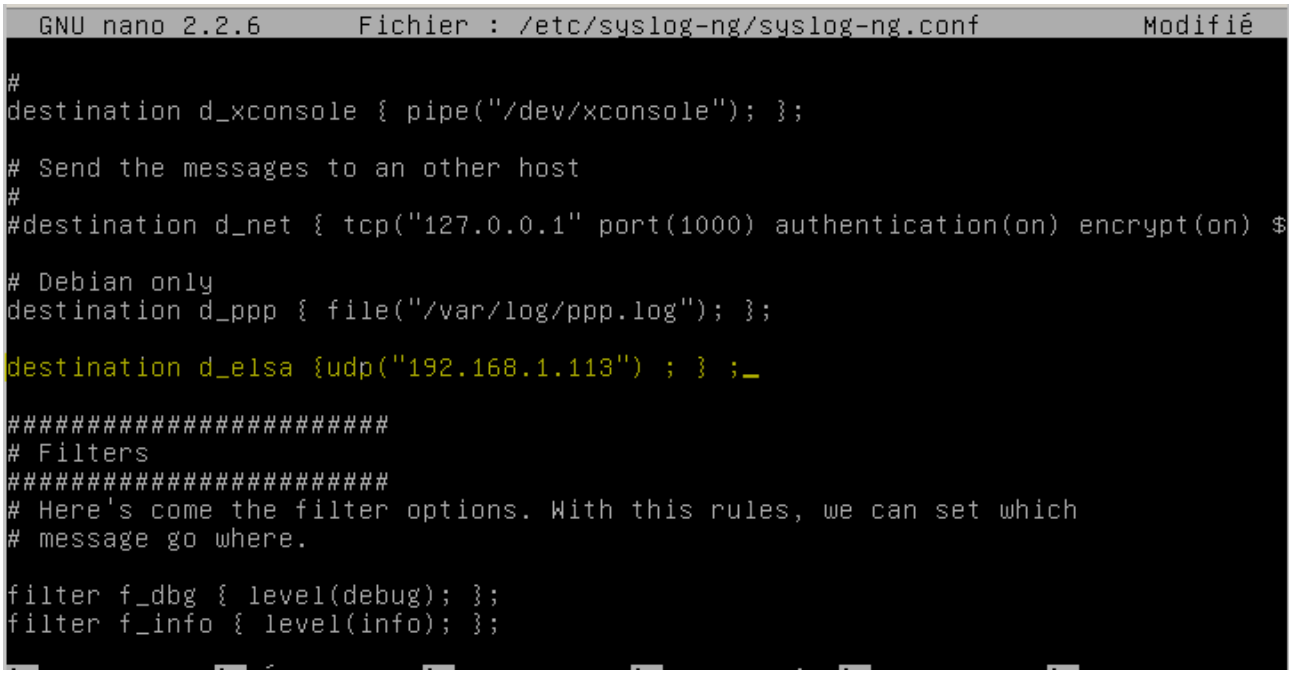
```
  GNU nano 2.2.6      Fichier : /etc/syslog-ng/syslog-ng.conf          Modifié

};

# If you wish to get logs from remote machine you should uncomment
# this and comment the above source line.
#
#source s_net { tcp(ip(127.0.0.1) port(1000) authentication(required) encrypt(a

source s_squid_cache.log {file("/var/log/squid3/cache.log" program_override ("s
source s_squid_access.log {file("/var/log/squid3/access.log" program_override (
_
########################
# Destinations
########################
# First some standard logfile
#
destination d_auth { file("/var/log/auth.log"); };
destination d_cron { file("/var/log/cron.log"); };
destination d_daemon { file("/var/log/daemon.log"); };
destination d_kern { file("/var/log/kern.log"); };
destination d_lpr { file("/var/log/lpr.log"); };
```

* Seul les deux lignes source de couleur ont été ajoutées ici.
* A la fin de ces deux lignes juste après program_override, ajouter ceci :

```
("squid")) ; } ;
```

Centralisation des logs (debian 7.7)

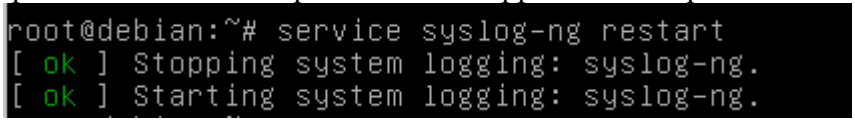
Version 1.0(01/04/2015)



* On ajoute la ligne destination de couleur sur la prise d'écran ci-dessus en renseignant l'adresse IP du serveur de logs.

* On ajoute les lignes log en couleur sur la prise d'écran ci-dessus.

On redémarre ensuite le service syslog-ng afin de vérifier que le fichier ne renvoie aucune erreur et que les modifications que l'on vient d'apporter soient prises en compte :



Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

Ensuite, on installe Squid :

```
root@debian:~# apt-get install squid3_
```

Puis on se dirige dans le répertoire /squid3 pour copier le fichier de configuration de Squid afin d'avoir un back-up par précaution :

```
root@debian:~# cd /etc/squid3/
root@debian:/etc/squid3# ls
errorpage.css  msntauth.conf  squid.conf
root@debian:/etc/squid3# cp squid.conf squid.conf.olde
root@debian:/etc/squid3# _
```

Ensuite, effectuer cette commande :

```
root@debian:/etc/squid3# cat squid.conf.olde | grep -v ^# | grep -v ^$ > squid.c
onf_
```

Puis, on édite le fichier de configuration de Squid et on ajoute à la fin de ce fichier les lignes figurant sur la prise d'écran ci-dessous :

```
root@debian:/etc/squid3# nano squid.conf_
```

```
  GNU nano 2.2.6              Fichier : squid.conf                        Modifié

http_port 3128
coredump_dir /var/spool/squid3
refresh_pattern ^ftp:           1440    20%     10080
refresh_pattern ^gopher:        1440    0%      1440
refresh_pattern -i (/cgi-bin/|\?) 0     0%      0
refresh_pattern .               0       20%     4320

#Utlisateur faisant les requêtes sur le serveur
cache_effective_user proxy
cache_effective_group proxy

#Emplacement de stockage des données et réglage des niveaux
cache_mem 16 MB
cache_dir ufs /var/spool/squid3 120 16 128_
```

* Les lignes rajoutées commence à partir du commentaire « # Utilisateur faisant les requêtes sur le serveur » jusqu'à la fin.

Puis on installe le paquet ACL :

```
root@debian:/etc/squid3# apt-get install acl_
```

On édite de nouveau le fichier de configuration de Squid et on ajoute à la fin de ce fichier les lignes figurant sur la prise d'écran ci-dessous :

```
root@debian:~# nano /etc/squid3/squid.conf_
```

```
acl lan src 192.168.1.113/24

#ajout du drioit au dessus des autres http-access
http_access allow lan_
```

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

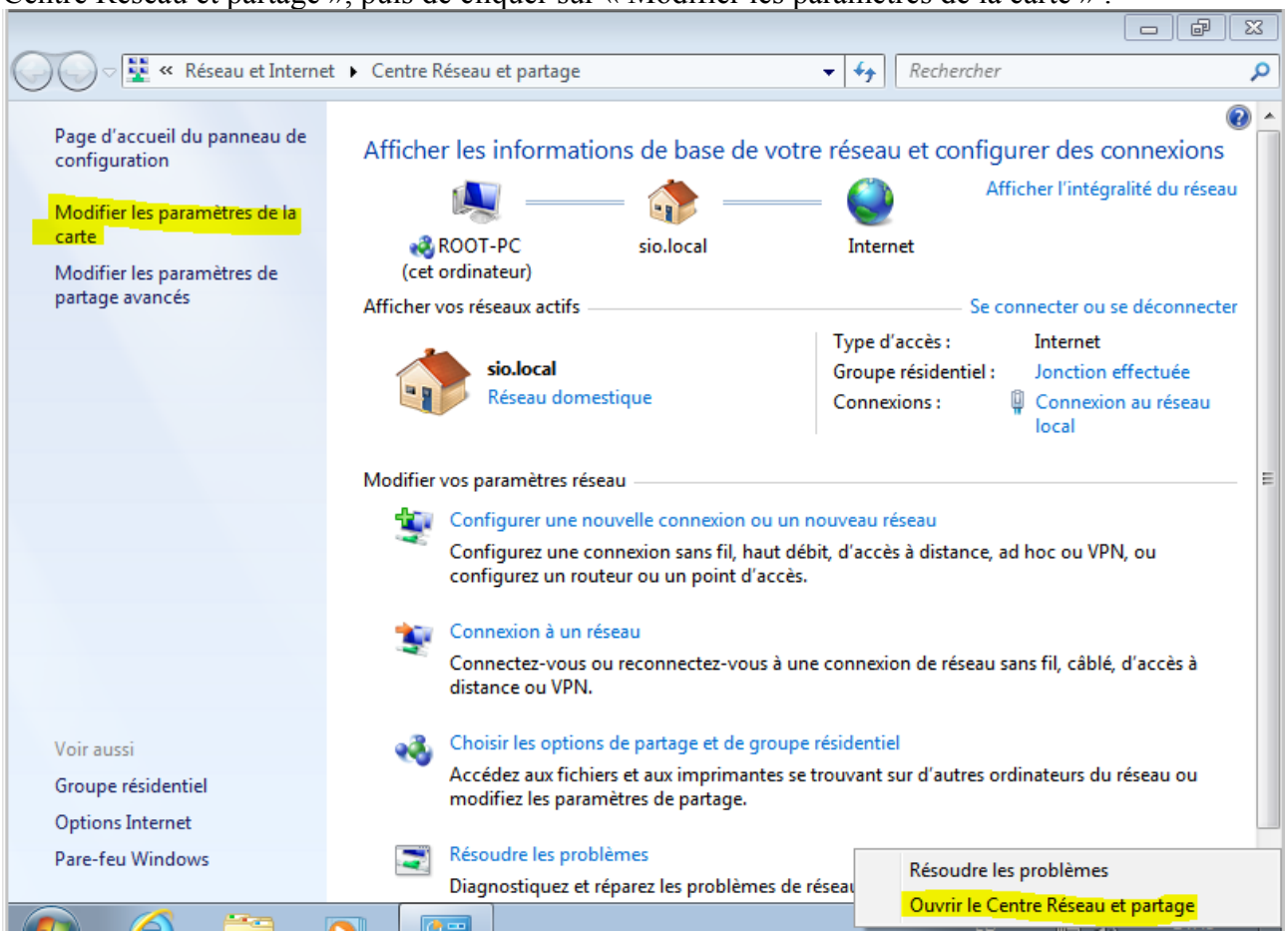
On redémarre ensuite le service squid3 afin de vérifier que le fichier ne renvoie aucune erreur et que les modifications que l'on vient d'apporter soient prises en compte :

```
root@debian:~# service squid3 restart
[....] Restarting Squid HTTP Proxy 3.x: squid32015/03/31 14:47:52| aclIpParseIpD
ata: WARNING: Netmask masks away part of the specified IP in '192.168.1.113/24'
. ok
root@debian:~# _
```

Désormais, on se rend sur la machine cliente Windows.

On va modifier les paramètres IP de la machine.

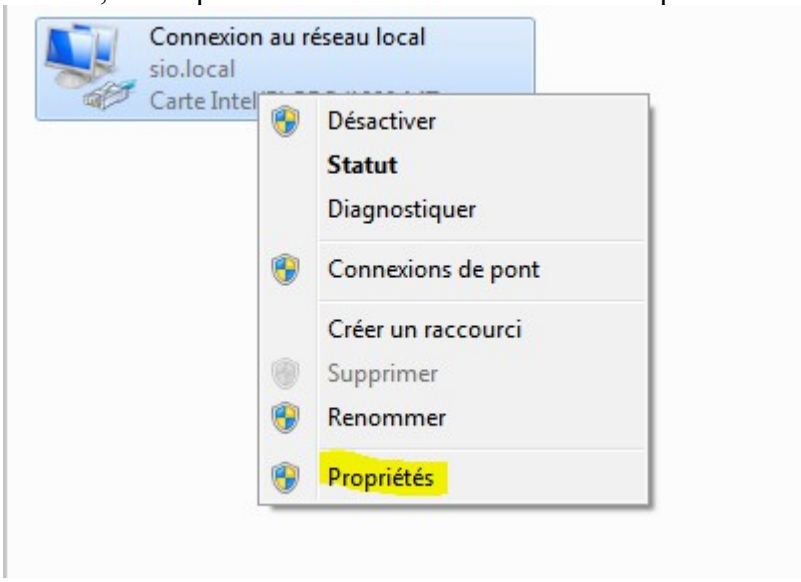
Il s'agit de faire un clique droit en bas à droite de l'écran sur l'icône réseau, de cliquer sur « Ouvrir le Centre Réseau et partage », puis de cliquer sur « Modifier les paramètres de la carte » :



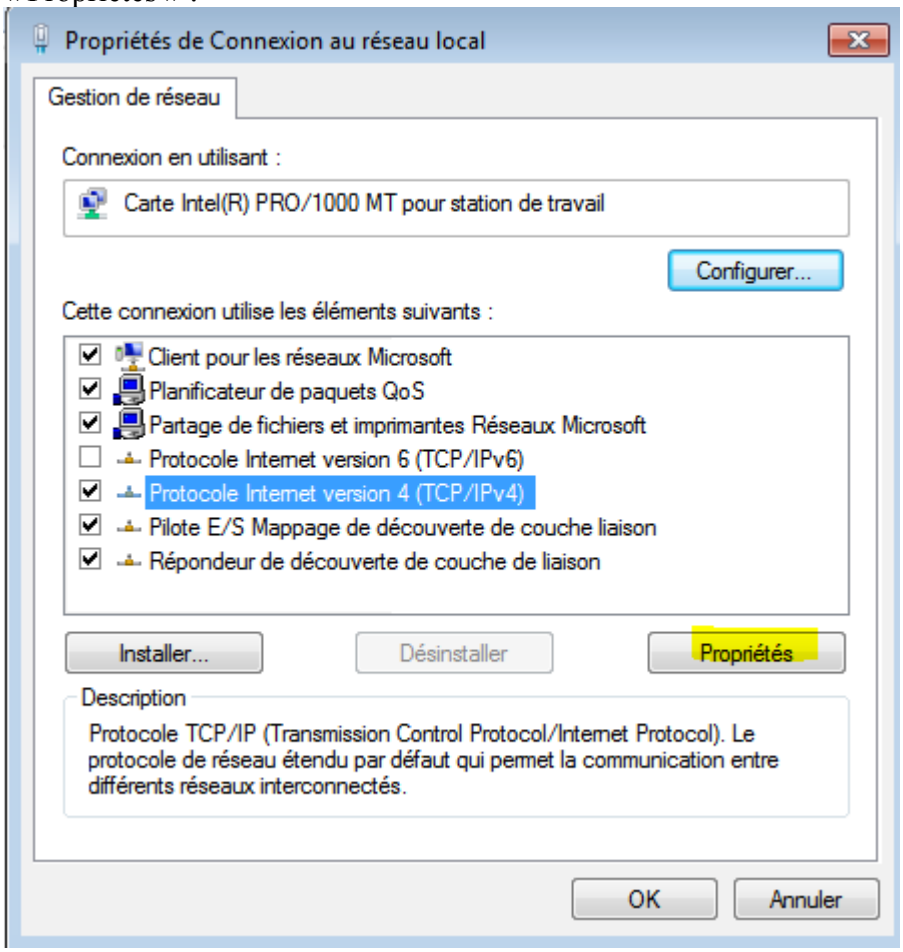
Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

Ensuite, on clique droit sur la carte réseau et on clique sur « Propriétés » :



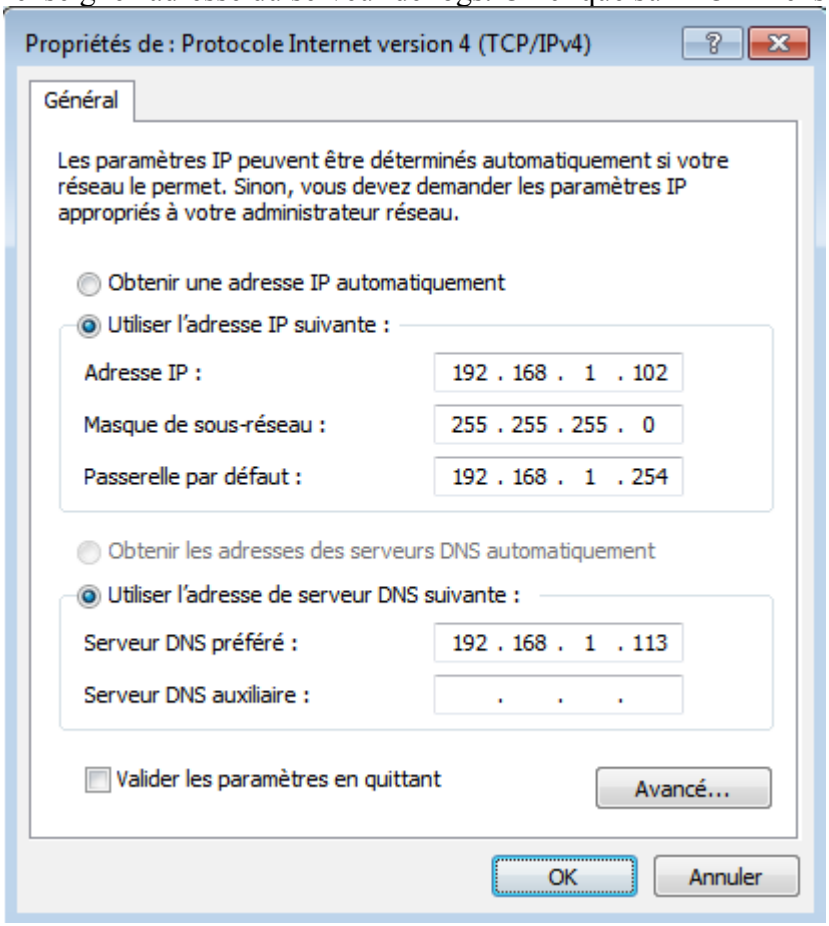
On décoche la case « TCP/IPv6 » et on clique sur « Protocole Internet version 4 » et sur « Propriétés » :



Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

On affecte ici une adresse IP, un masque de sous-réseau, une passerelle par défaut et en DNS, on renseigne l'adresse du serveur de logs. On clique sur « OK » ensuite.



Toujours sur la machine cliente Windows, on ouvre une invite de commande en tapant « cmd » dans la barre de recherche du menu démarrer.
On tape la commande « cd c:\Users\root\Desktop\Evtsys_4.5.1_64-Bit\64-Bit ».

On entre ensuite la commande ci-dessous pour une installation simple et rapide :

```
C:\Users\root\Desktop\Evtsys_4.5.1_64-Bit\64-Bit>evtsys.exe -i -a -h 192.168.1.1
13
```

Voici la signification des paramètres :

-i : permet l'installation du service
-a : permet l'utilisation de l'adresse IP ou le nom FQDN dans les logs Windows
-h : définit le serveur de logs

L'adresse IP renseignée correspond à celle du serveur de logs.

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

Maintenant, on passe à la configuration des logs pour un Switch Cisco.

Il s'agit dans un premier temps d'affecter une adresse IP au Switch. Pour ce faire, on se rend sur l'interface du Vlan 1 qui est le Vlan d'administration par défaut d'un Switch et on lui affecte donc une adresse IP :

```
Switch(config)#interface vl
Switch(config)#interface vlan 1
Switch(config-if)#ip add
Switch(config-if)#ip address 192.168.1.116 255.255.255.0
Switch(config-if)#exit
Switch(config)#
```

Puis toujours sur l'interface du Vlan 1, on l'active :

```
Switch(config)#interface vlan 1
Switch(config-if)#no sh
```

Ensuite, faire cette commande :

```
Switch(config)#logging trap
```

On configure l'étiquette associée à chaque message (ici local7) ainsi que l'adresse IP du serveur de logs :

```
Switch(config)#logging facility local7
Switch(config)#logging 192.168.1.113
```

Maintenant, on passe à la mise en place de la rotation des logs avec logrotate.

Logrotate est un outil de gestion de fichiers de logs. Il permet d'archiver, d'organiser et de sauvegarder les journaux systèmes automatiquement.

L'outil est essentiellement composé d'un script de rotation des logues (logrotate) et de ses fichiers de configuration (« /etc/logrotate.conf » et « /etc/logrotate.d/* »).

Le déclenchement du script est effectué par le cron. Lorsque le script est appelé, il examine les fichiers de logues qui ont été spécifié dans « /etc/logrotate.conf » ou « /etc/logrotate.d/* », et y applique le traitement définit dans le fichier de configuration (compression, numérotation, archivage, etc..).

On installe logrotate :

```
root@logs:~# apt-get install logrotate
```

La configuration de logrotate est scindée en deux parties :

- les options générales ;
- les fichiers à archiver et les options spécifiques à leurs archivages.

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

Le fichier de configuration de logrotate est quasiment toujours « /etc/logrotate.conf ».

Dans les options les plus utilisées, nous trouvons :

- **compress** : impose la compression des anciens fichiers de log au format gzip. **compresscmd** permet de définir la commande à exécuter pour la compression. **coompressext**, **uncompresscmd** et **compressoptions** permettent respectivement de définir l'extension des fichiers compressés, la commande utilisée pour décompresser les archives et les options de compressions à passer à l'outil de compression.

- **copytruncate** : copie le fichier de log original, le compresse et vide ensuite le journal système d'origine. Cette notion est très importante car elle répond aux problèmes du type : « je logue dans le fichier que j'archive.. » qui peut poser quelques problèmes à certains services.

- **create** : Permet de spécifier les droits, le propriétaire et le groupe auquel un nouveau fichier de log vide devra appartenir après avoir été archivé.

- **daily**, **weekly**, **monthly** et **yearly** : spécifie quand un fichier doit être arcgivé.

- **dateext** : par défaut, les fichiers archivés sont numérotés de 1 à N, l'option **dateext** remplace cette numérotation par la date. Si le format de date par défaut ne vous convient pas, il peut être spécifié par l'option « **dateformat** ».

- **ifempty** : oblige la rotation des logs même si le fichier est vide.. (ce qui simplifie les recherches parfois..).

- **mailfirst**, **maillast**, **mail** **adresse@domain** : après un certain délai, les archives peuvent être automatiquement détruites. Dans ce cas, si maillast est définit, le fichier est détruit est envoyé par mail à l'adresse spécifiée par la variable « mail ». Si mailfirst est définit, c'est la dernière rotation qui est expédiée.

- **maxage** : définit l'âge maximum des archives (en jours).

- **minsize** : demande la rotation des logs si le fichier fait au minimum la taille définie par cette variable.

- **missingok** : aucune erreur n'est remontée si la rotation d'un fichier de log spécifié est absent.

- **notifempty** : n'archive aucun fichier vide.

- **size** : les fichiers ne sont archivés que si leur taille dépasse la valeur définie ici.

- **olddir** : précise le répertoire dans lequel placer les archives.

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

- **prerotate, postrotate/endscript** : définissent respectivement une séquence de script à effectuer avant ou après la rotation des logs (redémarrage d'un service, etc).

- **firstaction, lastaction/endscript** : définissent des séquences de scripts à exécuter avant et après avoir archivé des logs.

- **rotate** : spécifie le nombre d'archives à conserver. Passé ce nombre, les archives sont soit détruite (comportement par défaut), soit envoyées par mail voir mail* ci-dessus.

On édite donc le fichier de configuration de logrotate :

```
root@logs:~# nano /etc/logrotate.conf
```

Et on le modifie de la façon suivante :

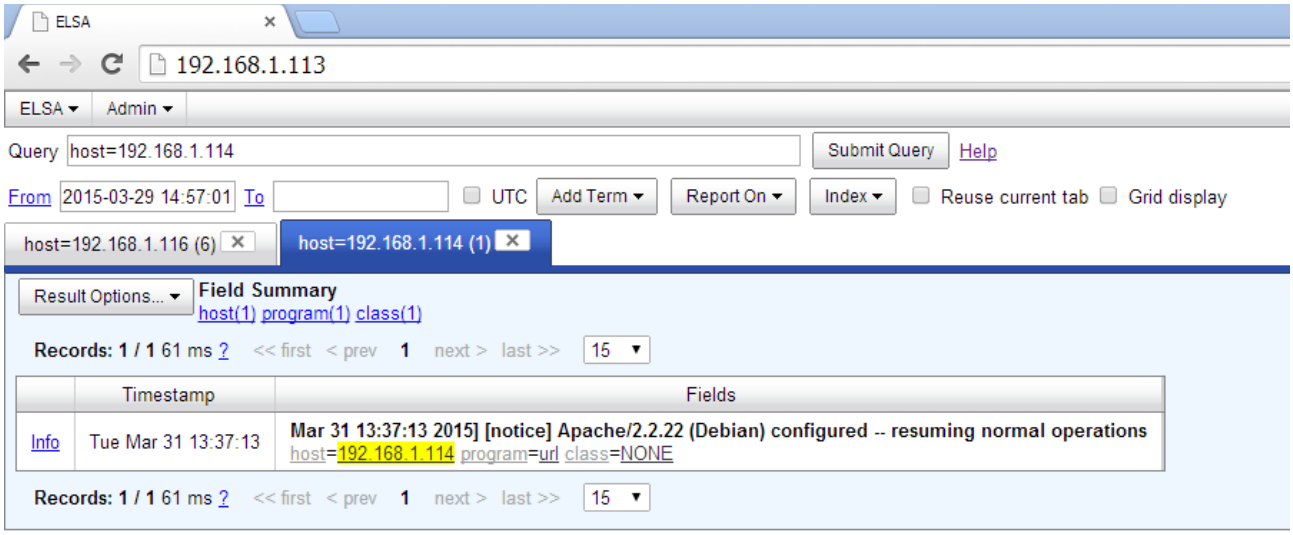
```
  GNU nano 2.2.6                          Fichier : /etc/logrotate.conf

# see "man logrotate" for details
# rotate log files weekly
weekly

# keep 4 weeks worth of backlogs
rotate 4

# create new (empty) log files after rotating old ones
create

# uncomment this if you want your log files compressed
compress

# Frequence de rotation par défaut
daily
# La rotation est effectuée, par défaut, même si le fichier est vide afin de faciliter ifempty

# Les recherches dans les logs au jour le jour
ifempty

# Par défaut, une rotation par jour est effectuée. 365,25 jours par ans, nous conservons donc une profondeur de 366 jours
rotate 366

# Envois par mail du fichier sur le point d'expirer
maillast

# Le fichier sur le point d'expirer est envoyé à l'adresse définie ici
mail fanfan@thibi.jb

# Si l'un des journaux décrit est manquant, ancune erreur n'est remontée
missingok

# Insertion de la date dans l'archivage des journaux
dateext
# Lors de l'archivage de plusieurs fichiers de logues, les scripts pre et post rotate sont executé une fois par fichier

# L'option ci-dessous permet de fixer l'execution de ces scripts à une seule fois
sharedscripts

# Lors d'une rotation, un fichier de remplacement est créé avec les droits 0640 appartenant à root et au groupe root
create 0640 root root

# Toutes les archives sont stockées dans /var/log/archives
olddir /var/log/archives

# Définition de la rotation des logs d'argus
/var/log/argus/argus.log {
        olddir /var/log/archives/argus
}

# Inclusion des scripts propres aux logs
include /etc/logrotate.d
```

Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

```
# packages drop log rotation information into this directory
include /etc/logrotate.d

# no packages own wtmp, or btmp -- we'll rotate them here
/var/log/wtmp {
    missingok
    monthly
    create 0664 root utmp
    rotate 1
}

/var/log/btmp {
    missingok
    monthly
    create 0660 root utmp
    rotate 1
}

# system-specific logs may be configured here
```

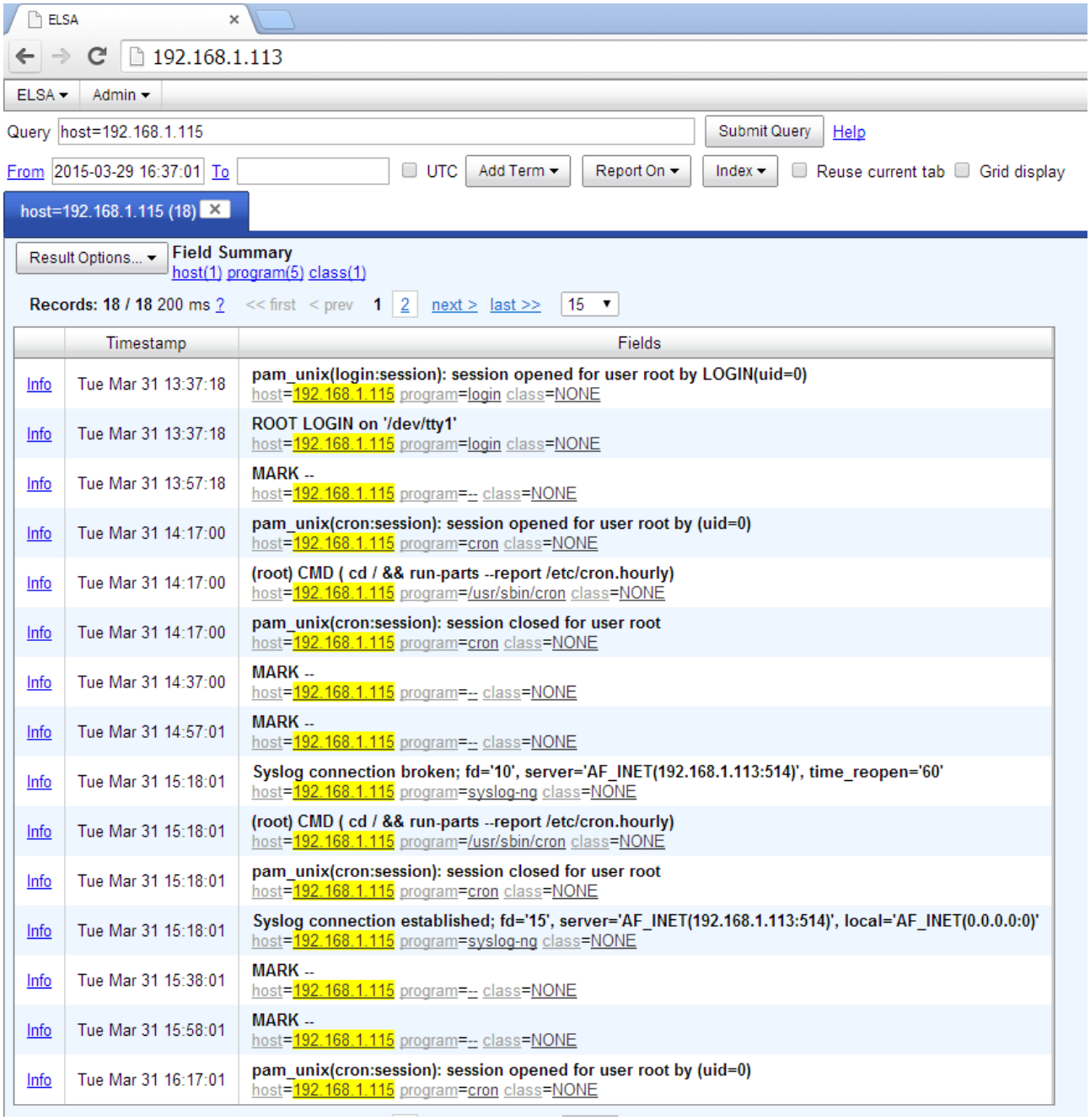
A présent, on teste de centraliser les logs du serveur Web en tapant dans le champ Query « host=192.168.1.114 » :



Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

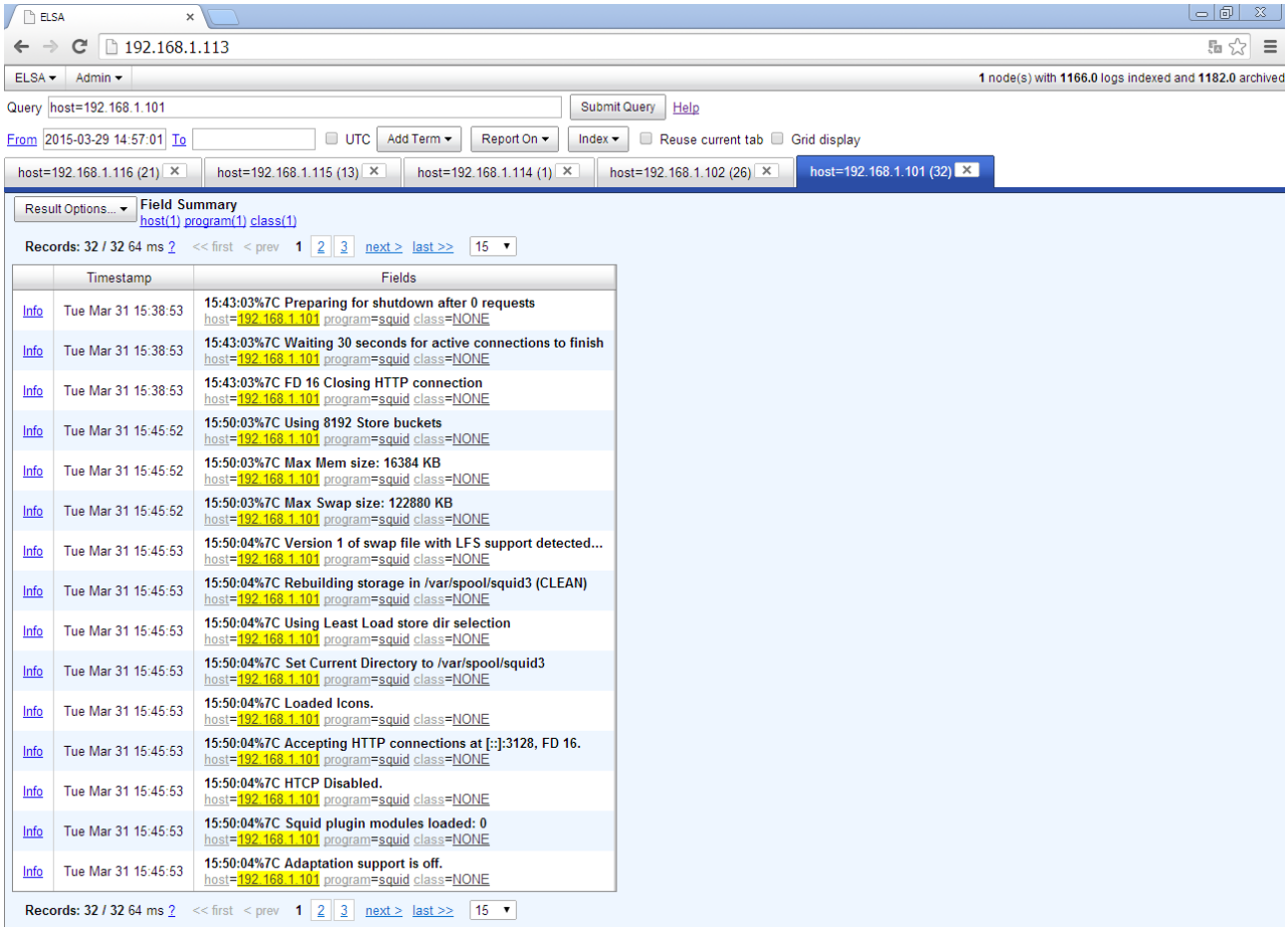
On teste de centraliser les logs du client de logs en tapant dans le champ Query « host=192.168.1.115 » :



Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

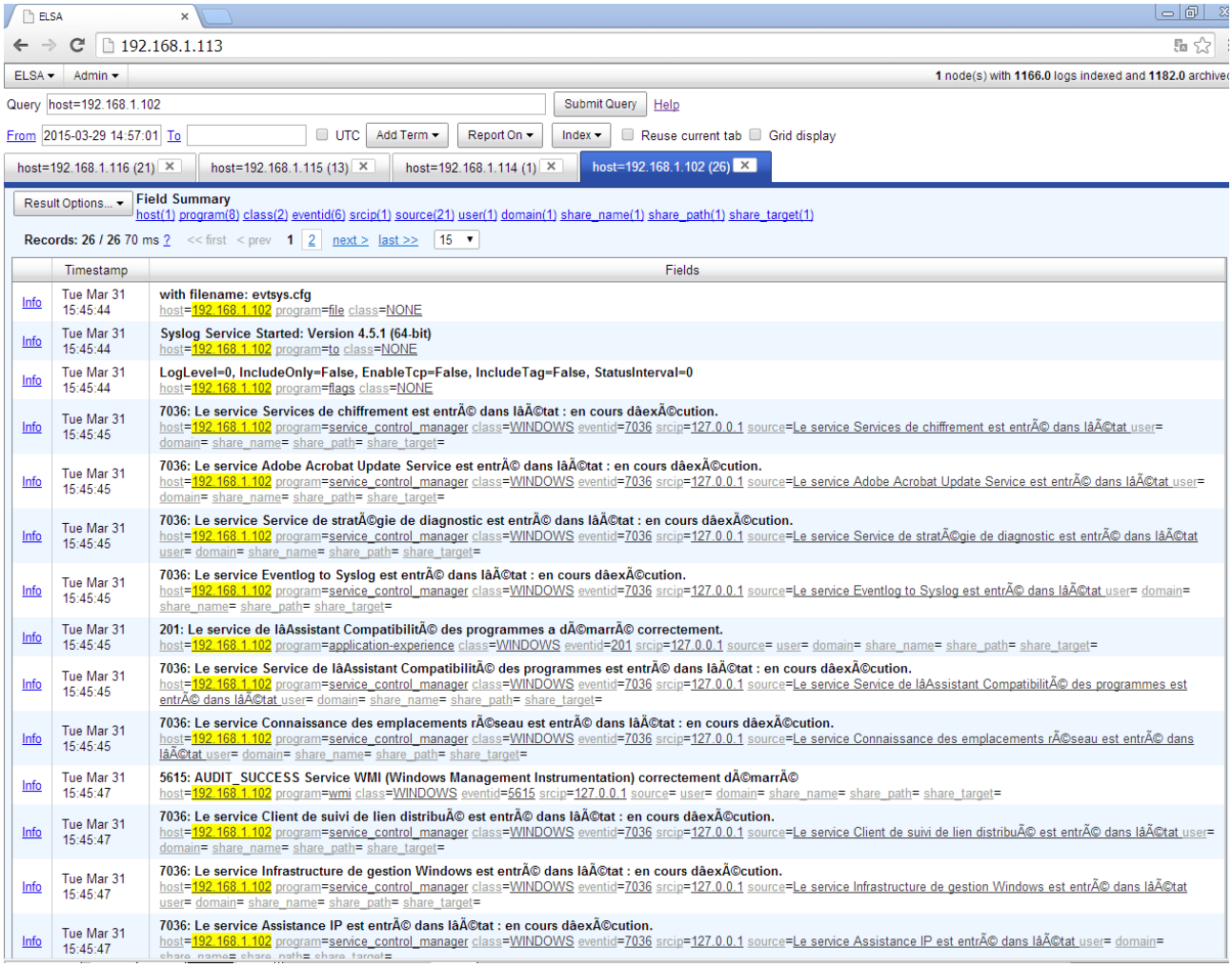
On teste de centraliser les logs du serveur Proxy en tapant dans le champ Query « host=192.168.1.101 » :



Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

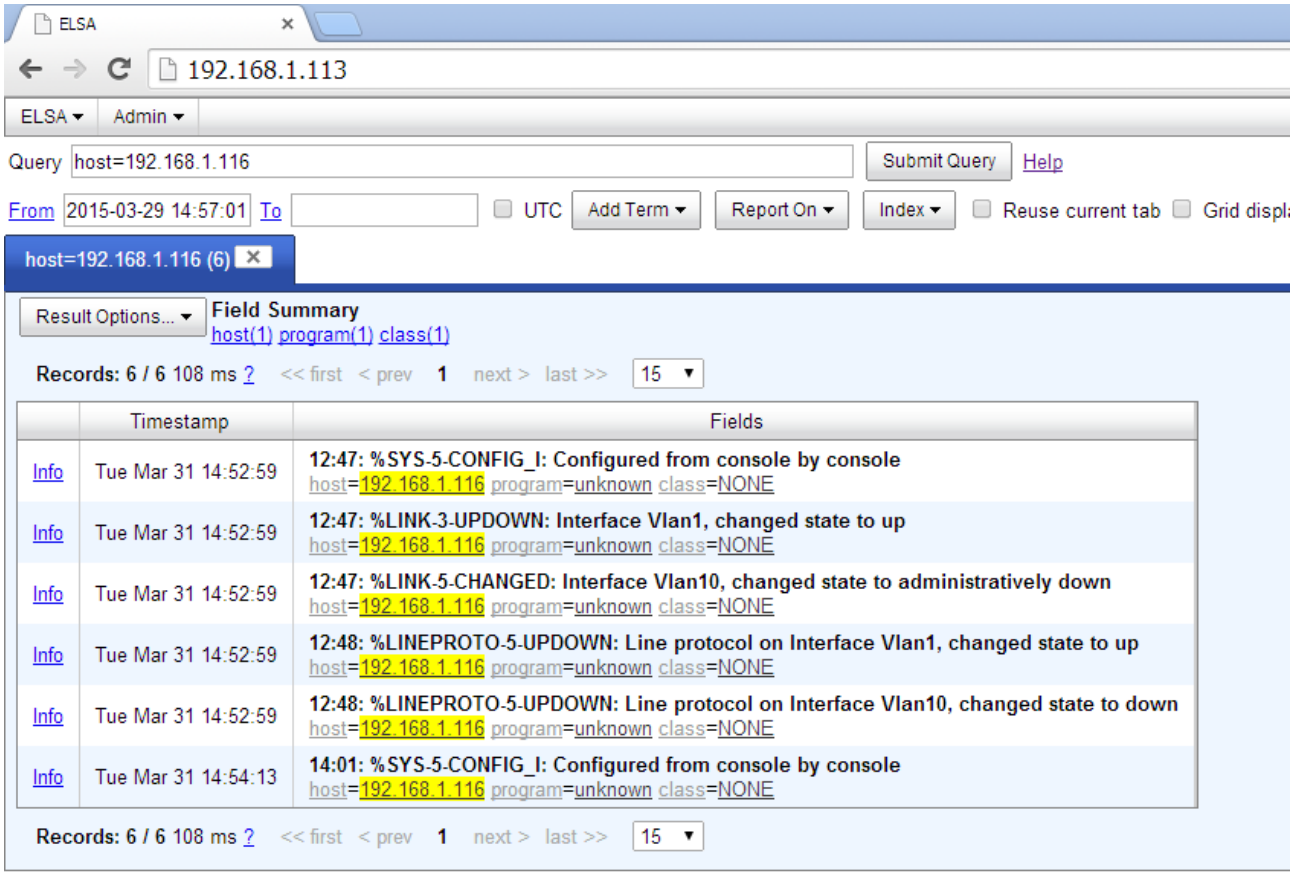
On teste de centraliser les logs du client Windows en tapant dans le champ Query « host=192.168.1.102 » :



Centralisation des logs (debian 7.7)

Version 1.0(01/04/2015)

On teste de centraliser les logs du Switch Cisco en tapant dans le champ Query « host=192.168.1.116 » :



Conclusion :

Le serveur de centralisation des logs est opérationnel. Nous pouvons bien centraliser les logs de tous les services paramétrés ici.