
www.cnrs.fr

# Gestion des traces : les pratiques de la DR7

Ernest CHIARELLO
Grégory HAKON

Plan :

1) Rappel sur la réglementation CNRS
https://intranet.cnrs.fr/extranet/cnrs/fsd/documents/Po_gest_traces.pdf

2) Centralisation des traces avec syslog-ng

3) La rotation des traces avec logrotate

4) L'analyse des traces avec logwatch et logcheck

Objectif : assurer la sécurité et le contrôle de l'utilisation des moyens télématiques et informatiques

Contrainte : garantir les droits individuels de chaque agent

La politique de gestion des traces doit aider les responsables du traitement informatique à s'assurer que :

- La collecte des informations n'est ni frauduleuse, ni déloyale, ni illicite ; elle s'accompagne d'une bonne information des personnes ;
- Les informations ne sont pas conservées au-delà de la durée prévue ;
- Les informations ne sont pas communiquées à des personnes non autorisées ;
- Le traitement ne fait pas l'objet d'un détournement de finalité ;
- L'accès aux résultats des traitements et aux données collectées fait l'objet d'une sécurité optimale, afin qu'aucun détournement de la finalité ne puisse avoir lieu ;
- Les applications à caractère nominatif font l'objet de demandes d'avis préalables à la CNIL.

La collecte des traces a plusieurs objectifs :

- La métrologie du réseau : contrôler le volume d'utilisation de la ressource ;
- détecter des anomalies afin de mettre en place de la qualité de service, faire évoluer les équipements en fonction des besoins ;
- Vérifier que les règles en matière de SSI sont correctement appliquées et que la sécurité des systèmes d'information et du réseau telle qu'elle a été définie par la politique de sécurité de l'unité est assurée ;
- Détecter toute défaillance ou anomalie de sécurité, volontaire ou accidentelle, passive ou active, d'origine matérielle ou humaine ;
- Détecter toute violation de la loi ou tout abus d'utilisation des moyens informatiques pouvant engager la responsabilité du CNRS ;
- Être à même de fournir des preuves nécessaires pour mener les enquêtes en cas d'incident de sécurité et de répondre à toute réquisition officielle présentée dans les formes légales.

Quelles traces conserver ? Et combien de temps ?

Les traces à enregistrer de manière systématique portent sur l'utilisation des moyens suivants :

- Les serveurs et postes de travail ;
- Les équipements d'extrémité de réseau et la surveillance des services réseau (routeurs, pare-feux, ...) ;
- Les équipements de surveillance du trafic réseau (IDS, antivirus, antispam, ...) ;
- Les applications spécifiques.

Des durées maximales de conservation sont indiquées pour chacun de ces types de traces.

En revanche, les durées minimales de conservation sont laissées à l'appréciation des gestionnaires du système.

Elles pourront, en fonction de l'évolution de la législation, être précisées dans des textes ultérieurs.

Le problème : les traces sont nominatives.

Les objectifs précités imposent d'aller au-delà d'un enregistrement et d'une exploitation de données statistiques.

Ils impliquent nécessairement l'enregistrement, la conservation temporaire et l'éventuelle exploitation de données nominatives, dans la mesure où des éléments contenus dans les traces permettent de remonter à l'utilisateur.

Ces traces et leur traitement doivent également respecter les droits de chacun et notamment être conformes à la loi du 6 janvier 1978 dite "Informatique et libertés".

Elles doivent avoir satisfait au principe d'information préalable et de transparence.

Les serveurs de messagerie enregistrent pour chaque message émis ou reçu les informations suivantes :

- L'adresse de l'expéditeur et éventuellement des éléments identifiant celui qui s'est connecté au serveur SMTP ;
- L'adresse du destinataire ;
- La date et l'heure de la tentative ;
- Les différentes machines (relais de messagerie) dont il est reçu des messages ou auxquelles il en est envoyé ;
- Le traitement « accepté ou rejeté » du message (on peut rejeter des messages ne respectant pas les standards) ;
- Le sujet du message dans le cas où il ne contiendrait pas que des caractères standards ;
- Parfois la taille du message ainsi que l'en-tête "message-id" qui peut contenir, en fonction des outils utilisés, des éléments formés à partir d'adresse électronique ;
- Le cas échéant le résultat du traitement antispam ou antivirus sur ce message.

Ces données sont conservées au maximum pour une durée d'un an.

# Pour résumer

Le triple objectif de ces traitements est de veiller :

- au respect de la politique de sécurité ;
- au bon fonctionnement du matériel et logiciel ;
- à l'équilibrage de charge des équipements et logiciels.

Pour tout traitement répondant à d'autres objectifs que ceux-là, une demande d'autorisation spécifique devra être faite à la CNIL.

# Syslog : un protocole et un logiciel

Syslog est un protocole définissant un service de journaux d'événements d'un système informatique.
C'est aussi le nom du format qui permet ces échanges.

En tant que protocole, Syslog se compose d'une partie cliente et d'une partie serveur.
- la partie cliente émet les informations sur le réseau, via le port UDP 514 ;
- les serveurs collectent l'information et se chargent de créer les journaux.

L'intérêt de Syslog est donc de centraliser les journaux d'événements, permettant de repérer plus rapidement et efficacement les défaillances d'ordinateurs présents sur un réseau.

Il existe aussi un logiciel appelé Syslog, qui est responsable de la prise en charge des fichiers de journalisation du système. Ceci inclut aussi le démon klogd, responsable des messages émis par le noyau linux.

# Syslog, Metalog ou Syslog-ng : lequel choisir ?

Metalog n'est pas capable de journaliser sur des serveurs à distance.
- journalise par nom de programme, importance ou fonction (comme syslogd) ;
- permet l'analyse des journaux avec des expressions rationnelles permettant de déclencher l'exécution de commandes.

Syslog-ng :
- mêmes fonctionnalités que syslog et metalog
- filtre les messages en se basant sur un niveau d'exécution et sur un contenu,
- gère des journaux distants comme syslogd,
- exploite des journaux venant de syslogd,
- écrit sur une console TTY,
- exécute des programmes et peut être paramétré comme serveur de journaux.

Syslog-ng  représente actuellement le meilleur système, combinant les qualités de Metalog et de Syslog en ajoutant des options de configuration avancées.

# La solution mise en oeuvre à la DR7

Centralisation des traces sur un serveur GNU/Linux-Gentoo / Syslog-ng (3.1.4)
- syslog-ng sur tous les serveurs Linux
- snare sur tous les serveurs Windows

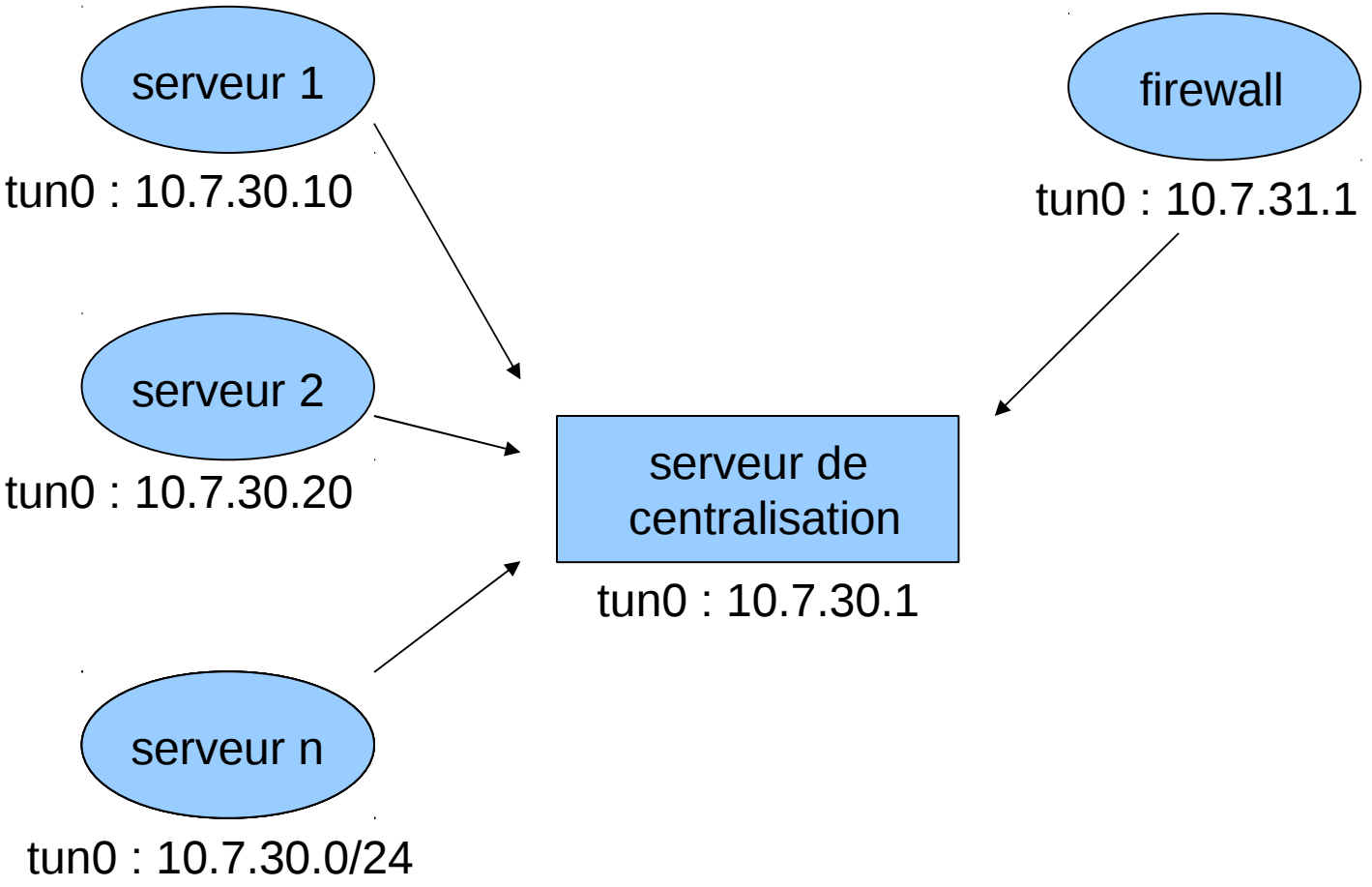
Les traces sont classées par date, serveur et service.
/var/log/syslog-ng/AAAA-MM-JJ/serveur/service

Les répertoires de plus d'un an sont détruits chaque jour.

Syslog-ng utilise UDP... Nous choisissons OpenVPN pour sécuriser le flux.

Le serveur est hébergé dans le laboratoire voisin, et accessible via une fibre optique dédiée. Merci Laurence !

# Schéma simplifié de la centralisation des traces



# Les sources de syslog-ng

source <identifier> { source-driver(params); source-driver(params); ... };

Therefore, the following means : src gets messages from /dev/log socket and syslog-ng.

source src { unix-stream("/dev/log"); internal(); };

The kernel sends log messages to /proc/kmsg and the file() driver reads log messages from files. Therefore, the following means : kernsrc gets messages from file /proc/kmsg

source kernsrc { file("/proc/kmsg"); };

Previously, the source in the default configuration file was defined as:

source src { unix-stream("/dev/log"); internal(); pipe("/proc/kmsg"); };

# Les destinations de syslog-ng

destination <identifier> {destination-driver(params); destination-driver(params); ... };

This means sending authlog messages to /var/log/auth.log:

destination authlog { file("/var/log/auth.log"); };

If you want to send console messages to root's terminal if it is logged in:

destination console { usertty("root"); };

The following sends xconsole messages to the pipe /dev/xconsole.

destination xconsole { pipe("/dev/xconsole"); };

To send messages on the network, use udp().

destination remote_server { udp("10.0.0.2" port(514)); };

# Les filtres de syslog-ng

filter <identifier> { expression; };

To filter those messages coming from authorisation, use the following:
filter f_auth { facility(auth); };

The following filter selects those messages not coming from authorisation, network news or mail:
filter f_debug { not facility(auth, authpriv, news, mail); };

The function level() selects messages based on its priority level, so if you want to select informational levels:
filter f_info { level(info); };

The following line filters messages with a priority level from informational to warning not coming from atuh, authpriv, mail and news facilities:
filter f_messages { level(info..warn) and not facility(auth, authpriv, mail, news); };

# Configuration d'un client (syslog-ng.conf)

```
options {
        chain_hostnames(on); create_dirs(yes); perm(0644); dir_perm(0755);
        log_fifo_size (3072); flush_lines(0); stats_freq(43200);
};

source src { unix-stream("/dev/log" max-connections(256)); internal(); file("/proc/kmsg"); };
source kernsrc { file("/proc/kmsg"); };

destination service { file("/var/log/service.log"); };
filter f_service { program("service")); };
log { source(src); filter(f_service); destination(service); };

destination serveur { udp("adr.vpn.srv.log" port(5140));};
log { source(src); destination(serveur); };
log { source(kernsrc); destination(serveur); };
```

# Configuration du serveur de centralisation des traces (syslog-ng.conf)

```
options {
        chain_hostnames(on); create_dirs(yes); perm(0644); dir_perm(0755);
        log_fifo_size (3072); flush_lines(0); stats_freq(43200);
};

source src { unix-stream("/dev/log" max-connections(256)); internal(); file("/proc/kmsg"); };
source kernsrc { file("/proc/kmsg"); };

source UDP { udp ( port(5140)); };

filter f_service { program("service"); };

destination SERVICE {
    file("/var/log/syslog-ng/$YEAR-$MONTH-$DAY/$HOST/service.log");};

log { source(UDP); filter(f_service); destination(SERVICE);  };
```

# Générer syslog-ng.conf avec un script sur chaque serveur-client

```
# ls /etc/syslog-ng
syslog-ng.service_1      syslog-ng.base      syslog-ng.conf
syslog-ng.service_2      syslog-ng.serveur
...
syslog-ng.service_n
```

Le script :

```
cat /etc/syslog-ng/syslog.base > /etc/syslog-ng/syslog.conf
for file in service_1 service_2 ... service_n
    cat /etc/syslog-ng/syslog.$file >> /etc/syslog-ng/syslog.conf
done
cat /etc/syslog-ng/syslog.serveur > /etc/syslog-ng/syslog.conf
```

# Logrotate pour épurer les traces des serveurs

Main configuration file is /etc/logrotate.conf
Other configuration files are included from /etc/logrotate.d/*

Common Options Reference

* compress - compress rotated logs using gzip
* create *mode owner group* - specify permissions and ownership for logs
* daily, weekly, monthly - how often should logs be rotated
* size *size*[G|M|k] - rotate, if log-file size exceeds size
* mail *address* - mail rotated logs to specified address
* olddir *directory* - move rotated logs to the specified directory
* rotate *count* - keep count rotated log files

for more options see man logrotate !

# Exemple sur un serveur de messagerie [non externalisée]

```
/var/log/mail.log {
    prerotate
        /usr/bin/perl /usr/local/bin/up_maillog.pl
    endscript
    daily
    compress
    rotate 30
    sharedscripts
    postrotate
        /bin/kill -HUP `cat /var/run/syslogd.pid 2> /dev/null` 2> /dev/null || true
    endscript
}
```

# Analyse des traces avec Logwatch et Logcheck

Journaliser les événements n'est que la moitié de la bataille.

Logwatch émet un rapport toutes les 24h synthétisant les évènements par service. La grande force de logwatch, c'est la sommation des entrées répétitives qui optimise la taille du rapport.

Logcheck est un script accompagné d'un programme binaire nommé logtail, qui compare vos journaux à un ensemble de règles pour repérer une éventuelle activité suspecte. Il envoie ensuite le résultat par courrier à l'utilisateur root.

Nous n'utilisons ni l'un ni l'autre... Cela fait partie des choses à faire !

# Conclusion

La centralisation des traces : c'est facile.
Néanmoins, cela prend du temps...

Cela permet :

- sécuriser la gestion des traces (Disponibilité, Intégrité, Confidentialité)
- faciliter leur analyse.

Un préalable : prévenir les utilisateurs.


www.cnrs.fr