Surveillance et analyse du trafic réseau

Table of Contents

Avec une sonde et un collecteur, on peut surveiller le trafic global d'un réseau (sa bande passante) et ainsi prévenir les éventuelles perturbations. De plus, une analyse fine de ce trafic est possible avec l'identification des applications.

Alors que les autres IDS du marché comme Snort ou Suricata requièrent des ressources systèmes plus importantes pour fonctionner, les solutions proposées peuvent se satisfaire d'un petit Raspberry Pi.

Le petit processeur du Rasbperry Pi sera tout de même fortement sollicité en fonction de la bande passante de la connexion vers Internet. Si jamais la solution n'arrive pas à analyser l'ensemble du trafic, un certain nombre de paquets réseau seront exclus de l'analyse et on ne pourra observer qu'une partie du trafic analysé.

Installation et configuration du système

Installation de Raspbian sur Raspberry Pi

Raspbian Strech est installé sur Raspberry Pi. nProbe est installé sur le dessus de Raspbian Stretch.

Voici la liste des mots de passe par défaut:

Appareil	nom d'utilisateur/mot de passe
Raspberry Pi	pi/raspberry

Appareil	**nom d'utilisateur/mot de passe**
ntopng	admin/admin123

Télécharger la dernière version de Raspbian Stretch

```
 wget https://downloads.raspberrypi.org/raspbian_lite_latest
```

Extraire l'image.

```
unzip raspbian_lite_latest
```

Il faut s'assurer que la carte SD n'est pas montée. Si oui, démonter la carte.

```
sudo umount /dev/mmcblk0
```

Copier l'image piCore extraite sur la carte SD.

```
sudo dd bs=4M if=2018-06-27-raspbian-stretch-lite.img of=/dev/mmcblk0
status=progress conv=fsync
```

Retirer la carte SD et l'insér dans Raspberry Pi. Se connecter avec le nom d'utilisateur **pi** et le mot de passe **raspberry**.

Par défaut, l'image n'utilise que 4 Go de l'espace de la carte SD. Lancer la commande ci-dessous.

```
raspi-config
```

Accéder à `Advanced Options-> A1 Expand Filesystem` Garantit que tout le stockage de la carte SD est disponible pour le système d'exploitation. Le système de fichiers sera agrandi au prochain redémarrage.

Définir une adresse IP statique pour l'interface eth0.

```
echo "interface eth0" >> /etc/dhcpcd.conf
echo "static ip_address=172.17.100.50/16" >> /etc/dhcpcd.conf
echo "static routers=172.17.100.1" >> /etc/dhcpcd.conf
echo "static domain_name_servers=172.17.100.1 8.8.8.8" >> /etc/dhcpcd.conf
```

Revenir dans raspi-config:

```
raspi-config
```

Accéder à `Interface Options->` SSH et cliquer sur **yes** pour activer le serveur SSH.

Il faut configurer le bon fuseau horaire.

```
dpkg-reconfigure tzdata
```

Installation de FreeBSD sur un Raspberry Pi

FreeBSD est une distribution Unix libre de la famille BSD, FreeBSD ne contient pas de noyau Linux mais un noyau FreeBSD monolithique modulaire. FreeBSD est largement utilisé sur des plateformes minimalistes avec de forts besoins sécuritaires : Firewall, stockage ... C'est d'ailleurs sur FreeBSD que se repose la solution **Pfsense**, largement déployée comme solution de Firewall.

FreeBSD supporte l'architecture ARM du Raspberry Pi depuis quelques années.

Voici la liste des mots de passe par défaut:

Utilisateur	**nom d'utilisateur/mot de passe**
utilisateur standard	freebsd/freebsd
administrateur	root/root

Après avoir téléchargé l'image, dézipper l'archive avec un outil de décompression, insérer une carte Micro SD dans l'hôte et flasher la carte Micro SD avec l'image téléchargée.

```
wget
https://download.freebsd.org/ftp/snapshots/arm/armv6/ISO-IMAGES/13.0/FreeBSD
-13.0-STABLE-arm-armv6-RPI-B-20211028-74efe421ea0-247853.img.xz
```

Liste des images pré-construites disponibles:

Version	**ARCH**	**Releases**	**Snapshots**
RPI-B	armv6 (tous)	12+	12-STABLE, 13-STABLE, 14-CURRENT
RPI2	armv6 (12+), armv7 (12+)	armv6 12+, armv7 12+	12-STABLE, 13-STABLE, 14-CURRENT
RPI3	aarch64 (12)	12	12-STABLE
RPI	aarch64 (13+)	13+	13-STABLE, 14-CURRENT

Pour Raspberry **Pi Zero**, utiliser l'image pour **RPI-B**. Pour Raspberry **Pi 3**, **Pi 4** et **Pi 400**, utiliser l'image pour **RPI** sur 13+.(Pi 400 est connu pour fonctionner avec l'instantané 20210624)

Une fois l'opération réalisée, insérer la carte dans le Raspberry Pi et brancher sur le réseau.

On a deux choix :

- Connecter le Rasbperry Pi à un clavier et un écran et configurer le Rasbperry Pi localement.
- Se connecter en SSH au Raspberry Pi avec l'adresse IP que le serveur DHCP lui a donné.

Pour réaliser les opérations suivantes, prendre les droits root dans le terminal.

```
su
```

Une fois cette opération réalisée, l'étape suivante consiste à changer le fuseau horaire :

```
ln -s  /usr/share/zoneinfo/Europe/Paris /etc/localtime
ntpdate  ntp.nic.fr
```

Changer la structure du clavier :

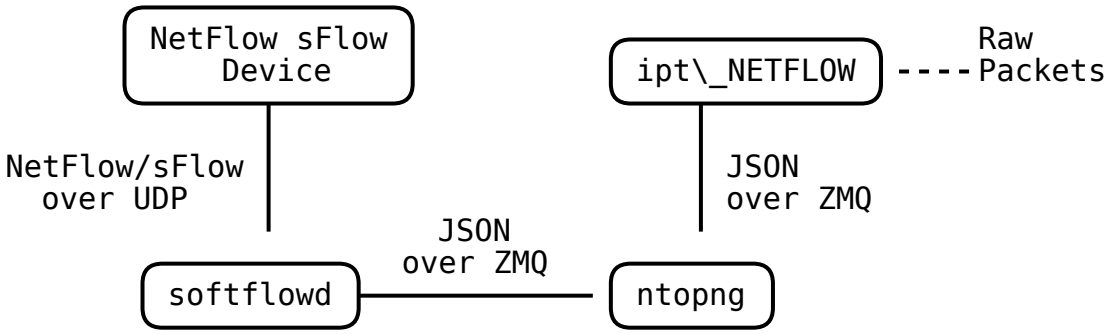
```
kbdmap
```

Mettre à jour le gestionnaire de paquets :

```
pkg update
```

Sonde ntopng

ntopng présente l'utilisation du réseau graphiquement. Il est basé sur **libpcap** et il a été écrit dans un format portable afin de fonctionner virtuellement sur toutes les plateformes Unix. Il fournit une interface utilisateur Web intuitive et cryptée pour l'exploration des informations de trafic en temps réel et historiques.



Les échanges de données nécessaires nécessaires à l'affichage de données sont :

- Récupération et envoie des données à un collecteur.
- Le collecteur les transforme en données ZMQ et envoie à **Ntopng**.
- Réception et traitement des informations par **Ntopng**.

  NProbe n'étant pas gratuit, on utilisera les collecteurs: **fprobe**, **Softflowd** ou **ipt_NETFLOW**. La communication entre les collecteurs et **ntopng** s'effectue via ZeroMQ, un protocole de publication.

Collecteur fprobe

fprobe, permet de collecte les statistiques des interfaces réseau et les transmettre au collecteur.

Pour installer **fprobe** sur Ubuntu/Debian utiliser la commande :

```
apt-get install fprobe
```

Pour installer **fprobe** sur FreeBSD :

Vérifier la présence de **fprobe** parmis les outils installables :

```
pkg search fprobe
```

Installer le package

```
pkg install net-mgmt/fprobe
```

Pour configurer **fprobe**, éditer son fichier de configuration :

```
vi /etc/default/fprobe
```

Et par exemple, indiqer:

```
INTERFACE="eth0"
FLOW_COLLECTOR="localhost:555"
OTHER_ARGS="-fip"
```

- **INTERFACE** indique depuis quelle interface intercepter le trafic, si nécessaire sur toutes celles disponibles indiquer «any».
- **FLOW_COLLECTOR** indique l'adresse IP et le port sur lequel le collecteur reçoit les paquets UDP.
- **OTHER_ARGS** indique d'autres options. Dans ce cas, un filtre est spécifié qui sélectionne uniquement les paquets IP (les paquets ARP ne seront pas pris en compte).

Redémarrer **fprobe** pour appliquer les modifications :

```
service fprobe restart
```

Vérifier si **fprobe** est en cours d'exécution et si le collecteur est en cours d'exécution :

```
service fprobe status
netstat -anpl |grep fprobe
ps ax | grep fprobe
netstat -anpl |grep 555
```

Pou s'assurer qu'il est ajouté à startup au démarrage du système d'exploitation :

```
systemctl is-enabled fprobe.service
systemctl enable fprobe.service
```

Collecteur ip_NETFLOW

Basé sur iptables, **ipt_NETFLOW** est un module d'exportation de données de flux Netflow. Il est connu pour sa grande rapidité d'export étant un module kernel.

installer le package iptables-netflow-dkms, et configurer iptables.

```
#Configuration d'IPTABLES
iptables -I OUTPUT -j NETFLOW
iptables -I INPUT -j NETFLOW
iptables -I FORWARD -j NETFLOW

#Lancement d'ipt_NETFLOW
modprobe ipt_NETFLOW destination=127.0.0.1:2055 promisc=1 protocol=10

#promisc=1: Acceptation par la carte réseau de tous les paquets
#protocol=10: Protocol Netflow IPFIX
```

Collecteur SoftFlowD

SoftFlowD comme alternative en tant que sonde/agent NetFlow. Ce qui suit est une description de la façon dont on peut installer une sonde basée sur TCP Flow pour capturer les données entrant et sortant d'un serveur Linux et pour les exporter au format NetFlow Version 5 vers un collecteur pour une analyse plus approfondie.

Tout d'abord, il faut s'assurer que quelques utilitaires sont installés sur le serveur pour satisfaire les dépendances: `libtool automake autoconf python-dev libpcap-dev`

Une fois ceux-ci installés, récupérer une copie des fichiers source compressés softflowd et construire les bianires:

```
git clone https://github.com/irino/softflowd.git
cd softflowd/
autoconf
autoreconf --install
./configure --enable-ntopng
make
make install
```

Pour installer **SoftFlowD** sur FreeBSD :

Vérifier la présence de **SoftFlowD** parmis les outils installables :

```
pkg search softflowd
```

Installer le package

```
pkg install net-mgmt/softflowd
```

Options de démarrage :

Option	Description
-n	spécifie le noeud réseau et le port sur lequel fonctionnera softflowd
-i	interface sur laquelle fonctionnera softflowd
-r pcap_file	lecture des informations d'un fichier, pas d'une interface réseau
-p pidfile	emplacement alternatif pour stocker l'identifiant du processus, standard /var/run/softflowd.pid
-c ctlsock	emplacement alternatif pour le socket, standard /var/run/softflowd.ctl
-m max_flows	nombre maximum de threads pour le suivi simultané
-6	considére également les données IPv6
-D	mode débogage
-T track_level	niveau de suivi, peut être complet, proto, ip
-v version_netflow	version netflow

Après l'installation, éditer le fichier de configuration :

```
sudo vi /etc/default/softflowd
```

Et préciser les paramètres, par exemple :

```
INTERFACE="any"
OPTIONS="-n 192.168.1.40:5556"
```

Après les modifications, effectuez un redémarrage :

```
sudo /etc/init.d/softflowd restart
```

Pour voir ci cela fonctionne, regarder les statistiques de softflowd :

```
softflowctl statistics
```

S'il ne fonctionne pas, il y aura une erreur :

```
ctl connect(“/var/run/softflowd.ctl”) error: Connection refused
```

Installation de Netflow2ng

Netflow2ng permet de transformer les données envoyés par les sondes en ZMQ pour **Ntopng**.

Ce collecteur doit être capable de gérer un flux important de données d’où le choix de ZMQ qui

permet de gérer un grand nombre de flux, en ouvrant et fermant plusieurs de ces threads.

Installer le collecteur à partir du Github

exécuter le fichier binaire

```
./netflow2ng/netflow2ng-v0.0.2-8-g887b99b-linux-x86_64 &
[1] 2000
INFO[0000] Starting netflow2ng
INFO[0001] Started ZMQ listener on: tcp://*:5556
INFO[0001] Listening on UDP :2055                        Type=NetFlow
```

ajouter des options à l'exécution du collecteur

```
./netflow2ng/netflow2ng-v0.0.2-8-g887b99b-linux-x86_64 & -port 2056 -
zmq.listen tcp://*:5557 # -port: port d'écoute du NetFlow # -zmq.listen:
écoute toutes les adresses sur le port 5556 pour transmettre les données
```

La sonde est prête à collecter et envoyer des données sur **Ntopng**. La dernière étape, installer **Ntopng**.

Installation de ntopng

Sur Raspbian télécharger et importer la clé publique.

```
wget http://packages.ntop.org/apt/ntop.key
sudo su

apt-key add ntop.key
```

Ajouter le référentiel ntop.

```
wget https://packages.ntop.org/RaspberryPI/apt-ntop_1.0.190416-469_all.deb
sudo dpkg -i apt-ntop_1.0.190416-469_all.deb
sudo apt-get update
```

Installer **ntpong** à partir du référentiel.

```
sudo apt-get install ntopng
```

Sur FreeBSD, vérifier la présence de **NtopNG** parmis les outils installables :

```
pkg search ntopng
ntopng-4.0.d20200710,1         Network monitoring tool with command line and
web interfaces
```

Installer **Ntopng**

```
pkg install ntopng
```

Et enfin installer **Redis**[1)], une dépendance forte de **NtopNG** pour fonctionner.

```
pkg install redis
```

Pour installer **Ntopng** sur FreeBSD :

Vérifier la présence de NtopNG parmis les outils installables :

```
pkg search ntopng
ntopng-4.0.d20200710,1         Network monitoring tool with command line and
web interfaces
```

Installer Ntopng :

```
pkg install ntopng
```

Et enfin installer **Redis**[2)], une dépendance forte de **NtopNG** pour fonctionner.

```
pkg install redis
```

Une fois l'ensemble des outils installer, il faut ajouter les lignes suivantes dans le fichier `/etc/rc.conf` du Raspberry Pi pour que ces services puissent démarrer automatiquement à chaque redémarrage système.

```
vi /etc/rc.conf

redis_enable="YES"
ntopng_enable=YES
ntopng_flags=/usr/local/etc/ntopng.conf
```

On peut maintenant démarrer le processus **Redis** et **Ntopng** :

```
service redis onestart
service ntopng onestart
```

Pour une meilleure visualisation de la bande passante, on peut utiliser **Grafana** qui est un outil de génération de tableaux de bord.

Pour faire la liaison de **Ntopng** avec **Grafana**, il faut utiliser **InfluxDB** une base de

données orientée haute performance (comme le collecteur ZMQ), afin d'enregistrer les flows reçus.

Configuration de la sonde

ntoptng peut être configurer en tant que collecteur sflow (si l'appliance réseau prend en charge l'export de flux Netflow/sFlow, on peut envoyer les données de flux à un serveur distant exécutant **ntopng**). Cette configuration n'est pas appropriée lorsqu'on a besoin d'une dissection détaillée de l'application L7 ou d'une analyse en temps réel par paquet. Il faut configurer le collecteur de flux intermédiaire (**fProbe**, **ipt_netflow** ou **SoftflowD**), qui à son tour enverra des flux à **ntopng**.

Voici un exemple de configuration minimale et fonctionnelle de **ntopng**.

ntopng peut être démarré manuellement

```
ntopng -i tcp://127.0.0.1:5556
```

Modifier le fichier de configuration de **Ntopng** pour démarrer les sockets TCP qui correspondent aux adresses des collecteurs dans le fichier de configuration, puis redémarrer le service.

```
#location: /etc/ntopng/ntopng.conf
-G=/var/run/ntopng.pid

#Collecteur ipt_netflow
-i=tcp://172.20.44.58:5556

#Port d'écoute pour l'interface web
-w=3000

#Rétention des données dans une base de données MySQL
-F "mysql;localhost;ntopng;flowsv4;usr;pwd"
```

Voici un exemple sur la façon de configurer plusieurs instances **nProbe** (deuxième approche) :

```
#location: /etc/ntopng/ntopng.conf
-G=/var/run/ntopng.pid

#Collecteur ipt_netflow
-i=tcp://172.20.44.58:5556

#Collecteur PfSense
-i=tcp://172.20.44.58:5557

#Collecteur OpenWRT
-i=tcp://172.20.44.58:5558
```

```
#Port d'écoute pour l'interface web
-w=3000

#Rétention des données dans une base de données MySQL
-F "mysql;localhost;ntopng;flowsv4;usr;pwd"
```

Sonde Nfsen

Nfsen est un collecteur et analyseur Netflow open source qui affiche des statistiques dans une interface Web sous forme de graphiques.

Avant d'installer **Nfsen**, il faut installer **Nfdump**, comme décrit ci-dessous.

Installation de Nfdump

Nfdump est une collection d'outils pour collecter et traiter les données netflow.

Pour installer **Nfdump** sur Ubuntu/Debian, faire :

```
sudo apt-get install nfdump
```

Pour installer Nfdump sur FreeBSD :

Vérifier la présence de **Nfdump** parmis les outils installables :

```
pkg search nfdump
```

Installer le package

```
pkg install databases/rrdtool net-mgmt/nfdump
```

nfdump se compose de :

- **nfcapd** - récupére les données du réseau netflow et les enregistre dans des fichiers.
- **nfdump** - lit les données netflow à partir de fichiers.
- **nfprofile** - lit les données netflow à partir de fichiers à l'aide de filtres et enregistre le résultat dans un fichier.
- **nfreplay** - lit les données netflow des fichiers et les envoie à un autre nœud du réseau.
- **nfclean.pl** - un script pour nettoyer les anciennes données.
- **ft2nfdump** - lit et convertit les données.

après l'installation, **nfcapd** démarre, écoute sur le port 2055 et écrit les données dans /var/cache/nfdump.

Pour exécuter manuellement **nfcapd** et voir les données collectées :

```
mkdir /tmp/nfcap-test
nfcapd -E -p 9001 -l /tmp/nfcap-test
CTRL+C
ls -l /tmp/nfcap-test
nfdump -r /tmp/nfcap-test/nfcapd.* | moins
nfdump -r /tmp/nfcap-test/nfcapd.* -s srcip/octets
```

On peut voir les processus nfcapd en cours d'exécution avec la commande :

```
ps auxwww | grep nfcapd
```

Voici quelques exemples de statistiques :

	`nfdump -r /var/cache/nfdump/nfcapd.202001292351 -c 100`
	`nfdump -o raw -r /var/cache/nfdump/nfcapd.202001292351`
	`nfdump -r /var/cache/nfdump/nfcapd.202001292351 -c 100 'proto tcp and ( src ip 172.17.5.21 or dst ip 172.17.5.22 )'`
	`nfdump -r /var/cache/nfdump/nfcapd.202001292351 'proto tcp and ( src ip 172.17.5.21 and dst port 80 or dst port 443 and dst ip 192.168.1.5 )'`
	`nfdump -M /srv/nfsen/profiles-data/live/upstream1/2020/01/30:/srv/nfsen/profiles-data/live/upstream1/2020/01/30 -R nfcapd.202001300000:nfcapd.202001300455 -s record -n 20 -o extended`
afficher tous les fichiers dans le répertoire spécifié	`nfdump -R /var/cache/nfdump/ 'proto tcp and (dst port 80 or dst port 443)'`
générer les résultats dans un fichier, par exemple, pour une adresse IP spécifique qui a visité des ressources Web	`nfdump -R /srv/nfsen/profiles-data/live/upstream1/2020/04/06/ 'proto tcp and src ip 192.168.5.5 and (dst port 80 or dst port 443)' > file.txt`
récupérer tout le trafic TCP pour une adresse IP spécifique	`nfdump -R /srv/nfsen/profiles-data/live/upstream1/2020/04/06/ 'proto tcp and src ip 192.168.5.5' > tcp_flows_src.txt` `nfdump -R /srv/nfsen/profiles-data/live/upstream1/2020/04/06/ 'proto tcp and dst ip 192.168.5.5' > tcp_flows_dst.txt`
convertir des adresses IP en domaines	`nfdump -R /srv/nfsen/profiles-data/live/upstream1/2020/04/06/ 'proto tcp and src ip 192.168.5.5' | perl -i -p -e 'use Socket; s#(\d{1,3}(\.\d{1,3}){3})\b#gethostbyaddr(inet_aton($1), AF_INET) or sprintf($1)#egi' > 06.04.2020_all_tcp_resolved.txt`

Il faut également configurer un capteur qui lui transmettra des données, par exemple **fprobe** ou configurer netflow sur le commutateur.

Pour s'assurer que **nfdump** démarre au démarrage du système d'exploitation :

```
systemctl is-enabled nfdump
```

```
systemctl enable nfdump
systemctl status nfdump
```

Si iptables est utilisé, ouvrir le port du capteur :
`iptables -A ENTRÉE -s 192.168.5.5/32 -p udp --dport 2055 -j ACCEPTER`

Installation de Nfsen

Pour installer Nfsen sur Debian:

Installer les composants nécessaires :

```
sudo add-apt-repository universe
sudo apt-get install apache2 php libapache2-mod-php librrds-perl librrdp-perl librrd-dev libmailtools-perl build-essential autoconf rrdtool
```

Télécharger **Nfsen** sur https://sourceforge.net/projects/nfsen/ et décompresser :

```
mkdir /srv/nfsen
cd /srv/nfsen
tar xzfv nfsen-1.3.8.tar.gz
```

Créer un fichier de configuration :

```
cd nfsen-1.3.8/etc
cp nfsen-dist.conf nfsen.conf
nanonfsen.conf
```

Editer le fichier et modifier les éléments suivants (apach2 et nginx fonctionnent par défaut à partir de l'utilisateur www-data):

```
$REPBASE = "/srv/nfsen" ;
$PREFIX = '/usr/bin';
$USER = "www-données" ;
$WWWUSER = "www-données" ;
$WWWGROUP = "www-données" ;

%sources = (
    'upstream1' => { 'port' => '555', 'col' => '#0000ff', 'type' => 'netflow' },
);
```

Exécuter le script d'installation nfsen :

```
cd ..
./install.pl ./etc/nfsen.conf
```

Pour installer Nfsen sur FreeBSD :

Vérifier la présence de Nfsen parmis les outils installables

```
pkg search nfsen
```

Installer le package

```
pkg install databases/rrdtool mail/p5-Mail-Tools net/p5-Socket6 net-mgmt/nfdump lang/php74 lang/perl5.32 www/php74-session net/php74-sockets net-mgmt/nfsen
```

Exécuter nfsen :

```
/srv/nfsen/bin/nfsen start
```

Si on doit exécuter nfsen sur le port 2055/udp et qu'il a été pris par défaut par nfdump (par le processus nfcapd), arrête puis relancer nfsen :

```
systemctl is-enabled nfdump
systemctl stop nfdump
netstat -anpl | grep 2055
kill -9 PID_NUMBER
netstat -anpl | grep nfcapd
```

Dans la configuration, on a spécifié en amont1 avec le port 555, donc après avoir exécuté **nsfsen**, il démarrera automatiquement **nfcapd** sur le port 555 et écrira des données dans le répertoire `/srv/nfsen/profiles-data/live/upstream1/`.

Pour un démarrage automatique au démarrage du système d'exploitation, exécuter les commandes suivantes :

```
ln -s /srv/nfsen/bin/nfsen /etc/init.d/nfsen
update-rc.d nfsen defaults 20
```

Il reste à configurer le serveur web ou simplement à créer un lien symbolique dans le répertoire www (après cela on pourra ouvrir **nfsen** dans un navigateur, par exemple http://ixnfo.com/nfsen/nfsen.php) :

```
ln -s /srv/nfsen/www/ /var/www/html/nfsen
```

```
ln -s /var/www/nfsen/ /var/www/html/nfsen
```

Après avoir modifié la configuration, par exemple, lorsqu'on a besoin d'ajouter ou de modifier des sources :

```
cd /srv/nfsen/bin
./nfsen reconfig
```

Après un certain temps, les données devraient apparaître sur les graphiques, également via tcpdump, on peut voir si les données proviennent du capteur :

```
port tcpdump 555 -e -n
```

Pour s'assurer que **nfsen** démarre au démarrage du système d'exploitation :

```
systemctl is-enabled nfsen
systemctl is-enabled nfdump
systemctl enable nfsen
systemctl status nfsen
```

Si flow-tools est installé dans le système d'exploitation, on peut le désactiver comme ceci :

```
systemctl is-enabled flow-capture
systemctl disable flow-capture
systemctl status flow-capture
systemctl stop flow-capture
```

Dans le fichier php.ini, il faut spécifier le bon fuseau horaire, par exemple :

```
date.timezone = Europe/Kiev
```

[1)] , [2)]
Redis est un magasin clé-valeur utilisé par **ntopng** pour mettre en cache les données et les préférences. **Redis** s'exécute en tant que service externe. **ntopng** se connecte à **Redis** à l'aide de sockets. Ne pas se connecter à **Redis** entraînerait un dysfonctionnement de **ntopng**. **Redis** doit toujours être opérationnel et accessible pour garantir l'intégralité des fonctionnalités.