Collection Formation Réseau Cisco CCNA

IP ROUTING

# Surveillance du réseau

## CDP, LLDP, NTP, SYSLOG, Netflow, TFTP

François-Emmanuel Goffinet
Formateur IT
Version 15.11

# Sommaire

# Surveillance : aperçu

- Surveillance locale, diagnostic manuel : show, cdp, debug, ...
- Surveillance centralisée :
  - des protocoles : syslog, SNMP, Netflow
  - des outils de test : syslogd, snmpwalk, cacti, ntopng, nagios, oreon, zenoss
  - des outils commerciaux divers.
- Station de surveillance Linux voir Annexe.

# Objectifs ICND1/ICND2

**ICND1**

**4. Technologies de routage IP**

- 4.4. Vérifier la configuration du routeur et la connectivité réseau
- 4.4.e Show cdp neighbors

**5. Services IP**

- 5.6. Configurer et vérifier un client NTP

**ICND2**

**2. Technologies de routage IP**

- 2.3 Gérer des fichiers Cisco IOS

**3. Services IP**

- 3.2 Configurer et vérifier syslog
- 3.2.a Utiliser les sorties syslog
- 3.3 Décrire SNMP v2 and v3

**4. Diagnostic**

- 4.2 Utiliser les données netflow

# 1. Voisinage L2
# CDP et LLDP

# Introduction protocole de voisinage

## L2 : couche Liaison de données

- Utilité : identification, diagnostic, surveillance, gestion et configuration des périphériques
- CDP et LLDP
- Vulnérable dans l'environnement L2

## L3 : couche Internet

- Résolution d'adresses et maintien de relations de voisinage au niveau d'IPv6 (L3)
- ND (ICMPv6)
- Vulnérable au sein de l'environnement L3 (domaine de broadcast/multicast)

# Protocoles de voisinage L2

On peut trouver deux protocoles de voisinage de couche L2 :

- CDP (propriétaire Cisco, activé par défaut sur les routeurs et commutateurs).
- LLDP (standard inter-opérable)

Leur objectif principal est d'échanger des informations entre périphériques intermédiaires qui peuvent s'identifier finement à travers des TLVs.

# Caractéristiques des protocoles de voisinage L2

- Uniquement transmis dans des trames
- Portée L2
- Adresses L2 Multicast réservées
- Versions de protocoles
- Délais mise-à-jour/durée de vie
- Informations transportées

# Voisinage de couche 3

Neighbor Discovery (ND, ICMPv6) est aussi un protocole de voisinage mais de couche 3 :

- Voisins = Noeuds attaché au même lien (L2)
- Objectif : maintenir les informations L2, “Link-Layer”
- Utilisé en IPv6 par Neighbor Discovery (ICMPv6/ND) pour la résolution d’adresse
- La détection des voisins
- Le maintien des informations de voisinage
- La détection du routeur et des paramètres du réseau
- de manière active

# Cisco Discovery Protocol CDP

- Cisco Discovery Protocol, propriétaire
- Protocole de couche 2 (embarqués dans des trames) :
- Adresse de destination **01:00:0c:cc:cc:cc**
- Logical-Link Control
  - DSAP: SNAP (0xaa)
  - …
  - Organization Code: Cisco (0x00000c)
  - PID: CDP (0x2000)
- Mises à jour par défaut toutes les 60 secondes.
- Informations retenues (holdtime) 3 X 60 secondes = 180 s.
- **Activé par défaut** sur toutes les interfaces
- A désactiver (globalement ou par interfaces) car très indiscret

# CDP en Cisco IOS

```
gw#show cdp
Global CDP information:
    Sending CDP packets every 60 seconds
    Sending a holdtime value of 180 seconds
    Sending CDPv2 advertisements is enabled
gw#show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone
Device ID    Local Intrfce   Holdtme    Capability   Platform    Port ID
SW0          Fas 0/0          148            S       2960        Gig 1/2
Switch       Fas 0/1          178            S       2950        Fas 0/1
gw(config)#cdp timer 30
gw(config)#cdp holdtime 120
```

## Désactivation par interface ou globale :

```
gw(config)#int f0/1
gw(config-if)#no cdp enable
gw(config-if)#exit
gw(config)#no cdp run
```

# Détails CDP sous Cisco IOS

```
#show cdp neighbors detail
Device ID: Switch
Entry address(es):
IP address : 192.168.10.254
Platform: cisco 3560, Capabilities:
Interface: GigabitEthernet0/2, Port ID (outgoing port): GigabitEthernet0/1
Holdtime: 175

Version :
Cisco IOS Software, C3560 Software  (C3560-ADVIPSERVICESK9-M), Version 12.2
(37)SE1, RELEASE SOFTWARE (fc1)
Copyright (c) 1986-2007 by Cisco Systems, Inc.
Compiled Thu 05-Jul-07 22:22 by pt_team

advertisement version: 2
Duplex: full
---------------------------
```

# Captures CDP

- CDPv1
- CDPv2
- CDPv2 avec vlan voice
- CDPv2 HDLC

Sources : https://wiki.wireshark.org/CDP

# Link Layer Discovery Protocol (LLDP)

Link Layer Discovery Protocol (LLDP) est un protocole normé dans la publication **IEEE 802.1ab**. C'est un protocole destiné à remplacer un bon nombre de protocoles propriétaires (Cisco CDP, Extreme EDP, etc.) utilisés dans la découverte des topologies réseau de proche en proche, mais aussi à apporter des mécanismes d'échanges d'informations entre équipements réseaux, et utilisateurs finaux.

LLDP est un protocole ouvert constitué de :

- un entête et une fin de message fixe
- un ensemble de conteneurs d'information (TLV)

# Intérêts de LLDP

L'intérêt de LLDP vient du modèle ouvert de gestion des TLV :

1. Si un équipement de transit reçoit un message LLDP, il le lit dans son intégralité, et interprète tous les TLVs qu'il peut interpréter.
2. S'il lit un TLV qu'il ne sait pas interpréter, il le conserve tel quel dans le message et ne le prend pas en compte localement
3. Il retransmet ensuite le message originel en modifiant les TLV interprétés s'il y a besoin de les modifier, et les TLV non interprétés en les laissant tel quels.

# Media endpoint discovery extension

Media Endpoint Discovery (LLDP-MED) est une amélioration de  LLDP qui offre les fonctionnalités suivantes :

- Découverte automatique des LAN policies (VLAN, priorités L2, Diffserver) activant le réseau plug-and-play
- Localisation de périphériques par découverte permettant la création de bases de données (avec la VoIP, permettant un service d'urgence)
- Gestion de l'alimentation PoE étendue et automatisée pour les périphériques terminaux
- Gestion d'inventaire permettant de suivre les périphériques et de collecter leur caractéristiques.

# LLDP sous Cisco IOS

```
(config)#lldp run
#show lldp
Global LLDP Information:
    Status: ACTIVE
    LLDP advertisements are sent every 30 seconds
    LLDP hold time advertised is 120 seconds
    LLDP interface reinitialisation delay is 2 seconds
#show lldp neighbors
Capability codes:
    (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
    (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other

Device ID           Local Intf     Hold-time  Capability      Port ID
R2                  Et0/1          120        R               Et0/0
```

# Soucres CDP / LLDP

- https://wiki.wireshark.org/CDP
- https://en.wikipedia.org/wiki/Link_Layer_Discovery_Protocol
- https://wiki.wireshark.org/LinkLayerDiscoveryProtocol?action=show&redirect=LLDP
- http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/cdp/configuration/15-mt/cdp-15-mt-book/nm-cdp-discover.html
- https://supportforums.cisco.com/discussion/12285176/voice-vlan-cdp-role
- http://www.cisco.com/c/en/us/support/docs/network-management/discovery-protocol-cdp/118736-technote-cdp-00.html
- http://standards.ieee.org/getieee802/download/802.1AB-2009.pdf

# 2. Synchronisation temporelle : NTP

# Network Time Protocol

Le Protocole d'Heure Réseau (Network Time Protocol ou NTP) est un protocole qui permet de synchroniser, via un réseau informatique, l'horloge locale d'ordinateurs sur une référence d'heure.

La version 3 de NTP est la plus répandue à ce jour. Elle est formalisée par la RFC1305. NTP utilise le port **UDP123**.

http://fr.wikipedia.org/wiki/Network_Time_Protocol

http://www.pool.ntp.org/fr/

# Configuration de NTP (client)

Vérification du statut NTP :

```
show clock
show calendar
show ntp status
show ntp associations
```

Configuration client NTP :

```
ntp server <ntp_server>
```

# Ajuster l’heure

```
(config)#clock timezone GMT+1 +1

(config)#clock summer-time BE recurring last
SUN MAR 02:00 last SUN OCT 02:00
```

# Vérifications NTP client (1/2)

```
R1#show ntp status
Clock is synchronized, stratum 4, reference is
193.190.138.68
nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz,
precision is 2**18
reference time is D9D11FF3.360B31ED (20:17:55.211 UTC Tue
Oct 20 2015)
clock offset is -1383.6025 msec, root delay is 25.33 msec
root dispersion is 1421.69 msec, peer dispersion is 4.50
msec
```

# Vérifications NTP client (2/2)

```
R1#show ntp associations

      address         ref clock       st   when   poll reach
delay   offset    disp
*~193.190.138.68   195.13.23.5       3     52     64   377
20.1   -1383.     4.5
 ~172.16.124.134   0.0.0.0          16      7    128     0
0.0     0.00   16000.
 * master (synced), # master (unsynced), + selected, -
candidate, ~ configured
```

# 3. Journalisation : Syslog

# Syslog

Syslog est un **protocole** qui définit 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.

Syslog est la solution de journalisation standard sur les systèmes Unix et Linux, il y a également une variété d'implémentations syslog sur d'autres systèmes d'exploitation (Windows notamment). Il est en général présent dans les périphériques réseau tels que les commutateurs ou routeurs.

# Format Syslog

Un journal au format syslog comporte dans l'ordre les informations suivantes : la date à laquelle a été émis le log, le nom de l'équipement ayant généré le log (hostname), une information sur le processus qui a déclenché cette émission, le niveau de gravité du log, un identifiant du processus ayant généré le log et enfin un corps de message.

Certaines de ces informations sont optionnelles.

exemple :

```
Sep 14 14:09:09 machine_de_test dhcp service[warning] 110 corps du message
```

Les origines peuvent être multiples et sont juxtaposées à l'aide d'un ';'.

Elles sont construites sous la forme :

```
facility.criticity
```

La criticité doit être comprise comme la criticité minimale, ainsi user.critical correspond au message d'origine utilisateur pour le niveau de criticité critical et les niveaux supérieurs, en l'occurrence alert et emergency.

Le mot clef "none" peut lui aussi être utilisé afin de filtrer les messages, il est alors utilisé en lieu et place de la criticité.

# Niveaux de sévérité/gravité

## Security levels

| | | |
|---|---|---|
| 0 | Emerg (emergency) | Système inutilisable |
| 1 | Alert | Une intervention immédiate est nécessaire |
| 2 | Crit (critical) | Erreur critique pour le système |
| 3 | Err (error) | Erreur de fonctionnement |
| 4 | Warning | Avertissement |
| 5 | Notice | Événement normal méritant d'être signalé |
| 6 | Info (informational) | Pour information seulement |
| 7 | Debug | Déboggage |

# Facilité

Outre les niveaux de gravité, les messages sont orientés au regard de leur origine, dont les codes sont regroupés suivant des types que l'on appelle des "facilités"  de local0 à local7. Les 24 facilités syslog sont

- **AUTH**, Message de sécurité/autorisation
- **AUTHPRIV**, Message de sécurité/autorisation (privé).
- **CRON**, Message d'un démon horaire
- **DAEMON**, Démon du système sans classification particulière.
- **FTP**, Démon ftp.
- **KERN**, Message du noyau.
- **LOCAL0** à **LOCAL7**, Réservé pour des utilisations locales.
- **LPR**, Message du sous-système d'impression.
- **MAIL**, Message du sous-système de courrier.
- **NEWS**, Message du sous-système des news USENET.
- **SYSLOG**, Message interne de **syslogd**
- **USER** (défaut), Message utilisateur générique.
- **UUCP**, Message du sous-système UUCP.

# Configuration client Syslog

```
logging <ip_adress>
logging facility <niveau>
```

Local0 à Local7 (Local7 par défaut)

```
logging trap <niveau>
```

Emergency: 0, Alert: 1, Critical: 2, Error: 3, Warning: 4, Notice: 5, Informational: 6, Debug: 7

## Configuration de l'horodatage :

```
service timestamps debug datetime
service timestamps log datetime
```

## Vérification :

```
show logging
```

# Sources Syslog

https://fr.wikipedia.org/wiki/Syslog

https://en.wikipedia.org/wiki/Syslog

# 4. Supervision : SNMP

# Principe

Simple Network Management Protocol (abrégé SNMP), en français « protocole simple de gestion de réseau », est un protocole de communication qui permet aux administrateurs réseau de gérer les équipements du réseau, de superviser et de diagnostiquer des problèmes réseaux et matériels à distance.

SNMP utilise les ports **UDP161** et **UDP162**.

# Version de SNMP

Il existe actuellement 3 versions différentes du protocole SNMP :

- SNMP v1 (RFC1155, RFC1157 et RFC1212).
- SNMP v2c (RFC1901 à 1908).
- SNMP v3 (RFC3411 à 3418).

La co-existence des trois versions est détaillée dans le RFC3584.

# Éléments SNMP

Les systèmes de gestion de réseau sont basés sur trois éléments principaux :

- un superviseur,
- des nœuds (ou nodes)
- et des agents.

# Superviseur

Dans la terminologie SNMP, le synonyme manager est plus souvent employé que superviseur.

Le superviseur est la console qui permet à l'administrateur réseau d'exécuter des requêtes de management.

Il est client SNMP

# Agents et noeuds

Les agents sont des entités qui se trouvent au niveau de chaque interface, connectant au réseau l'équipement géré (nœud) et permettant de récupérer des informations sur différents objets.

Ils sont serveurs SNMP (UDP161/UDP162).

# Objets OID

Les commutateurs, routeurs, postes de travail et serveurs (physiques ou virtuels) sont des exemples d'équipements contenant des objets gérables.

Ces objets gérables peuvent être des informations matérielles, des paramètres de configuration, des statistiques de performance et autres objets qui sont directement liés au comportement en cours de l'équipement en question.

Ces objets sont classés dans une sorte de base de données arborescente définie par l'ISO appelée MIB (« Management Information Base »). SNMP permet le dialogue entre le superviseur et les agents afin de recueillir les objets souhaités dans la MIB.

# En bref

- Les équipements gérés (managed devices) sont des éléments du réseau (ponts, commutateurs, concentrateurs, routeurs ou serveurs), contenant des « objets de gestion » (managed objects) pouvant être des informations sur le matériel, des éléments de configuration ou des informations statistiques ;
- Les agents, c'est-à-dire les applications de gestion de réseau résidant dans un périphérique, sont chargés de transmettre les données locales de gestion du périphérique au format SNMP ;
- Les systèmes de gestion de réseau (network management systems notés NMS), c'est-à-dire les consoles à travers lesquelles les administrateurs peuvent réaliser des tâches d'administration.

# En pratique

Concrètement, dans le cadre d'un réseau, SNMP est utilisé :

- pour administrer les équipements
- pour surveiller le comportement des équipements

# Polling

SNMP peut être utilisé de deux manières distinctes : le polling et les traps.

Le polling consiste simplement à envoyer une requête à intervalles réguliers pour obtenir une valeur particulière. Cette technique est appelée « vérification active ». Vous pouvez, par programme ou script, vérifier si les valeurs sont correctes. Si la requête échoue, il est possible qu'il y ait un problème avec le périphérique. Cependant, vu que le SNMP s'appuie sur UDP, il est conseillé de réitérer la requête pour confirmer le problème (surtout dans le cas d'une vérification au travers d'Internet).

# Traps

Les traps consistent à faire de la vérification passive ; en gros, on configure l'agent SNMP pour qu'il contacte un autre agent SNMP en cas de problème. C'est-à-dire que l'on peut configurer un périphérique réseau (comme un routeur) pour qu'il envoie un trap SNMP lors de certains événements. Par exemple, le routeur peut envoyer un trap lorsqu'il détecte que la ligne est coupée (down). Quand un événement trap apparait, l'agent sur le périphérique va envoyer le trap vers une destination pré-configurée communément appelé trap host. Le trap host possède son propre agent SNMP qui va accepter et traiter les traps lorsqu'ils arrivent. Le traitement des traps est effectué par des trap handlers. Le handler peut faire ce qui est approprié pour répondre au trap, comme envoyer un courriel d'alerte ou faire ce qu'on veut.

# Sécurité

Les schémas de sécurité dépendent des versions de SNMP (v1, v2 ou v3).

Dans les versions 1 et 2, une requête SNMP contient un nom appelé **communauté**, utilisé comme un mot de passe. Sur de nombreux équipements, la valeur par défaut de communauté est *public* ou *private*. Pour des raisons de sécurité, il convient de modifier cette valeur. Un nom de communauté différent peut être envisagé pour les droits en lecture et ceux en écriture.

Les versions 1 et 2 du protocole SNMP comportent de nombreuses lacunes de sécurité. C'est pourquoi les bonnes pratiques recommandent de n'utiliser que la version 3.

# Configuration SNMP

En configuration globale

```
(config)#
snmp-server community <nom> RO
```

alternative RW

# Sécurisation de SNMP

- Choose a Good SNMP Community String
- Setup SNMP View
- Setup SNMP Community with access-list
- Setup SNMP Version 3
- Setup ACL on Interfaces

http://www.cisco.com/en/US/tech/tk648/tk362/technologies_tech_note09186a0080094489.shtml

# Test SNMP

En ligne de commande :

http://sourceforge.net/projects/net-snmp/files/net-snmp%20binaries/5.6.1.1-binaries/

```
snmpwalk -v2c -c <nom> <périphérique>
```

En GUI

http://sourceforge.net/p/snmpb/wiki/Home/

# Exemple RW

Exemple de rapatriement de configuration via SNMP.

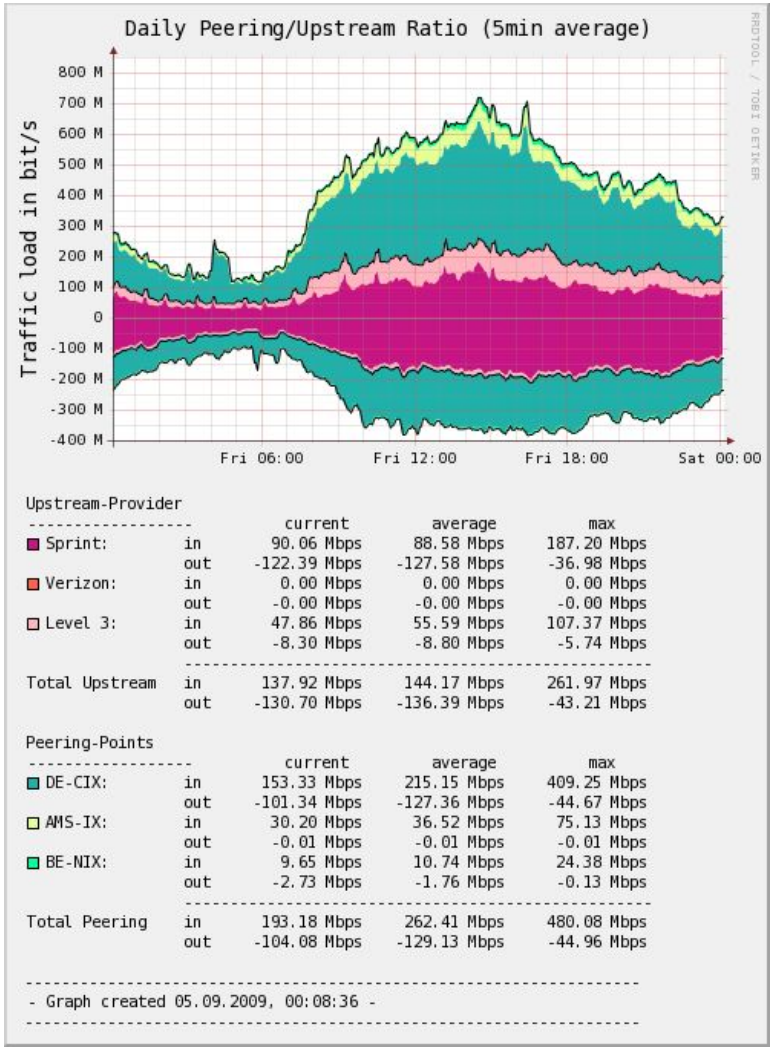
Logiciel net-snmp :

```
snmpset -c [snmp-community-string] -v 2c [ip-device] 1.3.6.1.4.1.9.9.96.1.1.1.1.2.336 i 1
snmpset -c [snmp-community-string] -v 2c [ip-device] 1.3.6.1.4.1.9.9.96.1.1.1.1.3.336 i 4
snmpset -c [snmp-community-string] -v 2c [ip-device] 1.3.6.1.4.1.9.9.96.1.1.1.1.4.336 i 1
snmpset -c [snmp-community-string] -v 2c [ip-device] 1.3.6.1.4.1.9.9.96.1.1.1.1.5.336 a [ip-tftp-server]
snmpset -c [snmp-community-string] -v 2c [ip-device] 1.3.6.1.4.1.9.9.96.1.1.1.1.6.336 s [file-name]
snmpset -c [snmp-community-string] -v 2c [ip-device] 1.3.6.1.4.1.9.9.96.1.1.1.1.14.336 i 1
```

# Configuration SNMP Windows

http://www.lovemytool.com/blog/2012/02/how-to-enable-snmp-in-windows-7-by-tony-fortunato.html

# Graphes MRTG/RRD




Upstream-Provider

| | | current | average | max |
|---|---|---|---|---|
| Sprint: | in | 90.06 Mbps | 88.58 Mbps | 187.20 Mbps |
| | out | -122.39 Mbps | -127.58 Mbps | -36.98 Mbps |
| Verizon: | in | 0.00 Mbps | 0.00 Mbps | 0.00 Mbps |
| | out | -0.00 Mbps | -0.00 Mbps | -0.00 Mbps |
| Level 3: | in | 47.86 Mbps | 55.59 Mbps | 107.37 Mbps |
| | out | -8.30 Mbps | -8.80 Mbps | -5.74 Mbps |
| Total Upstream | in | 137.92 Mbps | 144.17 Mbps | 261.97 Mbps |
| | out | -130.70 Mbps | -136.39 Mbps | -43.21 Mbps |

Peering-Points

| | | current | average | max |
|---|---|---|---|---|
| DE-CIX: | in | 153.33 Mbps | 215.15 Mbps | 409.25 Mbps |
| | out | -101.34 Mbps | -127.36 Mbps | -44.67 Mbps |
| AMS-IX: | in | 30.20 Mbps | 36.52 Mbps | 75.13 Mbps |
| | out | -0.01 Mbps | -0.01 Mbps | -0.01 Mbps |
| BE-NIX: | in | 9.65 Mbps | 10.74 Mbps | 24.38 Mbps |
| | out | -2.73 Mbps | -1.76 Mbps | -0.13 Mbps |
| Total Peering | in | 193.18 Mbps | 262.41 Mbps | 480.08 Mbps |
| | out | -104.08 Mbps | -129.13 Mbps | -44.96 Mbps |

- Graph created 05.09.2009, 00:08:36 -

RRDtool est un outil de gestion de base de données RRD (Round-Robin database) créé par Tobias Oetiker. Il est utilisé par de nombreux outils open source, tels que Cacti, collectd, Lighttpd, et Nagios, pour la sauvegarde de données cycliques et le tracé de graphiques, de données chronologiques. Cet outil a été créé pour superviser des données serveur, telles la bande passante et la température d'un processeur. Le principal avantage d'une base RRD est sa taille fixe.

RRDTool inclut également un outil permettant de représenter graphiquement les données contenues dans la base.

RRDTool est un logiciel libre distribué selon les termes de la GNU GPL.

Source : https://fr.wikipedia.org/wiki/RRDTool

# 5. Supervision Netflow

# Netflow

NetFlow est une architecture de surveillance des réseaux développée par Cisco Systems qui permet de collecter des informations sur les flux IP. Elle définit un format d'exportation d'informations sur les flux réseau nommé NetFlow services export format (format d'exportation des services NetFlow, en abrégé protocole NetFlow). Elle permet de superviser de façon fine les ressources du réseau utilisées. Le protocole utilise le port **UDP2055**.

En 2004, Cisco a publié les caractéristiques de la version 9 du protocole NetFlow dans la RFC 3954. L'IETF en a dérivé le protocole IPFIX (IP Flow Information Export), normalisé en 2008 dans les RFC 5101, RFC 5102 et RFC 5103.

# Architecture

Des *éléments réseau* (commutateurs et routeurs) établissent des statistiques sur les données des ***flux réseau*** qu'ils exportent vers des *collecteurs*. Ces statistiques détaillées peuvent porter sur les nombres de paquets et d'octets, les ports applicatifs, les adresses IP, les champs de qualité de service, les interfaces par lesquelles ils transitent, etc.

# Flux réseau

Un flux réseau NetFlow est unidirectionnel.

Il est caractérisé par 7 champs clés :

1. le protocole de couche 3 (en général IPv4, mais d'autres protocoles sont possibles)
2. l'adresse IP source
3. l'adresse IP destination
4. le port source (UDP ou TCP, 0 pour les autres protocoles)
5. le port destination
6. le champ Type of Service
7. l'interface en entrée

Les paquets appartenant à un même flux (même adresse IP source, même adresse IP destination, etc.) sont décomptés dans les statistiques. On remarque que l'interface de sortie ne caractérise pas un flux, ce qui est une bonne chose sur les routeurs où les routes de sortie peuvent changer.

Il existe aussi des champs non clés qui ne caractérisent pas un flux, mais dont la valeur est relevée. En règle générale, seule la valeur pour le premier paquet du flux est indiquée. On peut par exemple relever la date et l'heure du début du flux.

# Données exportées

L'équipement réseau envoie un enregistrement décrivant le flux quand le flux s'achève. Un flux est considéré comme achevé lorsqu'il n'y a plus de paquets qui passent pendant un certain temps, ou quand la connexion TCP est close. On peut aussi configurer l'équipement pour envoyer des enregistrements à intervalles réguliers, même quand le flux est encore en train de s'écouler.

Ces enregistrements NetFlow sont en général transportés par UDP. L'adresse IP du collecteur auquel ils sont envoyés doit être configurée sur l'équipement émetteur. Un paquet NetFlow peut regrouper plusieurs enregistrements en un seul envoi. Par défaut, on utilise le port UDP 2055, mais il est courant de choisir un autre port.

Pour des raisons d'efficacité, si un de ces enregistrements NetFlow est perdu pour cause de congestion du réseau ou de paquet corrompu, l'équipement réseau est dans l'incapacité de le renvoyer, car il n'en conserve pas une copie. Cela peut conduire à des statistiques dégradées. Pour cette raison, certaines implémentations récentes de NetFlow utilisent SCTP à la place de UDP pour garantir que les statistiques seront reçues (TCP ne convient pas, car il est trop lourd).

# Version de Netflow

| Version | Commentaire |
|---|---|
| v1 | Première implémentation, à présent dépassée. Limitée à IPv4 sans masque réseau ni numéro de système autonome. |
| v2 | Version interne à Cisco, jamais publiée. |
| v3 | Version interne à Cisco, jamais publiée. |
| v4 | Version interne à Cisco, jamais publiée. |
| v5 | La version la plus courante (en 2009) sur de nombreux équipements de différentes marques, mais restreinte aux flux IPv4. |
| v6 | Version qui n'est plus prise en charge par Cisco. |
| v7 | Comme la version 5, avec un champ « routeur source ». |
| v8 | Agrégation de plusieurs informations. |
| v9 | S'appuie sur des modèles (templates), ce qui permet d'ajouter des champs sans redéfinir le standard. Permet de rapporter des flux IPv6, MPLS, ou le prochain saut BGP en IPv4. |
| v10 | Connue comme IPFIX. Champs définis par les utilisateurs, champs en longueur variable. |

# Prise en charge de Netflow

En plus de Cisco, de nombreux constructeurs d'équipements réseau offrent une prise en charge de NetFlow sur leurs boitiers. La liste comprend Juniper, Alcatel-Lucent et Nortel, entre autres. En ce qui concerne les plates-formes logicielles, il y a une prise en charge sur serveurs VMWare et sous Linux.

Certains constructeurs utilisent un autre nom pour cette technologie, sans doute parce que NetFlow est ressenti comme une marque déposée de Cisco:

- Jflow ou cflowd chez Juniper Networks
- NetStream chez 3Com/HP
- NetStream chez Huawei Technologies
- Cflowd chez Alcatel-Lucent
- Rflow chez Ericsson
- AppFlow chez Citrix

Il existe aussi des alternatives. sFlow est un protocole concurrent. IPFIX est le standard de l'IETF dérivé de NetFlow.

# Configuration de Netflow

Sur l'interface à surveiller :

```
(config-if)#ip route-cache flow
(config-if)#ip flow {ingress | egress}
```

En configuration globale :

```
(config)#ip flow-export destination <ip_address>
```

Optionnellement :

```
(config)#ip flow-export source Loopback 0
(config)#ip flow-export version 5
```

Régularité :

```
(config)#ip flow-cache timeout active 5
(config)#ip flow-cache timeout inactive 15
```

# Vérification Netflow

## #sh run | begin ip flow

```
ip flow-cache timeout active 5
ip flow-export version 5
ip flow-export destination 172.16.124.134 23456
```

## #show ip flow interface

```
FastEthernet0/0
  ip route-cache flow
FastEthernet0/1
  ip route-cache flow
```

# Vérification Export Netflow

```
#show ip flow export
Flow export v5 is enabled for main cache
  Export source and destination details :
  VRF ID : Default
    Destination(1)  172.16.124.134 (23456)
  Version 5 flow records
  97 flows exported in 65 udp datagrams
  0 flows failed due to lack of export packet
  0 export packets were sent up to process level
  0 export packets were dropped due to no fib
  0 export packets were dropped due to adjacency issues
  0 export packets were dropped due to fragmentation failures
  0 export packets were dropped due to encapsulation fixup failures
```

# Cache Netflow (1/3)

**#show ip cache flow**

```
IP packet size distribution (128 total packets):
   1-32   64   96  128  160  192  224  256  288  320  352  384  416  448
480
   .000 .000 .101 .296 .421 .093 .007 .000 .062 .000 .015 .000 .000 .000 .
000

    512  544  576 1024 1536 2048 2560 3072 3584 4096 4608
   .000 .000 .000 .000 .000 .000 .000 .000 .000 .000 .000
```

# Cache Netflow (2/3)

```
IP Flow Switching Cache, 278544 bytes
  3 active, 4093 inactive, 98 added
  1570 ager polls, 0 flow alloc failures
  Active flows timeout in 5 minutes
  Inactive flows timeout in 15 seconds
IP Sub Flow Cache, 25800 bytes
  3 active, 1021 inactive, 89 added, 89 added to flow
  0 alloc failures, 0 force free
  1 chunk, 1 chunk added
  last clearing of statistics never
```

# Cache Netflow (3/3)

```
Protocol         Total    Flows   Packets Bytes  Packets Active(Sec) Idle(Sec)
--------         Flows     /Sec     /Flow  /Pkt     /Sec     /Flow     /Flow
UDP-NTP             13      0.0         1    76      0.0       0.0      15.3
UDP-other           61      0.0         1   160      0.0       0.0      15.4
ICMP                20      0.0         2   127      0.0       1.9      15.7
Total:              94      0.0         1   138      0.0       0.4      15.5

SrcIf         SrcIPaddress    DstIf         DstIPaddress    Pr SrcP DstP  Pkts
Fa0/1         74.125.232.146  Local         172.16.124.131  01 0000 0000     5
Fa0/1         172.16.124.1    Null          172.16.124.255  11 445C 445C     1
Fa0/1         172.16.124.2    Local         172.16.124.131  11 0035 D60A     1
```

# Références


- http://fr.wikipedia.org/wiki/NetFlow
- http://gregsowell.com/?p=610
- http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6555/ps6601/prod_white_paper0900aecd80406232.html
- http://www.cisco.com/en/US/docs/ios/12_2/switch/configuration/guide/xcfnfc.html
- https://supportforums.cisco.com/thread/2185457
- http://www.manageengine.com/products/netflow/help/cisco-netflow/cisco-ios-netflow.html

# 6. Transfert de fichiers TFTP, FTP, SSH

# Client TFTP

Par exemple, un backup de configuration :

```
#copy run tftp
Address or name of remote host []? 172.16.124.134
Destination filename [r1-confg]? r1-confg-test
!!
1763 bytes copied in 1.924 secs (916 bytes/sec)
```

# Server TFTP

```
(config)#tftp-server ?
  archive:  Allow URL file TFTP load requests
  flash:    Allow URL file TFTP load requests
  null:     Allow URL file TFTP load requests
  nvram:    Allow URL file TFTP load requests
  slot0:    Allow URL file TFTP load requests
  system:   Allow URL file TFTP load requests
  tmpsys:   Allow URL file TFTP load requests
  xmodem:   Allow URL file TFTP load requests
  ymodem:   Allow URL file TFTP load requests
```

# 7. Station de supervision sous Linux

# Supervision Open Source

Sous Windows :

- TFTPD32 : Serveur DHCP, TFTP, DNS, SNTP, Syslog,TFTP client, prêt en IPv6

En Applicance ou logiciel Linux

- NTOP : notamment Netflow collector
- Cacti : outils de graphes basé SNMP
- Nagios, Icinga, Zenoss, Zabbix, Cricket

# Serveur de supervision Linux

Voici un aide mémoire pour monter un serveur de supervision avec **Ubuntu Server** :

- Rapporteur CDP
- Synchronisation temporelle NTP : openntpd
- Journalisation Syslog : rsyslog (embarqué)
- Supervision SNMP : snmp
- Supervision NetFlow : nfdump, nfcapd
- Transferts de fichier TFTP : tftpd-hpa
- Serveur SSH : openssh-server

# Rapporteur CDP

```
# apt-get install cdpr
# cdpr -help
# cdpr
# cdpr -d eth0 -vvv
```

# Serveur NTP

Installation d'un serveur NTP

```
# apt install openntpd
```

Fichier de configuration

```
# mv /etc/openntpd/ntpd.conf /etc/openntpd/ntpd.conf.old
# vi /etc/openntpd/ntpd.conf
    listen on *
    server pool.ntp.org
```

Redémarrage du service

```
# /etc/init.d/openntpd restart
```

Vérification

```
# grep ntpd /var/log/syslog
# netstat -an | grep :123
```

# Serveur Syslog

```
# grep -v ^# /etc/rsyslog.conf | grep -v ^$
$ModLoad imudp
$UDPServerRun 514
$ModLoad imtcp
$InputTCPServerRun 514
...
# grep -v ^# /etc/rsyslog.d/50-default.conf | grep -v ^$
local7.*                -/var/log/cisco.log
# touch /var/log/cisco.log
# chown syslog /var/log/syslog
# service rsyslog restart
# netstat -an | grep :514
# tail -f /var/log/cisco.log
```

# Serveur TFTP

```
# apt install tftpd-hpa
# chmod 777 /var/lib/tftpboot
# cat /etc/default/tftpd-hpa
TFTP_USERNAME="tftp"
TFTP_DIRECTORY="/var/lib/tftpboot"
TFTP_ADDRESS="0.0.0.0:69"
TFTP_OPTIONS="--secure --create -v"

# service tftpd-hpa restart
# netstat -an | grep :69
# ls /var/lib/tftpboot
```

# Outils SNMP

```
# apt install snmp
# man snmpwalk
# man snmpset
```

# Collecteur Netflow

```
# apt install nfdump
# mkdir -p /var/lib/netflow/test
# nfcapd -w -D -l /var/lib/netflow/test -p 23456
# netstat -an | grep :23456
# nfdump -R /var/lib/netflow/test
```

# Notes

# Améliorations

- Comparatif protocoles de voisinage L2
- NTP sécurisé, serveur NTP, captures NTP, mise à jour de l'horloge matérielle, client NTP
- SYSLOG serveur de log, log des authentification
- SNMP SET GET TRAPs différences SNMP SNMPv3 sécurisé ...
- Netflow et tous les autres → alignement certification
- Révision textes Wikipedia
- Container Dockers pour les services ?
  - Intégrer les services de supervision dans les sections

# Droits