



TP1

Auteur: Raki HAMMAMI

Année universitaire 2018-2019

TP1 : Quelques techniques d'attaque classiques

Objectif :

L'objectif de ce TP est de mettre en évidence quelques techniques attaques classiques. L'étudiant est amené à se familiariser avec quelques outils open source connu dans l'univers de sécurité et de tirer les leçons aux risques de sécurité potentiels.

Prérequis :

- Chaque étudiant doit disposer d'une machine avec un système d'exploitation Linux
- Des connaissances de base en Unix/Linux
- Les packages d'installations qui seront fournis par votre enseignant

A. Cracking : Test de robustesse des mots de passe

Le crackage des mots de passe consiste à deviner le mot de passe de la victime. Malheureusement, beaucoup d'utilisateurs mal avertis de cette technique mettent des mots de passe évidents comme leur propre prénom ou ceux de leurs enfants. Ainsi, si un pirate, qui a espionné sa victime auparavant, teste quelques mots de passe comme le prénom des enfants de la victime, il aura accès à l'ordinateur. D'où l'utilité de mettre des bons mots de passe. Mais même les mots de passe les plus robustes peuvent être trouvés à l'aide de logiciels spécifiques appelés craqueur (John the ripper, L0phtCrack pour Windows).

1. Mode opératoire

Les craqueurs de mots de passe s'appliquent souvent à un fichier contenant le nom des utilisateurs ainsi que leur mot de passe encrypté. Ces fichiers sont nécessaires pour permettre l'authentification sur un système. L'encryptage des mots de passe s'effectue à l'aide d'une fonction de hachage. Les fonctions de hachage sont des fonctions univoques, c'est-à-dire qu'il est impossible de les inverser pour décrypter un mot de passe encrypté. Une autre particularité importante des fonctions de hachage est que deux mots de passe différents auront forcément un hachage différent. Ainsi, il est impossible de décrypter un mot de passe encrypté. En revanche, il est possible d'encrypter un mot au moyen de cette fonction et de comparer le résultat avec le mot de passe encrypté. S'il y a correspondance, on a deviné le mot de passe. Mais, il est fastidieux d'encrypter des milliers de mots pour trouver les mots de passe. C'est là qu'intervient l'utilité d'un craqueur.

Ces logiciels peuvent tester des mots de passe selon trois méthodes :

a) Attaque par dictionnaire :

Le logiciel teste tous les mots de passe stockés dans un fichier texte. Cette méthode est redoutable car en plus de sa rapidité, elle aboutit généralement puisque les mots de passe des utilisateurs sont souvent des mots existants.



b) Attaque hybride :

Le logiciel teste tous les mots de passe stockés dans un fichier texte et y ajoute des combinaisons. Par exemple, thomas01. Cette méthode est redoutable également puisque beaucoup de personnes mettent des chiffres après leur mot de passe pensant bien faire.

c) Attaque brute-force :

le logiciel teste toutes les combinaisons possibles. Ainsi ce genre d'attaque aboutit à chaque fois. Heureusement, tester toutes les combinaisons prends beaucoup de temps. D'où l'utilité de changer de mots de passe régulièrement.

Le fichier contenant les mots de passes encryptés est donc à protéger. Chaque système d'exploitation à sa méthode :

- **Windows NT**

Ce fichier s'appelle la base SAM. Ce fichier est verrouillé par le noyau dès son démarrage. Ainsi, un utilisateur ne peut pas copier le fichier, ni le lire. Mais, il existe des méthodes permettant de se le procurer.

- **Unix**

Ce fichier est en fait séparé en deux fichiers shadow et passwd. Le fichier passwd contient les noms d'utilisateurs accessibles par tout le monde, et le fichier shadow contenant les mots de passe, accessible uniquement par root.

2. LAB1 : John the ripper

a) Installation

Récupérer sur <http://www.openwall.com/john/> une version récente ou bien demander à votre enseignant le package d'installation.

Décompressez le package en tapant

```
# #bzip2 -cd john-1.7.8.tar.bz2 | tar xvf -
```

Accédez au répertoire john-1.7.8

```
#cd john-1.7.8/src
```

```
#make
```

Notez le type de votre système and tapez:

```
#make clean SYSTEM
```

avec SYSTEM c'est votre système approprié. Exemple : #make clean linux-x86-any

Si votre système n'est pas listé avec make, tapez :

```
#make clean generic
```

Si tout se passe bien, cela va créer les exécutables pour John sous "../run/".

Pour tester :

```
#!/john --test
```

b) Récupération du fichier de mots de passe chiffré

Nous allons récupérer le mot de passe depuis d'autre machine. Pour le cas de notre TP, les fichiers seront fournis par votre enseignant.

Rq : Vous pouvez aussi échanger vos fichiers passwd et tester. Il suffit d'avoir de combiner les 2 fichiers : /etc/passwd et /etc/shadow :

```
#!/usr/bin/unshadow /etc/passwd /etc/shadow > /root/passwd.cop
```

c) Cassage des mots de passe

Dans le répertoire run lancer déjà avec la configuration de base

```
#!/john pass.1
```

```
#!/john pass2
```

Nous allons tester par la suite sur un troisième mot de passe plus complexe

```
#!/john pass3
```

Pour afficher les mots de passe craqués, tapez :

```
#!/john - - show <nom de fichier>
```

Que remarquez-vous ?

Par défaut le système utilise les noms des utilisateurs comme source de mot de passe, puis des combinaisons du petit dictionnaire password.lst avant de se lancer dans une recherche par force brute.

Pour une mise en production dans la vraie vie il faudrait utiliser un dictionnaire de mots plus Important spécialisé par exemple dans le français. On pourrait regarder comment en rajouter un.

Pour utiliser uniquement la méthode d'attaque par dictionnaire (wordlist), utiliser la commande :

```
#!/john --wordlist=password.lst --rules passwd
```

La method « single crack » est plus rapide, elle utilise le nom d'utilisateur et les noms de son Home Directory comme mot de passe :

```
#!/john - -single
```

d) Conclusion

- Choisir un mot de passe robuste et ne pas l'écrire sur un support(papiers, ...) puisque rien n'empêche un pirate de fouiller les poubelles par exemple.
- Un mot de passe robuste doit satisfaire à plusieurs critères :
 - ✓ plus de 8 caractères
 - ✓ utiliser la casse (majuscule/minuscule)
 - ✓ utiliser des chiffres
 - ✓ Une bonne méthode consiste à apprendre par cœur une phrase et à prendre les premières lettres du mot.

- Changer régulièrement de mot de passe pour éviter que ce dernier ne soit trouvé par un tel outil.
- Utiliser régulièrement des outils de craquage de mots passe pour éliminer les mots de passe trop faibles des utilisateurs et intégrer ces outils aux systèmes de changement de mots de passe.

B. Sniffing : Utilisation de dsniff

1. Mode opératoire

Le reniflement (en anglais Sniffing) est une technique qui consiste à analyser le trafic réseau. Lorsque deux ordinateurs communiquent entre eux, il y a un échange d'informations (trafic). Mais, il est toujours possible qu'une personne malveillante récupère ce trafic. Elle peut alors l'analyser et y trouver des informations sensibles.

Exemple : Soit une entreprise possédant 100 ordinateurs reliés entre eux grâce à un hub. Maintenant, si un pirate écoute le trafic réseau entre 8h et 10h (heure de connexion du personnel), il pourra lire tous les noms d'utilisateurs ainsi que leur mot de passe.



LAB2 : Dsniff

On va automatiser le concept en utilisant le logiciel dsniff qui est capable de reconstruire le contenu d'une communication utilisant certains protocoles.

Le serveur Telnet est installé sur la machine WinXP. On veut faire un Telnet depuis la machine Ubuntu et qu'un pirate situé derrière une troisième Machine (Backtrack) fait de l'écoute sur le réseau et essaye de capturer la communication.

Démarrer les machines Ubuntu et Win XP

Lancer sur Backtrack l'outil Dsniff, qui va entrer en écoute en permanence

```
#dsniff -m
```

et essayer de vous connecter depuis la machine Ubuntu vers la machine WinXp avec le protocole telnet :

```
#telnet 10.10.10.2
```

```
Login :admin
```

```
Password : admin123
```

```
root@ubuntu: /home/ubuntu
root@ubuntu:/home/ubuntu# telnet 10.10.10.2
Trying 10.10.10.2...
Connected to 10.10.10.2.
Escape character is '^]'.
Welcome to Microsoft Telnet Service

login: admin
password:

*=====
Bienvenue ♦ Microsoft Telnet Server.
*=====
C:\Documents and Settings\admin>
```

Lancer des commandes sur l'invite de commande puis faire un exit.

Que constatez-vous à la fin de la connexion ?

2. Conclusion

- Utiliser de préférence un switch (commutateur) plutôt qu'un hub.
- Utiliser des protocoles chiffrés pour les informations sensibles comme les mots de passe.
- Utiliser un détecteur de sniffer.

C. Spoofing : Usurpation de l'adresse IP

Introduction

Le protocole **ARP (Address Resolution Protocol)** : implémente le mécanisme de résolution d'une adresse IP en une adresse MAC Ethernet. Les équipements réseaux communiquent en échangeant des trames Ethernet (dans le cas d'un réseau Ethernet bien sûr) au niveau de la couche liaison de données. Pour pouvoir échanger ces informations il est nécessaire que les cartes réseau possèdent une adresse unique au niveau Ethernet, il s'agit de l'adresse MAC (*Media Access Control*).

Quand un paquet IP doit être envoyé la machine expéditrice a besoin de l'adresse MAC du destinataire. Pour cela une requête ARP en broadcast est envoyée à chacune des machines du réseau physique local. Cette requête pose la question : « Quelle est l'adresse MAC associée à cette adresse IP ? ». La machine ayant cette adresse IP répond via un paquet ARP, cette réponse indiquant à la machine émettrice l'adresse MAC recherchée. Dès lors, la machine source possède l'adresse MAC correspondant à l'adresse IP destination des paquets qu'elle doit envoyer. Cette correspondance sera gardée pendant un certain temps au niveau d'un cache (pour éviter de faire une nouvelle requête à chaque paquet IP envoyé).

1. Mode opératoire

Cette attaque, appelée aussi ARP Redirect, redirige le trafic réseau d'une ou plusieurs machine vers la machine du pirate.

C'est une technique de spoofing efficace bien que détectable dans les logs d'administration; elle consiste à s'attribuer l'adresse IP de la machine cible, c'est-à-dire à faire correspondre son adresse IP à l'adresse MAC de la machine pirate dans les tables ARP des machines du réseau. Pour cela il suffit en fait d'envoyer régulièrement des paquets ARP_reply en broadcast, contenant l'adresse IP cible et la fausse adresse MAC. Cela a pour effet de modifier les tables

dynamiques de toutes les machines du réseau. Celles-ci enverront donc leurs trames ethernet à la machine pirate tout en croyant communiqué avec la cible, et ce de façon transparente pour les switches. De son côté, la machine pirate stocke le trafic et le renvoie à la vraie machine en forgeant des trames ethernet comportant la vraie adresse MAC (indépendamment de l'adresse IP).

Cette technique est très puissante puisqu'elle opère au niveau ethernet, permettant ainsi de spoofer le trafic IP et même TCP (cela dépend entre autres des délais engendrés par la machine pirate). D'autre part, elle permet de contourner les barrières que constituent habituellement les switches (partitionnement de réseaux).

LAB3 : Arp Spoofing / Arp poisoning

Soit la machine WinXP 10.10.10.2, sa passerelle par défaut la machine Ubuntu 10.10.10.3 et la machine Backtrack du pirate 10.10.10.4.

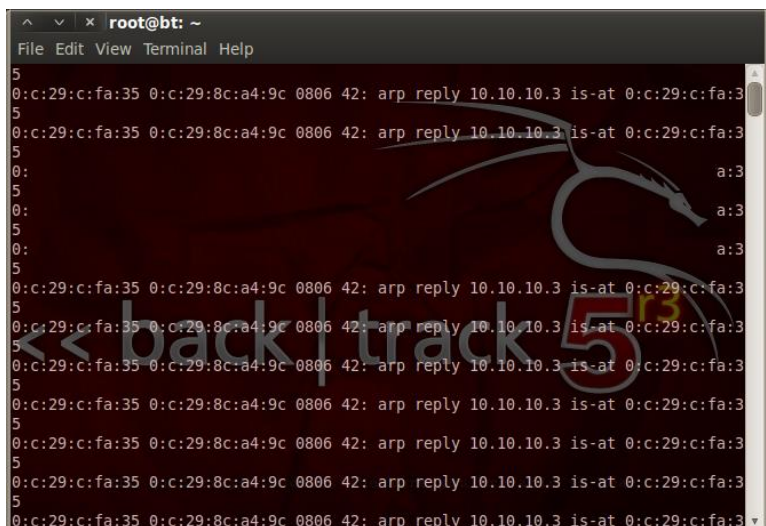
Avant l'attaque, nous allons jeter un œil sur la configuration et le cache ARP de la machine cible (machine WinXP) :

```
#traceroute 10.10.10.3
```

```
#arp -a
```

Depuis la machine Backtrack, lancer alors ARPSpoof :

```
#arp spoof -t 10.10.10.2 10.10.10.3
```



Sur la machine cible, vérifiez le cache ARP. Que remarquez-vous ?

Comment peut-on vérifier que le trafic passe maintenant par la machine 10.10.10.4 ?

Pour compléter la deuxième partie de l'attaque Man in Middle, nous devons lancer l'Arpspoof vers la machine Ubuntu :

```
#arp spoof -t 10.10.10.3 10.10.10.2
```

2. Conclusion :

- L'usurpation d'identité est parmi les attaques les plus difficiles à les détecter.
- Ne pas utiliser uniquement l'adresse IP comme méthode d'authentification.
- Utiliser des solutions anti arp-spoofing : DHCP snooping (cisco..), Xarp,...

D. DoS : Syn Flooding

Le SYN flood est une attaque informatique visant à atteindre un déni de service. Elle s'applique dans le cadre du protocole TCP et consiste à envoyer une succession de requêtes SYN vers la cible.

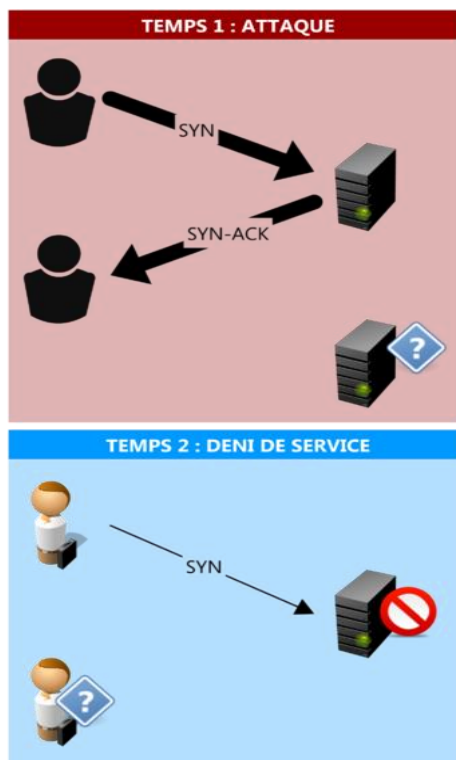
1. Mode opératoire

Lors de l'initialisation d'une connexion TCP entre un client et un serveur, un échange de messages a lieu. Le principe est celui du three-way handshake, qui, dans le cas d'une connexion normale sans volonté de nuire, se déroule en trois étapes :

1. le client demande une connexion en envoyant un message SYN (pour *synchronize*) au serveur ;
2. le serveur accepte en envoyant un message SYN-ACK (*synchronize-acknowledgment*) vers le client ;
3. le client répond à son tour avec un message ACK (*acknowledgment*) ; la connexion est alors établie.

Un client malveillant peut supprimer la dernière étape et ne pas répondre avec le message ACK. Le serveur attend un certain temps avant de libérer les ressources qui ont été réservées pour le client, car le retard du message ACK pourrait être causé par la latence du réseau. Cette période d'attente par le serveur était d'environ 75 secondes lors des premières attaques *SYN flood*.

Après l'étape 2, la connexion est semi-ouverte et consomme un certain nombre de ressources du côté du serveur (mémoire, temps processeur, etc.). En générant suffisamment de connexions incomplètes de ce type, il est possible de surcharger les ressources du serveur et ainsi d'empêcher le serveur d'accepter de nouvelles requêtes, avec pour résultat un déni de service. Dans certains cas, le serveur peut même planter par manque de ressources.



L'attaquant envoie une série de messages SYN, mais laisse les connexions semi-ouvertes. La file d'attente du serveur se remplit et le nouveau client ne peut plus se connecter.

LAB4 : SynFlooding

Lancer l'outil Hping3 sur une cible sur un port spécifique (ex 80) en mentionnant l'attaque Flood comme option


```
#hping3 --flood -p 80 -S <adresse ip cible>
```

(hping --help pour l'explication des options)

Cette commande va envoyer des paquets TCP SYN à la cible, c'est à dire des demandes de connexions, en boucle et sans effectuer d'acquittement de ces connexions. C'est une attaque "SYN flood". Elle a pour conséquence de réserver des ressources sur la cible qui attend un acquittement pour toutes les demandes de connexions qui sont faites, mais elle n'en recevra jamais.

Mais avant que la mémoire ou le CPU ne soit surchargé, on arrive à la saturation du lien car la cible ne fait que envoyer des réponses (SYN-ACK) à toutes les requêtes SYN reçues. Et le débit montant sur un freebox étant beaucoup plus faible que le débit descendant, le lien montant est saturé de réponses et ne tarde pas à être inutilisable

3. Conclusion :

Les façons de parer ces attaques sont :

- la limitation du nombre de connexions depuis la même source ou la même plage d'adresses IP ;
- la libération des connexions semi-ouvertes selon un choix de client et un délai aléatoire ;
- la réorganisation de la gestion des ressources allouées aux clients en évitant d'allouer des ressources tant que la connexion n'est pas complètement établie
-