



Ecole d'ingénieurs
du Canton de Vaud

Intrusion Management

Travail de diplôme 2002

Auteur : Massimo Iritano

Professeur : Stefano Ventura

Expert : Sylvain Maret

Département : E + I

Filière : Télécommunications

Orientation : Réseaux et Services

Cahier des charges

Département E+I
Filière : **Télécommunications**
Candidat : **Massimo Iritano**

TRAVAIL DE DIPLOME 2002

Télécommunications et Téléinformatique

Enterprise Security- Inside and outside: Intrusion Management

Projet du centre de compétence CCTI de l'HES-SOe VPN II

Données de base

Ce travail a pour objet les intrusions à l'intérieur et l'extérieur de l'entreprise. Le terme d'*Intrusion management* englobe quatre types d'activité qui assurent la prévention contre les intrusions (Avoidance, Assurance, Detection) ou les mesures de réparation à prendre en cas d'intrusion réussite (Investigation). Les objectifs de ce travail concernent uniquement certains aspects du management de l'intrusion à savoir : l'étude et présentation des différentes stratégies d'intrusion à l'extérieur et l'intérieur de l'entreprise et les aspects de prévention (avoidance) basés sur une authentification forte.

Cahier des charges

Ce travail englobe les activités et phases suivantes :

Etude et développement d'une plate-forme de laboratoire permettant de simuler et détecter les intrusions

1. Etude et présentation sous forme d'un tutorial des principales stratégies d'intrusions. Il ne s'agit pas de traiter toutes les attaques (voir http://www.iss.net/security_center/advice/Intrusions/default.htm) mais de les classer et traiter les plus représentatives pour chaque

type. Un accent tout particulier sera mis sur les attaques « Inside » de type « layer 2 ». En ce qui concerne les attaques *outside* se référer au travail de diplôme de M Reis de l'EIG

2. Le tutorial sera accompagné par un laboratoire permettant de simuler les différentes attaques à l'intérieur de l'entreprise « *inside* ».
3. Pour chaque type d'attaque simulée proposer et déployer les outils de détections adéquats. Une attention toute particulière doit être aussi donnée aux systèmes de détection basées sur des MIB standardisées.

Intrusion Avoidance

4. Présentez les principales stratégies capables de prévenir les intrusions.
5. Réaliser un démonstrateur capable d'illustrer les stratégies préconisées sous le pt 4 Ce démonstrateur devant compléter le laboratoire selon le pt. 2

Résultats

Lors de l'évaluation finale du travail de diplôme une importance toute particulière sera donnée au respect des directives suivantes :

1. Les résultats du développement feront l'objet d'un rapport circonstancié. Un aperçu du logiciel et de ses caractéristiques sera publié sur Internet, au travers du World Wide Web. Cette publication sera accessible sur internet et présentée lors de la défense du travail de diplôme.
2. De plus le rapport sera accompagné par une présentation sous forme Power Point à présenter lors de la défense de diplôme.
3. Le rapport contiendra toutes les indications nécessaires pour un éventuel développement futur du logiciel par de tierces personnes, ainsi qu'un mode d'emploi suffisamment précis pour pouvoir servir de base à une éventuelle publication. L'utilisation de Word est requise.
4. Les produits réalisés sont à livrer en même temps que le rapport, sous une forme utilisable par le laboratoire :
5. les logiciels sous une forme exécutable, accompagnés des sources et des fichiers nécessaires à la recompilation et à la régénération du produit, seront déposés sur le serveur du laboratoire de téléinformatique;
6. les logiciels, accompagnés des sources et des fichiers nécessaires à la recompilation et à la régénération du produit, seront également joints au rapport sur un CD avec une procédure d'installation ;

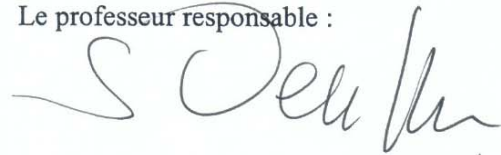
7. le rapport ainsi que la présentation Power Point seront mis à disposition sous forme de CD lors de la défense du travail de diplôme ;

De plus la présentation et le contenu du rapport doivent correspondre aux consignes reçues en annexe.

Le candidat :



Le professeur responsable :



La cheffe du département E+I :



Prof. S. Villa

Yverdon-les-Bains, le vendredi 11 octobre 2002

Table des matières

1	Introduction	9
1.1	Résumé	9
1.2	Mots clés	9
1.3	Avant propos.....	9
1.4	Convention des illustrations.....	10
1.5	Matériel.....	10
2	La recherche d'informations	11
2.1	Introduction.....	11
2.2	Méthodes de recherche.....	11
2.3	Social Engineering.....	11
2.3.1	<i>Par téléphone</i>	11
2.3.2	<i>Par courrier</i>	11
2.3.3	<i>Par contact direct</i>	12
2.3.4	<i>Contre mesure</i>	12
2.4	Les scanners.....	13
2.4.1	<i>Les scanners d'adresses</i>	13
2.4.2	<i>Les scanners de ports</i>	13
2.4.3	<i>Les scanners de vulnérabilité</i>	13
3	Intrusion.....	17
3.1	Déni de Service (DoS)	17
3.1.1	<i>DoS de niveau 2</i>	17
3.1.2	<i>DoS de niveau 3</i>	21
3.1.3	<i>DDoS</i>	25
3.2	Spoofing	26
3.2.1	<i>ARPspoofing</i>	26
3.2.2	<i>IPspoofing</i>	28
3.2.3	<i>DNSspoofing</i>	34
3.3	Hijacking	36
3.4	Buffer Overflow.....	37
3.5	Attaque niveau 2	39
3.5.1	<i>Attaque sur un switch</i>	39
3.5.2	<i>VLAN hopping</i>	47
3.5.3	<i>Détournement de session</i>	48
3.6	Man in the middle.....	53
3.7	Troyen	55

3.8	Réseau sans fil	56
3.9	Mots de passe	57
4	Intrusion Detection System	59
4.1	Port de sécurité	59
4.2	Arpwatch et Arpsnmp	61
4.2.1	<i>Arpwatch</i>	61
5	Outils	67
5.1	Nessus.....	67
5.2	Winarp.....	69
5.2.1	<i>Winarp_ske</i>	69
5.2.2	<i>Winarp_tcom</i>	71
5.3	Dsniff.....	72
5.3.1	<i>Arpspoof</i>	72
5.3.2	<i>Macof</i>	72
5.3.3	<i>Dnspooof</i>	73
5.3.4	<i>Webmitm</i>	73
5.3.5	<i>Dsniff</i>	74
5.4	Ettercap	75
5.5	Trace route.....	76
5.6	Snmpwalk.....	76
5.7	Autres	77
5.7.1	<i>Serveur Apache-SSL</i>	77
5.7.2	<i>htaccess</i>	77
5.7.3	<i>Script Linux</i>	78
6	Tutorial.....	78
7	Laboratoire	78
8	Conclusion.....	79
9	Remerciements.....	80
10	Références.....	81
11	Glossaire.....	83
12	Annexes.....	85

Table des illustrations

Figure 1	Information sur l'EIVD	12
Figure 2	Résultat de l'analyse de la machine Linux	14
Figure 3	Résultat graphique de l'analyse de la machine Linux.....	15
Figure 4	Résultat graphique de l'analyse de la machine Windows XP	15
Figure 5	Résultat de l'analyse de la machine Windows XP avec les deux firewalls.....	16
Figure 6	Réseau sans DoS	17
Figure 7	Réseau avec DoS	18
Figure 8	Topologie du banc de test de DoS sur un utilisateur	18
Figure 9	Connexion FTP avec DoS.....	19
Figure 10	Trame de la connexion FTP avec DoS.....	19
Figure 11	Topologie du banc de test de DoS sur un réseau	20
Figure 12	Cache ARP du routeur, acceptation de l'adresse MAC.....	20
Figure 13	Cache ARP du routeur, refus de l'adresse MAC.....	20
Figure 14	En-tête IP	21
Figure 15	Trame Ethernet (taille en bytes).....	22
Figure 16	Encapsulation de l'ICMP dans de l'IP.....	22
Figure 17	Fragmentation du paquet IP.....	23
Figure 18	Recomposition du paquet IP.....	23
Figure 19	Test d'un Ping of Death.....	24
Figure 20	Ouverture d'une connexion TCP	24
Figure 21	Attaque par DDoS	25
Figure 22	Réseau sans ARPspoofing	26
Figure 23	Réseau avec ARPspoofing.....	27
Figure 24	Topologie du banc de test de l'ARPspoofing.....	27
Figure 25	Cache ARP de la victime.....	28
Figure 26	Cache ARP de la victime empoisonnée.....	28
Figure 27	Configuration de l'adresse IP sous Windows	29
Figure 28	Format du champ option (taille en bytes)	30
Figure 29	Test d'un Ping avec l'option Timestamp (-s).....	31
Figure 30	Test d'un Ping avec l'option Record Route (-r)	31
Figure 31	Trame de l'option Source Routing (taille en bits)	32
Figure 32	Attaque Ipspoofing	33
Figure 33	Organisation des domaines.....	34
Figure 34	Topologie du banc de test du DNSspoofing.....	35
Figure 35	Transfert TCP	36
Figure 36	Désynchronisation de la connexion TCP.....	37
Figure 37	Opérations normales sur la pile	37
Figure 38	Ecrasement de la pile	38
Figure 39	Cas de buffer overflow	38
Figure 40	Attaque sur un switch.....	39
Figure 41	Topologie du banc de test pour attaquer le switch.....	40
Figure 42	Table CAM pleine	40

Figure 43	Timeout de la CAM	41
Figure 44	Nouvelle topologie du banc de test pour attaquer le switch	42
Figure 45	Topologie du banc de test pour attaquer le switch avec port de uplink.....	43
Figure 46	Diagramme fléché	43
Figure 47	Topologie du banc de test des switchs avec VLANs	44
Figure 48	Schéma équivalent à l'attaque sur les VLANs	44
Figure 49	Information sur le Trunk	45
Figure 50	Configuration d'un port	45
Figure 51	Table CAM saturée avec les ports sécurisés	46
Figure 52	Topologie avec VLANs	47
Figure 53	Attaque VLAN Hopping.....	47
Figure 54	Topologie du banc de test pour capturer un mot de passe FTP	48
Figure 55	Mot de passe FTP	49
Figure 56	Connection Telnet sur Ettercap.....	49
Figure 57	Connection Telnet visible par Ettercap.....	50
Figure 58	Réseau avec cas de ICMP redirect.....	51
Figure 59	Diagramme fléché avec ICMP redirect.....	51
Figure 60	Résultat d'un tracert en étant spoofer par arpspoof.....	52
Figure 61	Résultat d'un tracert en étant spoofer par ettercap.....	52
Figure 62	Transaction avec le man in the middle (source : cours de cryptographie de M. Jaton)	53
Figure 63	Topologie du banc de test du man in the middle.....	53
Figure 64	Problème de connection sécurisée	55
Figure 65	Capture Wireless.....	56
Figure 66	Message SNMP trap provenant du switch (capture Ethereal).....	59
Figure 67	Topologie du banc de test de arpswatch.....	62
Figure 68	New Station dans le fichier syslog	63
Figure 69	Mail détectant une nouvelle station (New Station).....	63
Figure 70	Table de correspondance au lancement.....	64
Figure 71	Flip flop dans le fichier syslog.....	64
Figure 72	Table de correspondance après ARPspoofing	64
Figure 73	Mail détectant un changement d'adresse MAC pour une station (flip flop).....	65
Figure 74	Changement d'adresse IP dans le fichier syslog.....	65
Figure 75	Changement d'adresse MAC dans le fichier syslog	66
Figure 76	Configuration du client Nessus.....	68
Figure 77	Trame ARP (taille en bytes).....	69
Figure 78	Options de Winarp_sk	70
Figure 79	Options de Winarp_tcom	71
Figure 80	Lancement d'ettercap.....	75
Figure 81	Choix d'ettercap.....	75

1 INTRODUCTION

1.1 RÉSUMÉ

Ce travail a pour objectif les intrusions à l'intérieur et à l'extérieur de l'entreprise. Le terme d'*Intrusion Management* englobe quatre types d'activité qui assurent la prévention contre les intrusions (Avoidance, Assurance, Detection) ou les mesures de réparation à prendre en cas d'intrusion réussie (Investigation). Les objectifs de ce travail concernent uniquement certains aspects du management de l'intrusion à savoir : l'étude et la présentation des différentes stratégies d'intrusion à l'extérieur et à l'intérieur de l'entreprise et les aspects de prévention (avoidance) basés sur une authentification forte.

Dans cette optique, une sélection des différentes attaques a été faite dans l'objectif de les présenter sous forme de tutorial destiné à l'apprentissage des futurs ingénieurs. Toujours dans le but de l'apprentissage, un laboratoire a été mis en place pour pouvoir mettre en pratique une partie des attaques présentées dans le tutorial ainsi que des méthodes de détection d'intrusion.

Après l'approche théorique, il a fallu mettre en place des bancs de test pour pouvoir appliquer certaines des attaques traitées dans la partie théorique avec une priorité sur les attaques de niveau 2. Puis il a fallu trouver le moyen de mettre en place des méthodes de détection de ces attaques. S'ajoute une partie d'investigation sur les attaques.

1.2 MOTS CLÉS

Prévision, Intrusion, Détection, Investigation

1.3 AVANT PROPOS








Ce document a été partagé en plusieurs parties mais à chacune d'elles, une partie théorique introduit le sujet et pour certaines d'entre elles, une partie pratique a été réalisée.

Ce document traite différents types d'attaques qui ont été classées selon leur catégorie. Pour pouvoir réaliser la présentation de ces attaques, une partie du temps a été consacré à la recherche d'informations sur chacune des attaques.

Pour mettre en évidence l'aspect des attaques, des bancs de test ont été réalisés sur certaines attaques, ceci ayant été effectué uniquement s'il y a une partie pratique associée à la partie théorique.

À chacune des attaques testées, une solution théorique ou pratique est proposée de manière à comprendre comment ces solutions viennent se greffer au réseau déjà existant ou à l'implémentation des protocoles informatiques.

1.4 CONVENTION DES ILLUSTRATIONS

	Utilisateur		Routeur (symbole Cisco)
	Serveur		
	Hacker		Switch (symbole Cisco)
	Analyseur		Hub (symbole Cisco)

1.5 MATÉRIEL

Le matériel employé durant ce travail de diplôme se compose de :

- Une machine Windows 2000.
- Une machine Windows XP.
- Deux machines Linux Debian.
- Trois machines de laboratoire employées comme analyseurs avec *Ethereal*.
- Deux switches Cisco Catalyst 1900.

2 LA RECHERCHE D'INFORMATIONS

2.1 INTRODUCTION

Avant toute attaque, qu'elle soit interne ou externe, il faut d'abord passer par la phase de la prise d'informations : lieu, adresse IP, type d'OS, etc ... Tout comme un voleur ne cambriolera pas une maison sans avoir repéré les lieux et s'être informé sur le système de détection. C'est après cette prise d'informations que la stratégie d'attaque sera établie.

2.2 MÉTHODES DE RECHERCHE

Pour parvenir à trouver des informations sur une entreprise ou un nom de domaine il y a plusieurs moyens, dont :

- Les annuaires Whois
- Nslookup
- Traceroute

Ces annuaires contiennent des informations sur les entreprises ou les domaines, qui ont été fournies lors de l'enregistrement aux organismes responsables. On peut les consulter sur Internet ou par des logiciels. Par exemple, pour la Suisse, il faut s'adresser à Switch (<http://www.switch.ch>). *Nslookup* consiste à interroger un serveur DNS ou un serveur de noms de domaine à l'aide d'un client *nslookup* disponible sur toutes les plates-formes. Les logiciels traceurs de route comme *traceroute*, *visual route* et d'autres, consistent à établir la topologie entre la personne possédant le programme et une cible.

Pour plus d'informations sur ce qui précède, consultez le travail de présentation de M. Andrades et moi-même concernant l'identification d'adresses IP (référence [RL 5]), où vous trouverez les explications sur ces méthodes de recherche.

2.3 SOCIAL ENGINEERING

Il s'agit d'une méthode qui n'a pas recours au logiciel (pour une fois) : les informations vont être collectées directement auprès des utilisateurs par des stratégies de persuasion. Il y a différentes manières d'entrer en contact avec les utilisateurs (référence [RW 1]).

2.3.1 PAR TÉLÉPHONE

C'est la méthode la plus courante ; la personne appelant aura préparé son texte et son personnage de façon à être le plus crédible possible. Pour augmenter la crédibilité, des bruits de fond peuvent être ajoutés comme les bruits des collègues de bureau. Les informations les plus soutirées sont les mots de passe.

2.3.2 PAR COURRIER

Par courrier il est possible de recevoir une lettre au format réalisé avec logo, adresse, numéro de téléphone et avec, comme adresse de retour, une boîte aux lettres d'une société fictive.

2.3.3 PAR CONTACT DIRECT

C'est une méthode très difficile pour le hacker mais encore plus difficile pour la victime qui doit se rendre compte qu'il s'agit d'une personne mal intentionnée du fait que la tenue vestimentaire et l'attitude influenceront la victime.

2.3.4 CONTRE MESURE

Nous avons vu au chapitre 2.2 page 11 qu'il était possible de trouver des informations sur les entreprises. Sur la Figure 1 on peut voir les informations liées à l'école d'Yverdon, provenant d'une recherche sur annuaire Whois du site www.ripe.net. On peut aussi remarquer qu'il y a des noms de contact comme le directeur et le responsable technique (référence [RL 3]).

```
%REFERRAL START
whois: This information is subject to an Acceptable Use Policy.
See http://www.switch.ch/id/terms/aup.html

Domain name:
eivd.ch

Holder of domain name:
Ecole d'Ingenieurs du Canton de Vaud
Christian Kunze
Direction
route de Cheseaux 1
CH-1400 Yverdon-les-Bains
Switzerland

Technical contact:
Ecole d'Ingenieurs du Canton de Vaud
Michel Burnand
Informatique
route de Cheseaux 1
CH-1401 Yverdon
Switzerland

Name servers:
eivdns.eivd.ch [193.134.216.150]
scsnms.switch.ch [130.59.1.30]
scsnms.switch.ch [130.59.10.30]

Date of last registration:
09.10.1997

Date of last modification:
20.11.2000
%REFERRAL END
```

Figure 1 Information sur l'EIVD

Les noms disponibles sur ces annuaires devraient être des noms fictifs de telle sorte que, lorsqu'une personne appelle ou se présente au guichet d'informations dans l'intention de parler au responsable nommé X, cela devrait éveiller les soupçons de la victime et du coup engendrer de la méfiance.

Pour pallier ce phénomène, il faudrait sensibiliser le personnel de l'entreprise et lui apprendre à dialoguer sans révéler des données importantes en lui montrant toutes les techniques et scénarios possibles, car un hacker ne se contentera pas d'en appliquer une mais il organisera tout un scénario qui pourrait commencer par un coup de téléphone et par la suite un rendez-vous et ainsi de suite.

2.4 LES SCANNERS

2.4.1 LES SCANNERS D'ADRESSES

Les scanners permettent à un utilisateur de connaître tous les hôtes actifs dans un réseau, en employant la méthode de scrutation d'adresse IP qui consiste à envoyer des requêtes ICMP à toute la plage d'adresses du réseau, puis il suffit d'attendre lesquelles de ces adresses répondent aux requêtes. Si ce test est effectué sur plusieurs jours et à des intervalles d'heures, il est possible de distinguer entre les serveurs et les postes de travail. (A moins que le test ne soit fait à l'EIVD, les postes restent constamment allumés). Cette méthode de recherche est efficace dans une entreprise, mais à l'extérieur de celle-ci elle dépend de la politique de l'entreprise, car si celle-ci fait du NAT il ne sera pas possible de tester les adresses privées.

2.4.2 LES SCANNERS DE PORTS

Les scanners de ports permettent de connaître lesquels des services sont actifs sur la machine. Le scanner va envoyer un message sur chaque port et en conséquence de la réponse le scanner détermine s'il est actif ou pas. Il faut utiliser un scanner qui puisse tester chaque port par les protocoles TCP et UDP. M. Reis a détaillé dans son travail de diplôme le fonctionnement des scanners (référence [RL 4]).

2.4.3 LES SCANNERS DE VULNÉRABILITÉ

Les scanners de vulnérabilité possèdent une base de données de toutes les vulnérabilités des systèmes et des attaques possibles, de telle manière à informer sur toutes les vulnérabilités du système sondé. Il y a différents scanners de vulnérabilité (payants et non payants), dont un seul sera examiné. Ce scanner est *Nessus* (chapitre 5.1 page 67) qui est un programme non payant fonctionnant avec un modèle client-serveur.

Essais

Les essais vont être effectués depuis un client *Nessus* d'une machine Windows comme celui de la Figure 76 à la page 68. Dans le menu 'Plugins' du client il y a différents types de failles à tester, notamment : les DoS, les backdoors, les serveurs FTP et bien d'autres. Pour les essais, toutes les options seront sélectionnées de façon à tout tester.

Linux

Ce premier essai est réalisé sur une machine Linux avec la distribution Debian, ayant comme services un serveur de types SSH, FTP,SMTP, HTTP et HTTPS. Lors du lancement, le programme va scanner tous les ports de la machine pour connaître les services qui sont disponibles, puis les essais de vulnérabilité commencent : le test a duré environ 55 minutes. Le résultat de l'attaque est visible sur la Figure 2, mais on peut constater que ce n'est pas très parlant. Il est possible de sauver ce test en sélectionnant le troisième bouton et en indiquant le type de format : HTML, LaTeX, ASCII ou HTML avec graphiques.

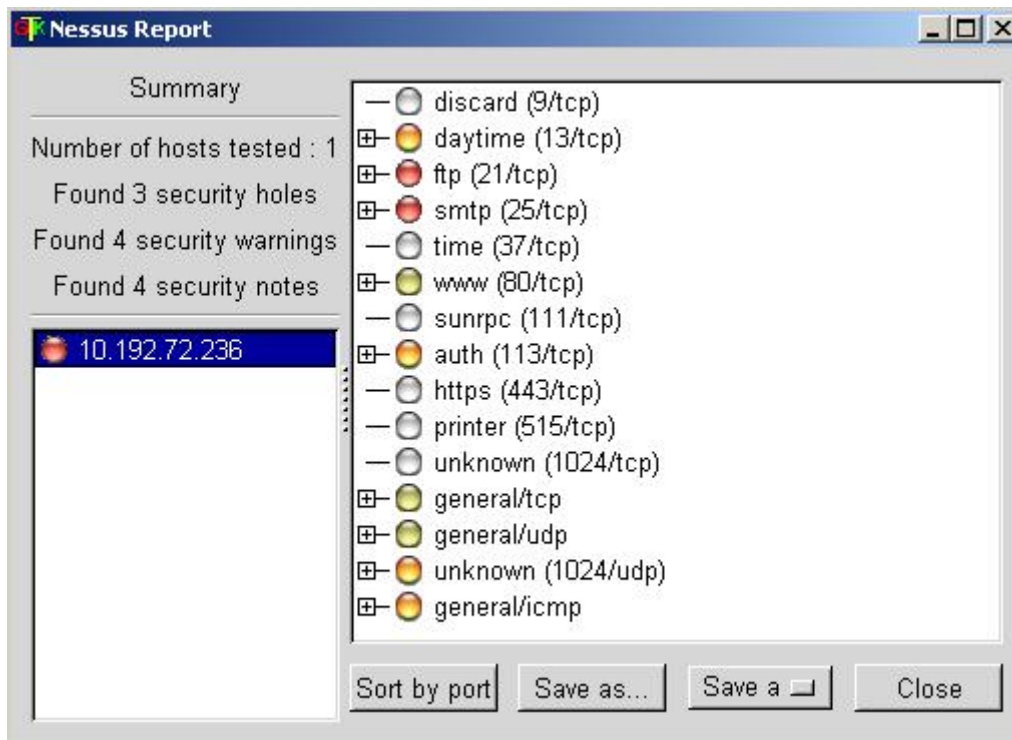


Figure 2 Résultat de l'analyse de la machine Linux

Il est préférable de sauver sous le format HTML avec graphiques, de cette manière il est possible d'analyser graphiquement les vulnérabilités du système. Sur la Figure 3, on peut voir le résultat graphique en mode secteur et l'on peut constater que 16% des vulnérabilités sont à haut risque, ceci provient des serveurs FTP et SMTP. Sur cette page HTML sont aussi expliqués le type d'attaque des vulnérabilités et les solutions. Voici ce qui est dit au sujet du serveur FTP : il était possible de mettre hors service le serveur FTP en faisant 3000 connections sur celui-ci : un attaquant peut employer ce défaut pour empêcher ce service de travailler correctement. Et la solution à ce problème est de télécharger la mise à jour du serveur.

Ainsi, une telle explication est faite à chaque vulnérabilité détectée. Pour plus d'informations sur ce test, veuillez consulter la référence [RW 4].

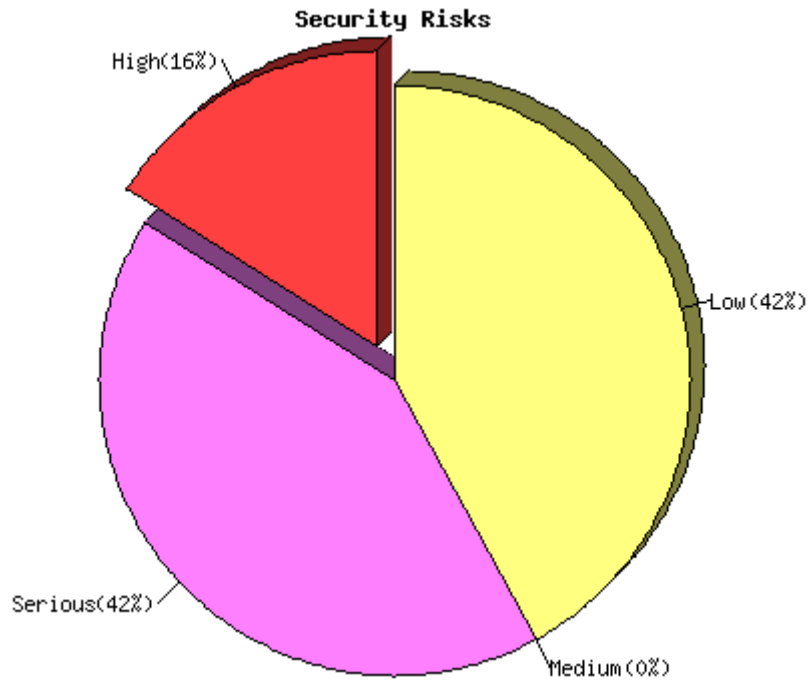


Figure 3 Résultat graphique de l'analyse de la machine Linux

Windows XP

Après avoir testé un serveur Linux, il serait aussi intéressant de tester de la même manière un poste utilisateur de dernière génération des produits Microsoft, c'est-à-dire Windows XP. Sur la Figure 4, on peut voir le résultat graphique de l'analyse.

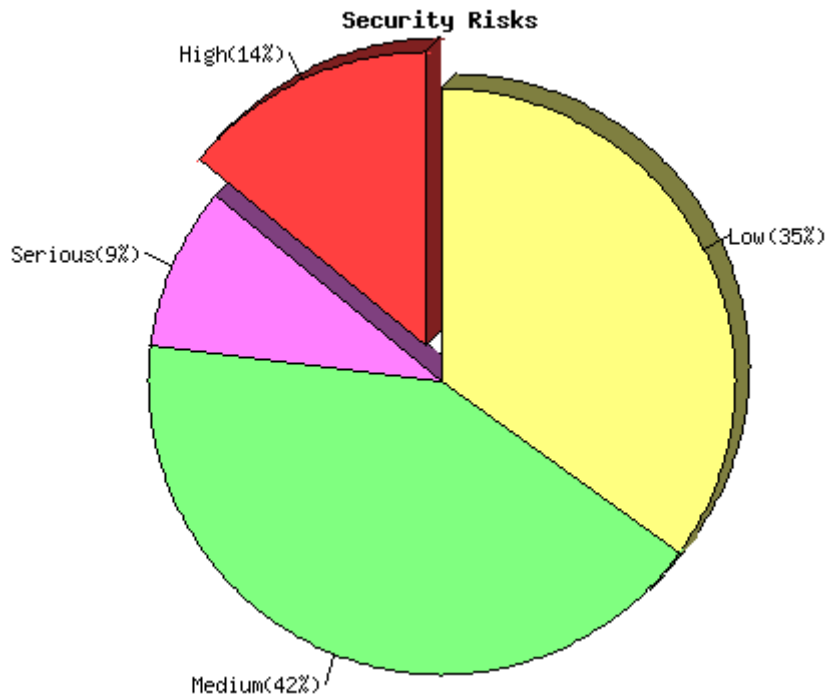


Figure 4 Résultat graphique de l'analyse de la machine Windows XP

Ce haut risque de vulnérabilité de 14% provient du port 139 netbios-ssn, qui informait : qu'il est possible d'enregistrer dans l'hôte une session nulle, le concept d'une session nulle doit fournir un nul 'username' et un mot de passe 'nul', qui accorde l'accès 'guest' à l'utilisateur ou bien qu'il était possible de se loguer avec les combinaisons 'administrator/', 'administrator/'administrator', 'guest/' ou 'guest/'guest'.

Windows XP avec Firewall

Cet essai a été réalisé sur la même machine avec le firewall de Windows XP et avec un firewall disponible gratuitement sur Internet qui est ZoneAlarm. Avec ces deux firewalls, les résultats obtenus sont les mêmes et on constate, sur la Figure 5, qu'aucun problème n'est détecté.

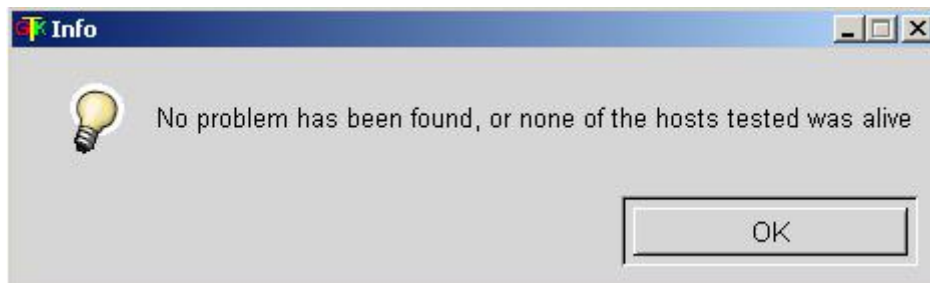


Figure 5 Résultat de l'analyse de la machine Windows XP avec les deux firewalls

Le firewall de Windows XP ne possède pas beaucoup d'options de configuration. Pour configurer le firewall il faut sélectionner le menu 'connexion réseau' puis l'interface et ensuite le bouton 'propriétés' : à ce stade, il est possible de configurer les adresses IP et, sur cette même fenêtre, il y a un menu 'propriétés avancées' où se trouve le menu du firewall. Il est possible d'activer des services comme les serveurs FTP, Telnet, HTTP et d'autres, et de configurer les actions sur le protocole ICMP comme, par exemple, ne pas répondre aux requêtes ICMP (pings) : c'est tout ce qui est configurable. Par défaut rien n'est activé ; cela peut paraître normal qu'il n'y ait pas grand chose de configurable du fait que ce produit est destiné à un large public et dans ce dernier il n'y a qu'un faible pourcentage qui s'y connaisse en matière de protocole réseau.

Le choix du firewall ZoneAlarm provient du fait qu'il était gratuit et qu'il est un des plus répandus. Pour la configuration, il faut, lors de l'installation, spécifier son niveau : de débutant à expert dans ce domaine. Puis le menu est modifié en fonction de son niveau.

Conclusion

Ce genre de test est important pour un administrateur réseau (pour autant que le serveur soit mis à jour régulièrement) afin de connaître les vulnérabilités des machines, des serveurs et des éléments de réseau de toute l'entreprise.

3 INTRUSION

3.1 DÉNI DE SERVICE (DOS)

Les attaques par déni de service (Denial of Service, DoS) ont pour même but de rendre indisponible la victime ou le service de la victime ou de rendre inopérant tout un réseau. Ce genre d'attaques peut être exécuté à différents niveaux du modèle OSI. Ces attaques vont profiter de la faiblesse de l'implémentation de différents protocoles ou profiter d'erreurs de programmation des applications. Seule une partie de ces attaques sera présentée dans ce document, de plus le tutorial de M. Reis traite les attaques de type 'flooding' (référence [RL 4]).

3.1.1 DOS DE NIVEAU 2

3.1.1.1 DoS sur un utilisateur

En se servant de l'aide d'un switch, il est possible de rendre une machine inaccessible au réseau de l'entreprise, du fait que le switch associe l'adresse MAC des machines avec les ports. Lors de la mise en marche d'un switch, celui-ci va utiliser la première trame avec une adresse MAC source inconnue émise sur chaque port pour pouvoir l'associer au port du switch : cette association est gardée dans une table CAM du switch (Figure 6). Selon le mode de configuration du switch, cette première trame est transmise sur le port de Uplink ou sur tous les ports (comme un hub).

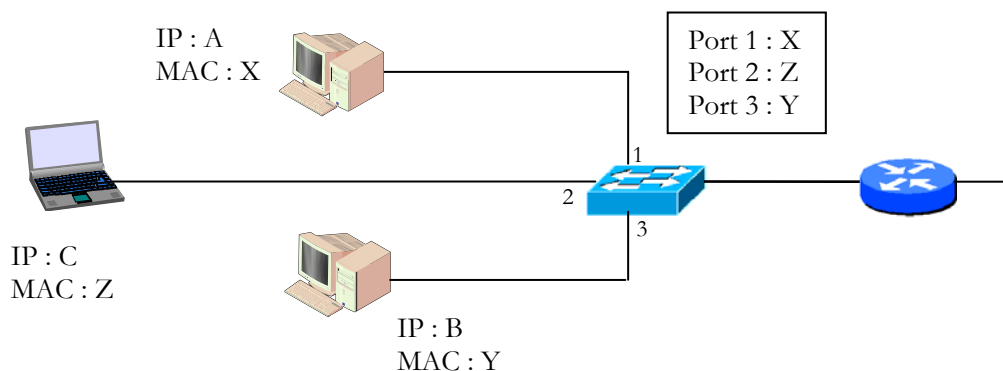


Figure 6 Réseau sans DoS

Pour pouvoir provoquer un déni de service, il faut générer des trames avec l'adresse MAC de la victime sur un port différent du switch de celui où la victime est connectée, pour faire croire au switch que la victime a déplacé sa machine, comme l'illustre la Figure 7. La victime ne pourra plus accéder (ou partiellement) au réseau car le switch a associé son adresse MAC avec un autre port (celui du hacker).

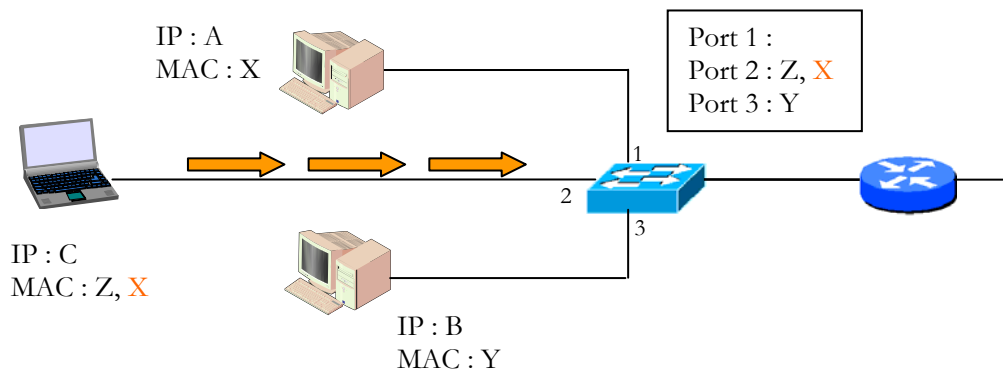


Figure 7 Réseau avec DoS

Partie pratique

Avant toute chose, il va falloir connaître l'adresse MAC de la victime et pour cela il y a plusieurs méthodes : on peut y parvenir en lui envoyant un ping et avec un analyseur on retrouve l'adresse MAC et si on a pas d'analyseur on envoie toujours un ping et on va regarder dans le cache ARP de la machine par l'instruction `arp -a` (idem sous Windows et Linux).

Puis on va pouvoir changer l'adresse MAC du système Linux : pour cela il va falloir quelques lignes de commandes (en étant root sur la machine, référence [RW 7]). Voilà les instructions :

```
ifconfig eth0 down    démonte l'interface eth0 ; selon certaines distributions de Linux
                      il est possible d'utiliser ifdown eth0
ifconfig eth0 hw ether xx:xx:xx:xx:xx spécifie l'adresse MAC
ifconfig eth0 up      remonte l'interface (ifup eth0)
```

Une fois l'adresse MAC changée, les essais vont pouvoir commencer. L'architecture du réseau sera comme à la Figure 8.

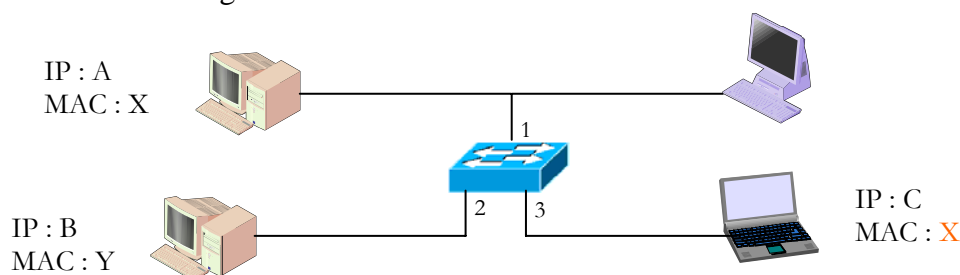


Figure 8 Topologie du banc de test de DoS sur un utilisateur

Le premier essai se fait en envoyant des trames ICMP (pings) depuis la machine ayant l'adresse IP C sur la machine IP B, de telle manière à ce que dans la table d'adressage du switch a associé l'adresse MAC X sur le port 3. Sur la machine ayant l'IP A, des trames ICMP sont aussi envoyées à la machine ayant l'adresse IP B. Les trames sont envoyées (par les deux systèmes) à une cadence d'environ un message par seconde, par conséquent le switch arrive à gérer ces messages et à les envoyer au bon destinataire. La victime ne voit rien et il n'y a pas de déni de service.

Alors il va falloir envoyer un flux plus rapide de messages : pour cela il y a la commande `ping -f` qui envoie des messages mille fois plus rapidement que la commande normale. Dans ce cas de figure, le switch n'arrive plus, ou presque plus, à modifier sa table d'adressage du fait de la grande quantité de données émises sur le port du hacker. Dans ce cas de figure, il se peut que les messages de retour n'arrivent pas au bon destinataire, cela étant indépendant du destinataire. En effet la victime du déni de service n'a reçu en retour qu'environ 4% des messages envoyés. Dans ce cas de figure, le déni de service est atteint (partiellement) car même si une partie des messages lui parviennent lors d'une connexion TCP (ou autre) il ne faut pas qu'il y ait un message qui se perde lors de la négociation de l'ouverture, sous peine de recommencer. La Figure 9 montre une connexion FTP sur un serveur, dont la victime subissait un déni de service, et sur plusieurs tentatives, aucune n'ayant abouti. La Figure 10 montre les trames FTP qui ont été émises par la victime et on constate qu'elle n'en a reçu aucune. Mais ceci est un exemple, il se pourrait qu'une fois la connexion avec le serveur aboutisse.

```
C:\>ftp
ftp> open 10.192.72.236
> ftp: connect :Connection timed out
ftp>
```

Figure 9 Connexion FTP avec DoS

14	10.512225	10.192.73.119	10.192.72.236	TCP	2763	> ftp [SYN]
23	13.432703	10.192.73.119	10.192.72.236	TCP	2763	> ftp [SYN]
36	19.451900	10.192.73.119	10.192.72.236	TCP	2763	> ftp [SYN]

Figure 10 Trame de la connexion FTP avec DoS

3.1.1.2 DoS sur un routeur

En envoyant des réponses ARP en destination d'un routeur on peut obtenir un déni de service sur tout le réseau, car le routeur n'est plus accessible. Ces réponses ARP qui sont envoyées au routeur contiennent une adresse MAC (différente de celle du routeur) associée à l'adresse IP du routeur. Le routeur va accepter ces réponses ARP pour éviter une tempête de messages à enregistrer. Lorsque les machines vont faire des requêtes ARP pour connaître l'adresse MAC du routeur, celui-ci va leur répondre avec une mauvaise adresse ce qui provoquera un déni de service (référence [RW 8]).

Partie pratique

Cet essai va être fait sur un routeur Cisco 2500 ayant un IOS de juillet 1999, la topologie du banc de test se trouve sur la Figure 11. Pour forger ces messages réponses ARP il a fallu employer le programme `winarp_sk` (chapitre 5.2.1 page 69), la commande utilisée était :

```
winarp_sk -m 2 -D 00-00-0c-09-86-2d -S aa-aa-aa-aa-aa-aa
-F aa-bb-cc-bb-bb-bb -s 10.192.72.10 -T 00-00-0c-09-86-2d
-d 10.192.72.10
```

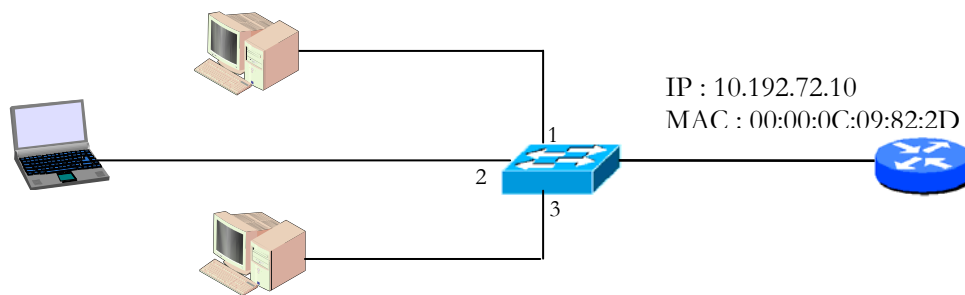


Figure 11 Topologie du banc de test de DoS sur un réseau

Après deux réponses ARP envoyées au routeur, celui-ci accepte l'adresse MAC. Sur la Figure 12 on peut voir que le routeur affiche sur le terminal un message de dupliques d'adresse sur l'adresse IP de l'interface du routeur (10.192.72.10) et on peut voir que l'adresse MAC aa-bb-cc-bb-bb-bb est associée à l'adresse IP 10.192.72.10. Les machines du réseau qui ont dans leur cache ARP l'adresse MAC du routeur peuvent toujours y accéder comme si rien ne s'était passé, mais lors d'une requête ARP pour connaître l'adresse MAC du routeur, celui-ci ne répond pas à la requête. Dans ce cas, plus aucune machine ne peut sortir du réseau.

```
Router#show arp
Protocol Address      Age (min)  Hardware Addr  Type  Interface
Internet 10.192.72.10    -          0000.0c09.862d ARPA  Ethernet0
Router#
1d03h: %IP-4-DUPADDR: Duplicate address 10.192.72.10 on Ethernet0, sourced by aa
bb.cbcb.bbbb
Router#show arp
Protocol Address      Age (min)  Hardware Addr  Type  Interface
Internet 10.192.72.10    -          aabb.cbcb.bbbb ARPA  Ethernet0
```

Figure 12 Cache ARP du routeur, acceptation de l'adresse MAC

Comment se protéger

Pour empêcher cette attaque, il est possible de mettre l'adresse MAC du routeur en mode statique (sur le routeur). Une autre façon consiste à mettre à jour l'IOS du routeur. Sur la Figure 13, un test a été réalisé sur un routeur Cisco 2500 avec un IOS d'août 2002 et l'on constate qu'il signale un problème et qu'il ne prend pas en compte les réponses ARP.

```
QoS_Router_2#sh arp
Protocol Address      Age (min)  Hardware Addr  Type  Interface
Internet 10.192.72.10    -          0010.7b38.6224 ARPA  Ethernet0
QoS_Router_2#
4d04h: %IP-4-DUPADDR: Duplicate address 10.192.72.10 on Ethernet0, sourced by aa
cc.ddbb.bbbb
4d04h: %IP-4-DUPADDR: Duplicate address 10.192.72.10 on Ethernet0, sourced by aa
cc.ddbb.bbbb
QoS_Router_2#sh arp
Protocol Address      Age (min)  Hardware Addr  Type  Interface
Internet 10.192.72.10    -          0010.7b38.6224 ARPA  Ethernet0
```

Figure 13 Cache ARP du routeur, refus de l'adresse MAC

3.1.2 DOS DE NIVEAU 3

3.1.2.1 Ping de la mort (Ping of Death)

Le principe est d'envoyer à la victime un paquet IP surdimensionné pour que le système de la victime se mette en attente ou crashe, car le système n'est pas capable de gérer le surplus de données. Le protocole IP a défini qu'une trame IP ne doit pas dépasser 65535 octets et c'est en envoyant une trame IP supérieure à 65535 octets que l'on obtient la chute de la victime (référence [RW 9]).

Pour mieux comprendre comment cela fonctionne, voyons tout d'abord comment est structuré un paquet IP et comment celui-ci est envoyé dans un réseau. Sur la Figure 14 on peut voir l'en-tête d'un paquet IP.

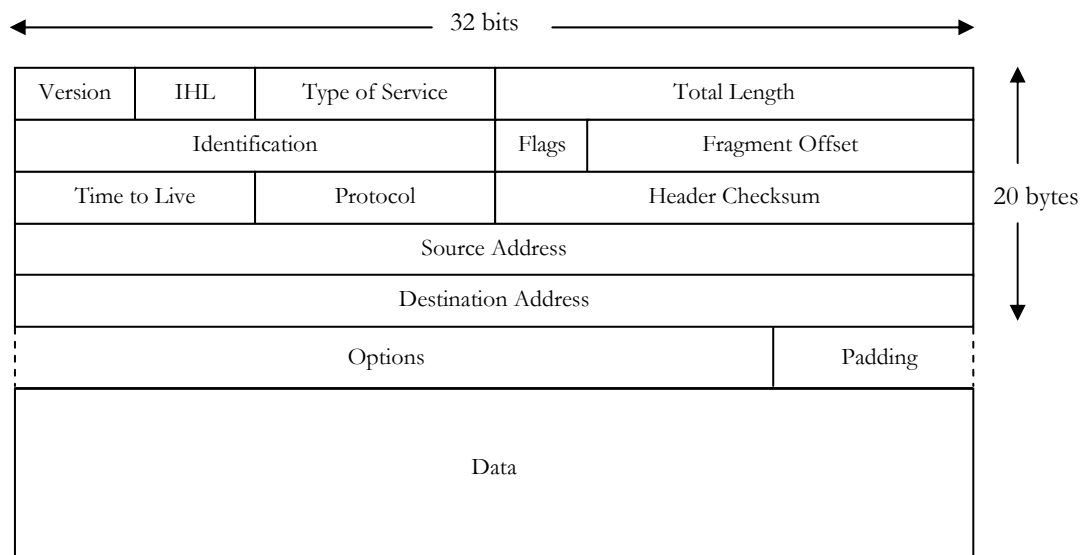


Figure 14 En-tête IP

Tous les champs ne seront pas expliqués dans cette partie (pour plus d'informations : [RL 1] page 249), parmi les champs de l'en-tête il y en a certains qui sont importants lors de l'envoi des paquets. :

Total Length : indique la longueur totale du datagramme en y incluant la longueur de l'en-tête. Ce champ permet de distinguer la partie information utile des bits de bourrage (Padding) en faisant la comparaison avec le champ IHL (16 bits).

Flags : ce champ est constitué de trois bits (0/DF/MF) qui sont utilisés lors de la fragmentation des paquets.

Bit 0 réservé et a comme valeur 0

Bit DF autorisation de fragmenter (oui = 0 et non = 1)

Bit MF indicateur du dernier fragment (dernier = 0 et encore d'autres = 1)

Fragment Offset : indique la position du premier octet dans le datagramme total (non fragmenté), indication par multiples de 8 (13 bits).

Grâce à ces trois champs, un récepteur arrive à reconstituer un paquet IP qui a été fragmenté. La fragmentation des paquets IP est nécessaire lorsque ceux-ci dépassent 1500 octets car les paquets du niveau 3 (réseau, IP) sont transmis à la couche inférieure (niveau 2, liaison). Cette couche 2 permet l'envoi de messages point à point entre deux éléments réseau. Sur la Figure 15 on peut voir la trame Ethernet.

8	6	6	2	46 - 1500	4
Preamble	Dest. Add.	Sour. Add.	Type	Data	FCS

Figure 15 Trame Ethernet (taille en bytes)

Lors de l'envoi d'un ping, le protocole utilisé est de l'ICMP, celui-ci est encapsulé dans la trame IP, comme l'illustre la Figure 16. Lorsque le ping est de grande taille (exemple : `ping -l 60000`) les data de la trame ICMP seront d'une taille de 60000 octets, puis seront ajoutés 8 octets de l'en-tête ICMP, par conséquent la trame IP sera grande de 60028 octets.

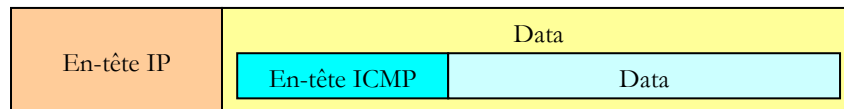


Figure 16 Encapsulation de l'ICMP dans de l'IP

Comme dit précédemment, la trame Ethernet ne peut accueillir que des data d'une longueur maximum de 1500 octets, donc la couche 3 va devoir fragmenter la trame à transmettre en trames de 1500 octets, comme l'illustre la Figure 17. Le champ Total Length indiquera la longueur de chaque trame IP : dans cet exemple cette valeur sera toujours de 1500 octets sauf pour la dernière trame. Le champ Flags sera toujours 001 sauf pour la dernière trame qui vaudra 000 indiquant que c'est le dernier paquet. Le champ Fragment Offset part depuis 0 et s'incrémente de 185 à chaque trame. Cette valeur (185) provient du fait qu'une trame est longue de 1500 octets mais on peut y soustraire 20 octets de l'en-tête IP qui font 1480 et on peut ensuite diviser 1480 par 8 qui font bien 185. De cette manière un peu plus de 40 trames Ethernet seront nécessaires pour transmettre un `ping -l 60000`.

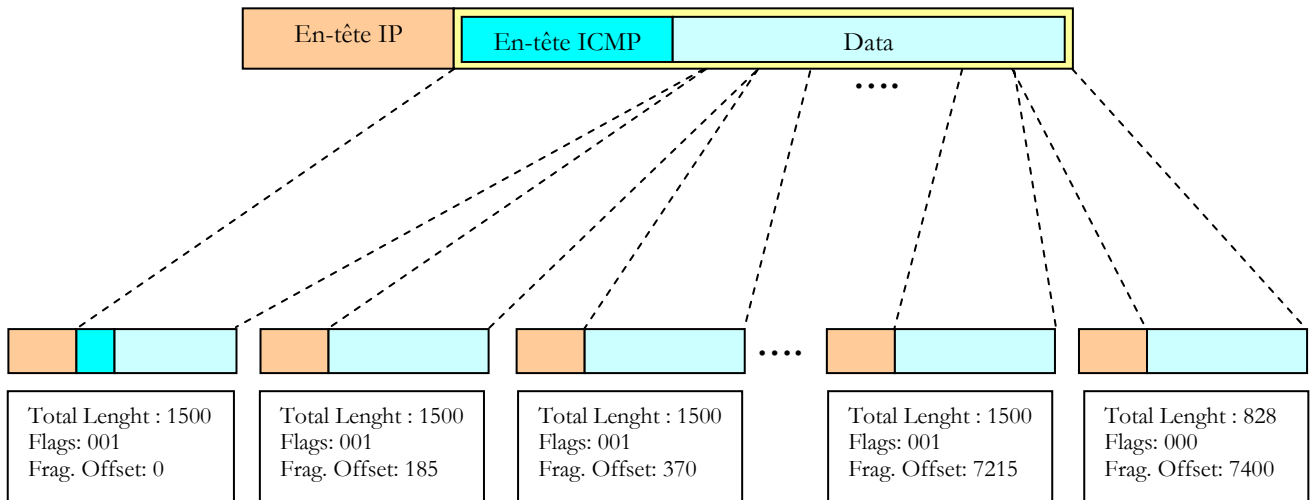


Figure 17 Fragmentation du paquet IP

Lors de la réception de ces trames Ethernet, le système va stocker dans un 'buffer' les informations contenues dans le champ Data pour reconstituer la trame IP et à l'aide du champ Fragment Offset il va pouvoir remettre les paquets dans le bon ordre (si nécessaire).

Donc, pour pouvoir orchestrer un ping de la mort, il va falloir envoyer des data de la trame ICMP plus grands que 65507 octets ($65535 - 8 - 20$). Si on divise 65535 par 1480 on obtient 44.28 trames, par conséquent seront envoyées 45 trames, donc si à la 45^{ème} trame il y a plus de data que prévu, lorsque le système récepteur, qui stocke les paquets dans un stack, va compter la longueur totale du paquet à l'aide d'un compteur de 16 bits (le comptage maximum du compteur est 65535 qui correspond à $(2^{16}) - 1$), ce compteur finira par buguer et le stack débordera, engendrant un blocage du système.

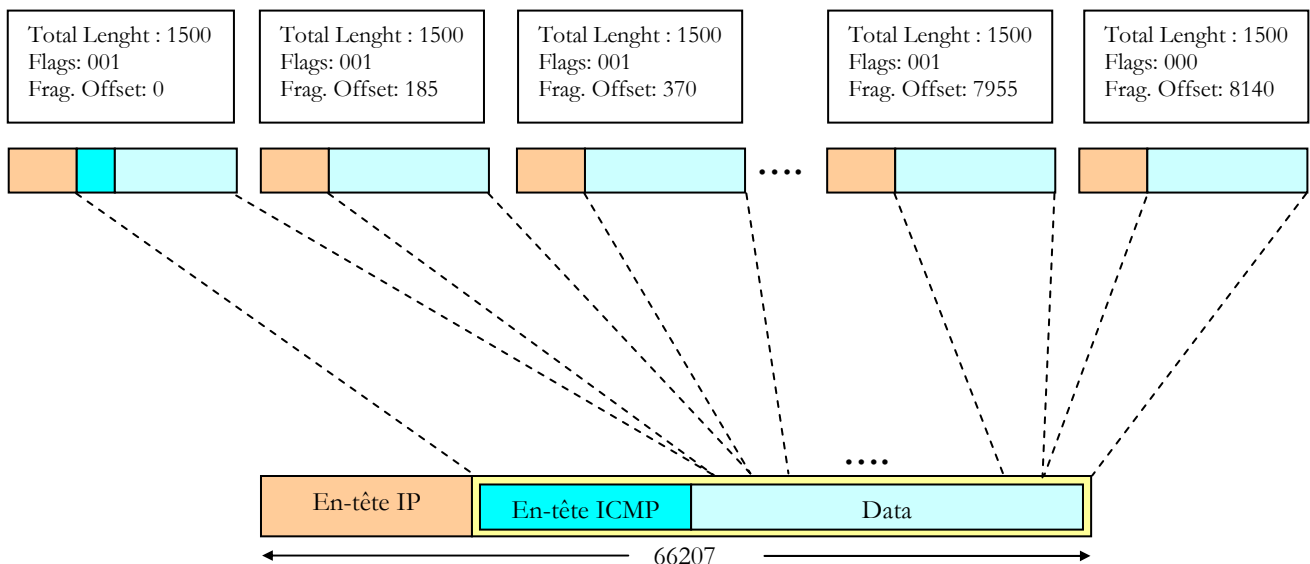


Figure 18 Reconstitution du paquet IP

Cette attaque était très populaire en 1997 mais de nos jours il est moins facile d'envoyer un ping avec une taille supérieure à 65535 octets car les systèmes actuels testent la taille avant d'envoyer le paquet. A la Figure 19, on peut voir un essai d'un ping d'une taille supérieure à 65535 octets qui a été refusé par un système Windows 2000. Mais il n'est pas exclu qu'un programme implémentant les fonctionnalités du ping puisse accepter ce genre de taille.

```
C:\>ping -l 65555 www.monsite.com
Bad value for option -l, valid range is from 0 to 65500.
```

Figure 19 Test d'un Ping of Death

D'autres types de ce genre existent : Jolt, sPing, IceNewk, etc..

3.1.2.2 Land Attack

En faisant une recherche des ports TCP ouverts de la victime, il faut lui envoyer un paquet TCP SYN encapsulé dans une trame IP qui comporte l'adresse IP source et le port TCP source identiques à ceux de la victime (référence [RW 10]).

La trame TCP SYN est la première trame des trois qui doit être envoyée lors d'une ouverture de connexion TCP avec un serveur (Figure 20). Lorsque le serveur va répondre par une trame SYN/ACK, il ne va pas savoir comment traiter cette réponse, du fait que l'adresse et le port source sont ceux du serveur et par conséquent cela provoque le crash ou une mise en attente de la machine.

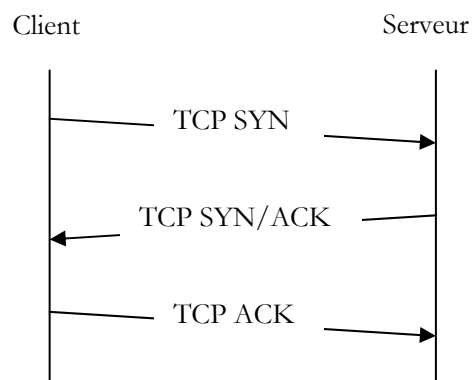


Figure 20 Ouverture d'une connexion TCP

Pour se prémunir de cette attaque, il va falloir installer les correctifs logiciels ou installer un FW. De plus, si l'entreprise dispose d'un réseau interne, un paquet provenant de l'extérieur de l'entreprise ne peut avoir une adresse source qui se trouve à l'intérieur de cette même entreprise.

3.1.3 DDOS

Ce type d'attaque a pour même but de rendre inopérante une machine mais ici le hacker se fait aider par d'autres machines pour exécuter cette attaque, tout cela au même moment. Lorsqu'il y a plusieurs machines comme source de l'attaque, il est plus difficile de se défendre étant donné que ces attaques proviennent de plusieurs adresses IP différentes.

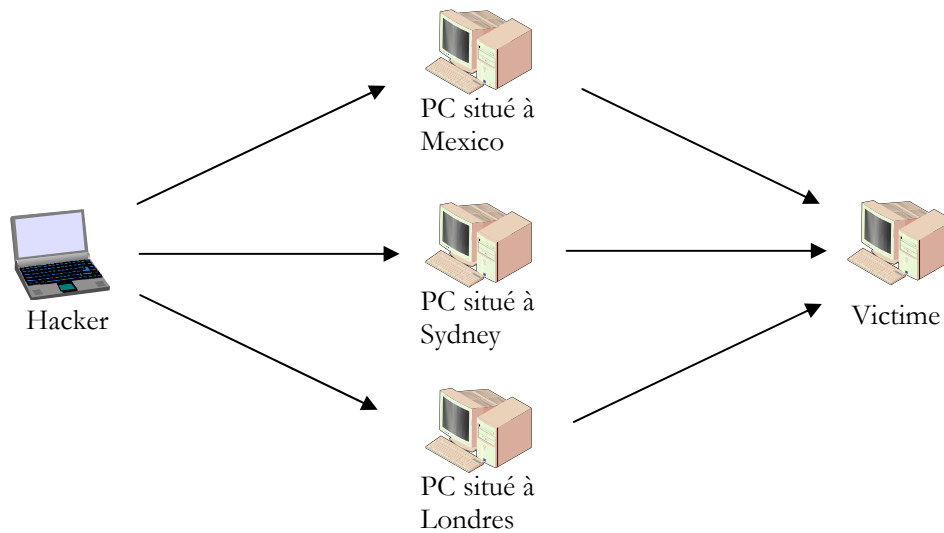


Figure 21 Attaque par DDoS

3.2 SPOOFING

Le terme ‘spoofing’ signifie usurpation d’identité, par conséquent les attaques de cette famille consistent à se faire passer pour une machine dans le but d’obtenir la confiance ou l’approbation de la victime.

3.2.1 ARPSPOOFING

Cette attaque est une attaque de niveau 2 (couche liaison) car elle utilise le protocole ARP, qui est utilisé pour la résolution d’adresse IP en une adresse MAC dans les réseaux Ethernet.

Lorsqu’une machine veut envoyer des messages à un destinataire, il utilise son adresse IP comme adresse destinatrice mais pour envoyer ces messages la machine source doit connaître l’adresse MAC de ce destinataire (si celui-ci se trouve dans le même réseau) ou bien de l’élément réseau qui va devoir acheminer ce message (en général c’est un routeur).

Pour connaître l’adresse MAC du destinataire, la machine source va devoir émettre une requête ARP (en broadcast) en spécifiant l’adresse IP du destinataire et seul le destinataire ou l’élément de réseau concerné va lui répondre en donnant son adresse MAC. Cette réponse est conservée dans le cache ARP qui va contenir toutes ces associations d’adresses pour une durée déterminée afin d’éviter de refaire des requêtes à chaque paquet envoyé. Sur la Figure 22, on peut visualiser un ordinateur accédant à Internet sans être victime d’ARPspoofing.

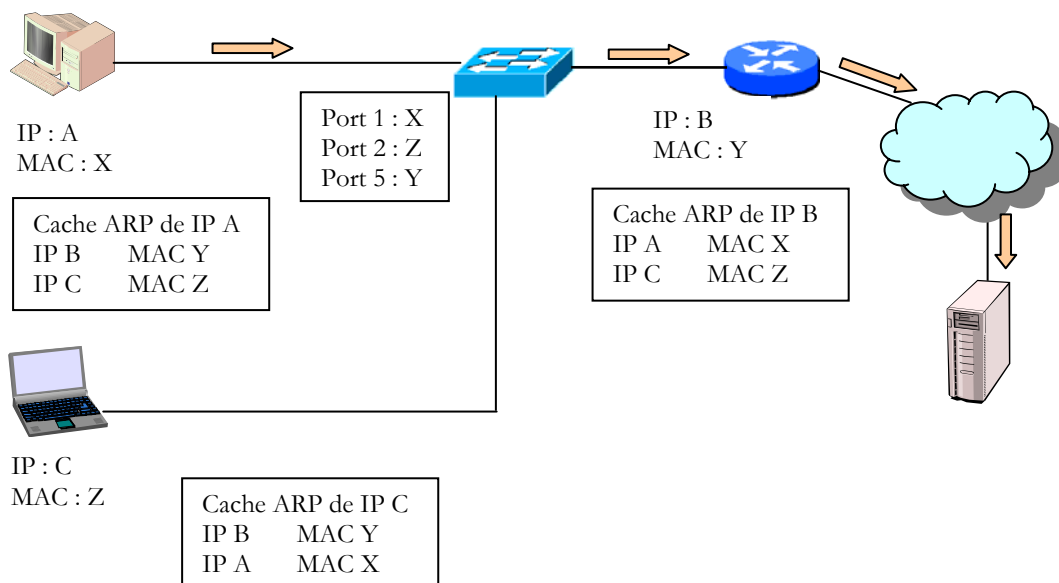


Figure 22 Réseau sans ARPspoofing

Après avoir vu le principe des requêtes ARP, l’ARPspoofing consiste à envoyer des réponses ARP à la victime dans le but de modifier son cache, car le cache peut être modifié sans qu’une requête ait été envoyée. Par conséquent, en envoyant des réponses ARP à la victime et en lui donnant l’adresse MAC du hacker (associée à l’adresse IP du gateway) on va pouvoir rediriger le trafic de la victime sur la machine du hacker, comme l’illustre la Figure 23. Cette attaque est possible sur toutes les machines du sous-réseau (jusqu’au routeur). De plus, il est possible d’empoisonner le cache ARP du gateway pour que les informations de retour passent par la machine du hacker.

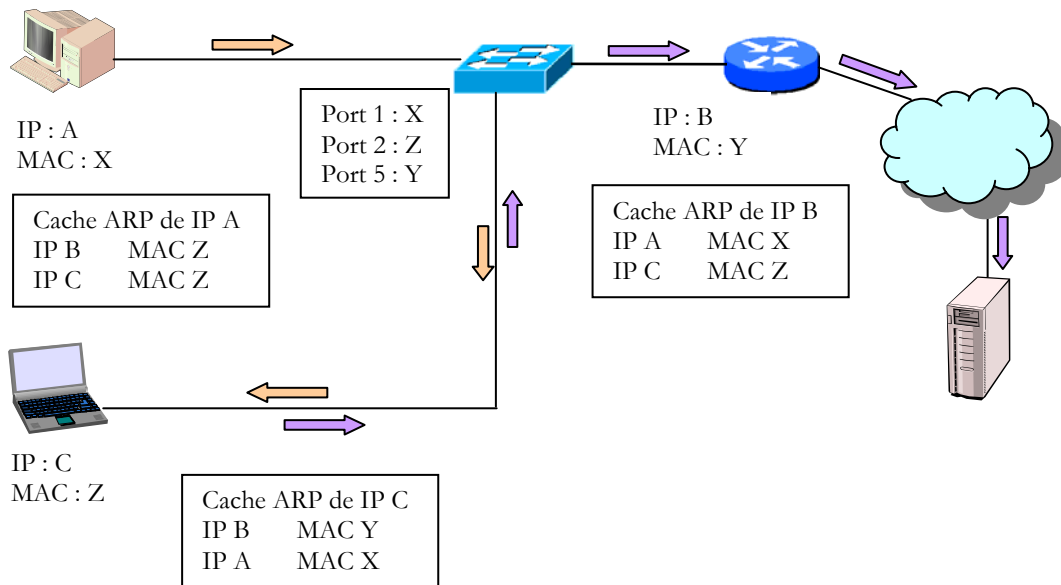


Figure 23 Réseau avec ARPspoofing

Partie pratique

La Figure 24 illustre la topologie du banc de test avec une machine Windows qui fera office de victime et deux systèmes Linux dont un sera un serveur FTP et l'autre le hacker.

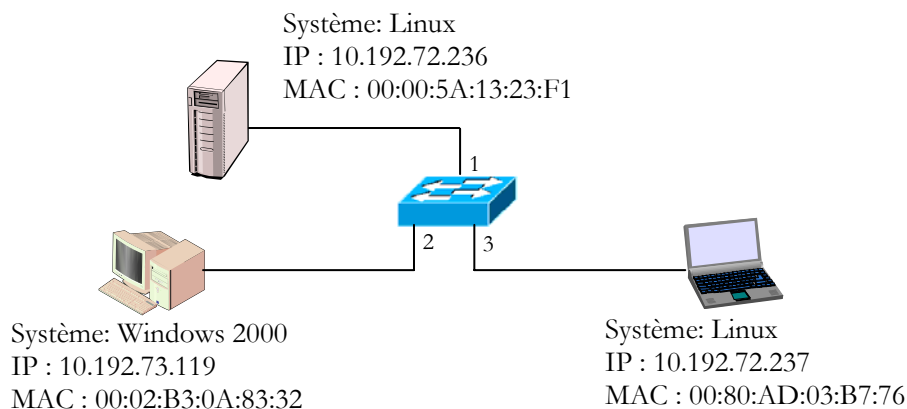


Figure 24 Topologie du banc de test de l'ARPspoofing

Dans cette topologie, la victime fait régulièrement des connexions FTP sur le serveur et l'on peut voir, sur la Figure 25, le cache ARP de la (future) victime. Le programme utilisé sera *arp spoof* (chapitre 5.3.1 page 72) et il est lancé par la commande `arp spoof -i eth0 -t 10.192.73.119 10.192.72.236`. Sur la victime, le programme va tout d'abord faire deux requêtes ARP : l'une cherchant l'adresse MAC de la victime, qui lui servira pour lui envoyer les réponses ARP, et l'autre pour connaître l'adresse MAC du serveur FTP qui sera stockée par le programme *arp spoof*, car lorsque le hacker stoppe l'attaque, le programme va envoyer trois réponses ARP à la victime pour lui remettre dans le cache ARP la bonne adresse MAC du serveur de telle manière à ce que la victime ne remarque rien de l'attaque. Sur la Figure 26, on peut voir le cache ARP de la victime qui a été empoisonnée.

```

Interface: 10.192.73.119 on Interface 0x1000003
Internet Address      Physical Address      Type
10.192.72.236        00-00-5a-13-23-f1    dynamic
10.192.72.237        00-80-ad-03-b7-76    dynamic
    
```

Figure 25 Cache ARP de la victime

```

Interface: 10.192.73.119 on Interface 0x1000003
Internet Address      Physical Address      Type
10.192.72.236        00-80-ad-03-b7-76    dynamic
10.192.72.237        00-80-ad-03-b7-76    dynamic
    
```

Figure 26 Cache ARP de la victime empoisonnée

Comment se protéger

Sur la Figure 26, l'on peut remarquer que les adresses qui figurent dans cette table ont été introduites de manière dynamique, donc pour contrer cette attaque la victime pourrait entrer de manière statique la relation d'adresse du default gateway : pour cela il suffit d'insérer la commande suivante :

```
arp -s adresse_IP adresse_MAC
```

Identique sous Windows et Linux

Même si le système subit une attaque de ce genre, le cache ARP prendra l'adresse qui aura été introduite de la manière statique. Malheureusement, il y a des inconvénients à cette méthode : lorsqu'il y a plusieurs machines dans un réseau, ce n'est pas pratique pour l'administrateur réseau qui doit l'appliquer sur toutes les machines. De plus, il y a un autre inconvénient plus important qui est dû au fait que lorsque les machines sont réallumées, elles perdent les informations introduites de manière statique. En outre, les systèmes Windows 2000 sont toujours sensibles à l'attaque, même en mettant en statique (est-ce un Bug ??). Par contre les systèmes Linux et Windows XP sont insensibles à l'attaque (heureusement). Cette méthode n'est pas efficace du fait qu'il ne faudrait pas éteindre les machines (si une coupure de courant survient ?) et, de plus, les machines les plus répandues sont des Windows 2000 et pas encore XP.

3.2.2 IPSPOOFING

Cette attaque est une attaque de niveau 3 (couche réseau). L'IPspoofing permet de cacher la source de l'attaque sur une cible, mais l'inconvénient est que l'adresse IP source de l'attaque a été usurpée, par conséquent les messages de retour de la victime iront à la personne ayant son adresse IP usurpée. C'est pour cela que ce genre d'attaque s'appelle 'attaque à l'aveugle' (Blind Spoofing) mais elle implique une condition importante qui est de pouvoir prédire les numéros de séquences du protocole TCP. Étant donné que les messages ne viennent pas en retour, il faut avant tout que la machine qui a l'adresse usurpée soit mise hors service (DoS).

Pour commencer, il faut changer l'adresse IP de la machine : cela on peut faire simplement en suivant les indications qui suivent (référence [RL 3]).

Windows

Il faut double-cliquer sur l'icône 'Réseau' qui se trouve soit dans le 'Panneau de configuration' soit dans les 'Paramètres' (selon les versions de Windows). Sélectionner le protocole TCP/IP et enfin il ne reste plus qu'à entrer les adresses voulues selon la Figure 27. Puis il est possible de vérifier la configuration par la commande DOS `ipconfig`.

Lorsqu'un changement d'adresse IP s'opère de cette manière, le système Windows va faire une requête ARP sur la nouvelle adresse IP spécifiée pour savoir si cette adresse est déjà utilisée par une autre machine. Si l'adresse est déjà utilisée, la machine qui utilise cette adresse recevra un message lui indiquant un conflit d'adresse IP. Par conséquent, il faut mettre hors service l'autre machine ou adopter une autre méthode pour pallier ce problème.

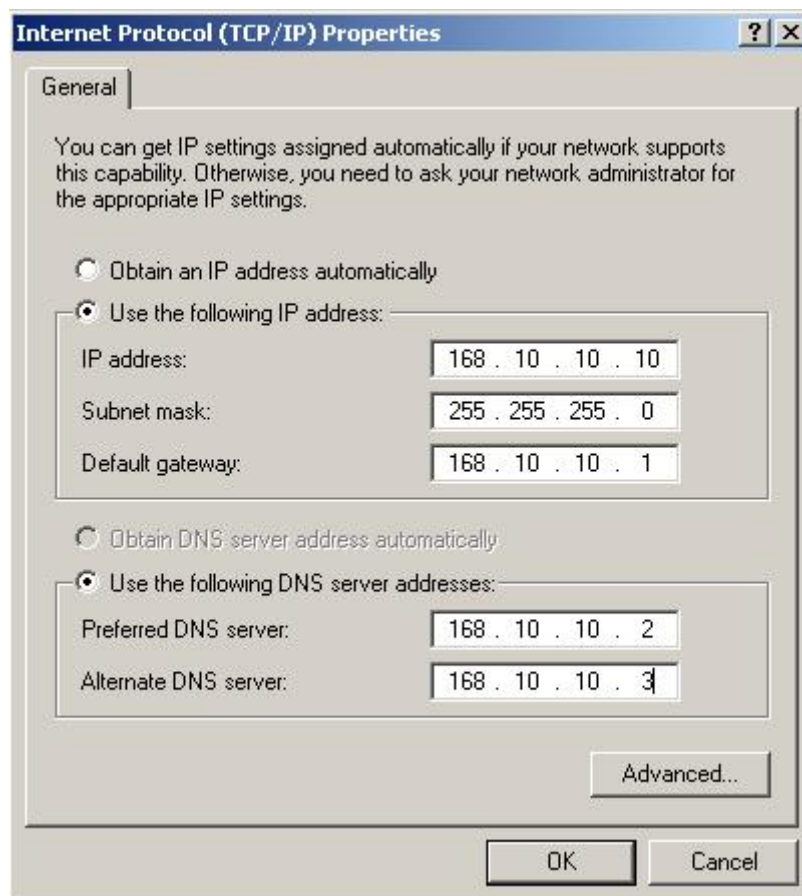


Figure 27 Configuration de l'adresse IP sous Windows

UNIX

Sur les machines UNIX, il va falloir faire appel à la commande `ifconfig`. En tapant uniquement cette commande on obtient la configuration de l'interface réseau. Cette commande est équivalente à `ipconfig` sous Windows. Alors, pour changer l'adresse IP, il suffit d'insérer la commande suivante :

```
ifconfig <interface> X.X.X.X netmask 255.X.X.X broadcast X.X.X.X
```

Le nom de l'interface est inscrit lorsqu'on tape la commande `ifconfig`, mais généralement c'est `eth0`. Lorsque la configuration se fait de cette manière, la table de routage

de l'ordinateur n'est pas conservée, alors il va falloir la reconfigurer (si nécessaire) pour cela il faut utiliser la commande `route add` de cette manière :

```
route add -host 192.168.0.0 eth0           pour une machine
route add -net 192.168.0.0 netmask 255.255.255.0 eth0  pour un réseau
route add default gw 192.168.0.1 netmask 0.0.0.0      pour le gateway
```

Ce mode de configuration est temporaire, car lorsqu'on redémarre le système, la configuration initiale est rétablie, ce qui n'est pas le cas sous Windows et contrairement à Windows, les systèmes UNIX ne cherchent pas à savoir si l'adresse IP est un doublon.

Comme dit précédemment, les paquets de retour ne parviennent pas au hacker mais à la personne qui a son adresse IP usurpée. Il y a deux méthodes pour que les paquets puissent revenir vers le hacker :

Source Routing

Le source routing est une option de l'en-tête IP (Figure 14, page 21). Sans ces options l'en-tête IP à une longueur minimale de 20 octets mais si des options sont implémentées celle-ci sera plus grande.

Le champ option permet d'implémenter certaines propriétés facultatives qui sont :

- Sécurité
- Internet Timestamp
- Record Route
- Source Routing

Ce champ est de taille variable (Figure 28) mais doit être un multiple de 32 bits (qui est rempli par des zéros comme bits de bourrage/Padding).

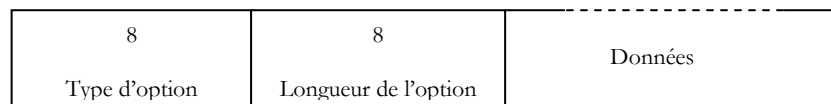


Figure 28 Format du champ option (taille en bytes)

Sécurité

Ce champ permet aux hôtes de spécifier des paramètres de sécurité, de compartimentation, de restriction, de traitement et des groupes utilisateurs fermés. Mais cette option n'est pas, voire très peu utilisée.

Internet Timestamp

Par cette option on va pouvoir enregistrer les marques de temps (durée en millisecondes écoulées depuis minuit GMT) correspondant à l'heure de réception de chaque routeur ainsi que leur adresse IP. Lorsqu'on fait un `ping -s 4`, cette option est utilisée mais est limitée à un maximum de quatre adresses parce que l'en-tête IP est de longueur finie (maximum 65536 octets). Exemple de cette option à la Figure 29 sur le site de l'EPFL.

```
C:\>ping -s 4 www.epfl.ch

Pinging empc19.epfl.ch [128.178.50.92] with 32 bytes of data:

Reply from 128.178.50.92: bytes=32 time=180ms TTL=243
    Timestamp: 10.192.1.1 : 3988013983 ->
                10.192.1.4 : 2488415746 ->
                193.134.216.129 : 2255880194 ->
                130.59.38.6 : 1824988674
```

Figure 29 Test d'un Ping avec l'option Timestamp (-s)

Record Route

Cette option est presque similaire à celle de Timestamp mais n'enregistre pas le temps, donc en faisant un `ping -r` on va pouvoir visualiser uniquement les adresses des routeurs (adresses de l'interface d'entrée) rencontrés sur le parcours, mais, comme énoncé précédemment, le nombre d'adresses est limité : dans ce cas c'est maximum 9.

```
C:\>ping -r 9 www.epfl.ch

Pinging empc19.epfl.ch [128.178.50.92] with 32 bytes of data:

Reply from 128.178.50.92: bytes=32 time=20ms TTL=243
    Route: 10.192.1.1 ->
            193.134.216.130 ->
            193.134.216.2 ->
            130.59.38.6 ->
            130.59.33.238 ->
            130.59.33.65 ->
            130.59.36.26 ->
            130.59.36.49 ->
            192.33.209.33
```

Figure 30 Test d'un Ping avec l'option Record Route (-r)

Source Routing

Lorsque des paquets sont envoyés dans les réseaux, ce sont les routeurs qui sont chargés de trouver le chemin approprié pour acheminer ces paquets. Par cette option il est possible de spécifier par où doivent passer les paquets. Cette spécification doit se faire à la source, c'est pour cela qu'il est possible au hacker de recevoir les paquets en retour. Il y a deux routages à la source :

LSR, Loose Source Routing (routage à la source lâche) : l'expéditeur définit une liste d'adresses IP que les paquets doivent emprunter, mais ceux-ci peuvent passer par d'autres adresses. En d'autres termes, on ne se préoccupe pas des endroits où passent les paquets pour autant qu'ils passent par les adresses spécifiées (type 137).

SSR, Strict Source Routing (routage à la source strict) : l'expéditeur définit une liste d'adresses IP que les paquets doivent emprunter. Si le chemin est inaccessible ou impossible alors les paquets seront supprimés et un message ICMP 'destination inaccessible' sera envoyé à son expéditeur. Dans ce cas les routeurs ne doivent pas influencer la destination des paquets (type 131).

Sur la Figure 31, on peut voir la trame de cette option. Avec cette option le hacker peut rediriger les paquets en sa direction jusqu'à un routeur dont il a le contrôle. Mais la majeure partie des implémentations des piles TCP/IP rejettent cette option à cause de leurs configurations par défaut.

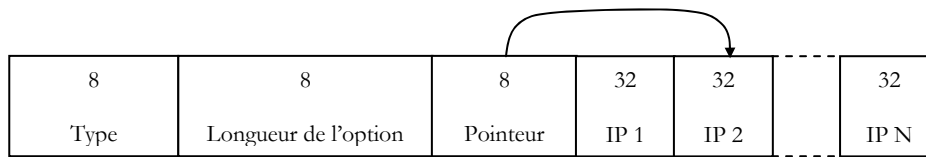


Figure 31 Trame de l'option Source Routing (taille en bits)

Reroutage

Dans cette méthode, le hacker doit aussi maîtriser un routeur. Ainsi il lui suffit d'envoyer des messages de routage RIP aux autres routeurs pour rerouter les messages de retour dans sa direction.

Ces techniques (reroutage ou Source Routing) ne sont plus ou difficilement utilisées, par conséquent il faudra utiliser l'attaque à l'aveugle. Pour cela l'attaque se fera en plusieurs étapes :

- Détermination de l'adresse IP de confiance
- Mise hors service de la machine de confiance
- Prédiction du numéro de séquence TCP

Pour pouvoir déterminer l'adresse IP de confiance, il est possible d'utiliser la commande `showmount -e` qui montre où sont exportés les systèmes de fichiers ou `rpcinfo` qui apporte des informations supplémentaires (référence [RW 11]).

Pour la mise hors service de la machine de confiance, il faut appliquer une des méthodes de déni de service (chapitre 3.1 page 17). Si cela n'est pas fait, lors de l'envoi du message SYN sur la victime celle-ci va répondre par ACK/SYN qui sera envoyé à la machine ayant l'adresse IP usurpée et celle-ci répondra par un RST (reset).

La pile TCP/IP du système d'exploitation peut générer le numéro de séquence TCP de différentes manières :

- Linéaire
- Dépendante du temps
- Pseudo-aléatoire
- Aléatoire selon les systèmes

Pour que cette attaque soit possible, le système de génération doit être linéaire ou dépendant du temps. Pour pouvoir le déterminer il faut faire plusieurs connections TCP sur la victime et analyser ces connections de telle manière à prédire l'ISN (Initial Sequence Number). Le numéro de séquence est codé sur 32 bits et, au démarrage de la machine l'ISN, est initialisé à 1, puis il est incrémenté de 128000 par seconde et de 64000 par connection, de cette manière ce mécanisme évite qu'une ancienne connection vienne perturber une autre connection par un numéro trop proche. Ceci est valable pour les systèmes dépendant du temps (référence [RW 12]).

Lors de la réception de paquets TCP, le numéro d'acknowledge peut être dans trois situations selon la valeur attendue :

- Egal le paquet est accepté
- Inférieur le paquet est supprimé
- Supérieur s'il est dans la limite acceptable par la fenêtre de transmission alors le paquet est maintenu en attente sinon il est supprimé.

Sur la Figure 32 on peut voir les trames lancées sur la victime. Lors de la connection on remarque bien que la réponse de la victime n'atteint pas le hacker. Le flag PSH indique à la couche TCP de donner les données à la couche supérieure.

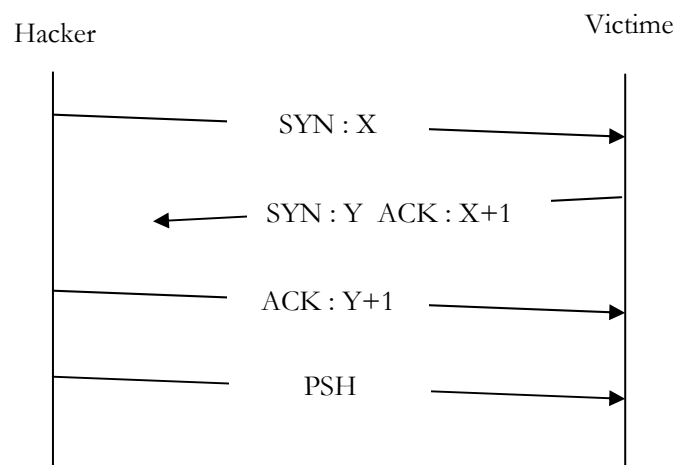


Figure 32 Attaque Ipspoofing

Ce type d'attaque est orchestré lorsque la victime ne se trouve pas dans le même réseau que le hacker : si c'est le cas, il vaut mieux employer une autre méthode que le 'blind spoofing', comme celle consistant à empoisonner le cache ARP de la victime et faire toujours un déni de service sur la machine ayant l'adresse IP usurpée.

Cette attaque est possible lorsque, pour s'authentifier, le protocole utilise l'adresse IP source comme authenticateur ; il serait préférable d'employer d'autres protocoles comme SSH ou à la limite Telnet (mieux vaut l'éviter car le mot de passe est transmis en clair).

3.2.3 DNSSPOOFING

Le principe de cette attaque est de rediriger un internaute sur un autre serveur Web sur lequel il désirait se rendre. Pour pouvoir se rendre sur un site tel que www.truc.com, l'ordinateur ne sait pas quel est ce serveur Web, donc il y a le protocole DNS qui sert à faire une correspondance entre un nom de machine et son adresse IP. Ce protocole va interroger le serveur DNS du réseau qui, lui, interrogera son supérieur, comme l'illustre la Figure 33.

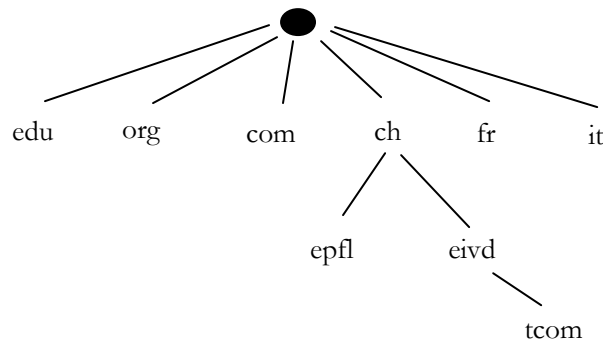


Figure 33 Organisation des domaines

Il y a différentes attaques sur les serveurs DNS (référence [RW 15]).

3.2.3.1 DNS ID spoofing

Lorsqu'une machine va faire une requête DNS au serveur, celle-ci va émettre cette requête en spécifiant l'URL du site et le champ ID de cette requête. Le hacker doit sniffer le réseau pour pouvoir connaître ce numéro ID pour ensuite communiquer à la victime l'adresse IP de son serveur Web avec le numéro ID de la requête.

3.2.3.2 DNS Cache Poisoning

Le hacker va faire une requête sur le serveur DNS de l'entreprise ayant comme URL le site à corrompre ; le serveur DNS va lui-même faire la requête à son supérieur. Puis, le hacker va répondre au serveur DNS comme si c'était son supérieur qui lui répondait, en spécifiant l'adresse IP d'un serveur du hacker. Cette information est stockée dans un cache par le serveur DNS pour éviter un surplus de trafic lorsque d'autres internautes aimeraient accéder à ce même site.

Partie pratique

Pour les essais pratiques, seule l'attaque DNS ID spoofing a été réalisée, étant donné que les essais doivent être réalisés sur un réseau indépendant de celui de l'école. Sur la Figure 34, on peut voir la topologie du banc de test. Il y a une victime et un hacker. La victime va émettre des requêtes DNS et le hacker va diriger ces requêtes sur sa machine et renvoyer son adresse IP. Pour que le hacker reçoive les requêtes DNS il faut lui empoisonner son cache ARP en lui spécifiant que l'adresse IP du serveur DNS est celle du hacker : pour cela il suffit de taper la commande :

```
arp spoof -t 10.192.72.235 10.192.72.46
```

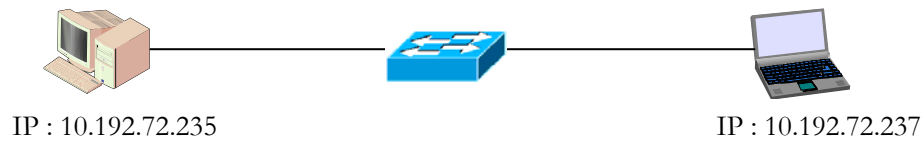


Figure 34 Topologie du banc de test du DNSspoofing

Toutes les requêtes en direction du serveur DNS seront envoyées sur la machine du hacker, donc il suffit de lancer le programme *dnsspoof* par la commande :

```
dnsspoof -f /root/dnsspoof/dns
```

Le fichier dns contient le nom des sites à spoofer ; voici le contenu :

```
10.192.72.237      *.toto.ch
10.192.72.237      *.tutu.ch
```

Donc lorsque la victime va faire des requêtes DNS sur les sites www.toto.ch et www.tutu.ch elles seront dirigées sur le serveur du hacker. Et si la victime fait des requêtes pour d'autres sites, elle n'aura aucune réponse étant donné que le réseau est isolé. Si cet essai est réalisé sur le réseau de l'entreprise, il faut activer le fichier *ip_forward* pour que les requêtes des autres sites aient des réponses par le bon serveur DNS. Au chapitre 3.5.3 de la page 48 sera spécifié comment activer le fichier *ip_forward*.

Comment se protéger

Pour pouvoir assurer que les réponses des serveurs DNS proviennent bien de celui-ci, il faudrait mettre en place un IDS qui contrôle que toutes les réponses DNS proviennent bien des serveurs de l'entreprise et non de la machine d'un hacker. Pour cela il faut que cette IDS puisse analyser tout le trafic transitant sur le réseau.

3.3 HIJACKING

Cette attaque consiste à détourner une connection TCP qui était établie entre deux machines pour la détourner sur soit-même. Cette attaque utilise une faiblesse du protocole TCP, en désynchronisant une connection TCP pour la reprendre à son profit. La contrainte est que l'une des machines doit se trouver sur le même réseau de telle manière à pouvoir sniffer le trafic. Pour pouvoir faire une désynchronisation il faut envoyer un paquet ayant un numéro de séquence ne correspondant pas à celui attendu par le système receveur et ceci étant valable pour une transmission client-serveur ou serveur-client.

Lors de l'envoi de paquets TCP, les numéros de séquence évoluent selon la taille de la cargaison de la trame TCP, comme l'illustre la Figure 35.

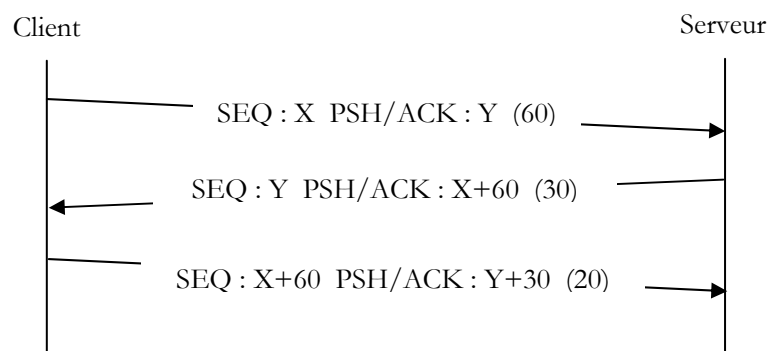


Figure 35 Transfert TCP

Pour pouvoir faire cette attaque, il faut attendre que la victime se soit connectée et après qu'elle s'est authentifiée par son mot de passe, il est possible de désynchroniser la connection TCP. Comme l'illustre la Figure 36, on voit que le hacker envoie une trame au serveur de telle manière à désynchroniser la connection entre A et B (il est clair que cette trame a l'adresse IP source de la victime) et lorsque le serveur recevra la trame de la victime (après celle du hacker) le serveur n'acceptera pas la trame du fait que le numéro de séquence n'est plus celui attendu, alors le serveur renvoie un ACK en spécifiant le numéro de séquence attendu et lorsque la victime reçoit ce ACK, elle générera aussi un ACK parce que le numéro de séquence n'est aussi pas celui attendu et alors on se trouve dans une situation qui s'appelle 'Ack Storm' (multitude de ACK générés). Pour pallier ce problème de 'Ack Storm', le hacker pourrait empoisonner le cache du serveur en spécifiant que l'adresse IP de la victime se trouve sur son adresse MAC, ainsi la victime ne recevra pas de ACK.

Cette technique est réalisable sur des protocoles comme Telnet ou FTP, mais si le hacker veut se connecter sur le serveur, il pourrait le faire d'une autre manière : le hacker se trouve sur le même réseau que la victime alors il peut lui casser sa connection TCP de telle manière à ce que la victime se reconnecte et le hacker puisse prendre le mot de passe.

Pour se prémunir de cette attaque il vaudrait mieux utiliser SSH et un serveur FTP sécurisé.

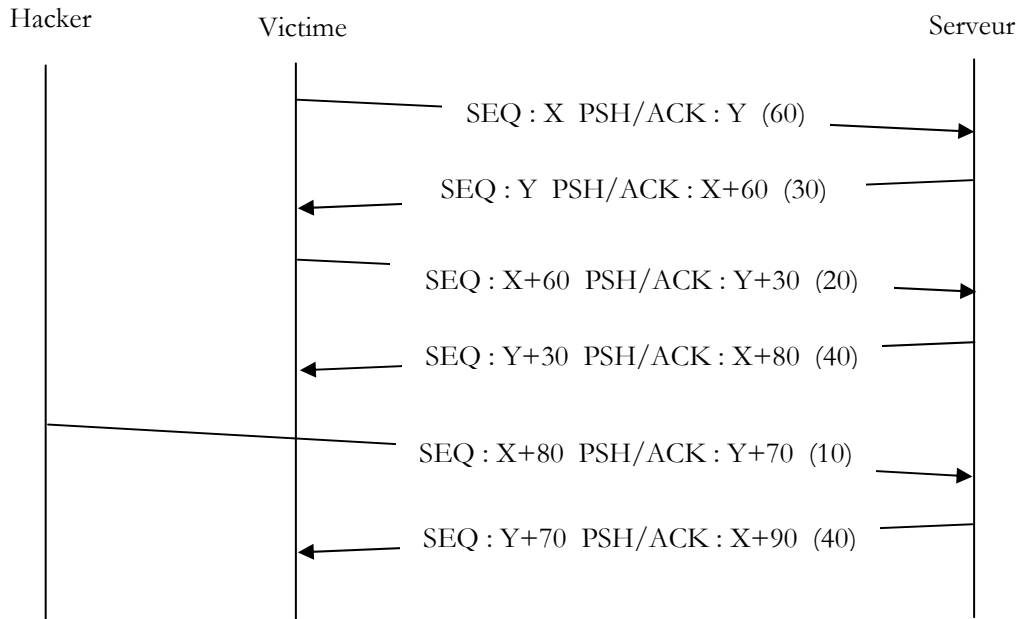


Figure 36 Désynchronisation de la connection TCP

3.4 BUFFER OVERFLOW

C'est une attaque qui vise à exploiter les failles des applications, ces failles existant pour différentes raisons. Lorsqu'un logiciel est programmé par une équipe, celle-ci est soumise à des contraintes dont généralement le temps. C'est à cause de cette contrainte que le logiciel est peu, voire pas, soumis à la phase de test. C'est à cause de cela qu'il y a des bugs dans les logiciels. Dans la majeure partie des programmes, il y a des sous-programmes ou fonctions. Lorsque le programme principal rencontre un appel à une procédure, il va falloir se souvenir (pour le programme principal) ou revenir dans le code, ceci est possible à l'aide d'un pointeur. Le pointeur et d'autres informations sont stockés dans la RAM qui fonctionne comme une pile de type LIFO. Admettons qu'une procédure est appelée dans un programme et que cette procédure a deux variables de retour et un paramètre en appel. Le programme principal va mettre dans la pile l'argument d'appel, le pointeur de retour et les deux paramètres de retour, comme l'illustre la Figure 37.

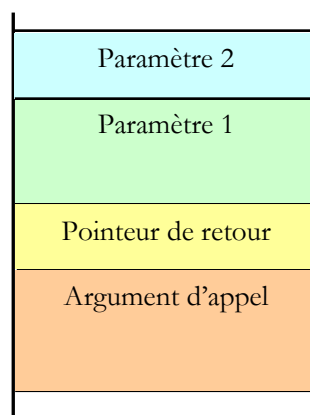


Figure 37 Opérations normales sur la pile

Les deux paramètres de retour ont une taille définie par le programmeur, par exemple : comme un 'string', on spécifie le nombre maximum. Donc si le programme n'a pas de test sur ces paramètres, lorsqu'ils vont être insérés, ils vont écraser ce qu'il y avait en dessous de la pile, comme l'illustre la Figure 38. Par conséquent, il est possible de spécifier un nouveau pointeur de retour de telle manière à exécuter un code voulu par le hacker qui pourrait faire ce que bon lui semble. Le code qui sera exécuté par cette attaque s'exécutera avec les mêmes privilèges que le programme qui subit cette attaque (référence [RW 13]).

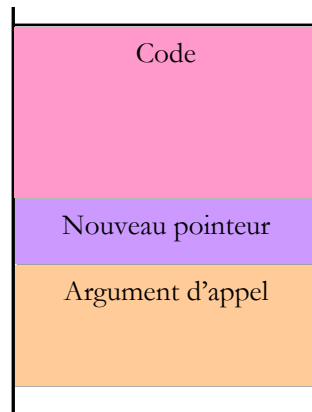


Figure 38 Ecrasement de la pile

Le suivant programme *vulnstrcpy* est un cas typique où il n'y a pas de contrôle de la taille de l'élément qui sera copié dans le buffer, car la fonction *strcpy* ne contrôle pas la taille des éléments copiés.

```
main (int argc, char *argv[])
{
    char buffer [256];

    if (argc > 1) strcpy (buffer, argv[1]);
}
```

Code tiré de la référence [RL 9]

Lorsqu'on exécute le programme et que le paramètre est supérieur à la taille du buffer, on obtient une 'Segmentation fault' comme l'illustre la Figure 39.

```
EDU-PC-LINUX1:~/hack# ./vulnstrcpy `perl -e "{print 'A'x260}"`
Segmentation fault
```

Figure 39 Cas de buffer overflow

3.5 ATTAQUE NIVEAU 2

3.5.1 ATTAQUE SUR UN SWITCH

Cette attaque a pour objectif de remplir la table CAM du switch pour qu'il devienne un hub. Pour cela il faut générer des paquets avec des adresses MAC sources différentes pour que le switch associe ces différentes adresses MAC à un port et lorsque la table est pleine et que le switch reçoit des messages destinés à une adresse MAC inconnue (de la table CAM), il ne saura où les envoyer et les enverra sur tous les ports, ce qui a comme effet un hub. La Figure 40 illustre cette attaque (référence [RL 7]).

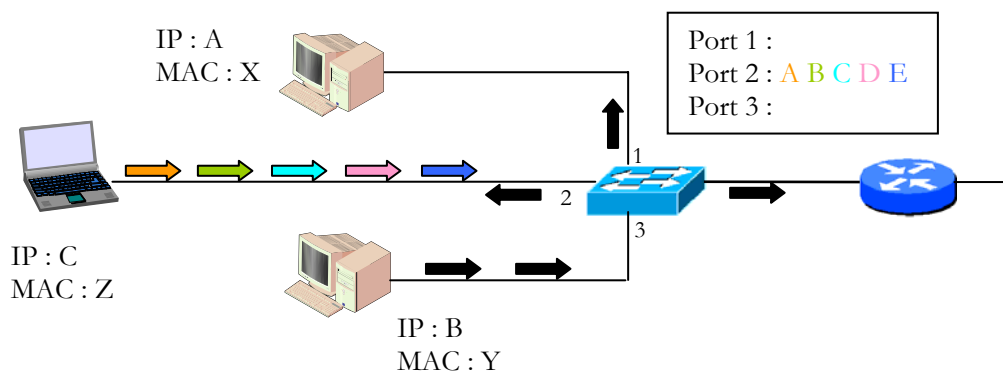


Figure 40 Attaque sur un switch

Partie pratique

Catalyst 1900

Les essais vont être réalisés sur un switch Cisco Catalyst 1900. Le premier essai est réalisé avec la configuration initiale fournie par le constructeur du matériel, ce qui implique :

- Pas de port de Uplink lorsqu'une trame avec une adresse de destination est inconnue de la table CAM, cette trame sera transmise en 'flooding' sur tous les ports.
- Timeout de la CAM : 300 sec lorsqu'une adresse MAC est inactive durant cette période, alors elle est enlevée de la table.
- FragmentFree lorsque le switch reçoit les premiers 64 bytes de la trame, il commence à la transmettre. Autre possibilité : Store-and-Forward, stocke tout puis transmet.
- Pas de collision cette option permet de créer des collisions sur la ligne lorsque le buffer du switch est plein pour que la machine répète la trame, seulement si activé.

Voici les seules configurations qui sont faites lors de la livraison (ou d'un reset) qui seront importantes pour les essais. La topologie du banc de test est sur la Figure 41 ; il y a deux analyseurs, de cette manière il sera possible de voir si cet effet de hub se voit sur tous les ports

ou uniquement sur le port où la saturation d'adresse est effectuée. Le programme qui sera employé est *macof* (chapitre 5.3.2 page 72), et la commande sera :

```
macof -i eth0 -s 10.192.72.33 -d 10.192.72.44 -e aa:aa:aa:aa:aa:aa
```

L'adresse MAC de destination est inconnue et on devrait avoir ces trames qui partent sur tous les ports.

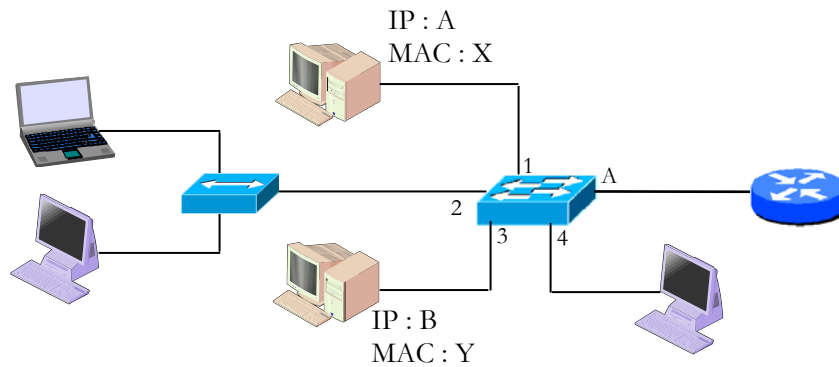


Figure 41 Topologie du banc de test pour attaquer le switch

Lorsque le programme est lancé, celui-ci n'arrête pas d'envoyer des messages sur tous les ports ; les deux analyseurs enregistrent toutes les trames émises. En visualisant la table CAM sur la Figure 42, on peut constater que seule l'adresse MAC du routeur et les adresses MAC générées par *macof* figurent sur cette table. Les deux utilisateurs ne figurent pas dans la CAM car ils sont débranchés du switch, comme si les machines étaient éteintes.

Port	Addresses
1 :	Unaddressed
2 :Dynamic 967	Static 0
3 :	Unaddressed
4 :	Unaddressed
5 :	Unaddressed
6 :	Unaddressed
7 :	Unaddressed
8 :	Unaddressed
9 :	Unaddressed
10 :	Unaddressed
11 :	Unaddressed
12 :	Unaddressed
AUI:	Unaddressed
A :Dynamic	00-00-0C-09-86-2D
B :	Unaddressed

Figure 42 Table CAM pleine

Puis, les deux machines sont branchées sur le switch (simulant l'allumage des celles-ci) et des pings sont envoyés depuis l'adresse IP B vers l'adresse IP A. Ces messages ICMP sont vus sur les deux analyseurs durant une période d'environ 288 sec ce qui correspond à 576 messages ICMP.

Cette expérience a été réalisée avec des temps de timeout de la CAM différents et cela plusieurs fois et, avec toutes ces mesures, différents résultats sont obtenus.

Lorsque les expériences sont faites sans perdre de temps entre les différentes étapes, alors le temps pendant lequel le switch est un hub est très proche de la durée du timeout de la table CAM qui est programmée. C'est à dire, lorsque le programme *macof* est lancé et tout de suite après que la table est pleine, les machines sont branchées et les pings lancés, alors dans ce cas la durée est proche de la durée de timeout de la table.

La conclusion serait que le temps maximum que le switch est un hub correspond à la valeur de timeout de la CAM. Sur la Figure 43 on peut voir une représentation du moment auquel le programme *macof* est lancé et plus tard le timeout de la CAM qui enlève les adresses MAC inactives et qui les remplace par des nouvelles. Les différentes valeurs obtenues dépendaient du moment auquel les pings étaient lancés. Puis, au moment du timeout de la table, les adresses MAC X et Y des postes venaient dans la table. Car, rappelons que *macof* génère des adresses MAC source aléatoires, par conséquent ce ne sont des adresses actives qu'une seule fois.



Figure 43 Timeout de la CAM

Les deux analyseurs ont affiché les mêmes résultats à chaque fois, donc le switch est un hub sur tous les ports et non pas uniquement sur le port qui est saturé.

Pour vérifier la conclusion précédente, les essais se feront avec la même topologie du banc de test mais en n'employant qu'un seul analyseur, mais surtout en ne faisant pas constamment fonctionner le programme *macof*: cela a la conséquence de surcharger tous les ports. Donc *macof* sera lancé puis arrêté après quelques secondes, le temps nécessaire de saturer la CAM. Puis les machines sont branchées et des pings envoyés. Les résultats sont identiques.

Donc, le switch efface les adresses MAC inactives après le temps de timeout. C'est pour cela que les adresses MAC X et Y sont prises dans la table à ce moment-là. Il est arrivé deux fois que, lorsque *macof* fonctionnait constamment, au moment du timeout, les adresses MAC X et Y n'étaient pas prises en compte ce qui a eu la conséquence de doubler le temps. Même conséquence avec X ou Y prise en compte.

L'idée serait donc de générer des paquets avec des adresses MAC différentes (comme *macof*) mais d'un nombre fini qui correspondrait au maximum possible dans la table CAM. Pour ce switch la valeur max est de 968 adresses MAC (967 + 1 de la Figure 42).

Le programme *winarp_sk* (chapitre 5.2.1 page 69) permet de forger des messages ARP avec des adresses MAC source et destination modifiables. Donc, pour tester, un script *arp.vbs* (annexe 1) a été créé. Celui-ci appelle 250 fenêtres DOS dont chacune lance le programme *winarp_sk* avec une adresse MAC source différente. Comme un seul ordinateur ne pouvait faire fonctionner 1000 fenêtres DOS en concurrence, il a fallu l'aide de 4 ordinateurs avec chacun 250 fenêtres DOS. Dans ce cas, des trames ARP (réponses) ont été envoyées depuis

ces quatre ordinateurs avec une adresse destination inconnue et une adresse source différente pour tous les 1000 programmes. Dans ce cas le switch reste toujours un hub.

A partir de ce résultat concluant est né le programme *winarp_tcom* (chapitre 5.2.2 page 71). Lorsque le paramètre N vaut 1000, le programme met 333 millièmes de seconde pour répéter la même adresse MAC. Le test a été effectué toujours avec la même topologie et avec le programme *winarp_tcom*, par la commande :

```
winarp_tcom.exe -m 2 -D bb-bb-bb-bb-bb-bb -S aa-aa-aa-aa-aa-aa -N 1000
                -F ca-ca-ca-ca-ca-ca -T cc-cc-cc-cc-cc-cc -d 10.10.10.10
                -s 20.20.20.20
```

Le timeout du switch a été paramétré au minimum ce qui correspond à 10 sec ; les étapes étaient toujours les mêmes et, dans ce cas, l'effet du hub sur le switch est permanent, du fait que ces milles adresses MAC sont toujours actives. Pendant 10 sec, la même adresse MAC est répétée environ 30 fois.

Pour éviter que toutes ces trames soient envoyées en 'flooding' sur tous les ports, il vaut mieux spécifier une adresse MAC destination valide (et que cette adresse soit active pour qu'elle figure dans la CAM), comme ça les trames ne seront pas envoyées en 'flooding' et seuls les messages des autres machines seront envoyés en 'flooding'.

Tous les tests qui ont été effectués jusqu'à maintenant simulaient l'allumage des machines après que l'attaque a été lancée. Maintenant ça sera le contraire, l'attaque sera lancée après que les utilisateurs se sont connectés. Sur la Figure 44, on peut voir la nouvelle topologie. Cette fois un seul analyseur sera employé car, comme dit précédemment, l'effet hub était visible sur tous les ports.

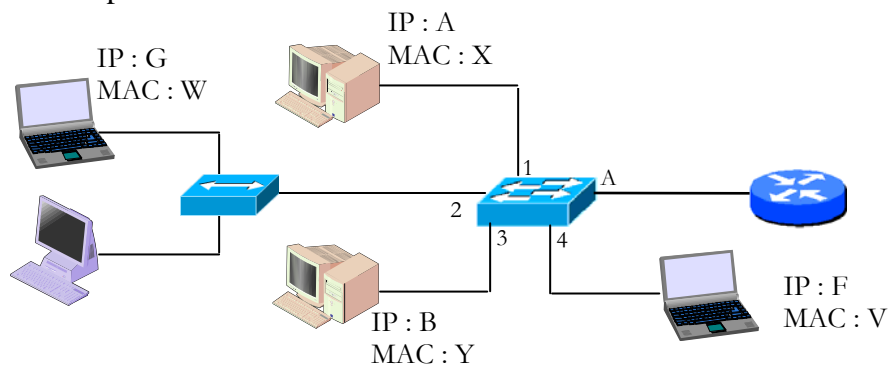


Figure 44 Nouvelle topologie du banc de test pour attaquer le switch

Pour commencer, des pings sont envoyés entre les adresses IP A et B pour que leur adresse MAC soit inscrite dans la CAM, puis des pings sont aussi envoyés entre l'IP G et F pour que l'adresse MAC V soit aussi inscrite dans la CAM. Ensuite, le programme *winarp_tcom* est lancé depuis la machine ayant l'adresse IP G : les paquets ont l'adresse MAC de destination 'V' pour que les messages ne soient pas diffusés sur tous les ports. Cet essai a été réalisé avec des temps de timeout différents et le résultat reste identique, c'est à dire : l'analyseur ne peut pas percevoir les messages ICMP transmis entre les adresses IP A et B, car leur adresse MAC figure toujours dans la CAM. Pour pouvoir percevoir ces messages il faut que les adresses MAC X et Y fassent un temps d'arrêt de transmission supérieur à la valeur de timeout du switch et comme cela on se retrouverait dans la situation des essais précédents.

Tous ces essais ont été faits selon la configuration initiale (au moment du déballage de l'appareil). Il serait possible, selon certaines configurations, de freiner la machine émettant tous ces messages : en activant 'Store-and-Forward' et en activant le système de collision.

En faisant des mesures, les résultats sont identiques aux précédents et le temps de transmission entre la même adresse MAC qui était de 333 millièmes de seconde a très légèrement augmenté, atteignant environ 335 millièmes de seconde.

Catalyst 1900 avec port de Uplink

Les essais qui vont suivre seront faits avec la même configuration sauf qu'un port de 'uplink' sera défini ce qui implique que tout paquet ayant une adresse de destination inconnue de la CAM sera envoyé sur le port de 'uplink'.

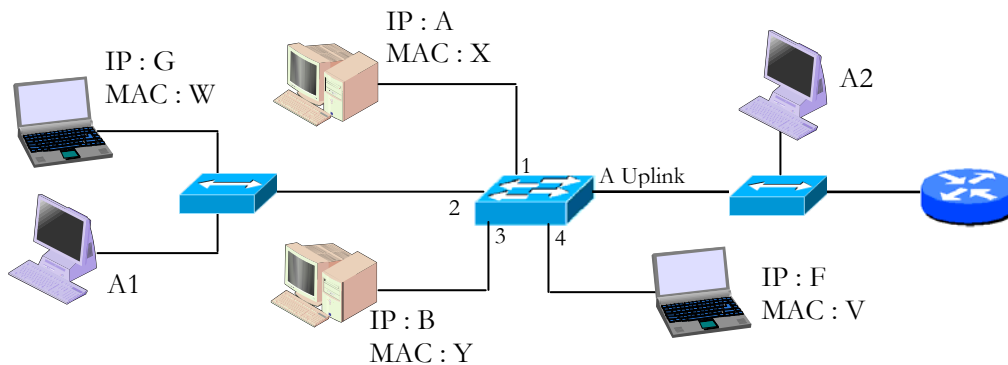


Figure 45 Topologie du banc de test pour attaquer le switch avec port de uplink

Pour commencer, des pings sont envoyés depuis l'adresse IP F vers l'IP G, puis la table CAM est remplie par le programme *winarp_tcom* depuis l'adresse IP G et les trames sont dirigées vers l'adresse IP F. Après, les deux utilisateurs sont connectés sur le switch et des pings sont envoyés depuis l'adresse IP B en destination de l'IP A. Aucun des deux analyseurs ne perçoit des messages ICMP, car la machine ayant l'IP B va faire une requête ARP en broadcast pour connaître l'adresse MAC de l'IP A, puis la machine A va répondre à B par une réponse ARP en unicast, mais comme l'adresse MAC Y ne figure pas dans la CAM alors ce message part en direction du routeur, comme l'illustre la Figure 46.

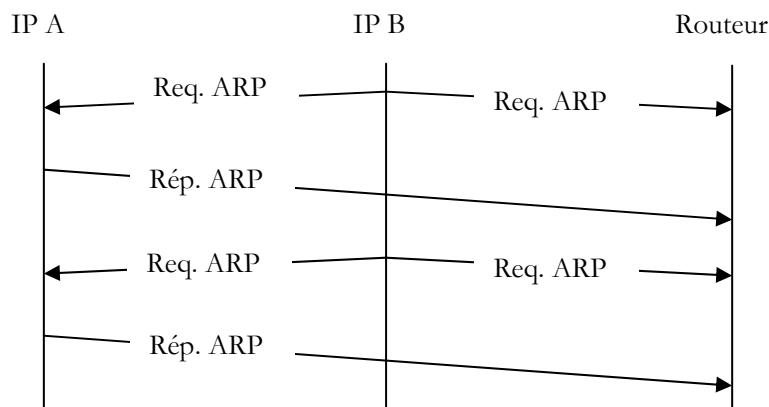


Figure 46 Diagramme fléché

Si les adresses MAC sont mises en statique juste pour les essais, les pings sont perçus uniquement par l'analyseur A2 qui se trouve sur le chemin du routeur. Dans ce cas de figure, nous n'avons pas l'effet d'un hub mais d'un déni de service sur les utilisateurs branchés sur le switch.

Si *macof* est employé au lieu de *winarp_tcom*, le déni de service sera observé durant la période qui est programmée dans le switch pour le timeout de la table CAM.

Catalyst 1900 avec VLANs

Ce switch Cisco permet de configurer des VLANs, alors ce test permettra de savoir si ce qui a été testé précédemment peut être appliqué lorsque des VLANs sont configurés sur ce même switch. Sur la Figure 47 on peut voir le banc de test. Sur les deux switches, ont été configurés deux VLANs (VLAN1 et VLAN2). Les ports 1 à 8 appartiennent au VLAN1 et les ports 9 à 12 appartiennent au VLAN2. Comme on peut le voir, les analyseurs 1 et 2 sont sur le VLAN1 et l'analyseur 3 est sur le VLAN2.

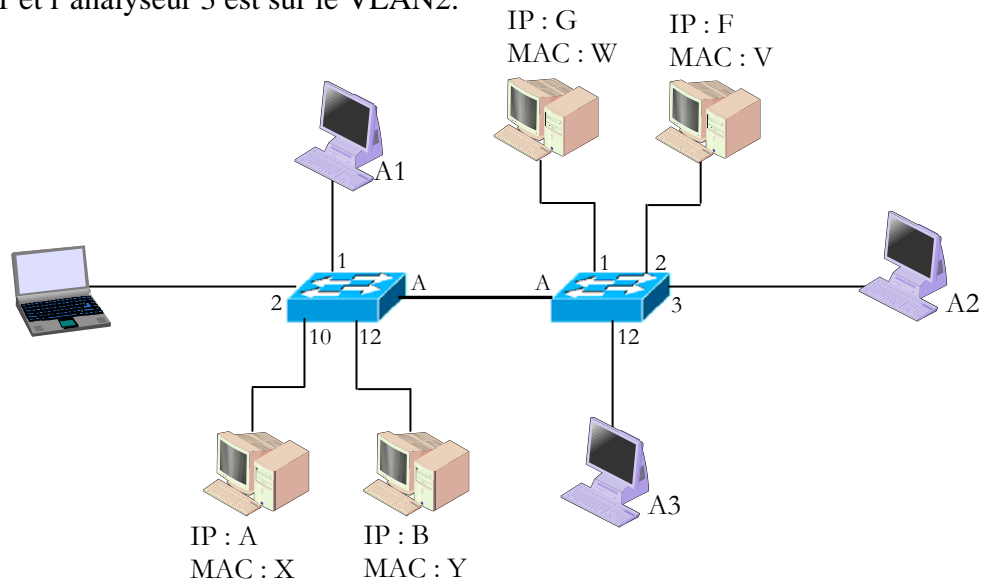


Figure 47 Topologie du banc de test des switches avec VLANs

Les étapes sont identiques à celles faites précédemment. Le hacker lance le programme *winarp_tcom* avec l'adresse MAC destination inconnue, les messages sont envoyés sur tous les ports appartenant au VLAN1 (perçus par les analyseurs A1 et A2), donc sur les deux switches. Les deux tables CAM sont saturées par les messages. Puis les quatre utilisateurs sont connectés et des pings sont envoyés entre les utilisateurs A et B et entre les utilisateurs G et F. Les pings transitant sur le VLAN1 (pings entre G et F) sont perçus sur les analyseurs A1 et A2 et les pings transitant sur le VLAN2 (pings entre A et B) sont perçus sur l'analyseur A3. La confidentialité entre VLANs est garantie mais les switches sont devenus des hubs. La Figure 48 illustre le schéma équivalent mais avec le VLAN1 saturé de messages.



Figure 48 Schéma équivalent à l'attaque sur les VLANs

L'attaque aurait pu se faire avec une adresse MAC destination connue (avec un complice) sur le même VLAN, mais sur l'autre switch. Cette attaque est équivalente à deux switches (sans VLAN) mis ensemble (comme la Figure 47) avec le port de uplink non configuré et on aurait obtenu deux hubs mis ensemble.

Lorsque des VLANs sont configurés et un port est configuré en tant que trunk et que des messages sont envoyés avec une adresse MAC inconnue, le switch envoie ce message sur tous les ports du même VLAN. Sur ce modèle de switch Cisco, il n'est pas possible de configurer les messages en flooding comme l'illustre la Figure 49. Le label N/A signifie non applicable.

```

Trunking status: On      Encapsulation type: ISL
----- Information -----
Transmit Flood traffic to VLANs      N/A
Receive Flood traffic from VLANs     N/A
    
```

Figure 49 Information sur le Trunk

Comment se protéger

On a pu voir plusieurs topologies et plusieurs configurations du (des) switch(s). Lorsque des VLANs n'étaient pas configurés (et uniquement dans ce cas-là) il était possible de configurer un port en tant que port de uplink, ce qui a pour effet d'envoyer les messages avec une adresse de destination inconnue uniquement sur ce port mais il faut que l'élément qui se trouve de l'autre côté soit configuré correctement. Si c'est un switch et que le flooding est possible, alors c'est l'autre switch qui sera un hub.

Il est possible de configurer chaque port de manière indépendante, et dans les options de configuration (Figure 50), il est possible d'interdire le flooding sur le port donc, dans le cas des VLANs, il est possible de configurer tous les ports sauf celui du trunk en interdisant le flooding. De la sorte ce sera comme un port de uplink.

```

----- Settings -----
[T] Address table size      Unrestricted
[S] Addressing security     Disabled
[K] Clear addresses on link down Disabled
[U] Flood unknown unicasts  Enabled
[M] Flood unregistered multicasts Enabled
    
```

Figure 50 Configuration d'un port

Mais, pour contrer ce type d'attaque, existe une configuration le permettant. Il est possible de configurer chaque port en tant que port de sécurité ('Addressing security' sur la Figure 50), ce qui permet d'accueillir un maximum de 132 adresses statiques par port. 132 est un maximum, ce paramètre est modifiable sur le menu 'Address table size'. Pour cela, il faut configurer chaque port du switch. Il est concevable d'entrer toutes les adresses statiques manuellement, alors le switch enregistre les 132 premières adresses en mode statique pour chaque port.

Puis, dans le menu principal, il faut configurer l'action lorsque la saturation est atteinte sur un port : pour cela il y a trois actions possibles :

- Suspendre (par défaut) le port peut encore envoyer des messages mais uniquement avec les adresses enregistrées ; les autres messages sont ignorés.
- Désactive le port est désactivé et plus aucun message n'entre ni ne sorte.
- Ignore comme son nom l'indique, rien est fait.

Lorsqu'un port est configuré comme sûr, l'une des trois actions ci-dessus est applicable mais selon n'importe laquelle des actions il y a un message SNMP qui est envoyé à l'administrateur.

Si un port est saturé, la led au-dessus du port est de couleur orange et même une coupure de courant ne réinitialise pas le switch. Il faut que l'administrateur réactive le port et dans le statut de celui-ci le statut indique 'Suspend-violation' qui informe pourquoi le port est bloqué.

Test des ports sécurisés

Tous les ports du switch ont été programmés comme sûrs, avec le maximum de 132 adresses statiques par port ; si on multiplie 132 fois 12 ports (10 Mbit/s), on obtient 1584 adresses statiques, ce qui est nettement supérieur à 968 qui est le maximum de la CAM. Alors, pour tester, chaque port sera saturé avec le programme *macof*, et on peut voir sur la Figure 51 la table CAM.

Port	Addresses
1 :Secured	Static 132 Max 132
2 :Secured	Static 132 Max 132
3 :Secured	Static 132 Max 132
4 :Secured	Static 132 Max 132
5 :Secured	Static 132 Max 132
6 :Secured	Static 132 Max 132
7 :Secured	Static 49 Max 132
8 :Secured	Unaddressed
9 :Secured	Unaddressed
10 :Secured	Unaddressed
11 :Secured	Unaddressed
12 :Secured	Unaddressed
AUI:	Unaddressed
A :Secured	Unaddressed

Figure 51 Table CAM saturée avec les ports sécurisés

Le total des adresses est de 841 ($6 * 132 + 49$) donc il reste encore 127 adresses possibles à mettre, mais en mode statique ce n'est plus possible, les adresses restantes sont donc mises en mode dynamique, par conséquent le programme *winarp_tcom* est branché sur le port 12 et sont aussi branchés un analyseur sur le port 11 et deux utilisateurs sur les ports 8 et 9. A nouveau, on obtient l'effet hub sur le switch s'il n'y a pas de port de uplink configuré, sinon c'est un déni de service.

3.5.2 VLAN HOPPING

Il s'agit d'une attaque orchestrée sur des utilisateurs se trouvant sur un VLAN différent de celui du hacker. Lorsque des VLANs sont créés (Figure 52), les utilisateurs du même VLAN peuvent échanger du trafic mais le trafic entre VLANs n'est (théoriquement) pas possible. Pour que les VLANs puissent être garantis entre des switches, les paquets transitent sur le trunk avec un protocole tel que : 802.1Q ou ISL. De cette manière, le switch recevant un paquet saura sur quel VLAN le router.

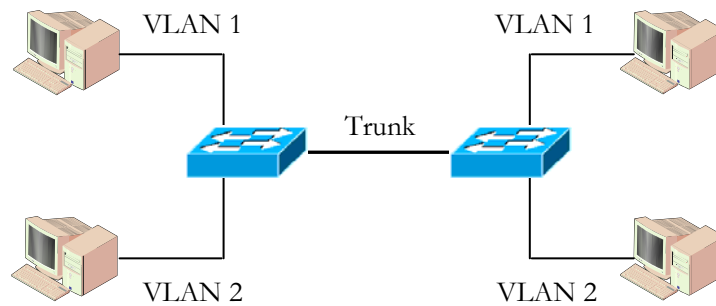


Figure 52 Topologie avec VLANs

L'attaque consiste à envoyer un paquet ayant une double encapsulation du protocole 802.1Q de telle manière à ce que la première encapsulation soit détectée par le premier switch et le second switch va percevoir la seconde encapsulation et diriger les paquets sur le VLAN de la cible. Dans le protocole 802.1Q il y a un champ qui identifie le VLAN ; sur la première encapsulation ce champ correspond au VLAN du hacker et dans la seconde le champ correspondra au VLAN de la victime.

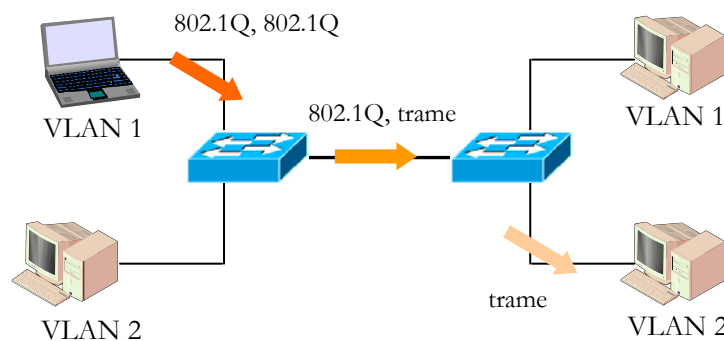


Figure 53 Attaque VLAN Hopping

Cette attaque est tirée de la référence [RL 7].

3.5.3 DÉTOURNEMENT DE SESSION

Nous avons vu au chapitre 3.2.1 page 26 l'attaque de l'ARPspoofing qui consistait à empoisonner le cache ARP de la victime pour qu'elle envoie ses paquets vers le hacker, mais il faut que celui-ci renvoie ces paquets vers la bonne adresse MAC sinon la victime subira l'effet d'un déni de service qui éveillerait des soupçons en elle. Pour cela, il faut que la machine du hacker devienne une sorte de proxy. Il va falloir activer le routage : par défaut, sous Linux, cette possibilité n'est pas activée, alors on peut l'activer en mettant à 1 la valeur du fichier virtuel `ip_forward`, par la commande :

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

Pour vérifier la valeur du fichier, on utilise la commande `cat ip_forward`. Pour désactiver ce routage, il suffit de réécrire l'instruction précédente en y mettant la valeur 0, mais de toute façon lorsque le système est éteint la valeur revient à 0 (référence [RW 14]).

Une fois que le trafic est dirigé sur le hacker et que celui-ci reroute le trafic permettant à la victime d'atteindre le service désiré, le hacker peut analyser le trafic qui transite par lui comme par exemple prendre des mots de passe ou d'autres informations.

Partie pratique

Sur la Figure 54, on peut voir la topologie du banc de test qui permettra de voir comment capturer un mot de passe lors d'une connexion FTP. Pour commencer, il ne faut pas oublier d'activer le fichier virtuel (sous cause de créer un déni de service) : ceci se fait par la commande vue précédemment. Ensuite, il va falloir empoisonner le cache ARP de la victime avec le programme *arpspoof* (chapitre 5.3.1 page 72), par la commande :

```
arpspoof -i eth0 -t 10.192.72.235 10.192.72.236
```

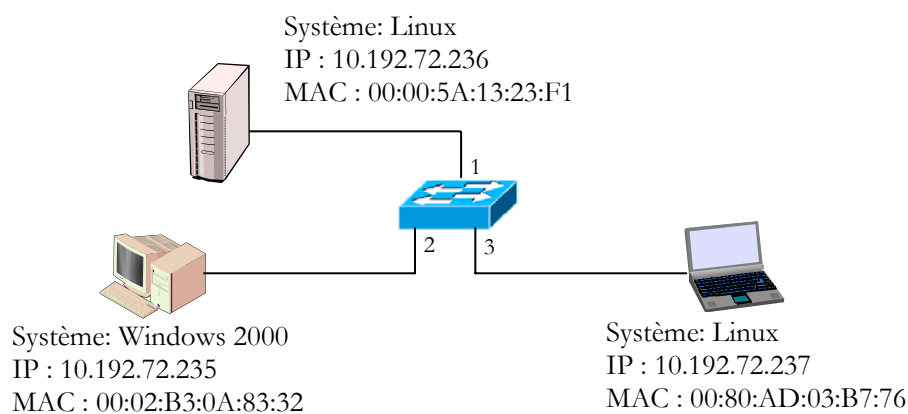


Figure 54 Topologie du banc de test pour capturer un mot de passe FTP

Ensuite il faut lancer le programme *dsniff* (chapitre 5.3.5 page 74) pour capturer les mots de passe : ceci se fait par la commande :

```
Dsniff -c dst port 21
```

Le paramètre 'c' indique qu'il faut réassembler en half-duplex le flux TCP car l'empoisonnement du cache ARP ne se fait que sur la victime, par conséquent le flux TCP sera asymétrique (du client au serveur le flux passera par le hacker, mais du serveur au client le hacker ne voit pas le flux). Puis 'dst port 21' permet de filtrer uniquement le flux TCP dirigé sur le port 21 qui est du FTP. Enfin, il suffit d'attendre patiemment que la victime fasse une connection FTP sur le serveur. Sur la Figure 55, on peut voir que le user et le mot de passe s'affichent à l'écran, ainsi que les adresses IP source et destination et aussi le protocole.

```
EDU-PC-LINUX1:~# dsniff -c dst port 21
dsniff: listening on eth0 [dst port 21]
-----
12/13/02 04:14:45 tcp 10.192.72.235.1602 -> 10.192.72.236.21 (ftp)
USER toto
PASS tcom
```

Figure 55 Mot de passe FTP

Pour ce genre d'attaque de mot de passe, on peut aussi utiliser le programme *Ettercap* (chapitre 5.4 page 75), la topologie du réseau restant identique à celle de la Figure 54. On lance le programme, puis on sélectionne les adresses IP dont on veut prendre le mot de passe. Cette fois ce sera sur une connection Telnet entre la machine Windows et le switch qui simulerait un administrateur voulant configurer le switch. L'adresse IP source à sélectionner est 10.192.72.235 et l'adresse IP destination est 10.192.72.100. Une fois l'adresse sélectionnée, on presse la touche 'a' pour de l'ARPspoofing. A ce moment, le programme va empoisonner le cache ARP du poste Windows et du switch de telle manière à analyser le trafic dans les deux sens. Lors de l'emploi de ce logiciel, il n'y a pas besoin d'aller modifier la valeur du fichier ip_forward.

Lorsque l'administrateur ouvre la connection Telnet sur le switch, apparaît à l'écran du hacker cette connection, comme le montre la Figure 56.

```
----- ettercap 0.6.4 -----
SOURCE: 10.192.72.235 <- Filter: OFF
DEST : 10.192.72.100 <- doppleganger - illithid (ARP Based) - ettercap
Active Dissector: ON

5 hosts in this LAN (10.192.72.237 : 255.255.254.0)
1) 10.192.72.235:3772 <--> 10.192.72.100:23 silent telnet
```

Figure 56 Connection Telnet sur Ettercap

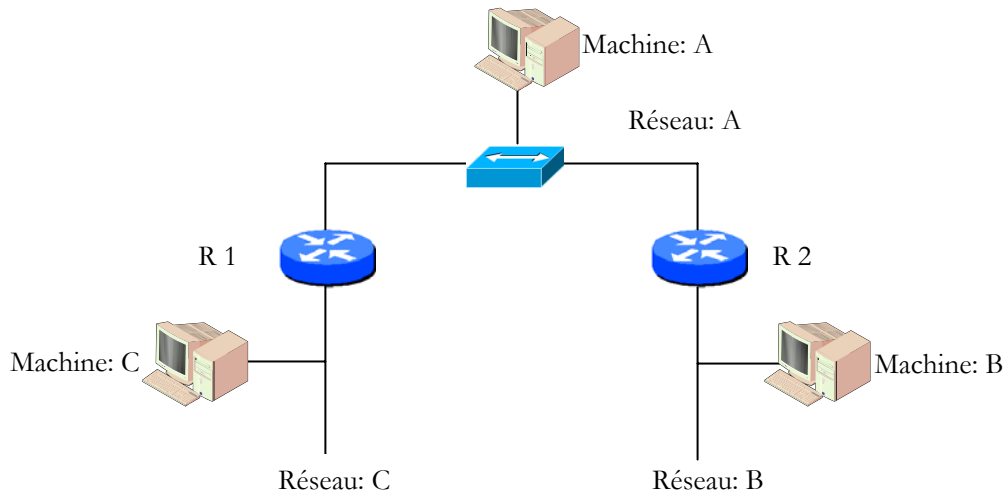


Figure 58 Réseau avec cas de ICMP redirect

Lorsque la machine A va envoyer un ping à la machine B, elle va l'envoyer à son 'default gateway' qui est R 1, mais comme la machine B ne se trouve pas sur le réseau C le routeur enverra un message ICMP redirect à la machine A en lui spécifiant que le bon routeur est R 2. Sur la Figure 59 on peut voir le diagramme en flèches.

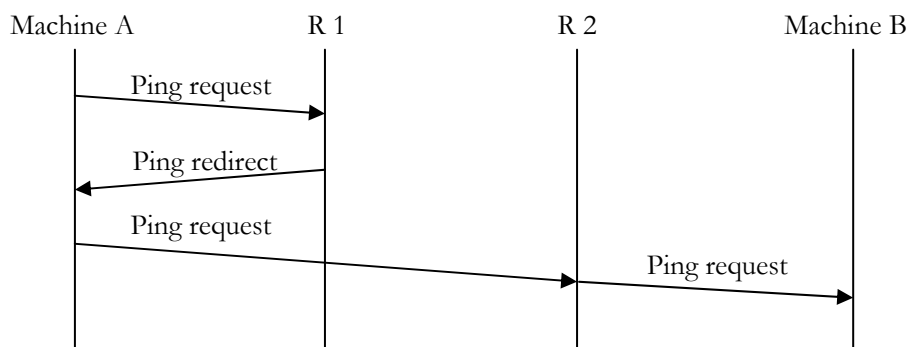


Figure 59 Diagramme fléché avec ICMP redirect

Etercap n'envoie pas ces messages ICMP redirect. Pour empêcher que ces messages ne soient envoyés lors de l'emploi du programme *arp spoof*, il faut modifier la valeur du fichier `send_redirects` qui est localisé à trois endroits ; ceci se fait par les commandes suivantes (référence [RW 16]) :

```
echo 0 > /proc/sys/net/ipv4/conf/all/send_redirects
echo 0 > /proc/sys/net/ipv4/conf/default/send_redirects
echo 0 > /proc/sys/net/ipv4/conf/eth0/send_redirects
```

Pour éviter d'écrire ces commandes plusieurs fois, deux scripts (chapitre 5.7.3 page 78) activation et désactivation ont été créés. Annexe 2.

Comment détecter

Pour détecter si nous sommes victimes d'ARPspoofing, il est possible de le savoir à l'aide du programme *tracert* (chapitre 5.5 page 76). De cette manière il serait possible de voir si un hacker se trouve sur le chemin.

Ce premier essai se fera avec l'utilisation du programme *arpspoof* en gardant toujours la topologie de la Figure 54 de la page 48. Sur la Figure 60 on peut voir le résultat et on remarque bien l'adresse IP du hacker qui se trouve entre la victime et le serveur FTP.

```
C:\>tracert 10.192.72.236

Tracing route to 10.192.72.236 over a maximum of 30 hops

  1  <10 ms  <10 ms  <10 ms  10.192.72.237
  2  <10 ms  <10 ms  <10 ms  10.192.72.236

Trace complete.
```

Figure 60 Résultat d'un tracert en étant spoofer par arpspoof

Ce second essai se fera avec l'utilisation du programme *ettercap*. Sur la Figure 61 on peut constater que la machine du hacker ne figure pas sur le chemin.

```
C:\>tracert 10.192.72.236

Tracing route to 10.192.72.236 over a maximum of 30 hops

  1    40 ms   40 ms   40 ms  10.192.72.236

Trace complete.
```

Figure 61 Résultat d'un tracert en étant spoofer par ettercap

Cette méthode n'est pas infaillible car *ettercap*, lorsqu'il reçoit un paquet et qu'il le retransmet, ne modifie pas la valeur du champ TTL, qui doit être décrétementée à chaque passage d'un élément réseau. C'est pour cela qu'*ettercap* n'est pas visible. Par contre, le défaut de *ettercap* est qu'il envoie les réponses ARP toutes les 30 secondes, contrairement à *arpspoof* qui les envoie toutes les 3 secondes.

Comment se protéger

Pour se prémunir de cette attaque, il est possible de mettre en statique les adresses IP dans le cache ARP mais, comme dit au chapitre 3.2.1 page 26, cette solution est pénible à mettre en place et a des inconvénients avec certaines versions de Windows. Or il y a un moyen de détecter cette attaque : c'est d'utiliser un IDS, qui sera présenté au chapitre 4 page 59.

Ces attaques sont exécutées sur des protocoles non sûrs, pour cela il ne faut plus les utiliser et privilégier les protocoles qui cryptent les données, comme le protocole SSL. Lorsqu'on subit une attaque d'ARPspoofing et que le hacker ne peut pas voir ce qui transite car les informations transitant à la couche application seront cryptées, cette attaque n'a plus vraiment de sens à moins de créer un DoS.

3.6 MAN IN THE MIDDLE

Cette attaque consiste à s'intercaler dans une connection Internet sécurisée d'un internaute lors de l'échange des certificats. Lorsqu'une connection sécurisée commence, le site envoie son certificat contenant différentes informations dont la clé publique. Puis, le navigateur de l'internaute génère une clé symétrique (nécessaire pour la suite du trafic des données) et qui est cryptée à l'aide de la clé publique du site de telle manière que seul le détenteur de la clé privée pourra décrypter la clé symétrique. Dans le cas du 'man in the middle', le hacker va devoir échanger les certificats à l'établissement de la connection, comme l'illustre la Figure 62.

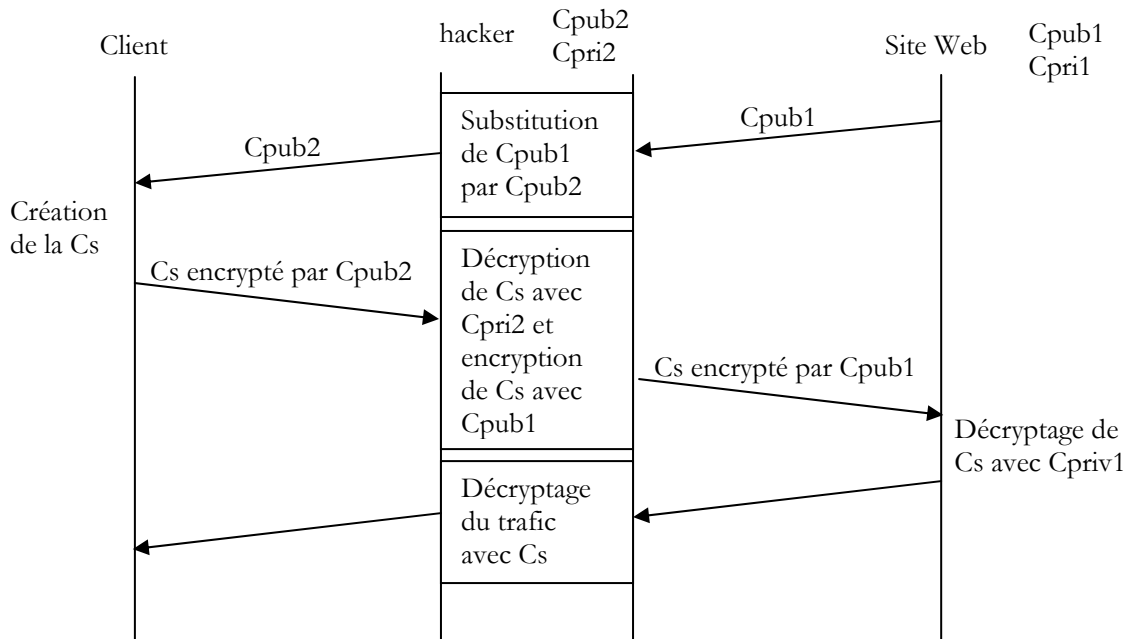


Figure 62 Transaction avec le man in the middle (source : cours de cryptographie de M. Jaton)

Partie pratique

Sur la Figure 63, on peut voir la topologie du banc de test. La machine ayant l'adresse IP 10.192.72.237 dispose d'un serveur 'apache-ssl' selon la configuration décrite au chapitre 5.7.1 à la page 77.

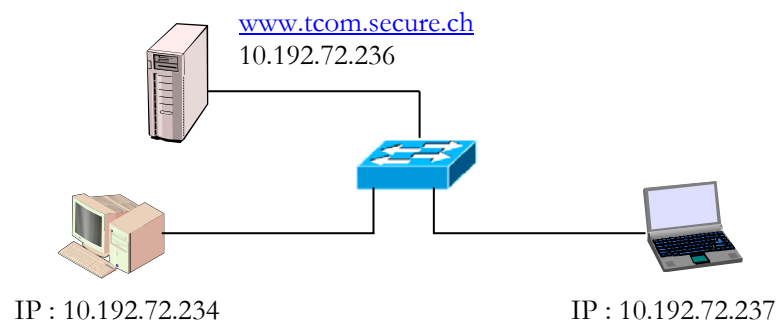


Figure 63 Topologie du banc de test du man in the middle

Avant de commencer, il faut établir une connection depuis le poste de la victime sur le serveur Web pour que la victime puisse installer le certificat du serveur comme étant un serveur de confiance.

Pour pouvoir faire cette attaque il faut avant tout activer le routage et empêcher que les messages ICMP redirect ne soient envoyés par la machine du hacker. Ceci se fait en lançant le script activation (annexe 2) par la commande : `./activation`.

Le hacker va devoir empoisonner le cache ARP de la victime de telle manière à prendre tout le trafic envoyé sur le serveur DNS, ceci se faisant par la commande :

```
arpspoof -t 10.192.72.234 10.192.72.46
```

Ensuite il faut lancer le programme *dnsspoof* (chapitre 5.3.3 page 73) pour pouvoir répondre à la place du serveur DNS, ceci se faisant par la commande :

```
dnsspoof -f /root/webmitm/mim
```

Le fichier mim dont le programme a besoin, contient l'adresse IP du hacker associée au nom du site à spoofer. Voilà ce qu'il contient :

```
10.192.72.237          *.tcom.secure.ch
```

Et enfin, il faut lancer le programme *webmitm* qui va se charger de faire l'échange des certificats. Lorsque le programme est lancé, étant donné que c'est la première fois, il va demander à remplir les champs du certificat. Les champs insérés à ce moment là sont les mêmes que ceux insérés au chapitre 5.7.1 de la page 77. En temps normal, pour que le hacker connaisse les champs à introduire, il se connectera sur le serveur et regardera la valeur de ces champs. Le programme se lance par la commande :

```
webmitm 10.192.72.236
```

Maintenant le client va se connecter sur le serveur Web en mettant dans l'url du navigateur ceci <https://www.tcom.secure.ch>. Normalement, lorsque le client se connecte, il ne devrait rien se passer à part que d'afficher la page HTML. Mais dans ce cas, le client obtient le message de la Figure 64 qui lui signale que le certificat n'est pas émis par une société de confiance (normal, c'est celui du hacker), que la date de validation est bonne et que le certificat correspond bien à la page demandée. Dans ce genre de situations, les utilisateurs cliquent sur 'oui' et dans ce cas ils accèdent à la page désirée mais avec un intrus au milieu du trafic. Ce qui peut être déroutant lorsqu'on veut se connecter de manière sécurisée.

Cette attaque est possible par négligence des utilisateurs, par conséquent il faut bien lire les messages affichés même s'ils ne sont pas toujours clairs et, dans ce genre de cas, faire confiance à son navigateur qui conseillait de cliquer sur 'non'.

La pire des choses qui puissent arriver, c'est qu'il y aie des faux certificats dans le navigateur : dans ce cas ni, l'utilisateur ni le navigateur ne se rendrait compte que le certificat présenté est non certifié par une autorité compétente. Et l'utilisateur se croirait en train de surfer en toute tranquillité et de payer avec sa carte de crédit.

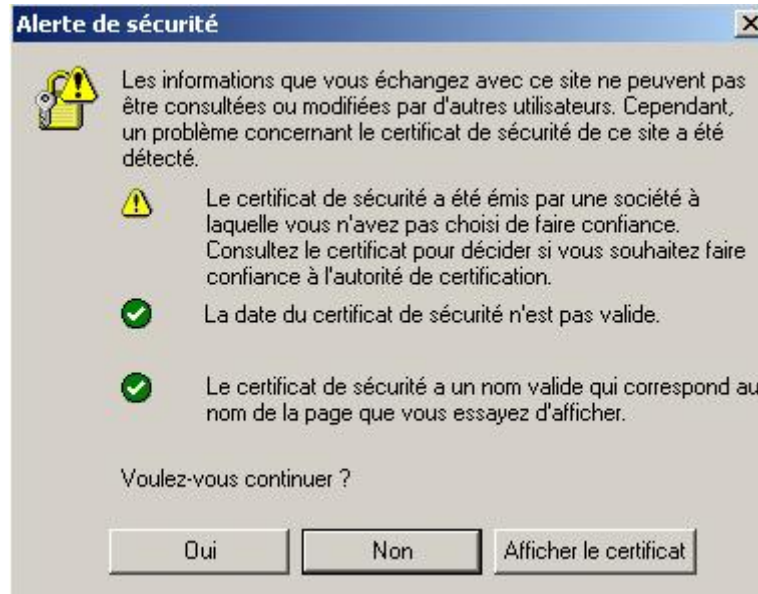


Figure 64 Problème de connexion sécurisée

3.7 TROYEN

Le nom troyen provient de la légende mythologique du 'cheval de Troie' qui fut offert à la ville de Troie. C'est de là qu'on appelle troyens les programmes qui ouvrent des 'backdoors'.

L'objectif de ces programmes est de laisser une porte ouverte (backdoor) après une intrusion sur le système, de telle manière le hacker pourra revenir sans problème. Ils peuvent aussi être installés en ayant exécuté une petite application reçue par un e-mail. Une fois installés, ils laissent un port ouvert et certains envoient des mails au hacker en lui signalant l'adresse IP de la victime.

Une fois sur le système, les troyens se lient à des applications, modifient la base de registre de manière à être exécutés dès le lancement du système. Ils sont actifs en permanence et sont difficilement ou pas détectable par le système. En allant regarder dans le gestionnaire de tâches, celui-ci n'affichera pas ou affichera un nom de programme banal, comme : note.exe, winamp34.exe.

Pour les détecter, les anti-virus parviennent à en détecter certains types mais si les codes sources de ces programmes sont disponibles, alors un hacker peut reprendre les codes sources, les recompiler et ainsi modifier l'empreinte du programme de manière à exécuter ce que voulait le programmeur et surtout à ne plus être détectable par les anti-virus. Il est possible de détecter ces programmes en scannant les ports de manière à trouver un port ouvert qui ne devrait pas l'être. Référence sur ce paragraphe, [RL 3] et [RW 18].

3.8 RÉSEAU SANS FIL

Maintenant, pour relier les ordinateurs au réseau de l'entreprise, il est possible d'utiliser la technologie sans fil de Wireless Lan (802.11) qui est bien pratique pour les postes qui sont en déplacement constant, mais l'inconvénient est que le réseau de l'entreprise ne se limite plus physiquement au mur extérieur de celle-ci mais est à la portée des ondes émises par cette technologie, qui sont à une fréquence de 2.4 GHz. Cette fréquence est proche de celle utilisée par la technologie GSM et on sait très bien qu'elle est utilisable à l'intérieur comme à l'extérieur des habitations. Donc, en ayant des émetteurs de réseau sans fil à l'intérieur, il est possible pour une personne possédant un ordinateur portable avec une carte Wireless de se connecter au réseau de l'entreprise. Cette personne se trouve physiquement à l'extérieur de l'entreprise mais malheureusement elle est dans le réseau de l'entreprise. Il est possible par des logiciels, de détecter les ondes Wireless, comme l'illustre la Figure 65. Cette capture a été effectuée à l'extérieur et à une distance d'environ 15 mètres des parois et l'émetteur se situait à environ 5 mètres de la paroi, le signal étant très faible mais suffisant pour se connecter. Par conséquent, il ne faut pas oublier d'activer l'encryption de données.

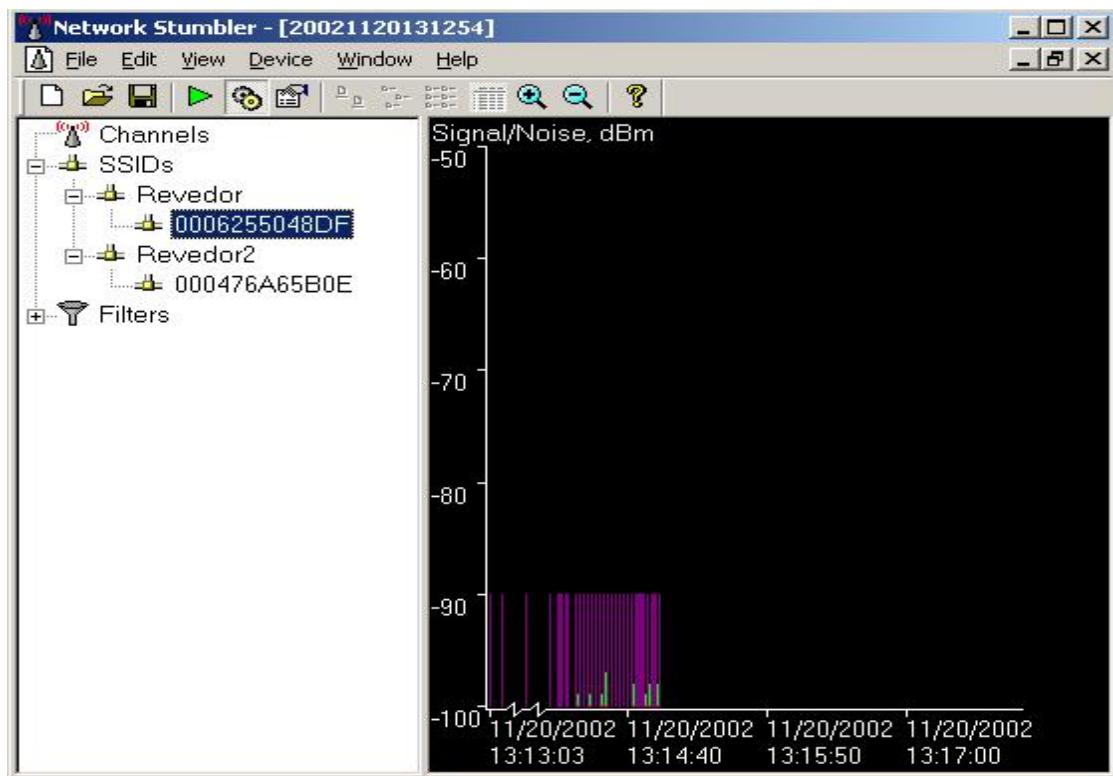


Figure 65 Capture Wireless

Malheureusement le protocole d'encryption de données WEP n'est pas fiable, celui-ci utilise la même clé secrète (connue au préalable par l'AP et les stations lors de la configuration) pour toutes les transmissions. Du fait que c'est une clé statique fixe, des programmes comme *airsnort* peuvent la découvrir en analysant le trafic. Référence sur le programme *network stumbler* [RW 19] et pour le programme *airsnort* [RW 20].

3.9 MOTS DE PASSE

Le mot de passe est la première et souvent la seule barrière d'un système, c'est aussi la façon la plus répandue d'identifier une personne (en associant le mot de passe à nom de 'login'). Sur la plupart des machines, pour pouvoir travailler, il faut entrer un nom de 'login' et un mot de passe mais il est encore fréquent de trouver ce mot de passe sous le clavier ou écrit sur un post-it collé sur l'écran ou encore écrit sur un papier qui finit à la corbeille. De plus, les éléments réseau tels que switch, routeur, AP, etc... sont fournis d'usine avec un mot de passe par défaut qui est généralement le nom du constructeur ou 12345. Sur le site de la référence [RW 17] on peut trouver les mots de passe par défaut des éléments réseau les plus utilisés. Donc si les administrateurs ne modifient pas ces mots de passe et que quiconque modifie la configuration des ces éléments, ça serait catastrophique pour une entreprise.

Mot de passe robuste

Il faut que tous les mots de passe soient du type 'robuste', mais qu'est-ce donc un mot de passe robuste ?

- Il doit être changé régulièrement, généralement entre 30 et 60 jours
- Il doit avoir une longueur minimale de 10 caractères
- Il contient au moins un caractère alphanumérique, un chiffre et un caractère spécial
- Il ne contient pas de mot du dictionnaire
- Il doit être différent des cinq derniers mots de passe utilisés

Voilà à quoi doit ressembler un mot de passe robuste, ce qui consiste à avoir un mot de passe comme : `bw%t6q»/7j`, mais ce mot de passe a un sérieux défaut, c'est que personne n'arrivera à se souvenir facilement d'un mot de passe aussi barbare. La plupart des mots de passe sont des mots du dictionnaire c'est pour cela que certains programmes cassent les mots de passe à l'aide d'un dictionnaire et d'autres utilisent la force brute, mais cette technique fonctionne bien lorsque le mot de passe est très court. Voilà pourquoi le mot de passe doit respecter les conditions précédentes.

Comment s'en souvenir ?

Voici un autre exemple de mot de passe : `mv@1c&rac`, il est du même style que le précédent et tout aussi facile à retenir. Mais en réalité ce mot de passe provient de cette phrase : **ma** voiture **a** 115 chevaux **et** roule **au** colza. Voici comment se souvenir d'un mot de passe, mais dans ce mot de passe ne doit pas figurer un mot du dictionnaire. Vous n'avez qu'à faire travailler votre imagination pour vous souvenir d'un mot de passe robuste.

A bannir

Il a été précédemment expliqué comment obtenir mot de passe robuste et ceci est aussi valable pour votre téléphone portable, carte de crédit et autres. Il ne faut surtout pas utiliser des chiffres qui vous sont associés, comme : toutes les dates (à la limite celle où vous avez perdu votre première dent !!), numéro de plaque, numéro de téléphone, etc... tous ces chiffres vous sont associés et ce sont les premiers testés donc à proscrire.

Mise en garde

Lorsque vous êtes au bureau ou dans la rue faites attention au regard indiscret des autres, comme : vos collègues de bureau, votre voisin au bancomat, etc... Et comme il a été dit au chapitre 2.3 page 11, méfiez-vous des techniques de ‘social engineering’.

Solutions

Les mots de passe ne peuvent pas être exclus, mais ils peuvent être associés à d’autres techniques comme :

La biométrie

La biométrie est une science comportementale ou physiologique de la personne permettant de reconnaître un individu sans ambiguïté ; les méthodes les plus connues sont :

- L’empreinte digitale
- La rétine
- La voix

Pour plus d’informations sur la biométrie, consultez la référence [RL 10].

Les PKI

L’utilisateur possédant une clé (token) ou une carte à puce, dispose des ses certificats à l’intérieur de celle-ci, ce qui évite de laisser toute trace des certificats sur le disque dur de l’ordinateur. Par ce système, la clé est transportable et utilisable sur différents postes. De cette manière c’est la clé qui réalise l’authentification numérique de l’utilisateur, et celui-ci pourra s’en servir pour signer et chiffrer des documents. Par contre, cela implique une infrastructure réseau au niveau de l’entreprise pour pouvoir authentifier les utilisateurs, tels que serveurs de certificats, annuaire LDAP, etc.).

Changement de mot de passe

Pour éviter de se souvenir du mot de passe, il y a des systèmes qui changent de mot de passe régulièrement (environ toutes les minutes). Ce changement s’opère chez le client et chez le serveur (de telle manière à avoir le même, ce qui serait mieux). Ces systèmes sont très employés dans le cas du télétravail pour pouvoir se connecter à l’entreprise en créant un VPN.

4 Intrusion Detection System

4.1 PORT DE SÉCURITÉ

Ce n'est pas un système de détection conçu pour cela, c'est juste un mode de configuration du switch qui permettra de détecter des anomalies ou des intrusions sur le réseau. Le switch employé a la possibilité de configurer les ports comme étant sûrs et lorsque cette configuration est établie, le port ne peut accepter qu'un nombre maximum définis d'adresses MAC sur le port. Le nombre maximum est de 132 mais il peut être modifié selon les besoins, cependant toutes les adresses MAC des ordinateurs seront mises en mode statique dans la table CAM. Pour entrer les adresses MAC en statique, il est possible de le faire par le menu de configuration du switch ou simplement en branchant la machine sur le port désiré.

Lorsqu'un problème survient, le switch envoie des messages SNMP à l'administrateur en spécifiant le type d'alerte qui vient de survenir. Reprenons par exemple l'attaque de déni de service du chapitre 3.1.1.1 de la page 17 qui consistait à avoir la même adresse MAC que la victime de telle manière à prendre tout le trafic qui lui était destiné. Alors dans ce cas avec les ports sécurisés, il ne peut y avoir qu'une seule fois la même adresse MAC, par conséquent, lorsque le hacker se branche sur un autre port avec cette même adresses MAC des messages SNMP trap sont envoyés à l'administrateur réseau. Sur la Figure 66 on peut voir le message SNMP qui a été envoyé par le switch, lorsqu'une machine s'est branchée sur un port et que l'adresse MAC était déjà enregistrée sur un autre port.

```

Simple Network Management Protocol
  Version: 1
  Community: public
  PDU type: TRAP-V1
  Enterprise: 1.3.6.1.4.1.437.1.1.3
  Agent address: 10.192.72.100
  Trap type: ENTERPRISE SPECIFIC
  Specific trap type: 3 (0x3)
  Timestamp: 205447
  Object identifier 1: 1.3.6.1.2.1.2.2.1.1.9
  Value: INTEGER: 9 (0x9)

```

Figure 66 Message SNMP trap provenant du switch (capture Ethereal)

On peut constater les différents champs qui nous informent d'un problème :

- Version : version 1, par conséquent les mots de passe sont transmis en clair (gare à *dsniff*).
- Entreprise : d'après la référence [RW 24], cette valeur a comme signification : 'ipAddressChange'.
- Agent address : adresse IP du switch qui a émis l'alerte.
- Trap type : spécifique à Cisco.
- Timestamp : Temps écoulé depuis la dernière réinitialisation de l'entité.
- Object identifier : spécifie que c'est l'interface numéro 9 du switch.

Sur la référence [RW 25], la valeur 1.3.6.1.2.1.2.2.1.1 spécifie les interfaces du switch.

Lorsqu'il y a saturation d'un des ports par une attaque de type *macof*, l'alerte est la même que celle de la Figure 66, mis à part qu'il y aura la valeur du champ 'value' qui différera et qui indiquera le port qui a subi l'attaque.

Avec un outil d'administrateur de protocole SNMP fourni par le constructeur (Cisco), l'administrateur devrait avoir plus d'informations concernant le problème survenu sur le switch, étant donné que certaines valeurs des champs sont propres à Cisco.

Inconvénient

L'inconvénient c'est que le protocole qui s'occupe d'acheminer les trames SNMP est le protocole UDP et celui-ci envoie les paquets sans attendre un acquittement pour savoir s'il est bien arrivé. Par conséquent, si le hacker fait une quelconque attaque et qu'il envoie des réponses ARP en spécifiant que l'adresse MAC de l'administrateur et la sienne, dans ce cas l'administrateur ne sera pas au courant qu'une attaque survient sur son réseau. Il en sera au courant seulement quand le hacker arrêtera d'empoisonner le cache ARP du switch.

Si le switch est configuré avec les ports sécurisés, et que le switch se trouve dans un laboratoire, l'administrateur recevra sans cesse des messages indiquant que les machines ont changé de port et les utilisateurs ne pourront pas changer leur machine de place.

Investigation

Pour rétablir un port bloqué, il faut que l'administrateur le reconfigure et, dans le menu qui sert à le réactiver, il est spécifié pour quelle raison le port était bloqué, puis en allant regarder la liste des adresses MAC mises en statiques, il est possible de déterminer si l'attaque est du type *macof* (la table doit être pleine avec des adresses MAC étranges, surtout si les machines de l'entreprise sont achetées chez le même fabricant) à été orchestrée, ou si c'est un doublon d'adresse MAC.

Conclusion

Ce type de configuration apporte au réseau un niveau de sécurité en plus mais selon la politique de l'entreprise et par l'utilisation d'ordinateur portable, cette stratégie amène vite des complications. Il a été démontré que cette protection n'est pas fiable à 100% (comme tout d'ailleurs).

4.2 ARPWATCH ET ARPSNMP

Pour pouvoir détecter ce type d'attaque, il existe un moyen qui consiste à mettre une machine qui écoute tous les messages transitant sur le réseau de telle manière à collecter toutes les correspondances des adresses MAC et IP. Pour cela, il va falloir configurer les éléments de réseau pour que cette machine puisse tout écouter, car si celle-ci est sur un hub il n'y a pas de problème mais, sur un switch, elle ne pourra écouter que les messages qui lui sont destinés. Une fois cette configuration effectuée, la machine est en mesure d'écouter tout le trafic. Le logiciel *arpwatch* ou *arpsnmp* permet de stocker toutes les correspondances et lorsqu'un problème surgit, les informations sont notifiées par mail et par syslog.

Idéalement, la machine qui serait sur ce port d'écoute devrait être un serveur DHCP avec un programme de surveillance, de telle manière à ce que cette machine distribue les adresses et les contrôles en même temps. Il y a deux choix pour le programme de surveillance :

Arpwatch

Arpwatch stocke dans un fichier la correspondance des adresses et lorsqu'un problème survient, il notifie par mail et syslog à l'administrateur du réseau. Ce programme peut fonctionner seulement s'il n'y a qu'un seul réseau IP sur le segment où il est positionné. S'il y a d'autres adresses (différentes de celle du réseau de la machine) il notifiera les adresses IP différant de son réseau.

Arpsnmp

Arpsnmp n'écoute pas directement sur un port, il récolte les informations par le protocole SNMP sur des agents tels que des routeurs, switches, etc... Une fois les informations récoltées, il va les stocker sur un fichier et lorsqu'un problème survient il notifie aussi par mail et par syslog. Ce programme est plus approprié lorsqu'une entreprise possède plusieurs réseaux IP.

En ce qui concerne le choix du programme, ce sera *arpwatch* étant donné que les tests seront effectués sur un réseau IP isolé du reste du réseau de l'école.

4.2.1 ARPWATCH

Arpwatch va permettre de détecter les changements de correspondance d'adresses MAC et IP. De cette manière l'ARPspoofing va être détecté et de même le problème de gestion d'adresses IP lorsque plusieurs machines disposent de la même adresse IP. Il y a plusieurs types d'actions sur le réseau tels que :

- New station notifie la détection d'une nouvelle machine sur le réseau
- New activity notifie lorsqu'une machine n'avait plus d'activité depuis plus de 6 mois.
- Changed ethernet address notifie lorsqu'une adresse IP a changé d'adresse MAC.
- Reused old ethernet address notifie lorsqu'une adresse IP reprend une ancienne adresse MAC répertoriée.
- Flip flop notifie lorsqu'une adresse IP a permuté entre deux adresses MAC en moins de 24 heures.

Ce programme, comme *arpsnmp*, fonctionne uniquement sous Linux. En ce qui concerne l'installation, sur une distribution Debian, il faut suivre les instructions qui suivent.

Installation

Pour installer le programme il faut taper la commande : `apt-get install arpwatch`

Puis il faut créer le fichier `arp.dat` (ou d'un autre nom) qui contiendra la table de correspondance. Celui-ci doit être créé dans le répertoire de travail qui est `/usr/share/arpwatch/` ; dans ce répertoire doit aussi se trouver le fichier `ethercodes.dat`.

Puis, pour lancer le programme, il faut insérer l'une des commandes suivantes :

<code>arpwatch -f /usr/share/arpwatch/arp.dat</code>	spécifie où se trouve le fichier de correspondance
<code>arpwatch -r /etc/arpwatch.conf</code>	spécifie où se trouve le fichier de configuration

Il est préférable d'utiliser la seconde commande car ainsi le programme va chercher dans le fichier `arpwatch.conf` la configuration requise. De plus, lorsqu'il y a une panne et que le serveur redémarre, le programme va aller chercher les informations dans ce même fichier. Dans ce fichier il suffit de mettre l'instruction suivante :

```
Eth0 -f /usr/share/arpwatch/arp.dat
```

Essai

Avant toute chose, il faut configurer le réseau. Les tests ont été effectués sur un switch Cisco Catalyst 1900. Sur cet appareil, dans le menu principal, il y a un menu qui s'appelle 'Monitoring'. Par ce menu il est possible de configurer un port d'écoute. S'il y a plusieurs VLANs configurés, le port d'écoute peut écouter tous les VLANs. Pour plus d'informations sur la configuration du switch, consulter la référence [RL 6].

Sur la Figure 67 on peut voir la topologie du banc de test. Le programme *arpwatch* a été installé sur le système avec l'adresse IP 10.192.72.236 et le port numéro 1 du switch est configuré comme port d'écoute.

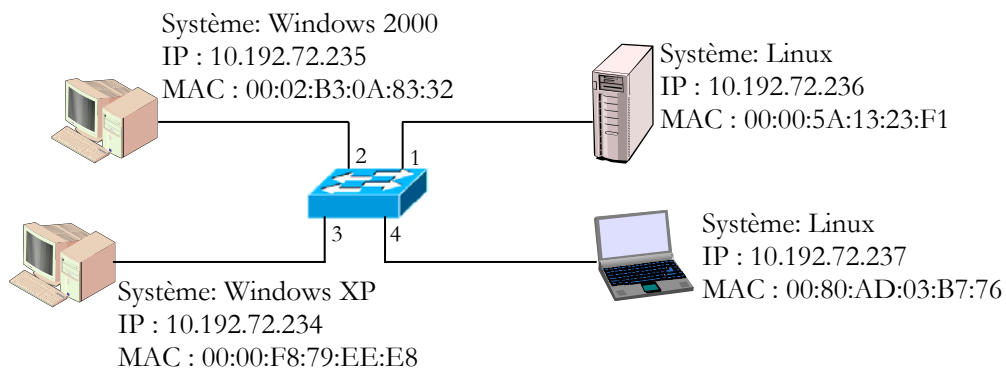


Figure 67 Topologie du banc de test de arpwatch

Lancement

Lors du lancement du programme, le fichier qui maintient la table de correspondance est vide, par conséquent le programme détecte des nouvelles stations sur le réseau. Le lancement a été effectué de cette manière : `arpwatch -f /usr/share/arpwatch/test.dat` et le fichier de correspondance dans ce cas est `test.dat`. Lors des premières trames émises sur le réseau, le programme a généré des messages 'New Station' pour signaler que des nouvelles machines ont été détectées. Ceci étant visible dans le fichier `syslog` par une commande qui permet de visualiser les dix dernières lignes : `tail /var/log/syslog`. Sur la Figure 68 on peut voir ce qui est dans le fichier `syslog`, de plus il est possible de voir les mails qui devraient être envoyés à l'administrateur lors de la génération d'un message (les mails ne peuvent être envoyés du fait que le banc de test est isolé du réseau). Ceux-ci peuvent être visibles dans un fichier `root` dont le chemin est `/var/mail/root`. La Figure 69 montre un des mails qui devraient être envoyés à l'administrateur.

```
Nov 21 11:21:17 EDU-PC151 arpwatch: listening on eth0
Nov 21 11:22:21 EDU-PC151 arpwatch: new station 10.192.72.234 0:0:f8:79:ee:e8
Nov 21 11:22:41 EDU-PC151 arpwatch: new station 10.192.72.235 0:2:b3:a:83:32
Nov 21 11:22:41 EDU-PC151 arpwatch: new station 10.192.72.236 0:0:5a:13:23:f1
Nov 21 11:23:01 EDU-PC151 arpwatch: report: pausing (cdepth 3)
Nov 21 11:23:01 EDU-PC151 /USR/SBIN/CRON[1572]: (mail) CMD ( if [ -x /usr/sbin/
exim -a -f /etc/exim/exim.conf ]; then /usr/sbin/exim -q ; fi)
Nov 21 11:24:01 EDU-PC151 arpwatch: new station 10.192.72.237 0:80:ad:3:b7:76
EDU-PC151:~#
You have mail in /var/mail/root
```

Figure 68 New Station dans le fichier `syslog`

```
From: arpwatch@edu-pc151.mail (Arpwatch)
To: root@edu-pc151.mail
Subject: new station
Message-Id: <E18EoW6-0000Pg-00@EDU-PC151.Diplome>
Date: Thu, 21 Nov 2002 11:25:02 +0100

    hostname: <unknown>
    ip address: 10.192.72.237
    ethernet address: 0:80:ad:3:b7:76
    ethernet vendor: Cnet Technology, Inc. [CNet Technology Used by Telebit (am
ong others)]
    timestamp: Thursday, November 21, 2002 11:22:12 +0100
```

Figure 69 Mail détectant une nouvelle station (New Station)

Dans le fichier `test.dat`, il est possible de voir la correspondance entre les adresses MAC et IP des machines du banc de test qui ont été détectées, ceci étant visible sur la Figure 70.

```
EDU-PC151:~# cat /usr/share/arpwatch/test.dat
O:0:f8:79:ee:e8 10.192.72.234 1037876591
O:2:b3:a:83:32 10.192.72.235 1037878394
O:0:5a:13:23:f1 10.192.72.236 1037877819 EDU-PC151
O:80:ad:3:b7:76 10.192.72.237 1037876591
```

Figure 70 Table de correspondance au lancement

Arpspoofing

Maintenant que les quatre machines ont été détectées par le programme, l'essai consiste à faire de l'ARPspoofing sur la machine 10.192.72.234 (victime) depuis le poste 10.192.72.237 (hacker) voulant se faire passer pour la machine 10.192.72.235. L'ARPspoofing n'a pas duré très longtemps, juste le temps nécessaire d'empoisonner le cache ARP de la victime et d'y remettre la bonne adresse MAC une fois l'opération terminée. Sur la Figure 71 on peut voir ce que le programme a inscrit dans le fichier syslog. On constate que les messages indiquent que l'adresse IP 10.192.72.235 a changé d'adresse MAC, du fait que les messages ARP Reply (réponse ARP) envoyés par le hacker et destinés à la victime 10.192.72.234 indiquaient que l'adresse IP 10.192.72.235 se trouve à l'adresse MAC 00:80:AD:03:B7:76. Et le dernier message indique que l'adresse IP 10.192.72.235 a repris son ancienne adresse MAC. Par conséquent, les messages nous indiquent pour quelle adresse IP le hacker s'est fait passé et l'adresse MAC de la machine du hacker, mais il n'est possible de voir qui était la victime de cette attaque.

```
Nov 21 12:47:45 EDU-PC151 arpwatch: changed ethernet address 10.192.72.235 0:80:ad:3:b7:76 (0:2:b3:a:83:32)
Nov 21 12:48:13 EDU-PC151 arpwatch: flip flop 10.192.72.235 0:2:b3:a:83:32 (0:80:ad:3:b7:76)
Nov 21 12:48:33 EDU-PC151 arpwatch: flip flop 10.192.72.235 0:80:ad:3:b7:76 (0:2:b3:a:83:32)
Nov 21 12:48:54 EDU-PC151 arpwatch: report: pausing (cdepth 3)
Nov 21 12:49:26 EDU-PC151 arpwatch: flip flop 10.192.72.235 0:2:b3:a:83:32 (0:80:ad:3:b7:76)
Nov 21 12:49:46 EDU-PC151 arpwatch: report: pausing (cdepth 3)
Nov 21 12:49:54 EDU-PC151 arpwatch: flip flop 10.192.72.235 0:80:ad:3:b7:76 (0:2:b3:a:83:32)
Nov 21 12:50:14 EDU-PC151 arpwatch: flip flop 10.192.72.235 0:2:b3:a:83:32 (0:80:ad:3:b7:76)
```

Figure 71 Flip flop dans le fichier syslog

Le programme a modifié le fichier test.dat qui contient une nouvelle correspondance sur l'adresse IP 10.192.72.235 : celle-ci est visible sur la Figure 72.

```
EDU-PC151:~# cat /usr/share/arpwatch/test.dat
O:0:f8:79:ee:e8 10.192.72.234 1037879293
O:2:b3:a:83:32 10.192.72.235 1037879421
O:80:ad:3:b7:76 10.192.72.235 1037879418
O:0:5a:13:23:f1 10.192.72.236 1037879515 EDU-PC151
O:80:ad:3:b7:76 10.192.72.237 1037879295
```

Figure 72 Table de correspondance après ARPspoofing

Sur la Figure 73 on peut voir le mail qui serait envoyé à l'administrateur, celui-ci est le premier mail qui indique l'ancienne et la nouvelle adresse MAC de l'adresse 10.192.72.235, puis lorsque le hacker remet la bonne adresse MAC dans le cache ARP de la victime le programme enverrait un autre mail à l'administrateur pour indiquer un nouveau changement d'adresse MAC. Ce deuxième mail correspond au dernier message affiché sur la Figure 71.

```
From: arpwatch@edu-pc151.mail (Arpwatch)
To: root@edu-pc151.mail
Subject: flip flop
Message-Id: <E18EpsN-0000RD-00@EDU-PC151.Diplome>
Date: Thu, 21 Nov 2002 12:52:07 +0100

        hostname: <unknown>
        ip address: 10.192.72.235
        ethernet address: 0:80:ad:3:b7:76
        ethernet vendor: Cnet Technology, Inc. [CNet Technology Used by Telebit (among others)]
old ethernet address: 0:2:b3:a:83:32
old ethernet vendor: Intel Corporation
        timestamp: Thursday, November 21, 2002 12:50:02 +0100
previous timestamp: Thursday, November 21, 2002 12:50:01 +0100
        delta: 1 second
```

Figure 73 Mail détectant un changement d'adresse MAC pour une station (flip flop)

Doublon d'adresse IP

Ce n'est pas très fréquent mais il peut arriver qu'il y ait des doublons d'adresse IP ou qu'une machine fasse de l'IPspoofing sur le réseau et qu'elle ait une adresse existante. Pour savoir comment changer l'adresse IP du système, il faut revoir le chapitre 3.2.2 page 28 qui expliquait comment s'y prendre. Dans ce cas, nous avons une adresse IP associée à deux adresses MAC. Le programme détecte un changement d'adresse MAC pour la machine déjà existante. Sur la Figure 74 on peut voir le premier message 'changed ethernet address' et les messages suivants 'flip flop'. Dans ce cas aussi, des mails sont sensés être envoyés à l'administrateur mais ils restent semblables à ceux vus précédemment.

```
Nov 21 15:19:27 EDU-PC151 arpwatch: changed ethernet address 10.192.72.234 0:2:b3:a:83:32 (0:0:f8:79:ee:e8)
Nov 21 15:19:47 EDU-PC151 arpwatch: flip flop 10.192.72.234 0:0:f8:79:ee:e8 (0:2:b3:a:83:32)
Nov 21 15:22:11 EDU-PC151 arpwatch: flip flop 10.192.72.234 0:2:b3:a:83:32 (0:0:f8:79:ee:e8)
Nov 21 15:22:31 EDU-PC151 arpwatch: flip flop 10.192.72.234 0:0:f8:79:ee:e8 (0:2:b3:a:83:32)
```

Figure 74 Changement d'adresse IP dans le fichier syslog

Doublon d'adresse MAC

C'est le cas de figure qui a été montré au chapitre 3.1.1 page 17, c'est à dire le déni de service sur un utilisateur dont il y avait un doublon d'adresse MAC pour empêcher l'autre utilisateur d'accéder au réseau. Dans ce cas, le programme va détecter que la personne voulant créer un déni de service a changé d'adresse MAC. Mais cela pourrait être aussi un utilisateur qui a changé de carte réseau à son ordinateur.

```
Nov 21 16:50:45 EDU-PC151 arpwatch: changed ethernet address 10.192.72.237 0:0:f8:79:ee:e8 (0:0:5a:13:23:f1)
```

Figure 75 Changement d'adresse MAC dans le fichier syslog

Par contre, si une personne change l'adresse MAC et l'adresse IP et que ces deux adresses appartiennent à la machine victime du déni de service, dans ce cas le programme ne détecte rien et le déni de service est réussi.

Investigation

Il a été vu plus haut que ce système de détection d'intrusion permettait de détecter : de l'ARPspoofing, doublon d'adresse IP, doublon d'adresse MAC mais pas le doublon des deux adresses.

Si ce logiciel est installé dans une entreprise dont toutes les machines sont connues et qu'un employé mal intentionné fait de l'ARPspoofing (dans le but de prendre le mot de passe du voisin), le logiciel va détecter que le routeur ou un serveur a changé d'adresse MAC, qui est en réalité l'adresse MAC de l'employé. Le seul indice qui mène au hacker est l'adresse MAC. Par conséquent, si l'entreprise a répertorié toutes les caractéristiques des ordinateurs de tout le personnel pour démasquer l'employé mal intentionné (il est peu probable que ce soit le cas), alors il faut utiliser le programme *ettercap* qui permettrait de connaître l'adresse IP du coupable (*ettercap* va faire des requêtes ARP sur toutes les adresses IP du réseau). Maintenant il va falloir déterminer dans quel bureau se trouve cet employé, pour cela il faut consulter les MIBs des éléments réseau (l'attaque est de l'ARPspoofing, donc il se trouve dans le même réseau que la machine ayant *arpsnmp*). En utilisant *snmpwalk* et en spécifiant l'object ID : 1.3.6.1.2.1.4.22.1.2, on aura la liste des adresses IP et MAC des machines connectées sur le switch ainsi que le numéro des ports. Cela est à faire pour tous les switch du réseau. Puis, il ne reste plus qu'à suivre le câble pour trouver l'employé mal intentionné.

Par contre, si ce n'est pas un employé mais un visiteur qui se connecte sur le réseau, il faudra agir rapidement pour le démasquer sous peine de le laisser filer.

Par ce système, il est difficile d'être certain qu'il y a une attaque sur le réseau, il se peut que ce soit une mauvaise configuration de l'adresse IP ou un changement de carte réseau. Pour cela, il vaut mieux faire une recherche pour en être certain.

Mais généralement l'ARPspoofing est appliqué en se faisant passer soit pour un routeur soit pour un serveur. De plus, pour ne pas prendre le flux asymétrique, l'ARPspoofing se fera sur la victime et sur le routeur ou serveur. Et dans ce cas, il sera plus facile de repérer l'attaque.

5 OUTILS

5.1 NESSUS

C'est un scanner non payant qui permet de sonder différentes plates-formes UNIX et Windows. Ce scanner fonctionne grâce à un système de client et de serveur : le serveur fonctionne sur une machine UNIX tandis que le client fonctionne sur UNIX et Windows (référence [RW 2]).

Installation du serveur

Pour commencer, il faut écrire la commande `apt-get install nessusd`. Une fois l'installation terminée, il faut créer l'utilisateur du serveur Nessus : pour cela il y a le script `nessus-adduser` qui se trouve dans `/usr/sbin/`. Une fois lancé ce script, il faut entrer le paramètre de l'utilisateur.

1. Login : nom de login de l'utilisateur.
2. Authentication method : permet de choisir la méthode de sauvegarde du mot de passe (cipher ou plaintext). Par défaut c'est 'cipher', qui veut dire chiffré.
3. Source host or network : permet de spécifier par quelle adresse IP ou quel réseau l'utilisateur peut se connecter. Par défaut c'est 'anywhere', qui veut dire partout.
4. One time password : mot de passe de l'utilisateur
5. User rules : permet de contraindre les adresses ou réseaux pouvant être scannés par l'utilisateur. Ceci se faisant par la syntaxe `accept|deny ip/mask`. Pour plus d'info consulter `man nessus-adduser`, si rien n'est spécifié, l'utilisateur peut tout scanner. Une fois terminé, presser Ctrl-D.
6. Is that ok ? le système affiche toutes les données insérées et il suffit de confirmer par y ou n.
7. User added confirmation par le système.

La configuration du serveur est sauvegardée dans le fichier `nessud.conf` qui se situe dans `/etc/nessus/`. Pour le visualiser il suffit d'écrire la commande `nessusd -s`. Une fois terminé, il faut lancer le serveur : pour cela il faut écrire la commande `nessusd -D`, l'option D permet au serveur de fonctionner en arrière-plan.

Installation du client

Pour l'installation du client sous Linux il faut écrire la commande `apt-get install nessus`, puis pour le lancer, il suffit d'écrire `nessus`.

Pour l'installation de client sous Windows, il faut aller chercher sur le site www.nessus.org, puis aller dans le champ 'download' et télécharger le client Windows. Une fois téléchargé le dossier 'WinNessus.zip' il suffit de prendre l'exécutable et les bibliothèques qui sont jointes et de les placer sur le poste.

La suite des explications est identique pour les deux clients. Lors du lancement du client Nessus, celui-ci génère la clé privée personnelle (private personal key) de l'utilisateur et pour protéger cette clé le système demande de rentrer une phrase personnelle (passphrase) qui sera demandée à chaque fois que le programme sera utilisé. Puis il faut configurer l'adresse IP du serveur, le port, la méthode de cryptage et le nom de l'utilisateur, comme l'illustre la Figure 76. Référence de l'installation [RW 3]

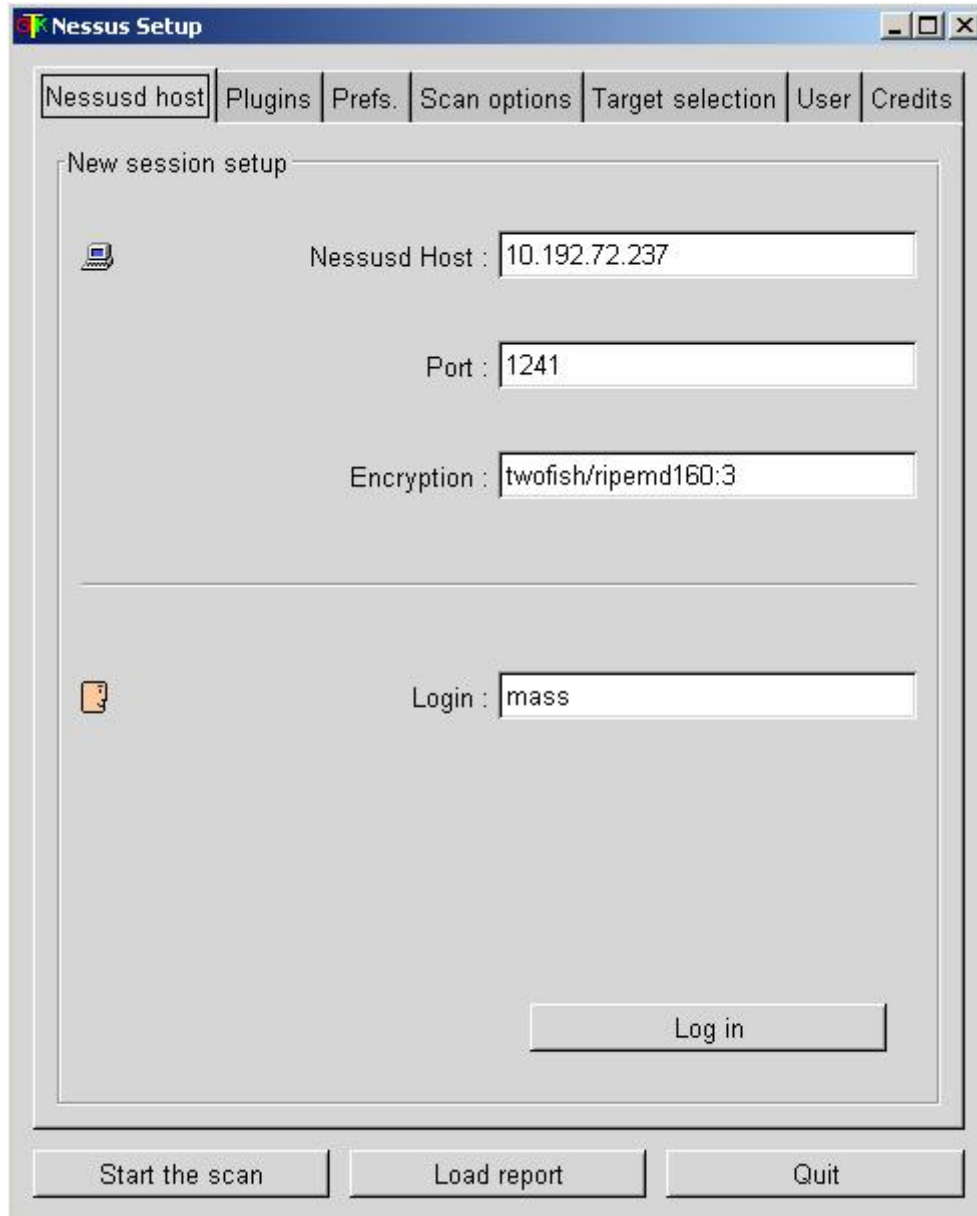


Figure 76 Configuration du client Nessus

5.2 WINARP

5.2.1 WINARP_SK

Cet outil permet de forger ses propres messages ARP de telle manière à spécifier certains champs du protocole ARP et les adresses MAC de la trame MAC. Sur la Figure 77, l'on peut voir la trame ARP qui est encapsulée dans la trame MAC de la Figure 15 de la page 22.

2	2	1	1	2	6	4	6	4
Type de matériel	Type de protocole	HLen	PLen	Opération	Add. MAC source	Add. IP source	Add. MAC destination	Add. IP destination

Figure 77 Trame ARP (taille en bytes)

- Type de matériel :** ce champ indique le type de réseau, sur le réseau Ethernet 10 Mbits/s ce champ vaut 1.
- Type de protocole :** ce champ indique le type de protocole de la couche supérieure pour laquelle l'adresse matérielle est demandée. Dans les réseaux TCP/IP, l'adresse protocole est IP et le champ vaut 800 en hexadécimale.
- Hlen :** ce champ correspond à la longueur en octets de l'adresse matérielle. Ce champ est déterminé par le champ type de matériel.
- Plen :** ce champ correspond à la longueur de l'adresse de protocole en octets. Ce champ est déterminé par le champ type de protocole.
- Opération :** ce champ indique si le paquet contient une requête ARP ou une réponse ARP. Requête = 1, réponse = 2.
- Add MAC source :** ce champ contient l'adresse MAC du nœud ayant émis le message ARP ; dans le cas d'une réponse, c'est cette adresse qui sera prise en compte pour la relation IP – MAC du cache ARP.
- Add IP source :** ce champ contient l'adresse IP du nœud ayant émis le message ARP.
- Add MAC dest. :** ce champ contient l'adresse MAC du nœud pour lequel le message ARP est émis. Dans le cas d'une requête, l'adresse MAC n'est pas connue alors ce champ vaut 00-00-00-00-00-00.
- Add IP dest. :** ce champ contient l'adresse IP du nœud pour lequel le message ARP est émis.

Pour plus d'informations sur ces champs qui viennent d'être spécifiés, voir référence [RL 1] page 197.

Ce programme est disponible sur www.arp-sk.org. Pour pouvoir lancer ce programme, il faut utiliser une fenêtre DOS et en tapant le nom du programme *winarp_sk* celui-ci nous affiche les options pour l'utilisation (Figure 78).

WinARP Swiss Knife version 0.9.2

```
usage: winarp_sk -m mode [-D dst_ether_addr] [-S src_ether_addr] [-F sender_MAC]
-d sender_IP [-T target_MAC] -s target_IP [-t delay] [-c count]
```

Ethernet options:

```
-D ethernet address of destination [MAC of ARP target]
-S ethernet address of source [selected adapter MAC address]
```

ARP options:

```
-m ARP mode <request = 1 and reply = 2>
-F MAC address of sender [selected adapter MAC address]
-s IP address of sender
-T MAC address of target [MAC of ARP target]
-d IP address of target
```

Misc. options:

```
-c number of packets to send [infinity]
-t time between successive packets in ms [2000 ms]
-h help
```

Standalone options:

```
-a show ethernet address of adapter
-i show ip address
-g ip_addr : get the remote MAC address of a host
```

Figure 78 Options de Winarp_sk

Il y a différentes options mais les plus importantes sont les suivantes :

- D spécifie l'adresse MAC de destination de la trame Ethernet.
- S spécifie l'adresse MAC source de la trame Ethernet.
- m spécifie si c'est une requête (1) ou une réponse (2).
- F spécifie l'adresse MAC du nœud émettant le message.
- s spécifie l'adresse IP du nœud émettant le message.
- T spécifie l'adresse MAC du nœud recevant le message ; si c'est une requête, mettre 00-00-00-00-00-00.
- d spécifie l'adresse IP du nœud recevant le message.
- t spécifie l'intervalle entre chaque paquet, par défaut 2000 ms.

5.2.2 WINARP_TCOM

Ce programme a été conçu dans le cadre de ce travail de diplôme. Il permet les mêmes fonctionnalités que le programme de base (*winarp_sk*) mais ayant une option supplémentaire (-N) qui lui permet d'envoyer des trames ARP avec l'adresse source différente (Figure 79).

WinARP Swiss Knife version 0.9.2

Modifié par Iritano Massimo pour Tcom

```
usage: winarp_sk -m mode [-D dst_ether_addr] -S src_ether_addr -N nb_src_ether
ddr [-F sender_MAC] -d sender_IP [-T target_MAC] -s target_IP [-t delay] [-c c
nt]
```

Ethernet options:

```
-D adresse MAC de destination [MAC of ARP target]
-S adresse MAC source
-N nombres d'adresses MAC différentes, max 2000 (à partir de la source)
```

ARP options:

```
-m ARP mode (request = 1 and reply = 2)
-F MAC address of sender [selected adapter MAC address]
-s IP address of sender
-T MAC address of target [MAC of ARP target]
-d IP address of target
```

Misc. options:

```
-c number of packets to send [infinity]
-t time between successive packets in ms [0 ms]
-h help
```

Standalone options:

```
-a show ethernet address of adapter
-i show ip address
-g ip_addr : get the remote MAC address of a host
```

Figure 79 Options de Winarp_tcom

Toutes les options sont identiques sauf celles-ci :

- S spécifie l'adresse MAC source de la trame Ethernet, mais seulement la partie constructeur sera prise en compte.
- N permet de spécifier le nombre d'adresses MAC sources différentes à envoyer. A partir de celle fournie à l'option S, les 3 derniers bytes sont mis à '0' puis incrémentés par un compteur jusqu'au nombre N.

Pour réaliser ce programme, il a fallu reprendre le code source qui était disponible sur le site www.arp-sk.org et prendre aussi les codes sources de la librairie WinPcap sur le site <http://winpcap.polito.it> (puis dans le champ 'download'). Cette librairie fut nécessaire pour la compilation du programme. Le seul fichier qui fut modifié est winarp-sk.cpp, qui est disponible sur : CD/Intrusion/Programme/winarp/winarp_sk/compilation/winarp-sk.cpp

5.3 DSNIFF

Dsniff est une collection d'outils pour sniffer le réseau et faire des tests d'intrusion, cet outil étant disponible uniquement pour les plates-formes Linux (référence [RW 22]). Pour l'installation de cet outil il a fallu taper la commande `apt-get install dsniff`.

5.3.1 ARPSPOOF

Cet outil permet d'empoisonner le cache ARP d'une victime de telle manière à fausser la correspondance IP – MAC du cache ARP de la victime. Pour le lancer il faut insérer la commande :

```
Arpspoof [-i interface] [-t IP_victime] IP_du_récepteur
```

-i interface	spécifie l'interface à utiliser
-t IP_victime	spécifie l'adresse IP de la victime, si cette option n'est pas spécifiée alors les messages seront diffusés en broadcast à toutes les machines du réseau
IP_du_récepteur	spécifie pour quelle adresse IP on veut se faire passer, généralement cela sera l'adresse du default gateway ou d'un serveur

Lorsque ce programme est lancé, il va faire une requête ARP sur l'adresse 'IP du récepteur' de telle manière à connaître cette adresse MAC. Puis il envoie des réponses ARP à la victime spécifiant que l'adresse 'IP du récepteur' se trouve sur l'adresse MAC de la machine envoyant ces messages (hacker). Ces réponses ARP sont envoyées à intervalle d'environ 3 secondes sur la victime de telle manière à ce que l'ordinateur de la victime ne fasse pas une requête ARP. Lorsque le programme est arrêté, il va envoyer trois réponses ARP avec la bonne adresse MAC qu'il avait collectée auparavant, pour que la victime ne s'aperçoive de rien sinon elle aurait un déni de service lorsque le programme est arrêté.

5.3.2 MACOF

Cet outil permet d'inonder un réseau avec des adresses MAC sources aléatoires. Il n'envoie que des trames TCP SYN et il est possible de définir différentes options à partir de la commande :

```
macof [-i interface] [-s scr] [-d dst] [-e tha] [-x sport]  
      [-y dport] [-n times]
```

-i interface	spécifie l'interface à utiliser
-s scr	spécifie l'adresse IP source
-d dst	spécifie l'adresse IP de destination
-e tha	spécifie l'adresse MAC de destination
-e tha	spécifie l'adresse MAC de destination
-x sport	spécifie le port TCP source
-y dport	spécifie le port TCP de destination
-n times	spécifie le nombre de paquets à envoyer

Tous ces champs sont des options si ceux-ci ne sont pas spécifiés, alors ils prendront des valeurs aléatoires comme l'adresse MAC source et si le nombre de paquets n'est pas spécifié alors le programme ne s'arrête pas d'envoyer des trames.

5.3.3 DNSSPOOF

Cet outil permet de contrefaire les réponses aux requêtes DNS ; pour le lancer, il suffit de taper la commande :

```
dnsspoof [-i interface] [-f hostfile] [expression]
```

-i interface	spécifie l'interface à utiliser
-f fichier	spécifie le chemin au fichier
expression	spécifie l'expression d'un filtre <i>tcpdump</i> , pour filtrer le trafic à analyser. Pour plus d'informations, consultez <code>man tcpdump</code> ou la référence [RL 8].

Le fichier permet de spécifier le nom des sites à contrefaire les réponses DNS. Si ce fichier n'est pas spécifié alors le programme renverra sa propre adresse IP à toutes les requêtes DNS. Le fichier doit contenir l'adresse IP et le nom du site, comme cela :

```
X.X.X.X      *.truc.com
```

Pour plus d'informations sur le format du fichier, consultez `man hosts`. Si aucune expression n'est spécifiée, le programme utilise ce filtre par défaut :

```
udp dst port 53 and not src adresse_IP_du_système
```

5.3.4 WEBMITM

Cet outil permet de faire l'attaque du 'man in the middle' lors des connections sécurisées sur Internet avec le protocole SSL. Pour le lancer il suffit de taper la commande :

```
webmitm [-d] [host]
```

-d	active le mode débogage. Peut être spécifié plusieurs fois pour plus d'effet.
Host	spécifie un système à qui relayer.

Lors du premier lancement du programme, celui-ci va générer la clé privée et va demander à l'utilisateur les champs du certificat à insérer, de manière à générer un certificat ayant des champs identiques à celui de l'original. Le certificat généré est nommé `webmitm.crt`. Les champs du certificat sont spécifiés au chapitre 5.7.1 page 77.

5.3.5 DSNIFF

Cet outil permet de renifler les mots de passe de différents protocoles tels que Telnet, FTP, SNMP, HTTP et bien d'autres. Pour le lancer il faut taper la commande :

```
dsniff [-c] [-d] [-m] [-n] [-i interface] [-s snaplen] [-f services]
      [-t trigger [,...]] [-r | -w fichier] [expression]
```

-c	permet de réassembler en half-duplex le flux TCP pour pouvoir supporter le trafic asymétrique lorsque l'ARPspoofing est fait que sur une seule machine.
-d	active le mode débogage
-m	active la détection automatique du protocole
-n	désactive la résolution de nom
-i interface	spécifie l'interface à utiliser
-s snaplen	spécifie le nombre de bytes à analyser de chaque connexion tcp, au lieu des 1024 par défaut
-f services	spécifie le fichier services pour charger les numéros de ports de chaque service. Par défaut dsniff.services. Pour plus d'informations, consultez man services.
-f trigger [,...]	spécifie et charge les numéros de ports de chaque service depuis une liste séparée par des virgules, spécifiée ainsi port/protocole=service (expl : 80/tcp=http).
-r fichier	spécifie le fichier à lire qui a été écrit par l'option -w.
-w fichier	spécifie le chemin et le nom du fichier ou sauve les mots de passes capturés
expression	spécifie l'expression d'un filtre <i>tcpdump</i> , pour filtrer le trafic à analyser. Pour plus d'informations, consultez man tcpdump ou la référence [RL 8].

Tous ces champs sont des champs à option, si aucun n'est spécifié, alors *dsniff* analysera tous les 1024 premiers bytes de toutes les trames TCP sur les protocoles définis dans le fichier dsniff.services.

5.4 ETTERCAP

Cet outil est un sniffer multiusages pour les réseaux LAN. Dans ce chapitre, tout ne sera pas spécifié au sujet de ce logiciel : si vous voulez plus d'informations, je vous envoie au man `ettercap`. Lorsque `ettercap` est lancé sans aucun paramètre, il va faire des requêtes ARP sur toutes les adresses IP des machines du réseau (pour connaître toutes les adresses du réseau il utilise son adresse IP et le masque du réseau et avec ces deux informations toutes les adresses sont connues). A partir des réponses ARP, il va pouvoir déterminer toutes les machines actives du réseau comme l'illustre la Figure 80.

```

ettercap 0.6.4
-----
5 hosts in this LAN (10.192.72.237 : 255.255.254.0)
1) 10.192.72.237 1) 10.192.72.237
2) 10.192.72.100 2) 10.192.72.100
3) 10.192.72.234 3) 10.192.72.234
4) 10.192.72.235 4) 10.192.72.235
5) 10.192.72.236 5) 10.192.72.236
    
```

Figure 80 Lancement d'ettercap

Avec les touches directionnelles du clavier (en bas à droite), on peut déplacer la bande de couleur verte qui se situe dans le carré principal bleu de la Figure 80. Puis, par la touche 'enter', on peut sélectionner les adresses IP, la colonne de droite indiquant les adresses IP sources et la colonne de gauche les adresses IP destination (Figure 81).

```

ettercap 0.6.4
SOURCE: 10.192.72.234
DEST  : 10.192.72.236
-----
5 hosts in this LAN (10.192.72.237 : 255.255.254.0)
1) 10.192.72.237 1) 10.192.72.237
2) 10.192.72.100 2) 10.192.72.100
3) 10.192.72.234 3) 10.192.72.234
4) 10.192.72.235 4) 10.192.72.235
5) 10.192.72.236 5) 10.192.72.236
Help Window
[qQ] [F10] - quit
[return]  - select the IP
[space]   - deselect the IPs
[tab]     - switch between source and dest
[aa]     - ARP poisoning based sniffing
           . for sniffing on switched LAN
           . for man-in-the-middle technique
[sS]     - IP based sniffing
[mM]     - MAC based sniffing
[dD]     - delete an entry from the list
[xX]     - Packet Forge
[pP]     - run a plugin
[fF]     - OS fingerprint
[oO]     - passive host identification
[cC]     - check for other poisoner...
[rR]     - refresh the list
[kK]     - save host list to a file
[hH]     - this help screen
    
```

Figure 81 Choix d'ettercap

En appuyant sur la touche 'h', on accède au menu de l'aide et là on peut voir qu'il y a plusieurs options à choix, comme l'option 'a' qui fait de l'ARPspoofing.

Ce programme aurait pu envoyer des requêtes ICMP au lieu des requêtes ARP mais certains systèmes ne répondent pas aux requêtes ICMP (selon la configuration), alors qu'avec le protocole ARP il est plus sûr de connaître toutes les machines.

5.5 TRACE ROUTE

Programme servant à indiquer la route qu'empruntent les paquets IP jusqu'à une destination. Il va indiquer les adresses IP des interfaces d'entrée de tous les éléments qui seront rencontrés sur le chemin. Le programme envoie des messages ICMP avec des valeurs différentes dans le champ TTL (Time to Live) de l'entête IP (Figure 14 page 21).

Linux

Pour installer ce programme il a fallu taper la commande `apt-get install traceroute`. Puis, pour l'utilisation, il suffit de taper l'une des commandes suivantes :

```
traceroute adresse_IP
traceroute nom_dns, exemple : traceroute yahoo.com
```

Windows

Sur les systèmes Windows, ce programme est déjà installé. Pour l'utiliser il faut ouvrir une fenêtre DOS et taper l'une des commandes suivantes :

```
tracert adresse_IP
tracert nom_dns, exemple : tracrt yahoo.com
```

5.6 SNMPWALK

Programme servant à aller consulter les MIBs des éléments réseau ; il est possible de consulter uniquement les MIBs et de les modifier.

Linux

Pour installer ce programme il a fallu taper la commande `apt-get install snmp`. Puis, pour l'utilisation, il suffit de taper l'une des commandes suivantes :

```
snmpwalk adresse_IP communauté Object_ID
    communauté           spécifie la communauté ; généralement pour consulter
                        c'est 'public'
    Object ID            spécifie l'object à vouloir consulter, exemple : 1.3.6.1.2.
```

Windows

L'utilisation est identique à celle de `snmpwalk` de Linux, il faut ouvrir une fenêtre DOS et taper la commande :

```
snmpwalk -M .\mibs adresse_IP communauté Object_ID
```

Ce programme est disponible sur : CD/Intrusion/Programme/snmpwalk.

5.7 AUTRES

5.7.1 SERVEUR APACHE-SSL

Le serveur Apache-SSL est un serveur de page Internet avec la sécurité du protocole SSL. Pour installer le serveur il faut taper la commande `apt-get install apache-ssl`.

Une fois que l'installation est terminée, le système va générer la clé privée qui sera mise dans le fichier `apache.pem` qui se trouve dans `/etc/apache-ssl/apache.pem`. Puis, il génère le fichier `ssleay.cnf` qui est un exemple de comment remplir le certificat du serveur : ce fichier se trouve dans `/usr/share/apache-ssl/ssleay.cnf`.

À partir de maintenant il va falloir remplir les champs du certificat du serveur :

1. Country Name (2 letter code) : CH
2. State or Province Name : Switzerland
3. Locality Name : Yverdon
4. Organisation Name : Tcom
5. Organisational Unit Name : B05
6. Server Name : `www.tcom.secure.ch`, c'est le nom qui se retrouve dans l'url du navigateur.
7. Email Address : `adminServerTcom@eivd.ch`

Une fois cette configuration terminée, le système va compléter le fichier `apache.pem` pour y mettre la partie du certificat (disponible sur CD/Intrusion/certificat apache/apache.pem). Ensuite le système va générer les fichiers de configuration qui sont `httpd.conf`, `access.conf` et `srm.conf` et qui se trouvent dans `/etc/apache-ssl/`.

5.7.2 HTACCESS

Lorsqu'on crée un site Web sur le serveur Apache, la page qui est consultée en premier est `index.html` puis depuis celle-ci on en appelle des autres qui se trouvent soit dans le même répertoire soit dans un autre répertoire. Le serveur Apache peut protéger un répertoire par la méthode `htaccess` qui demandera un mot de passe à l'utilisateur. Pour cela, il faut créer dans le dossier à protéger (qu'on appellera 'exemple') un fichier `.htaccess` (le point devant signale que c'est un fichier caché). Dans ce fichier il faut mettre les lignes suivantes :

```
AuthType Basic
AuthName "Autorisation requise"
AuthUserFile /var/www/exemple/pass/password
Require valid-user
```

Dans ce répertoire 'exemple', il y a un autre répertoire nommé 'pass' qui contient un fichier `password` qui contient le 'username' et le 'password' des utilisateurs. Il faut créer le dossier 'pass' et le fichier `password`, puis pour y insérer le mot de passe et le nom d'utilisateur, il faut le faire par la commande :

```
htpasswd -b /var/www/exemple/pas/password nom_d_utilisateur mot_de_passe
```

La méthode *htpasswd* va insérer dans le fichier le nom de l'utilisateur et le mot de passe qui sera encrypté. Puis, il faut aller modifier le fichier `httpd.conf` qui se trouve dans `/etc/apache/`. La modification à apporter est à la ligne numéro 328, et consiste à remplacer la ligne par `AllowOverride ALL` (auparavant il y avait `AllowOverride NONE`), ce qui spécifie que le serveur doit regarder dans les répertoires pour trouver un fichier `.htaccess`. Référence sur ce paragraphe [RW 26].

5.7.3 SCRIPT LINUX

Pour éviter d'écrire plusieurs fois les mêmes commandes, il est préférable de créer un script. Pour cela il suffit de créer un fichier sans extension du nom voulu et la première ligne du script doit être `#!/bin/sh`. Ensuite, il suffit de spécifier les commandes voulues. Pour que ce script soit exécuté, il faut lui changer les droits d'accès du fichier en le rendant exécutable par la commande :

```
Chmod +x nom_du_fichier
```

Puis, pour l'exécuter, il faut taper `./nom_du_fichier`.

6 TUTORIAL

Le tutorial destiné aux étudiants reprend uniquement les parties théoriques des chapitres vues précédemment, sans spécifier précisément comment les attaques doivent être orchestrées et sans le nom des outils permettant de le faire. Ceci dans le but de ne pas distribuer ce genre d'informations aux étudiants pour éviter qu'ils puissent eux-mêmes faire leurs propres tests dans l'école ou dans un autre encadrement. Document annexé à l'annexe 4.

7 LABORATOIRE

Le laboratoire a été réalisé sur la base des attaques faites durant le travail de diplôme, reprenant les plus représentatives de façon à montrer comment les attaques peuvent être réalisées et comment elles peuvent être évitées ou contrées.

En ce qui concerne le déroulement du laboratoire, celui-ci se déroule sur des bancs de test séparés du réseau de l'école, afin d'éviter d'éventuels dérapages de la part de étudiants durant leurs manipulations, étant donné qu'ils doivent appliquer certaines attaques comme des dénis de service ou même des captures de mot de passe.

Ces manipulations se feront sur des systèmes Windows et Linux Debian. Ces machines sont des systèmes déjà présents au laboratoire à l'exception de deux machines qui devront être démarrées sur des CD-Rom. Ces CD-Rom contiendront des systèmes Linux ainsi que des programmes utiles pour les intrusions. Cette précaution est prise dans le but de ne pas laisser ce genre d'outils sur les machines, du fait que ces machines sont accessibles aux étudiants sans aucun contrôle et les machines de laboratoire sont connectées au réseau de l'école.

Étant donné que l'utilisation d'ordinateurs portables est en nette augmentation, pour éviter que ces outils d'intrusion ne soient facilement installés sur les machines des étudiants, les outils employés au laboratoire seront renommés selon ces propositions :

macof	→	flooder
dnsspoof	→	changedns
webmitm	→	mim

Les étudiants devront disposer des deux CDs pour le laboratoire, les deux CD contiennent :

CD1	un serveur FTP un serveur apache-ssl
CD2	arpwatch dnsspoof et webmitm

La donnée du laboratoire se trouve à l'annexe 5 et le corrigé se trouve à l'annexe 6.

8 CONCLUSION

Pour pouvoir accroître la sécurité dans les entreprises, il faudrait commencer par sensibiliser le personnel de celle-ci de manière à montrer aux employés l'attitude à adopter dans certaines situations (social engineering, man in the middle), et montrer aux employés ainsi qu'aux administrateurs réseaux que des mots de passe circulent en clair sur le réseau ainsi il faudrait favoriser (si c'est possible) l'utilisation des méthodes d'encryption pour sécuriser les transmissions.

Il a été montré que des méthodes de détection d'intrusions sont nécessaires pour accroître la sécurité ainsi que la cryptographie assure une confidentialité des transmissions dans les réseaux. C'est pour cela qu'il est préférable de mettre en place ce type de protection.

L'objectif de ce travail de diplôme a été atteint, mais il resterait encore bien du travail à faire du fait que beaucoup de voies ont été ouvertes et entamées. Malheureusement, le temps m'a permis d'explorer certaines d'entre elles. Notamment une partie pratique d'hijacking et d'Ipspoofing et l'utilisation du détecteur d'intrusion *snort*.

Ce travail de diplôme m'aura permis d'approfondir un sujet en nette expansion ainsi que l'apprentissage du système Linux Debian qui est l'OS où fonctionne la majeure partie des programmes de hacking.

Yverdon, le 19.12.2002

Iritano Massimo

9 REMERCIEMENTS

J'aimerais remercier toutes les personnes qui m'ont aidé durant ces quelques semaines de travail de diplôme. Ces personnes sont :

- M. Ventura pour son suivi et ses conseils.
- M. Maret pour ses conseils apportés durant les séances.
- M. Tettamanti pour ses conseils dans l'utilisation du système Linux et ainsi que ses conseils dans le domaine des intrusions.
- M. Gachet pour ses conseils dans l'utilisation du système Linux et ainsi que ses conseils dans le domaine des PKI.
- M. Vares pour son expérience dans le domaine des MIBs.
- M. Andrades pour ses conseils dans la programmation C++.

10 RÉFÉRENCES

- [RL 1] TCP/IP
Karanjit S. Siyan
CampusPress (2001)
ISBN : 2-7440-1119-3
- [RL 2] Sécurité Optimale
Anonyme
CampusPress (1999)
ISBN : 2-7440-0723-4
- [RL 3] Hackers Attention Danger
Eric Cole
CampusPress (2001)
ISBN : 2-7440-1273-4
- [RL 4] Détection d'intrusions
Reis Nelson
Disponible : CD/Intrusion/documentation/Reis/rapport.pdf
- [RL 5] Identification d'adresses IP
Andrades Angel & Iritano Massimo
Disponible : CD/Intrusion /documentation/adresseIP.pdf
- [RL 6] Configuration Catalyst 1900
Iritano Massimo
Disponible : CD/Intrusion /documentation/Switch/catalyst1900.doc
- [RL 7] Hacking Layer 2
Sean Convery, Cisco Systems
Disponible : CD/Intrusion /documentation/hackingL2.pdf
www.blackhat.com
- [RL 8] man de tcpdump
Disponible : CD/Intrusion /documentation/man-tcpdump.pdf
- [RL 9] Exploitation avancée de buffer overflows
Gay Olivier
Département d'informatique de l'EPFL
Disponible : CD/Intrusion /documentation/buffer-overflow.pdf
- [RL 10] Biométrie
Baumgartner Karl
Disponible : CD/Intrusion /documentation/biometrie.pdf
- [RL 11] Sécurité et PKI
Gachet Pascal
Disponible : CD/Intrusion /documentation/sécurité et PKI.doc
- [RL 12] Tutorial VPN
Tettamanti Christian
Disponible : CD/Intrusion /documentation/Tutorial VPN.doc
- [RW 1] Sécurité info, rubrique Social Engineering
www.securiteinfo.com/attaques/divers/social.shtml
- [RW 2] Nessus, scanner de vulnérabilité
www.nessus.org/
- [RW 3] Introduction à Nessus, aide à l'installation
www.linuxfrench.net/article.php?id_article=938

- [RW 4] Rapport du test de vulnérabilité du serveur Linux
Disponible : CD/Intrusion/nessus/Linux serveur/index.html
- [RW 5] Rapport du test de vulnérabilité d'une machine Windows XP
Disponible : CD/ Intrusion/nessus/Windows XP/index.html
- [RW 6] ZoneLabs, site où se trouve le firewall ZoneAlarm
www.zonelabs.com
- [RW 7] Changer son adresse MAC
www.os3b.org/web/docs/tips/dhcp_and_mac_address
- [RW 8] Avis du certa, corruption du cache ARP des équipements Cisco
www.certa.ssi.gouv.fr/site/CERTA-2001-AVI-147/index.html.2.html
- [RW 9] Détails sur le ping de la mort
www.insecure.org/splotts/ping-o-death.html
- [RW 10] Détails sur la Land attaque
www.insecure.org/splotts/land.ip.DOS.html
- [RW 11] Miscmag magazine, article sur les attaques externes
www.security-labs.org/www.miscmag.com/articles/index.php3?page=106
- [RW 12] Guill, article sur l'IPspoofing
www.guill.net/index.php3?cat=4&sec=4
- [RW 13] Site introduisant les buffer overflows
www.montefiore.ulg.ac.be/~briquet/buff/buff.html
- [RW 14] Site parlant du routage sous Linux
<http://christian.caleca.free.fr/masquerade/ipchains.htm>
- [RW 15] Sécurité info, rubrique sur le DNS spoofing
www.securiteinfo.com/attaques/hacking/dnsspoofing.shtml
- [RW 16] Site spécifiant des configurations réseau
www.linux-france.org/prj/inetdoc/i/net/guides/Advanced-routing-Howto/Advanced-routing-Howto.v0.9-15.html
- [RW 17] Site donnant tous les mots de passe par défaut des éléments réseau
www.rezalfr.org/normes/passwords.htm
- [RW 18] Sécurité info, rubrique sur les troyens
www.securiteinfo.com/attaques/divers/troie.shtml
- [RW 19] Site permettant de trouver le programme Netwok Stumbler
www.netstumbler.com
- [RW 20] Site permettant de trouver le programme airtort
<http://airsnort.shmoo.com>
- [RW 21] arp-sk, site parlant de diverses attaques ARP
www.arp-sk.org
- [RW 22] groar..org, traduction en français de l'explication de dsniff
www.groar.org/trad/dsniff/dsniff-faq/faq.html
- [RW 23] Site où est disponible la librairie Winpcap
<http://winpcap.polito.it>
- [RW 24] Information sur le champ entreprise valant : 1.3.6.1.4.1.437.1.1.3
www.tivoli.com/support/public/Prodman/public_manuals/td/TRM/GC32-0703-01/fr_FR/HTML/user228.htm
- [RW 25] Information sur la MIB
www.alvestrand.no/objectid/1.3.6.1.2.1.2.2.1.1.html
- [RW 26] Tutorial sur htpaccess
www.essebe.ch/tuthtacc/index.php

11 GLOSSAIRE

ARP	Address Resolution Protocol : protocole utilisé dans les réseaux Ethernet pour la résolution d'adresses IP en une adresse MAC.
AP	Access Point : éléments réseau permettant la connexion au réseau de l'entreprise par une technologie sans fil (Wireless).
ASCII	American Standard Code for Information Interchange : la table ASCII est une correspondance entre des valeurs numériques et des caractères.
Backdoor	Porte dérobée : lorsqu'une intrusion est réussie, le hacker peut laisser un backdoor, ce qui correspond à une porte cachée faite généralement par un troyen, par un port ouvert, par un compte FTP ou encore par une modification du firewall.
CAM	Content Adresable Memory : table se trouvant dans les switchs pour permettre d'associer l'adresse MAC d'un système au port du switch (table dynamique).
DoS	Denial of Service: déni de service est un type d'attaque réseau servant à rendre indisponible une machine ou un service sur une machine.
FIFO	First In First Out : méthode de récupération des informations dans une queue. La première information introduite est la première à en être extraite.
HTML	Hyper Text Markup Language : langage à balises employé pour faire des pages Internet.
HTTP	HyperText Transfert Protocol : protocole utilisé par les navigateurs pour consulter les pages Internet situées sur des serveurs HTTP.
HTTPS	Protocole http sécurisé par le protocole SSL.
IOS	Internetwork Operating System : nom donné au software des différents éléments Cisco.
ICMP	Internet Control Message Protocol : protocole de contrôle de messages Internet qui est utilisé par les routeurs pour générer des erreurs et par le programme <i>ping</i> .
ISN	Initial Sequence Number : lors de l'ouverture d'une connection TCP, le client et le serveur doivent insérer leur ISN.
LAN	Local Area Network : réseau local de l'entreprise.
LIFO	Last In First Out : méthode de récupération des informations dans une pile. La dernière information introduite est la première à en être extraite.

MIB	Management Information Base : table contenant des informations réseau se trouvant dans les éléments réseau.
NAT	Network Address Translation: traduction d'adresses réseau effectuée généralement par les firewalls, traduction faite entre un réseau d'adresses IP privé et un réseau d'adresses IP public.
OS	Operating System: système d'exploitation tels que Windows, UNIX ou MAC
OSI	Modèle développé par l'ISO (International Organization of Standards) afin que soit défini un standard utilisé dans le développement de systèmes ouverts. Modèle à sept couches : 1 physique, 2 liaison, 3 réseau, 4 transport, 5 session, 6 présentation, 7 application.
PKI	Public Key Infrastructure : infrastructure de clé employant la cryptologie asymétrique (clé publique et clé privée) pour les authentications, signatures, etc.
RAM	Random Access Memory : mémoire vive de l'ordinateur qui stocke des données pour une courte durée, contrairement au disque dur ou autres systèmes de stockage. A chaque arrêt de l'ordinateur, les données sont perdues.
SMTP	Simple Mail Transfert Protocol : protocole utilisé pour le transfert de courrier sur Internet.
SNMP	Simple Network Management Protocol : protocole employé pour la gestion réseau, notamment pour consulter les MIBs.
SSL	Secure Sockets Layers : protocole servant à sécuriser les applications. Il se situe au-dessus de la couche TCP.
SSH	Secure Shell : correspond à un Telnet avec la couche SSL.
TCP	Transmission Control Protocol : protocole de transport (couche 4 du modèle OSI) orienté connection qui garantit la fiabilité.
UDP	User Datagram Protocol : protocole de transport (couche 4 du modèle OSI) ne garantissant pas la fiabilité à l'inverse de TCP.
VLAN	Virtual Local Area Network : réseau LAN virtuel permettant de séparer des groupes de stations liées physiquement au sein d'un même réseau.
VPN	Virtual Private Network : réseau privé virtuel construisant une connection cryptée transitant par un réseau public (Internet).
WEP	Wired Equivalent Privacy: protocole sécurisé employé dans les réseaux WLAN par la norme 802.11b.
WLAN	Wireless Local Area Network : réseau LAN sans fils.

12 ANNEXES

- Annexe 1** Script arp.vbs servant à lancer plusieurs fenêtres DOS avec le programme *winarp_sk*. Utilisé au chapitre 3.5.1 page 39.
- Annexe 2** Script Linux activation et désactivation, nécessaire lors de l'utilisation du programme *arpspoof* (chapitre 3.5.3 page 48).
- Annexe 3** Configuration Catalyst 1900, document permettant la configuration du switch Cisco Catalyst 1900.
- Annexe 4** Tutorial 'Intrusion' destiné aux étudiants.
- Annexe 5** Donnée du laboratoire 'Intrusion'.
- Annexe 6** Corrigé du laboratoire 'Intrusion'.
- Annexe 7** CD-Rom