Réseau et Sécurité
TP 5 - ARP spoofing

Avertissement préalable. Les outils présentés ici doivent être utilisés **uniquement dans un cadre autorisé**. Tout usage sur un réseau tiers sans autorisation explicite, est **formellement interdit**.

Article 323-1 du Code pénal : Le fait d'accéder ou de se maintenir, frauduleusement, dans tout ou partie d'un système de traitement automatisé de données est puni de **2 ans d'emprisonnement** et de **60 000 € d'amende**.

Si cette intrusion entraîne : la suppression ou modification de données, ou une altération du fonctionnement du système, la peine peut aller jusqu'à **3 ans d'emprisonnement** et **100 000 € d'amende**.

Exercise 1 Mise en place

Lors de ce TP, vous aurez besoin de trois machines pour réaliser l'attaque. Afin de simuler ces trois acteur (client, serveur et attaquant) vous allez utiliser des VM Kali Linux dans virtualbox. Récupérez une copie via la commande :

```
wget https://perso.isima.fr/~chaolivi/reseausecu/kali-vm.ova.tar.gz
```

importez la dans virtual box et clonez la afin d'en avoir 3 copies.

Créez trois VM dans VirtualBox sur un même réseau local virtuel à l'aide de VirtualBox. Les login et mot de passe : **kali**. Les claviers des VM sont en `qwerty` par défaut, vous pouvez les passer en `azerty` avec la commande `setxkbmap fr`.

1. Quel sont les adresses MAC et IP des trois machines ? Vérifiez que le client peut communiquer avec le serveur. Sinon créez un reseau connectant les machines de virtualbox et vérifiez à nouveau.

2. Lancer `Wireshark` sur la machine de l'attaquant et lancer une analyse sur le reseau connectant les 3 machines. Laissez tourner l'analyse.

3. Affichez la table `arp` sur la machine client et la machine serveur. Vérifiez que le client et le serveur se connaissent bien.

4. Démarez le serveur apache2 sur le serveur. Connectez vous à partir de la VM client au serveur apache2 du serveur.

5. Sur `Wireshark` recupérez l'ip de la machine client et serveur.

Exercise 2 ARP Spoofing

Le protocole ARP sert à traduire une adresse réseau IP en une adresse physique. Les requêtes ARP ne passent pas les routeurs, qui relaient des informations au niveau de la couche réseau mais par du trafic broadcast MAC.

L'ARP spoofing est une attaque sur les réseaux locaux où un attaquant envoie de fausses informations ARP pour associer son adresse MAC à l'IP d'une autre machine, souvent la passerelle. Cela lui permet d'intercepter, modifier ou bloquer le trafic réseau entre les victimes et la passerelle (*man-in-the-midle*).

Avec la commande `ettercap -T -i eth0 -M arp:remote /<ip client>/<ip serveur>/` ou bien `arpspoof -i eth0 -t <ip_client> <ip_serveur>` exécuté par l'attaquant, interceptez les communications entre le client et le serveur lorsque le premier se connecte au serveur du second. Pour les deux commandes :

1. Vérifiez l'influence sur la table `arp` de la cible et du serveur. Que constatez vous ?
2. Essayez de vous connecter au serveur sur la machine cliente. Que constatez-vous ?
3. Observer les éléments présent dans la tram réseau sur `Wireshark`.

   Un fois que vous avez analysé ces points pour les deux outils, expliquez les différences.

Exercise 3 Modification du trafic par MITM

Dans cette partie, vous allez montrer qu'un attaquant placé en *Man-in-the-Middle* peut non seulement intercepter, mais aussi modifier les communications entre un client et un serveur.

Laissez tourner l'attaque `arp` avec `arpspoofing`.

Fichiers fournis :

— `replace_html.py` : script mitmdump à modifier
`wget https://perso.isima.fr/~chaolivi/reseausecu/replay_html.py`
— `fake.html` : page HTML que vous souhaitez injecter
`wget https://perso.isima.fr/~chaolivi/reseausecu/fake.html`

1. Activez le routage IP sur votre machine adversaire : `sudo sysctl -w net.ipv4.ip_forward=1`
2. Configurez la redirection des flux HTTP et HTTPS vers mitmdump :

   ```
   # HTTP
   sudo iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j REDIRECT --to-port 8080
   # HTTPS
   sudo iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 443 -j REDIRECT --to-port 8080
   ```

3. Dans un premier temps, vous pouvez tenter d'enregistrer le flux du client se connectant au serveur :
   `sudo mitmdump -w session.flows --mode transparent --listen-port 8080`
4. Modifiez le script `replace_html.py` pour qu'il remplace les réponses HTML du serveur par le contenu de `fake.html`.
5. Lancez le script modifié avec mitmdump : `sudo mitmdump -s replace_html.py`

   Charger ensuite la page depuis le client pour vérifier que la page injectée s'affiche correctement (pensez a rafraichir la page avec `ctrl+F5`). Vous devez avoir une idication de succès du côté de l'adversaire.

Exercise 4 Mise en place de contre-mesures avec HTTPS

Cette étape consiste à montrer que l'utilisation de TLS (HTTPS) permet de protéger la communication contre une attaque MITM.

1. Configurez le serveur pour qu'il utilise HTTPS avec un certificat auto-signé (cf. TP2).
2. Relancez l'attaque MITM (`arpspoof` + `mitmproxy`) et observez les résultats.
3. Accédez à la ressource protégée depuis le client à l'aide de `curl` ou d'un navigateur.
4. Analysez le trafic avec Wireshark : que constatez-vous concernant la lisibilité des échanges ?
5. Expliquez pourquoi l'attaquant ne peut plus modifier le contenu sans générer un avertissement de certificat.

Question : En quoi l'utilisation de TLS garantit-elle la confidentialité et l'intégrité des données échangées ?