16/10/2017

Léo Guilpain

Compte Rendu TP1

ARP Poisoning

Table des matières

[Introduction : 2](#_Toc496682321)

[Question 1 : 2](#_Toc496682322)

[Question 2 & 3 & 4 : 2](#_Toc496682323)

[Question 5 : 3](#_Toc496682324)

[Question 6 : 4](#_Toc496682325)

[Question 7 : 5](#_Toc496682326)

[Question 8 : 5](#_Toc496682327)

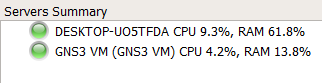
[Question 9 : 6](#_Toc496682328)

[Conclusion : 6](#_Toc496682329)

# Introduction :

Dans ce tp, nous allons tester différentes techniques de hack dans un réseau. Ces techniques vont nous aider à se protéger dans le futur.

# Question 1 :



Comme vous pouvez le voir GNS3 a bien été configuré avec une machine virtuelle.

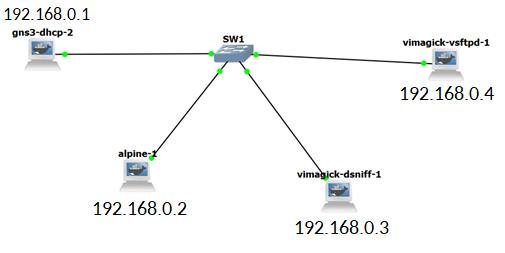
# Question 2 & 3 & 4 :

Une image contenant ciel

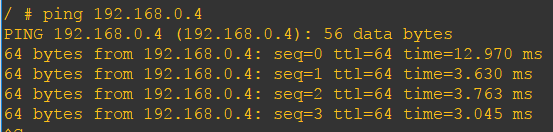
Description générée avec un niveau de confiance très élevé

Après avoir créé cette topologie et après l’avoir connectée, il a fallu placer toutes ces machines dans le même réseau afin qu’elles puissent communiquer.

Pour cela, il a fallu changer leur adresse IP : **« ifconfig eth0 @IP »**



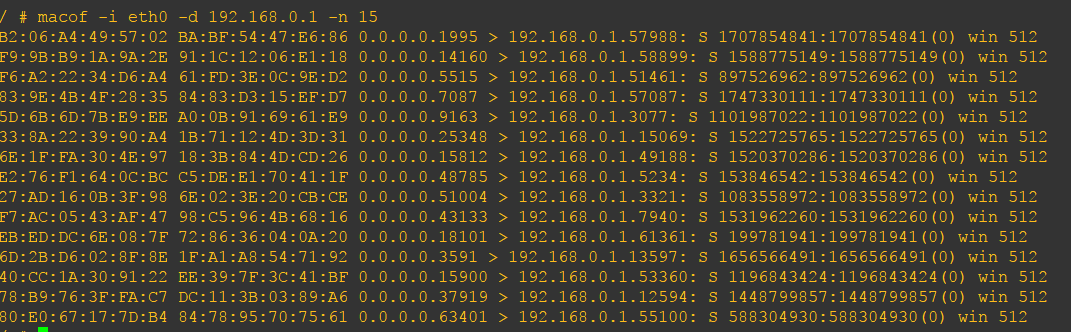
Pour vérifier que les machines ont bien été configurées dans le même réseau, on va tenter du pinger une machine à une autre.



Comme vous pouvez le voir, le ping de la machine Alpine-1 vers l’adresse IP 192.168.0.4 a bien fonctionné.

# Question 5 :

Le but est de faire en sorte que notre switch se comporte comme un hub. Pour cela, on va tenter de le faire « beuguer » en jouant sur la taille de la table. On la sature en envoyant des trames aléatoires avec des adresses mac différentes. Le switch n’arrivera plus à aiguiller les trames donc il va broadcast toutes les informations. A l’aide de la commande **« macof »,** le switch est inondé (floodé) d’adresses mac différences.

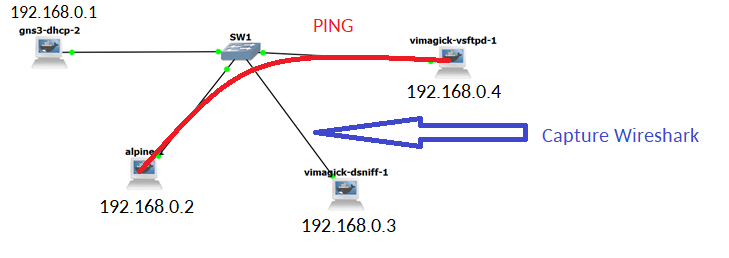


Ici, la machine Vimagick-dsniff-1 envoie 15 adresse mac différentes à l’adresse 192.168.0.1. Pour pouvoir flooder le switch, il faut en envoyer plus que 15. Les tests suivants ont été fait avec n = 100 000.

Une image contenant capture d’écran

Description générée avec un niveau de confiance très élevé

On peut voir sur la capture Wireshark que les adresses sources sont différentes alors qu’on envoie bien de la même machine.



La capture Wireshark a été faite entre dsniff et le switch. Pendant la capture wireshark, la machine alpine1 a pingé la machine vsftpd. On a obtenu ceci sur la capture Wireshark :



Lorsqu’une machine ping une autre machine, elle utilise le protocole ICMP. En passant par le switch, aucune autre machine ne devrait pouvoir voir ce protocole à part la machine destinataire.

Comme on peut le voir, la machine dsniff reçoit un protocole ICMP. Le switch se comporte donc comme un HUB et transmet les informations à toutes les machines qui lui sont reliées. Ainsi, on peut donc dire que le switch a été inondé.

# Question 6 :

La commande « **echo 1 > /proc/sys/net/ipv4/ip\_forward »** permet au pirate de rediriger les paquets ARP qui ne suit sont pas destinés.

# Question 7 :

Une image contenant capture d’écran

Description générée avec un niveau de confiance très élevé

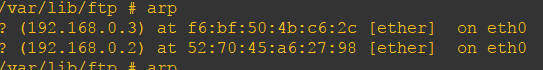
Avec cette capture on peut voir les caches arp des différentes machines. Certains cache arp sont vides, cela est dû au fait que certaines machines ne se sont jamais pingées et donc qu’elles ne se connaissent pas. On peut voir que dans ces caches, deux données sont stockées, l’adresse IP et l’adresse mac de la machine connue.

# Question 8 :

On effectue la commande **« arpspoof -t 192.168.0.4 192.168.0.2 »**.

Avec cette commande, on va faire en sorte de se faire passer pour la machine 192.168.0.2 auprès de la machine 192.168.0.4.

Avant d’effectuer cette commande, le cache arp de la machine 192.168.0.4 est :

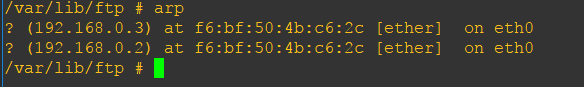


192.168.0.2 correspond à une adresse IP pingé précédemment.

192.168.0.3 correspond à notre adresse IP (du pirate).

Toutes les deux possèdent leur propre adresse MAC.

Après cette commande, on peut voir que le cache de la machine 192.168.0.4 a changé.

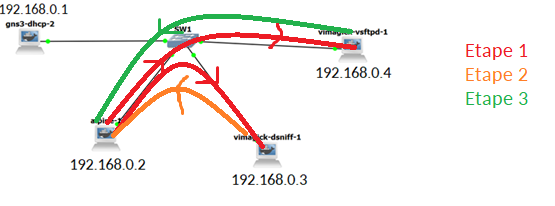


Les deux adresse IP sont toujours présentes. Cependant, l’adresse mac du pirate a changé. Il a copié celle de l’adresse IP 192.168.0.2.

À partir de maintenant, lorsque la machine avec l’adresse IP 192.168.0.4 voudra envoyer des données à la machine possédant l’adresse IP 192.168.0.2, les données seront envoyées à la machine pirate.   
C’est à partir de là que la commande echo rentre en jeu. En effet, si on n’avait pas l’echo, la machine 192.168.0.2 ne recevrait jamais les données et pourrait par la suite détecter un problème.  
Or l’echo permet au pirate de renvoyer les données reçues à la machine destinataire de base.

***(Suite à la perte du projet sur GNS3, j’ai recréé un projet. Cependant, lors de l’ARP poisoning, j’ai poisonné le cache de la machine 2 en faisant passer la machine pirate pour la machine 4, cela me donne donc des résultats différents dans la suite).***

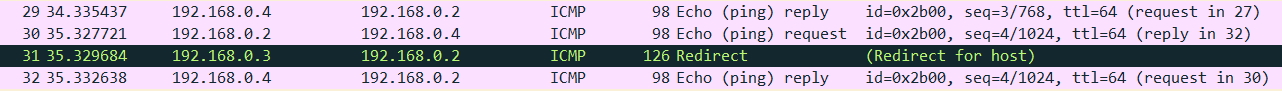
La machine 2 a voulu pinger la machine 4.



Etape 1 : Machine 2 envoie « request » vers machine 4. Comme la machine 2 pense que la machine 3 est la machine 4, elle envoie également à la machine 3.

Etape 2 : Machine 3 renvoie un echo afin que la machine 2 sache que cette machine est bien présente.

Etape 3 : Machine 4 renvoie « reply » à la machine 2.

Une image contenant capture d’écran

Description générée avec un niveau de confiance très élevé

# Question 9 :

Pour limiter l’accès aux données, il faut mettre du contrôle d’accès et du filtrage.

On peut également :

* Utiliser des tables ARP statiques (lourd à mettre en place)
* Obtenir l’adresse IP/mac en main propre
* Sécurisé en demandant une authentification avec des mots de passe

# Conclusion :

Comme on a pu le voir dans ce TP, on a réalisé deux attaques différentes : **MAC Flooding** et **ARP Poisoning**.

Le MAC flooding nous permis de flooder le switch afin de faire buguer son aiguillage. Son aiguillage n’étant plus opérationnel, ce dernier va se comporter comme un hub.

Ensuite, nous avons réalisé un ARP Poisoning. Le but ici a été de poisonner le cache ARP de la machine souhaitée afin de pouvoir écouter le flux de communication entre différentes machines.