ITÉRATION 5

Parefeu

Modalités

- Travail individuel en autonomie
- 3 jours en présentiel (dont 1 jour en autonomie)

Objectifs

L'objectif de ce module est d'avoir un aperçu des techniques de pare-feu modernes, à la fois pare-feu système et pare-feu réseau. Nous allons configurer les deux types en utilisant iptables/nftables (pour le pare-feu système) et pfSense (pour le pare-feu réseau). En plus de cela nous allons regarder les informations apportées par un système de détection d'intrusion.

Compétences

- Configurer le pare-feu du système
- Connaissances des pare-feu du système et des pare-feu du réseau

1.1 — Principaux concepts de réseau

2h — Présentiel

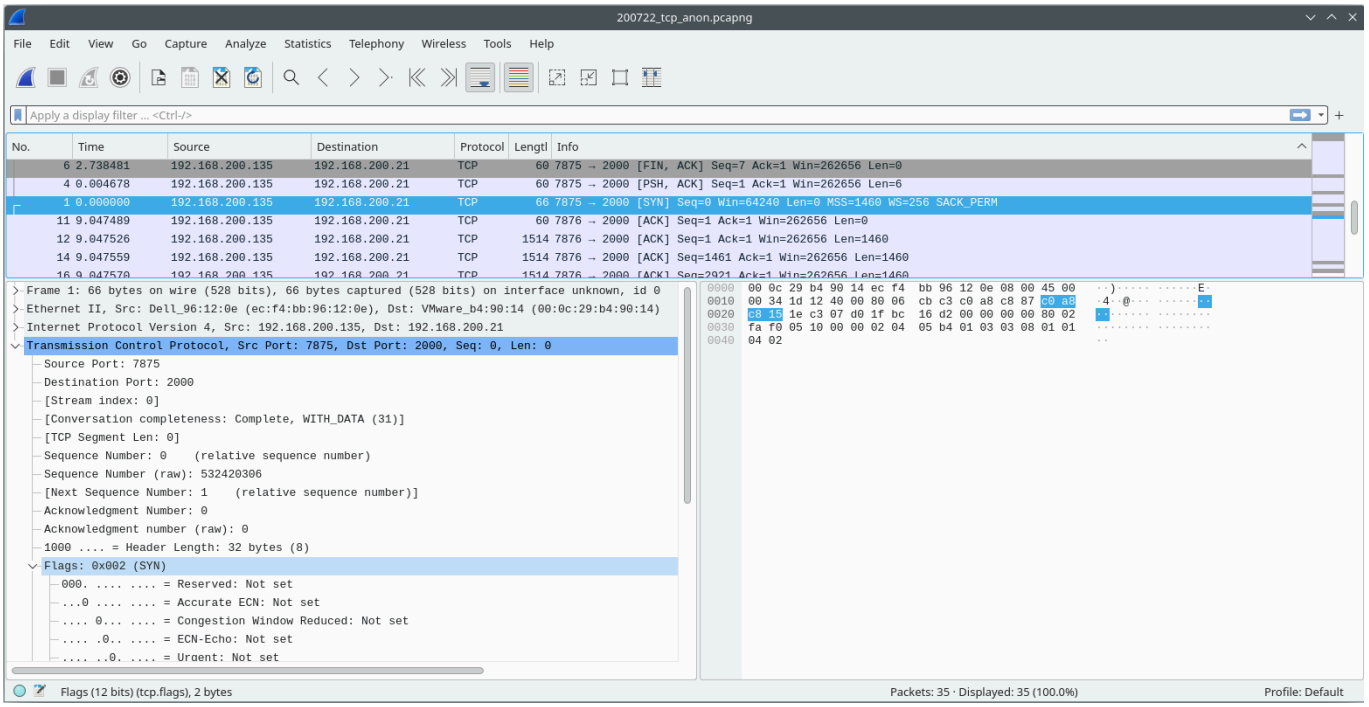
Revenons à la base des protocoles réseau. Tous les échanges de données consistent en plusieurs **paquets** qui sont échangés entre deux systèmes. Ces paquets contiennent des adresses IP (numériques) et, dans la plupart des protocoles, des **numéros de port**.

Les numéros de port permettent de distinguer plusieurs applications sur le même système. Le protocole TCP, utilisé pour SSH, les accès web et la plupart des applications en général, commence par un échange de paquets spécifiques (avec un champ spécifique contenant le bit SYN).

Il y a également des paquets dits "keepalive" garantissant que la connexion est toujours en cours. En fin de connexion, il y a aussi un échange spécifique de paquets (avec le flag FIN).

Les pare-feu interagissent avec les paquets à ce niveau bas. Il est une bonne pratique d'avoir une référence de protocole prête lorsque nous modifions les règles et qu'elles ne fonctionnent pas comme prévu.

Un outil utile pour comprendre ce qui se passe avec notre pare-feu est une **capture de paquets**. Vous pouvez capturer des paquets (dans un format PCAP) avec différents outils, comme `tcpdump` ou `wireshark`. Wireshark permet également une visualisation graphique.



Dans l'exemple de capture d'écran ci-dessus, nous pouvons voir la liste des paquets en haut, les détails d'un paquet spécifique en bas à gauche et le contenu binaire du paquet en bas à droite.

Pour commencer avec un pare-feu, installons wireshark et observons une connexion ssh. Pouvez-vous trouver les numéros de port ?

RESSOURCES

- Documentation de Wireshark https://www.wireshark.org/docs/wsug_html_chunked/
- Wikipedia sur TCP https://en.wikipedia.org/wiki/Transmission_Control_Protocol
- Exemple de traces de paquets pour plusieurs protocoles https://wiki.wireshark.org/SampleCaptures

COMPÉTENCES ASSOCIÉES

- Configurer le pare-feu du système
- Connaissances des pare-feu du système et des pare-feu du réseau

---

1.2 — iptables de base / nftables

4h — Présentiel

Le pare-feu est un logiciel permettant de contrôler quels paquets sont autorisés ou non. En pratique, cela permet de construire un système qui permet un accès externe à partir d'adresses données ou à des services donnés.

`Iptables` est l'outil standard sous Linux pour effectuer cette tâche. `iptables` est actuellement progressivement remplacé par `nftables`.

Nous allons utiliser `iptables`. Vous êtes plus susceptible de le rencontrer en pratique, pour l'instant. (l'alternative plus récente est nftables).

En consultant les tutoriels, créez des règles pour :

- désactiver tous les accès au port 22 (par ex. un serveur ssh), sauf à partir d'une adresse IP donnée
- désactiver tous les accès au port 80 (http) à partir d'un système extérieur, mais autoriser à partir de toutes les machines virtuelles et de votre système local
- activer et désactiver le ping de votre serveur/VM
- désactiver l'accès à tout service extérieur au port 80 (http, web non sécurisé), mais autoriser le port 443 (https, web sécurisé)
- tester quelle est la différence entre DROP et REJECT

Testez chacune de ces (ensemble de) règles séparément.

RESSOURCES

- Tutoriel iptables: https://docs.ovh.com/us/en/dedicated/firewall-iptables/
- Wiki debian avec script d'exemple: https://wiki.debian.org/iptables
- Wiki CentOS: https://wiki.centos.org/HowTos(2f)Network(2f)IPTables.html
- https://www.booleanworld.com/depth-guide-iptables-linux-firewall/
- (Optionnel) https://wiki.nftables.org/wiki-nftables/index.php/What_is_nftables%3F
- https://wiki.nftables.org/wiki-nftables/index.php/Main_Page

COMPÉTENCES ASSOCIÉES

- Configurer le pare-feu du système

1.3 — Implémentation de port knocking avec iptables/nftables 8h — Présentiel

"Port knocking" est une technique ressemblant à un jeu consistant à frapper une séquence secrète à la porte. Avec des paquets réseau à la place.

Ce n'est pas vraiment une technique de sécurité, mais elle peut aider à limiter les fausses connexions et les tentatives automatisées.

Cette fois, nous souhaitons configurer iptables afin qu'il autorise la connexion ssh (ou Web) au port 7843 uniquement en cas de séquence :

- une connexion au port 22
- dans les 30 secondes, une connexion au port 80
- dans les 15 secondes, une connexion au port 15022

RESSOURCES

- Un tutoriel: https://www.digitalocean.com/community/tutorials/how-to-configure-port-knocking-using-only-iptables-on-an-ubuntu-vps
- Un autre exemple: https://www.linuxtricks.fr/wiki/iptables-port-knocking-pour-ouvrir-ssh
- (Optionnel) Understanding State Machines https://www.freecodecamp.org/news/state-machines-basics-of-computer-science-d42855debc66/
- Implémentation avec nftables : https://wiki.nftables.org/wiki-nftables/index.php/Port_knocking_example

COMPÉTENCES ASSOCIÉES

- Configurer le pare-feu du système

1.4 — Parefeu réseau avec pfSense

8h — Présentiel

Le pare-feu avec lequel nous avons travaillé jusqu'à présent a modifié les règles d'une seule machine. C'est ce qu'on appelle un **pare-feu système**.

Dans les réseaux, nous utilisons également un autre type de pare-feu - un **pare-feu réseau**. Il protège l'ensemble du réseau, mettant en œuvre certaines règles spécifiques. Par exemple, il permet de désactiver l'accès à certains sites. Il peut également être utilisé pour créer une DMZ (zone démilitarisée, en anglais: demilitarized zone), un réseau spécial sans accès aux systèmes internes, mais permettant l'accès à Internet, pour les invités de l'entreprise.

Dans cet exercice, nous allons configurer un pare-feu réseau à l'aide de pfSense.

Attention : pfSense est basé sur BSD, pas sur Linux. Il est fortement recommandé de l'installer dans une VM.

Créez une DMZ pour une machine virtuelle dans pfSense. Les machines de la DMZ doivent avoir accès uniquement à Internet, pas entre d'autres machines virtuelles.

RESSOURCES

- Comment installer pfSense https://4sysops.com/archives/how-to-install-the-pfsense-firewall-on-a-virtual-machine/
- Configuration de firewall avec pfSense https://docs.netgate.com/pfsense/en/latest/firewall/index.html#
- https://docs.netgate.com/pfsense/en/latest/recipes/example-basic-configuration.html

COMPÉTENCES ASSOCIÉES

- Connaissances des pare-feu du système et des pare-feu du réseau

---

1.5 (Bonus) Détection d'intrusions avec Suricata

Les attaquants proposent régulièrement de nouvelles méthodes. Regarder le comportement du réseau dans son ensemble permet de voir si quelque chose d'étrange se passe ou non. Les systèmes de détection d'intrusion permettent cela - surveiller les séquences spécifiques. Ces séquences ne signifient pas nécessairement que quelque chose de mauvais se passe, mais nécessitent un peu d'attention pour être analysées. Ils peuvent donner des informations sur les nouvelles attaques, les systèmes mal déployés et plus encore.

La détection d'intrusion est généralement déployée dans les grands réseaux. Nous allons installer l'un de ces systèmes appelé Suricata avec les règles Emerging Threads et voir ce qui se passe dans le réseau local.

RESSOURCES

- https://docs.suricata.io/en/latest/what-is-suricata.html

COMPÉTENCES ASSOCIÉES

- Connaissances des pare-feu du système et des pare-feu du réseau

---

Livrables

Dans ce projet, nous devrions avoir :

➔ Une configuration iptables avec "port knocking"
➔ Une configuration fonctionnelle de pfSense