**TP Administration**

L’objectif de cette activité est de réaliser la mise en œuvre du pare-feu (firewall) UFW.

UFW (Uncomplicated FireWall) est un nouvel outil de configuration simplifié en ligne de commande de Netfilter, qui donne une alternative à l’outil iptables. Il existe aussi Gufw qui est une interface graphique pour ufw.

Documentation : <https://doc.ubuntu-fr.org/ufw>

Uncomplicated Firewall est pré-installé sous Ubuntu, mais si besoin il faudrait installer le paquet ufw.

Question 1. Vérifier l’état de UFW.

$ sudo ufw status verbose  
Status: inactive

UFW n’est pas activé par défaut, il faut donc l’activer avec la commande ufw :  
USAGE: ufw [--dry-run] enable|disable|reload

Question 2. Activer UFW. Le pare-feu sera actif et lancé automatiquement au démarrage du système.

Question 3. Vérifier maintenant l’état de UFW. Quelle est la politique de filtrage par défaut ?

$ systemctl status ufw  
ufw.service - Uncomplicated firewall  
Loaded: loaded (/lib/systemd/system/ufw.service; enabled; vendor preset: enabled)  
Active: active (exited) since Mon 2019-09-09 09:42:49 UTC; 2 days ago  
Docs: man:ufw(8)  
Main PID: 391 (code=exited, status=0/SUCCESS)  
Tasks: 0 (limit: 1109)  
CGroup: /system.slice/ufw.service  
...  
$ sudo ufw status verbose  
Status: active  
Logging: on (low)  
Default: deny (incoming), allow (outgoing), disabled (routed)  
New profiles: skip

**Les profils d’applications**

Différentes applications peuvent enregistrer leurs profils auprès de UFW lors de leur installation. Ces profils permettent à UFW de gérer ces applications par leur nom. Par exemple, OpenSSH a un profil enregistré auprès d’UFW. Il est possible de lister ces applications :

$ sudo ufw app list  
Available applications:  
OpenSSH

Et si vous avez installé le serveur HTTP Apache :

$ sudo ufw app list  
Available applications:  
Apache  
Apache Full  
Apache Secure  
OpenSSH

Trois profils sont disponibles pour Apache :

— Apache : n’ouvre que le port 80 (trafic Web normal et non chiffré)  
— Apache Full : ouvre le port 80 (trafic Web normal non crypté) et le port 443 (trafic crypté TLS/SSL)  
— Apache Secure : n’ouvre que le port 443 (trafic crypté TLS/SSL)

Question 4. Autoriser les connexions SSH. Vérifier l’état d’UFW. Tester. Il faut s’assurer que le pare-feu autorise les connexions SSH afin de pouvoir se reconnecter la prochaine fois :

$ sudo ufw allow OpenSSH  
Rule added  
Rule added (v6)  
$ sudo ufw status  
Status: active  
To Action From  
-- ------ ----  
OpenSSH ALLOW Anywhere  
OpenSSH (v6) ALLOW Anywhere (v6)

**Les règles par défaut**

Il est possible d’utiliser les règles par défaut :

USAGE: ufw default allow|deny|reject [incoming|outgoing]

De manière générale, on distingue deux types de politique de sécurité :

— politique permissive (open config) : Cette politique repose sur le principe que par défaut on laisse tout passer puis on va restreindre pas à pas les accès et les services mais la sécurité risque d’avoir des failles.

— politique stricte (close config) : Cette politique repose sur le principe inverse. On commence par tout interdire, puis on décide de laisser seulement passer les services ou adresses désirés ou indispensables. La sécurité sera meilleure mais le travail sera plus difficile et cela peut même bloquer plus longtemps que prévu les utilisateurs. C’est évidemment la politique conseillée pour un pare-feu.

Pour une politique « close config » :

$ sudo ufw default deny incoming  
La stratégie par défaut pour le sens « incoming » a été remplacée par « deny »  
(veillez à mettre à jour vos règles en conséquence)  
$ sudo ufw default deny outgoing  
La stratégie par défaut pour le sens « outgoing » a été remplacée par « deny »  
(veillez à mettre à jour vos règles en conséquence)  
$ sudo ufw status verbose  
État : actif  
Journalisation : on (low)  
Par défaut : deny (entrant), deny (sortant)

Pour une politique « open config » :

$ sudo ufw default allow incoming  
La stratégie par défaut pour le sens « incoming » a été remplacée par « allow »  
(veillez à mettre à jour vos règles en conséquence)  
$ sudo ufw default allow outgoing  
La stratégie par défaut pour le sens « outgoing » a été remplacée par « allow »  
(veillez à mettre à jour vos règles en conséquence)  
$ sudo ufw status verbose  
État : actif  
Journalisation : on (low)  
Par défaut : allow (entrant), allow (sortant)

**La journalisation**

Et pour gérer la journalisation :

// Activer la journalisation :  
$ sudo ufw logging on  
Journalisation activée  
// Désactiver la journalisation :  
$ sudo ufw logging off  
Journalisation désactivée

**Exemples**

L’ordre de déclaration des règles est très important, le système utilisant une politique « premier arrivé,premier servi ». Prenez donc soin d’ajouter vos règles spécifiques avant les règles générales lorsqu’elles concernent des éléments communs.

ufw [insert NUM] allow|deny|reject|limit [in|out] [log|log-all] PORT[/protocol]

Les exemples ci-dessous montrent l’utilisation de règles simples, par défaut les règles s’appliquent sur le trafic entrant (incoming) :

— Ouverture du port 22 en TCP et UDP :  
$ sudo ufw allow 22  
— Ouverture du port 22 en TCP uniquement :  
$ sudo ufw allow 22/tcp  
— Ouverture du service ssh (port 22 en TCP et UDP) :  
$ sudo ufw allow ssh  
— Autorise les requêtes DNS (port 53 en TCP et UDP) en sortie :  
$ sudo ufw allow out domain

**Visualisation des règles et de leurs numéros :**

$ sudo ufw status numbered  
État : actif  
Vers Action Depuis  
---- ------ ------  
[ 1] 22 ALLOW IN Anywhere  
[ 2] 53 ALLOW OUT Anywhere (out)  
[ 3] 22 ALLOW IN Anywhere (v6)  
[ 4] 53 ALLOW OUT Anywhere (v6) (out)

Suppression de la règle 1 :

$ sudo ufw delete 1  
Suppression de :  
allow 22  
Exécuter l’opération (o|n) ? o  
La règle a été supprimée

UFW regarde dans la liste de services connus (/etc/services) pour appliquer les règles standards associées à des services. Pour accéder au manuel : man ufw Et quelques règles plus complexes :

— Autoriser les accès du réseau local 10.0.0.0 :  
$ sudo ufw allow from 10.0.0.0/8

— Interdire les accès au port 5000 pour la machine 192.168.0.1  
$ sudo ufw deny proto udp from 192.168.0.1 to any port 5000  
— Ouverture du service ssh (port 22 en TCP et UDP) :  
$ sudo ufw allow ssh  
— Autoriser le trafic HTTP et HTTPS entrants sur l’interface enp0s3 :  
$ sudo ufw allow in on enp0s3 proto tcp from any to any port 80,443  
  
**Visualisation des règles et de leurs numéros :**

$ sudo ufw status numbered  
État : actif  
Vers Action Depuis  
---- ------ ------  
[ 1] Anywhere ALLOW IN 10.0.0.0/8  
[ 2] 5000/udp DENY IN 192.168.0.1  
[ 3] 80,443/tcp on enp0s3 ALLOW IN Anywhere  
[ 4] 80,443/tcp (v6) on enp0s3 ALLOW IN Anywhere (v6)

UFW applique des règles iptables par défaut lors de son lancement. Vous pouvez les consulter et les modifier directement dans le fichier /etc/ufw/before.rules. Par exemple, cela peut s’avérer utile si  
vous voulez interdire les requêtes de ping (ICMP Echo Request).

**Travail demandé**

Question 5. Mettre en place une politique close config par défaut. Afficher les règles de filtrage.

Question 6. Autoriser le service SSH. Afficher les règles de filtrage.

Question 7. Autoriser/Interdire seulement le trafic local du réseau 192.168.52.0/24. Tester. Afficher les règles de filtrage.

Question 8. Autoriser maintenant les connexions vers le port TCP 5001. Tester avec netcat. Afficher les règles de filtrage.

Question 9. Supprimer la règle autorisant les connexions vers le port TCP 5001. Tester avec netcat. Afficher les règles de filtrage.

Question 10. Interdire la possibilité de « pinger » votre serveur.