TP 02 : Services, processus signaux

Vous rédigerez un compte rendu, sur lequel vous indiquerez la réponse à chaque question ou points clefs, vos explications et commentaires (interprétation du résultat), et le cas échéant la ou les commandes utilisées.

Le compte rendu sera à rendre via votre depot Git à l'adresse suivante : lecocq@ipgp.fr avant la date et l'heure indiqué durant la scéance. N'oubliez pas de rediger un mail complet lors de votre envoie.

Utilisez le sujet suivant pour le mail : "[LP LPW 2023] compte rendu TP2 UNIX : Nom Prénom"

note : Tout fait partie de la note, le fond la forme, le respect des consignes, la citationn des sources ...

1 Secure Shell : SSH

1.1 Exercice : Connection ssh root (reprise fin tp-01)

Connectez vous en root via la console à votre machine et configurez `ssh`. Utilisez pour cela les commandes `apt search` et `apt install`. Changez la configuration de ssh pour permettre les connection root distante avec mot de pass.

- Expliquez à l'aide du `man sshd_config` l'élément de la configuration ssh que vous avez du changer les différentes options possibles, les avantages et leurs inconvénients. Dans quel cas faut t'il utiliser chacunes des options.

1.2 Exercice : Authentification par clef / Génération de clefs

Vous allez maintenant créer une clef d'authentification pour vous connecter directement à votre serveur Linux. Vous devez commencer par générer un couple de clef privé public sur votre session Linux sur la machine hote.

Dans le cadre du TP je vous invite à ne pas mettre de passphrase pour simplifier les choses, vous expliquerez pourquoi c'est une mauvaise idée dans un cas réel.

Vous sauvegarderez vos clef publiques et privées sur votre session Linux sur la machine hote.

Note : si vous utilisez votre machine personnelle expliquez les outils que vous avez utilisés.

1.3 Exercice : Authentification par clef / Connection serveur

Le but de cet exercice est de vous connecter à l'aide de votre clef publique sur votre serveur. Vous allez pour cela déposer votre clef publique sur le serveur.

Allez dans le dossier /root de l'administrateur de votre serveur et créer le dossier ".ssh" s'il n'existe pas déjà.
Le dossier ".ssh" est là où vont se situer les informations relatives aux comptes et aux connexions SSH de l'utilisateur. Nous y créerons le fichier "authorized_keys". Ce fichier contient toutes les clés publiques que permettent d'établir des connexions avec le serveur et l'utilisateur dans lequel il se trouve. Nous allons ensuite dans ce fichier mettre notre clé publique.

Il nous faudra ensuite donner les droits les plus restreints à ce fichier par sécurité et pour les exigences d'OpenSSH : Lecture/Ecriture/Execution pour le propriétaire seulement.

Notre clé est maintenant sur le serveur, il nous reste plus qu'à nous connecter pour tester !

1.4 Exercice : Authentification par clef : depuis la machine hote

note : utiliser la commande :`ssh -i maclef.pub root@ipserveur`

1.5 Exercice : Sécurisez

Sécurisez l'accé à votre machine via ssh pour root par clef seulement afin d'éviter les tentatitve d'authentification par brute force ssh. Qu'elle est la procédure ?
Expliquez en quoi consiste les attaques de type brute-force ssh.

Très souvent un serveur est rendu accésible à d'autres utilisateurs que root. Trouvez et expliquez d'autres techniques permettant à l'administrateur du serveur de ce protéger de ce type d'attaques. Note il existes plusieurs méthodes avec avantages et convénients (expliquez en fonction de vos choix).

2 Processus

2.1 Exercice : Etude des processus UNIX

1 - A l'aide de la commande ps, afficher la liste de tous les processus tournant sur votre machine, avec les informations suivantes :

USER	nom de l'utilisateur propriétaire du processus
PID	numéro d'identification
%CPU	
%MEM	
STAT	Etat
START	Date de début
TIME	
COMMAND	Commande utilisée pour lancer ce processus

Vous vous aiderez du manuel (man ps).

A quoi correspond l'information TIME ?
Quel est le processus ayant le plus utilisé le processeur sur votre machine ?
Quel a été le premier processus lancé après le démarrage du système ?
A quelle heure votre machine a-t-elle démarrée ? Trouvez une autre commande permetant de trouver le temps depuis lequel votre serveur tourne.
Pouvez-vous établir le nombre approximatif de processus créés depuis le démarrage ("boot") de votre machine ?

2 - Sous UNIX, chaque processus (excepté le premier) est créé par un autre processus, son processus père. Le processus père d'un processus est identifié par son PPID (Parent PID).

– Trouver une option de la commande ps permettant d'afficher le PPID d'un processus.
– Donner la liste ordonnée de tous les processus ancêtres de la commande ps en cours d'exécution.

3 - Reprendre la question précédente avec la commande pstree.

Vous devrez sans doute installer ce package : voir `apt update` ; `apt search` ; `apt install`.

4 - Essayez la commande `top`, qui affiche les mêmes informations que ps mais en raffraichissant périodiquement l'affichage.

- La touche ? permet d'afficher un résumé de l'aide de top. Afficher dans top la liste de processus triée par occupation mémoire ("resident memory") décroissante.
- Quel est le processus le plus gournabnd sur votre machine ? A quoi correspond-il ? (rappel : vous pouvez utiliser man truc pour découvrir ce que fait truc...).
- Trouvez les commandes interactives permettant de : passer l'affichage en couleur, mettre en avant le colonne de trie, changer la colonne de trie.
- Essayez la commande htop. Expliquez les avantages et/ou inconvénients à son utilisation par rapport à top.

3 Exercice 2 : Arrêt d'un processus

Ecrivez deux script shell contenant des boucles affichant la date.

fichier date.sh

```
#!/bin/sh
while true; do sleep 1; echo -n 'date '; date +%T; done
```

fichier date-toto.sh

```
#!/bin/sh
while true; do sleep 1; echo -n 'toto '; date --date '5 hour ago' +%T; done
```

- Lancer le 1er scripts. Le mettre en arrière plan (`CTRL-Z`).
- Lancer le 2eme scripts. Le mettre en arrière plan (`CTRL-Z`).
- A l'aide des commandes `jobs` `fg` et `CTRL-C`, arrêter les 2 horloges.
- Même question en utilisant les commandes `ps` et `kill` (avec un PID).
- Expliquer les scripts à l'aide du man.

4 Exercice 3 : les tubes

Quelle est la différence entre `tee` et `cat` ?

Que font les commandes suivantes :

```
$ ls | cat

$ ls -l | cat > liste

$ ls -l | tee liste

$ ls -l | tee liste | wc -l
```

5 Journal système rsyslog

- Le service rsyslog est-il lancé sur votre système ? Quel est le PID du démon ?

- Le principal fichier de configuration de rsyslog est `/etc/rsyslog.conf`. Dans quel fichier rsyslog écrit-il les messages issus des services standards ? Et la plupart des autres messages ? Vérifier le contenu de ces fichiers.

- A quoi sert le service cron ?

- Que fait la commande `tail -f` ? A l'aide de cette commande, placer en bas de votre écran un fenêtre qui permette de visualiser en "temps réel" le contenu du fichier `/var/log/messages`. Que voyez-vous si vous redémarrez le service cron depuis un autre shell ?

- Expliquer à quoi sert le fichier `/etc/logrotate.conf`.

- Examiner la sortie de la commande `dmesg`. Quel modèle de processeur linux détecte-il sur votre machine ? Quels modèles de cartes réseaux détecte-il ?