

Linux les bases

Vianney SELOSSE - Janvier 2023

Qui suis-je

- Ingénieur cybersécurité
- Manager équipe Redteam - Groupe Bancaire
- Spécialisé en hacking éthique (pentesting / tests d'intrusion)
- Passionné de cybersécurité



Sommaire

1. Introduction - Présentation de Linux
2. Installation de Linux et des logiciels
3. Shell et commandes système
4. Administration système
5. Administration réseau
6. Sécurité

Planning

1. Jour 1

- Présentation Linux + installation Distribution

2. Jour 2

- Présentation de commandes systèmes de base
- Fonctionnement OS Linux

3. Jour 3

- Commandes et fonctionnalités d'administration système

4. Jour 4

- Administration réseau et scripting bash

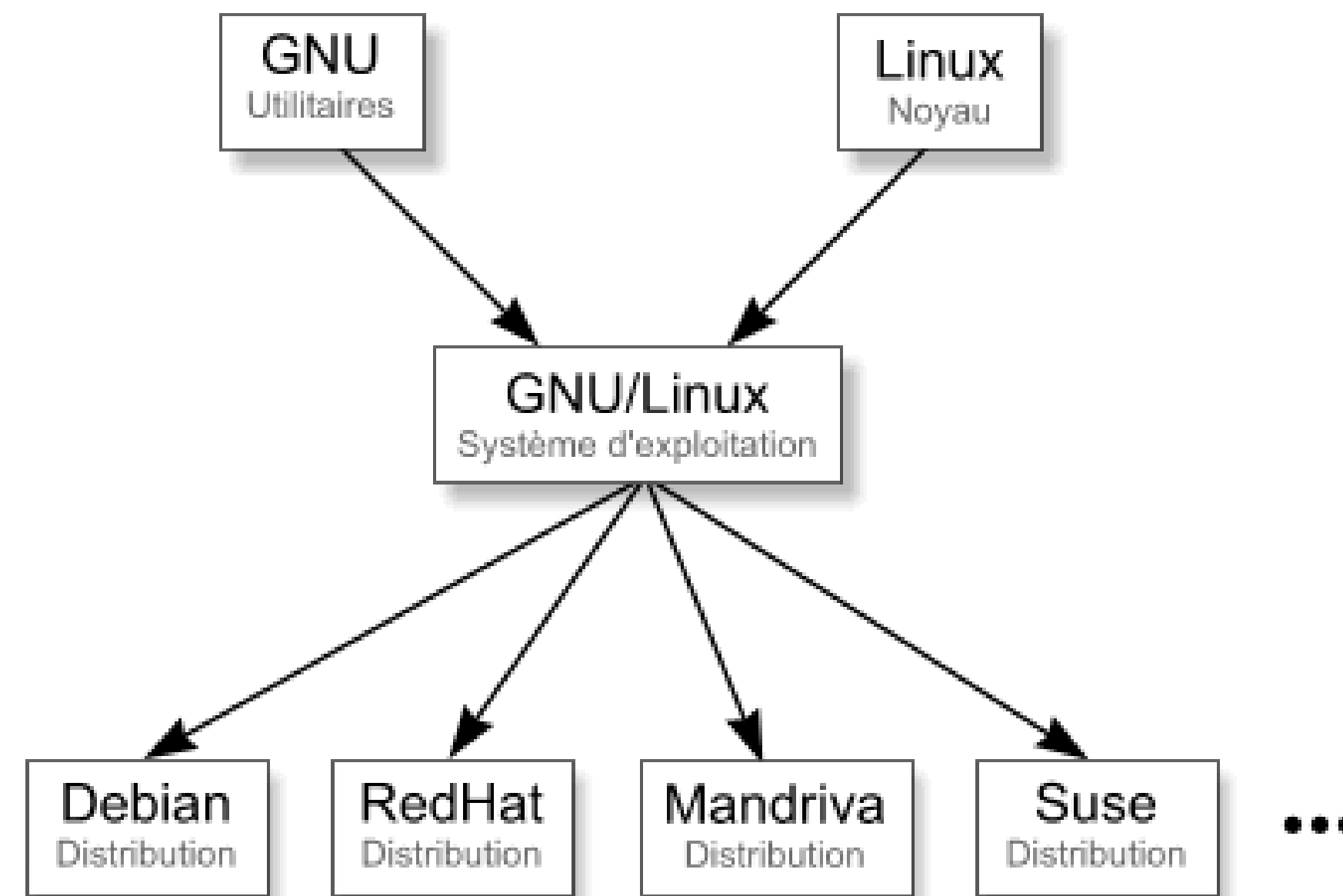
5. Jour 5

- Sécurité

1/ Introduction – Présentation de Linux

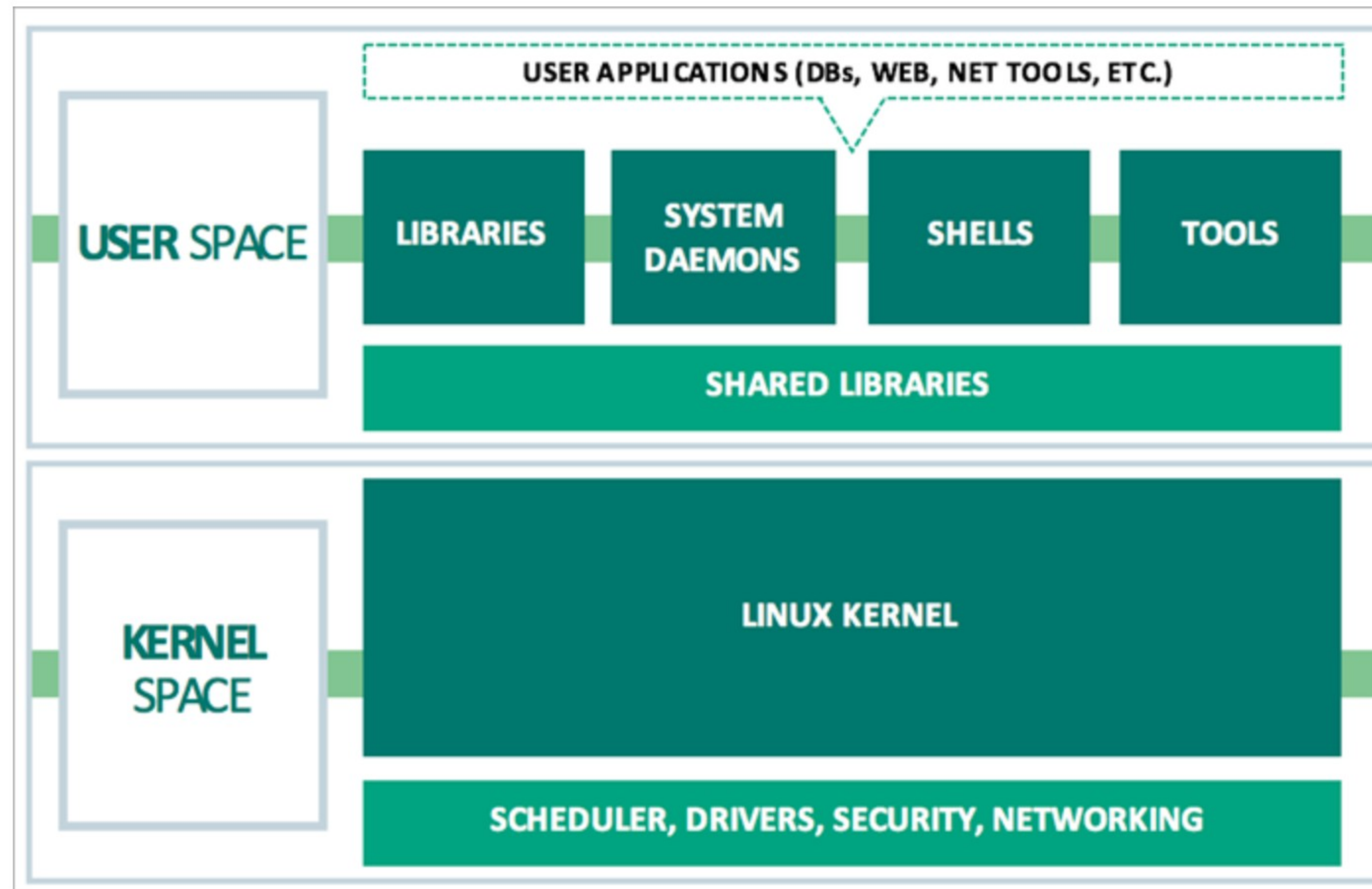
1/ Introduction - Présentation GNU/Linux

- Linux est le nom du noyau du système d'exploitation.
- GNU est un système d'exploitation (Comme Windows) utilisant le noyau Linux pour former le système GNU/Linux.



USERLAND vs KERNELLAND

Architecture OS basée sur Linux



Présentation GNU/Linux

Pour résumer :

- **Linux** est donc un **noyau** ;
- **GNU** est un ensemble de **programmes utilitaires** ;
- **GNU/Linux** est le système d'exploitation.

GNU/Linux est gratuit, des sociétés / communautés l'on repris et complété afin de customiser cet OS, ce qui a donné à plusieurs distributions.

Présentation GNU/Linux

Quelques distributions



Présentation GNU/Linux

OpenSource vs FreeWare

Qu'est ce que l'Open Source ?

- Définition Wikipédia :

« La désignation open source, ou logiciel libre, ou code source ouvert, s'applique aux logiciels (et s'étend maintenant aux œuvres de l'esprit) dont la licence respecte des critères précisément établis par l'Open Source Initiative, c'est-à-dire les possibilités de libre redistribution, d'accès au code source et de création de travaux dérivés. Mis à la disposition du grand public, ce code source est généralement le résultat d'une collaboration entre programmeurs. »

Différence entre Open Source et FreeWare ?

- C'est un logiciel gratuit, distribué par un auteur, sans avoir accès au code source.

Citer des outils OpenSource que vous connaissez.

Présentation GNU/Linux

Utilisation de Linux

Linux est utilisé dans quel domaine selon vous ?

- Desktop ;
- Mainframe ;
- Embarqué / IoT ;
- Mobile.

2/ Installation de Linux et des logiciels

2/ Installation de Linux et des logiciels

Distribution Debian

Nous allons voir deux méthodes d'installation d'OS via les deux distributions suivantes :

- Debian : <https://www.debian.org/index.fr.html>
- Kali : <https://kali.org>

La distribution Debian sera installée via son ISO netinstall (TP1), la distribution Kali directement depuis une image VM (TP1-Bis).

2/ Installation de Linux et des logiciels

Virtualisation

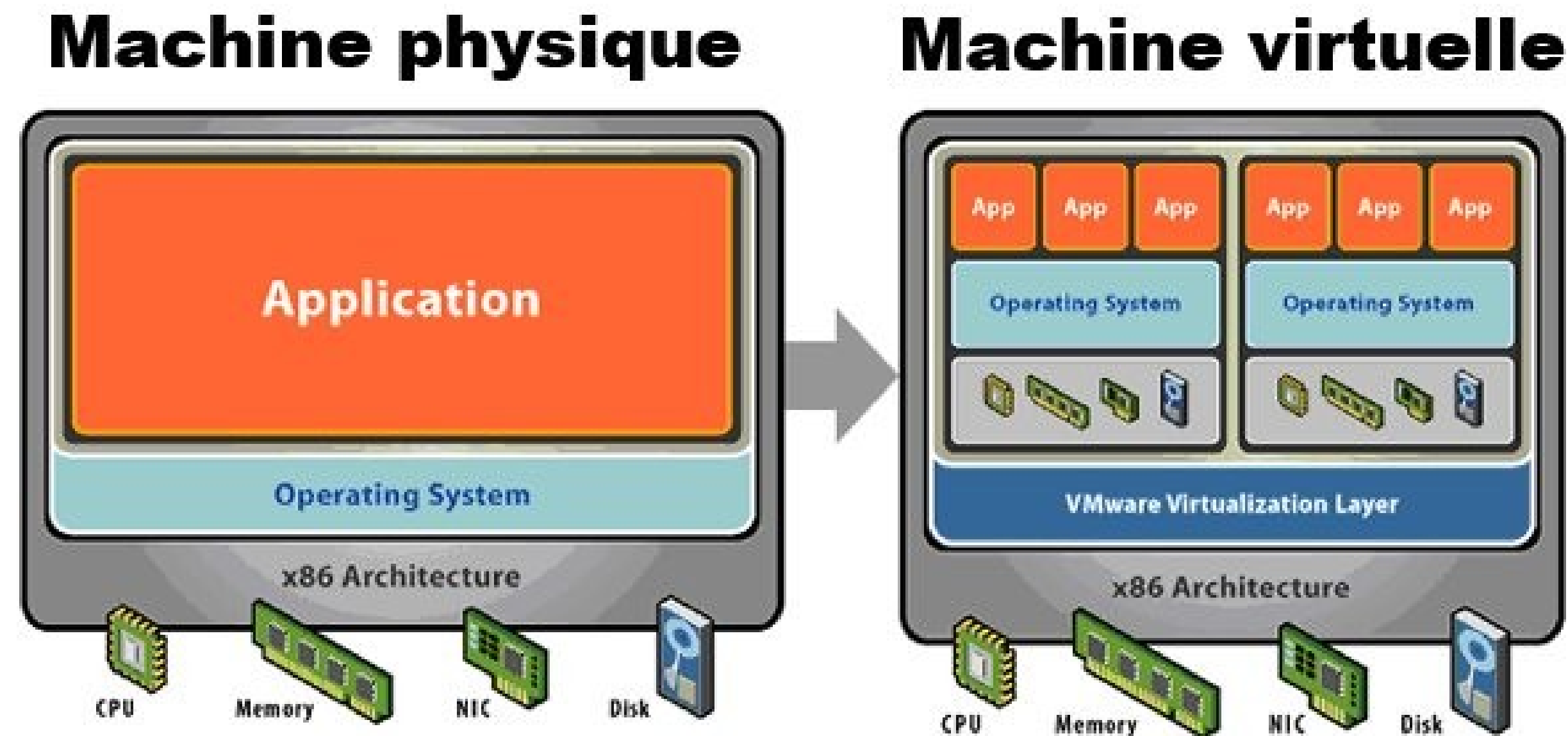
L'installation de notre distribution se fera via un environnement virtuel, pour plus de facilité.

Pour cela les concepts de virtualisation vont être détaillés.

2/ Installation de Linux et des logiciels

La virtualisation qu'est ce que c'est ?

C'est une technologie permettant de simuler un OS complet au sein d'un environnement hôte physique.



2/ Installation de Linux et des logiciels

Avantages de la virtualisation

La virtualisation permet de rapidement installer différents systèmes d'exploitation au sein d'une même machine physique.

Des systèmes de sauvegardes sont également implémentés permettant de restaurer rapidement le système en cas de crash.

La plupart des entreprises utilisent la virtualisation, mais également la conteneurisation (qui ne sera pas détaillé dans le cadre de cours).

2/ Installation de Linux et des logiciels

Outil de virtualisation

Nous utiliserons Virtualbox comme outil de virtualisation, afin d'installer nos distributions Linux.



Nous verrons son utilisation lors du prochain TP.

TP1 – Installation distribution Linux (ISO)

TP2 – Installation distribution Linux (OVA)

TP3 – Installation des Add-On invités

3/ Shell et commandes système

3/ Shell et commandes système

L'interpréteur de commande

Un **Shell** est l'interface entre l'utilisateur et le système d'exploitation. Dans le monde Linux est très utilisé, contrairement au monde Windows qui est plus basé sur le mode graphique (équivalent de PowerShell sous Windows).

Il existe différents types de shell, quelques exemples :

- **sh** (Bourne shell) ;
- **bash** (Bourne again shell) ;
- **zsh** (Zero shell).
- ...

3/ Shell et commandes système

Utilisation du shell

Le **Shell** fourni une fenêtre simpliste représentant un *prompt* appelé *invite de commande*.



C'est depuis cette interface que nous pouvons dialoguer avec le système.

Nous retrouvons le nom d'utilisateur connecté « @ » le nom machine

3/ Shell et commandes système

Commandes principales

Un TP dédié aux commandes Linux va permettre de mieux appréhender les concepts des commandes systèmes.

Les commandes principales pour utiliser un système sont :

- ls – List files
- cd – Change directory
- mkdir – Create directory
- cat – Show content of file

Une cheat sheet représentant les principales commandes est disponible au lien suivant :

- <https://cheatography.com/davechild/cheat-sheets/linux-command-line/>

3/ Shell et commandes système

Entrées – Sorties standards

Lors de l'exécution d'une commande, un processus est créé.

Il va alors ouvrir trois flux :

- **stdin** – Entrée standard (le clavier) ;
- **stdout** – Sortie standard (l'écran) ;
- **stderr** – Sortie d'erreur standard (écran).

3/ Shell et commandes système

Redirections

Il est possible de rediriger les flux d'entrée-sortie via des opérateurs dédiés :

- « **>** » – redirection de la sortie standard ;
- « **<** » – redirection de l'entrée standard ;
- « **>>** » – redirection de la sortie standard avec **concaténation** ;
- « **>&** » - redirection des sorties standard et d'erreur ;
- « **>!** » - redirection avec écrasement de fichier ;
- « **|** » - redirection de la sortie standard vers l'entrée standard (pipe).

3/ Shell et commandes système

Système de fichier – arborescence système

L'arborescence système est définie comme suit :

```
(kali㉿kali)-[~]  
$ tree / -L 1  
/  
├── 0  
├── bin -> usr/bin  
├── boot  
├── dev  
├── etc  
├── home  
├── initrd.img -> boot/initrd.img-6.0.0-kali5-amd64  
├── initrd.img.old -> boot/initrd.img-6.0.0-kali3-amd64  
├── lib -> usr/lib  
├── lib32 -> usr/lib32  
├── lib64 -> usr/lib64  
├── libx32 -> usr/libx32  
├── lost+found  
├── media  
├── mnt  
├── opt  
├── proc  
├── root  
├── run  
├── sbin -> usr/sbin  
├── srv  
├── swapfile  
├── sys  
├── tmp  
├── usr  
├── var  
├── vmlinuz -> boot/vmlinuz-6.0.0-kali5-amd64  
└── vmlinuz.old -> boot/vmlinuz-6.0.0-kali3-amd64  
  
22 directories, 6 files
```

3/ Shell et commandes système

Système de fichier – arborescence système

La racine du système est définie par un « / » (slash).

Description des sous-répertoires :

- /bin – exécutables essentiels au système ;
- /boot – contient les fichiers permettant à Linux de démarrer ;
- /dev – points d'entrée des périphériques (usb, etc) ;
- /etc – fichiers de configuration (réseau, utilitaires, groupes, ...) ;
- /home – le répertoire personnel des users ;
- /lib – les bibliothèques essentielles au système ;
- /mnt ; /media – contient les points de montage des partitions temporaires (cd-rom, usb, ..) ;
- /opt – packages d'applications supplémentaires ;
- /proc – fichiers contenant des informations sur la mémoire, etc ;
- /root – répertoire administrateur système ;
- /usr – hiérarchie secondaire (utilisateurs) ;
- /var – contient des données variables ;
- /tmp – fichiers temporaires (écrasés à chaque reboot du système).

3/ Shell et commandes système

Aides des commandes

Chaque commande dispose d'options. Heureusement il n'est pas nécessaire de connaître chaque commande par coeur.

Il est possible d'afficher l'aide de chaque commande par différents moyens :

- `man <commande>` ;
- `<commande> --help` ;
- `tldr <commande>`.

3/ Shell et commandes système

Redirections

Les redirections sont très utiles afin de récupérer l'output standard vers des fichiers par exemple.

Exemple d'une redirection de sortie standard vers un fichier via l'opérateur « > » :

```
(kali㉿kali)-[~]  
$ echo "redirection vers un fichier"  
redirection vers un fichier  
  
(kali㉿kali)-[~]  
$ touch test.fr  
  
(kali㉿kali)-[~]  
$ echo "redirection vers un fichier" > test.fr  
  
(kali㉿kali)-[~]  
$ cat test.fr  
redirection vers un fichier  
  
(kali㉿kali)-[~]  
$
```

3/ Shell et commandes système

Redirections

L'opérateur « >> » permet de rediriger la sortie standard de la même manière que l'opérateur simple « > », à la différence qu'il concatène, cela permet de ne pas supprimer l'ancien contenu.

Exemple :

```
(kali㉿kali)-[~]  
$ cat test.fr  
redirection vers un fichier  
  
(kali㉿kali)-[~]  
$ echo "Ajout texte à la fin du fichier" >> test.fr  
  
(kali㉿kali)-[~]  
$ cat test.fr  
redirection vers un fichier  
Ajout texte à la fin du fichier  
  
(kali㉿kali)-[~]  
$ echo "redirection simple" > test.fr  
  
(kali㉿kali)-[~]  
$ cat test.fr  
redirection simple
```

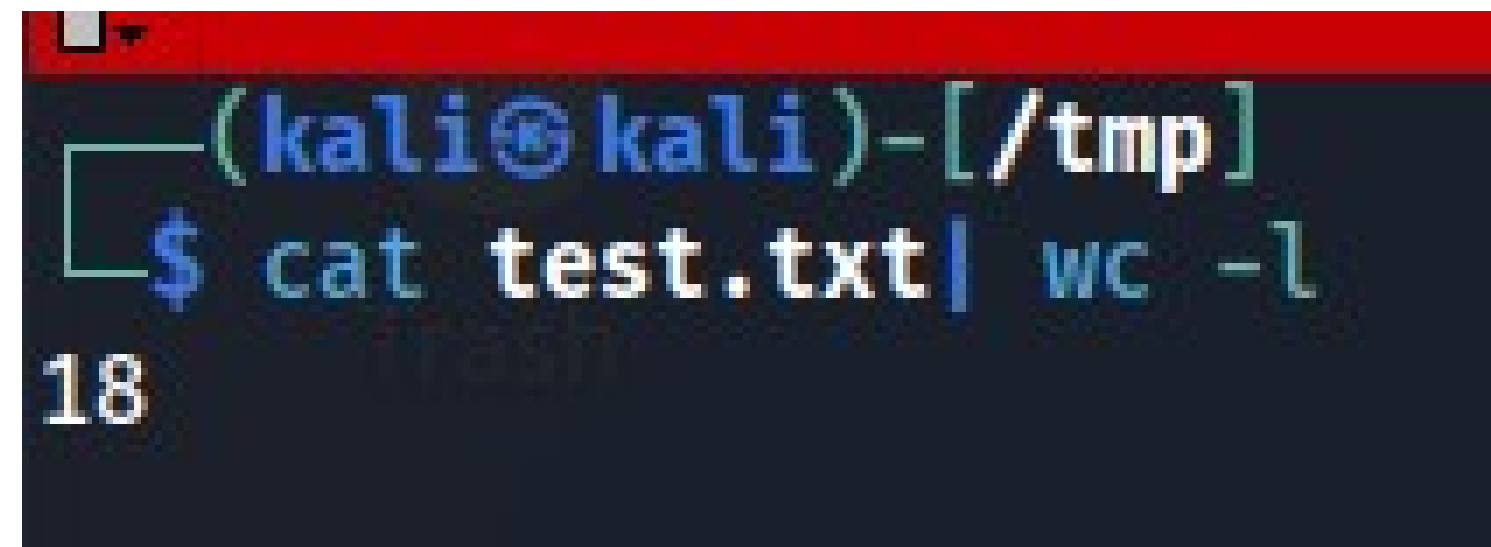
3/ Shell et commandes système

Enchaînements de commandes

Il est possible d'enchaîner plusieurs commandes de manière :

- *Séquentielle* : `command1 ; command2 ; command3`
- *Parallèle* : `command1 | command2 | command3`

Exemple :

A terminal window with a red title bar. The prompt is `(kali@kali)-[/tmp]`. The command `$ cat test.txt | wc -l` is entered. The output `18` is displayed on the line below.

```
(kali@kali)-[/tmp]
$ cat test.txt | wc -l
18
```


3/ Shell et commandes système

Fichiers cachés

Certains dossiers ou fichiers peuvent être cachés sur le système. Au même titre que sous windows, les fichiers cachés commencent par un « . ». Afin d'afficher les fichiers cachés, utiliser le paramètre « -a » de la commande « ls » :

Exemple :

```
(kali㉿kali)-[/tmp/fichierscachés]
$ touch fichierclassique.txt

(kali㉿kali)-[/tmp/fichierscachés]
$ touch .fichiercache.sh

(kali㉿kali)-[/tmp/fichierscachés]
$ mkdir .dossiercache

(kali㉿kali)-[/tmp/fichierscachés]
$ ls
fichierclassique.txt

(kali㉿kali)-[/tmp/fichierscachés]
$ ls -la
total 12
drwxr-xr-x  3 kali kali 4096 Dec 23 04:59 .
drwxrwxrwt 16 root root 4096 Dec 23 04:59 ..
drwxr-xr-x  2 kali kali 4096 Dec 23 04:59 .dossiercache
-rw-r--r--  1 kali kali   0 Dec 23 04:59 .fichiercache.sh
-rw-r--r--  1 kali kali   0 Dec 23 04:59 fichierclassique.txt

(kali㉿kali)-[/tmp/fichierscachés]
$
```

TP4 – Commandes Shell

4/ Administration système

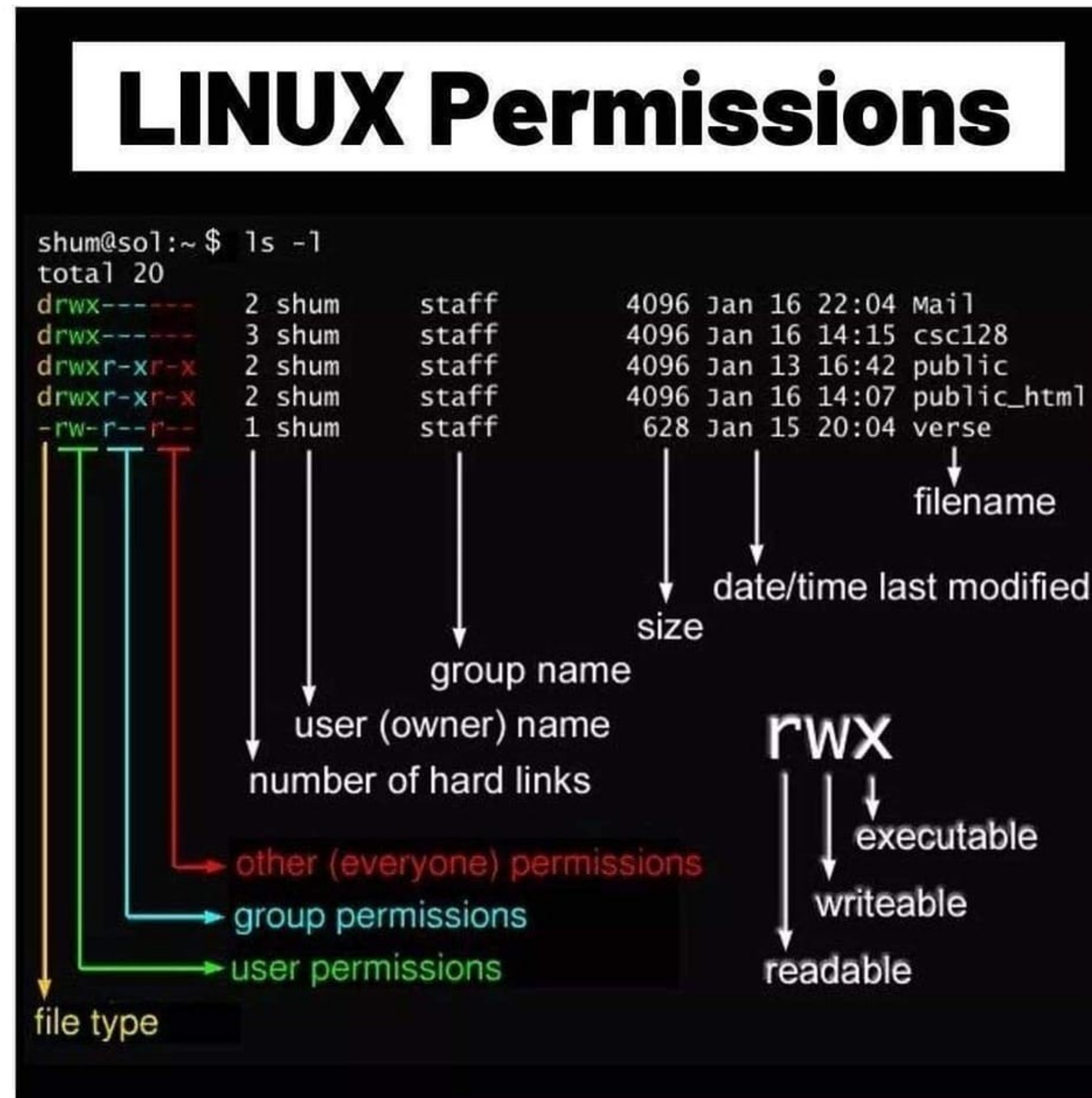
Rappel cours précédent

Questions

- Que permet de faire la commande netstat ?
- Que permet de faire la commande ps ?
- Affichez le fichier /etc/apt/sources.list et récupérer uniquement la ligne débutant par « *deb-src* », en enlevant le caractère « # »
- A quoi sert le répertoire /etc/ ?
- Comment recherche t-on dans l'historique terminal ?
- Avantage de la virtualisation comparé aux appliances physiques ?
- Quel outil permet de changer le propriétaire d'un fichier/dossier ?
- Quel sont les trois types d'entrées – sorties standard ?
- Raccourcis clavier pour effacer ma ligne en cours du terminal ?
- Quelle est l'utilité d'un alias ?

4/ Administration système

Permissions



4/ Administration système

Permissions

Les commandes principales pour gérer les permissions sont :

- `chmod` ;
- `chown`.

Ces deux commandes permettent respectivement de changer les droits des fichiers et dossiers ainsi que les propriétaires (owner).

4/ Administration système

Timestamp

L'horodatage (en anglais timestamping) est un mécanisme qui consiste à associer une date et une heure à un événement, une information ou une donnée informatique. Il a généralement pour but d'enregistrer l'instant auquel une opération a été effectuée.

La valeur représentant la date et l'heure est appelée timestamp (de l'anglais time, « heure » et stamp, marquage par un timbre ou un tampon) ou tout simplement « horodatage ».

4/ Administration système

Mise à jour système

Afin de garder son système à jour, deux commandes sont nécessaires.

- *sudo apt update* (mise à jour des dépôts de la distribution) ;
- *sudo apt upgrade* (mise à jour du système).

4/ Administration système

Installer un utilitaire

Pour installer un outil, utiliser apt :

- *sudo apt install <utilitaire>*

Recherche d'un utilitaire dans les sources :

- *sudo apt search <utilitaire>*

Supprimer utilitaire :

- *sudo apt remove <utilitaire>*

4/ Administration système

Installer un outil hors packages

Un grand nombre d'outils sont disponibles sur github, pour installer un outil git :

- *git clone https://github.com/<pathToGitUrl>.git*

Installation d'un .deb file :

- *sudo dpkg -i path/to/file.deb*

Lister les paquets installés :

- *sudo dpkg -l pattern*

4/ Administration système

Environnement virtuel

Il est possible de créer des environnements virtuels au sein de notre distribution Linux. Cela permet notamment d'éviter des problèmes de dépendances au sein des librairies.

Il est par exemple fort utilisé lorsque nous utilisons python, car différentes versions existent et certains scripts que l'on utilise peuvent tourner à la fois sur python2.7 par exemple et d'autres sous python3.

Cela peut engendrer des problèmes de dépendances.

4/ Administration système

Environnement virtuel

Pour créer un nouvel environnement virtuel, on utilisera « virtualenv ».

```
(kali㉿kali)-[/tmp]
$ virtualenv virtualenv
created virtual environment CPython3.10.8.final.0-64 in 560ms
creator CPython3Posix(dest=/tmp/virtualenv, clear=False, no_vcs_ignore=False, global=False)
seeder FromAppData(download=False, pip=bundle, setuptools=bundle, wheel=bundle, via=copy, app_data_dir=/home/kali/.local/share/virtualenv)
added seed packages: pip==22.3, setuptools==65.5.0, wheel==0.38.0
activators BashActivator,CShellActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator

(kali㉿kali)-[/tmp]
$ source virtualenv/bin/activate

(virtualenv)-(kali㉿kali)-[/tmp]
$
```

On peut visualiser l'apparition de (virtualenv) devant notre prompt. Cela indique qu'on est dans notre environnement virtuel cloisonné. Pour sortir de notre environnement :

```
(virtualenv)-(kali㉿kali)-[/tmp]
$ deactivate

(kali㉿kali)-[/tmp]
$
```

4/ Administration système

Environnement virtuel

La recherche de fichiers et contenu peut se faire sous Linux via deux commandes principales (parmi tant d'autres) :

- find (Ex : *find / -name "*.txt"*);
- grep (Ex : *grep -r "admin" /tmp*).

```
(kali@kali)-[/tmp]
$ sudo find / -name "monfichierArechercher.out"
find: '/run/user/1000/gvfs': Permission denied
/tmp/monfichierArechercher.out

(kali@kali)-[/tmp]
$
```

```
(kali@kali)-[/tmp]
$ grep -r bonjour
grep: systemd-private-7014dd4ddf20421eb0fed0d86a78aadd-s
grep: systemd-private-7014dd4ddf20421eb0fed0d86a78aadd-u
grep: systemd-private-7014dd4ddf20421eb0fed0d86a78aadd-c
grep: vmware-root_478-860529095: Permission denied
monfichierArechercher.out:bonjour
grep: systemd-private-7014dd4ddf20421eb0fed0d86a78aadd-s
grep: systemd-private-7014dd4ddf20421eb0fed0d86a78aadd-h
grep: systemd-private-7014dd4ddf20421eb0fed0d86a78aadd-M

(kali@kali)-[/tmp]
$
```

4/ Administration système

Persistance de session

Il est intéressant de garder une persistance de session lors de nos travaux.

Cela permet en cas de crash de pouvoir restaurer la session. C'est très utile lors de connexion distantes par exemple.

```
(kali@kali)-[/tmp]
$ screen -S sessionScreen
[detached from 6734.sessionScreen]

(kali@kali)-[/tmp]
$ screen -ls
There is a screen on:
  6734.sessionScreen      (12/23/2022 05:10:41 AM)      (Detached)
1 Socket in /run/screen/S-kali.

(kali@kali)-[/tmp]
$
```

Pour se rattacher à une session screen créée, utiliser la commande suivante :

screen -r <nomSession>

4/ Administration système

Alias

Il est possible de définir des alias. Les alias sont des substitutions de commandes répétitives et/ou longues à taper dans la console.

Il est possible de définir nos alias dans le fichier **.bashrc** du HOME.

```
(kali㉿kali)-[/tmp]
└─$ ls
script.sh                                systemd-private-1cd607812014498997933a9bc3b72863-Modem
ssh-XXXXXXhyp1SW                        systemd-private-1cd607812014498997933a9bc3b72863-syste
systemd-private-1cd607812014498997933a9bc3b72863-colord.service-0l9hCV  systemd-private-1cd607812014498997933a9bc3b72863-syste
systemd-private-1cd607812014498997933a9bc3b72863-haveged.service-YoIheD  systemd-private-1cd607812014498997933a9bc3b72863-upowe

(kali㉿kali)-[/tmp]
└─$ alias ls='ls -lh --color=tty'

(kali㉿kali)-[/tmp]
└─$ ls
total 44K
-rwxr-xr-x 1 kali kali 201 Jan 2 04:13 script.sh
drwx----- 2 kali kali 4.0K Jan 2 04:08 ssh-XXXXXXhyp1SW
drwx----- 3 root root 4.0K Jan 2 04:08 systemd-private-1cd607812014498997933a9bc3b72863-colord.service-0l9hCV
drwx----- 3 root root 4.0K Jan 2 04:08 systemd-private-1cd607812014498997933a9bc3b72863-haveged.service-YoIheD
drwx----- 3 root root 4.0K Jan 2 04:08 systemd-private-1cd607812014498997933a9bc3b72863-ModemManager.service-qPsDoi
drwx----- 3 root root 4.0K Jan 2 04:08 systemd-private-1cd607812014498997933a9bc3b72863-systemd-logind.service-7q3SQu
drwx----- 3 root root 4.0K Jan 2 04:08 systemd-private-1cd607812014498997933a9bc3b72863-systemd-timesyncd.service-Jyet4x
drwx----- 3 root root 4.0K Jan 2 04:08 systemd-private-1cd607812014498997933a9bc3b72863-upower.service-fBc86x
-rw-r--r-- 1 kali kali 78 Jan 2 04:08 test.txt
drwxrwxrwt 2 root root 4.0K Jan 2 04:08 VMwareDnD
drwx----- 2 root root 4.0K Jan 2 04:08 vmware-root_520-2957059313

(kali㉿kali)-[/tmp]
└─$
```

4/ Administration système

Crontab

Crontab permet d'exécuter des actions de manière automatique et programmée par le système.

Il est très utilisé dans le domaine de l'administration système.

- Pour éditer un fichier crontab : *crontab -e*
- Pour afficher la configuration crontab : *crontab -l*

```
# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed
17 * * * * root cd / && run-parts --report /etc/cron.hourly
25 6 * * * root test -x /usr/sbin/anacron || { cd / && run-parts --report /etc/cron.daily; }
47 6 * * 7 root test -x /usr/sbin/anacron || { cd / && run-parts --report /etc/cron.weekly; }
52 6 1 * * root test -x /usr/sbin/anacron || { cd / && run-parts --report /etc/cron.monthly; }
#
```


4/ Administration système

Crontab

Quelques exemples :

- Exécution chaque 1^{er} et 15 de chaque mois à minuit :
 - `0 0 1,15 * * <commande>`
- Reboot d'une machine chaque 1^{er} et 15 du mois à 2h30 du matin :
 - `30 2 1,15 * * /sbin/shutdown -r`
- Appeler un script de sauvegarde tous les mardis à 3h du matin :
 - `0 3 * * 2 /usr/bin/backup`

4/ Administration système

Editeur de texte - VIM

L'édition des fichiers sous Linux peut se faire via deux éditeurs principaux :

- vim ;
- nano.

Nous allons utiliser vim, qui est beaucoup plus puissant et intéressant en termes de fonctionnalités.

```
VIM - Vi IMproved

        version 9.0.813
        by Bram Moolenaar et al.
    Modified by team+vim@tracker.debian.org
    Vim is open source and freely distributable


    Help poor children in Uganda!
type  :help iccf<Enter>      for information

type  :q<Enter>              to exit
type  :help<Enter> or <F1>   for on-line help
type  :help version9<Enter> for version info
```

TP5 – VIM

4/ Administration système

Variables d'environnement

Les variables d'environnement permettent le contrôle du fonctionnement du shell et d'autres programmes Linux.

La commande ENV permet de les lister :

```
(kali@kali)-[/tmp]
$ env
TERMINATOR_DBUS_NAME=net.tenshu.Terminator21a9d5db22c73a993ff0b42f64b396873
SSH_AUTH_SOCK=/tmp/ssh-XXXXXXhyp1SW/agent.1042
SESSION_MANAGER=local/kali:~/tmp/.ICE-unix/1042,unix/kali:/tmp/.ICE-unix/1042
SSH_AGENT_PID=1113
LANG=en_US.UTF-8
XDG_CURRENT_DESKTOP=XFCE
POWERSHELL_UPDATECHECK=Off
TERMINATOR_DBUS_PATH=/net/tenshu/Terminator2
XDG_GREETER_DATA_DIR=/var/lib/lightdm/data/kali
TERMINATOR_UUID=urn:uuid:15ff4109-983a-45e7-8119-ae5a75c11bf
USER=kali
DESKTOP_SESSION=lightdm-xsession
XDG_MENU_PREFIX=xfce-
XDG_SESSION_PATH=/org/freedesktop/DisplayManager/Session0
HOME=/home/kali
COMMAND_NOT_FOUND_INSTALL_PROMPT=1
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
XDG_VTNR=7
XDG_SEAT=seat0
GTK_MODULES=gail:atk-bridge
XDG_DATA_DIRS=/usr/share/xfce4:/usr/local/share:/usr/share:/usr/share
XDG_CONFIG_DIRS=/etc/xdg
XDG_SESSION_DESKTOP=lightdm-xsession
QT_ACCESSIBILITY=1
XDG_VTNR=7
```

```
(kali@kali)-[/tmp]
$ echo $USER
kali

(kali@kali)-[/tmp]
$ echo $HOME
/home/kali
```

4/ Administration système

Scripting Bash

Il est possible de réaliser des scripts afin d'automatiser les tâches récurrentes.

Exemple :

```
1 #!/bin/bash
2
3 addition(){
4     sum=$(( $1+$2 ))
5     return $sum
6 }
7 read -p "Entrez un premier numéro : " int1
8 read -p "Entrez un deuxième numéro : " int2
9 addition $int1 $int2
10 echo "Le résultat est : " $?
```

4/ Administration système

Services

Les différents services sous Linux peuvent être relancés via deux moyen :

- *service <service> start/restart/stop*
- */etc/init.d/<service> start/restart/stop*
- *systemctl start/restart/stop <service>*

Exemple :

- *service ssh restart*
- */etc/init.d/cron restart*
- *systemctl restart networking.service*

TP6 – Administration système

5/ Administration réseau

5/ Administration réseau

Configurations réseau virtualisation

La virtualisation a été abordée en début de cours. Afin de mieux comprendre les concepts de la suite de ce cours, un détail va être fait sur les modes réseau virtuels.

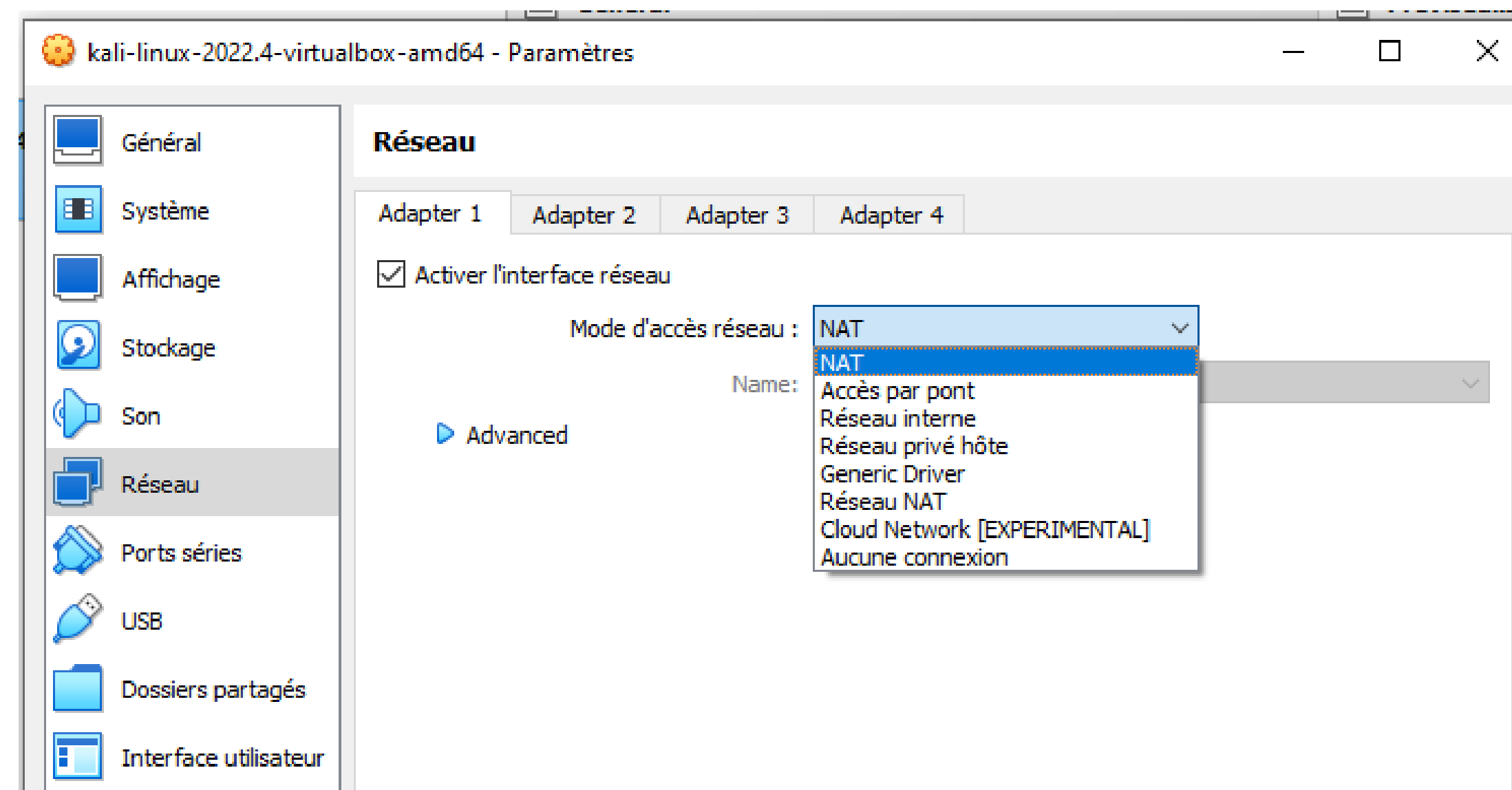
- Différentes possibilités de configuration réseau sont possibles.
 - NAT ;
 - Bridge ;
 - Réseau interne ;
 - Réseau privé.

5/ Administration réseau

Les types de connexion

Il existe différents modes réseau sous Virtualbox. Chaque mode possède ces avantages et inconvénients.

Cela se passe dans les paramètres de la VM, onglet Réseau :



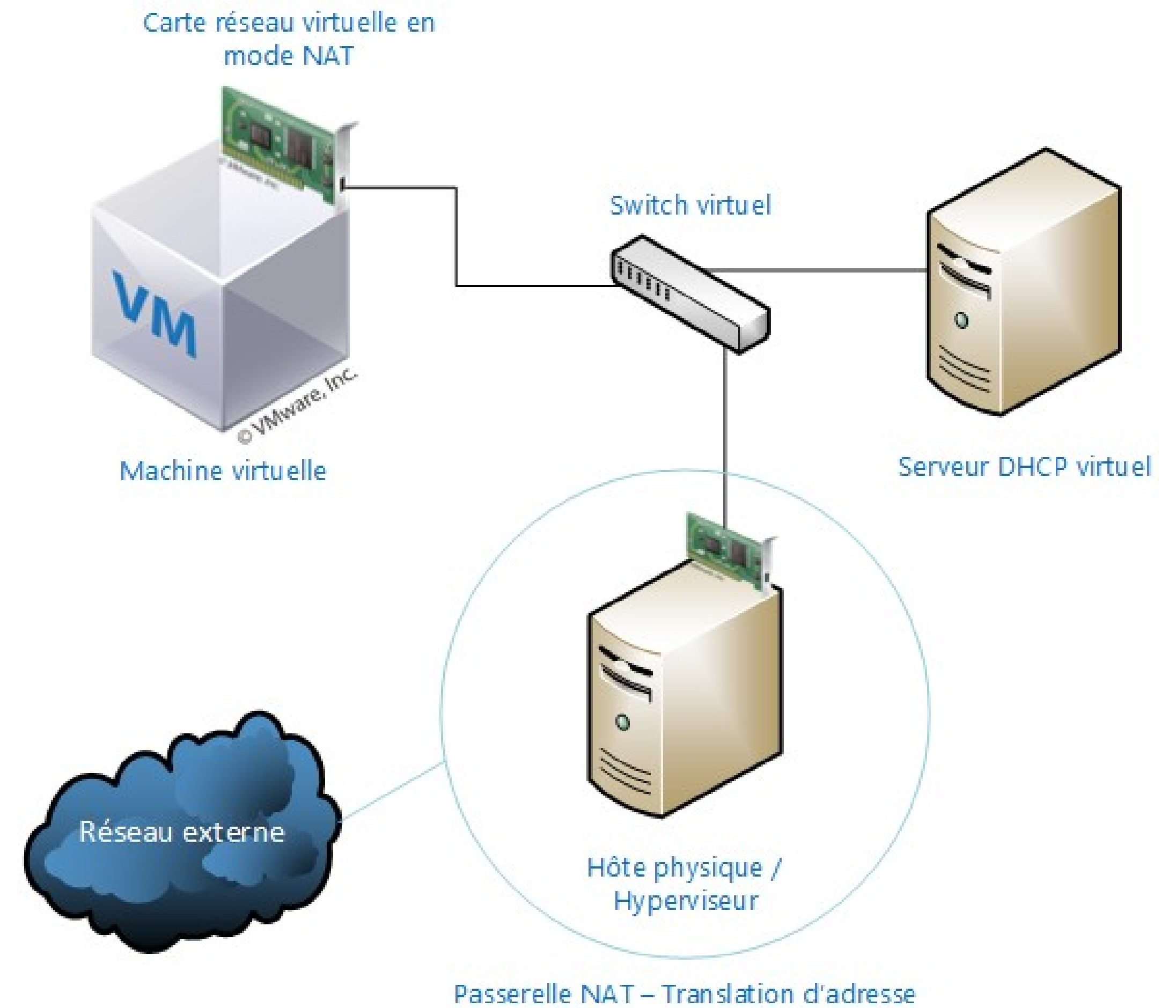
Nous allons détailler les principaux.

5/ Administration réseau

Les types de connexion - NAT

Un réseau **NAT (Network Address Translation)** permet d'utiliser la connexion de l'hôte. D'un point de vue réseau, la machine est vue comme l'hôte, c'est à dire qu'elle utilise l'adresse IP de la machine hôte pour agir.

Ce mode est intéressant puisqu'il permet à notre machine virtuelle d'accéder à notre réseau de façon totalement transparente puisque c'est l'adresse IP de la machine physique qui est utilisée grâce à la translation d'adresse du processus NAT.



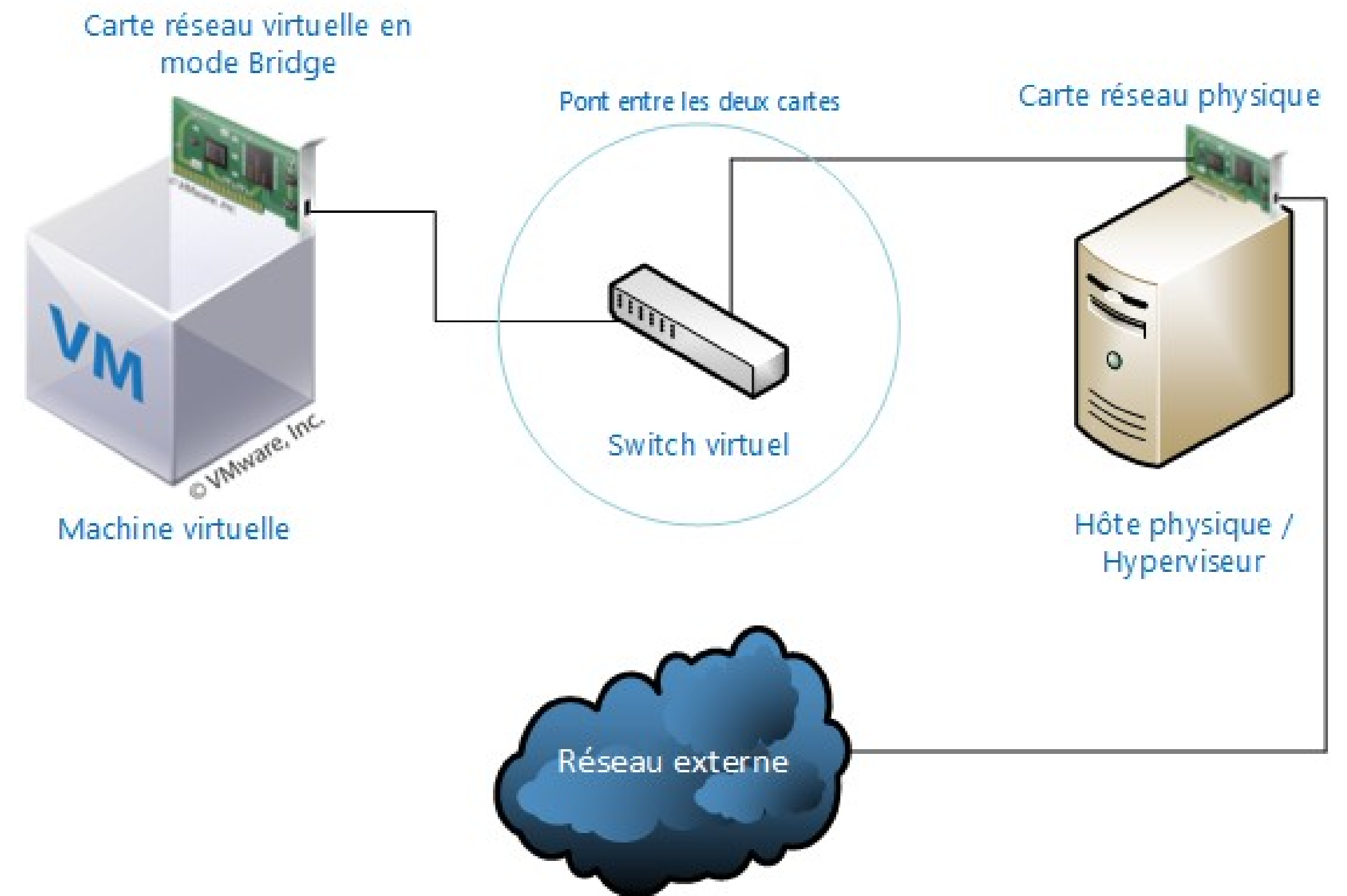
5/ Administration réseau

Les types de connexion - Bridge

Un réseau **Bridge (ou accès par pont)** permet à la machine virtuelle d'avoir sa propre adresse IP.

Ce mode est sûrement le plus utilisé puisqu'il permet de connecter une machine virtuelle directement sur le réseau physique sur lequel est branchée la carte réseau physique de l'hôte.

Pour cela, un bridge c'est-à-dire un pont est créé entre la carte réseau virtuelle de l'application de virtualisation et la carte réseau de votre hôte physique. C'est en quelque sorte un partage de carte réseau, où le système d'exploitation de votre hôte physique partage sa carte physique avec le système d'exploitation de votre ou vos machines virtuelles.



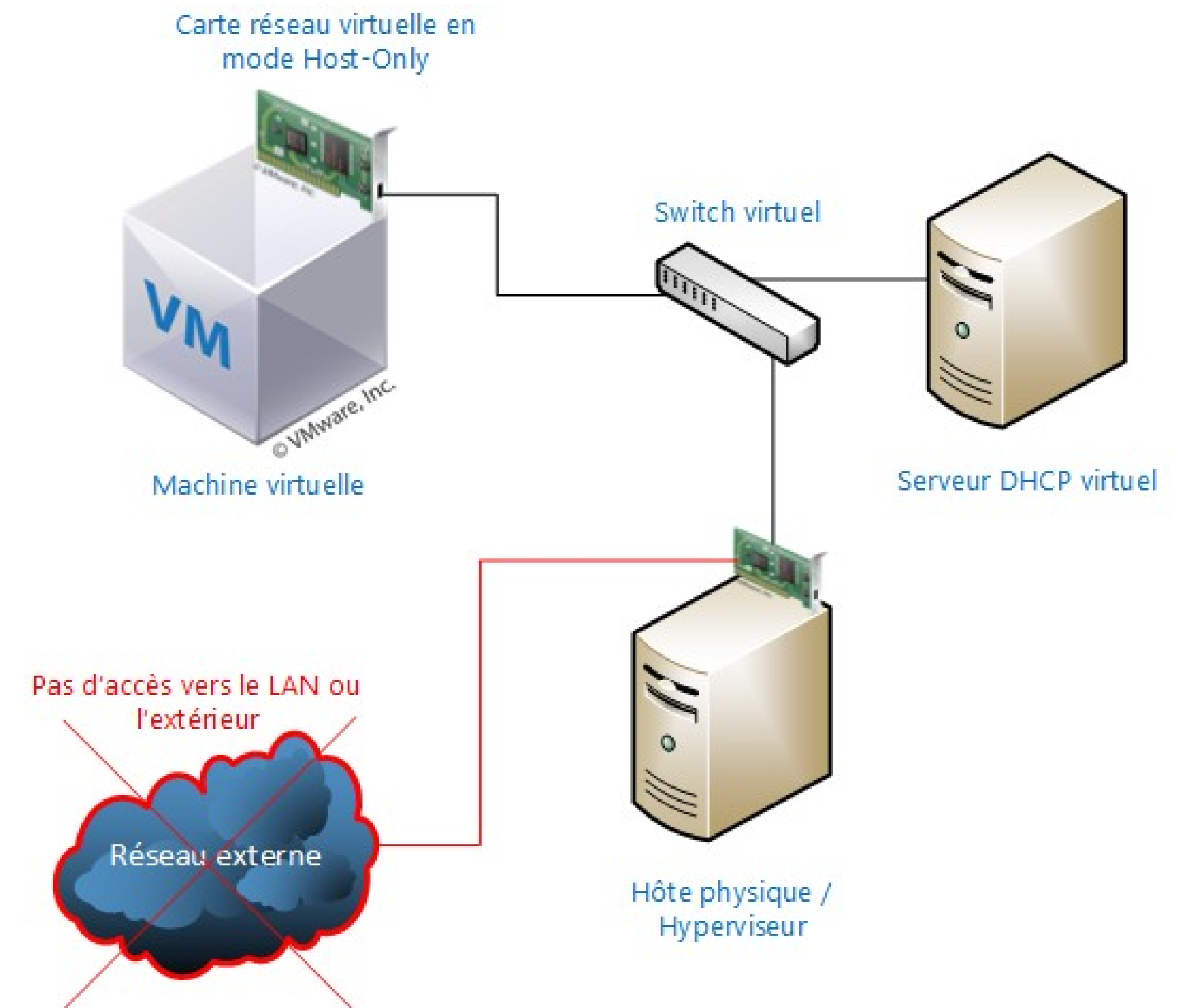
5/ Administration réseau

Les types de connexion – Host-Only

Un réseau **Host-Only (ou réseau privé hôte)** permet uniquement un accès réseau entre l'hôte et la machine virtuelle.

Ce type de connexion ne permet pas de sortir vers un réseau extérieur, ni d'accéder au réseau local par l'intermédiaire de la carte réseau physique de la machine physique hôte.

Comme son nom l'indique, ce mode permet uniquement d'établir une connexion entre la machine virtuelle et la machine physique.



5/ Administration réseau

Transfert de fichiers

La commande SCP permet de transférer des fichiers entre deux hôtes distants.

Commande : *scp -rp <fichier> user@<ip>:PATH*

D'autres outils graphiques peuvent être utilisés comme WinSCP et Filezilla.

5/ Administration réseau

Serveur Apache

Il est possible de monter un serveur Apache sur notre distribution. Pour cela rien de plus simple, il suffit de lancer le service apache.

- *service apache2 start*


```
(kali㉿kali)-[/tmp]
$ service apache2 start

(kali㉿kali)-[/tmp]
$ service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; disabled; preset: disabled)
   Active: active (running) since Mon 2023-01-02 05:14:38 EST; 3s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 18242 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
  Main PID: 18260 (apache2)
    Tasks: 6 (limit: 7105)
   Memory: 19.1M
      CPU: 61ms
   CGroup: /system.slice/apache2.service
           └─18260 /usr/sbin/apache2 -k start
             └─18262 /usr/sbin/apache2 -k start
               └─18263 /usr/sbin/apache2 -k start
                 └─18264 /usr/sbin/apache2 -k start
                   └─18265 /usr/sbin/apache2 -k start
                     └─18266 /usr/sbin/apache2 -k start
```

```
(kali㉿kali)-[/tmp]
$ netstat -tnlp
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp6      0      0 :::80                  :::*                    LISTEN      -

(kali㉿kali)-[/tmp]
$
```

127.0.0.1

 **debian**

Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

5/ Administration réseau

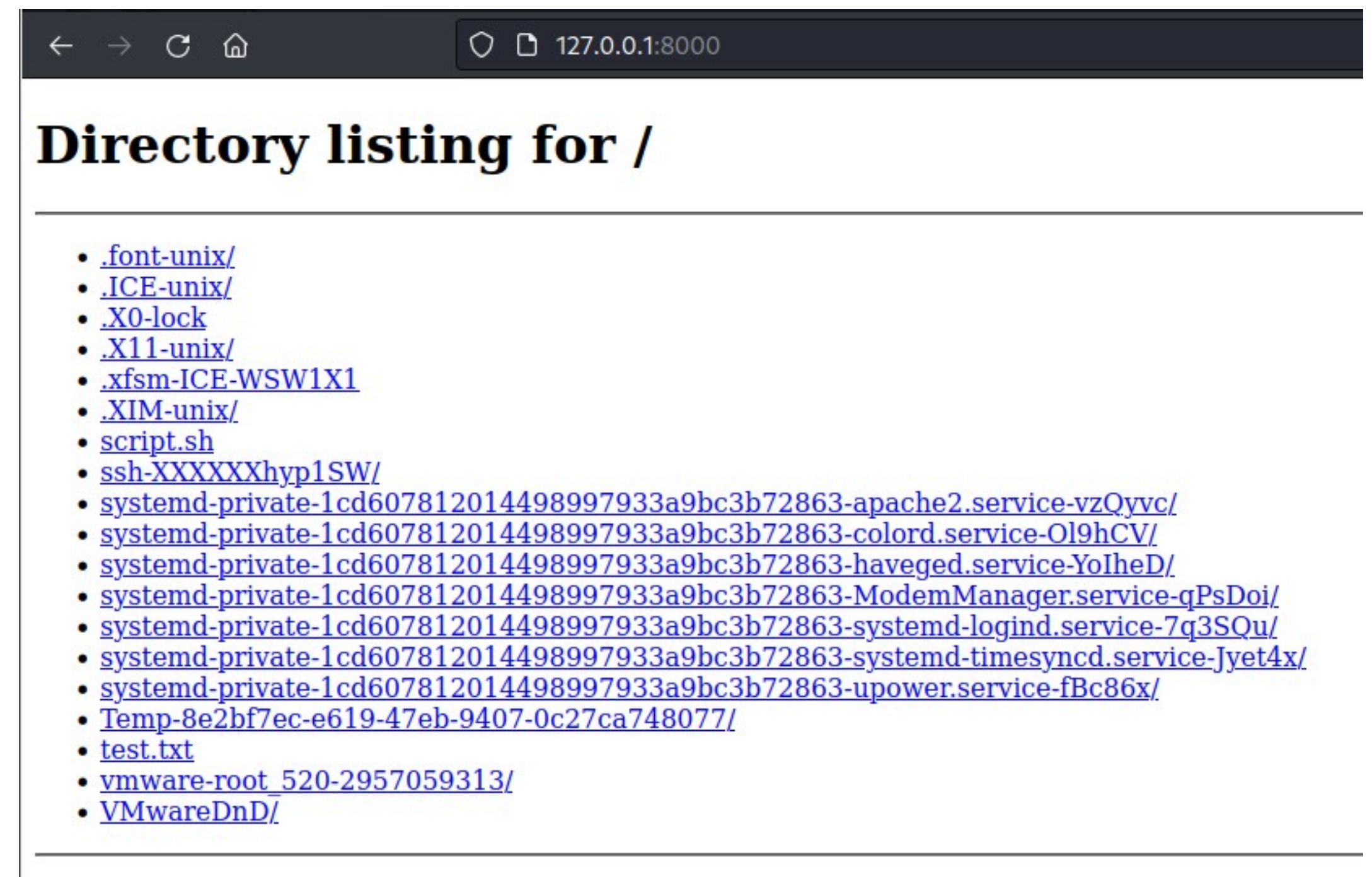
Serveur Python

Il est possible également de monter un serveur web via Python. Ceci est très utile pour partager des fichiers rapidement par exemple.

- *python3 -m http.server*

```
(kali@kali)-[/tmp]
$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
```

Un serveur sur le port 8000 par défaut est monté.



5/ Administration réseau

Investigation réseau

L'outil TCPDump peut s'avérer utile afin d'intercepter les flux sur notre interface réseau. C'est un équivalent en ligne de commande de Wireshark.

Pour lancer TCPDump :

- *`tcpdump -i eth0` (utiliser l'option `-w` pour sauver dans un dump)*

```
(kali㉿kali)-[/tmp]
└─$ sudo tcpdump -i eth0
[sudo] password for kali:
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
05:26:19.585222 IP 192.168.187.1.57621 > 192.168.187.255.57621: UDP, length 44
05:26:19.619896 IP 192.168.187.128.60184 > 192.168.187.2.domain: 64920+ PTR? 255.187.168.192.in-addr.arpa. (46)
05:26:19.621149 IP 192.168.187.2.domain > 192.168.187.128.60184: 64920 NXDomain 0/0/0 (46)
05:26:19.621280 IP 192.168.187.128.59338 > 192.168.187.2.domain: 7034+ PTR? 1.187.168.192.in-addr.arpa. (44)
05:26:19.622340 IP 192.168.187.2.domain > 192.168.187.128.59338: 7034 NXDomain 0/0/0 (44)
05:26:19.725803 IP 192.168.187.128.56874 > 192.168.187.2.domain: 46624+ PTR? 2.187.168.192.in-addr.arpa. (44)
05:26:19.727122 IP 192.168.187.2.domain > 192.168.187.128.56874: 46624 NXDomain 0/0/0 (44)
```

5/ Administration réseau

Fichier /etc/hosts

Le fichier /etc/hosts permet d'attribuer un nom d'hôte à une adresse IP.

Il est utilisé pour les réseaux locaux de petite taille. Le fichier hosts est utilisé sous tous les systèmes d'exploitations lors de l'accès à Internet, ce fichier est consulté avant l'accès au serveur DNS. C'est un simple fichier qui contient sur la même ligne une adresse IP et parfois le nom de domaine.

```
(kali㉿kali)-[~]  
$ cat /etc/hosts  
127.0.0.1    localhost  
127.0.1.1    kali  
10.129.228.50 shocker.htb  
10.129.228.71 metatwo.htb  
10.129.228.71 metapress.htb  
10.129.95.203 writeup.htb  
  
# The following lines are desirable for IPv6 capable hosts  
::1         localhost ip6-localhost ip6-loopback  
ff02::1     ip6-allnodes  
ff02::2     ip6-allrouters
```

5/ Administration réseau

DNS

Le fichier `/etc/resolv.conf` permet de définir les adresses IP des serveurs DNS.

Ce fichier est généré de manière automatique lors de la connexion. Il peut cependant être modifié afin d'ajouter d'autres serveurs, comme par exemple celui de google : `8.8.8.8`.

```
(kali@kali) [~]  
$ cat /etc/resolv.conf  
# Generated by NetworkManager  
search localdomain  
nameserver 192.168.187.2
```

5/ Administration réseau

Fichier /etc/network/interfaces

Ce fichier permet de définir et configurer les adresse IP des différentes interfaces disponibles du système. Par défaut les adresses sont générées automatiquement via DHCP (Dynamic Host Configuration Protocol), cependant il est possible de fixer nos adresses IP.

```
(kali@kali)-[~]  
$ cat /etc/network/interfaces  
# This file describes the network interfaces available on your system  
# and how to activate them. For more information, see interfaces(5).  
  
source /etc/network/interfaces.d/*  
  
# The loopback network interface  
auto lo  
iface lo inet loopback
```

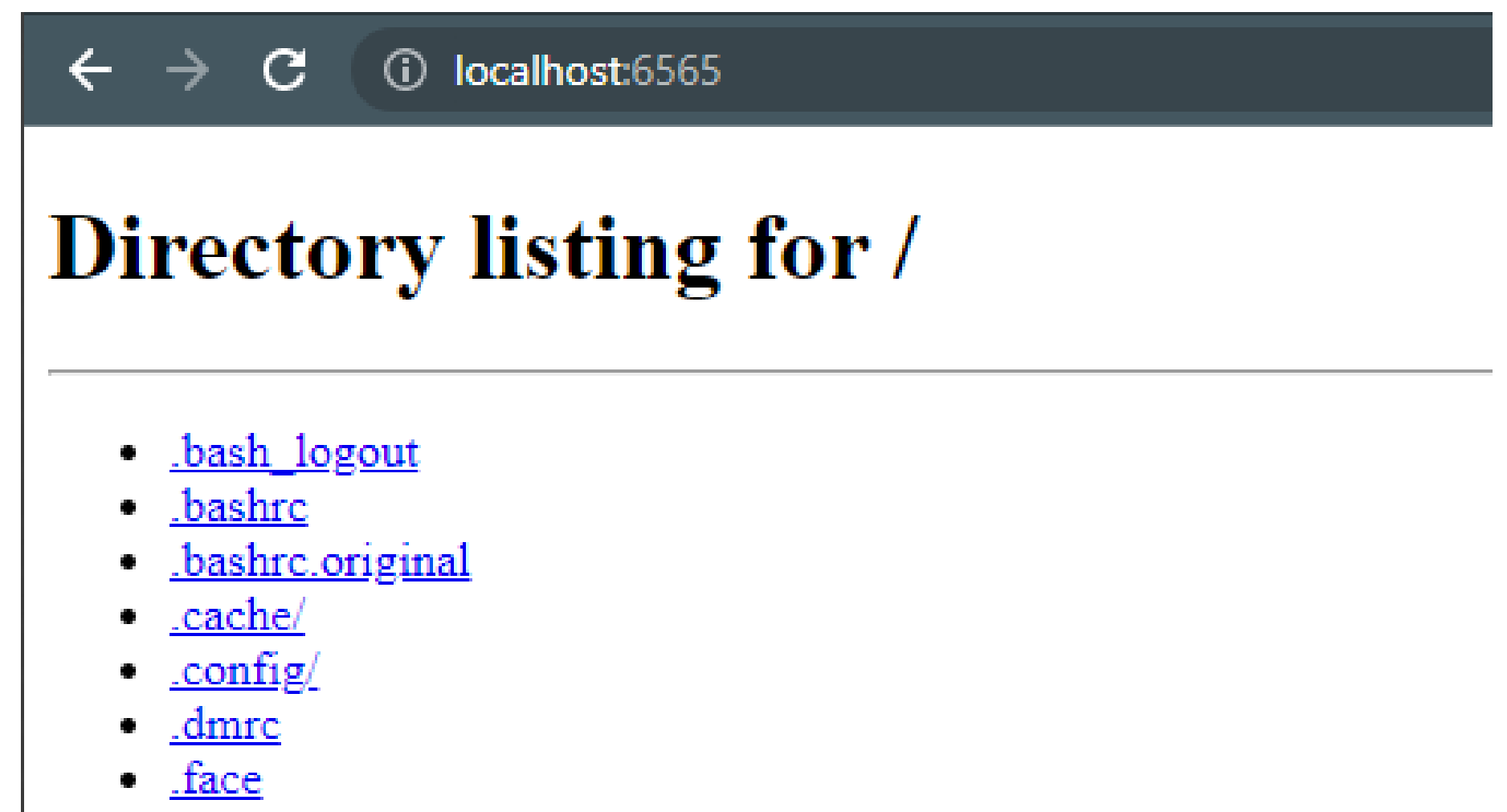
5/ Administration réseau

Redirection de port – NAT

Lorsque l'on est configuré en NAT, puisque la VM ne dispose pas de sa propre carte réseau (elle passe par l'hôte), il est plus difficile de dialoguer avec celle-ci. Pour cela, il est intéressant de réaliser une redirection de port.

Règles de redirection de ports

Nom	Protocole	IP hôte	Port hôte	IP invité	Port invité
Accès serveur python	TCP	127.0.0.1	6565	10.0.2.15	8000



5/ Administration réseau

Connexion par certificat - ssh-keygen

Lorsque l'on administre un grand nombre de serveur, il peut être compliqué de retenir la totalité des différents mots de passe.

Pour cela, nous pouvons réaliser une connexion par certificat (via ssh-keygen).

Permet notamment de n'avoir qu'une passphrase pour l'ensemble des serveurs sur lequel le certificat est déployé.



TP7 – Administration réseau

6/ Sécurité

6/ Sécurité

Nmap

L'outil Nmap est intéressant afin de scanner un hôte sur le réseau pour détecter des services accessibles (ports ouverts).

Nmap dispose également de scripts permettant de scanner de manière plus profonde un système.

Commande par défaut :

- *nmap <ip>*

Les différentes fonctionnalités et options de Nmap seront détaillées dans un TP dédié.

6/ Sécurité

Iptables

Iptables est le système de pare-feu interne au système Linux. Il permet de définir des règles servant à accepter ou rejeter des flux.

Pour voir les règles iptables :

- *iptables -L*

Pour supprimer les règles iptables :

- *iptables -F*

6/ Sécurité

SSH

Ssh est un utilitaire très intéressant, cependant il est important de vérifier qu'il est correctement configuré.

Le fichier de configuration se trouve dans */etc/ssh/sshd_config*

Pour cela :

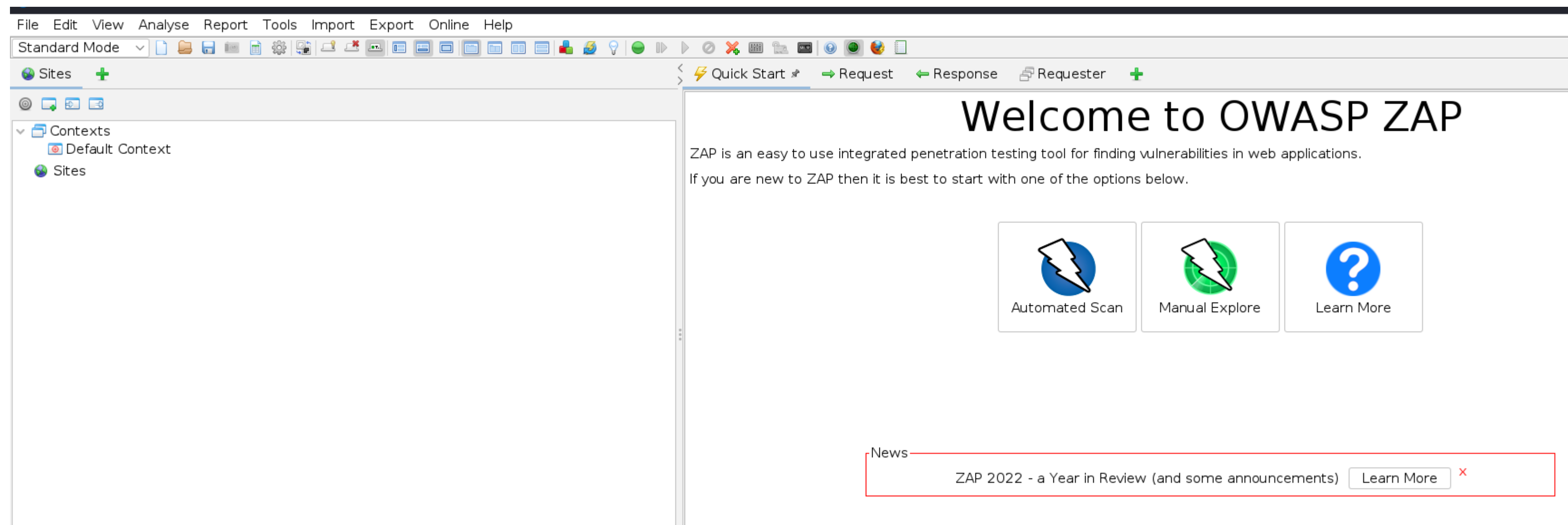
- Changer le port d'écoute et la version du protocole ;
- Utilisation de clefs privées/publiques ;
- Limiter les accès via *AllowUsers* ;
- Interdire l'accès à Root via « *PermitRootLogin no* » ;

6/ Sécurité

OWASP – ZAP Proxy

OWASP ZAP est un outil développé par la communauté OWASP. Vous retrouverez différents projets concernant l'OWASP, notamment le TOP TEN qui remonte les 10 vulnérabilités Web les plus utilisées dans le monde.

OWASP ZAP est un outil de scan open source qui fait également office de proxy.



6/ Sécurité

Sudo

La commande sudo permet d'exécuter une ou plusieurs commandes avec les privilèges de root.

Il n'est pas nécessaire de connaître son mot de passe.

Le fichier de configuration sudo est présent dans le répertoire suivant :

- */etc/sudoers*

Pour modifier le fichier de configuration, utiliser la commande suivante :

- *visudo*

6/ Sécurité

Mot de passe par défaut USER

Il est important de changer son mot de passe utilisateur, car il est souvent par défaut dans les distributions que l'on installe.

Exemple pour la distribution kali :

- kali:kali

Pour cela, utiliser la commande suivante :

- *passwd*

6/ Sécurité

Shadow file

Le fichier /etc/shadow est un fichier important et à sécuriser, car il contient les informations concernant les mots de passe des utilisateurs du système.

Si un hash est fuite de ce fichier, il est alors possible de le casser.

```
(kali㉿kali)-[~]  
$ sudo cat /etc/shadow | grep redteam  
redteam:$y$j9T$fKyMGtMjBz75snMjU15xh1$ucEvjgek9UhHxb6ozMwaGWLl9prUqY4Mf6SWPSkCq31:19372:0:99999:7:::  
  
(kali㉿kali)-[~]rch | Alerts | Output |  
$  
Filter: OFF | Export
```

6/ Sécurité

Crack d'un mot de passe

En cas de vol du hash d'un mot de passe, il est possible de tenter de retrouver le mot de passe, via plusieurs outils disponibles en open source.

Il est ainsi possible de tester la robustesse de son propre mot de passe en utilisant ces outils.

```
(kali@kali)-[/tmp]
$ sudo cat /etc/shadow | grep redteam > hash.out

(kali@kali)-[/tmp]
$ ll
total 160
-rw-r--r-- 1 kali kali    8 Jan 15 10:45 dico.lst
-rw-r--r-- 1 kali kali  101 Jan 15 10:45 hash.out
```


6/ Sécurité

Décodage - base64

Sur le web, un grand nombre de données peuvent être encodées en base64.

La forme de ce type d'encodage est la suivante :

```
(kali@kali)-[~]  
$ echo "bonjour" | base64  
Ym9uam91cgo=
```

TP8 – Sécurité

Contact

- Vianney SELOSSE
- Mail : vianney.selosse@ynov.com

