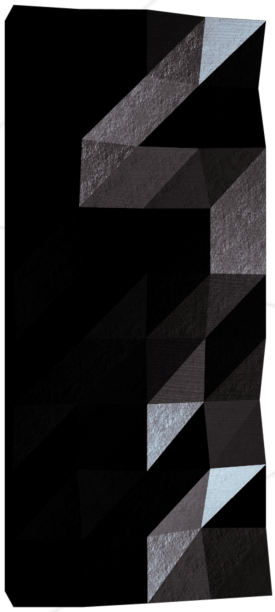


Born2beRoot

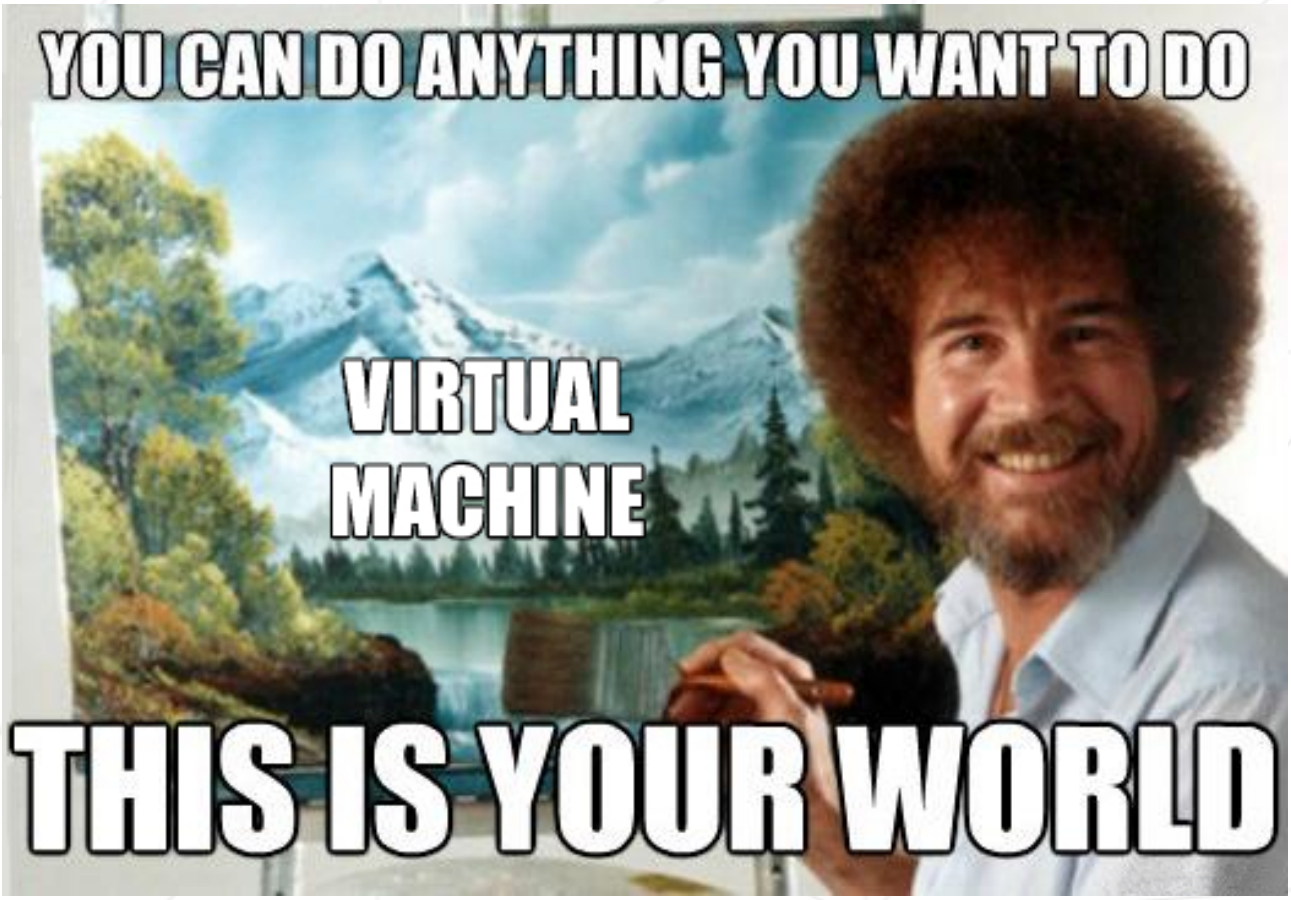
Resumen: Este documento es un ejercicio de administración de sistemas.

Versión: 3.4

Índice general

Capítulo I

Preámbulo



Capítulo II

Introducción

Este proyecto busca introducirte al maravilloso mundo de la virtualización.

Crearás tu primera máquina en `VirtualBox` (o `UTM` si no puedes utilizar `VirtualBox`) bajo instrucciones específicas. Por lo tanto, al final del proyecto, serás capaz de configurar tu propio sistema operativo utilizando reglas estrictas.

Capítulo III

Instrucciones generales

- El uso de `VirtualBox` es obligatorio (o `UTM` en caso de que `VirtualBox` no funcione en tu máquina).
- Solo debes entregar un archivo llamado `signature.txt` en la raíz de tu repositorio. Debes pegar en él la firma del disco virtual de tu máquina. Ve a Entrega y evaluación para más información.

Capítulo IV

Parte obligatoria

Este proyecto consiste en configurar tu primer servidor siguiendo una serie de normas concretas.

Como consiste en configurar un servidor, deberás instalar el número mínimo de servicios. Por este motivo, una interfaz gráfica no tiene sentido. Está prohibido por tanto instalar X.org o cualquier servidor gráfico equivalente. En caso de hacerlo, tu nota será 0.

Deberás elegir como sistema operativo la última versión estable de `Debian` (no testing/unstable), o la última versión estable de `Rocky`. Se recomienda encarecidamente `Debian` si no tienes experiencia en administración de sistemas.

Configurar Rocky es bastante complejo. Por lo tanto, no tienes que configurar KDump. Sin embargo, SELinux debe ejecutarse al iniciar y su configuración debe adaptarse a las necesidades del proyecto. AppArmor en Debian debe ejecutarse al inciar también.

Debes crear al menos 2 particiones cifradas usando `LVM`. Puedes encontrar un ejemplo de lo que se espera debajo:

```
wil@wil:~$ lsblk
NAME                  MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda                     8:0    0    8G  0 disk
├─sda1                  8:1    0  487M  0 part  /boot
├─sda2                  8:2    0    1K  0 part
└─sda5                  8:5    0  7.5G  0 part
  └─sda5_crypt        254:0    0  7.5G  0 crypt
    ├─wil--vg-root    254:1    0  2.8G  0 lvm   /
    ├─wil--vg-swap_1  254:2    0  976M  0 lvm   [SWAP]
    └─wil--vg-home    254:3    0  3.8G  0 lvm   /home
sr0                    11:0    1 1024M  0 rom
wil@wil:~$ _
```

Durante la defensa, se te harán unas preguntas sobre el sistema operativo que has elegido. Debes saber, por lo tanto, las diferencias entre aptitude y apt, o qué son SELinux y AppArmor. En definitiva, ¡entiende lo que estás utilizando!

El servicio SSH se ejecutará obligatoriamente en el puerto 4242 de tu máquina virtual. Por seguridad, no debe ser posible conectarte a través de SSH como root.

El uso de SSH será comprobado durante la defensa creando un nuevo usuario. Por lo tanto, debes entender cómo funciona.

Debes configurar tu sistema operativo con el firewall UFW, (o firewalld en Rocky) dejando solamente el puerto 4242 abierto en tu máquina virtual.

Tu firewall debe estar activo cuando ejecutes la máquina virtual. Para Para Rocky, debes usar firewalld en lugar de UFW

- El hostname de tu máquina virtual debe ser tu login terminado en 42 (por ejemplo, wil42). Deberás modificar este hostname durante tu evaluación.
- Debes implementar una política de contraseñas fuerte.
- Debes instalar y configurar sudo siguiendo reglas estrictas.
- Además del usuario root, un usuario con tu login como nombre debe existir.
- Este usuario debe pertenecer a los grupos user42 y sudo.

Durante la defensa, deberás crear un usuario y asignárselo a un grupo.

Para configurar una política de contraseñas fuerte, deberás cumplir los siguientes requisitos:

- Tu contraseña debe expirar cada 30 días.
- El número mínimo de días permitido antes de modificar una contraseña deberá ser 2.
- El usuario debe recibir un mensaje de aviso 7 días antes de que su contraseña expire.
- Tu contraseña debe tener como mínimo 10 caracteres de longitud. Debe contener una mayúscula, una minúscula y un número. Por cierto, no puede tener más de 3 veces consecutivas el mismo carácter.

- La contraseña no puede contener el nombre del usuario.
- La siguiente regla no se aplica a la contraseña para root: La contraseña debe tener al menos 7 caracteres que no sean parte de la antigua contraseña.
- Evidentemente, tu contraseña para root debe seguir esta política.

`Después de preparar tus archivos de configuración, deberás cambiar la contraseña de todas las cuentas presentes en la máquina virtual, root incluida.`

Para configurar una contraseña fuerte para tu grupo `sudo`, debes cumplir con los siguientes requisitos:

- Autenticarte con `sudo` debe estar limitado a tres intentos en el caso de introducir una contraseña incorrecta.
- Un mensaje personalizado de tu elección debe mostrarse en caso de que la contraseña introducida sea incorrecta cuando se utilice `sudo`.
- Para cada comando ejecutado con `sudo`, tanto el input como el output deben quedar archivados en el directorio `/var/log/sudo/`.
- El modo `TTY` debe estar activado por razones de seguridad.
- Por seguridad, los directorios utilizables por `sudo` deben estar restringidos. Por ejemplo:
  `/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin`

Finalmente, debes crear un script sencillo llamado `monitoring.sh`. Debe estar desarrollado en `bash`.

Cuando el servidor inicie, el script mostrará cierta información (listada debajo) en todos los terminales cada 10 minutos (Échale un vistazo a `wall`). El banner de wall es opcional. Ningún error debe ser visible.

Tu script debe siempre mostrar la siguiente información:

- La arquitectura de tu sistema operativo y su versión de kernel.
- El número de núcleos físicos.
- El número de núcleos virtuales.
- La memoria RAM disponible actualmente en tu servidor y su porcentaje de uso.
- La memoria disponible actualmente en tu servidor y su utilización como un porcentaje.
- El porcentaje actual de uso de tus núcleos.
- La fecha y hora del último reinicio.
- Si LVM está activo o no.
- El número de conexiones activas.
- El número de usuarios del servidor.
- La dirección IPv4 de tu servidor y su MAC (Media Access Control)
- El número de comandos ejecutados con `sudo`.

`Durante la defensa, te preguntarán cómo funciona este script. Deberás interrumpirlo sin modificarlo. Échale un vistazo a cron.`

Este es un ejemplo de cómo se espera que funcione tu script:

```
Broadcast message from root@wil (tty1) (Sun Apr 25 15:45:00 2021):

    #Architecture: Linux wil 4.19.0-16-amd64 #1 SMP Debian 4.19.181-1 (2021-03-19) x86_64 GNU/Linux
    #CPU physical : 1
    #vCPU : 1
    #Memory Usage: 74/987MB (7.50%)
    #Disk Usage: 1009/2Gb (49%)
    #CPU load: 6.7%
    #Last boot: 2021-04-25 14:45
    #LVM use: yes
    #TCP Connections : 1 ESTABLISHED
    #User log: 1
    #Network: IP 10.0.2.15 (08:00:27:51:9b:a5)
    #Sudo : 42 cmd
```

Debajo tienes dos comandos que puedes utilizar para comprobar algunos requisitos del subject:

Para `Rocky`:

```
[root@wil will]# head -n 2 /etc/os-release
NAME="Rocky Linux"
VERSION="8.7 (Green Obsidian)"
[root@wil will]# sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     actual (secure)
Max kernel policy version:      33
[root@wil will]# ss -tunlp
Netid State  Recv-Q Send-Q  Local Address:Port   Peer Address:Port Process
tcp   LISTEN 0      128           0.0.0.0:4242        0.0.0.0:*     users:(("sshd",pid=28429,fd=6))
tcp   LISTEN 0      128              [::]:4242           [::]:*     users:(("sshd",pid=28429,fd=4))
[root@wil will]# firewall-cmd --list-service
ssh
[root@wil will]# firewall-cmd --list-port
4242/tcp
[root@wil will]# firewall-cmd --state
running
[root@wil will]# _
```

Para `Debian`:

```
root@wil:~# head -n 2 /etc/os-release
PRETTY_NAME="Debian GNU/Linux 10 (buster)"
NAME="Debian GNU/Linux"
root@wil:/home/wil# /usr/sbin/aa-status
apparmor module is loaded.
root@wil:/home/wil# ss -tunlp
Netid State  Recv-Q Send-Q Local Address:Port  Peer Address:Port

tcp   LISTEN 0      128          0.0.0.0:4242       0.0.0.0:*     users:(("sshd",pid=523,fd=3))
tcp   LISTEN 0      128             [::]:4242          [::]:*     users:(("sshd",pid=523,fd=4))
root@wil:/home/wil# /usr/sbin/ufw status
Status: active

To                         Action      From
--                         ------      ----
4242                       ALLOW       Anywhere
4242 (v6)                  ALLOW       Anywhere (v6)
```

Capítulo V

Parte bonus

Lista de bonus:

- Configura correctamente las particiones para obtener una estructura similar a la mostrada debajo:

```
# lsblk
NAME                  MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINT
sda                     8:0    0  30.8G  0 disk
├─sda1                  8:1    0   500M  0 part  /boot
├─sda2                  8:2    0     1K  0 part
└─sda5                  8:5    0  30.3G  0 part
  └─sda5_crypt        254:0    0  30.3G  0 crypt
    ├─LVMGroup-root   254:1    0    10G  0 lvm   /
    ├─LVMGroup-swap   254:2    0   2.3G  0 lvm   [SWAP]
    ├─LVMGroup-home   254:3    0     5G  0 lvm   /home
    ├─LVMGroup-var    254:4    0     3G  0 lvm   /var
    ├─LVMGroup-srv    254:5    0     3G  0 lvm   /srv
    ├─LVMGroup-tmp    254:6    0     3G  0 lvm   /tmp
    └─LVMGroup-var--log 254:7  0     4G  0 lvm   /var/log
sr0                    11:0    1  1024M  0 rom
```

- Configura un sitio WordPress funcional con los siguientes servicios: lighttpd, MariaDB, y PHP.
- Configura un servicio de tu elección que consideres útil (NGINX / Apache2 excluidos). Durante la defensa, deberás justificar tu elección.

Para completar la parte bonus, tienes la posibilidad de configurar servicios adicionales. En este caso, puedes abrir más puertos de acuerdo a tus necesidades. Por supuesto, las reglas de UFW/Rocky deben adaptarse según sea necesario.

La parte bonus solo será evaluada si la parte obligatoria está PERFECTA. Perfecta significa que la parte obligatoria es integramente funcional y completa. Si no has completado TODA la parte obligatoria, tu parte bonus no será evaluada.

Capítulo VI

Entrega y evaluación

Solo deberás enviar un archivo `signature.txt` en la raíz de tu repositorio `Git`. Debes pegar en él la firma de tu disco virtual. Para obtener esta firma, debes primero abrir la ruta por defecto de instalación (es decir, donde tus VMs se guardan).

- Windows: `%HOMEDRIVE%%HOMEPATH%\VirtualBox VMs\`
- Linux: `~/VirtualBox VMs/`
- MacM1: `~/Library/Containers/com.utmapp.UTM/Data/Documents/`
- MacOS: `~/VirtualBox VMs/`

Recupera entonces la firma del archivo `".vdi"` (o `".qcow2"` para usuarios de `UTM`) de tu máquina virtual en formato `sha1`. Debajo tienes 4 ejemplos de comandos para un archivo `rocky_serv.vdi`:

- Windows: `certUtil -hashfile rocky_serv.vdi sha1`
- Linux: `sha1sum rocky_serv.vdi`
- For Mac M1: `shasum rocky.utm/Images/disk-0.qcow2`
- MacOS: `shasum rocky_serv.vdi`

Este es un ejemplo del tipo de resultado que obtendrás:

- `6e657c4619944be17df3c31faa030c25e43e40af`

Por favor, ten en cuenta que la firma de tu VM puede verse alterada tras tu primera evaluación. Para solucionar este problema, puedes duplicar tus máquinas virtuales o usar save state.

Evidentemente está PROHIBIDO entregar tu máquina virtual en tu repositorio de Git. Durante la defensa, el contenido del archivo signature.txt se comparará con la firma de tu máquina virtual. Si las dos no son idénticas, tu nota será 0.

0010 01 11 111 001 000 11 01 10 1 0000 01 1 1010 111 11 0 000
011 00 1 0000 1 0000 0 01 0100 1 0 010 10 01 1 0 0001 0 010 000
00 111 10 111 0010 001100 001100 001100