PROYECTO BORN2BEROOT

1.- Descargar la imagen de la maquina virtual

Me descargo el fichero de instalación del sistema operativo.
1.1 Descargar imágenes usando HTTP
1.2 Versión estable
1.3 amd64 (compatible con Intel de 64 bits
1.4 debian 12.5 amd64-netins.iso

2. Instalación de la maquina virtual vox.

Ir a virtualbox.org y descargar
2.1 Abrir virtual box
2.2 Debian
*Nueva → Nombre →DEbian64bit → Memoria1024→crear disco duro virtual→UDI→ reservado dinamicamente→ sgoinfre/goinfre/42 …..→ configuración→ seleccionar archivo de disco (desde donde quieres que arranque) (Añadir, Descargas, Seleccionar) -< Instal.

* Escogeremos el idioma que usaremos para la instalación y el predeterminado que se le quedará al sistema
* Introducimos nuestro País, territorio o zona. En mi caso pondré Other.
* Como he seleccionado other, debo indicar mi continente o región. En mi caso pongo Europe .
* Seleccionamos el país. En mi caso Spain .
* Seleccionamos United States.
* Importante seleccionar Spanish como configuración de teclado, ya que si no tendremos las teclas mal enlazadas.
* En este paso debemos elegir el Host Name de la máquina, el cual debe ser tu login seguido de 42.
* Este apartado lo dejaremos vacío, ya que el subject no mencionada nada de Domain name.
* Debemos introducir una contraseña para la cuenta de administración del sistema. Importante apuntarla o hacer una foto, ya que le daremos uso. Si quieres ver la contraseña para asegurarte de que la has escrito correctamente debes tabular hasta llegar a la opción Show Password in Clear debes darle a la barra espaciadora y se mostrará la clave.
* Repetimos el proceso de nuevo para comprobar que no la hayamos escrito mal.
* Elegimos el nombre de nuestro nuevo usuario. Como indica el subject, hay que crear un usuario adicional que no sea el root con nuestro login, por ese motivo llamaré gemartin a mi nuevo usuario.
* Volvemos a poner el nombre de usuario.
*Ahora debemos introducir la contraseña de nuestro nuevo usuario. Como la anterior, repetiremos el proceso para comprobar que no la hayas escrito mal y también es importante que la guardes porque le daremos uso más adelante.
* Seleccionamos la hora de nuestra ubicación.
* Escogeremos la tercera opción Guied - use entire disk and set up encrypted LVM, ya que el subject nos dice que deben ser particiones cifradas.
* Seleccionamos el disco en el que queremos hacer el particionado (Solo debe haber un disco)
*Una vez hayamos escogido el disco deberemos hacer el particionado tal y como nos piden. Para realizarlo adecuadamente debemos seleccionar la segunda opción Separate /home partition.
* Yes
* De nuevo deberemos poner una contraseña, esta vez será la frase de encriptación. Como te he comentado previamente deberás repetir el proceso y la debes anotar, ya que será importante en un futuro.
* En este paso debemos introducir la cantidad de volumen que usaremos para la partición guiada.
*Debemos introducir max o el número de tamaño máximo diEn este paso debemos introducir la cantidad de volumen que usaremos para la partición guiada. Debemos introducir max o el número de tamaño máximo disponible en mi caso es 12.4 GB.sponible en mi caso es 12.4 GB.
* Para finalizar la partición y escribir los cambios en el disco le daremos a la opción Finish partitioning and write changes to disk.
* Seleccionamos la opción Yes para continuar y confirmar que no queremos hacer más cambios en el disco.
*◦ Seleccionamos la opción No, ya que no necesitamos paquetes adicionales.
*◦ Escogemos nuestro País.
* ◦ Escogemos deb.debian.org, ya que es lo que recomienda Debian.
* ◦ Esta opción la dejaremos vacía y le daremos Continue.
* ◦ Seleccionamos la opcion No, ya que no queremos que los developers vean nuestras estadísticas aunque sean anónimas.
* Quitaremos todas las opciones de software (con la barra espaciadora) y le daremos a Continue.
SE QUITA CON EL ESPACIO, NO CON EL DELETE.
* Seleccionaremos Yes para instalar GRUB boot en el disco duro.

* Escogeremos el dispositivo para la instalación del cargador de arranque
  /dev/sda (ata_VBOX_HARDDISK).
*◦ Le daremos a Continue para finalizar la instalación.

4 Configuración de la máquina virtual

* Lo primero que debemos hacer es seleccionar Debian GNU/Linux.
* Debemos introducir la contraseña de encriptación que utilizamos previamente. Xxxxxxx
* Debemos introducir el usuario y contraseña que hemos creado. En mi caso el usuario es gemartin y la contraseña Hola42spain.

4.1 - Instalación de sudo y configuración de usuarios y grupos

* Para la instalación de sudo primero debemos estar en el usuario root, para ello pondremos Su en el terminal e introduciremos la contraseña, en mi caso es Hola42bcn. Una vez hemos accedido al usuario root, debemos poner el comando apt install sudo para así instalar los paquetes necesarios.

su – root (intro)

contraseña

apt install sudo

* Debemos reiniciar la máquina para que se apliquen los cambios. Para ello haremos uso del comando sudo reboot y esperaremos a que se reinicie.
Sudo reboot

* Una vez reiniciado debemos volver a introducir las contraseñas de cifrado y del usuario.

Para verificar que hayamos instalado sudo correctamente entraremos de nuevo en el usuario root y pondremos el comando sudo -V, este comando además de mostrarnos la versión de sudo también mostrará los argumentos pasados para configurar cuando se creó sudo y los plugins que pueden mostrar información más detallada. (Opcional) ➤ Puesto que el output del comando es muy largo, si deseamos verlo completamente debemos redireccionar la salida del mismo a un fichero sudo -V > file.txt y luego editar el fichero nano file.txt. O poner | more después del comando.

* Siguiendo en el usuario root crearemos un usuario con nuestro login con el comando sudo adduser login como nosotros ya hemos creado el usuario en la instalación nos debe aparecer que el usuario ya existe.

* Ahora deberemos crear un nuevo grupo llamado user42. Para crearlo debemos hacer

addgroup user42.

* Se ha creado correctamente el grupo? Lo cierto es que sí, ya que no ha habido ningún mensaje de error, aun así podemos comprobar si se ha creado con el comando

getent group nombre_grupo

o también podemos hacer

cat /etc/group

y podremos ver todos los grupos y los usuarios que hay dentro de ellos.

* Con el comando adduser user group incluiremos al usuario en el grupo. Debemos incluir al usuario en los grupos sudo y user42.

* Una vez los hayamos introducido para chequear que todo se haya hecho correctamente podemos ejecutar el comando getent group nombre_grupo o también podemos editar el fichero /etc/group nano /etc/group y en los grupos sudo y login42 deberá aparecer nuestro usuario.

Se puede crear un usuario de una sola vez haciendo lo siguiente :

sudo useradd – g users -G cdrom, dialout, sudo -m -d / *home / test test*

-g grupo principal

-G grupo secundarios (separados por coma)

-m crear las capetas que se pongan en -d

-d la direccion dodne van a estar los archivos del usuario que se crea

nombre del usuario que se va a crearemos

4.2 - Instalación y configuración SSH

* Lo primero que haremos será hacer sudo apt update para actualizar los repositorios que definimos en el archivo /etc/apt/sources.list

* Acto seguido instalaremos la herramienta principal de conectividad para el inicio de sesión remoto con el protocolo SSH, esta herramienta es OpenSSH. Para instalarla debemos introducir el comando sudo apt install openssh-server. En el mensaje de confirmación ponemos Y, acto seguido esperaremos a que termine la instalación.

Para comprobar que se haya instalado correctamente haremos sudo service ssh status y nos debe aparecer active.

* Una vez terminada la instalación se han creado algunos ficheros que debemos configurar. Para ello utilizaremos Nano, o si tú lo prefieres, otro editor de texto. El primer fichero que editaremos será /etc/ssh/sshd_config. Si no estas desde el usuario root no tendrás permisos de escritura, para ello haremos su y ponemos la contraseña para entrar al usuario root o si no quieres entrar en el usuario root, ponemos sudo al principio del comando sudo nano /etc/ssh/sshd_config.

* Los # al comienzo de una línea significan que está comentada, las líneas que vayamos a modificar deberás quitarle el comentario. Una vez estemos editando el fichero deberemos modificar las siguientes líneas:
  ➤ #Port 22 -> Port 4242 → Buscar en Vim es / Port (primera en mayuscula)

  ➤ #PermitRootLogin prohibit-password -> PermitRootLogin no

* Los # al comienzo de una línea significan que está comentada, las líneas que vayamos a modificar deberás quitarle el comentario. Una vez estemos editando el fichero deberemos modificar las siguientes líneas:
  port 22 → port 4242
* Por último, debemos reiniciar el servicio ssh para que así se actualicen las modificaciones que acabamos de realizar. Para ello debemos escribir el comando sudo service ssh restart y una vez reseteado miraremos el estado actual con sudo service ssh status y para confirmar que se hayan realizado los cambios en la escucha del servidor debe aparecer el Puerto 4242.

4-3 Instalación y configuración de UFW

* Lo primero que debemos hacer el instalar UFW, para ello haremos uso del comando sudo apt install ufw acto seguido escribiremos una y para confirmar que deseamos instalarlo y esperaremos a que termine.

* Una vez instalado debemos habilitarlo. Para ello debemos poner el siguiente comando sudo ufw enable y acto seguido nos debe indicar que el firewall está activo.

* Ahora lo que debemos hacer es que nuestro firewall permita las conexiones que se lleven a cabo mediante el puerto 4242. Lo haremos con el siguiente comando sudo ufw allow 4242.

* Por último, comprobaremos que está todo correctamente configurado mirando el estado de nuestro cortafuegos, en donde ya debe aparecer como permitidas las conexiones mediante el puerto 4242. Para ver el estado daremos uso del comando sudo ufw status.

4-4 Configurar contraseña fuerte para sudo

* Crearemos un fichero en la ruta /etc/sudoers.d/ a mi fichero yo le he decidido llamar sudo_config, ya que en ese fichero se almacenará la configuración de la contraseña. El comando exacto para crear el fichero es touch /etc/sudoers.d/sudo_config.
* Debemos crear el directorio sudo en la ruta /var/log porque cada comando que ejecutemos con sudo, tanto el input como el output, debe quedar almacenado en ese directorio. Para crearlo utilizaremos el comando mkdir /var/log/sudo.

* Debemos editar el fichero creado en el paso 1. Como he comentado anteriormente, puedes utilizar el editor que más te guste, pero yo haré uso de nano. Comando para editar el fichero: nano /etc/sudoers.d/sudo_config.

* Una vez estamos editando el fichero deberemos introducir los siguientes comandos para cumplir todos los requisitos que pide el subject.

```
Defaults  passwd_tries=3
Defaults  badpass_message="Mensaje de error personalizado"
Defaults  logfile="/var/log/sudo/sudo_config"
Defaults  log_input, log_output
Defaults  iolog_dir="/var/log/sudo"
Defaults  requiretty
Defaults  secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"
```

4-5 Configuración de política de contraseñas fuerte

* El primer paso será editar el fichero login.defs.

Login.defs sirve para cambair la politica de contraseñas para lso usuarios nuevos, los anteriores seguiran con la politica anterior.

* Una vez estemos editando el fichero, modificaremos los siguientes parámetros:

➤ PASS_MAX_DAYS 99999 -> PASS_MAX_DAYS 30

➤ PASS_MIN_DAYS 0 -> PASS_MIN_DAYS 2

*PASS_MAX_DAYS: Es el tiempo de expiración de la contraseña. El número corresponde a días.

PASS_MIN_DAYS: El número mínimo de días permitido antes de modificar una contraseña.

PASS_WARN_AGE: El usuario recibirá un mensaje de aviso indicando que faltan los días especificados para que expire su contraseña.

* Para poder seguir con la configuración debemos instalar los siguientes paquetes con este comando sudo apt install libpam-pwquality , acto seguido pondremos Y para confirmar la instalación y esperaremos a que termine.
* Lo siguiente que debemos hacer es volver a editar un fichero y modificar algunas líneas. Haremos nano /etc/pam.d/common-password.

*Después de retry=3 debemos añadir los siguientes comandos:

minlen=10 ➤ La cantidad mínima de caracteres que debe contener la contraseña.

ucredit=-1 ➤ Como mínimo debe contener una letra mayúscula. Ponemos el - ya que debe contener como mínimo un carácter, si ponemos + nos referimos a como máximo esos caracteres.

dcredit=-1 ➤ Como mínimo debe contener un dígito.

lcredit=-1 ➤ Como mínimo debe contener una letra minúscula.

maxrepeat=3 ➤ No puede tener más de 3 veces seguidas el mismo carácter.

reject_username ➤ No puede contener el nombre del usuario.

difok=7 ➤ Debe tener al menos 7 caracteres que no sean parte de la antigua contraseña.

enforce_for_root ➤ Implementaremos esta política para el usuario root.

4-6 Conectarse vía SSH

* Para conectarnos por SSH debemos cerrar la máquina, abrir VirtualBox y darle a configuración.

* Cerrar la maquina virtual y abrir virtual box

* Configuracion → Red-→ renvio de puertos-→ max en verde-→ puerto anfitrion 4242 → puerto invitado 4242

Para conectarse a una maquina virtual, hay que abrir un terminal.

Dentro del terminal:

ssh ~~p 4242 snunez~~z @localhost

p de puerto → snune-z el usuario con el que te quieres conectar-→ @ at localhost → el nombre de la maquina, en este caso como es la misma es localhost. Existe siempre en todas las maquinas.

5- Script

* Es una secuencia de comandos guardada en un fichero que cuando se ejecuta hará la función de cada comando y en orden establecido en el fichero.

Uname -a comando → para saber version de OS y kernel

* Para poder mostrar el número de núcleos físicos haremos uso del fichero /proc/cpuinfo el cual proporciona información acerca del procesador: su tipo, marca, modelo, rendimiento, etc.

* Usaremos el comando grep "physical id" /proc/cpuinfo | wc -l con el comando grep buscaremos dentro del fichero "physical id" y con wc -l contaremos las líneas del resultado de grep. Esto lo hacemos, ya que la manera de cuantificar los núcleos no es muy común. Si hay un procesador marcará 0 y si tiene más de un procesador, mostrará toda la información del procesador por separado, contando los procesadores usando la notación cero. De esta manera simplemente contaremos las líneas que hay, ya que es más cómodo cuantificarlo así.

*Los nucleos virtuales son los subprocesos.

grep processor /proc/cpuinfo | wc -l.

* Para mostrar la memoria RAM haremos uso del comando free para así ver al momento información sobre la RAM, la parte usada, libre, reservada para otros recursos, etc. Para más info sobre el comando, pondremos free --help. Nosotros daremos uso de free --mega , ya que en el subject aparece esa unidad de medida (Megabyte). Es importante poner --mega y no -m. Con -m nos referiremos a la unidad de medida Mebibyte y no es la que especifica el subject.

* Para filtrar la información e imprimir solo un campo concreto usamos el comando awk

```
free --mega | awk '$1 == "Mem:" {print $3}'
```

De la información que me devuelve free, en la linea 1 el campo de memoria usado que es el 3.

* Para obtener la memoria total el comando es prácticamente igual al anterior, lo único que deberemos cambiar es que en vez de printar la tercera palabra de la fila queremos la segunda

free --mega | awk '$1 == "Mem:" {print $2}'.

* Por último, debemos mostrar un porcentaje de la memoria usada. Para ello, de nuevo, utilizaremos un comando muy parecido a los dos anteriores. Lo único que cambiaremos es que combinaremos los dos comandos anteriores para tener dos variables, una que representa la memoria usada y la otra la total. Hecho esto haremos una operación para conseguir el tanto por ciento use/total*100 y el resultado de esta operación lo printaremos como aparece en el subject, entre paréntesis y con el símbolo % al final. El comando final es este: df -m | grep "/dev/" | grep -v "/boot" | awk '{use += $3} {total += $2} END {printf("(%d%%)\n"), use/total*100}'.

* Para poder ver la memoria del disco ocupada y disponible utilizaremos el comando df que significa "disk filesystem", se utiliza para obtener un resumen completo del uso del espacio en disco. Como en el subject indica la memoria utilizada se muestra en MB, así que entonces utilizaremos el flag -m. Acto seguido haremos un grep para que solo nos muestre las líneas que contengan "/dev/" y seguidamente volveremos a hacer otro grep con el flag -v para excluir las líneas que contengan "/boot". Por último utilizaremos el comando awk y sumaremos el valor de la tercera palabra de cada línea para una vez sumadas todas las líneas printar el resultado final de la suma. El comando entero es el siguiente:

df -m | grep "/dev/" | grep -v "/boot" | awk '{memory_use += $3} END {print memory_use}'.

* Para obtener el espacio total utilizaremos un comando muy parecido. Las únicas diferencias serán que los valores que sumaremos serán los $2 en vez de $3 y la otra diferencia es que en el subject aparece el tamaño total en Gb así que como el resultado de la suma nos da el número en Mb debemos transformarlo a Gb, para ello debemos dividir el número entre 1024 y quitar los decimales.

df -m |grep “/dev/” | grep -v *boot*” | awk ‘awk’{memory_result += $2} END {printf (“0fgb\n), memory_result/1024}’

* Por último, debemos mostrar un porcentaje de la memoria usada. Para ello, de nuevo, utilizaremos un comando muy parecido a los dos anteriores. Lo único que cambiaremos es que combinaremos los dos comandos anteriores para tener dos variables, una que representa la memoria usada y la otra la total. Hecho esto haremos una operación para conseguir el tanto por ciento use/total*100 y el resultado de esta operación lo printaremos como aparece en el subject, entre paréntesis y con el símbolo % al final.

df -m | grep "/dev/" | grep -v "/boot" | awk '{use += $3} {total += $2} END {printf("(%d%%)\n"), use/total*100}'.

* Para poder ver el porcentaje de uso de CPU haremos uso del comando vmstat. Este muestra estadísticas del sistema, permitiendo obtener un detalle general de los procesos, uso de memoria, actividad de CPU, estado del sistema, etc. Podríamos poner si ninguna opción, pero en mi caso pondré un intervalo de segundos de 1 a 4. También daremos uso del comando tail -1, que este lo que nos va a permitir es que solo produzca el output la última línea, entonces de las 4 generadas solo se printará la última. Por último, solo printaremos la palabra 15 que es el uso de memoria disponible. El comando entero es el siguiente: vmstat 1 4 | tail -1 | awk '{print $15}'. El resultado de este comando solo es una parte del resultado final, ya que todavía hay que hacer alguna operación en el script para que quede bien. Lo que habría que hacer es a 100 restarle la cantidad que nos ha devuelto nuestro comando, el resultado de esa operación lo printaremos con un decimal y un % al final y ya estaría hecha la operación.

vmstat 1 3| tail -1 | awk ‘{printf $15}’

* Para ver la fecha y hora de nuestro último reinicio haremos uso del comando who con el flag -b, ya que con ese flag nos mostrará por pantalla el tiempo del último arranque del sistema. Como ya nos ha pasado anteriormente, nos muestra más información de la que deseamos, así que filtraremos y solo mostraremos lo que nos interesa, para ello haremos uso del comando awk y compararemos si la primera palabra de una

línea es "system" se printará por pantalla la tercera palabra de esa línea, un espacio y la cuarta palabra. El comando entero sería el siguiente: who -b | awk '$1 == "system" {print $3 " " $4}'.

* Para checkear si LVM está activo o no haremos uso del comando lsblk, este nos muestra información de todos los dispositivos de bloque (discos duros, SSD, memorias, etc.) entre toda la información que proporciona podemos ver lvm en el tipo de gestor. Para este comando haremos un if, ya que o printaremos Yes o No. Básicamente la condición que buscamos será contar el número de líneas en las que aparece "lvm" y si hay más de 0 printamos Yes, si hay 0 se printará No. Todo el comando sería:

if [$(lsblk | grep "lvm" | wc -l) -gt 0]; then echo yes; else echo no; fi.

* Para mirar el número de conexiones TCP establecidas. Utilizaremos el comando ss sustituyendo al ya obsoleto netstat. Filtraremos con el flag -ta para que solo se muestren las conexiones TCP. Por último haremos un grep para ver las que están establecidas, ya que también hay solo de escucha y cerraremos con wc -l para que cuente el número de líneas. El comando queda tal que así:

ss -ta | grep ESTAB | wc -l.

* Daremos uso del comando users que nos mostrará el nombre de los usuarios que hay, sabiendo esto, pondremos wc -w para que cuente la cantidad de palabras que hay en la salida del comando. El comando entero queda así

users | wc -w.

* Para poder obtener el número de comandos que son ejecutados con sudo haremos uso del comando journalctl que este es una herramienta que se encarga de recopilar y administrar los registros del sistema. Acto seguido pondremos _COMM=sudo par así filtrar las entradas especificando su ruta. En nuestro ponemos _COMM , ya que hace referencia a un script ejecutable. Una vez tengamos filtrada la búsqueda y solo aparezcan los registros de sudo todavía deberemos filtrar un poco más, ya que cuando inicias o cierras sesión de root también aparece en el registro, entonces para terminar de filtrar pondremos un grep COMMAND y asi solo aparecerán las líneas de comandos. Por último pondremos wc -l para que asi nos salgan enumeradas las líneas. El comando entero es el siguiente: journalctl _COMM=sudo | grep COMMAND | wc -l). Para comprobar que funcione correctamente podemos correr el comando en el terminal, poner un comando que incluya sudo y volver a correr el comando y deberá incrementar el número de ejecuciones de sudo.

* Para poder obtener el número de comandos que son ejecutados con sudo haremos uso del comando journalctl que este es una herramienta que se encarga de recopilar y administrar los registros del sistema. Acto seguido pondremos _COMM=sudo par así filtrar las entradas especificando su ruta. En nuestro ponemos _COMM , ya que hace referencia a un script ejecutable. Una vez tengamos filtrada la búsqueda y solo aparezcan los registros de sudo todavía deberemos filtrar un poco más, ya que cuando inicias o cierras sesión de root también aparece en el registro, entonces para terminar de filtrar pondremos un grep COMMAND y asi solo aparecerán las líneas de comandos. Por último pondremos wc -l para que asi nos salgan enumeradas las líneas. El comando entero es el siguiente: journalctl _COMM=sudo | grep COMMAND | wc -l). Para comprobar que funcione correctamente podemos correr el comando en el terminal, poner un comando que incluya sudo y volver a correr el comando y deberá incrementar el número de ejecuciones de sudo.

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 por-

centaje.

• 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.