**BorntobeRoot**

**42 Madrid**

**Iguillen**

**1.Virtualización:**

a) Máquina virtual (VM):

* + Definición: Es un software que simula un ordenador completo dentro de otro ordenador.
  + Funcionamiento: Utiliza recursos del ordenador físico (CPU, RAM, almacenamiento) para crear un ordenador "virtual".
  + Uso: Permite ejecutar diferentes sistemas operativos en un solo ordenador físico.

b) Hipervisor Tipo 2 (VirtualBox):

* + Definición: Software que se instala en un sistema operativo existente para crear y gestionar máquinas virtuales.
  + Funcionamiento: Actúa como intermediario entre el hardware físico y las máquinas virtuales.
  + Uso: VirtualBox es un ejemplo popular, fácil de usar y gratuito.

c) Ventajas de la virtualización:

* + Aislamiento: Cada VM funciona independientemente, sin afectar al sistema anfitrión o a otras VMs.
  + Portabilidad: Las VMs se pueden mover fácilmente entre ordenadores físicos diferentes.
  + Snapshots: Permiten guardar el estado actual de una VM y volver a él si algo sale mal.

1. **Debian:**

a) Sistema operativo basado en Linux:

* + Definición: Distribución de Linux conocida por su estabilidad y seguridad.
  + Características: Gratuito, de código abierto, con una gran comunidad de soporte.

b) Filosofía de software libre:

* + Significado: Compromiso con el uso y distribución de software que respeta las libertades del usuario.
  + Implicaciones: Los usuarios pueden usar, estudiar, modificar y distribuir el software libremente.

c) Estabilidad y seguridad:

* + Ciclo de lanzamiento: Nuevas versiones estables cada 2 años aproximadamente.
  + Actualizaciones: Frecuentes parches de seguridad y correcciones de errores.

d) Gestión de paquetes:

* + APT (Advanced Package Tool): Sistema para instalar, actualizar y eliminar software.
  + Comandos básicos:
    - apt update: Actualiza la lista de paquetes disponibles.
    - apt upgrade: Actualiza todos los paquetes instalados a la última versión.
    - apt install [paquete]: Instala un nuevo paquete.
    - apt remove [paquete]: Elimina un paquete instalado.

1. **Particionamiento:**

a) LVM (Logical Volume Management):

* + Definición: Sistema flexible para gestionar el almacenamiento en disco.
  + Componentes:
    - Volúmenes físicos (PV): Discos duros o particiones reales.
    - Grupos de volúmenes (VG): Agrupación de PVs.
    - Volúmenes lógicos (LV): Particiones virtuales creadas dentro de un VG.
  + Ventajas: Permite redimensionar particiones fácilmente y añadir nuevo almacenamiento sin interrupciones.

b) Comandos LVM básicos:

* + pvcreate [dispositivo]: Crea un volumen físico.
  + vgcreate [nombre\_vg] [pv1] [pv2]...: Crea un grupo de volúmenes.
  + lvcreate -L [tamaño] -n [nombre\_lv] [nombre\_vg]: Crea un volumen lógico.

c) Particiones primarias vs lógicas:

* + Primarias: Máximo 4 por disco duro.
  + Lógicas: Creadas dentro de una partición extendida, sin límite práctico.

d) Esquema de particionamiento recomendado:

* + / (root): Sistema de archivos raíz, contiene todos los demás directorios.
  + /home: Almacena los directorios personales de los usuarios.
  + swap: Espacio usado como memoria virtual cuando la RAM se agota.

1. **Sistema de archivos:**

a) ext4:

* + Definición: Sistema de archivos estándar en la mayoría de las distribuciones Linux.
  + Características:
    - Journaling: Previene la pérdida de datos en caso de apagado inesperado.
    - Soporte para volúmenes grandes (hasta 1 exabyte).
    - Mejor rendimiento que sus predecesores.

b) Comparación con otros sistemas de archivos:

* + NTFS (Windows): Mejor para sistemas Windows, soporta permisos más complejos.
  + FAT32: Compatible con casi todos los sistemas, pero limitado en tamaño de archivo (4GB máximo).

c) Comandos útiles:

* + df -h: Muestra el espacio usado y disponible en los sistemas de archivos montados.
  + du -sh [directorio]: Muestra el tamaño total de un directorio.
  + fsck [dispositivo]: Comprueba y repara un sistema de archivos.

1. **Seguridad:**

a) AppArmor:

* + Definición: Sistema de control de acceso obligatorio que restringe las capacidades de los programas.
  + Funcionamiento: Usa perfiles para definir qué recursos puede usar cada programa.
  + Comandos básicos:
    - aa-status: Muestra el estado de AppArmor.
    - aa-enforce [perfil]: Activa un perfil en modo de aplicación.

b) UFW (Uncomplicated Firewall):

* + Definición: Interfaz simplificada para gestionar iptables (el firewall de Linux).
  + Comandos básicos:
    - ufw enable: Activa el firewall.
    - ufw allow [puerto/servicio]: Permite el tráfico en un puerto específico.
    - ufw deny [puerto/servicio]: Bloquea el tráfico en un puerto específico.
    - ufw status: Muestra el estado actual del firewall.

c) SSH (Secure Shell):

* + Definición: Protocolo para acceder de forma segura a sistemas remotos.
  + Configuración: Archivo principal en /etc/ssh/sshd\_config.
  + Comandos básicos:
    - ssh [usuario]@[ip\_o\_hostname]: Conecta a un sistema remoto.
    - scp [archivo] [usuario]@[ip\_o\_hostname]:[ruta]: Copia archivos de forma segura.

d) Política de contraseñas fuertes:

* + Configuración: Se realiza en /etc/login.defs y /etc/pam.d/common-password.
  + Aspectos: Longitud mínima, complejidad (mayúsculas, minúsculas, números, símbolos), caducidad.

1. **Gestión de usuarios y grupos:**

a) Comandos para gestión de usuarios:

* + useradd [opciones] [nombre\_usuario]: Crea un nuevo usuario.
  + usermod [opciones] [nombre\_usuario]: Modifica un usuario existente.
  + userdel [opciones] [nombre\_usuario]: Elimina un usuario.
  + passwd [nombre\_usuario]: Cambia la contraseña de un usuario.

b) Comandos para gestión de grupos:

* + groupadd [nombre\_grupo]: Crea un nuevo grupo.
  + groupmod [opciones] [nombre\_grupo]: Modifica un grupo existente.
  + groupdel [nombre\_grupo]: Elimina un grupo.
  + usermod -aG [grupo] [usuario]: Añade un usuario a un grupo.

c) Archivos importantes:

* + /etc/passwd: Contiene información básica de los usuarios.
  + /etc/shadow: Almacena las contraseñas encriptadas de los usuarios.
  + /etc/group: Contiene información sobre los grupos.

d) sudo vs su:

* + sudo [comando]: Ejecuta un comando con privilegios de superusuario.
  + su: Cambia al usuario root o a otro usuario.
  + Diferencia principal: sudo permite ejecutar comandos específicos con privilegios elevados, mientras que su cambia completamente al usuario root.

1. **Servicios del sistema:**

a) systemd:

* + Definición: Sistema de inicio y gestor de servicios en Linux moderno.
  + Comandos principales:
    - systemctl start [servicio]: Inicia un servicio.
    - systemctl stop [servicio]: Detiene un servicio.
    - systemctl restart [servicio]: Reinicia un servicio.
    - systemctl status [servicio]: Muestra el estado de un servicio.
    - systemctl enable [servicio]: Configura un servicio para que se inicie al arrancar el sistema.
    - systemctl disable [servicio]: Evita que un servicio se inicie al arrancar el sistema.

b) Ubicación de archivos de servicio:

* + /etc/systemd/system/: Archivos de unidad personalizados.
  + /lib/systemd/system/: Archivos de unidad predeterminados del sistema.

1. **Monitorización del sistema:**

a) Comandos útiles:

* + top: Muestra una vista dinámica de los procesos en ejecución. Uso: Ejecuta top en la terminal. Presiona 'q' para salir.
  + htop: Versión mejorada de top con interfaz más amigable. Uso: Ejecuta htop en la terminal. Usa las flechas para navegar, F10 para salir.
  + free -h: Muestra el uso de memoria en formato legible. Uso: Ejecuta free -h para ver la memoria total, usada y libre.
  + df -h: Muestra el uso del disco en formato legible. Uso: Ejecuta df -h para ver el espacio usado y disponible en todos los sistemas de archivos montados.
  + du -sh [directorio]: Muestra el tamaño total de un directorio. Uso: Ejecuta du -sh /home para ver el tamaño total del directorio home.

b) Logs del sistema:

* + Ubicación principal: /var/log/
  + Archivos importantes:
    - /var/log/syslog: Log general del sistema.
    - /var/log/auth.log: Registros de autenticación.
  + journalctl: Herramienta para consultar logs de systemd. Uso: journalctl -u [servicio] para ver los logs de un servicio específico.

1. **Cron:**

a) Definición: Demonio para ejecutar tareas programadas en intervalos regulares.

b) Uso básico:

* + crontab -e: Edita el archivo crontab del usuario actual.
  + crontab -l: Lista las tareas cron del usuario actual.

c) Sintaxis de crontab:

\* \* \* \* \* comando\_a\_ejecutar

- - - - -

| | | | |

| | | | +----- Día de la semana (0-7) (Domingo = 0 o 7)

| | | +------- Mes (1-12)

| | +--------- Día del mes (1-31)

| +----------- Hora (0-23)

+------------- Minuto (0-59)

d) Ejemplo: 0 2 \* \* \* /scripts/backup.sh Esto ejecutará el script backup.sh todos los días a las 2:00 AM.

1. **Redes:**

a) Configuración de red:

* + Archivo principal: /etc/network/interfaces
  + DHCP vs. Estática:
    - DHCP: Asigna IP automáticamente. Ejemplo de configuración:

auto eth0

iface eth0 inet dhcp

* + - Estática: Requiere configuración manual. Ejemplo:

auto eth0

iface eth0 inet static

address 192.168.1.100

netmask 255.255

**11 Línea de comandos:**

a) Navegación del sistema de archivos:

* + cd [directorio]: Cambia el directorio actual. Ejemplo: cd /home/usuario te lleva al directorio home del usuario.
  + ls [opciones] [directorio]: Lista el contenido de un directorio. Opciones comunes:
    - -l: Formato largo, muestra permisos, propietario, tamaño, etc.
    - -a: Muestra archivos ocultos. Ejemplo: ls -la /etc muestra todos los archivos (incluyendo ocultos) en /etc en formato detallado.
  + pwd: Muestra el directorio de trabajo actual.

b) Gestión de archivos:

* + cp [origen] [destino]: Copia archivos o directorios. Ejemplo: cp archivo.txt /home/usuario/ copia archivo.txt al directorio home del usuario.
  + mv [origen] [destino]: Mueve o renombra archivos o directorios. Ejemplo: mv archivo.txt nuevo\_nombre.txt renombra el archivo.
  + rm [opciones] [archivo]: Elimina archivos o directorios. Opción importante: -r para eliminar directorios y su contenido recursivamente. Ejemplo: rm -r directorio\_viejo elimina el directorio y todo su contenido.
  + mkdir [directorio]: Crea un nuevo directorio. Ejemplo: mkdir proyectos crea un directorio llamado "proyectos".

c) Gestión de paquetes con apt:

* + apt update: Actualiza la lista de paquetes disponibles. Uso: Ejecutar antes de instalar o actualizar paquetes.
  + apt upgrade: Actualiza todos los paquetes instalados a la última versión. Uso: Ejecutar periódicamente para mantener el sistema actualizado.
  + apt install [paquete]: Instala un nuevo paquete. Ejemplo: apt install nginx instala el servidor web Nginx.
  + apt remove [paquete]: Elimina un paquete instalado. Ejemplo: apt remove nginx desinstala Nginx.
  + apt search [término]: Busca paquetes relacionados con el término. Ejemplo: apt search text editor busca editores de texto disponibles.

d) Editores de texto:

* + nano:
    - Más sencillo y amigable para principiantes.
    - Comandos básicos:
      * Ctrl+O: Guardar cambios.
      * Ctrl+X: Salir del editor.
      * Ctrl+W: Buscar texto. Ejemplo de uso: nano /etc/hosts para editar el archivo hosts.
  + vim:
    - Más potente pero con una curva de aprendizaje más pronunciada.
    - Modos principales:
      * Modo normal: Para navegar y ejecutar comandos (se accede con Esc).
      * Modo inserción: Para escribir texto (se accede con 'i').
      * Modo comando: Para guardar, salir, etc. (se accede con ':' desde el modo normal).
    - Comandos básicos en modo comando:
      * :w: Guardar cambios.
      * :q: Salir (si no hay cambios sin guardar).
      * :wq o :x: Guardar y salir. Ejemplo de uso: vim /etc/ssh/sshd\_config para editar la configuración de SSH.

1. **SELinux vs AppArmor:**

**a) SELinux (Security-Enhanced Linux):**

* + Desarrollado por la NSA (Agencia de Seguridad Nacional de EE.UU.).
  + Características:
    - Implementa control de acceso obligatorio (MAC).
    - Más complejo y potente que AppArmor.
    - Utiliza etiquetas de seguridad para archivos y procesos.
  + Modos de operación:
    - Enforcing: Aplica la política de seguridad.
    - Permissive: Registra violaciones pero no las previene.
    - Disabled: SELinux está completamente desactivado.
  + Comandos básicos:
    - getenforce: Muestra el modo actual de SELinux.
    - setenforce [0|1]: Cambia entre modo Permissive (0) y Enforcing (1).
    - sestatus: Muestra el estado detallado de SELinux.

**b) AppArmor:**

* + Desarrollado por Canonical (la empresa detrás de Ubuntu).
  + Características:
    - Más fácil de configurar y mantener que SELinux.
    - Utiliza perfiles basados en rutas para restringir aplicaciones.
    - Menos granular pero más intuitivo que SELinux.
  + Modos de operación:
    - Enforce: Aplica las restricciones definidas en los perfiles.
    - Complain: Registra las violaciones pero no las previene.
  + Comandos básicos:
    - aa-status: Muestra el estado de AppArmor y los perfiles cargados.
    - aa-enforce [perfil]: Cambia un perfil a modo Enforce.
    - aa-complain [perfil]: Cambia un perfil a modo Complain.

c) Comparación:

* + SELinux ofrece un control más fino y es preferido en entornos de alta seguridad.
  + AppArmor es más fácil de aprender y configurar, ideal para usuarios menos experimentados.
  + SELinux es el estándar en distribuciones como Red Hat y Fedora.
  + AppArmor es el predeterminado en Ubuntu y se usa en Debian.

d) Razones por las que Debian usa AppArmor por defecto:

* + Facilidad de uso y configuración.
  + Menor impacto en el rendimiento del sistema.
  + Mejor integración con las herramientas y filosofía de Debian.
  + Curva de aprendizaje más suave para administradores de sistemas

Preguntas

**Diferencia entre Rocky y Debian:** Rocky Linux es un sistema operativo de código abierto diseñado como reemplazo de CentOS, compatible con Red Hat Enterprise Linux. Debian es una distribución Linux independiente, conocida por su estabilidad y amplio soporte de arquitecturas. Rocky se enfoca en entornos empresariales, mientras Debian es más versátil y se usa en diversos escenarios.

1. **Propósito de la máquina virtual y por qué la usamos:** El propósito de la máquina virtual en este contexto es proporcionar un entorno aislado y controlado para aprender y practicar la administración de sistemas Linux. La usamos porque permite experimentar de forma segura sin afectar el sistema host, facilita la creación de snapshots y la restauración del sistema, y simula un entorno de servidor real.
2. **Por qué elegimos ese sistema operativo:** Si elegimos Debian, es probablemente por su estabilidad, amplia documentación, y porque es una excelente opción para aprender los fundamentos de la administración de sistemas Linux. Además, Debian es muy versátil y se adapta bien a diferentes roles, desde servidores hasta estaciones de trabajo.
3. **Diferencia entre aptitude y apt (si elegimos Debian):** Ambos son gestores de paquetes para Debian, pero aptitude ofrece una interfaz más avanzada y maneja mejor las dependencias. apt es más rápido y simple, ideal para operaciones básicas. aptitude tiene una interfaz de texto interactiva, mientras que apt se usa principalmente desde la línea de comandos.
4. **Qué es AppArmor:** AppArmor es un sistema de seguridad para Linux que restringe las capacidades de los programas mediante perfiles. Ayuda a proteger el sistema contra vulnerabilidades conocidas y desconocidas, limitando los recursos a los que un programa puede acceder.
5. **Cómo funciona una máquina virtual:** Una máquina virtual funciona creando un entorno virtualizado que emula hardware físico. Utiliza software llamado hipervisor para asignar recursos del hardware físico (CPU, RAM, almacenamiento) a la máquina virtual. El sistema operativo invitado se ejecuta en este entorno virtualizado como si estuviera en hardware real, permitiendo la ejecución de aplicaciones y servicios de forma aislada del sistema host.

¿Qué es una máquina virtual? Una máquina virtual es un software que simula un sistema de computación y puede ejecutar programas como si fuera una computadora real. Permite crear un entorno aislado dentro de un sistema físico, con su propio sistema operativo y recursos asignados.

1. **¿Para qué sirve?** Las máquinas virtuales sirven para ejecutar múltiples sistemas operativos en un solo hardware, aislar aplicaciones, realizar pruebas de software en diferentes entornos, y mejorar la seguridad al separar servicios en diferentes máquinas virtuales.
2. ¿**Qué es LVM?** LVM (Logical Volume Manager) es un sistema de administración de almacenamiento que proporciona una capa de abstracción entre el almacenamiento físico y el sistema de archivos. Permite una gestión más flexible de los volúmenes de almacenamiento.
3. ¿**Qué es AppArmor?** AppArmor es un sistema de seguridad para Linux que restringe las capacidades de los programas mediante perfiles. Ayuda a proteger el sistema contra vulnerabilidades conocidas y desconocidas, limitando los recursos a los que un programa puede acceder.
4. **¿Qué es SSH?** SSH (Secure Shell) es un protocolo de red que permite una comunicación segura y encriptada entre dos sistemas. Se utiliza principalmente para acceder y administrar sistemas remotos de forma segura.
5. **¿Qué significa HTTP?** HTTP significa Hypertext Transfer Protocol. Es el protocolo de comunicación que permite las transferencias de información en la World Wide Web, principalmente utilizado para la carga de páginas web.
6. **¿Qué significa HTTPS?** HTTPS es la versión segura de HTTP (Hypertext Transfer Protocol Secure). Utiliza SSL/TLS para cifrar la comunicación entre el cliente y el servidor, proporcionando una capa adicional de seguridad para la transferencia de datos.
7. ¿**Qué es UFW?** UFW (Uncomplicated Firewall) es una interfaz simplificada para iptables que facilita la configuración de un firewall en sistemas Linux. Permite gestionar fácilmente las reglas de firewall para controlar el tráfico de red.

**¿Qué significa TCP?** TCP (Transmission Control Protocol) es un protocolo de comunicación fundamental en Internet que proporciona una transmisión de datos confiable y ordenada entre aplicaciones en dispositivos en red.

1. **¿Qué significa IP? I**P (Internet Protocol) es el protocolo principal de comunicación en Internet para enviar datos a través de una red. Define cómo se direccionan y enrutan los paquetes entre computadoras.
2. **¿Qué es un puerto?** Un puerto es un punto final de comunicación en un sistema operativo. Los puertos se identifican con números y permiten que diferentes servicios en un mismo dispositivo se comuniquen a través de la red sin interferir entre sí.
3. **¿Qué es un Firewall?** Un Firewall es un sistema de seguridad de red que monitorea y controla el tráfico entrante y saliente basándose en reglas predeterminadas. Actúa como una barrera entre redes confiables e no confiables.
4. **¿Qué significa DNS?** DNS (Domain Name System) es un sistema que traduce nombres de dominio legibles por humanos (como [www.ejemplo.com](http://www.ejemplo.com/)) a direcciones IP numéricas que las computadoras utilizan para identificarse en la red.
5. **¿Qué es cron?** Cron es un administrador de tareas en sistemas Unix que permite a los usuarios programar la ejecución automática de comandos o scripts en fechas y horas específicas o en intervalos regulares.