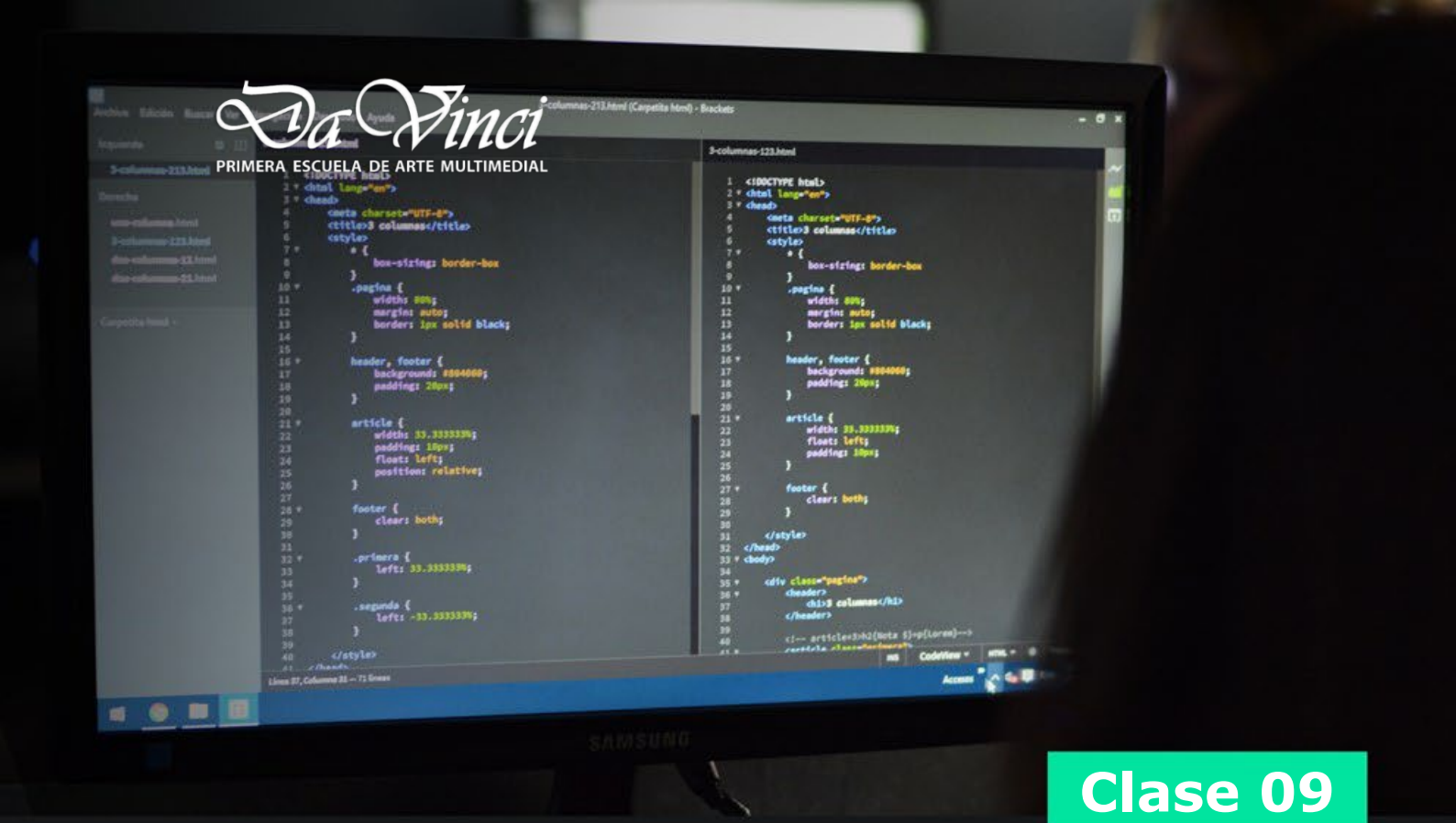


Da Vinci
PRIMERA ESCUELA DE ARTE MULTIMEDIAL

Clase 09

Diseño y Programación Web

Materia:
Sistemas Operativos

Docente contenidista: CARLASSARA, Fabrizio

Revisión: Coordinación

Contenido

Clase 9

¡Te damos la bienvenida a la materia **Sistemas Operativos**!

En esta clase vamos a ver los siguientes temas:

En esta clase vamos a comenzar a ver cómo administrar ciertos servicios de forma remota. Entre otras cosas veremos:

- Opciones para administración remota.
- Servidores y clientes SSH.
- Servidores y clientes FTP.
- Administración remota con Webmin.

Gestión remota

La gestión remota de un sistema Linux implica administrar y controlar un servidor o una computadora Linux desde otro dispositivo ubicado en un lugar diferente. Esto puede incluir tareas como monitoreo del sistema, instalación de software, configuración de servicios, resolución de problemas, y más, todo realizado a través de una conexión a Internet.

Hay varias opciones para realizar la gestión remota de un sistema Linux, estas son algunas:

- **SSH (Secure Shell):** SSH es una herramienta de acceso remoto estándar en sistemas Unix/Linux. Permite acceder de forma segura a la línea de comandos de un sistema Linux desde otro dispositivo. Con SSH, puedes ejecutar comandos, transferir archivos y administrar servicios de forma remota. Herramientas como OpenSSH son comunes en la mayoría de las distribuciones de Linux.
- **VNC (Virtual Network Computing):** VNC es una tecnología que permite ver y controlar la interfaz gráfica de un sistema Linux de forma remota. Con VNC, puedes interactuar con la interfaz de usuario como si estuvieras físicamente frente al sistema. Herramientas como TigerVNC y TightVNC son populares para esta tarea.
- **Gestión Web:** Algunas distribuciones de Linux, como Ubuntu Server, vienen con herramientas de gestión web integradas, como Webmin o Cockpit. Estas interfaces basadas en web proporcionan paneles de control para administrar varios aspectos del sistema, como configuración de red, administración de usuarios, monitoreo de recursos, entre otros.
- **Servicios de Administración Remota:** Hay varios servicios y herramientas diseñadas específicamente para la administración remota de sistemas Linux, como Ansible, Puppet y Chef. Estas herramientas permiten automatizar tareas de administración de sistemas en múltiples servidores Linux de forma remota, facilitando la gestión de grandes entornos.
- **Soluciones Propietarias:** Además de las opciones de código abierto mencionadas, también existen soluciones comerciales para la gestión remota de sistemas Linux. Estas pueden ofrecer características avanzadas y soporte técnico especializado, pero a menudo vienen con un costo asociado.

En esta clase vamos a tomar solo algunas de estas soluciones y ver cómo usarlas en nuestro CentOS.

SSH - Secure Shell

SSH, que significa "Secure Shell", es un protocolo de red que permite a los usuarios conectarse y administrar de forma segura un sistema remoto a través de una red no segura, como Internet. Proporciona un método seguro para acceder a la línea de comandos de un sistema Linux o Unix, permitiendo ejecutar comandos, transferir archivos y realizar otras tareas de administración.

Algunos conceptos más detallados de SSH incluyen:

- **Autenticación:** SSH utiliza un sistema de autenticación basado en clave pública/privada para verificar la identidad del usuario y el servidor. Cada usuario tiene un par de claves: una clave pública que se almacena en el servidor remoto y una clave privada que se mantiene en su propio dispositivo. Cuando un usuario intenta conectarse al servidor remoto, SSH utiliza estas claves para verificar la identidad del usuario.
- **Cifrado:** SSH cifra toda la comunicación entre el cliente y el servidor, lo que protege los datos confidenciales de ser interceptados por terceros. Utiliza algoritmos de cifrado como AES, 3DES y Blowfish para garantizar la confidencialidad de los datos transmitidos.
- **Integridad de los datos:** Además del cifrado, SSH también verifica la integridad de los datos para detectar cualquier manipulación durante la transmisión. Esto se logra utilizando algoritmos de hashing como SHA-1 o SHA-256 para firmar digitalmente los datos transmitidos.
- **Puerto estándar:** Por defecto, SSH utiliza el puerto 22 para las conexiones, aunque este puede ser cambiado a otro puerto para mejorar la seguridad. Esto ayuda a proteger contra escaneos automáticos de puertos por parte de posibles atacantes.
- **Flexibilidad:** SSH es una herramienta extremadamente versátil que se puede utilizar para una variedad de propósitos. Además de acceder a la línea de comandos de un sistema remoto, también se puede utilizar para transferir archivos de forma segura (a través de SFTP o SCP), reenviar puertos para acceder a servicios internos de una red y realizar túneles VPN para asegurar la comunicación entre dos puntos.

Instalar SSH en CentOS 9

Antes de instalar nuevas aplicaciones, es una buena práctica actualizar el sistema para asegurarte de tener los últimos paquetes y correcciones de seguridad. Podemos hacerlo ejecutando en la terminal:

```
sudo dnf update
```

En CentOS y otras distribuciones basadas en RHEL, el servidor SSH se proporciona a través del paquete openssh-server. Puede instalarse ejecutando el siguiente comando:

```
sudo dnf install openssh-server
```

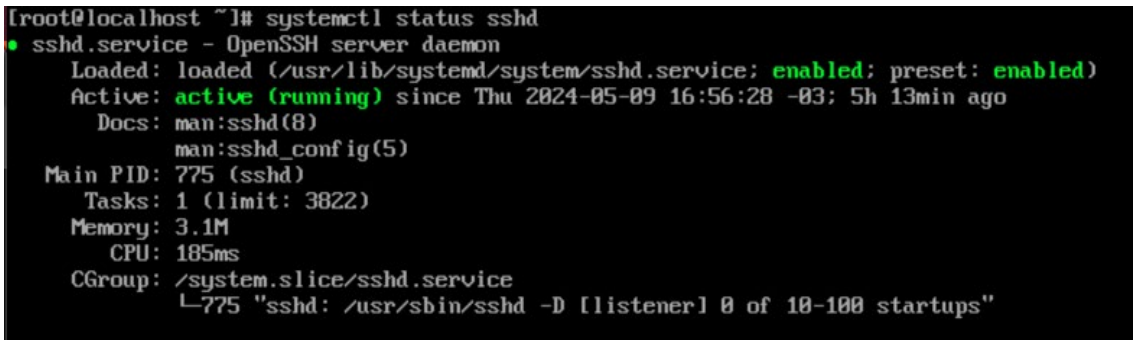
Después de instalar el paquete, necesitamos habilitar el servicio SSH para que se inicie automáticamente en el arranque del sistema y luego iniciarlo para que esté disponible de inmediato. Podemos hacerlo con los siguientes comandos:

```
sudo systemctl enable sshd
sudo systemctl start sshd
```

Con estos pasos, SSH debería estar instalado y en funcionamiento en nuestro sistema CentOS. Podemos verificar que esto sea así corriendo el comando:

```
sudo systemctl status sshd
```

Deberíamos ver un resultado como el siguiente:



Para asegurarnos que el firewall no bloquee el tráfico de datos en el puerto 22 que va a usar el SSH, podemos escribir lo siguiente:

```
sudo firewall-cmd --zone=public --add-service=ssh --permanent
sudo firewall-cmd --reload
```

Cliente de SSH

El método puede cambiar dependiendo de la herramienta con la que contemos, ya sea una computadora con Windows, Linux o MacOS o incluso un celular.

Desde computadoras con Linux o MacOS el proceso es sencillo, solamente tenemos que instalar un cliente de SSH que podemos correr desde nuestra terminal. El comando para instalarlo cambia ligeramente de acuerdo a la distribución.

En el caso de computadoras con Windows o celulares, tendremos que descargar alguna aplicación adicional. En el caso de clientes para celular no se va a trabajar especialmente en esta clase pero puede hacerse descargando una aplicación, como Termius que también está disponible para computadoras.

Linux de distribuciones con Ubuntu/Debian

En nuestra terminal, escribimos:

```
sudo apt update
sudo apt install openssh-client
```

Linux de distribuciones con CentOS/RHEL

En nuestra terminal, escribimos:

```
sudo yum install openssh-clients
```

Linux de distribuciones con Fedora

En nuestra terminal, escribimos:

```
sudo dnf install openssh-clients
```

macOS

Si no lo tenemos instalado, abrir nuestra terminal e instalar homebrew:

```
/bin/bash -c "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.
```

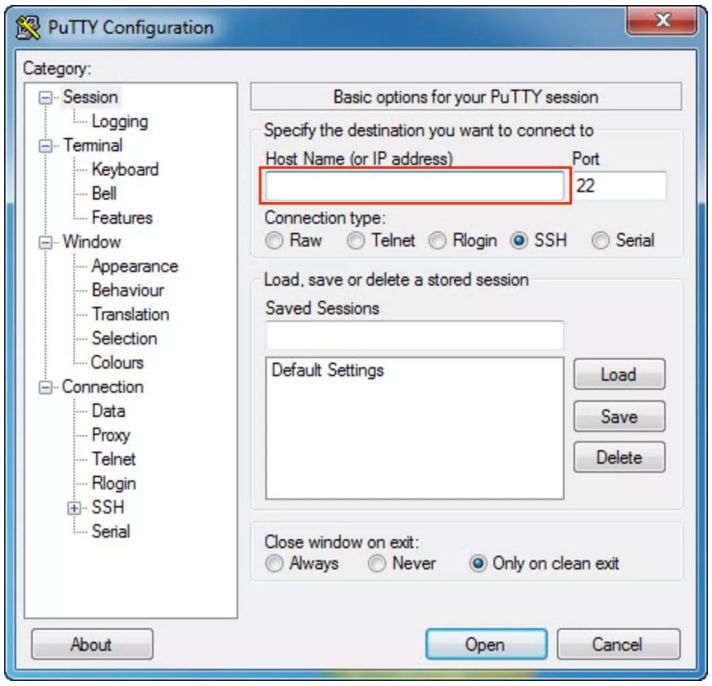
```
sh)"
```

Luego instalamos el cliente de SSH:

```
brew install openssh
```

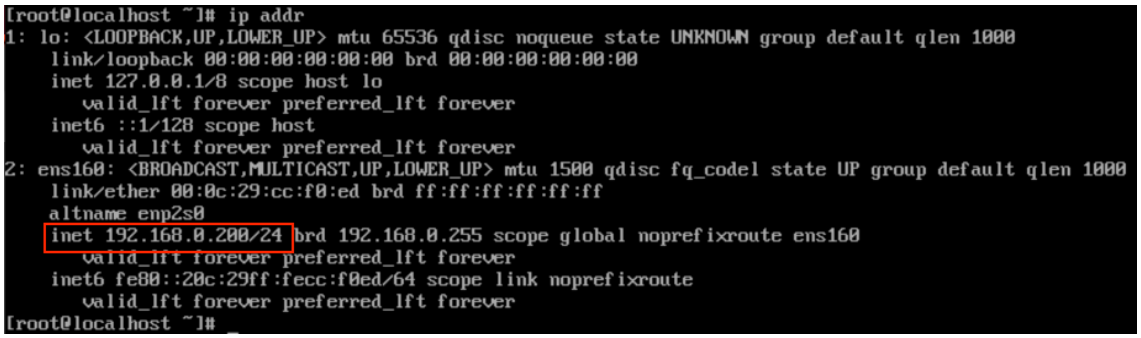
Windows

En este caso, tendremos que descargar un cliente de SSH como PuTTY desde este link e instalarlo normalmente. Al abrirlo, tendremos que escribir en el campo resaltado la IP del servidor SSH. Tendremos que asegurarnos que el puerto sea el 22 y el tipo de conexión sea SSH.



Establecer conexión con el servidor

Con cualquiera de los clientes que hayamos escogido o tengamos disponibles, vamos a tener que encontrar la dirección de IP de nuestro servidor. Esto es posible entrando a nuestra máquina virtual y escribiendo el comando: **ip addr**. El resultado de este comando es uno como el siguiente:



En este caso, por ejemplo, la dirección de IP es 192.168.0.200. Si no apareciera nada o si apareciera una dirección con el primer octeto en 10, tendremos que configurar la red correctamente como vimos en la clase anterior.

Cuando tengamos la dirección de IP, vamos a escribir en la terminal (para computadoras con Linux y macOS) el comando **ssh IP**. En el caso de computadoras con Windows que usen PuTTY, solamente tenemos que escribir la IP y abrir la conexión.

En cualquier caso, vamos a recibir un mensaje pidiéndonos que indiquemos el usuario con el que queremos loguearnos, luego de lo cual va a pedirnos una contraseña. El usuario y contraseña deben corresponder a uno que exista en la máquina virtual.

Una vez hecho esto y si las credenciales fueron correctas, vamos a estar dentro de nuestra máquina virtual con el usuario que elegimos. Ahora podemos interactuar con nuestra máquina virtual como si estuviéramos físicamente frente a ella. Cuando queramos salir, simplemente escribimos **exit**.

Servicio de FTP

Un servidor FTP, o Protocolo de Transferencia de Archivos (File Transfer Protocol, por sus siglas en inglés), es un tipo de servidor utilizado para el intercambio de archivos a través de una red, como Internet. Funciona como un sistema de almacenamiento centralizado al que múltiples usuarios pueden acceder para cargar (subir) y descargar archivos.

Algunas cualidades incluyen:

Aquí tienes una descripción más detallada de cómo funciona:

- **Acceso remoto:** Un servidor FTP permite a los usuarios acceder a los archivos almacenados en él desde cualquier lugar del mundo, siempre y cuando tengan las credenciales adecuadas (nombre de usuario y contraseña).
- **Subida y bajada de archivos:** Los usuarios pueden cargar archivos en el servidor FTP (subir) o descargar archivos del servidor FTP a su propio dispositivo (bajar). Esto es útil para compartir archivos grandes, como documentos, imágenes, vídeos, etc.
- **Estructura de directorios:** Al igual que en un sistema de archivos convencional, un servidor FTP organiza los archivos en una estructura de directorios y subdirectorios. Los usuarios pueden navegar por esta estructura para encontrar los archivos que necesitan.
- **Seguridad:** Los servidores FTP suelen incluir medidas de seguridad para proteger los archivos y la información del acceso no autorizado. Esto puede incluir autenticación de usuarios mediante contraseñas, encriptación de datos durante la transferencia (FTP seguro o FTPS) y control de acceso basado en permisos.
- **Gestión de usuarios:** Los administradores del servidor FTP pueden crear cuentas de usuario individuales y asignarles diferentes niveles de acceso y permisos. Esto les permite controlar quién puede acceder a qué archivos y realizar qué acciones.

Instalación de servidor FTP

Instalar un servidor FTP en nuestro CentOS es un proceso relativamente sencillo. Ahora vamos a instalar y configurar un servidor FTP utilizando vsftpd, que es uno de los servidores FTP más populares en entornos Linux.

Antes de instalar cualquier software, es una buena práctica asegurarse de que tu sistema esté actualizado. Podemos hacerlo ejecutando los siguientes comandos en la terminal:

```
sudo dnf update
```

Luego escribimos este comando para instalar el servidor vsftpd:

```
sudo dnf install vsftpd
```

Una vez que vsftpd esté instalado, podríamos configurarlo según nuestras necesidades. El archivo de configuración principal de vsftpd se encuentra en */etc/vsftpd/vsftpd.conf*. Podemos editarlo con cualquier editor de texto pero en este caso, vamos a dejarlo como está.

Luego, iniciamos el servicio vsftpd y lo habilitamos para que se inicie automáticamente en el arranque utilizando los siguientes comandos:

```
sudo systemctl start vsftpd
sudo systemctl enable vsftpd
```

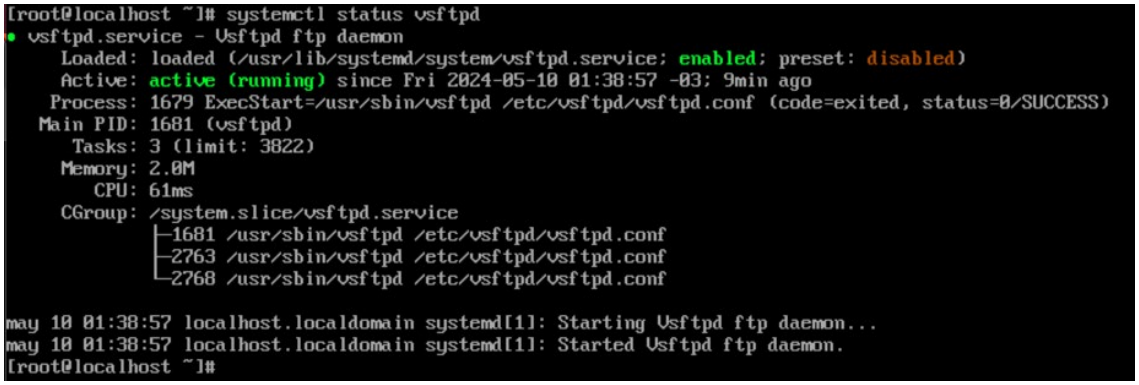
Para que no moleste el firewall, vamos a abrir el puerto 21 para permitir el tráfico entrante y saliente:

```
sudo firewall-cmd --zone=public --add-port=21/tcp --permanent
sudo firewall-cmd --reload
```

Podemos verificar que está correctamente habilitado usando:

```
sudo systemctl status vsftpd
```

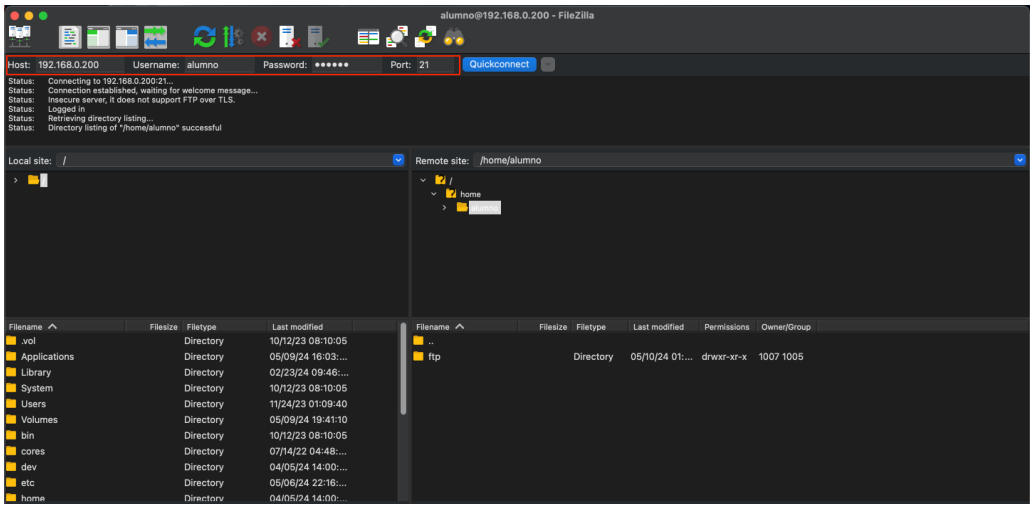
Donde tendríamos que ver un resultado como este:



Cliente de FTP

Algunos navegadores tienen soporte para funcionar como clientes FTP y hay además numerosas aplicaciones para elegir. En esta clase, vamos a hacer uso de FileZilla ya que es bastante sencilla e intuitiva de usar. Puede descargarse de este link y usarse en múltiples plataformas.

Una vez que lo tengamos, podemos abrirlo y conectarnos con nuestro servidor usando la IP que vimos cómo obtener anteriormente y un usuario y contraseña que exista en nuestra máquina virtual.



Una vez que nos conectamos, podremos ver a nuestra izquierda el sistema de archivos de nuestra computadora y a la derecha el sistema de archivos del servidor. A partir de aquí, podemos transferir archivos, copiarlos, crear directorios y todo lo que necesitemos entre ambas computadoras.

Cuando hayamos terminado, podemos desconectarnos del servidor con la misma interfaz.

Webmin

Webmin es una herramienta de administración de sistemas basada en web que permite a los administradores de sistemas Unix y Linux gestionar de forma remota sus sistemas a través de una interfaz gráfica de usuario. Con Webmin, los administradores pueden realizar una amplia gama de tareas de administración del sistema, como la configuración del sistema, la administración de usuarios y grupos, la configuración del servidor web, la gestión de bases de datos, la configuración del firewall, la programación de tareas, entre otras cosas.

La interfaz de Webmin es accesible a través de un navegador web estándar y proporciona una forma intuitiva de realizar tareas de administración del sistema sin necesidad de acceder directamente a la línea de comandos. Además, Webmin es altamente personalizable y extensible, lo que permite a los usuarios agregar módulos

adicionales para admitir nuevas funcionalidades o adaptar la interfaz a sus necesidades específicas.

Instalación de Webmin

De la documentación de Webmin, podemos ver que nos sugieren estos dos pasos:

```
curl -o setup-repos.sh
https://raw.githubusercontent.com/webmin/webmin/master/setup-repos.sh
sh setup-repos.sh
```

Esto va a configurar el repositorio y conseguirnos las claves para que la instalación sea más sencilla. Luego, podemos instalarlo con:

```
sudo dnf install webmin
```

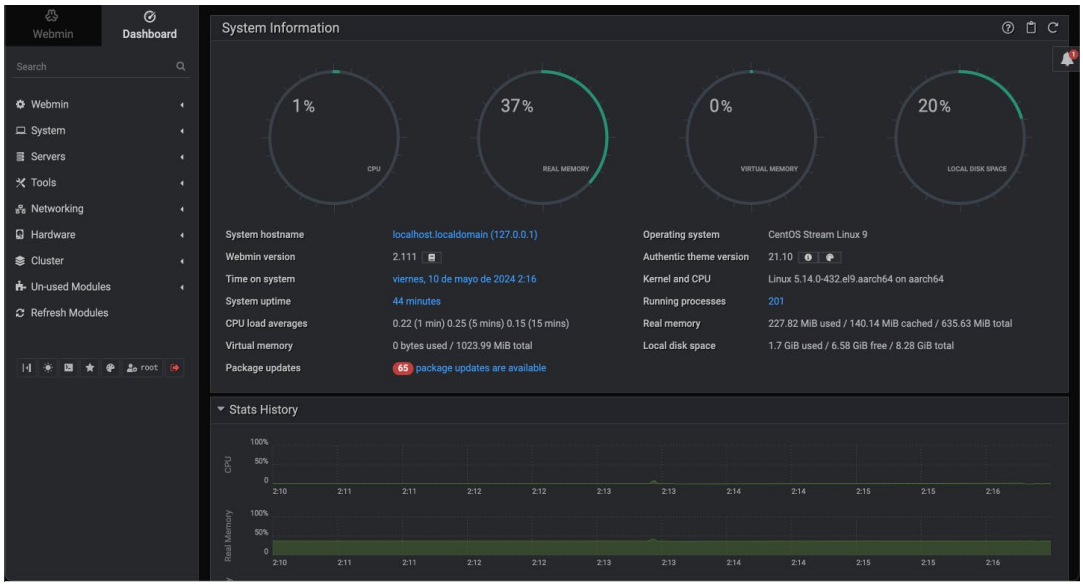
Podemos verificar que el servicio ya está corriendo con el comando:

```
sudo systemctl status webmin
```

Uso de Webmin

Podemos acceder a Webmin desde nuestro navegador ingresando la IP de nuestro servidor y el puerto 10000. Por ejemplo, en el caso que venimos analizando sería *https://192.168.0.200:10000*.

Cuando ingresemos, veremos una interfaz muy completa con una variedad de opciones para ver y controlar.



Algunas cosas que podemos hacer en esta interfaz incluyen:

- **Monitoreo del sistema:** La opción de monitoreo del sistema se encuentra disponible en el “Dashboard”. Aquí podremos ver información sobre el uso de recursos del sistema, como CPU, memoria y espacio en disco, así como acceder a registros del sistema para identificar problemas.
- **Procesos:** Podemos ver los procesos que están corriendo en el sistema en la sección de “Procesos” dentro de “Sistema”.
- **Administración de usuarios y grupos:** Podemos encontrar esta opción en el menú "Usuarios y grupos" dentro de la sección "Sistema". Aquí podremos crear, modificar y eliminar usuarios y grupos, así como gestionar sus permisos.
- **Configuración del sistema:** En la sección "Hardware" y "Sistema", encontrarás opciones para ajustar la configuración del sistema, como la configuración de red, la fecha y hora del sistema, y los servicios que se inician automáticamente.
- **Administración de archivos:** Podemos acceder a la administración de archivos desde la sección "Explorador de archivos" en “Herramientas”. Desde aquí podremos navegar por el sistema de archivos, crear, copiar, eliminar y cambiar permisos de archivos y directorios.
- **Estado de servicios:** En la sección “Estados de servidor y sistema” podemos ver el estado de distintos servicios y si se encuentran instalados o no. Tenemos la posibilidad de instalarlos desde esta opción.
- **Configuración de red:** En “Interfaces” dentro de “Red” podemos elegir y configurar distintas interfaces de red disponibles.
- **Instalación de módulos:** Hay muchos módulos comunes que podemos ver en la sección de módulos sin usar e instalar desde ahí.

Hemos llegado así al final de esta clase en la que vimos:

- Opciones para administración remota.
- Servidores y clientes SSH.
- Servidores y clientes FTP.
- Administración remota con Webmin.

Muchas de estas opciones nos van a proporcionar interfaces más amigables para trabajar con nuestros servidores y otras opciones para conectarnos a este cuando no tengamos acceso físico a él. Conocer estas herramientas puede ayudarnos a tener más flexibilidad a la hora de trabajar con servidores.

La próxima clase seguiremos viendo más servicios comunes de Linux.

Te recomendamos que puedas realizar el desafío semanal para que puedas verificar lo que aprendiste.

Estaremos en contacto ante cualquier consulta que tengas.

¡Hasta la próxima clase!

Te esperamos en la **clase en vivo** de esta semana.
No olvides realizar el **desafío semanal**.

¡Hasta la próxima clase!

Bibliografía

Eckert, J. W. (2020). Linux+ and LPIC-1: Guide to Linux Certification. Cengage.

Para ampliar la información

Webmin. (s. f.). Recuperado de https://webmin.com/

FileZilla. (s. f.). Recuperado de https://filezilla-project.org/

Hostinger. (s. f.). ¿Qué es el protocolo SSH y cómo funciona? En Hostinger. Recuperado de https://www.hostinger.com.ar/tutoriales/que-es-ssh#:~:text=SSH%20o%20Secure%20Shell%2C%20es,de%20un%20mecanismo%20de%20autenticaci%C3%B3n.

OpenSSH. (s. f.). Recuperado de https://www.openssh.com/

RedesZone. (s. f.). vsftpd: instalación y configuración. En RedesZone. Recuperado de https://www.redeszone.net/tutoriales/servidores/vsftpd-configuracion-servidor-ftp/