



INSTITUTO POLITECNICO NACIONAL

**UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERIAS Y
TECNOLOGIAS AVAN ZADA**

PROYECTO FINAL

ADMINISTRACION DE SISTEMAS OPPERATIVOS

PRESENTA

PALACIOS REYES LESLIE NOEMI

NUMERO DE BAOLETA:2022640128

Primera etapa

```
148.204.88.107 - PuTTY
login as: 
```

```
aso@aso-HVM-domU:~$
login as: aso
aso@148.204.88.107's password:
Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-37-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Se pueden aplicar 3 actualizaciones de forma inmediata.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

*** System restart required ***
Last login: Thu Jun 16 12:54:59 2022 from 187.191.38.210
aso@aso-HVM-domU:~$
```

```
aso@aso-HVM-domU:~$
login as: aso
aso@148.204.88.107's password:
Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-37-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Se pueden aplicar 3 actualizaciones de forma inmediata.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

*** System restart required ***
Last login: Thu Jun 16 12:54:59 2022 from 187.191.38.210
aso@aso-HVM-domU:~$ sudo adduser
[sudo] contraseña para aso:
sudo: se requiere una contraseña
aso@aso-HVM-domU:~$ sudo adduser leslie
[sudo] contraseña para aso:
Añadiendo el usuario 'leslie' ...
Añadiendo el nuevo grupo 'leslie' (1015) ...
Añadiendo el nuevo usuario 'leslie' (1015) con grupo 'leslie' ...
Creando el directorio personal '/home/leslie' ...
Copiando los ficheros desde '/etc/skel' ...
Nueva contraseña:
Vuelva a escribir la nueva contraseña:
passwd: contraseña actualizada correctamente
Cambiano la información de usuario para leslie
Introduzca el nuevo valor, o presione INTRO para el predeterminado
Nombre completo []: leslie mooni palacios reyes
Número de habitación []: 17M1
Teléfono del trabajo []:
Teléfono de casa []:
Otro []: 2022640128
¿Es correcta la información? [S/n]
```

```
aso@aso-HVM-domU:~$
login as: leslie
leslie@148.204.88.107's password:
Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-37-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Se pueden aplicar 3 actualizaciones de forma inmediata.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

*** System restart required ***

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

leslie@aso-HVM-domU:~$ mkdir public_html
leslie@aso-HVM-domU:~$ ls
public_html  snap
leslie@aso-HVM-domU:~$
```

```
leslie@aso-HVM-domU:~$
login as: leslie
leslie@148.204.88.107's password:
Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-37-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Se pueden aplicar 3 actualizaciones de forma inmediata.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

*** System restart required ***

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

leslie@aso-HVM-domU:~$ mkdir public_html
leslie@aso-HVM-domU:~$ ls
public_html  snap
leslie@aso-HVM-domU:~$
```

```
leslie@aso-HVM-domU:~$
login as: leslie
leslie@148.204.88.107's password:
Welcome to Ubuntu 22.04 LTS (GNU/Linux 5.15.0-37-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Se pueden aplicar 3 actualizaciones de forma inmediata.
Para ver estas actualizaciones adicionales, ejecute: apt list --upgradable

*** System restart required ***

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

leslie@aso-HVM-domU:~$ mkdir public_html
leslie@aso-HVM-domU:~$ ls
public_html  snap
leslie@aso-HVM-domU:~$
```

Segunda etapa

```
leslie@aso-HVM-domU: /home
*** System restart required ***

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

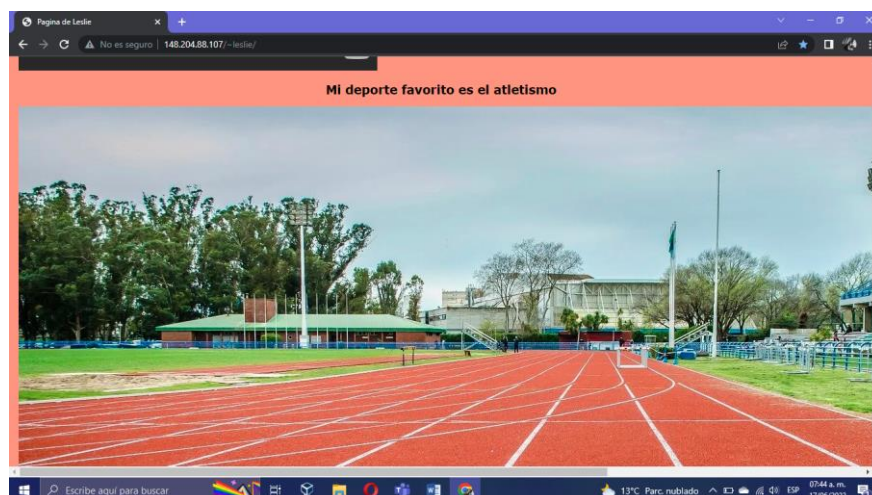
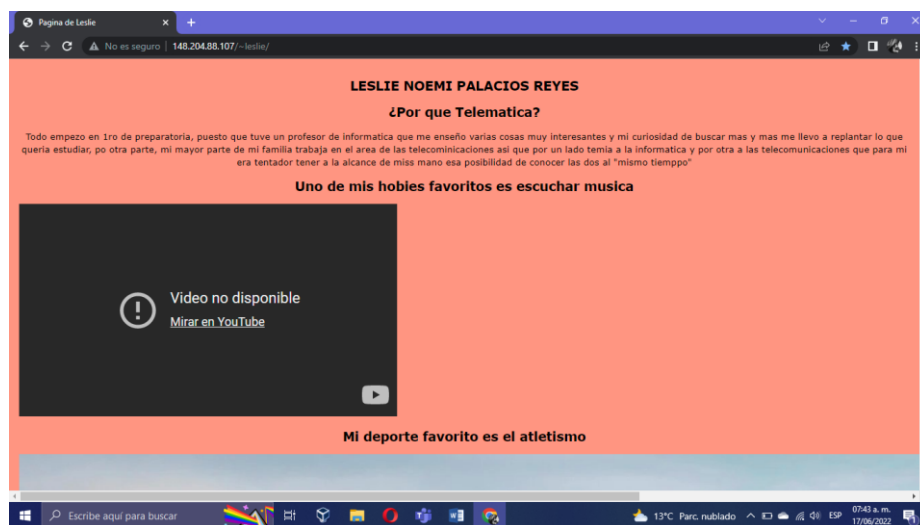
leslie@aso-HVM-domU:~$ mkdir public_html
leslie@aso-HVM-domU:~$ ls
public_html  snap
leslie@aso-HVM-domU:~$ cd ..
leslie@aso-HVM-domU:~/home$ ls
alanog    axelrm    erikbp    ivan      kronos    robert
alejandroas betoxm    fabricioep jessicacv leslie    rodolfojoen
aleydimc  brenda    fernanda  kaos      maferlr
aso       claudiaaa giovannigm karlarr   marioja
leslie@aso-HVM-domU:~/home$ chmod
chmod: falta un operando
Pruebe 'chmod --help' para más información.
leslie@aso-HVM-domU:~/home$ chmod 755 leslie
leslie@aso-HVM-domU:~/home$
```

```
leslie@aso-HVM-domU: ~/public_html

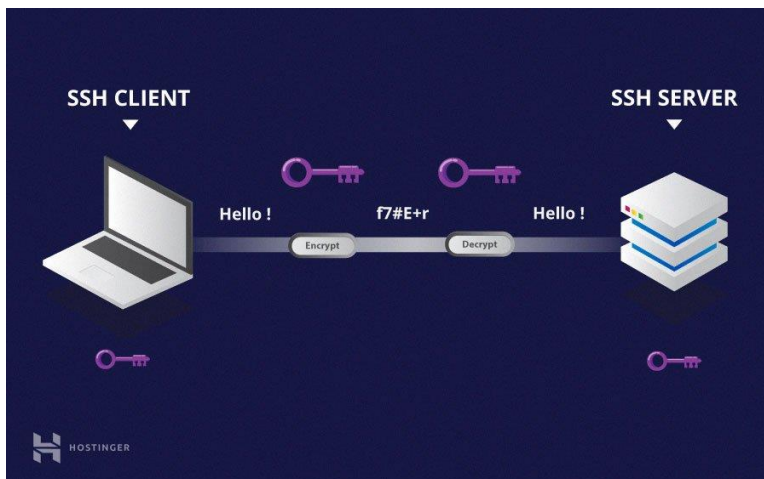
set number
syntax on
colorscheme blue
set autoindent
set tabstop=2

filetype indent plugin on

~/vimrc" [Nuevo] 8L, 96B escritos 8,0-1 Todo
```



Protocolo de SSH



El protocolo SSH se usa para proteger servicios de red en una red no segura. Actualmente se usa en casi todos los centros de datos. Secure Shell utiliza una arquitectura cliente-servidor para proporcionar un canal seguro dentro de una red no segura. Según lo define la Internet Society en el documento que presenta la arquitectura del protocolo Secure Shell (SSH). Este

protocolo de red criptográfico se usa para proteger todo tipo de servicios de red. Este utiliza la encriptación para proteger la conexión entre el cliente y el servidor SSH; lo cual protege frente a ataques en la red. SSH permite acceder a líneas de comandos, ejecutar comandos, iniciar sesión y realizar tareas de sysadmin de forma remota y segura. Además, el protocolo SSH también se usa en diversos mecanismos de transferencia de archivos. Por ejemplo:

- SFTP (*SSH File Transfer Protocol*; una alternativa segura a [FTP](#)).
- FASP (*Fast and Secure Protocol*).
- SCP (*Secure copy*).

La encriptación del protocolo tiene como objetivo proporcionar una fuerte integridad y confidencialidad de la información. SSH utiliza criptografía de clave pública para su mecanismo de autenticación —conocido como “autenticación de clave pública”—. Además, también es compatible con la autenticación basada en contraseñas.

Principales métodos para usar SSH

1. Usar pares de claves pública/privada generados automáticamente para encriptar una conexión de red e iniciar sesión usando una contraseña.
2. Usar pares de claves pública/privada generados manualmente para realizar la autenticación. De este modo los usuarios y los programas pueden iniciar sesión sin usar contraseña.

Antes de ser validadas, las claves públicas desconocidas siempre deben verificarse en todas las versiones de SSH; para evitar autorizar como usuario válido a un atacante no autorizado.

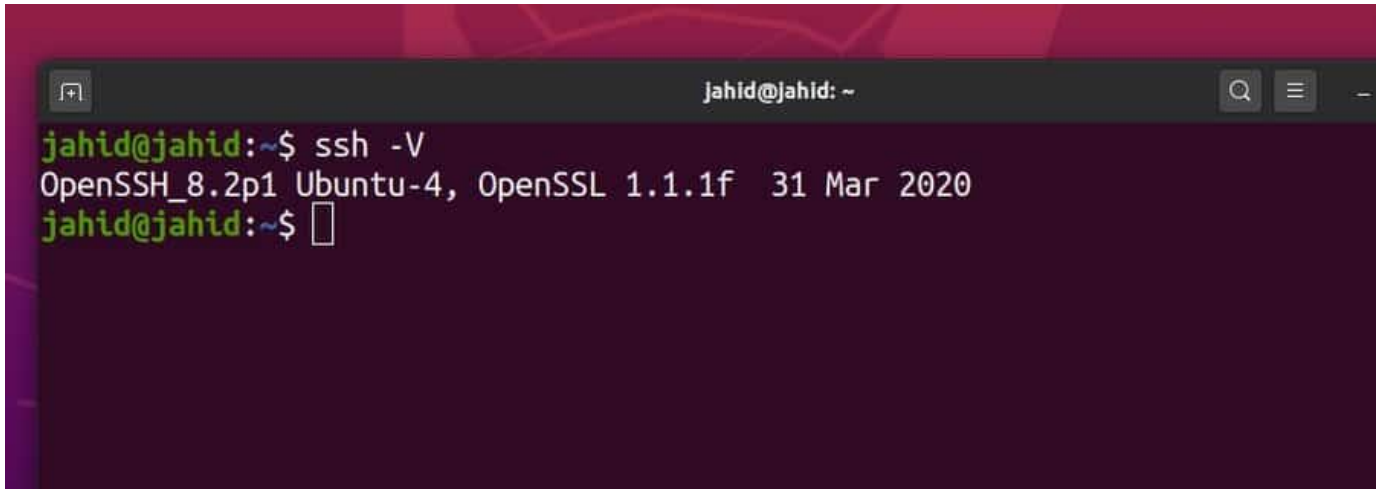
SSH y cloud computing

En cloud computing, SSH es útil para resolver problemas de conectividad y evitar problemas de seguridad. Un túnel SSH puede proporcionar una ruta segura en Internet, a través de un firewall, evitando la exposición de las máquinas virtuales directamente en Internet.

Configuración en Linux del servidor ssh.

El servicio SSH se instala de forma predeterminada en todos los sistemas operativos Linux o similares a Unix. Puede verificar si el servicio SSH está instalado dentro de su máquina Linux o no verificando la versión SSH. Si encuentra que su Ubuntu tiene un SSH instalado, está listo para comenzar. Si no puede encontrar el servicio de shell seguro en su Ubuntu Linux, puede instalarlo con el comando apt-get install.

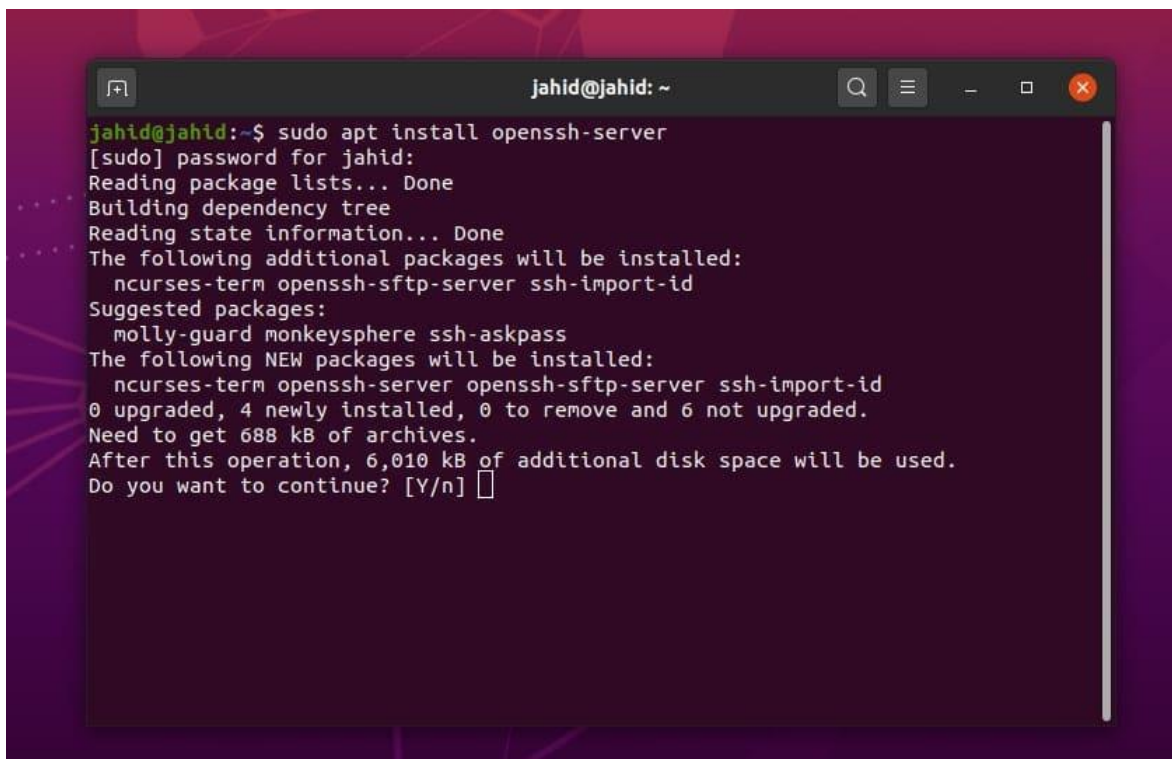
\$ ssh -V

A screenshot of a terminal window with a dark background and light green text. The window title bar shows 'jahid@jahid: ~'. The terminal content shows the command 'ssh -V' being executed, followed by the output 'OpenSSH_8.2p1 Ubuntu-4, OpenSSL 1.1.1f 31 Mar 2020'. The prompt 'jahid@jahid:~\$' is visible at the end of the line.

```
jahid@jahid:~$ ssh -V
OpenSSH_8.2p1 Ubuntu-4, OpenSSL 1.1.1f 31 Mar 2020
jahid@jahid:~$
```

Antes de instalar cualquier paquete, debe actualizar y actualizar el repositorio de Ubuntu. Luego instale el paquete Openssh Server con el comando terminal shell. Todas las líneas de comando de la terminal se dan a continuación.

```
$ sudo apt update
$ sudo apt upgrade
$ sudo apt install op
```

A terminal window titled 'jahid@jahid: ~' with search, menu, and window control icons. The terminal shows the command 'sudo apt install openssh-server' and its output. The output indicates that several additional packages will be installed along with the requested package. It shows the disk space requirements and asks for confirmation to continue.

```
jahid@jahid:~$ sudo apt install openssh-server
[sudo] password for jahid:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere ssh-askpass
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 upgraded, 4 newly installed, 0 to remove and 6 not upgraded.
Need to get 688 kB of archives.
After this operation, 6,010 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

enssh-server

2. **Habilitación de SSH en Red Hat Linux y CentOS**

Red Hat Linux se creó principalmente para la distribución de cargas de trabajo, la utilización de la nube y la ventana acoplable, y el propósito de la evolución. Aquí están las líneas de comando de la terminal para verificar, instalar, iniciar y habilitar el servicio SSH en Red Hat Linux. Al igual que otras distribuciones de Linux, Red Hat también usa el puerto 22 para establecer el servicio SSH. Es posible que también deba permitir el acceso al firewall para el servicio SSH en CentOS y Red Hat Linux.

\$	dnf	install	openssh-server
\$	yum	install	openssh-server
\$	systemctl	start	sshd
\$	systemctl	status	sshd
\$	systemctl	enable	sshd

firewall-cmd --zone=public --permanent --add-service=ssh

3. **Habilitación de SSH en Arch Linux**

Arch Linux usa el comando del administrador de paquetes (packman) para instalar cualquier aplicación. Primero, necesita actualizar el repositorio del sistema de Arch Linux. Luego puede

instalar el servicio OpenSSH en Arch Linux a través de los comandos packman. Puede iniciar o detener cualquier servicio SSH, verificar el estado de SSH y deshabilitar el servicio SSH en Arch Linux usando el systemctl comando de terminal.

```
$ sudo pacman -Sy openssh
$ sudo pacman -S openssh
$ sudo systemctl status sshd
$ sudo systemctl start sshd
$ sudo systemctl status sshd
$ sudo systemctl stop sshd
$ sudo systemctl enable sshd
$ sudo systemctl disable sshd
$ sudo systemctl restart sshd
```

Para configurar el script del servicio SSH en Arch Linux, debe abrir el archivo de configuración desde el /etc/ssh/ directorio.

```
$ man sshd_config / config files
$ sudo nano /etc/ssh/sshd_config
```

4. Habilitando SSH en Fedora Linux

Antes de instalar el servicio SSH en Fedora Linux, verifiquemos si el servicio SSH ya está instalado dentro de la máquina o no. Usaremos el comando de terminal grep para verificar la disponibilidad del servicio SSH en Fedora Linux. Fedora Linux también usa el puerto 22 para establecer conexiones de shell seguras.

Además, podemos verificar el estado total del servicio SSH usando el systemctl comando en el terminal shell. Además de estos, puede iniciar, detener, habilitar y deshabilitar el shell seguro en Fedora Linux usando las líneas de comando de terminal que se proporcionan a continuación.

```
$ rpm -qa | grep openssh-server
$ sudo dnf install -y openssh-server;
$ sudo systemctl status sshd
$ sudo ss -lt
$ sudo systemctl start sshd.service;
$ sudo systemctl stop sshd.service;
$ sudo systemctl disable sshd.service;
```

Algunos	comandos	principales	del	servicio	SSH
---------	----------	-------------	-----	----------	-----

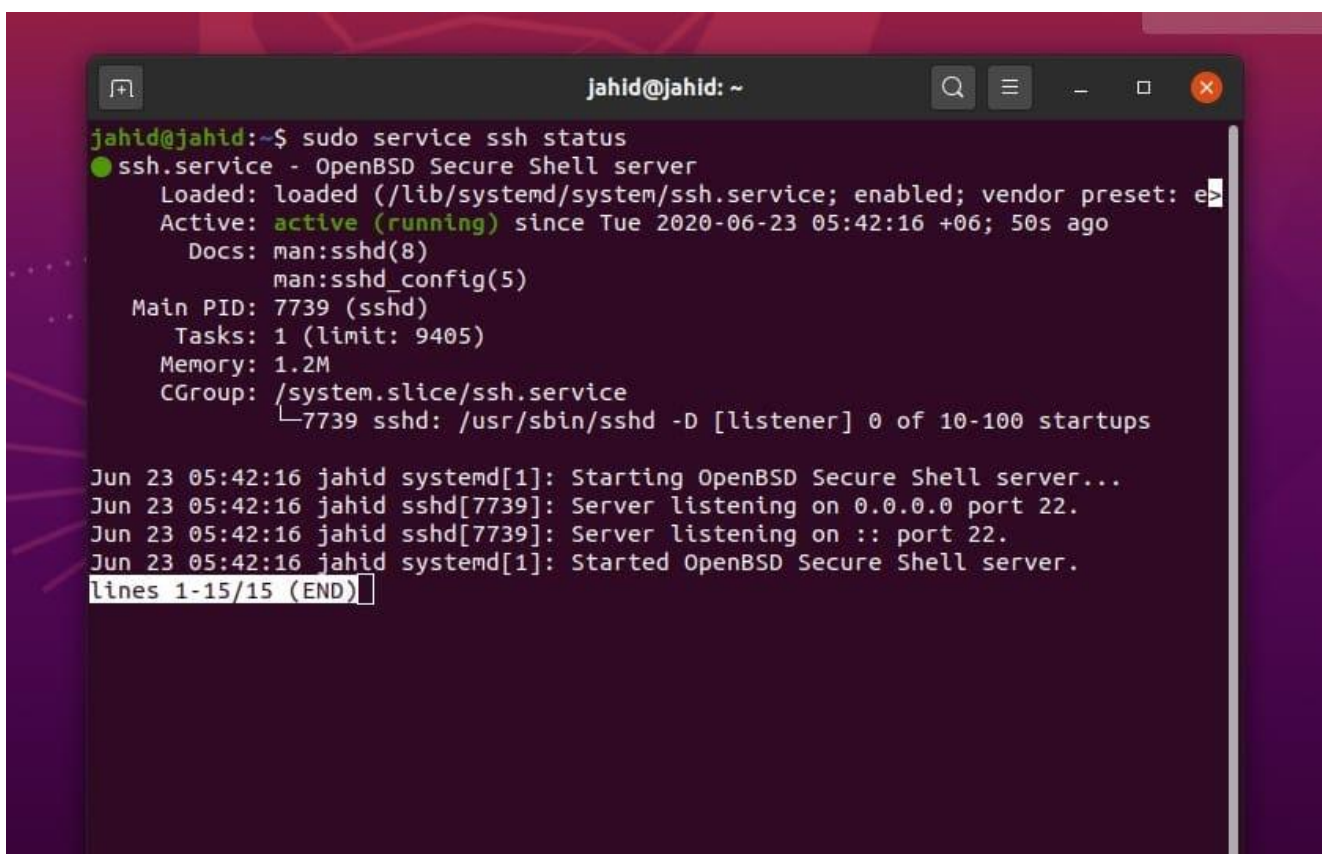
Hasta ahora, hemos pasado por el proceso de cómo habilitar y configurar el servicio SSH en distribuciones de Linux. Ahora veremos cómo ejecutar algunos comandos básicos del servicio SSH en Linux. Aquí, mostraré las reglas principales para establecer un servicio seguro, obtener acceso al

firewall y reenvío de túneles en Linux. Una vez que conozca los fenómenos fundamentales del servicio SSH, podrá habilitar y configurar otros servicios SSH por su cuenta.

Tarea 1: Comandos básicos del servicio SSH en Linux

Una vez que el servicio SSH está instalado dentro de su máquina Linux, ahora puede verificar el estado del sistema, habilitar el servicio SSH y comenzar con el sistema de shell seguro. Aquí, se dan algunos comandos SSH básicos. También puede apagar el sistema SSH si no lo necesita.

\$	sudo	systemctl	status	ssh
\$	sudo	service	ssh	status
\$	sudo	systemctl	enable	ssh
\$	sudo	systemctl	start	ssh
\$ sudo systemctl stop ssh				

A screenshot of a terminal window titled 'jahid@jahid: ~'. The user has entered the command 'sudo service ssh status'. The output shows that the 'ssh.service' is 'active (running)'. It provides details such as the loaded file path, active status since June 23, 2020, at 05:42:16, documentation files, main PID (7739), tasks, memory usage, and CGroup. At the bottom, there are log messages showing the service starting and listening on port 22.

```
jahid@jahid:~$ sudo service ssh status
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: en
   Active: active (running) since Tue 2020-06-23 05:42:16 +06; 50s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 7739 (sshd)
      Tasks: 1 (limit: 9405)
     Memory: 1.2M
    CGroup: /system.slice/ssh.service
            └─7739 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups

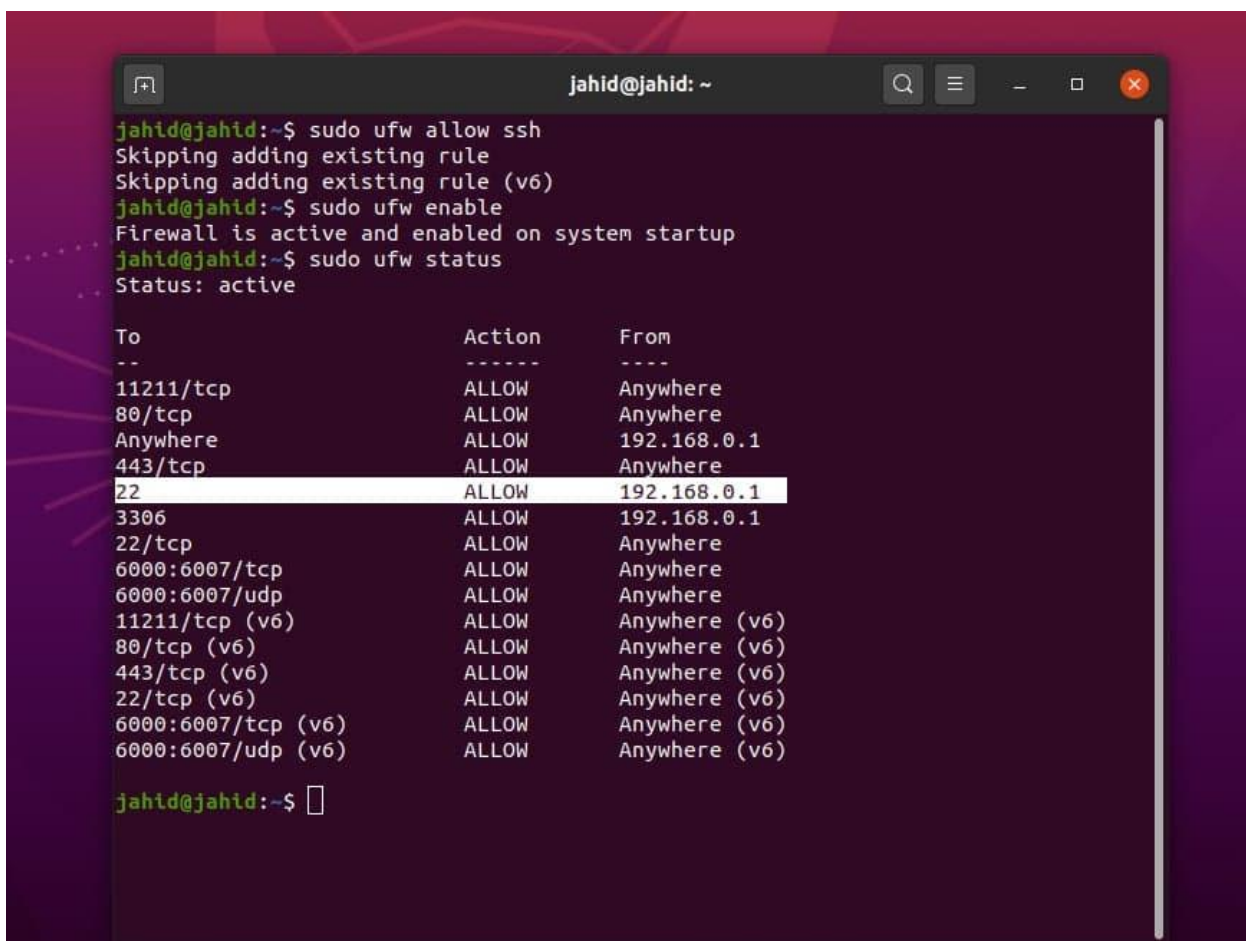
Jun 23 05:42:16 jahid systemd[1]: Starting OpenBSD Secure Shell server...
Jun 23 05:42:16 jahid sshd[7739]: Server listening on 0.0.0.0 port 22.
Jun 23 05:42:16 jahid sshd[7739]: Server listening on :: port 22.
Jun 23 05:42:16 jahid systemd[1]: Started OpenBSD Secure Shell server.
lines 1-15/15 (END)
```

Tarea 2: Obtener acceso al cortafuegos para el servicio SSH

Cuando se trata de un protocolo de transferencia de Internet, necesita obtener acceso al firewall. De lo contrario, el firewall puede bloquear e interrumpir su conexión. Aquí, estoy usando el sistema de firewall UFW para configurar el servicio SSH en Linux. Después de habilitar el firewall UFW, ahora

puede verificar el estado del firewall. El sistema de firewall monitoreará todas las redes entrantes y salientes de su dispositivo.

```
$ sudo ufw allow ssh
$ sudo ufw enable
$ sudo ufw status
```



```
jahid@jahid:~$ sudo ufw allow ssh
Skipping adding existing rule
Skipping adding existing rule (v6)
jahid@jahid:~$ sudo ufw enable
Firewall is active and enabled on system startup
jahid@jahid:~$ sudo ufw status
Status: active

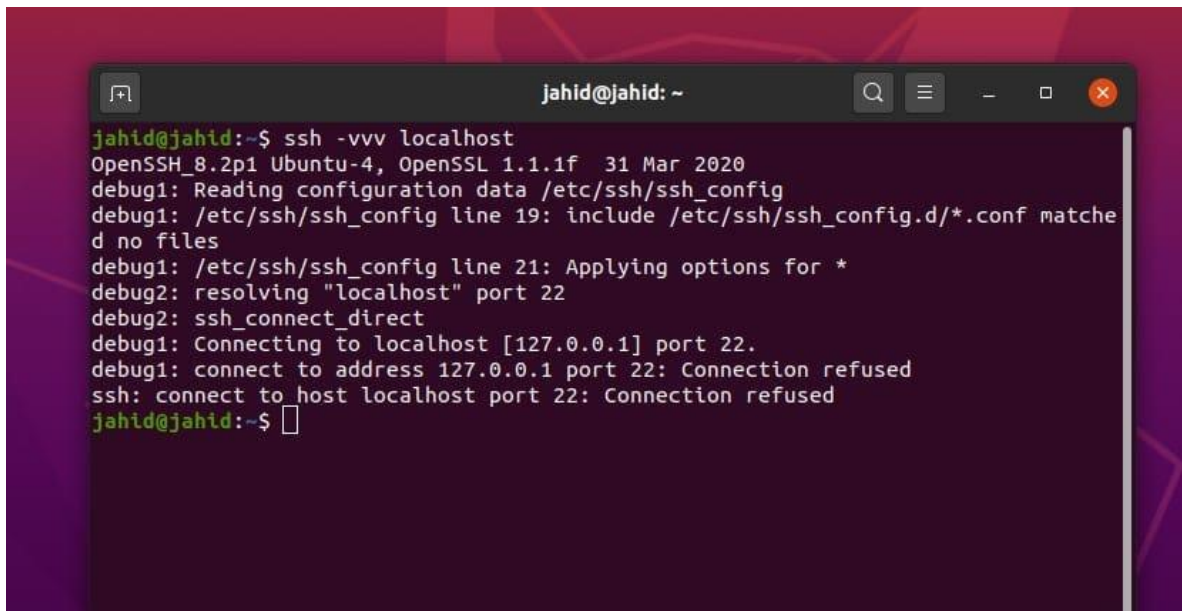
To Action From
--
11211/tcp ALLOW Anywhere
80/tcp ALLOW Anywhere
Anywhere ALLOW 192.168.0.1
443/tcp ALLOW Anywhere
22 ALLOW 192.168.0.1
3306 ALLOW 192.168.0.1
22/tcp ALLOW Anywhere
6000:6007/tcp ALLOW Anywhere
6000:6007/udp ALLOW Anywhere
11211/tcp (v6) ALLOW Anywhere (v6)
80/tcp (v6) ALLOW Anywhere (v6)
443/tcp (v6) ALLOW Anywhere (v6)
22/tcp (v6) ALLOW Anywhere (v6)
6000:6007/tcp (v6) ALLOW Anywhere (v6)
6000:6007/udp (v6) ALLOW Anywhere (v6)

jahid@jahid:~$
```

Tarea 3: Conexión a una IP específica a través del servicio SSH en Linux

Si está utilizando una dirección IP estática para la conexión de shell segura, puede establecer reglas para la red y el puerto específicos. Para el servicio SSH, el puerto predeterminado es 22. Puede cambiar el puerto si es necesario. Ejecutaremos el `vvv` comando para verificar y configurar el protocolo SSH contra una dirección IP específica. En mi caso, estoy intentando conectar el localhost a la red. Aquellos que no saben cómo obtener una red de host local en Linux pueden ver los procedimientos de cómo instalar el servidor Apache en Linux.

```
$ vvv-ssh
$ ssh -vvv localhost
```

A terminal window titled 'jahid@jahid: ~' showing an SSH connection attempt. The user enters 'ssh -vvv localhost'. The output shows debug messages: 'OpenSSH_8.2p1 Ubuntu-4, OpenSSL 1.1.1f 31 Mar 2020', 'debug1: Reading configuration data /etc/ssh/ssh_config', 'debug1: /etc/ssh/ssh_config line 19: include /etc/ssh/ssh_config.d/*.conf matched no files', 'debug1: /etc/ssh/ssh_config line 21: Applying options for *', 'debug2: resolving "localhost" port 22', 'debug2: ssh_connect_direct', 'debug1: Connecting to localhost [127.0.0.1] port 22.', 'debug1: connect to address 127.0.0.1 port 22: Connection refused', 'ssh: connect to host localhost port 22: Connection refused'. The prompt returns to 'jahid@jahid:~\$'.

Déjele saber su nombre de usuario y la dirección IP estática, ahora puede construir una red de shell segura desde su máquina Linux a otro dispositivo. Si no conoce su nombre de usuario, puede seguir el comando de terminal que se proporciona a continuación.

\$ whoami

Para conectarse a su dirección de host local, use estas líneas de comando de terminal en su shell de terminal de Linux. He demostrado varios métodos para acceder al servicio SSH con un nombre de usuario y una dirección IP específicos.

```
$ ssh [email protected]_address
$ ssh [email protected]
$ sss [email protected]
```

A terminal window titled 'jahid@jahid: ~' with standard window controls. The prompt is 'jahid@jahid:~\$'. The user has executed 'ssh jahid@localhost'. The terminal shows the password prompt, a welcome message for Ubuntu 20.04 LTS (GNU/Linux 5.4.0-37-generic x86_64), and links for documentation, management, and support. It then reports 6 updates available, including security updates, and suggests running 'apt list --upgradable'. It also mentions HWE support until April 2025 and the Ubuntu warranty disclaimer. The prompt returns to 'jahid@jahid:~\$'.

Para encontrar su dirección IP, puede utilizar los comandos básicos de la herramienta de red para obtener los detalles de su protocolo de Internet. Y ahora, supongo que conoce tanto su dirección IP como su nombre de usuario. Este es el método para conectarse a una dirección IP específica. Puedo mencionar que también puede conectarse a una dirección IP pública a través de un servicio SSH en Linux.

```
$ ip a
$ ifconfig
$ ssh [email protected]
$ ssh [email protected]_ip_address
```

Configuración mínima para garantizar la seguridad de la conexión.

Hay una serie de medidas de **seguridad para proteger los servidores** que resulta interesante conocer. Es importante plantear las necesidades, tanto actuales como futuras, de la compañía para conocer cuáles son las **medidas de seguridad que se deben implantar**.

Llaves SSH

Se conocen como **llaves SSH** a dos llaves criptográficas que permiten a los usuarios autenticarse para acceder a un servidor SSH. Se trata de una alternativa más segura a las tradicionales contraseñas. El par de llaves criptográficas está formado por una pública y una privada: mientras que la primera puede ser compartida con otros usuarios, la segunda mantiene la identidad del usuario de forma secreta.

La configuración de las llaves SSH es muy sencilla. Tan sólo hay que colocar la

llave pública que identifica al usuario en un directorio determinado ubicado dentro del servidor. Una vez el usuario se conecta al servidor en cuestión, debe hacer uso de la llave privada para acceder a él.

Cortafuegos

Otra de las grandes medidas de seguridad servidor web es el **cortafuegos**. Se trata de una pieza de software, o en ocasiones de hardware, que controla todos los servicios que de algún modo están expuestos a la red. Esto es, el cortafuegos se encarga de bloquear el acceso a todos los puertos a excepción de aquellos habilitados para el público.

El cortafuegos es una parte fundamental a la hora de configurar un servidor. Así, al reducir el software expuesto, se minimizan los puntos por los que el servidor puede ser atacado.

IDS

IDS se refiere a las siglas de Intrusion Detection System, lo que nos da una idea de en qué consiste este sistema. Sirve para detectar cualquier tipo de intrusión en el servidor. En función de la política de seguridad que establezca la empresa, realiza una serie de acciones, como envío de alertas indicando el número IP del ordenador desde el que se ha producido la conexión.

VPN

La **Red Privada Virtual (o VPN)** por sus siglas en inglés, permite crear conexiones seguras entre ordenadores remotos. Se trata por tanto de crear una red privada local. De este modo las empresas pueden configurar sus servicios del mismo modo que si estuviesen en una red privada, además de conectar servidores de un modo seguro.

Por supuesto, resulta mucho más conveniente implantar redes privadas en vez de públicas para las comunicaciones internas.

IPS

La herramienta **IPS** resulta muy efectiva para aumentar la seguridad servidor. Trabaja de manera conjunta con el IDS. Cuando la compañía tiene constancia de que se ha producido una conexión no autorizada al servidor, el IPS solicita al cortafuegos que impida el acceso a todos los puertos desde la IP del intruso.

Ambientes aislados de ejecución

Los **ambientes aislados de ejecución** son un método utilizado para que un determinado componente individual se ejecute en el interior de un espacio específicamente dedicado a él. Para ello, en ocasiones es necesario separar los componentes de una aplicación en servidores propios para cada uno de ellos. El

nivel de aislamiento depende en gran medida de los requerimientos de la propia aplicación, así como de las condiciones que ofrece la infraestructura.

Sin lugar a dudas, aislar los procesos en ambientes individuales de ejecución aumenta la capacidad para aislar cualquier tipo de problema de seguridad que se presente.

Encriptación SSL / TLS

Para mejorar la seguridad servidor, esta es una de las mejores soluciones en el ámbito corporativo. Los **certificados SSL o TLS** permiten autenticar diferentes entidades entre sí. Una vez la autenticación se ha realizado, también se pueden utilizar para establecer comunicaciones encriptadas y seguras.

Hardening

Se puede traducir como “**endurecimiento**”. Son una serie de prácticas que minimizan vulnerabilidades en el servidor. También permite impedir que salga root como usuario en el proceso de autenticación o limitar el acceso a determinados usuarios.

Auditoría de servicio

Independientemente del resto de medidas implantadas para mejorar seguridad servidor, la **auditoría de servicio** para comprobar seguridad servidor es imprescindible. Consiste en un proceso exhaustivo que está dirigido a comprobar cuáles son los servicios que se están ejecutando en los diferentes servidores, así como cuáles son los puertos que se utilizan para la comunicación y qué protocolos se aceptan. Información de gran valor para configurar de forma adecuada los parámetros del cortafuegos.

Seguir unas prácticas adecuadas y verificar seguridad servidor de forma periódica son dos pautas clave para minimizar el riesgo de sufrir cualquier tipo de problema de seguridad.

Diferencia del comando **adduser** a **useradd**

Adduser	Useradd
dduser es un script en perl que utiliza el binario useradd .	useradd es un comando que ejecuta un binario del sistema

userdel vs **deluser**

A la hora de gestionar los usuarios, tan importante es saber crearlos como saber eliminarlos. Ambos comandos sirven para borrar usuarios. Y al igual que **useradd** y **adduser**: el comando **userdel** es un fichero binario, mientras que **deluser** es un script en perl que usa el binario **userdel**.

Servidor apache

Installation Manager instala y configura HTTP Server de Apache como servidor web para Build Forge. El uso del HTTP Server de Apache proporcionado es el modo más rápido de configurar un servidor web para Build Forge.

Como alternativa a la configuración estándar, puede configurar un HTTP Server de Apache existente en lugar de uno instalado y configurado por Build Forge. Las instrucciones proporcionadas suponen que tiene experiencia instalando y configurando HTTP Server de Apache en el sistema operativo.

Para usar HTTP Server de Apache, modifique la instalación de la siguiente forma:

1. Modifique el archivo de configuración de Apache HTTP Server (httpd-vhosts.conf) para que señale a la aplicación Build Forge.
2. Instale PHP y configure los módulos PHP necesarios para HTTP Server de Apache, la base de datos de Build Forge y el cifrado de contraseña, si desea utilizar esta función de seguridad.
3. Configurar Apache para la base de datos.

Instalar Build Forge mediante Installation Manager

En Installation Manager, en la página Configuración de aplicación y servidor web, seleccione Sí en el indicador Proporcionar su propio servidor web.

Software de requisito previo

- Apache HTTP Server 2.2.4
- PHP 5.2.4

Edite el archivo de configuración del servidor de Apache

1. Localice el archivo http-vhosts.conf de Apache en el directorio extras de la instalación del servidor.
2. `cd <apache-dir>/conf/extras/`

vi httpd-vhosts.conf

3. Edite el archivo http-vhosts.conf de Apache. Para añadir información sobre Build Forge a httpd-vhosts.conf, añada las siguientes líneas:
4. `<VirtualHost *:80>`
5. `ServerAdmin build@sudominio.com`
6. `DocumentRoot /opt/buildforge/webroot/public`

7. ServerName ausbuild01.sudominio.com
8. ServerAlias build.sudominio.com mc.sudominio.com
9. ErrorLog logs/ausbuild.error_log
10. CustomLog logs/ausbuild.access_log common

</VirtualHost>

11. Modifique el valor de DocumentRoot para que señale la aplicación web de Build Forge. En este ejemplo, el directorio de instalación de Build Forge es /opt/buildforge.
12. Deje el puerto como 80 o cámbielo al puerto en el que se ejecute Apache HTTP Server localmente.

<VirtualHost *:80>

Importante: No utilice el puerto 8080; es el puerto predeterminado para Apache Tomcat.

13. Modifique cualquier otro valor de http-vhosts.conf según corresponda para Apache HTTP Server:
 - ServerAdmin: dirección de correo electrónico del administrador de Build Forge
 - DocumentRoot: ubicación de la página de entrada para la aplicación Build Forge
 - ServerName: servidor donde está instalada la aplicación Build Forge
 - ServerAlias: alias opcionales para el URL ServerName de Build Forge
 - ErrorLog: registro de errores de Apache para la aplicación Build Forge
 - CustomLog: registro de errores de Apache para registrar el acceso a la aplicación Build Forge

Instalar y configurar PHP para Apache HTTP Server

PHP no se instala con HTTP Server de Apache. Debe instalar PHP 5.2.4 y configurarlo para que señale el archivo httpd-vhosts.conf para Apache HTTP Server.

Instalar y configurar PHP para la base de datos de Build Forge

Durante la instalación de PHP, seleccione e instale las extensiones de PHP para el tipo de base de datos que utilice como base de datos de Build Forge.

(Opcional) Configure el módulo OpenSSL de PHP para dar soporte al cifrado de contraseña

Para dar soporte a SSL, Build Forge utiliza el módulo OpenSSL de PHP. Este soporte se proporciona con PHP 5.2.4; no se necesita configuración adicional.

Para dar soporte al cifrado de contraseña, se requiere configuración adicional. Se necesita PHP 5.2.4 para dar soporte a esta configuración. Debe localizar los archivos de parche para la extensión OpenSSL, instalarlos en el directorio de OpenSSL y recompilar PHP, de la siguiente forma:

1. Localice los archivos de parche `php_openssl.h` y `openssl.c` en el directorio `misc`, ubicado en el directorio de instalación de Build Forge, por ejemplo:

Windows	C:\Archivos de programa\IBM\Build Forge\misc
UNIX/Linux	/opt/buildforge/Platform/misc

2. Copie los archivos de parche en el directorio `openssl`, ubicado en el directorio de instalación de Build Forge.
3. Compile PHP utilizando la opción de configuración `--with-openssl=<vía_acceso_a_openssl>`, donde `<vía_acceso_a_openssl>` es el directorio `openssl` de Build Forge.

Configurar Apache para la base de datos

Necesita añadir información específica a `httpd.conf`, dependiendo de la base de datos.

Configuración de Apache para DB2

1. Añada la siguiente línea al principio del script de inicio de Apache (normalmente `/etc/init.d/httpd` o `/etc/init.d/apache2`, dependiendo de la distribución).

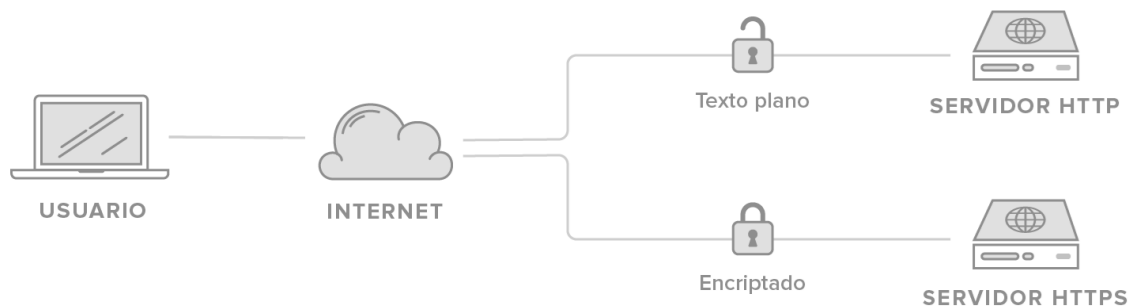
`source /home/db2bf/sqllib/db2profile`

2. Añada las líneas siguientes a `httpd.conf`:
3. `PassEnv LD_LIBRARY_PATH`
4. `PassEnv CLASSPATH`
5. `PassEnv LIBPATH`

`PassEnv VWSPATH`

Configuración mínima para garantizar la seguridad de la conexión web.

Encriptación SSL/TLS



PUERTO 80 Y 8080

HTTP (Puerto 80)

```
root@kali: /home/leurian
Archivo Editar Ver Buscar Terminal Ayuda
ipconfig
ipconfig
Configuración IP de Windows

Adaptador de Ethernet Conexión de Área Local 2:

    Sufixo DNS específico para la conexión. . . :
    Vínculo: dirección IPv6 local. . . : fe80::d05c:81e:dae1:eb02%17
    Dirección IPv4. . . . . : 5.5.233.137
    Máscara de subred . . . . . : 255.255.0.0
    Puerta de enlace predeterminada . . . . . :

Adaptador de Ethernet Conexión de Área Local:

    Sufixo DNS específico para la conexión. . . :
    Vínculo: dirección IPv6 local. . . : fe80::713c:ee85:9484:2955%11
    Dirección IPv4. . . . . : 192.168.1.44
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . . : 192.168.1.1
```

El protocolo de transferencia de hipertexto (HTTP, HyperText Transfer Protocol) es el protocolo usado en cada transacción de la Web (WWW). HTTP fue desarrollado por el consorcio W3C y la IETF.

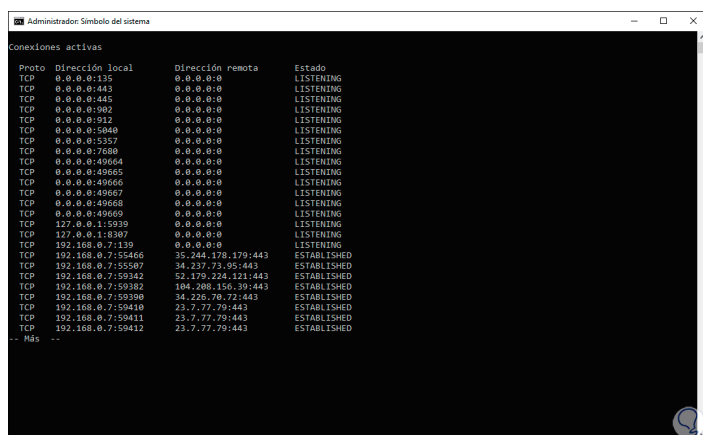
HTTP define la sintaxis y la semántica que utilizan los elementos software de la arquitectura web (clientes,

servidores, proxies) para comunicarse. Es un protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor. Al cliente que efectúa la petición (un navegador o un spider) se lo conoce como «user agent» (agente del usuario). A la información transmitida se la llama recurso y se la identifica mediante un URL. Los recursos pueden ser archivos, el resultado de la ejecución de un programa, una consulta a una base de datos, la traducción automática de un documento, etc.

HTTP es un protocolo sin estado, es decir, que no guarda ninguna información sobre conexiones anteriores. El desarrollo de aplicaciones web necesita frecuentemente

mantener estado. Para esto se usan las cookies, que es información que un servidor puede almacenar en el sistema cliente. Esto le permite a las aplicaciones web instituir la noción de «sesión», y también permite rastrear usuarios ya que las cookies pueden guardarse en el cliente por tiempo indeterminado.

PUERTO 8080



Administrador Símbolo del sistema

Conexiones activas

Proto	Dirección local	Dirección remota	Estado
TCP	0.0.0.0:135	0.0.0.0:0	LISTENING
TCP	0.0.0.0:443	0.0.0.0:0	LISTENING
TCP	0.0.0.0:445	0.0.0.0:0	LISTENING
TCP	0.0.0.0:900	0.0.0.0:0	LISTENING
TCP	0.0.0.0:912	0.0.0.0:0	LISTENING
TCP	0.0.0.0:5880	0.0.0.0:0	LISTENING
TCP	0.0.0.0:5357	0.0.0.0:0	LISTENING
TCP	0.0.0.0:7680	0.0.0.0:0	LISTENING
TCP	0.0.0.0:49664	0.0.0.0:0	LISTENING
TCP	0.0.0.0:49665	0.0.0.0:0	LISTENING
TCP	0.0.0.0:49666	0.0.0.0:0	LISTENING
TCP	0.0.0.0:49667	0.0.0.0:0	LISTENING
TCP	0.0.0.0:49668	0.0.0.0:0	LISTENING
TCP	0.0.0.0:49669	0.0.0.0:0	LISTENING
TCP	127.0.0.1:5939	0.0.0.0:0	LISTENING
TCP	127.0.0.1:5987	0.0.0.0:0	LISTENING
TCP	192.168.0.7:139	0.0.0.0:0	LISTENING
TCP	192.168.0.7:5866	35.244.178.179:443	ESTABLISHED
TCP	192.168.0.7:55587	34.237.73.95:443	ESTABLISHED
TCP	192.168.0.7:59342	52.179.224.121:443	ESTABLISHED
TCP	192.168.0.7:59382	108.200.156.39:443	ESTABLISHED
TCP	192.168.0.7:59390	34.226.70.72:443	ESTABLISHED
TCP	192.168.0.7:59410	23.7.77.79:443	ESTABLISHED
TCP	192.168.0.7:59411	23.7.77.79:443	ESTABLISHED
TCP	192.168.0.7:59412	23.7.77.79:443	ESTABLISHED

-- Más --

Abrir o cerrar puertos de internet 8080 – puertosabiertos.com. Una forma de navegar de forma más privada por Internet, ya que el servidor oculta tu IP al navegar por Internet.

El puerto por defecto para los servicios HTTP es el 80, y ahí puedes correr el IIS, el Apache, el Tomcat, el XAMPP o lo que quieras. La cuestión

del 8080 es que Tomcat supone que ya tienes algo en el puerto 80, el Apache, el IIS o el XAMPP, y para evitar problemas viene pre-configurado con el puerto 8080. Pero después se lo cambias sin más problemas y listo.

Sintaxis de systemctl

La sintaxis es la regla y el formato de cómo se puede usar el comando **systemctl**. Estas opciones de sintaxis se pueden reordenar, pero se debe seguir un formato.

La siguiente línea muestra un ejemplo de **sintaxis básica para utilizar el comando systemctl**:

```
1 systemctl [OPCIONES] {COMANDO}
```

Las opciones son indicadores que determinan cómo se ejecutan o controlan los comandos o modifican el comportamiento estos. El siguiente es un listado con algunas opciones que se pueden utilizar con el comando **systemctl**:

```

entreunosyceros@ubuntu-1804:~$ systemctl --help
systemctl [OPTIONS...] {COMMAND} ...

Query or send control commands to the systemd manager.

-h --help            Show this help
--version            Show package version
--system            Connect to system manager
--user              Connect to user service manager
-H --host=[USER@]HOST Operate on remote host
-M --machine=CONTAINER Operate on local container
-t --type=TYPE       List units of a particular type
--state=STATE        List units with particular LOAD or SUB or ACTIVE state
-p --property=NAME   Show only properties by this name
-a --all             Show all properties/all units currently in memory,
                    including dead/empty ones. To list all units installed on
                    the system, use the 'list-unit-files' command instead.
--failed             Same as --state=failed
-l --full            Don't ellipsize unit names on output
-r --recursive        Show unit list of host and local containers
--reverse            Show reverse dependencies with 'list-dependencies'
--job-mode=MODE      Specify how to deal with already queued jobs, when
                    queueing a new job
--show-types         When showing sockets, explicitly show their type
--value             When showing properties, only print the value
-i --ignore-inhibitors When shutting down or sleeping, ignore inhibitors
--kill-who=WHO       Who to send signal to
-s --signal=SIGNAL   Which signal to send
--now               Start or stop unit in addition to enabling or disabling it
--dry-run            Only print what would be done
-q --quiet           Suppress output
--wait              For (re)start, wait until service stopped again
--no-block           Do not wait until operation finished
--no-wall            Don't send wall message before halt/power-off/reboot
--no-reload          Don't reload daemon after en-/dis-abling unit files
--no-legend          Do not print a legend (column headers and hints)
--no-pager           Do not pipe output into a pager
--no-ask-password

```

- – -state=STATE → Con esta opción vamos a poder **enumerar unidades de un tipo particular de estado** de servicio: Activo o Inactivo.
- -a, – -all → Utilizaremos -a o – -all para **mostrar todas las propiedades / todas las unidades actualmente en memoria**. Para enumerar todas las unidades instaladas en el sistema, tendremos que utilizar el comando '*list-unit-files*' en su lugar.
- -r, – -recursive → Vamos a poder utilizar -r o – -recursive para **mostrar la lista de unidades de host y contenedores locales**.
- -H – -host = [USUARIO @] HOST → Nos va a permitir **operar en un host remoto**.
- is-system-running → Verificaremos **si el sistema está funcionando completamente**.
- hibernate → **Hibernación** del sistema.
- – -help → Nos va a mostrar **las opciones disponibles** mediante el mensaje de ayuda.

A continuación vamos a ver algunos ejemplos básicos de cómo ejecutar y **usar systemctl en Ubuntu 18.04**, que es el sistema que voy a utilizar para este ejemplo. Simplemente tendremos que ejecutar el comando systemctl para hacerlo trabajar.

Iniciar y detener servicios

Para **iniciar servicios utilizando el comando systemctl**, solo habrá que ejecutar algo como el siguiente comando:

```
entreunosyceros@ubuntu-1804:~$ sudo systemctl start apache2.service
entreunosyceros@ubuntu-1804:~$
```

```
1    sudo systemctl start application.service
```

También podemos **hacer referencia al nombre de la aplicación sin el .service final**. Para **detener el servicio**, el comando a utilizar será algo como:

```
entreunosyceros@ubuntu-1804:~$ sudo systemctl stop apache2.service
entreunosyceros@ubuntu-1804:~$
```

```
1    sudo systemctl stop application.service
```

Reiniciar y recargar servicios

Si buscas **reiniciar el servicio**, debes escribir en la terminal algo como:

```
entreunosyceros@ubuntu-1804:~$ sudo systemctl restart apache2.service
entreunosyceros@ubuntu-1804:~$
```

```
1    sudo systemctl restart application.service
```

Para **recargar el servicio**, el comando a utilizar será:

```
entreunosyceros@ubuntu-1804:~$ sudo systemctl reload apache2.service
entreunosyceros@ubuntu-1804:~$
```

```
1    sudo systemctl reload application.service
```

Al recargar un servicio solo se vuelven a cargar los cambios de configuración en un servicio en ejecución y no se reiniciará por completo el servicio. Para reiniciar completamente un servicio en ejecución, lo ideal es utilizar la opción *restart*.

Habilitar y deshabilitar servicios

```
entreunosyceros@ubuntu-1804:~$ sudo systemctl disable apache2.service
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install disable apache2
entreunosyceros@ubuntu-1804:~$ sudo systemctl enable apache2.service
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
entreunosyceros@ubuntu-1804:~$
```

Si queremos deshabilitar o habilitar un servicio, no habrá más que utilizar los siguientes comandos. Habilitar un servicio nos permitirá que se inicie automáticamente cada vez que se inicie el servidor. **Para habilitar un servicio** el comando que debemos utilizar debe ser algo como:

```
1 sudo systemctl enable application.service
```

Si deshabilitamos un servicio, el servicio no se ejecutará a menos que lo volvamos a habilitar. **Para deshabilitar un servicio** el comando debe ser:

```
1 sudo systemctl disable application.service
```

Verificar el estado del servicio

Para verificar el estado de un servicio, habrá que **utilizar la opción status** de la siguiente forma:

```
entreunosyceros@ubuntu-1804:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Drop-In: /lib/systemd/system/apache2.service.d
            └─apache2-systemd.conf
   Active: active (running) since Sun 2019-10-13 00:26:39 CEST; 7min ago
     Main PID: 3728 (apache2)
       Tasks: 55 (limit: 3401)
    CGroup: /system.slice/apache2.service
            └─3728 /usr/sbin/apache2 -k start
              └─3795 /usr/sbin/apache2 -k start
                └─3796 /usr/sbin/apache2 -k start

oct 13 00:26:39 ubuntu-1804 systemd[1]: Starting The Apache HTTP Server...
oct 13 00:26:39 ubuntu-1804 apachectl[3723]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1.
oct 13 00:26:39 ubuntu-1804 systemd[1]: Started The Apache HTTP Server.
oct 13 00:28:39 ubuntu-1804 systemd[1]: Reloading The Apache HTTP Server.
oct 13 00:28:39 ubuntu-1804 apachectl[3791]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1.
oct 13 00:28:39 ubuntu-1804 systemd[1]: Reloaded The Apache HTTP Server.
lines 1-18/18 (END)
```

```
1 sudo systemctl status application.service
```

Listar todos los servicios

Para **enumerar todos los servicios que se están ejecutando o están inactivos**, podemos ejecutar:

```
entreunosyceros@ubuntu-1804:~$ systemctl list-units --all --type=service --no-pager
```

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
accounts-daemon.service	loaded	active	running	Accounts Service
acpid.service	loaded	active	running	ACPI event daemon
alsa-restore.service	loaded	active	exited	Save/Restore Sound Card State
alsa-state.service	loaded	inactive	dead	Manage Sound Card State (restore and store)
anacron.service	loaded	inactive	dead	Run anacron jobs
apache2.service	loaded	active	running	The Apache HTTP Server
apparmor.service	loaded	active	exited	AppArmor initialization
apport.service	loaded	active	exited	LSB: automatic crash report generation
apt-daily-upgrade.service	loaded	inactive	dead	Daily apt upgrade and clean activities
apt-daily.service	loaded	inactive	dead	Daily apt download activities
auditd.service	not-found	inactive	dead	auditd.service
avahi-daemon.service	loaded	active	running	Avahi mDNS/DNS-SD Stack
colord.service	loaded	active	running	Manage, Install and Generate Color Profiles
connman.service	not-found	inactive	dead	connman.service
console-screen.service	not-found	inactive	dead	console-screen.service
console-setup.service	loaded	active	exited	Set console font and keymap
cron.service	loaded	active	running	Regular background program processing daemon
cups-browsed.service	loaded	active	running	Make remote CUPS printers available locally
cups.service	loaded	active	running	CUPS Scheduler
dbus.service	loaded	active	running	D-Bus System Message Bus
dns-clean.service	loaded	inactive	dead	Clean up any mess left by dnsmasq
emergency.service	loaded	inactive	dead	Emergency Shell
festival.service	not-found	inactive	dead	festival.service
friendly-recovery.service	loaded	inactive	dead	Recovery mode menu
fstrim.service	loaded	inactive	dead	Discard unused blocks
fwupd.service	loaded	active	running	Firmware update daemon

1 systemctl list-units --all --type=service --no-pager

El anterior comando debería listar todos los servicios y la pantalla de salida que mostrará, será similar a la anterior captura de pantalla. Si nos interesa **ver solo todos los servicios activos**, debemos utilizar el siguiente comando:

```
entreunosyceros@ubuntu-1804:~$ systemctl list-units --all --state=active
```

UNIT	LOAD	ACTIVE	SUB	DESCRIPTION
proc-sys-fs-binfmt-misc.automount	loaded	active	waiting	Arbitrary Executable File Formats File
dev-cdrom.device	loaded	active	plugged	VBOX_CD-ROM
dev-disk-by\x2did-ata\x2dvbox_CD\x2dROM_VB2\x2d01700376.device	loaded	active	plugged	VBOX_CD-ROM
dev-disk-by\x2did-ata\x2dvbox_HARDDISK_VB5e5239cc\x2dea6bd7a4.device	loaded	active	plugged	VBOX_HARDDISK
dev-disk-by\x2did-ata\x2dvbox_HARDDISK_VB5e5239cc\x2dea6bd7a4\x2dpart1.device	loaded	active	plugged	VBOX_HARDDISK 1
dev-disk-by\x2dpartuuid-23b67b08\x2d01.device	loaded	active	plugged	VBOX_HARDDISK 1
dev-disk-by\x2dpath-pci\x2d0000:00:01.1\x2data\x2d2.device	loaded	active	plugged	VBOX_CD-ROM
dev-disk-by\x2dpath-pci\x2d0000:00:0d.0\x2data\x2d1.device	loaded	active	plugged	VBOX_HARDDISK
dev-disk-by\x2dpath-pci\x2d0000:00:0d.0\x2data\x2d1\x2dpart1.device	loaded	active	plugged	VBOX_HARDDISK 1
dev-disk-by\x2duuid-37464a1a\x2d6292\x2d4652\x2daa18\x2dcbc15afe2888.device	loaded	active	plugged	VBOX_HARDDISK 1
dev-dvd.device	loaded	active	plugged	VBOX_CD-ROM
dev-loop0.device	loaded	active	plugged	/dev/loop0
dev-loop1.device	loaded	active	plugged	/dev/loop1
dev-loop2.device	loaded	active	plugged	/dev/loop2
dev-loop3.device	loaded	active	plugged	/dev/loop3
dev-loop4.device	loaded	active	plugged	/dev/loop4
dev-loop5.device	loaded	active	plugged	/dev/loop5
dev-loop6.device	loaded	active	plugged	/dev/loop6
dev-loop7.device	loaded	active	plugged	/dev/loop7
dev-rfkill.device	loaded	active	plugged	/dev/rfkill
dev-sda.device	loaded	active	plugged	VBOX_HARDDISK
dev-sda1.device	loaded	active	plugged	VBOX_HARDDISK 1
dev-sr0.device	loaded	active	plugged	VBOX_CD-ROM

1 systemctl list-units --all --state=active

Para **listar todos los servicios inactivos**, el comando a ejecutar será:


```
entreunoscyceros@ubuntu-1804:~$ systemctl list-units --all --state=inactive
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
proc-sys-fs-binfmt_misc.mount      loaded inactive dead Arbitrary Executable File Formats File System
tmp.mount                          not-found inactive dead tmp.mount
systemd-ask-password-console.path  loaded inactive dead Dispatch Password Requests to Console Directory Watch
alsa-state.service                 loaded inactive dead Manage Sound Card State (restore and store)
anacron.service                    loaded inactive dead Run anacron jobs
apt-daily-upgrade.service           loaded inactive dead Daily apt upgrade and clean activities
apt-daily.service                  loaded inactive dead Daily apt download activities
auditd.service                     not-found inactive dead auditd.service
connman.service                    not-found inactive dead connman.service
console-screen.service              not-found inactive dead console-screen.service
dns-clean.service                  loaded inactive dead Clean up any mess left by dnsmasq
emergency.service                   loaded inactive dead Emergency Shell
festival.service                    not-found inactive dead festival.service
friendly-recovery.service           loaded inactive dead Recovery mode menu
fstor.service                       loaded inactive dead Discard unused blocks
getty-static.service                loaded inactive dead Getty on tty2-tty6 if dbus and logind are not available
getty@tty1.service                 loaded inactive dead Getty on tty1
gpu-manager.service                 loaded inactive dead Detect the available GPUs and deal with any system changes
kbd.service                         not-found inactive dead kbd.service
motd-news.service                  loaded inactive dead Message of the Day
ondemand.service                   loaded inactive dead Set the CPU Frequency Scaling governor
plymouth-quit-wait.service          loaded inactive dead Hold until boot process finishes up
plymouth-quit.service              loaded inactive dead Terminate Plymouth Boot Screen
plymouth-read-write.service         loaded inactive dead Tell Plymouth To Write Out Runtime Data
plymouth-start.service              loaded inactive dead Show Plymouth Boot Screen
pppd-dns.service                   loaded inactive dead Restore /etc/resolv.conf if the system crashed before the ppp link was shut down
rc-local.service                   loaded inactive dead /etc/rc.local Compatibility
rescue.service                      loaded inactive dead Rescue Shell
resolvconf.service                 not-found inactive dead resolvconf.service
rsync.service                       loaded inactive dead fast remote file copy program daemon
```

1 `systemctl list-units --all --state=inactive`

Más información

Para **más información acerca del uso de systemctl**, no tendremos más que utilizar **la ayuda con la opción `--help` o consultar las páginas man:**

```
SYSTEMCTL(1)                                systemctl                                SYSTEMCTL(1)

NAME
    systemctl - Control the systemd system and service manager

SYNOPSIS
    systemctl [OPTIONS...] COMMAND [NAME...]

DESCRIPTION
    systemctl may be used to introspect and control the state of the "systemd" system and service manager. Please refer to systemd(1) for an introduction into the basic concepts and functionality this tool manages.

OPTIONS
    The following options are understood:

    -t, --type=
        The argument should be a comma-separated list of unit types such as service and socket.

        If one of the arguments is a unit type, when listing units, limit display to certain unit types. Otherwise, units of all types will be shown.

        As a special case, if one of the arguments is help, a list of allowed values will be printed and the program will exit.

    --state=
        The argument should be a comma-separated list of unit LOAD, SUB, or ACTIVE states. When listing units, show only those in the specified states. Use --state=failed to show only failed units.

        As a special case, if one of the arguments is help, a list of allowed values will be printed and the program will exit.

    -p, --property=
        When showing unit/job/manager properties with the show command, limit display to properties specified in the argument. The argument should be a comma-separated list of property names, such as "MainPID". Unless specified, all known properties are shown. If specified more than once, all properties with the specified names are shown. Shell completion is implemented for property names.

        For the manager itself, systemctl show will show all available properties. Those properties are documented in systemd-system.conf(5).

        Properties for units vary by unit type, so showing any unit (even a non-existent one) is a way to list properties pertaining to this type. Similarly, showing any job will list properties pertaining to all jobs. Properties for units are documented in systemd.unit(5), and the pages for individual unit types systemd.service(5), systemd.socket(5), etc.

Manual page systemctl(1) line 1 (press h for help or q to quit)
```

Sobre el directorio public_html

Directorio public_html

El directorio public_html es la raíz web para el nombre del dominio principal.

Esto significa que public_html es la carpeta donde se colocan todos los archivos del sitio web que se desea aparezcan cuando alguien escribe el dominio principal.

Dicho de otra manera, cuando alguien escribe el nombre de tu dominio en el navegador, lo que está en la carpeta public_html se le será mostrado.

Dominios adicionales y subdominios:

Puedes también crear dominios adicionales y subdominios que utilicen una carpeta dentro de public_html.

Un ejemplo podría ser:

- Si creas un dominio adicional DominioAdicional.com, este usará una subcarpeta similar a /public_html/DominioAdicional.com.

Permisos:

La carpeta public_html debe siempre tener 0750 permisos.

Todas las carpetas que se encuentren dentro de la carpeta public_html deben tener 0755 permisos.

Todos los archivos dentro de la carpeta public_html deben tener 0755 o 0644 permisos.

Conclusiones

HTML es un lenguaje de marcación que sirve para definir el contenido de las páginas web. Se compone en base a etiquetas, también llamadas marcas o tags, con las cuales conseguimos expresar las partes de un documento, cabecera, cuerpo, encabezados, párrafos, etc. En definitiva, el contenido de una página web.

HTML es el primer lenguaje que debe aprender cualquier persona interesada en construir un sitio web. A partir del HTML podemos pasar a muchos otros lenguajes interesantes que sirven para hacer cosas diversas y más avanzadas. Es decir, sea cual sea la tecnología, herramienta o gestor de contenido que nos hayamos propuesto aprender, o que tengamos que usar en nuestro día a día, HTML siempre será el lenguaje en el que toda web se construye y, por tanto, es de obligado conocimiento para todos.

Aprender HTML es sencillo. En pocos días o semanas serás capaz de entender y usar las etiquetas más comunes y componer documentos HTML (páginas web) correctas. Aquí encontrarás muchas ayudas para poder dar esos primeros pasos y luego profundizar en cualquier área que necesites.