Servido

Servidor LAMP

Jose Almiron Lopez

Tabla de contenido

[Instalación de Apache 2](#_Toc120643391)

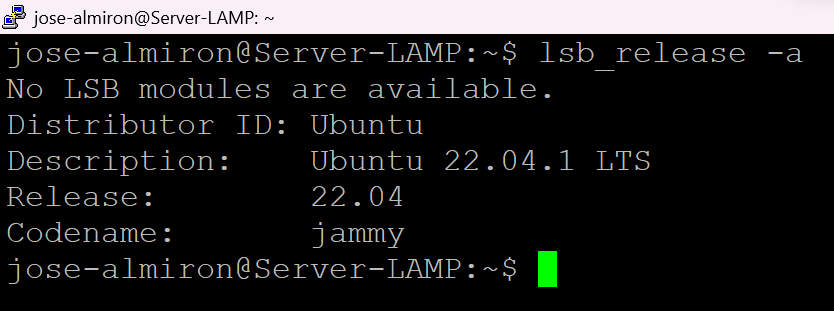
[Instalar PHP 8 con modulo de soporte para Apache 5](#_Toc120643392)

[Creación y comprobación de acceso al sitio web 9](#_Toc120643393)

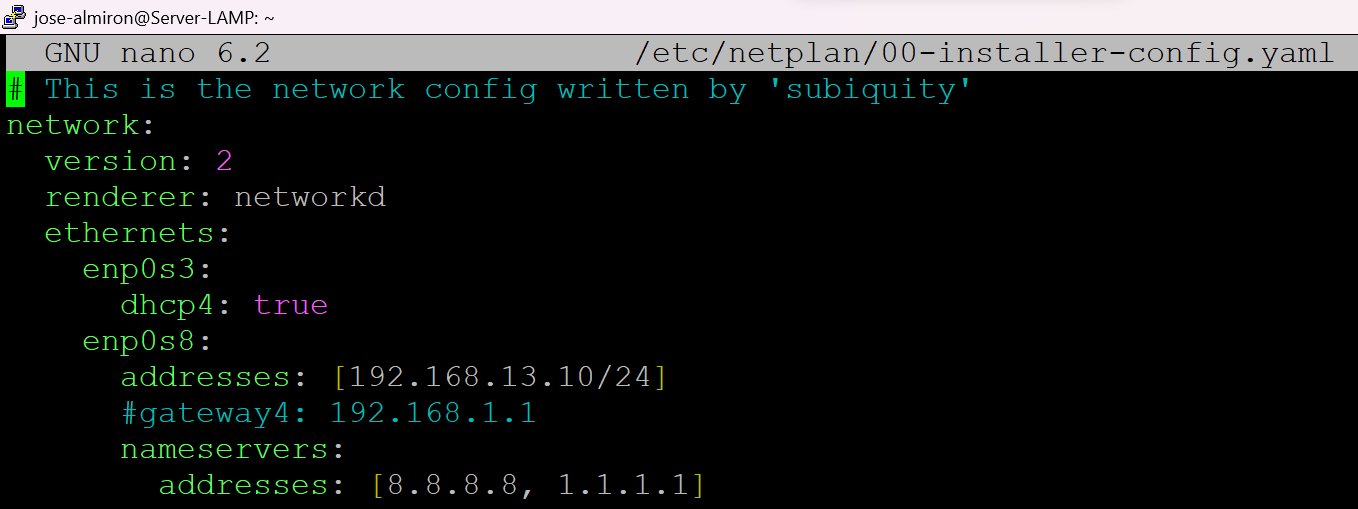
[Comprobaciones en el cliente 16](#_Toc120643394)

# Instalación de Apache

Empezamos la práctica con una máquina virtual corriendo Ubuntu server en su versión 22.04



Respecto a la configuración de red, partimos de unas dos interfaces una con adaptador puente y otra red interna

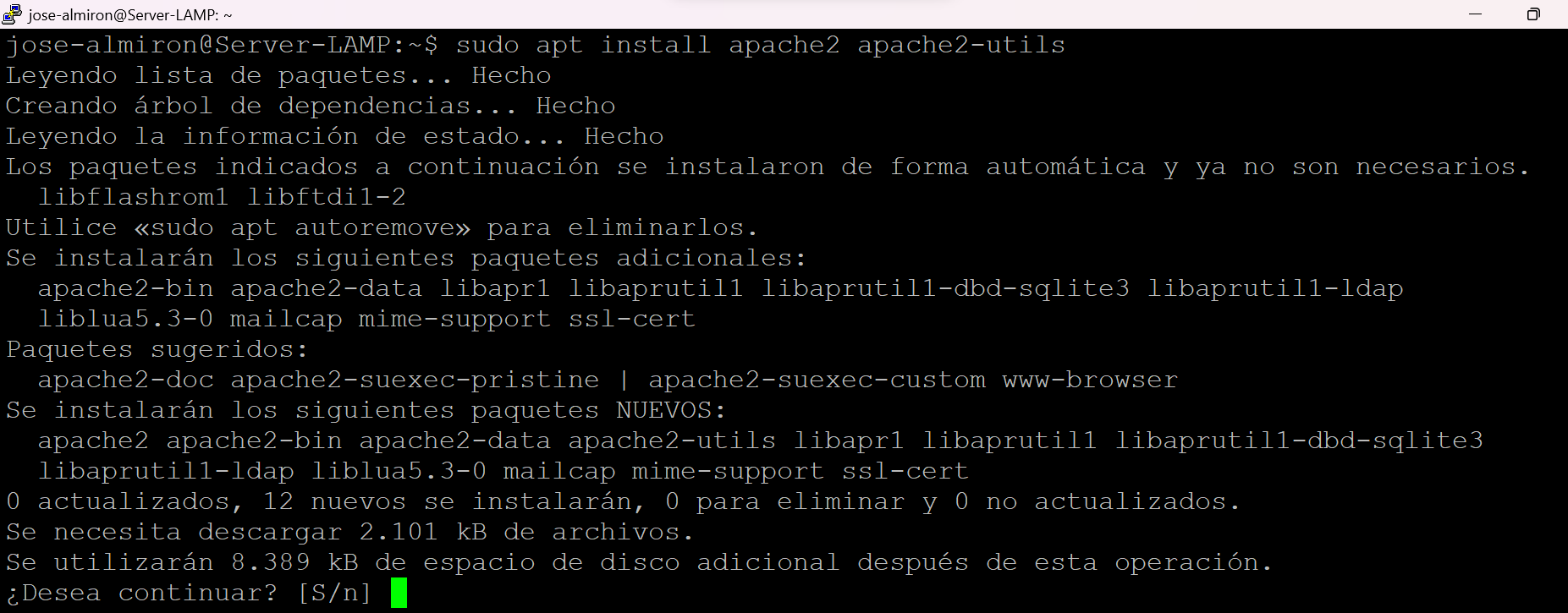


Procedemos a la actualización de los repositorios e instalación de Apache

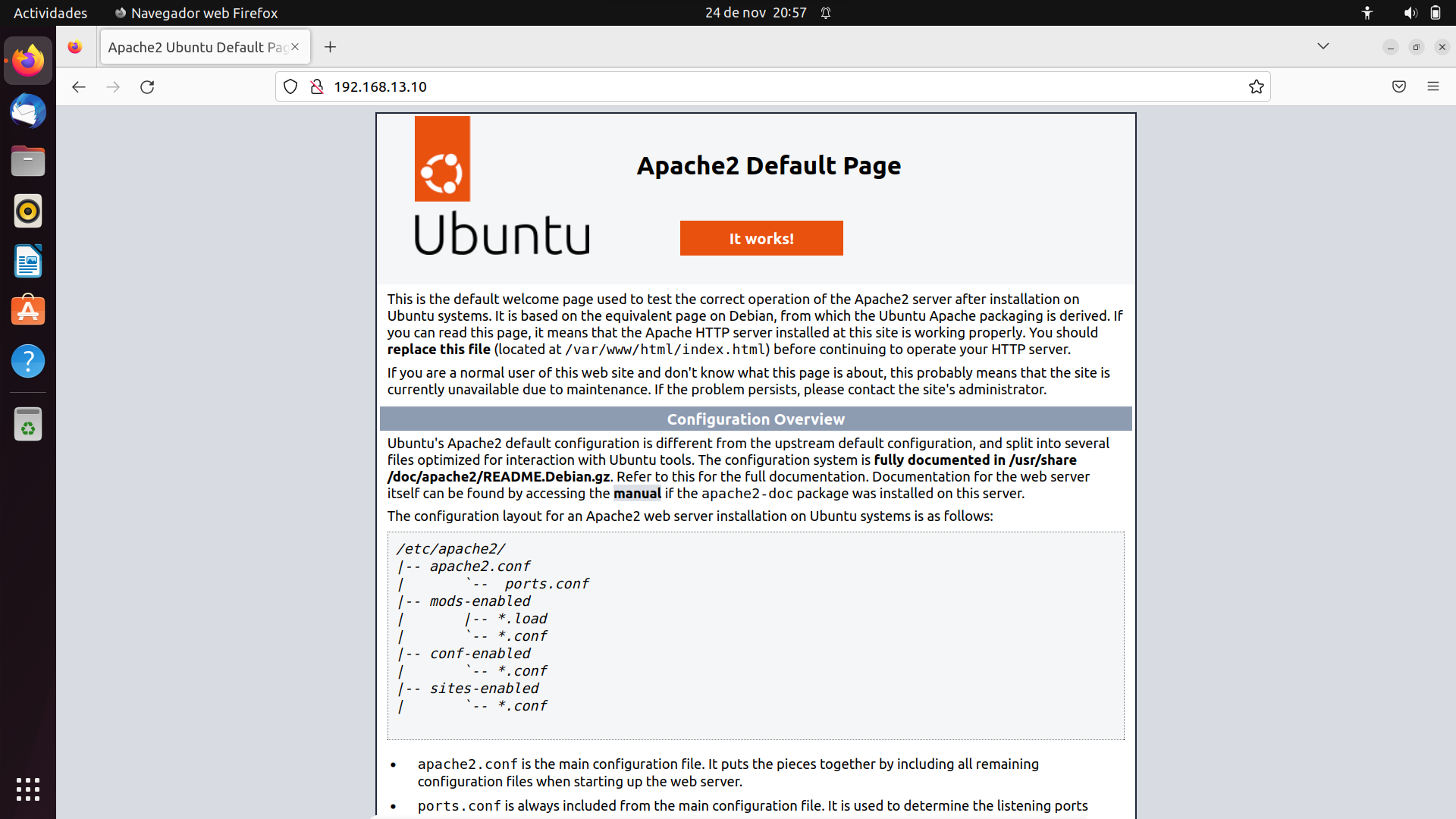
***sudo apt update***

***sudo apt upgrade***

***sudo apt install Apache2***

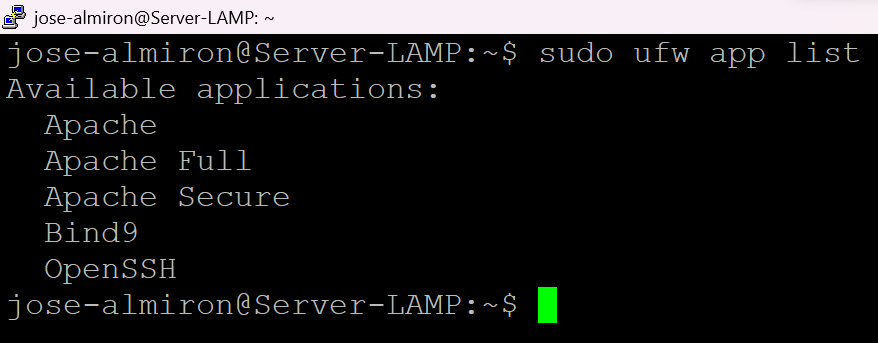


Probamos la carga de la web de inicio de Apache con las dos interfaces de red, la interna y de adaptador puente. Como estoy usando Ubuntu server no cuento con interfaz grafica y no puedo comprobar la web con localhost o 172.0.0.1

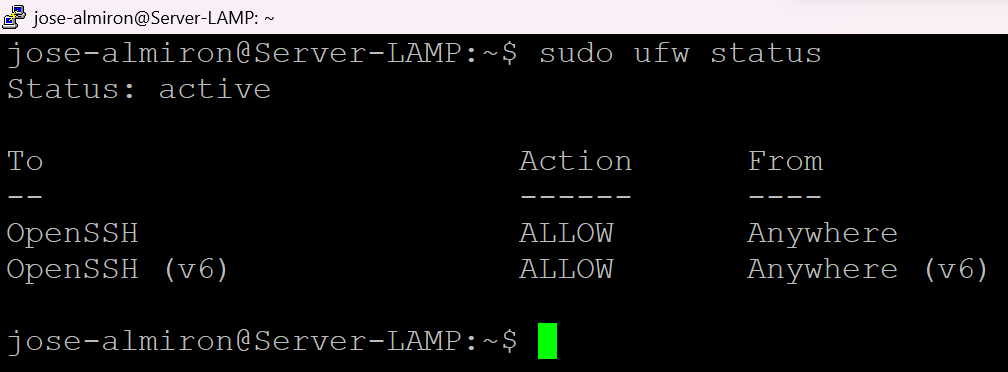




Configuración del firewall y carga del perfil de servicio correspondiente



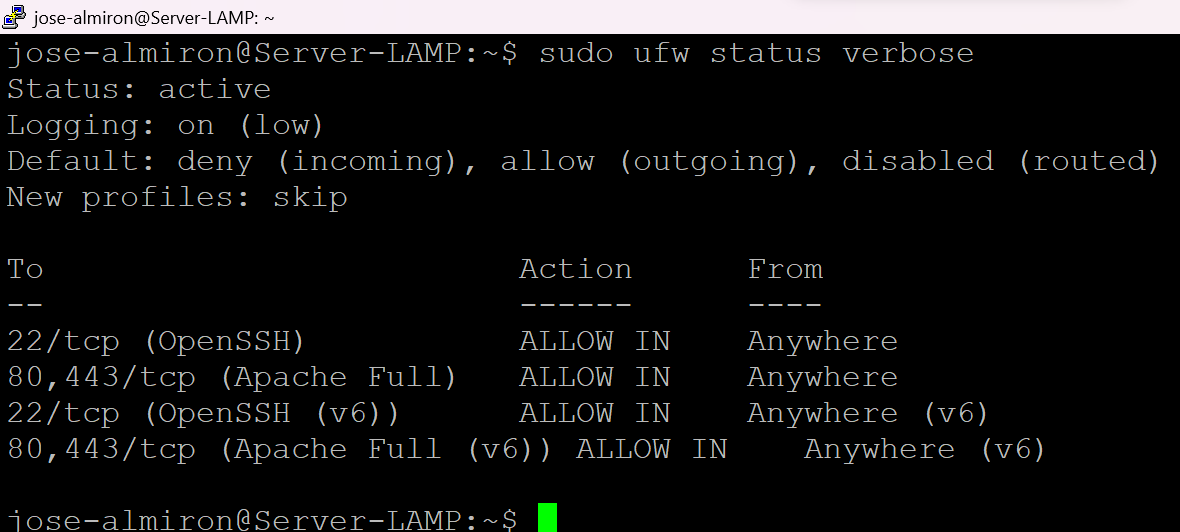
Habilitamos el firewall, y en mi caso habilito OpenSSH ya que es la forma que utilizo para conectarme a la maquina virtual



Habilitamos el perfil de Apache Full para tener acceso tanto por el puerto 80 como por el 443, es decir tener acceso con HTTP y HTTPS



Habilitamos el servicio de firewall y comprobamos las conexiones permitidas por el cortafuegos

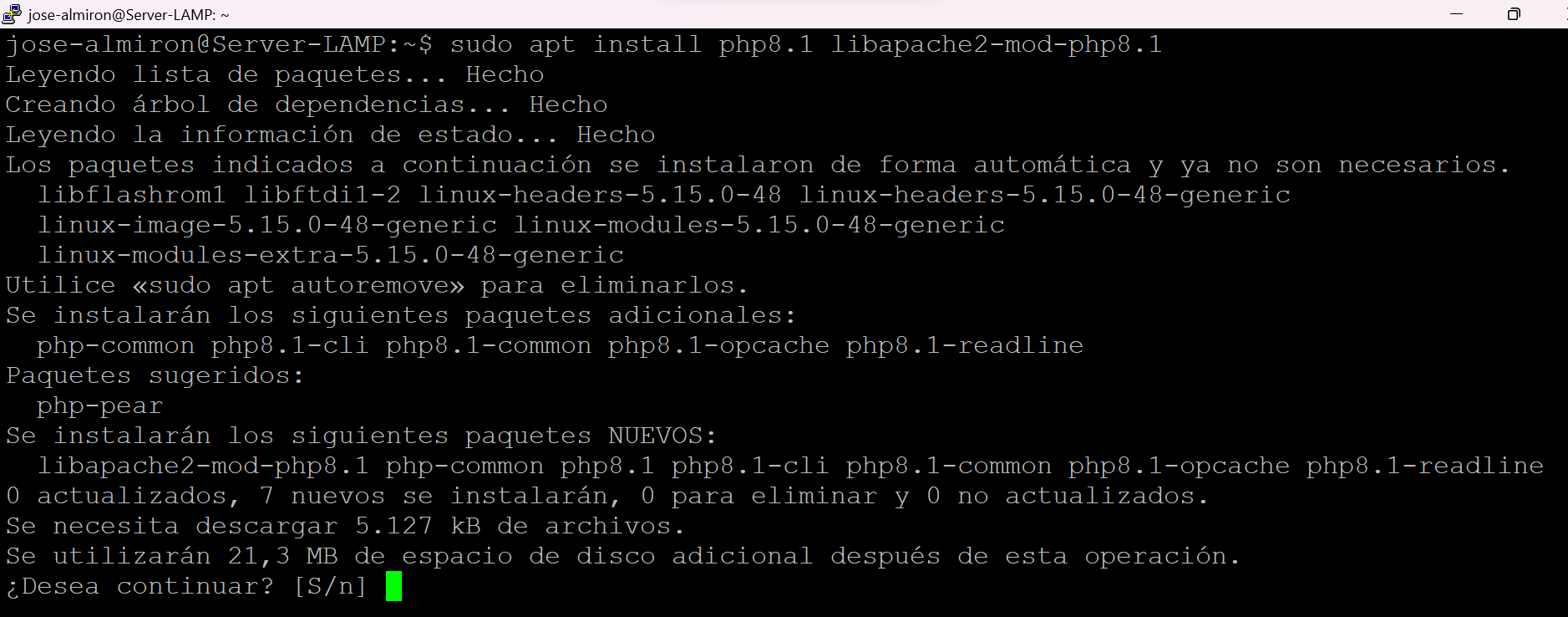


Comprobamos la carga de la web de inicio de Apache mediante HTTP

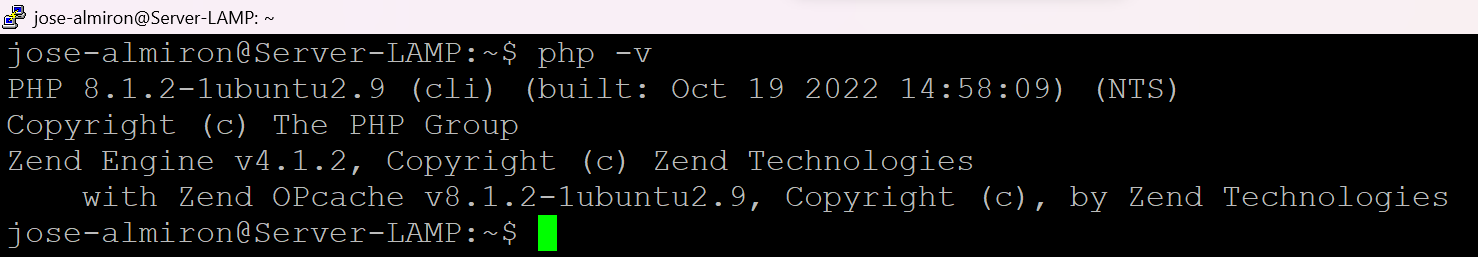


# Instalar PHP 8 con modulo de soporte para Apache

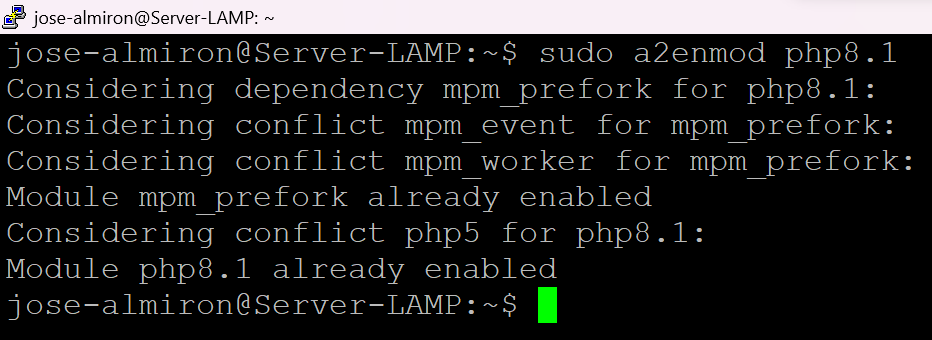
Instalamos php como módulo de apache2



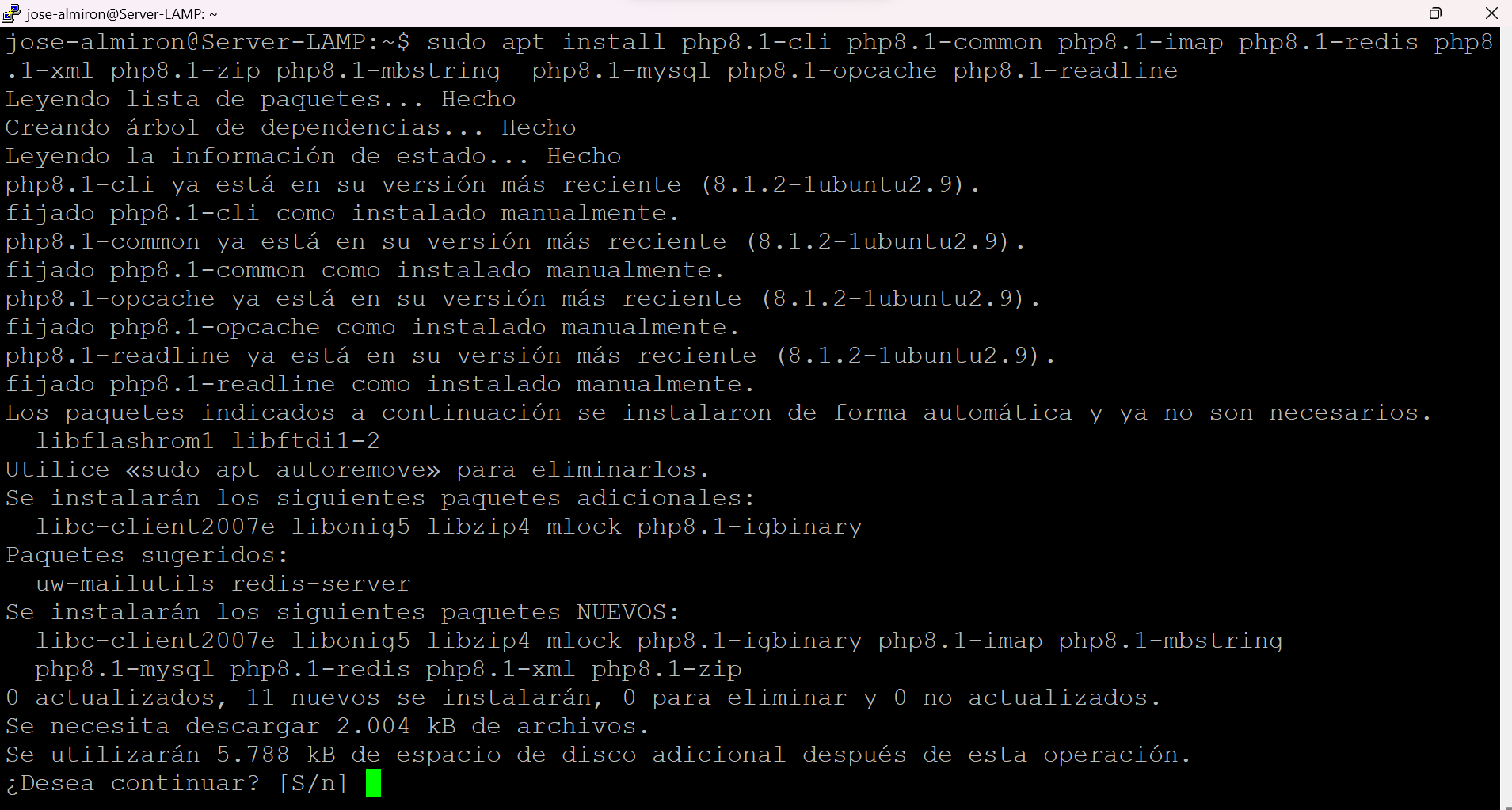
Mostramos la versión de php



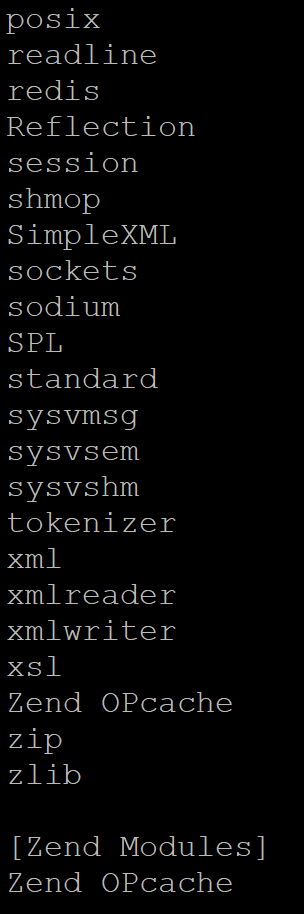
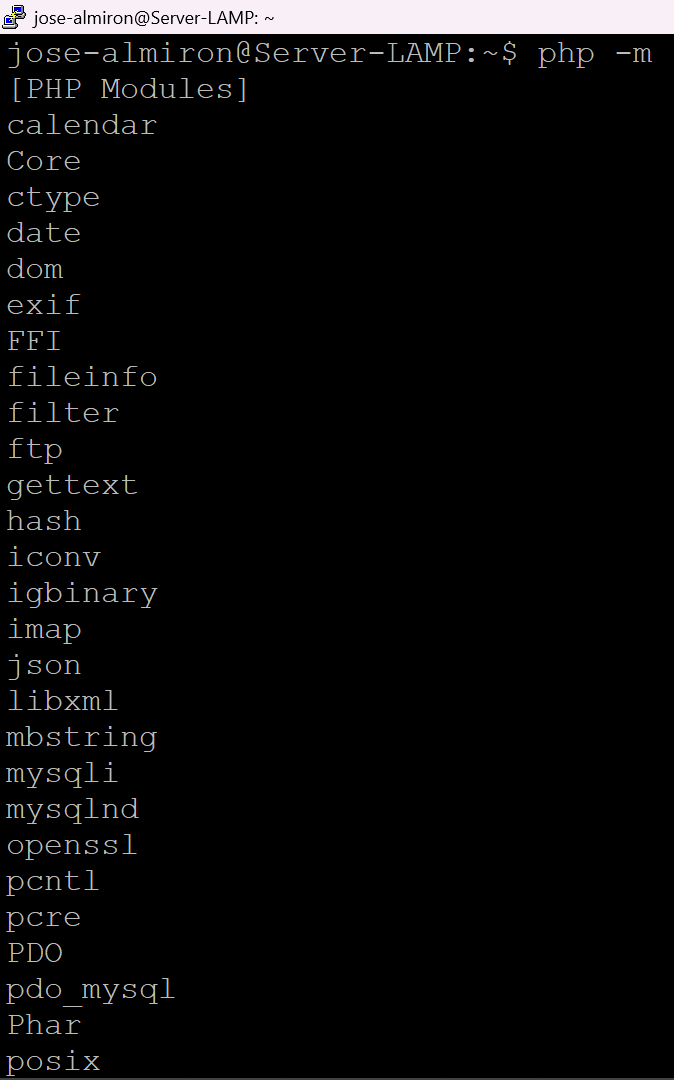
Tras la instalación activamos el modulo de php y reiniciamos el servidor de apache



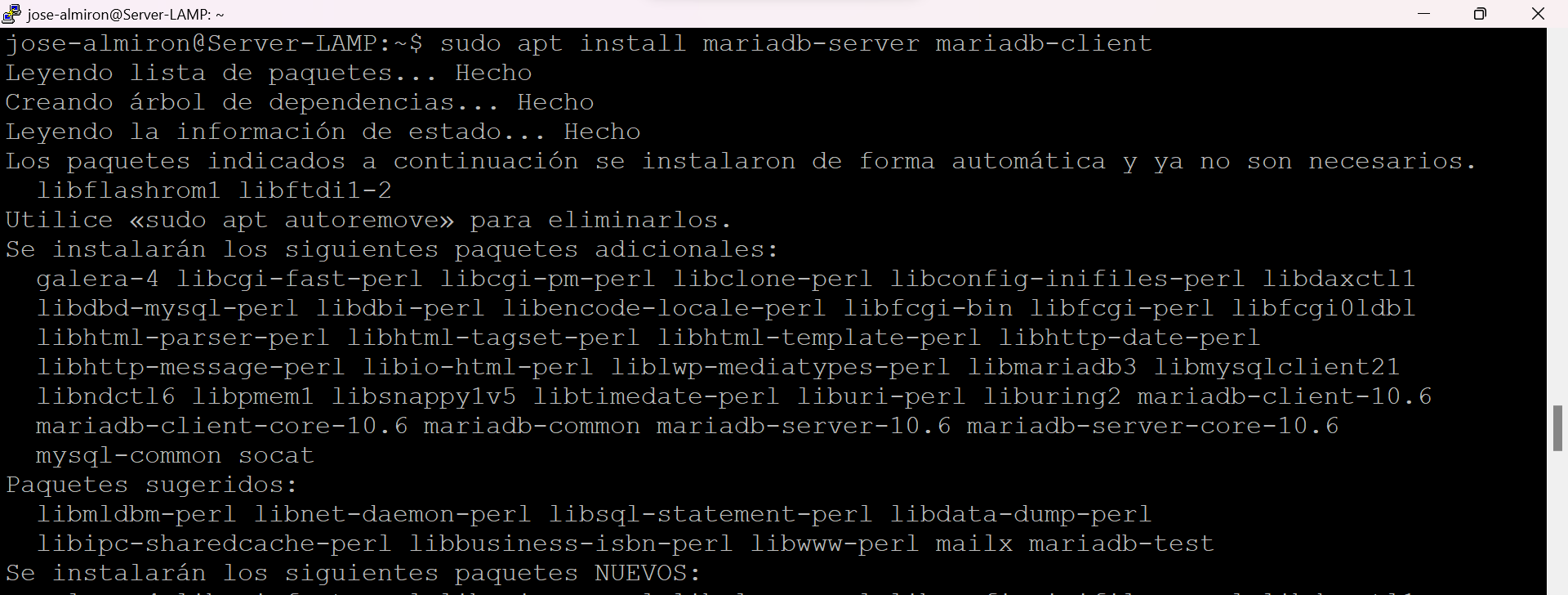
Instalamos algunos módulos de php para mejorar la sintaxis



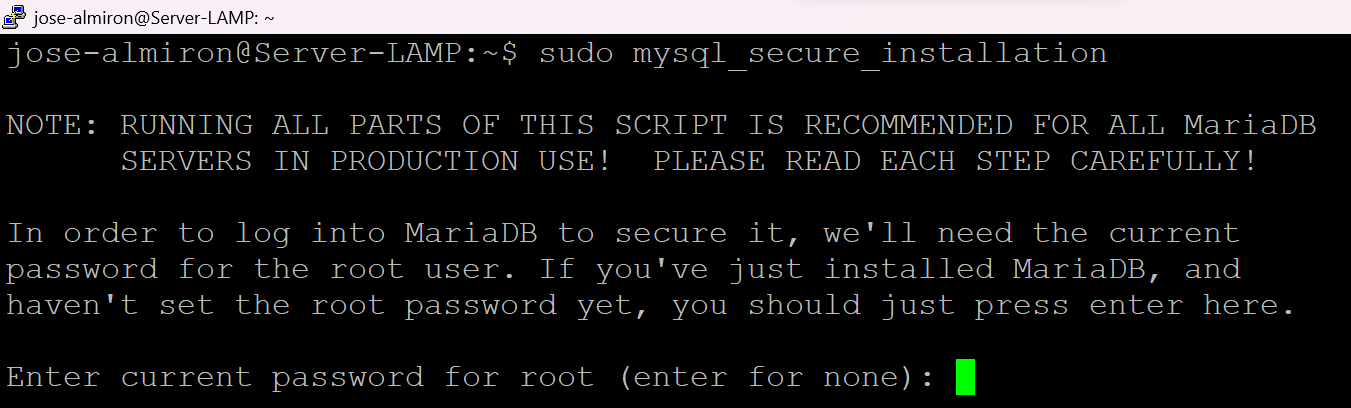
Comprobamos los módulos instalados



Instalamos MariaDB para poder servir paginas webs con acceso a bases de datos

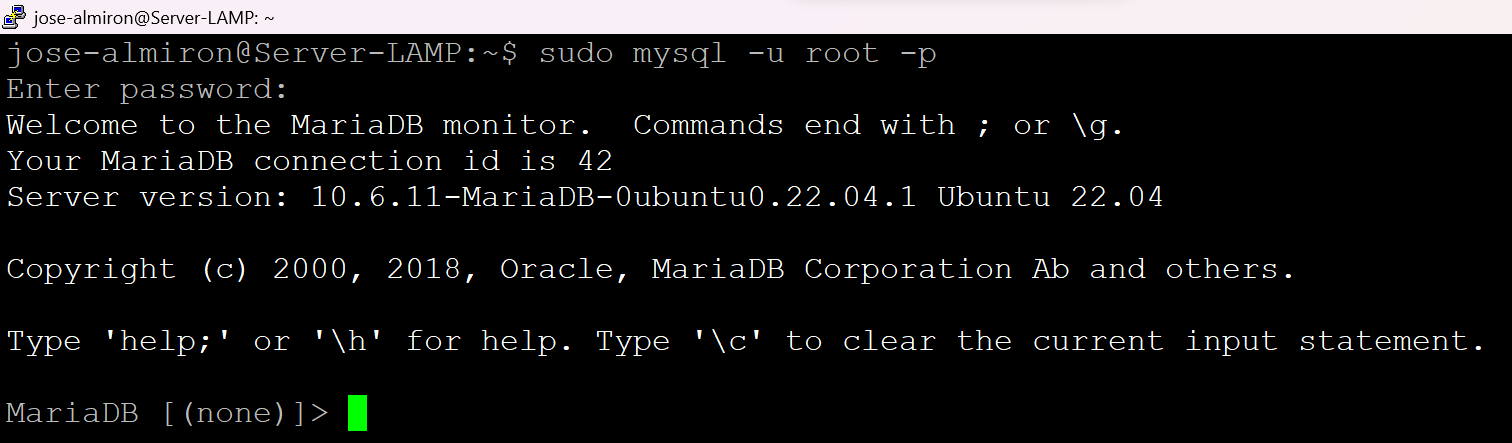


Configuramos mariadb ejecutando el script de seguridad





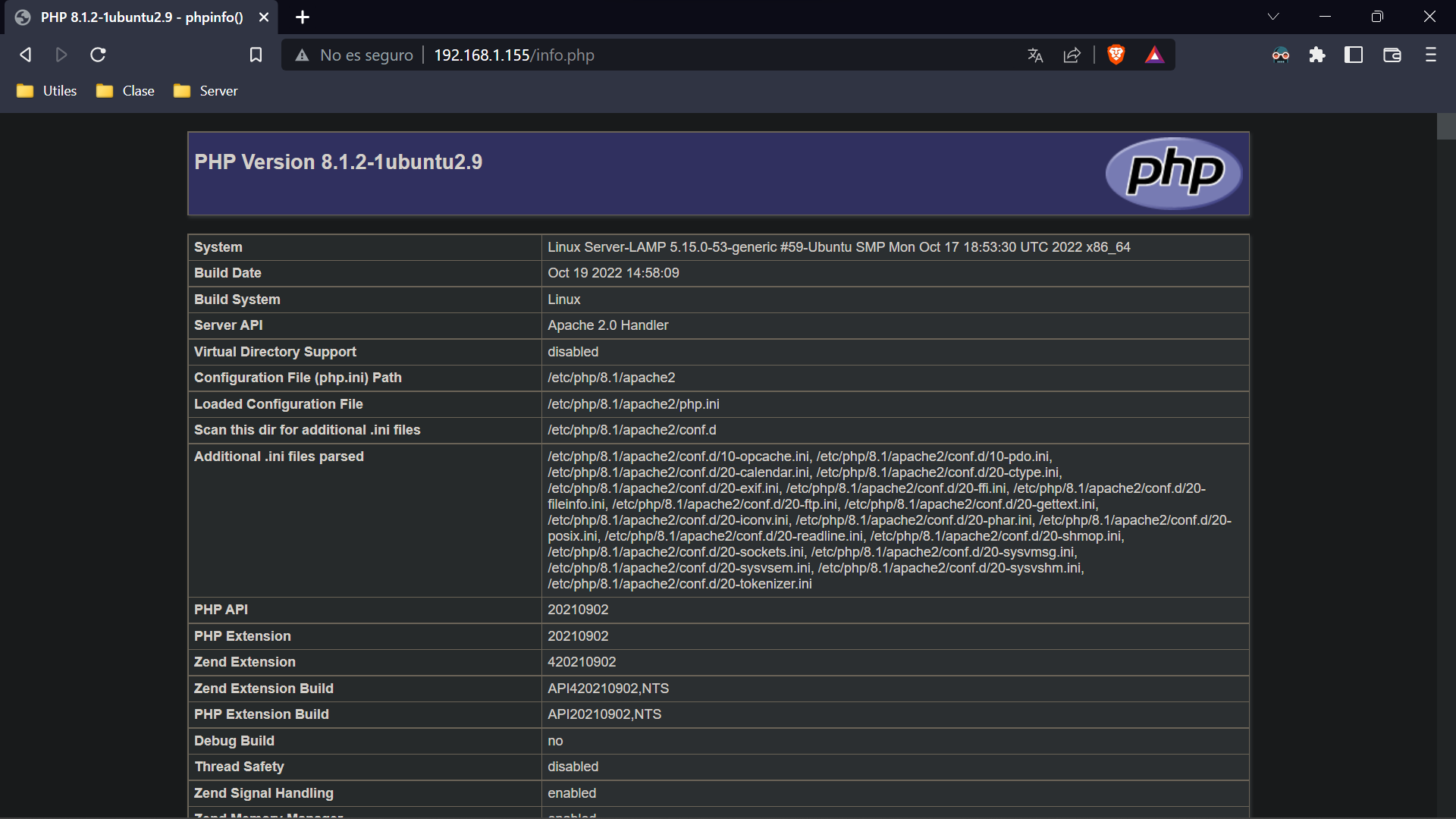
para probar la conexión al servidor de bases de datos como usuario root ejecutamos el siguiente comando



Y comprobamos la página de inicio de php que tendremos que crear nosotros mismos

***sudo anno /var/www/html/info.php***

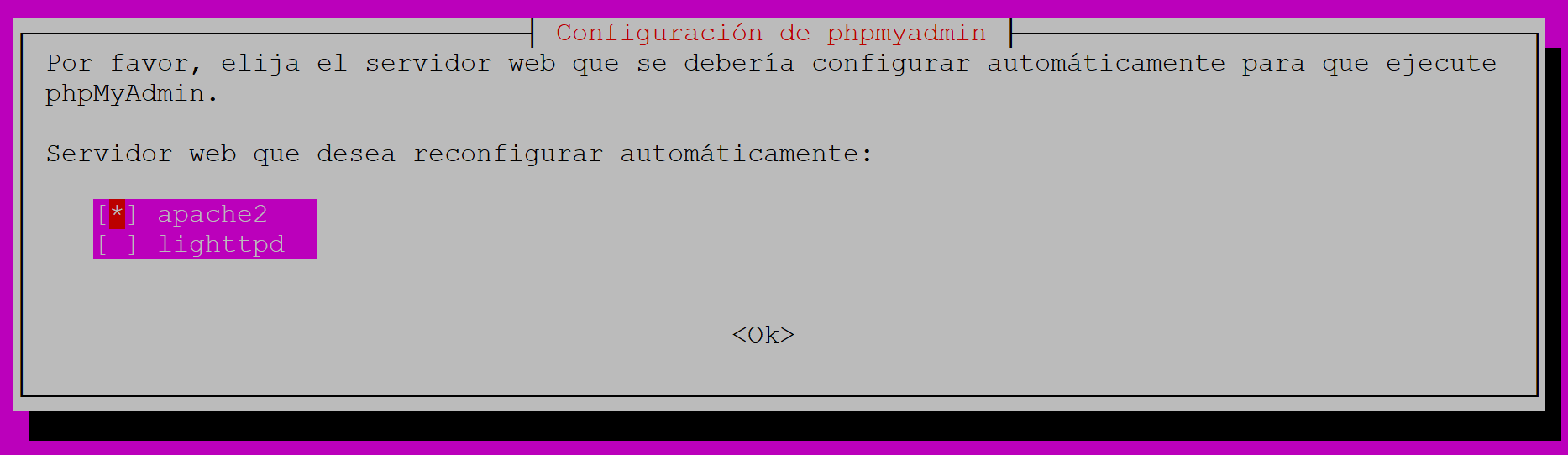
***<?php phpinfo();***



Instalamos phpmyadmin para poder gestionar mariadb por su pagina web



Seleccionamos apache y nos pedirá una contraseña para el usuario que acceda a phpmyadmin

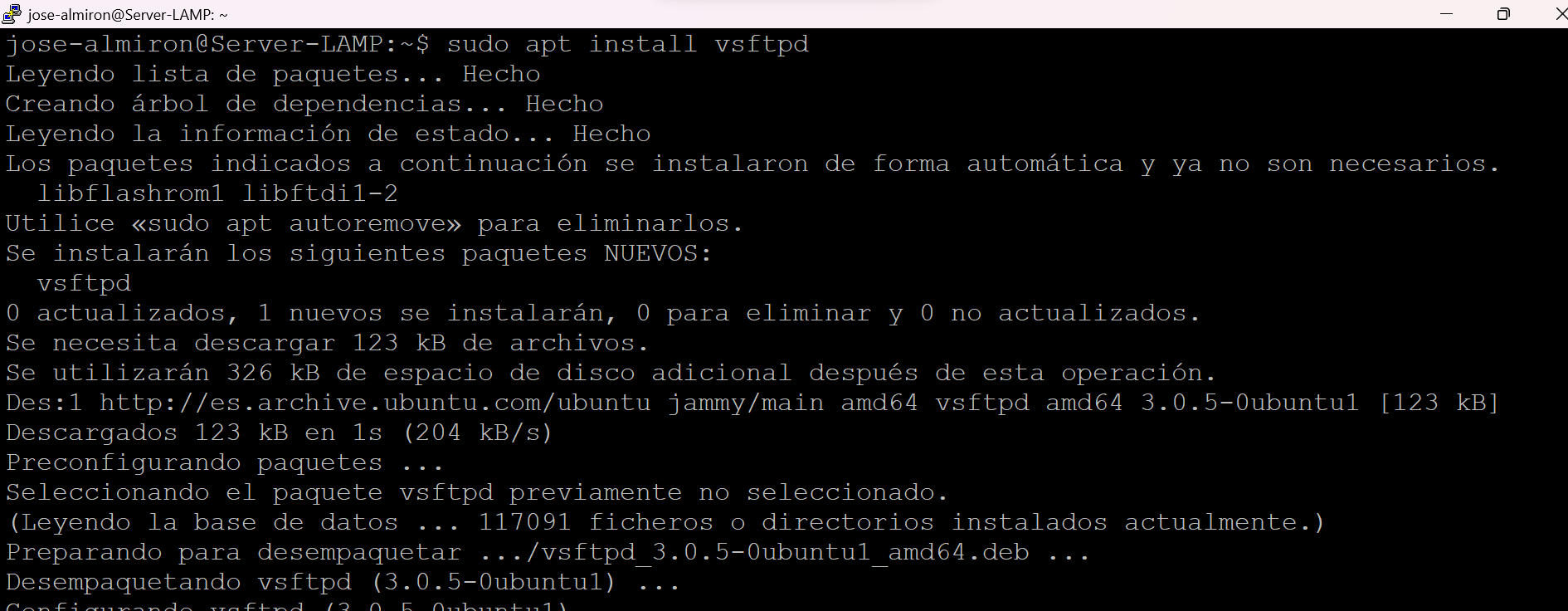


Y mostramos las pagina de phpmyadmin

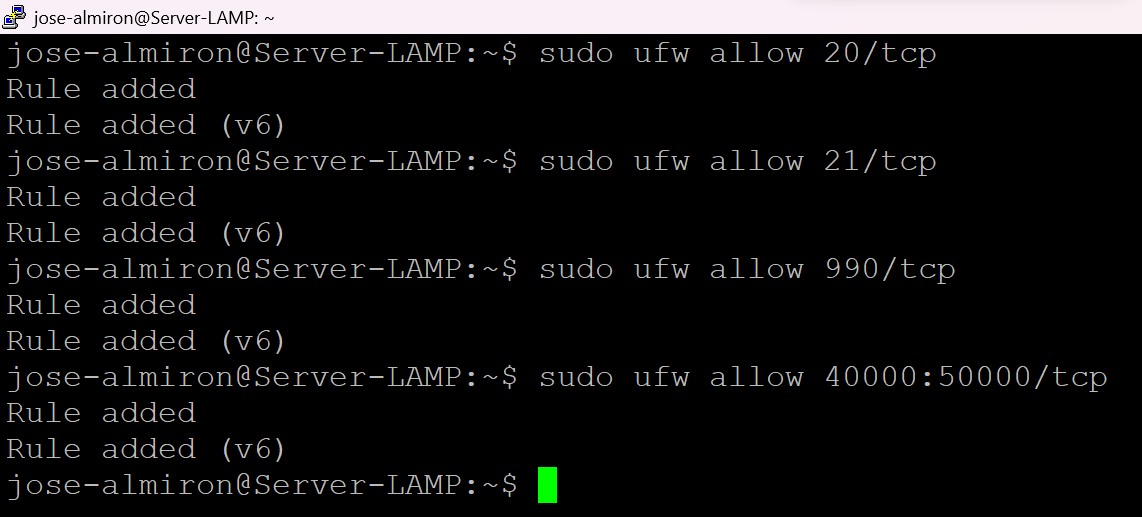


# Creación y comprobación de acceso al sitio web

Lo primero que hare será instalar y configurar el servicio FTP para poder transferir los ficheros de las webs a comprobar de una forma mas sencilla

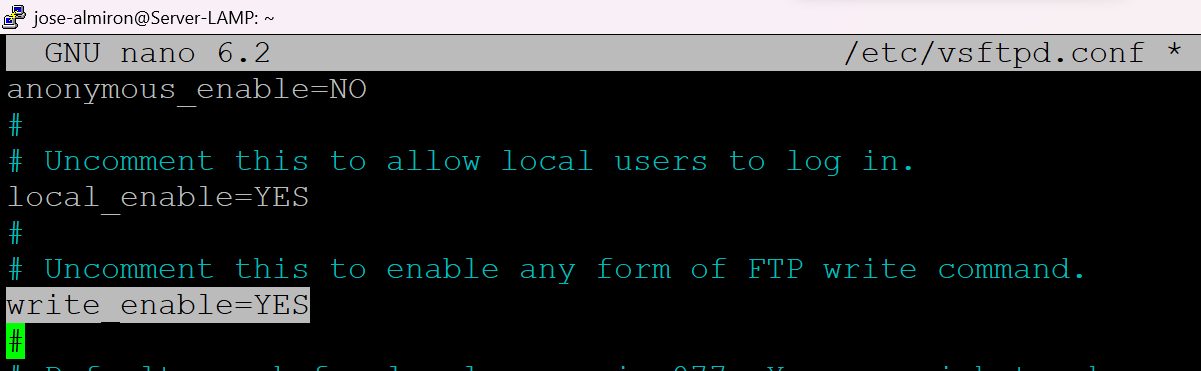


Permitimos el trafico FTP desde el firewall

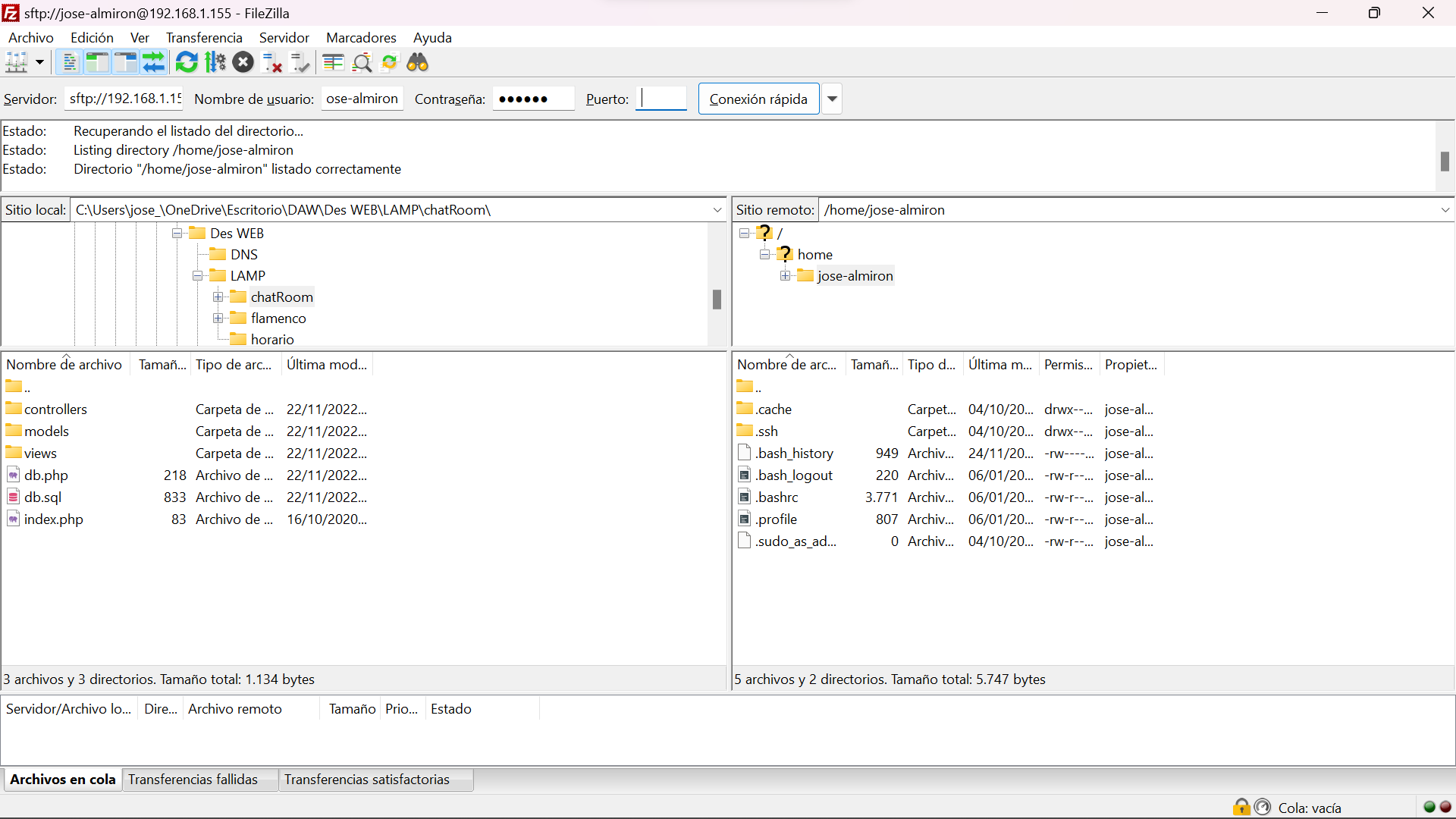


configuramos el servicio vsftpd a través de su fichero de configuración

descomentamos la línea write\_enable



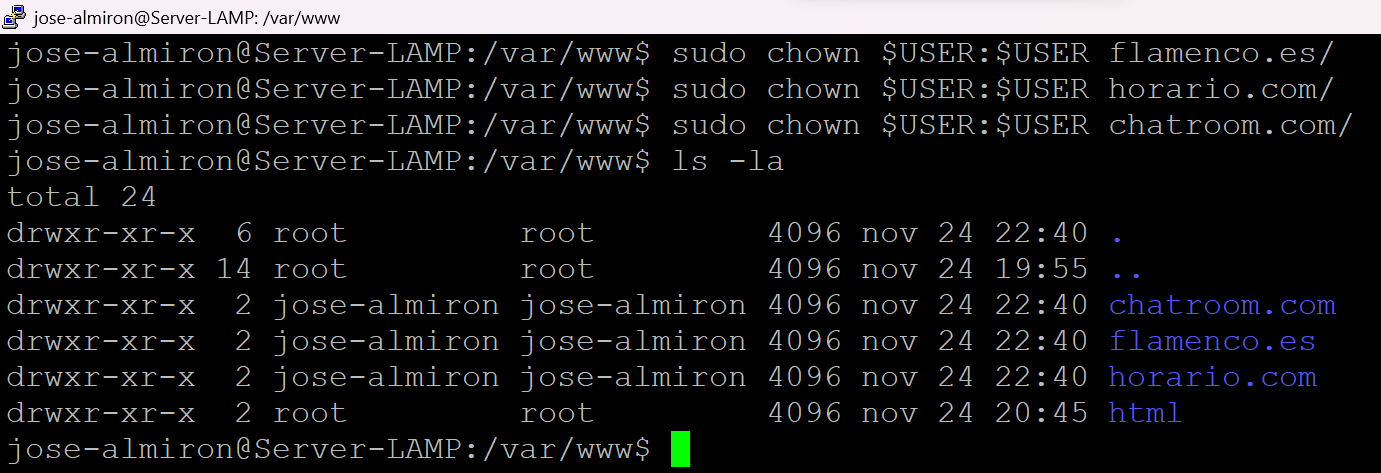
y comprobamos la conexión con filezilla



Ahora si procedemos a la creación de ls directorios que almacenaran las webs en la ruta /var/www



Como vamos a trabajar con ellas le asignamos permisos a nuestro usuario

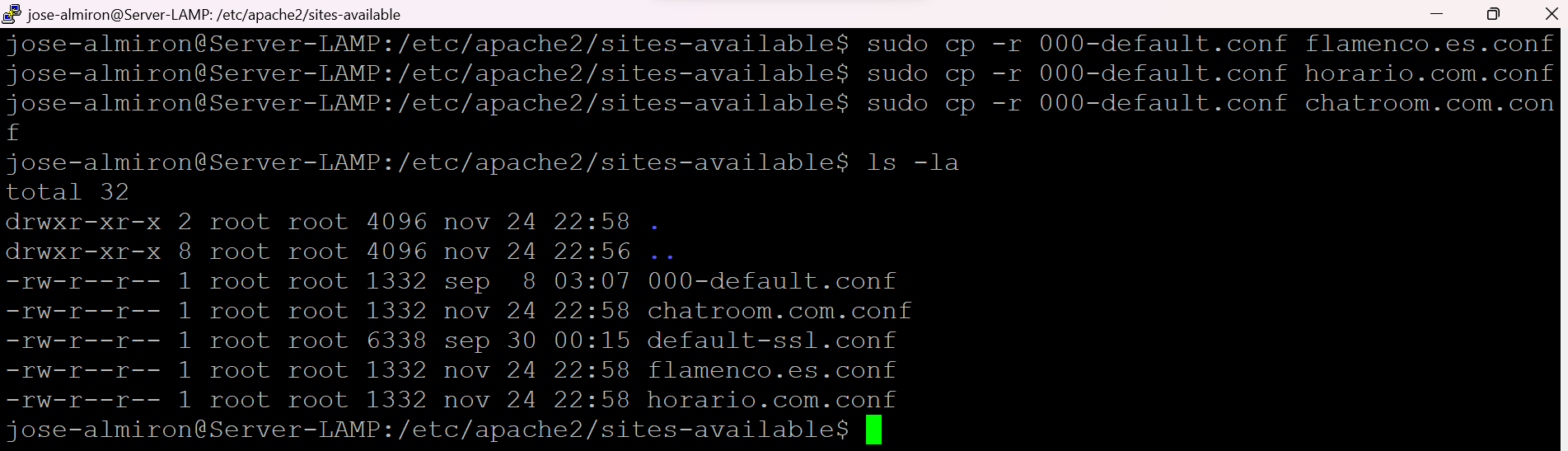


Nos conectamos por medio de FTP y transferimos la web de nuestro equipo al servidor

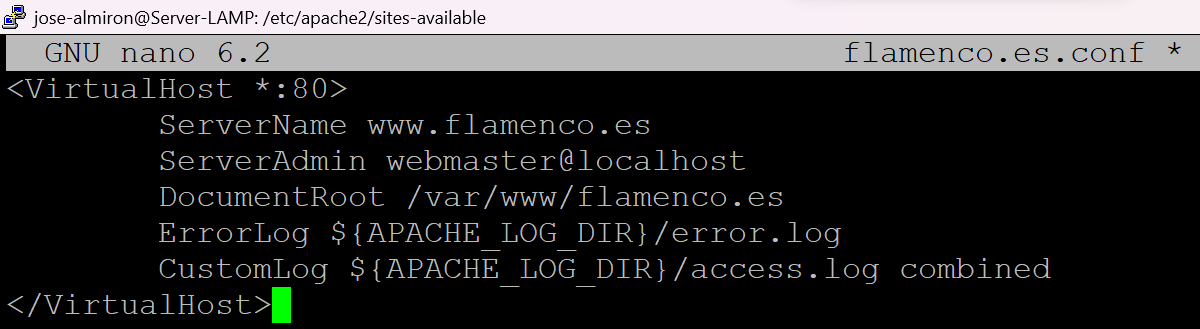




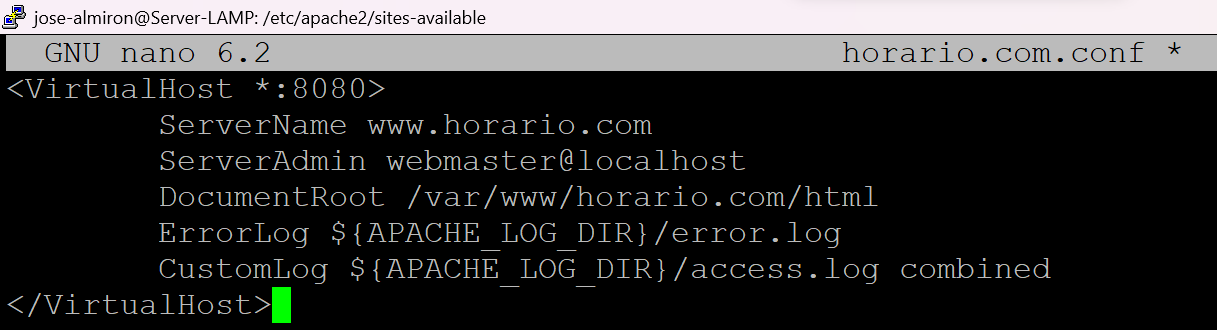
Creamos ficheros de alojamiento virtual para las 3 webs



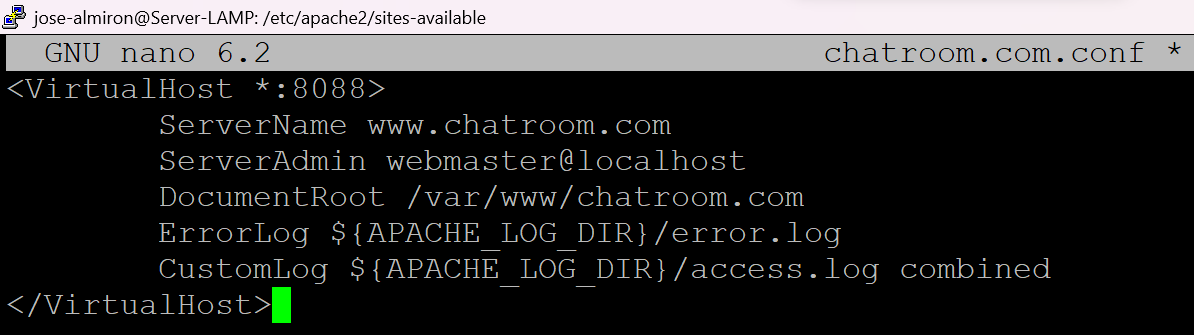
Una vez creados los configuramos



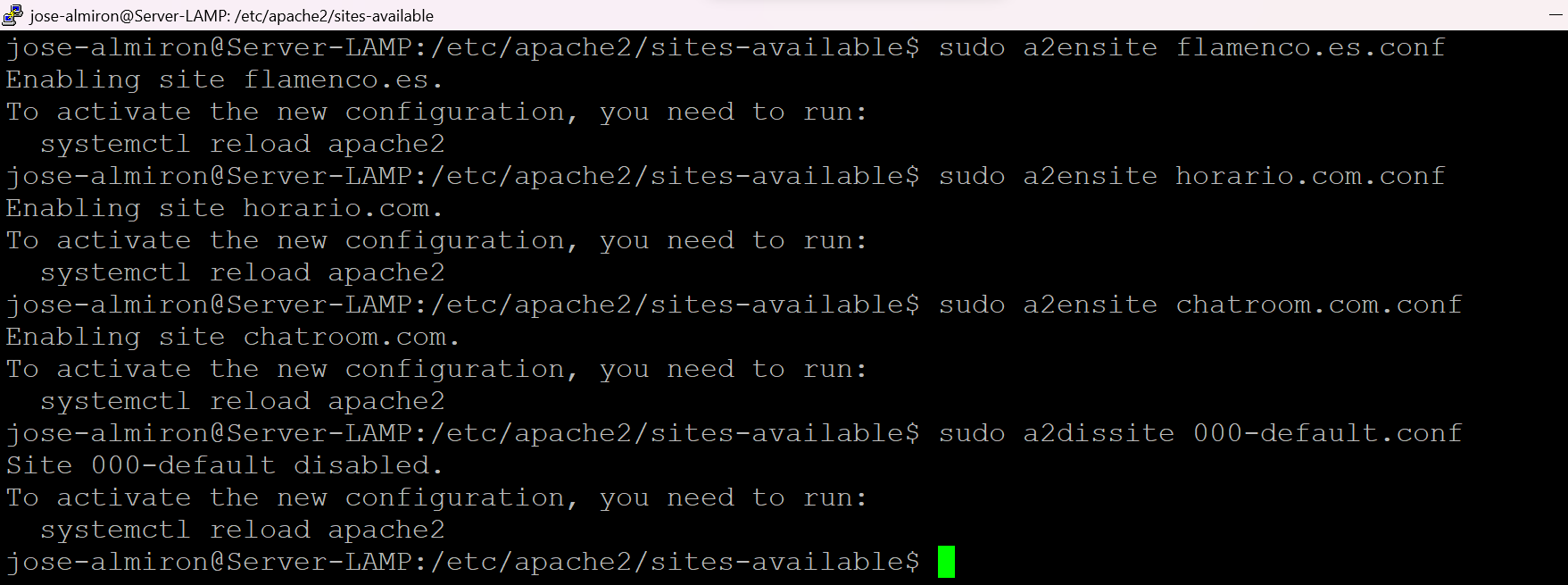
Le asignamos el puerto 8080



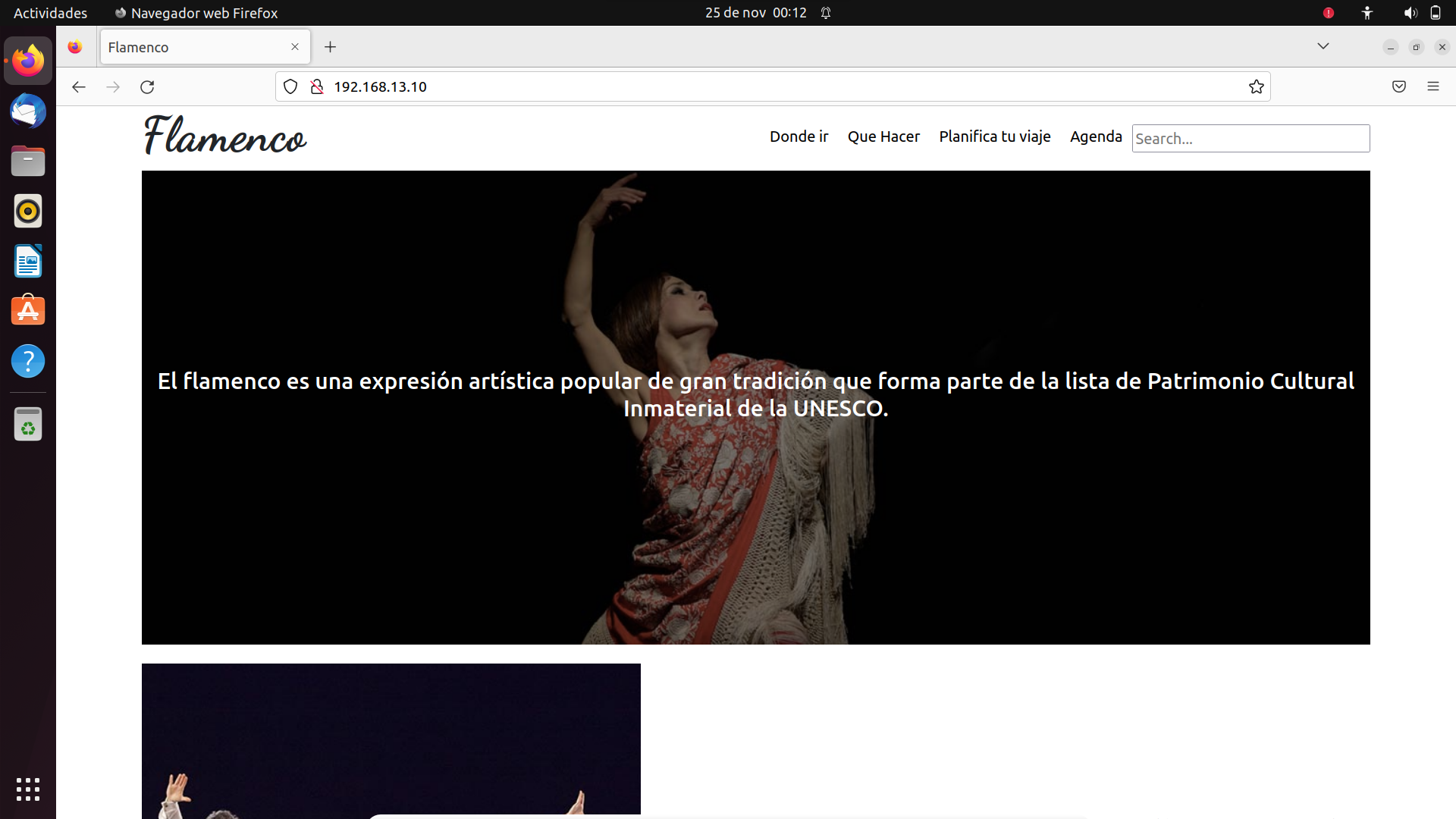
Le asignamos el puerto 8088



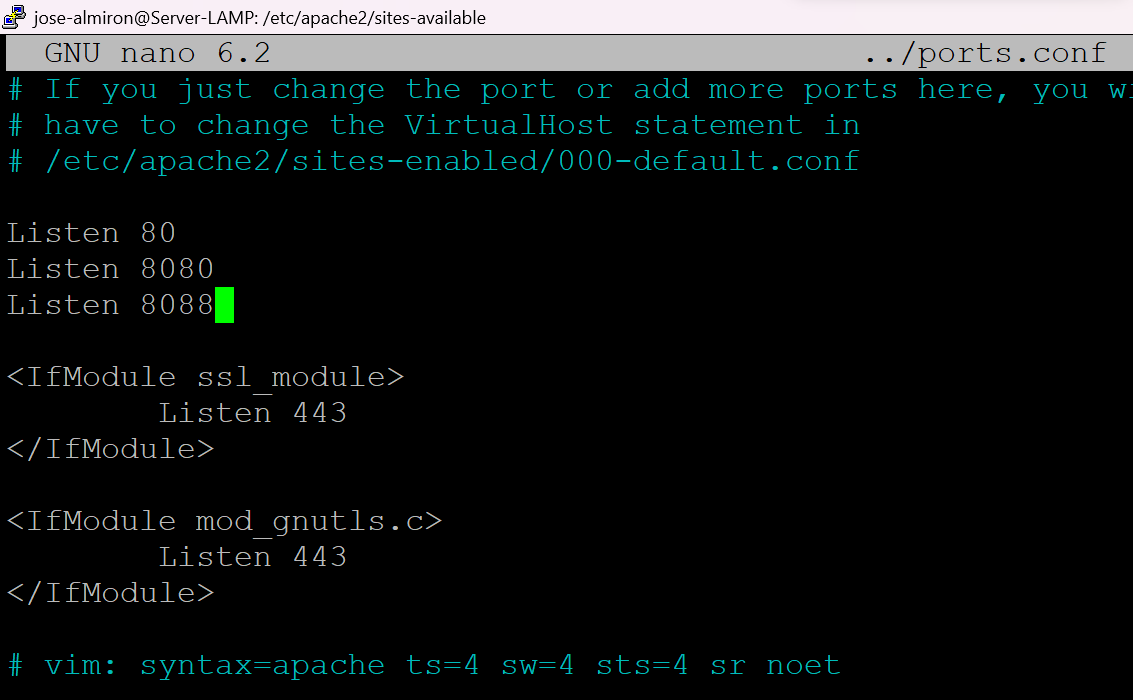
Habilitamos los ficheros para que pasen al directorio de sites-enabled, y deshabilitamos el fichero por defecto



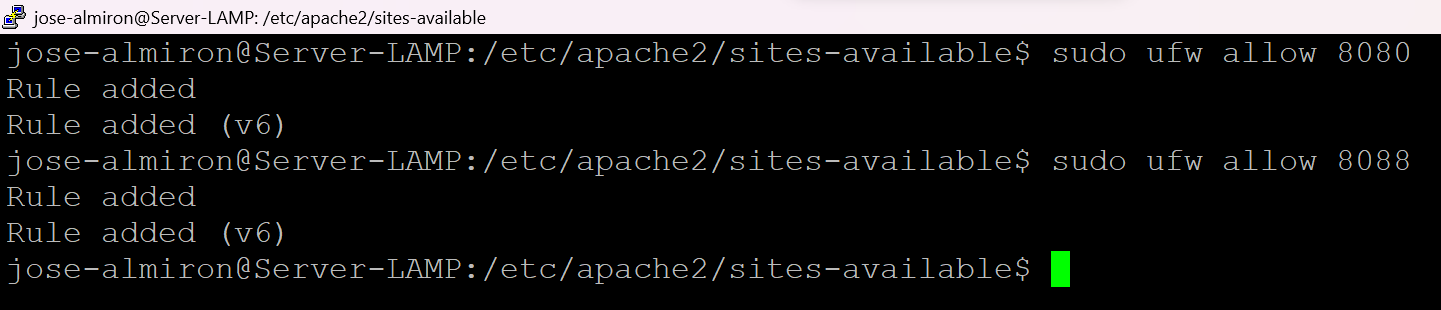
Ahora si reiniciamos el servicio de apache y comprobamos la carga de las webs



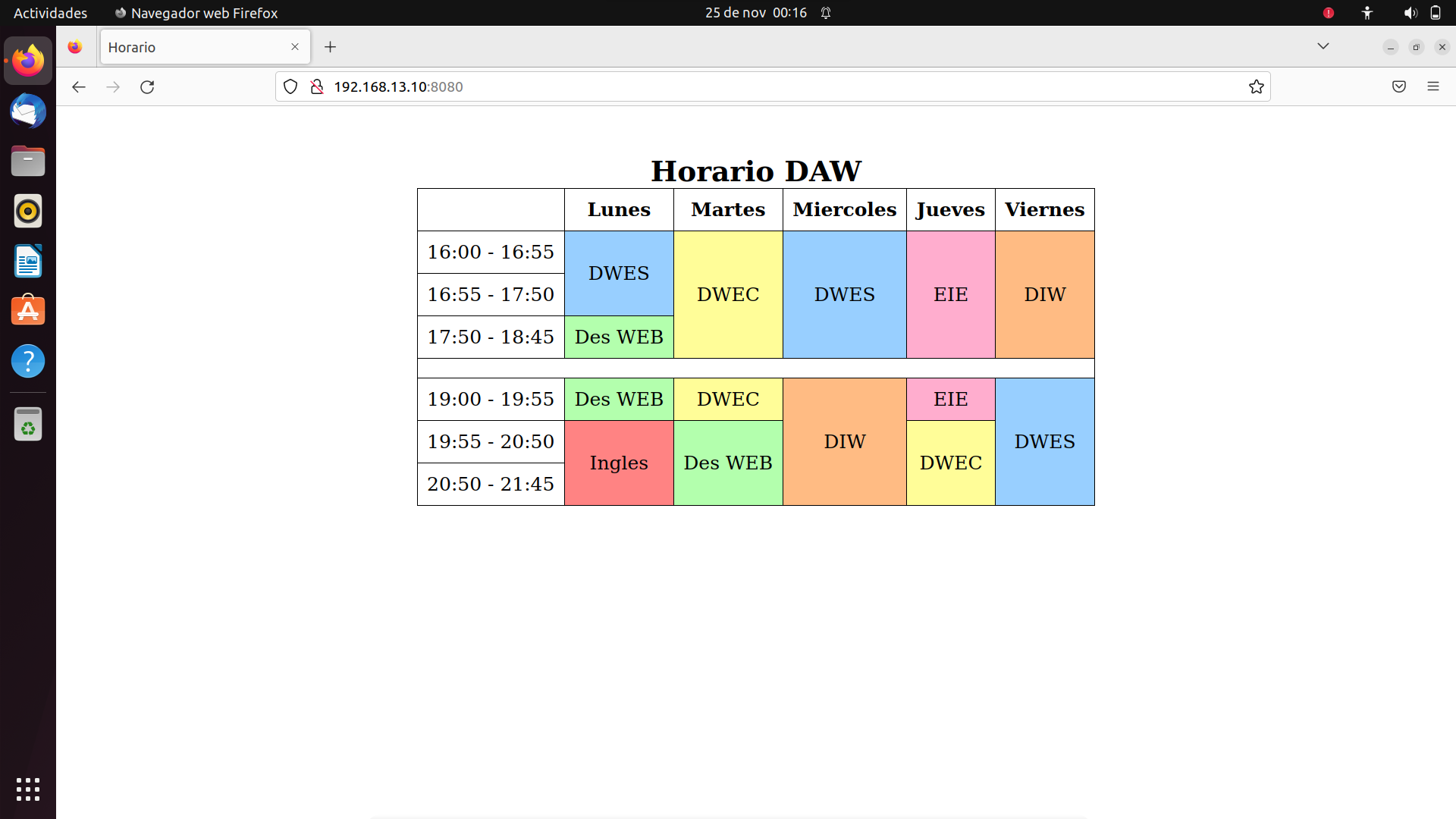
Para las otras webs tendremos que configurar el archivo de puertos y habilitar dichos puertos en el firewall



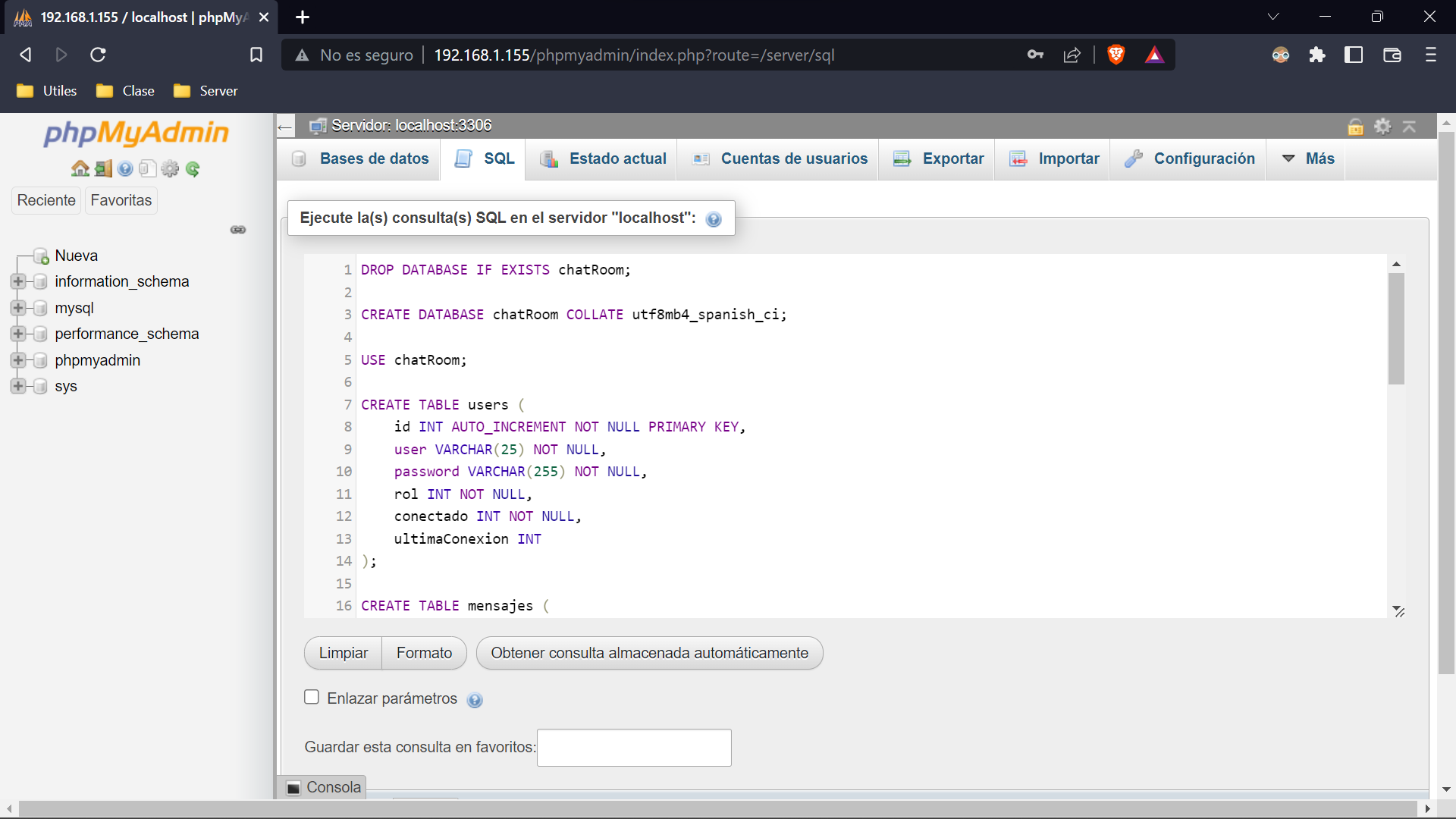
Habilitamos el trafico de los puertos por el firewall



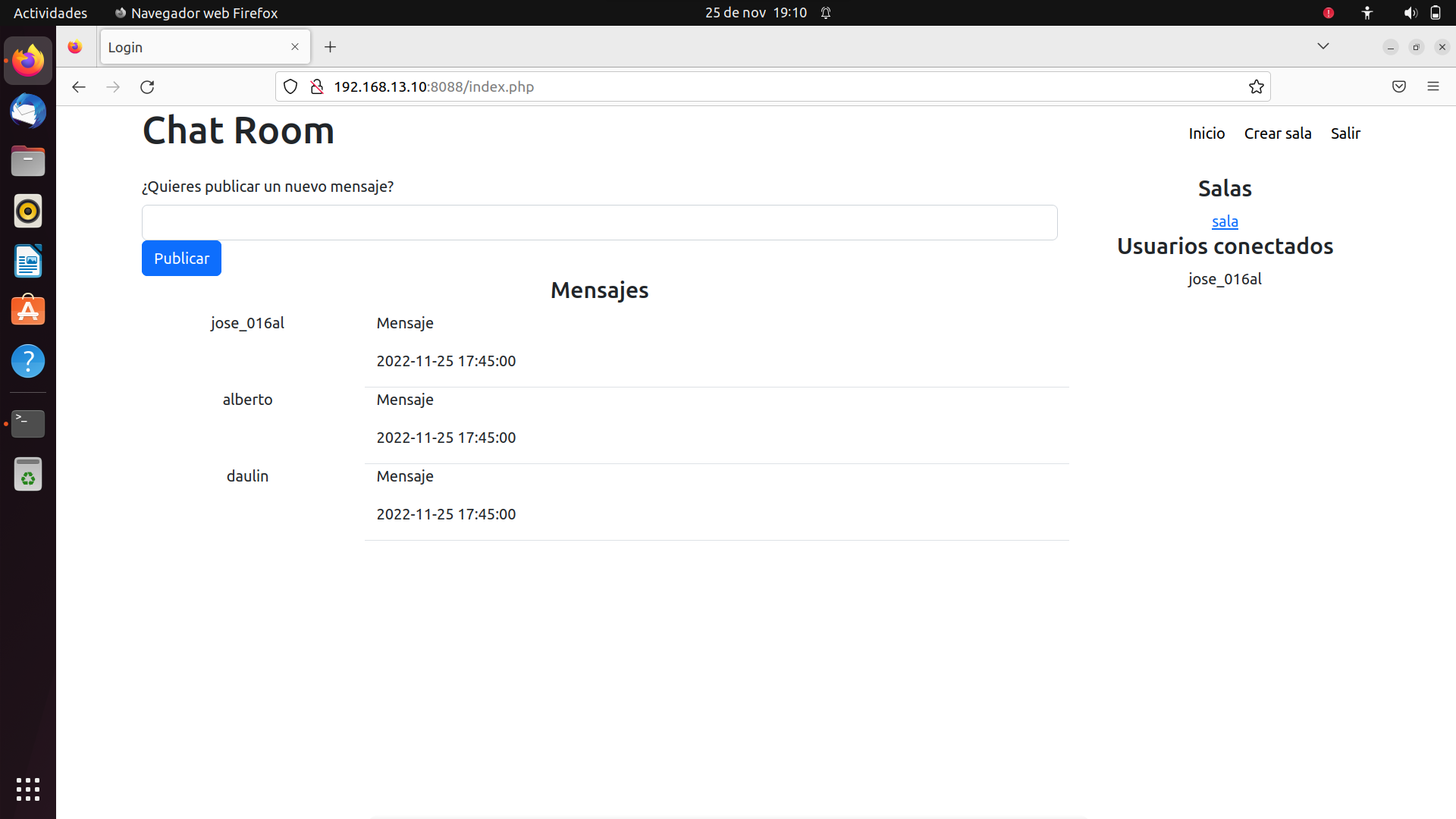
Ahora si podemos ver que la web de horario y chatroom cargan correctamente



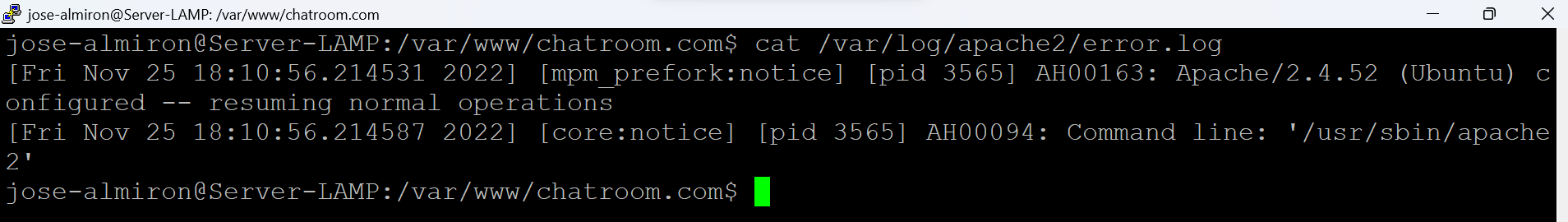
Para la web de chatroom tendremos que crear las tablas de SQL que se proporcionan en su directorio



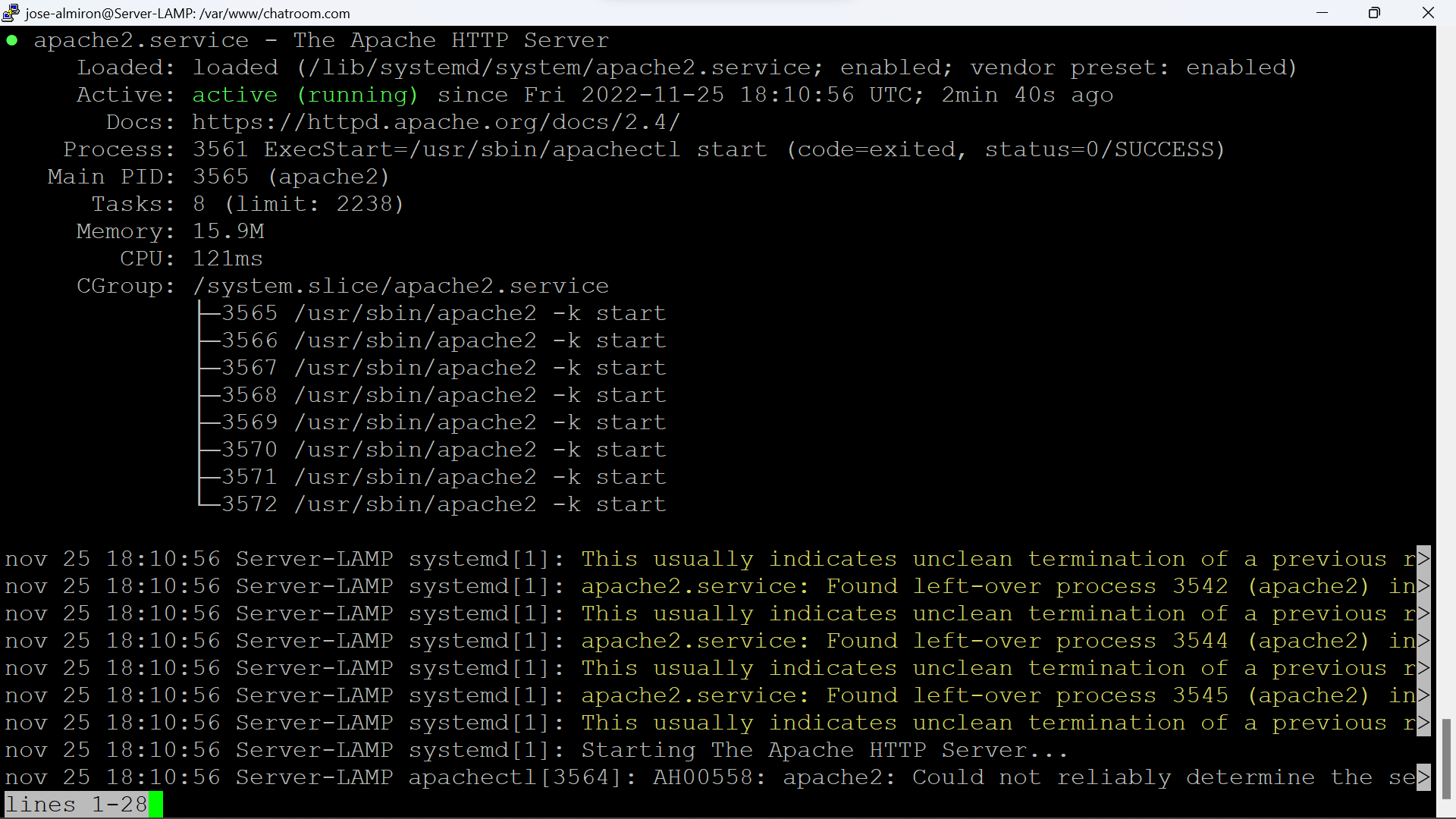
Una vez ejecutado el script ya podremos visualizar la web



Mostramos el archivo de errores /var/log/apache2/error.log

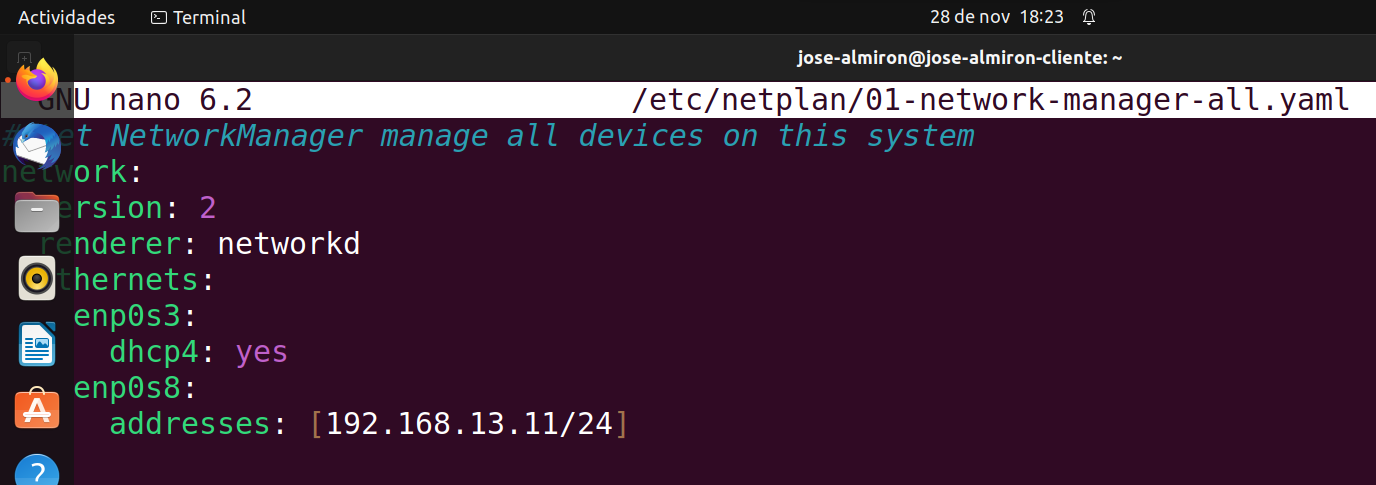


Comprobamos el estado del servicio

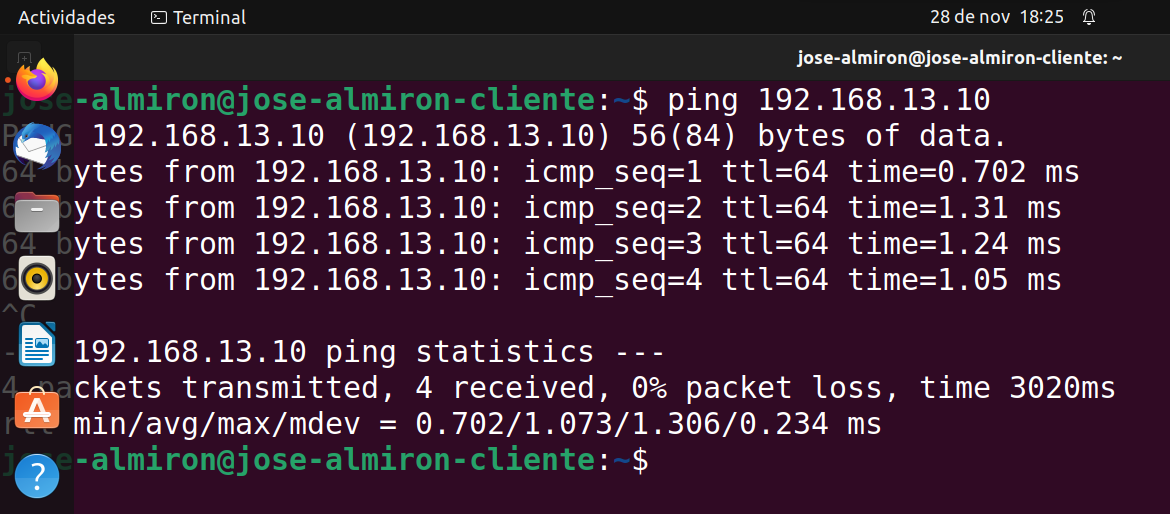


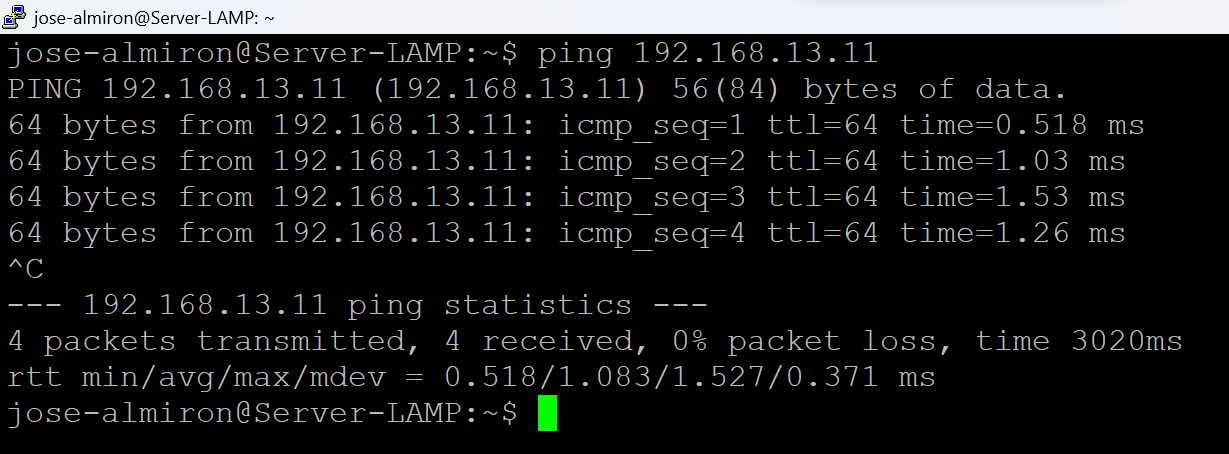
# Comprobaciones en el cliente

La configuración de red del cliente



Comprobamos la conexión entre ambas maquinas

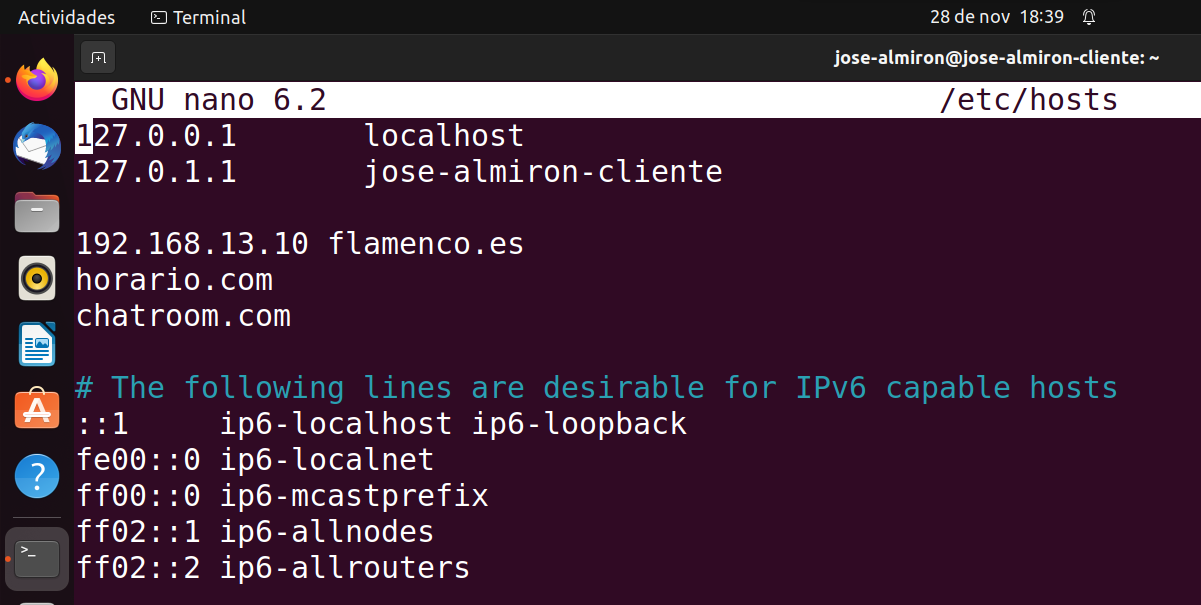




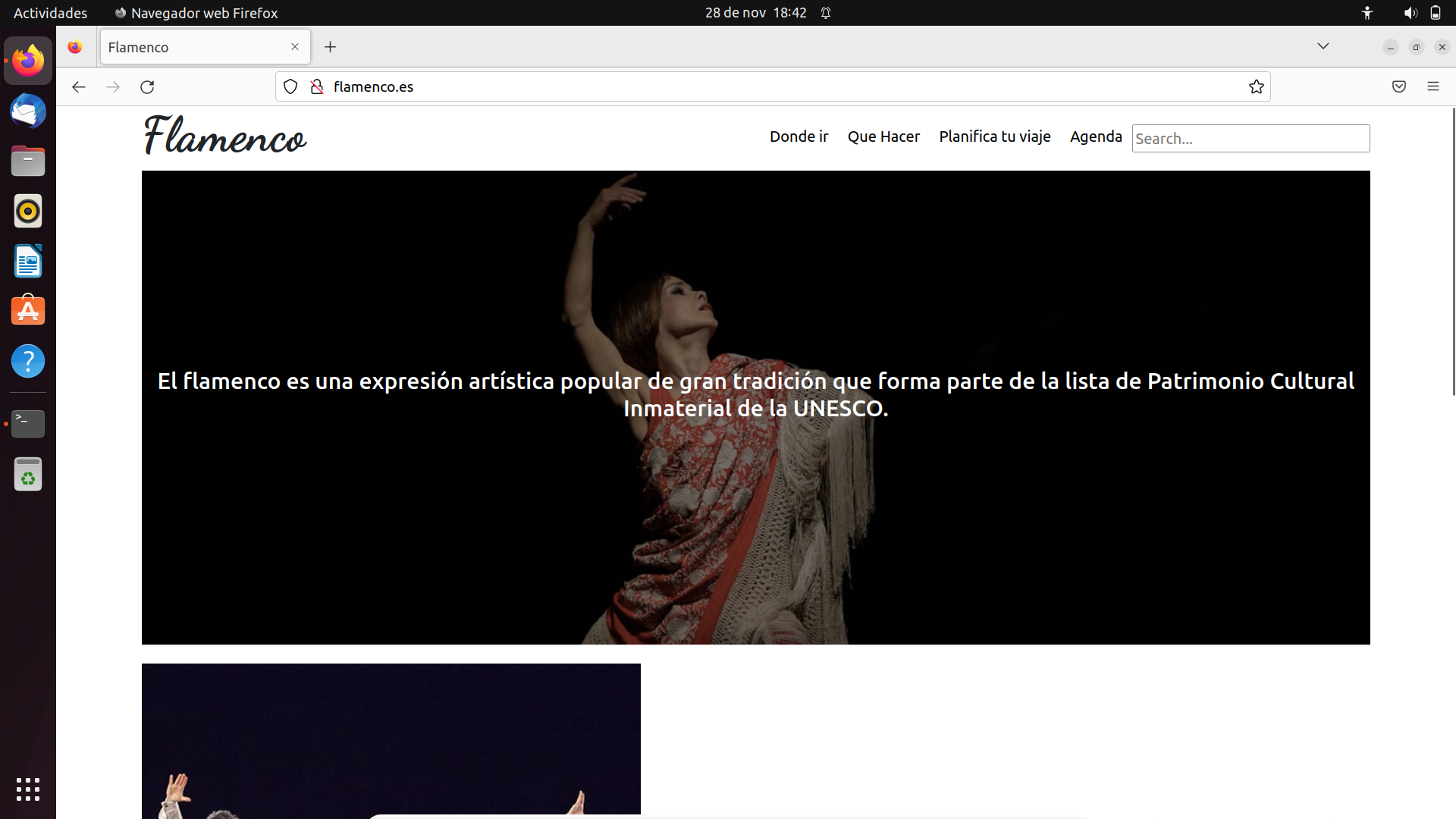
Carga de la pagina principal desde el cliente

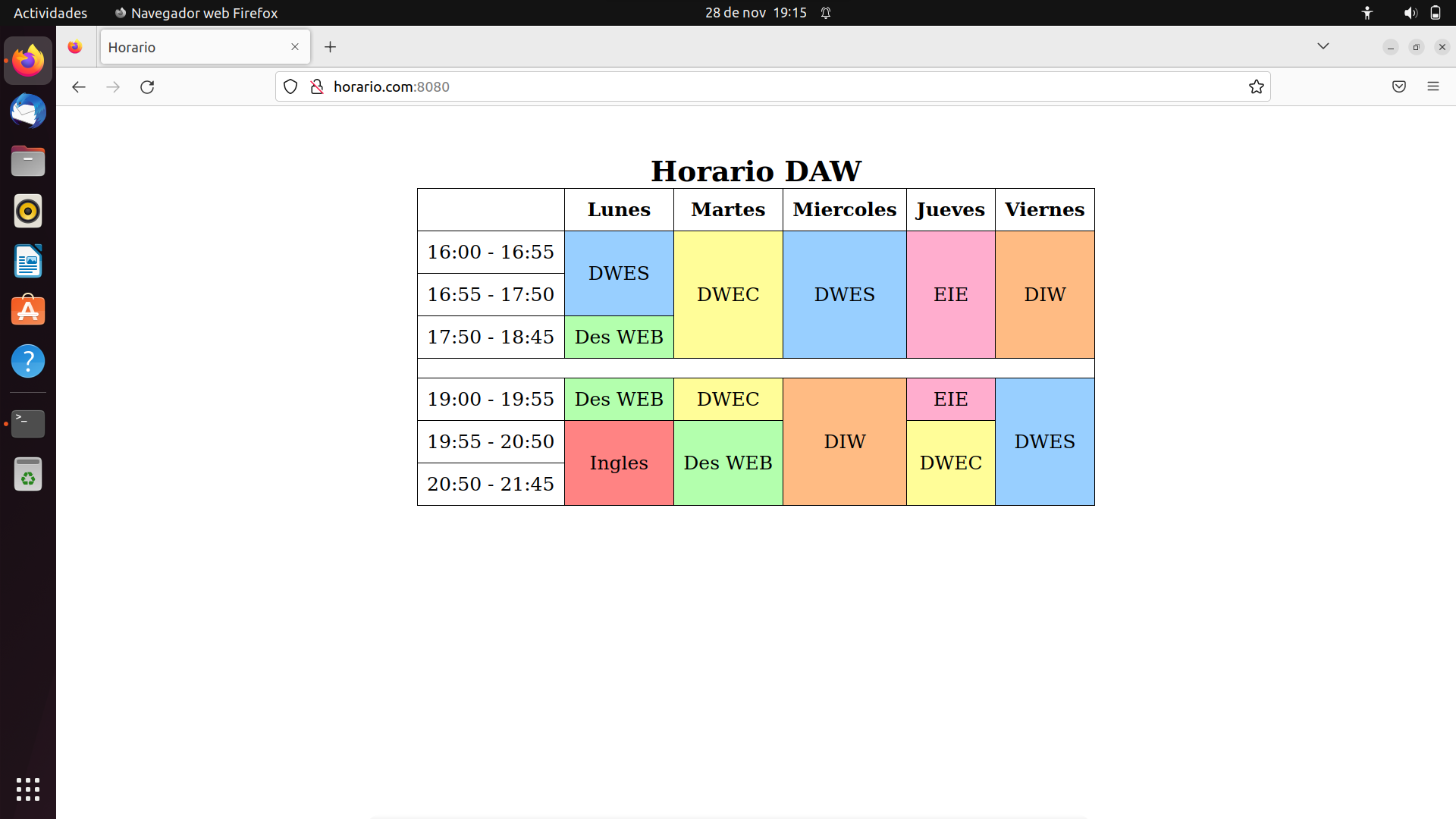


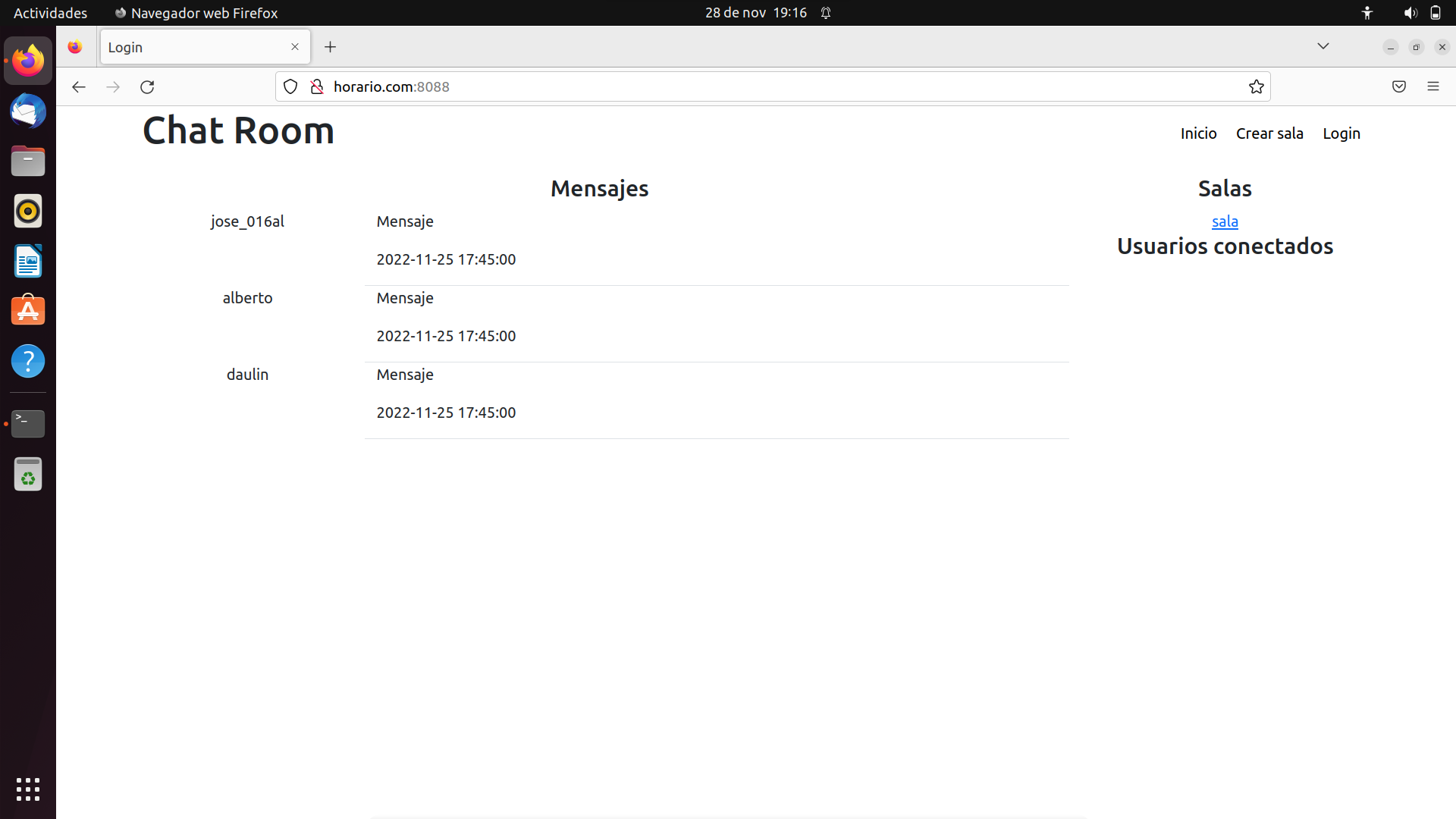
Para solventar la no configuración de un DNS



Mostramos carga de la pagina por medio de nombre de dominio







Para tener acceso ssh se abre el puerto Openssh en el firewall y nos ocnectamos por putty

