1. No hay ninguno activo
2. **Port 22**:  
   Esto establece que el servidor SSH escuchará conexiones en el puerto 22

**Protocol 2**:  
Especifica que el servidor solo permitirá el protocolo SSH versión 2, que es más seguro que la versión 1.

**HostKey /etc/ssh/ssh\_host\_rsa\_key**  
**HostKey /etc/ssh/ssh\_host\_dsa\_key**  
**HostKey /etc/ssh/ssh\_host\_ecdsa\_key**  
**HostKey /etc/ssh/ssh\_host\_ed25519\_key**:  
Estas líneas definen las rutas a las claves host del servidor SSH.

**UsePrivilegeSeparation yes**:  
Esta opción mejora la seguridad al ejecutar el proceso SSH con privilegios limitados durante el proceso de conexión, lo que reduce los daños en caso de un ataque.

**KeyRegenerationInterval 3600**:  
Especifica que el servidor SSH regenerará su clave cada 3600 segundos.

**ServerKeyBits 1024**:  
Define el tamaño de la clave temporal del servidor en bits.

**SyslogFacility AUTH**:  
Esto indica que los mensajes relacionados con la autenticación se registrarán a través de la instalación AUTH del syslog.

**LogLevel INFO**:  
Define el nivel de registro o logging. INFO proporciona un detalle moderado en los registros, lo suficiente para monitorear las conexiones SSH sin ser demasiado detallado.

**LoginGraceTime 120**:  
Establece el tiempo en segundos que un usuario tiene para iniciar sesión con éxito antes de que el servidor desconecte la sesión.

**PermitRootLogin prohibit-password**:  
Esta opción controla si el usuario root puede iniciar sesión a través de SSH. prohibit-password significa que el inicio de sesión como root solo se permite utilizando autenticación por clave pública, pero no mediante contraseña.

**StrictModes yes**:  
Al estar en yes, SSH verifica los permisos y la propiedad de los archivos y directorios de los usuarios para prevenir posibles problemas de seguridad. Asegura que el directorio personal y los archivos relacionados con SSH no sean escribibles por otros usuarios.

**RSAAuthentication yes**:  
Esto habilita la autenticación basada en RSA para los usuarios.

**PubkeyAuthentication yes**:  
Esto permite la autenticación mediante clave pública para los usuarios de SSH, lo que mejora la seguridad al usar claves criptográficas en lugar de contraseñas.

**IgnoreRhosts yes**:  
Con esta configuración, el servidor SSH ignorará los archivos ~/.rhosts y ~/.shosts de los usuarios. Estos archivos eran utilizados antiguamente para autenticación basada en hosts, lo cual es inseguro y por eso se deshabilita.

**HostbasedAuthentication no**:  
Esta opción deshabilita la autenticación basada en hosts. En este caso, no se confiará en los hosts que se encuentran en el archivo ~/.ssh/known\_hosts.

**PermitEmptyPasswords no**:  
Indica que las cuentas de usuario que tengan contraseñas vacías no pueden iniciar sesión a través de SSH. Esto está deshabilitado por razones de seguridad.

**ChallengeResponseAuthentication no**:  
Desactiva la autenticación basada en desafíos y respuestas, como puede ser la autenticación mediante preguntas o ciertos módulos PAM.

**X11Forwarding yes**:  
Activa el reenvío X11, permitiendo que las aplicaciones gráficas del servidor remoto se muestren en el cliente local.

**X11DisplayOffset 10**:  
Especifica el valor de desplazamiento para los números de pantalla de las conexiones X11 reenviadas.

**PrintMotd no**:  
Desactiva la impresión del archivo "Message of the Day" (motd) al iniciar sesión.

**PrintLastLog yes**:  
Habilita la impresión del último inicio de sesión exitoso de un usuario al iniciar sesión a través de SSH.

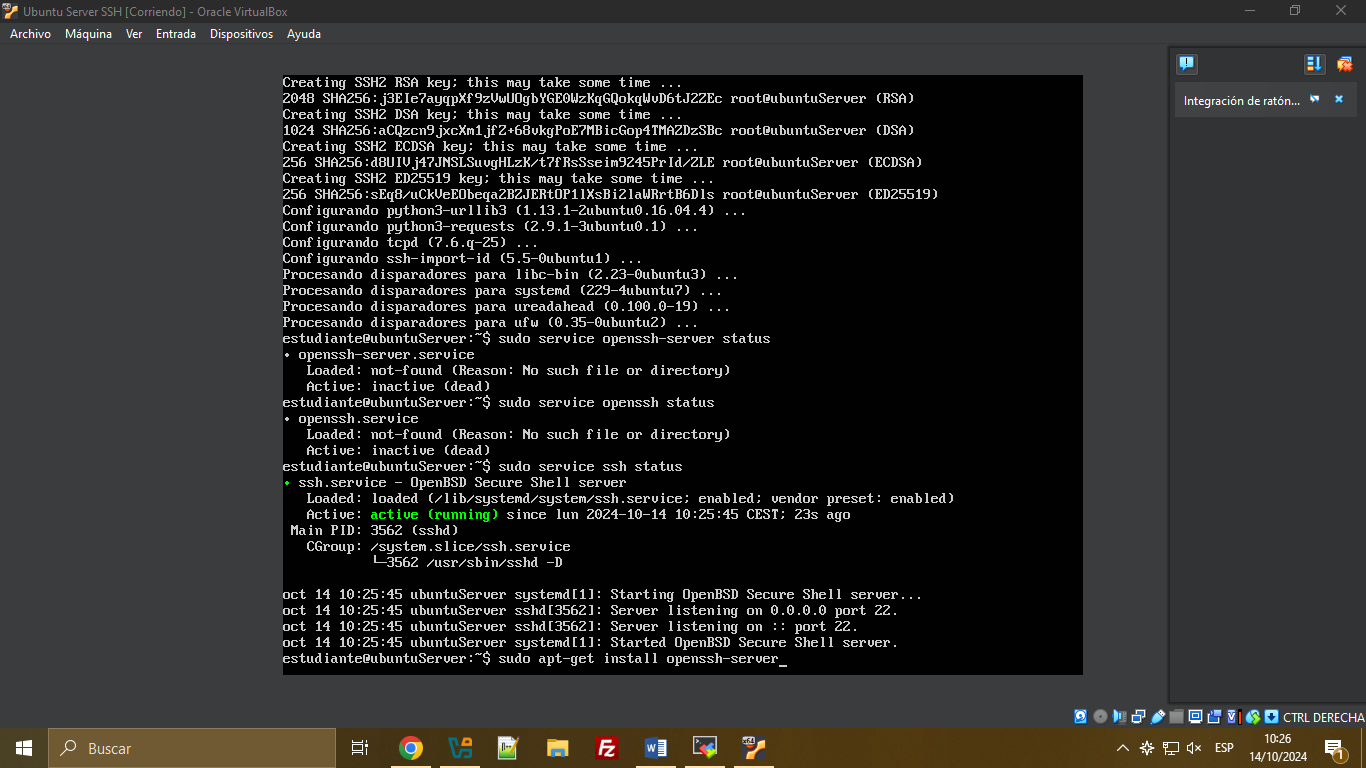
**TCPKeepAlive yes**:  
Activa la opción TCPKeepAlive, enviando paquetes para mantener viva la conexión y detectar fallos de red.

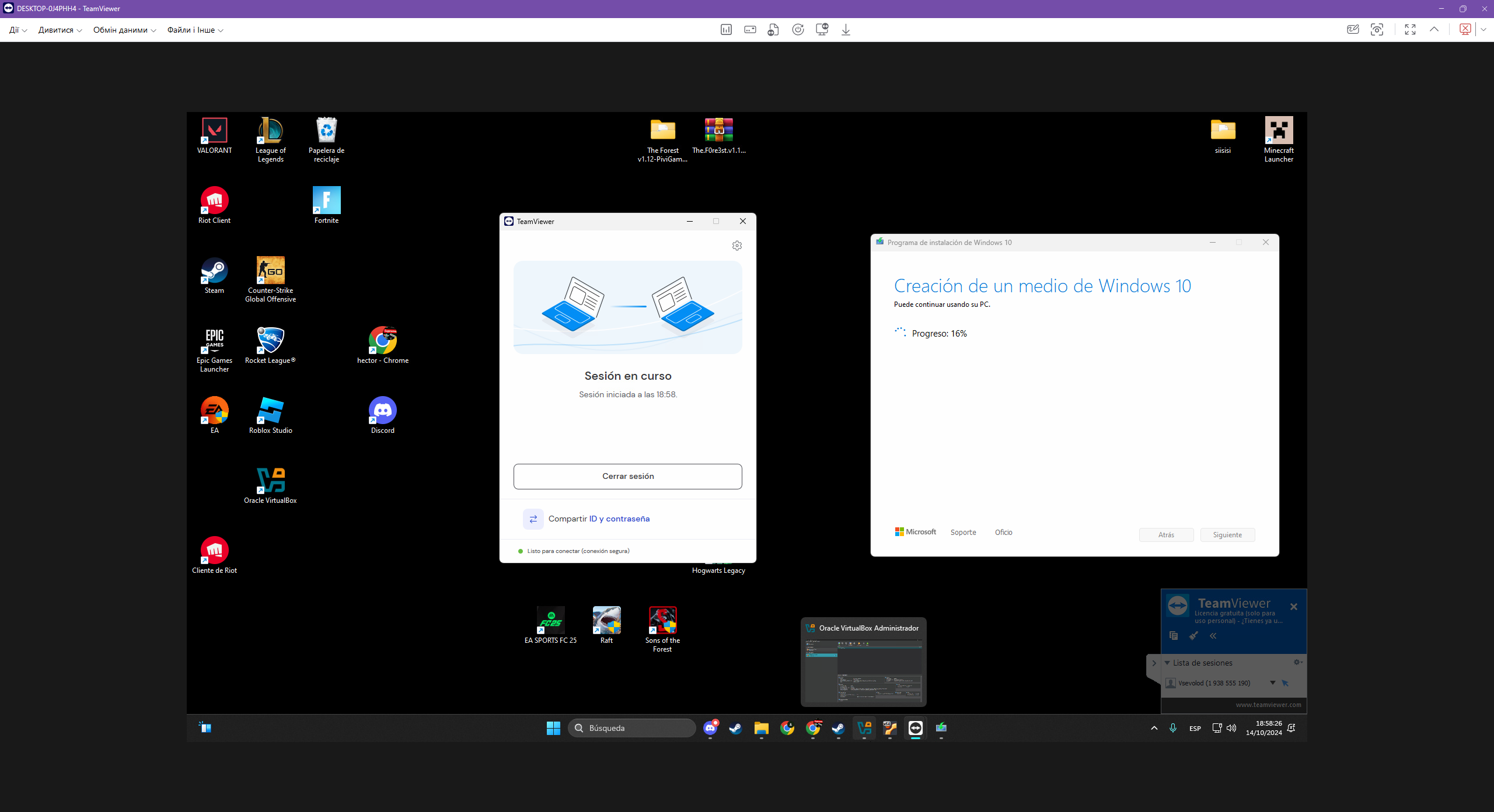
**AcceptEnv LANG LC\_\*:**  
Permite que el cliente SSH envíe variables de entorno relacionadas con el lenguaje y localización.

**Subsystem sftp /usr/lib/openssh/sftp-server:**  
Define el subsistema SFTP y especifica la ruta al servidor SFTP. Esto habilita la transferencia segura de archivos utilizando SSH.

**UsePAM yes:**

Habilita el uso de PAM (Pluggable Authentication Modules) para la autenticación, la gestión de sesiones y la configuración de cuentas.



1. Team Viewer
2. Discord

1. Transmisión en Discord:

Abre Discord y únete a un canal de voz.

Haz clic en "Compartir pantalla" y selecciona una ventana o toda la pantalla para transmitir.

Inicia la transmisión y toma capturas de pantalla del proceso.

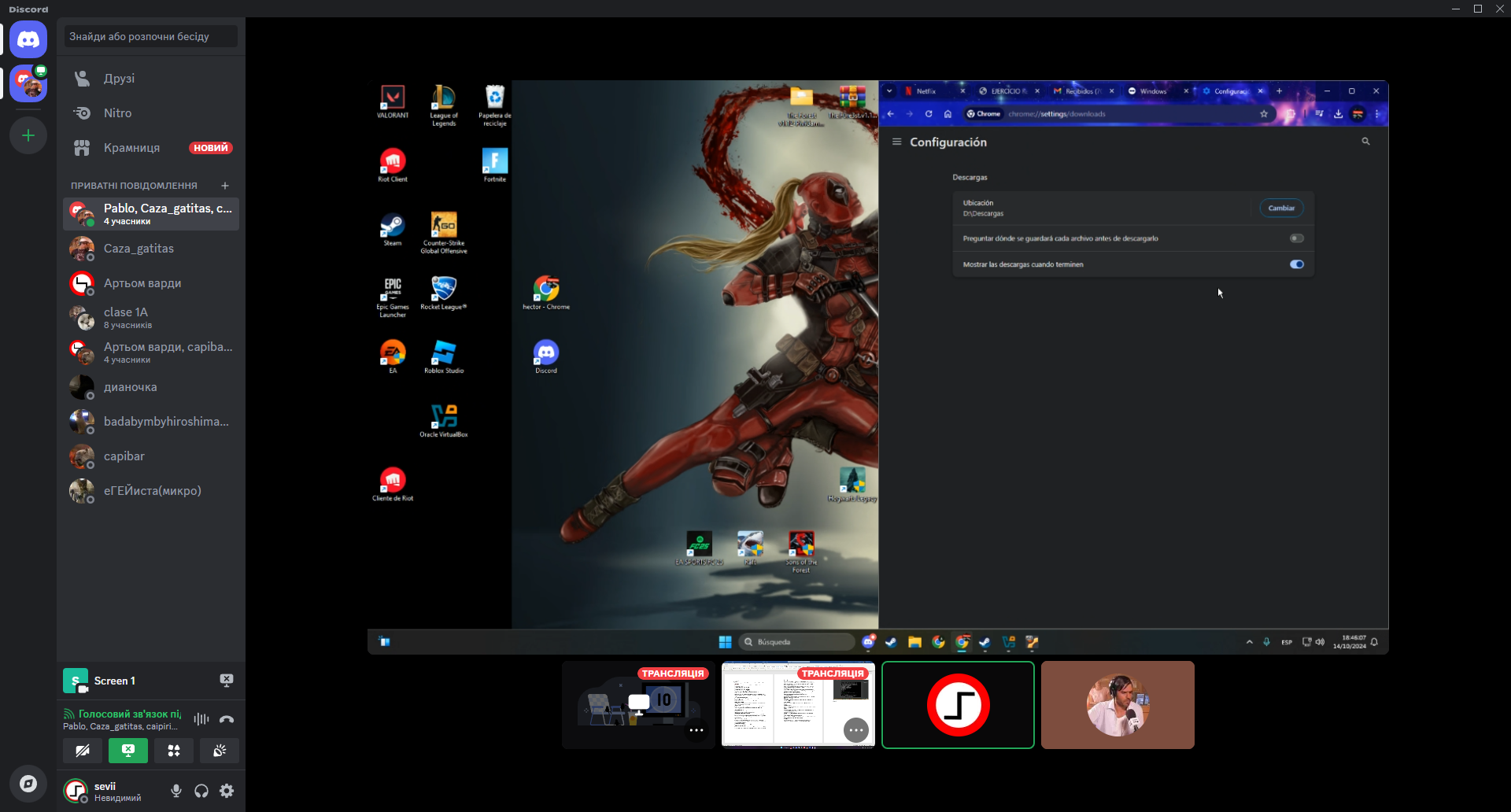
2. Comparación con una aplicación de videoconferencias:

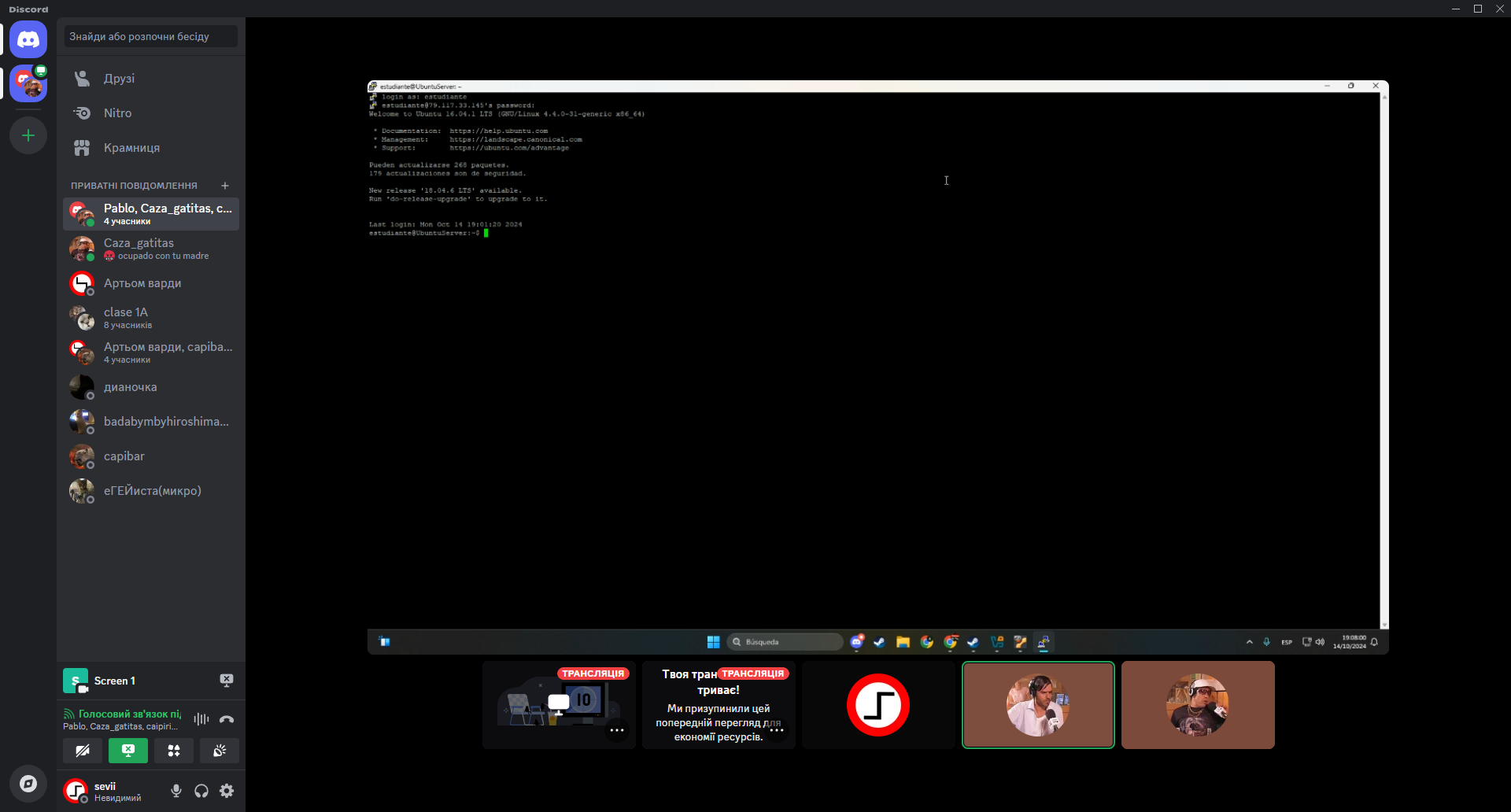
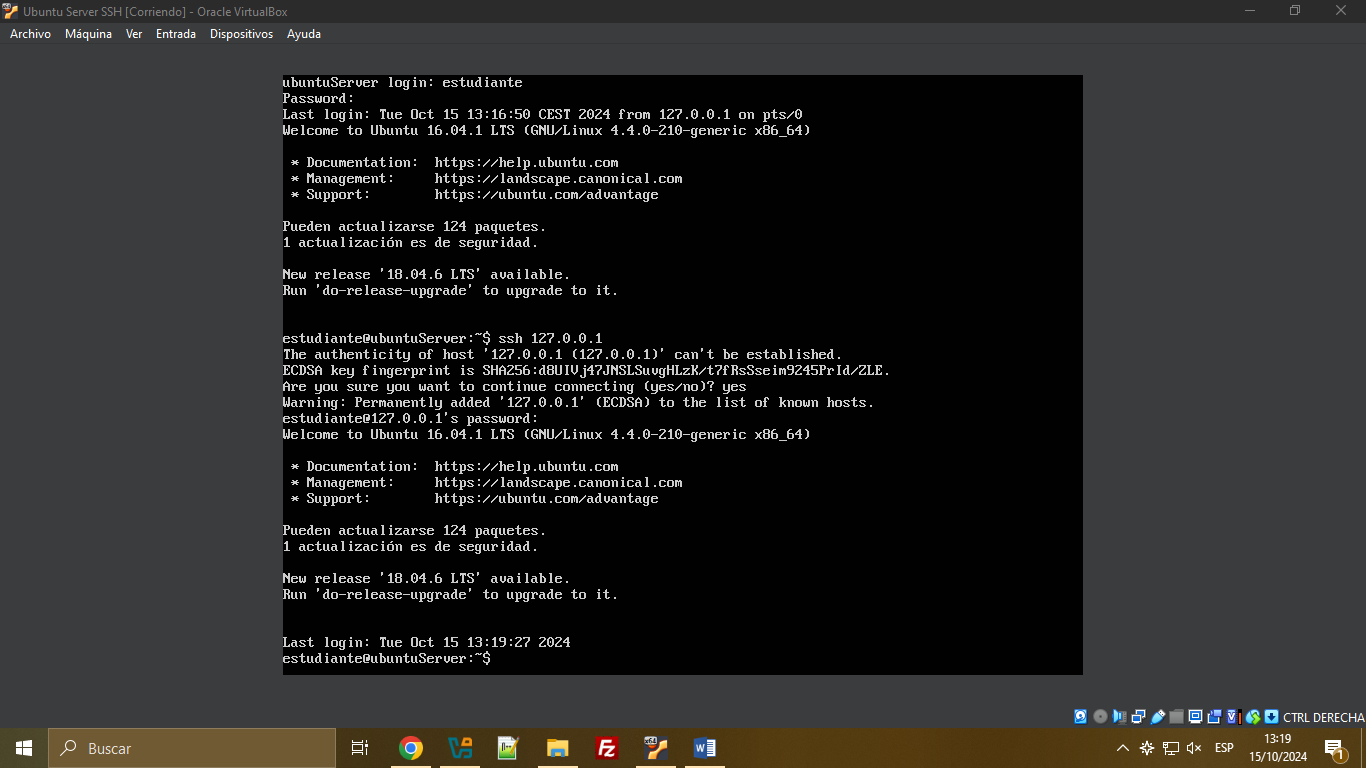
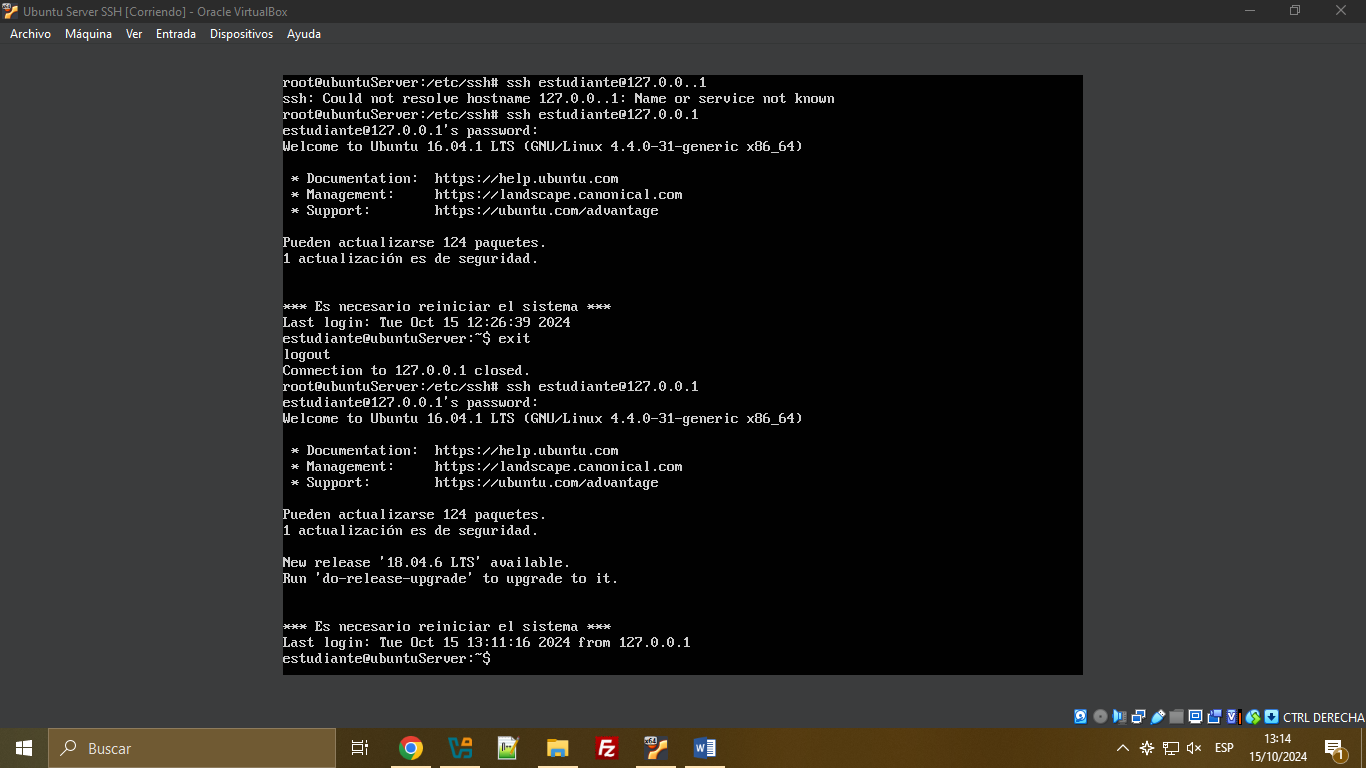
Discord es ideal para comunidades y jugadores, permitiendo crear canales de texto y voz personalizados.

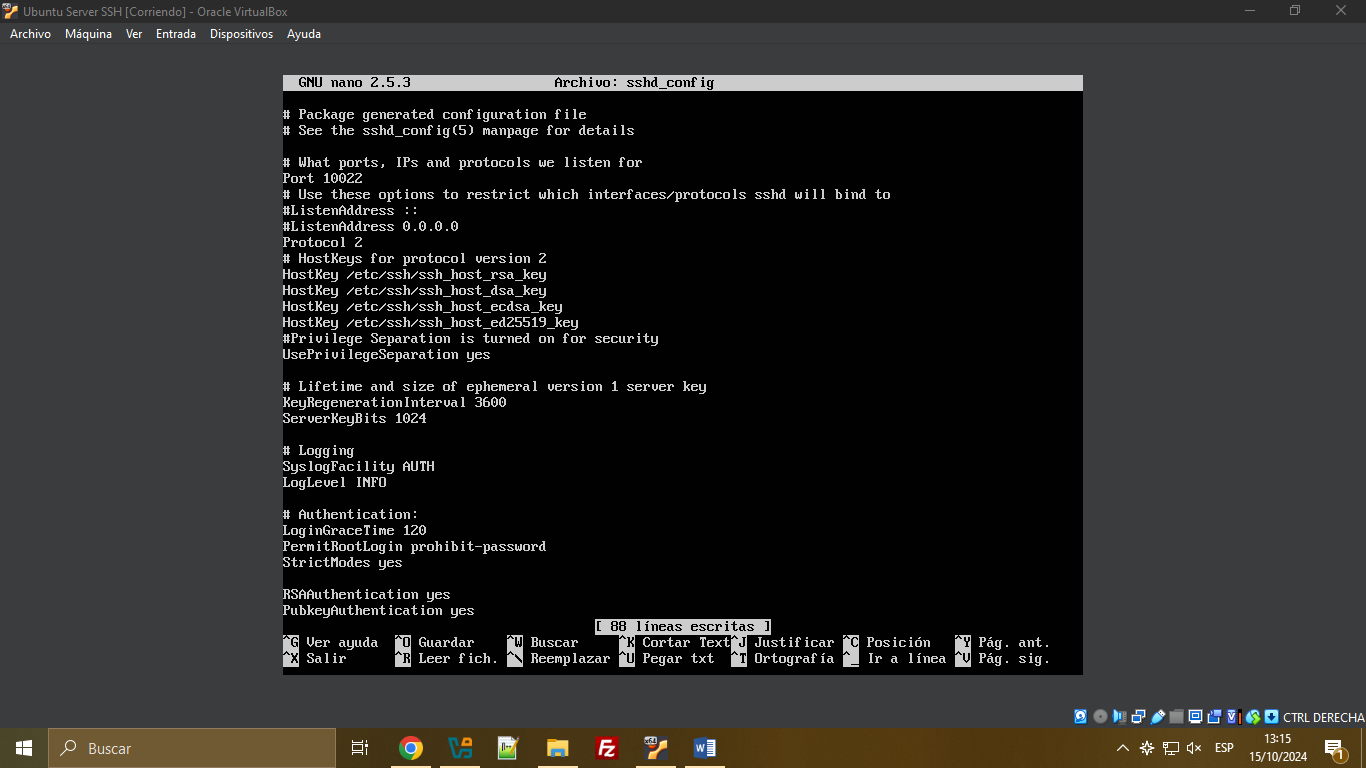
Las aplicaciones de videoconferencias (como Zoom o Meet) están más enfocadas en reuniones formales, con herramientas para gestionar eventos profesionales.

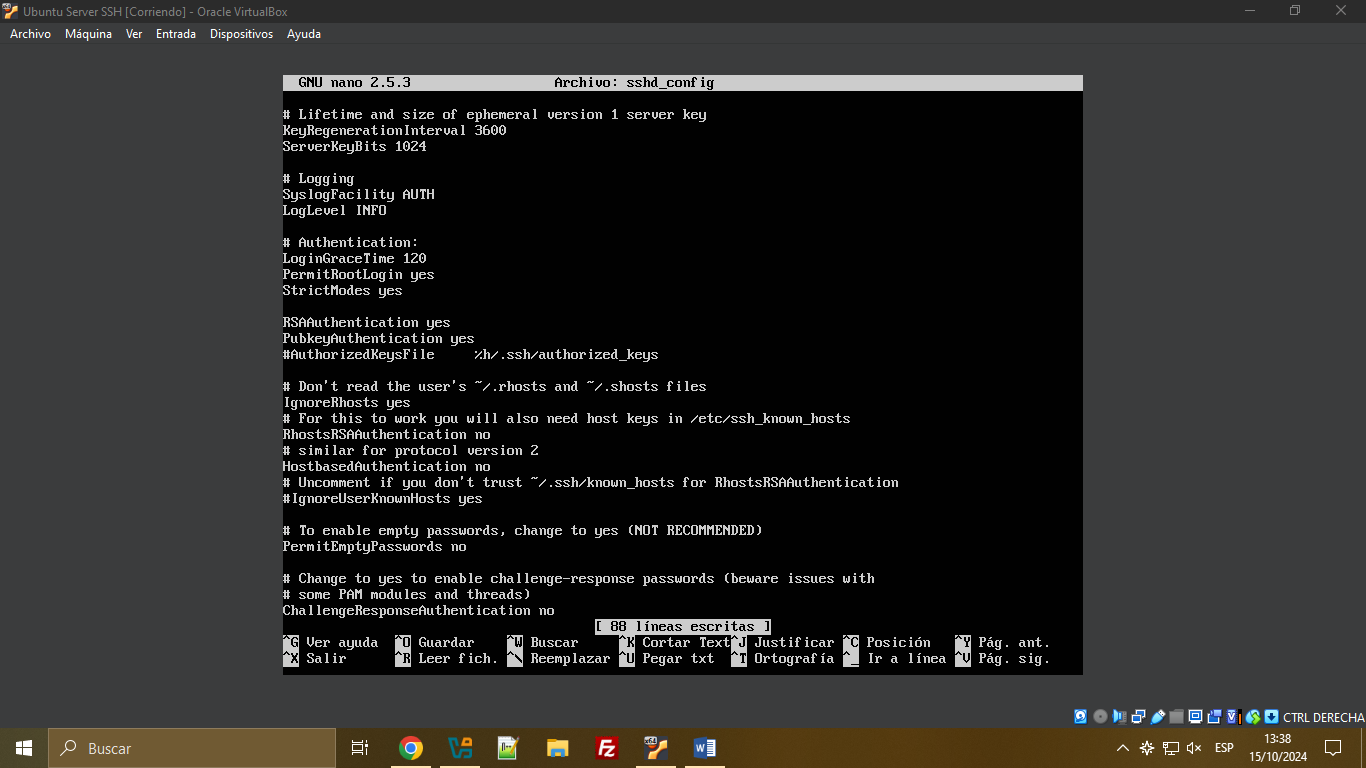
3. Comparación con Telnet y SSH:

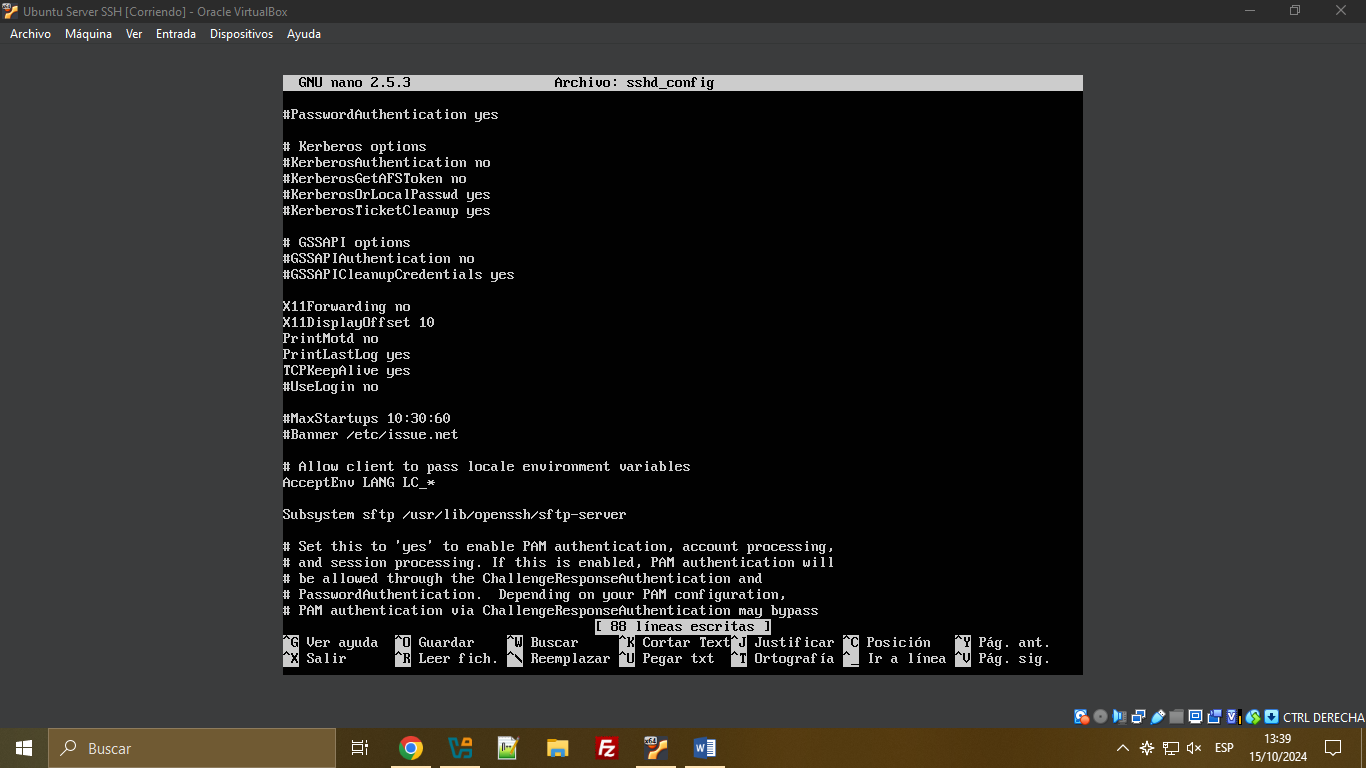
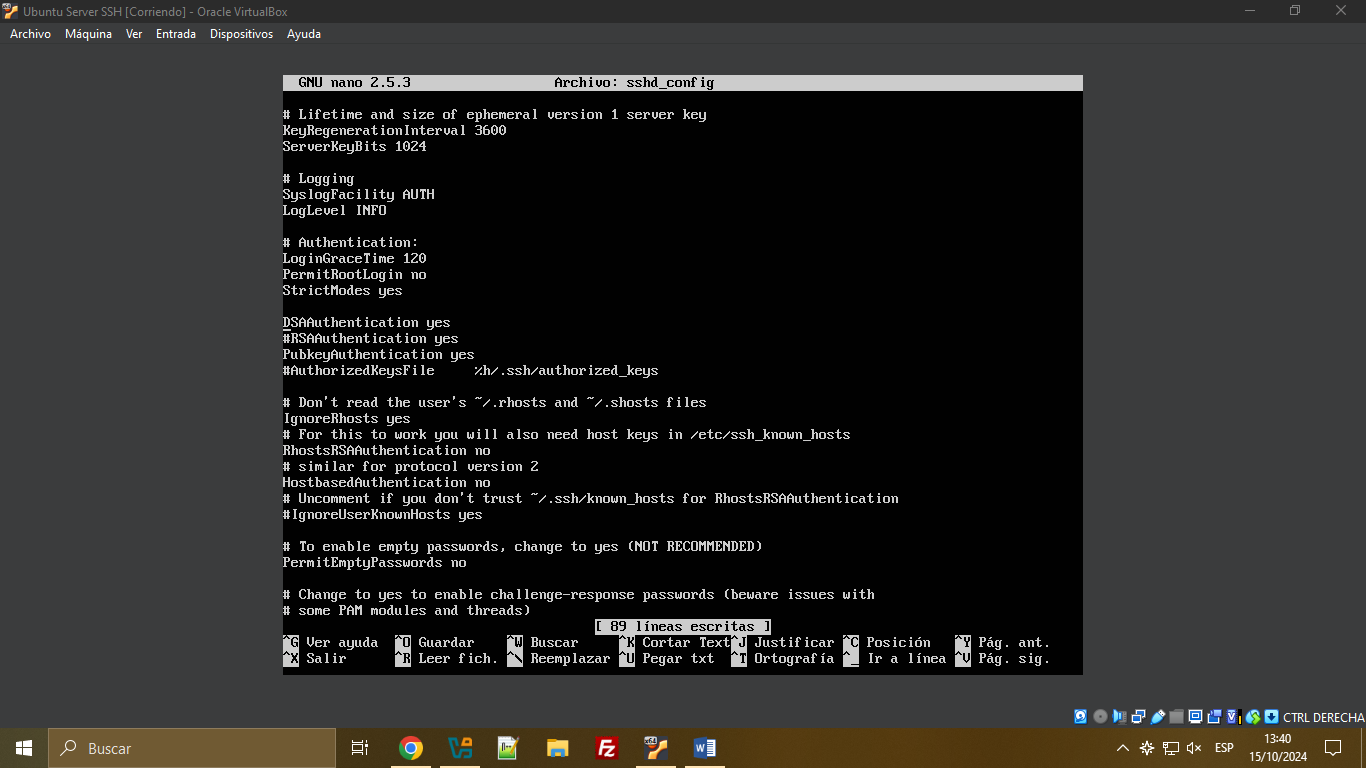
Telnet y SSH son protocolos para acceder a servidores de forma remota a través de comandos.

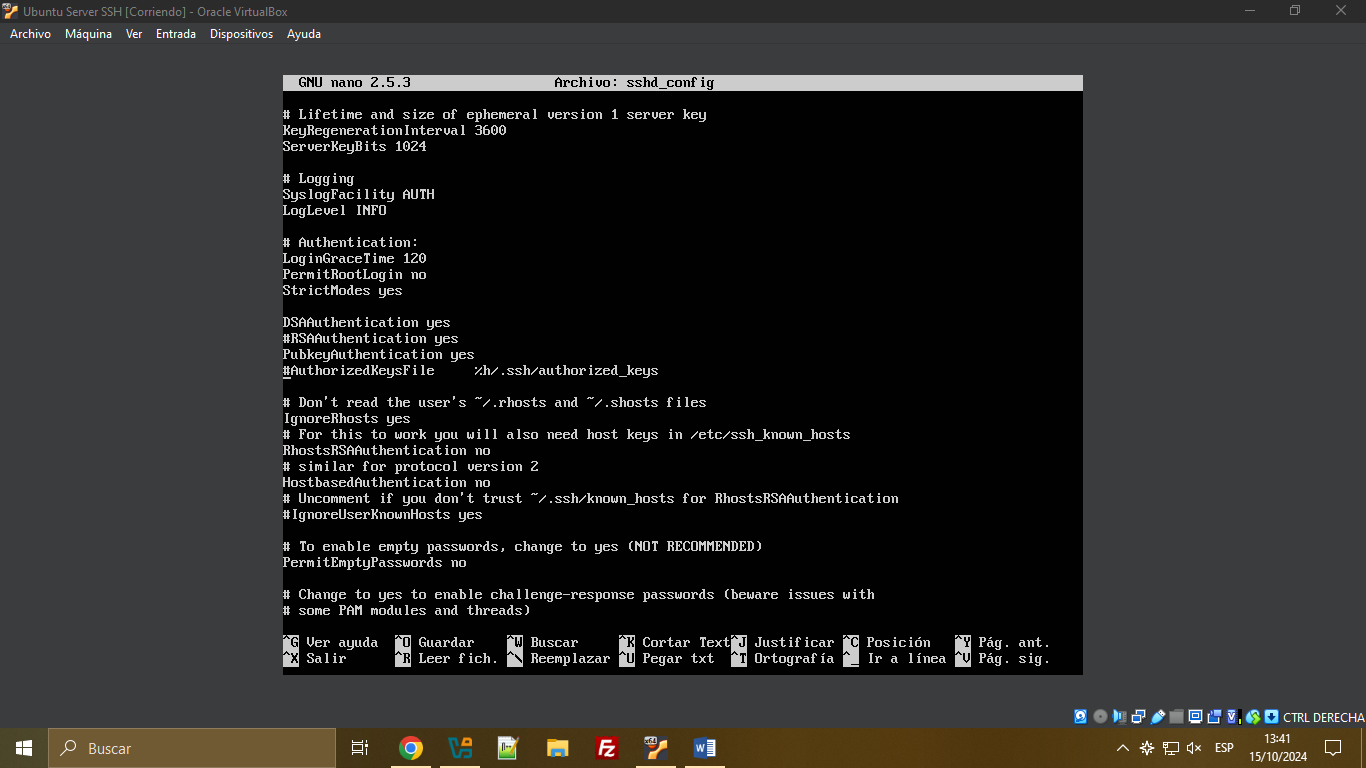
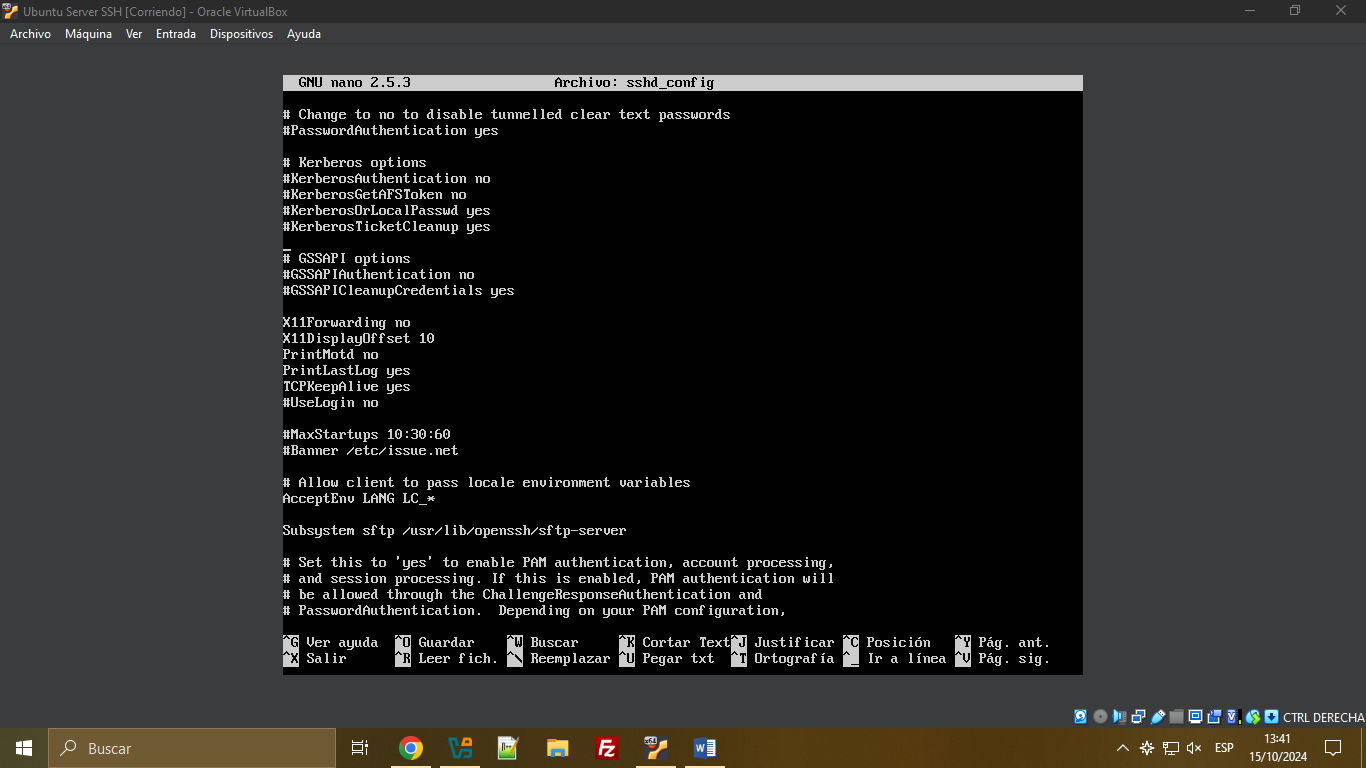
Discord está diseñado para la comunicación social, no para la administración de sistemas como Telnet o SSH.

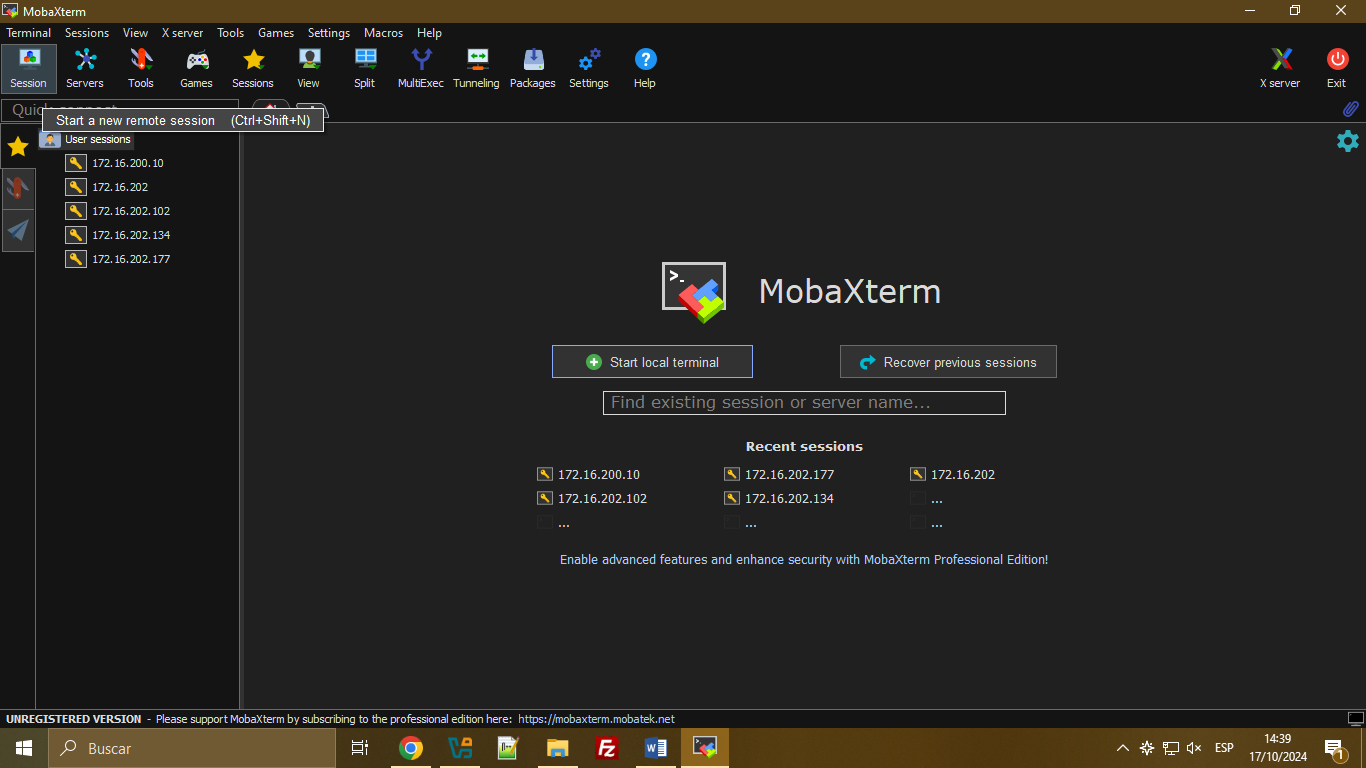
1. OpenSSH 6) Ssh 127.0.0.1
2. A. Ssh [estudiante@127.0.0.1](mailto:estudiante@127.0.0.1)

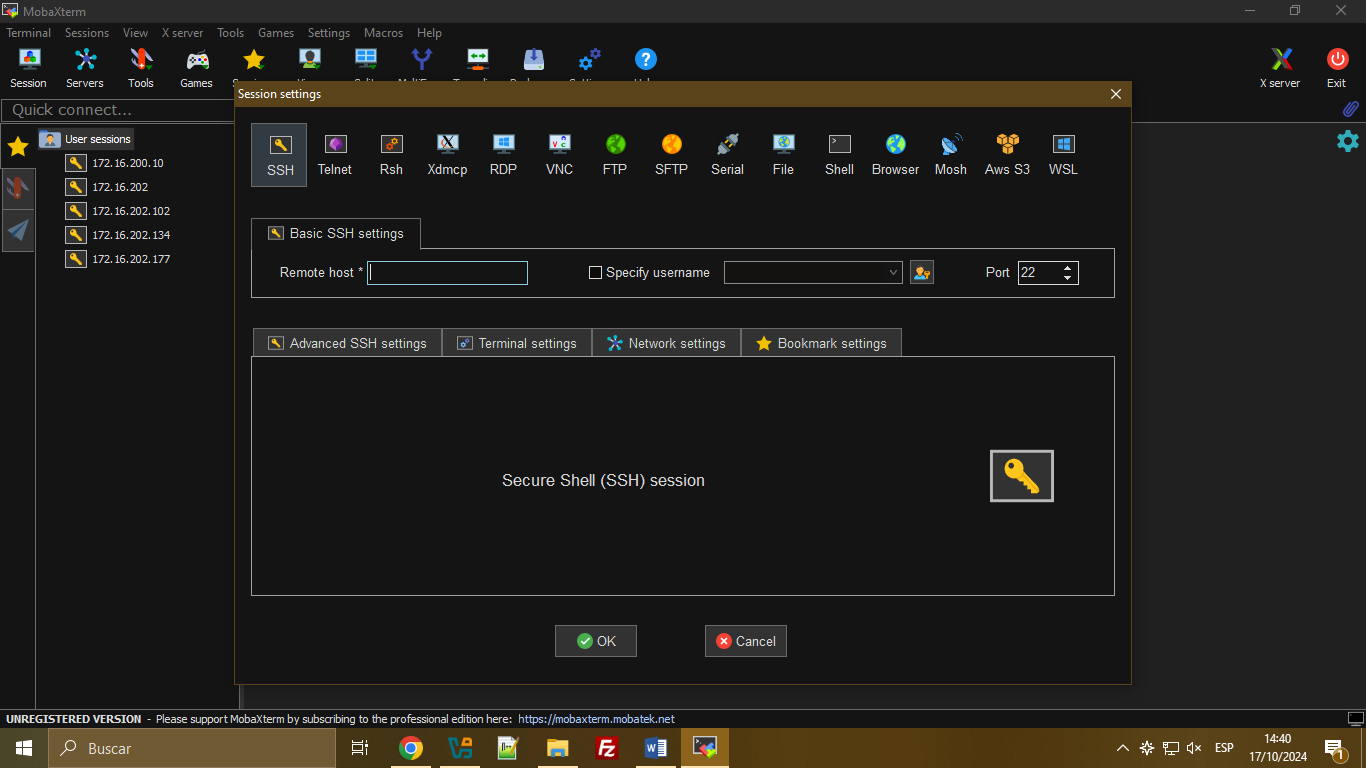
B. Ssh –p 10022 [estudiante@127.0.0.1](mailto:estudiante@127.0.0.1)

1. SSh a. 

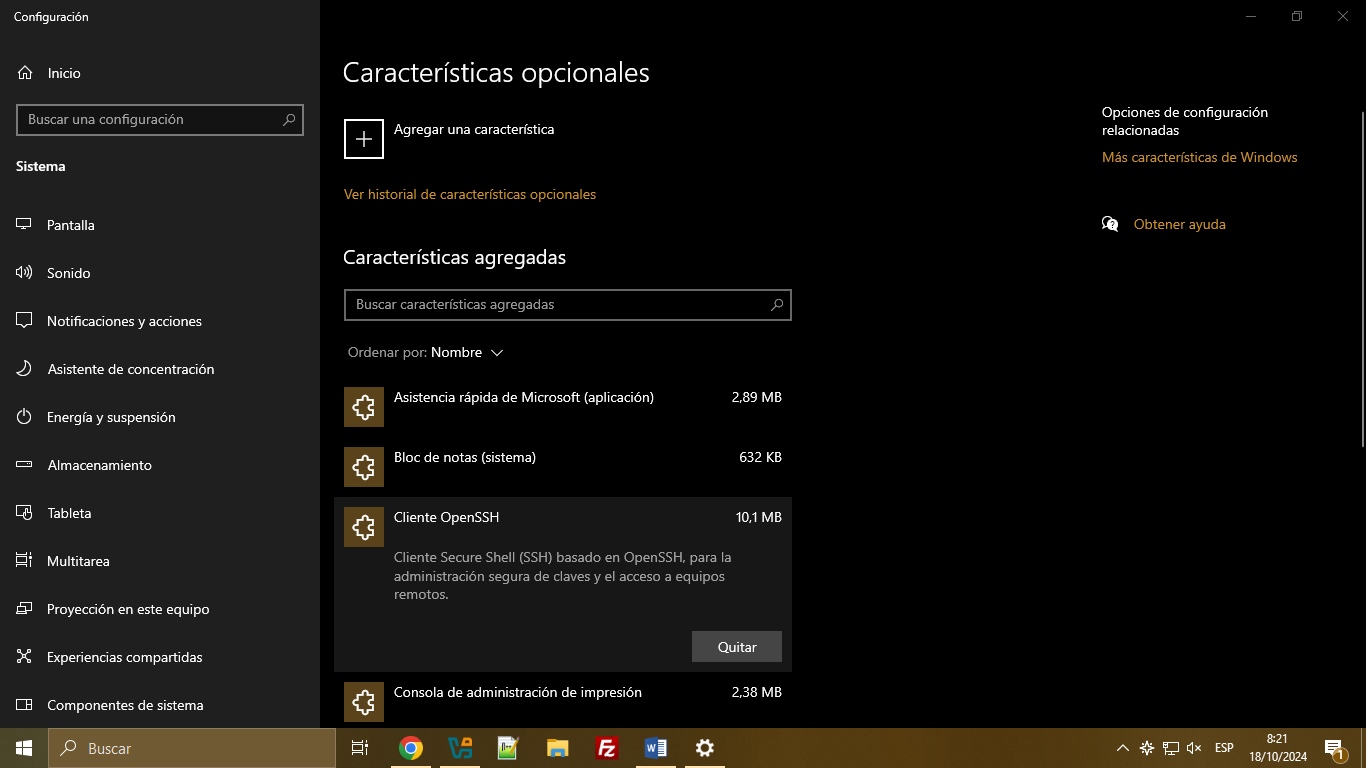
b.  

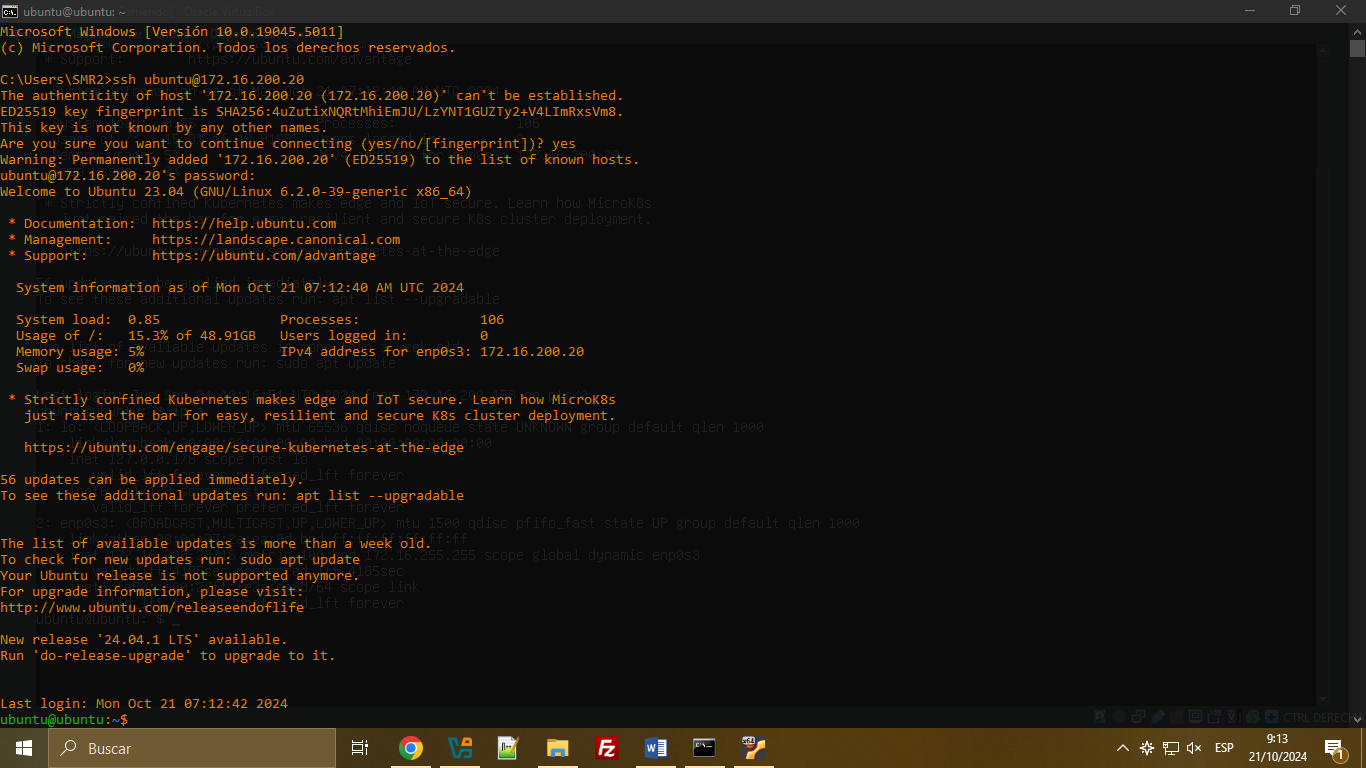
c.  

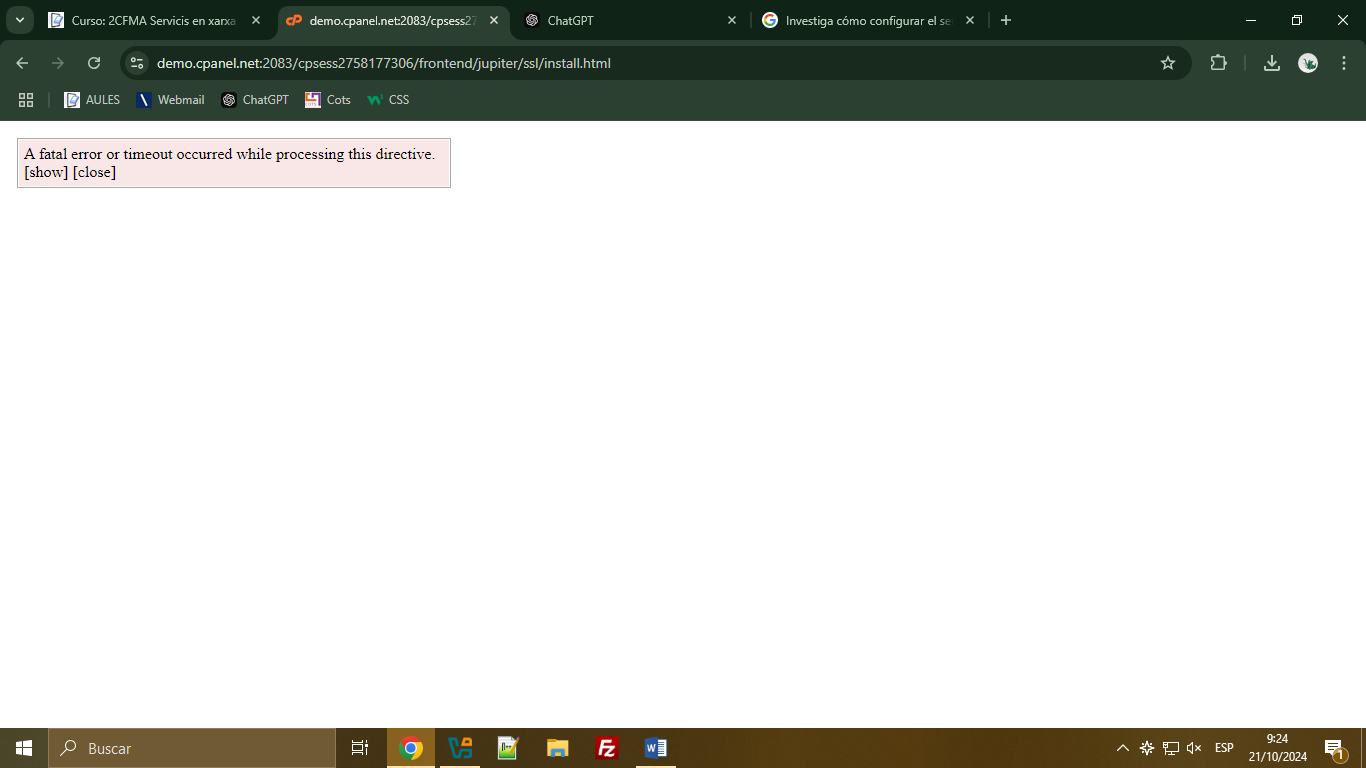
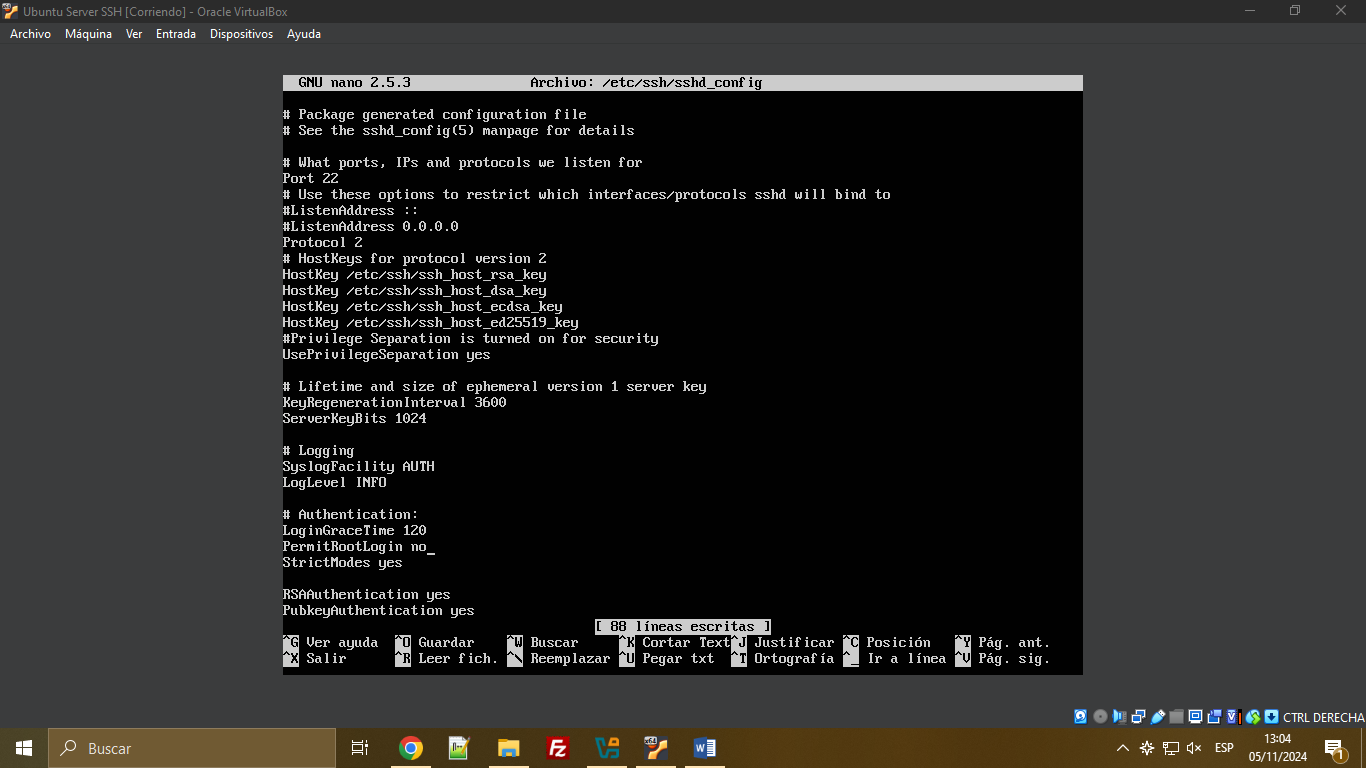
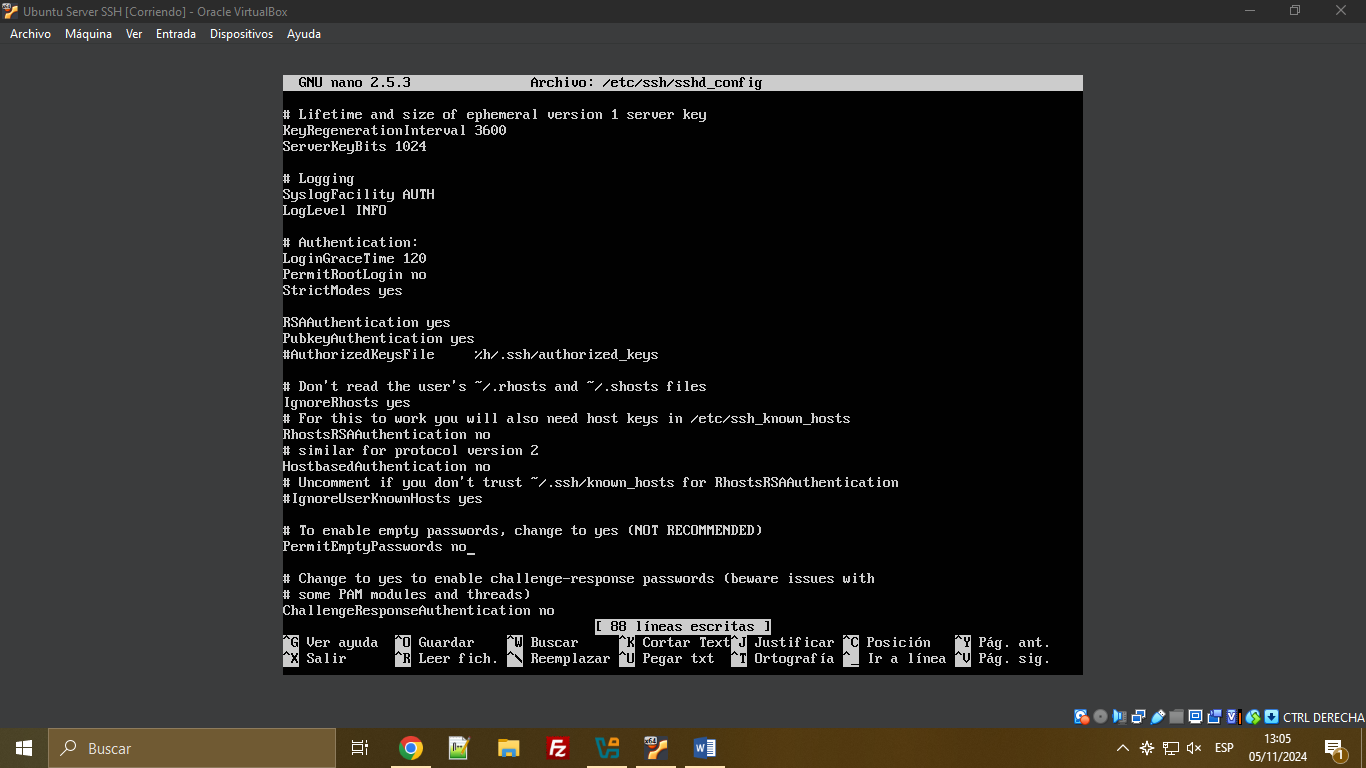
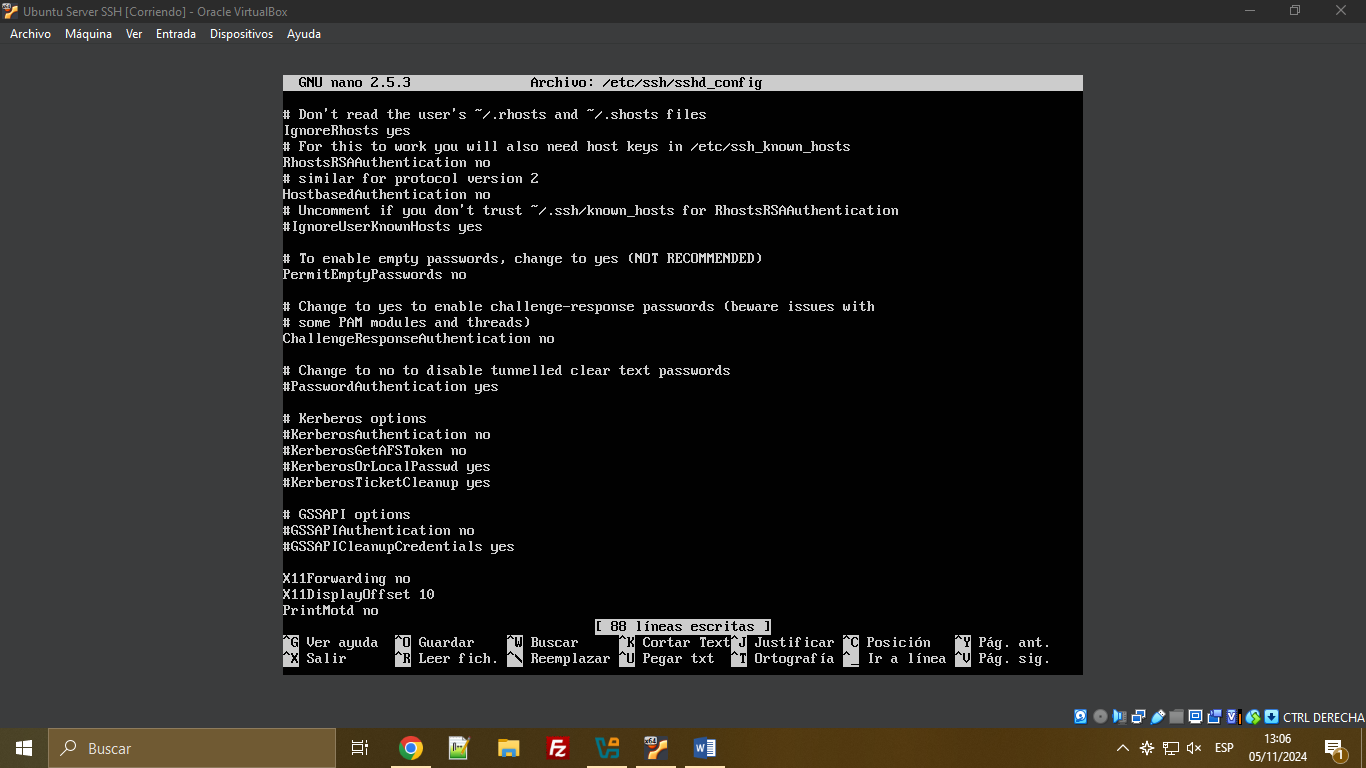
1. MobaXterm
   1. Para ver mi IP publica hay que ir a la pagina <https://www.profesionalhosting.com/ip/> y para configurar pulsamos en Session

Seleccionamos SSH y donde pone Remote host introducimos IP publica de servidor

* 1. En los cortafuegos de Windows no tienes que configurar, pero tienes que abrir puertos en el router
  2. Para configuración del servidor en red privada uso la ip privada, y para conectarse desde fuera uso la IP publica
  3. Los parámetros mínimos son IP del servidor, usuario y el puerto
  4. Parámetros:
     1. Host – el host que va a esperar por defecto escuchamos a todos
     2. Port – es siempre 22
     3. X11 con SSH – si permitimos terminal y grafico ponemos “yes” y si no pues “no”
     4. X11 MIT, XDM u otro – si permitimos conexión en grafico o no
  5. Comando para crear túnel ssh -L 10443:smtp.gmail.com:443 ubuntu@172.16.200.20

1. Abre Configuración (Win + I), Ve a Aplicaciones > Características opcionales, Haz clic en Agregar una característica, Busca "OpenSSH Server" en la lista y haz clic en Instalar

Y el comando es “ssh [ubuntu@172.16.200.20](mailto:ubuntu@172.16.200.20)”, yes, y la contraseña de usuario

1. No se puede con la versión gratuita
2. Un servidor NX permite a los usuarios acceder a escritorios remotos
3. Ejercicio 3,4
4. ---
5. En archivo SSH 14.docx
6. Configurar un servidor SSH   
   1. Encierre a los usuarios locales. – es de ftp
   2. Ponga los permisos de lectura para tí; de ejecución a tí y tu grupo y nada a los otros. – es de ftp