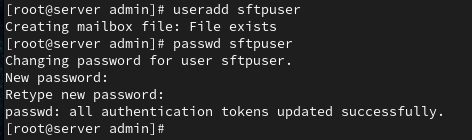
SFTP : is a secure file transfer protocol that we use for uploading, exporting, and managing files over a network, and it works based or on top of ssh protocol, for security purposes:

* Authentication, that can be done as how it is in ssh, password, or keys
* Data encryption
* Secure session, with frecommands that only lets you run sftp commands

TOPOLOGY:

* centos0 with 20.20.20.1 internal interface, and a bridge interface
* kali with 20.20.20.2 internal interface
* config the SFTP in centos

1. download openssh, already exist in centos by default for me
2. start and enable the sshd, with systemctl start sshd; systemct enable sshd
3. add user with whatever name, I will call it sftpuser, set a password for him, this user will be the one we aim to enter to his directory for managing file through sftp



1. change the sftp application/subsystem that runs when we start an sftp session, use the “Internal-sftp” instead, so its more suitable for chroot jailing



1. enter /etc/sshd/sshd\_config, and add the end set these lines :

Match User sftpuser

ForceCommand internal-sftp

PasswordAuthentication yes

ChrootDirectory /home/sftpuser

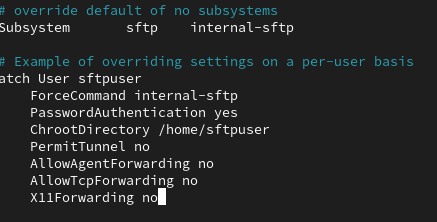
PermitTunnel no

AllowAgentForwarding no

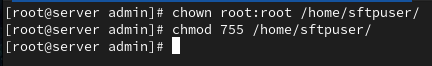
AllowTcpForwarding no

X11Forwarding no

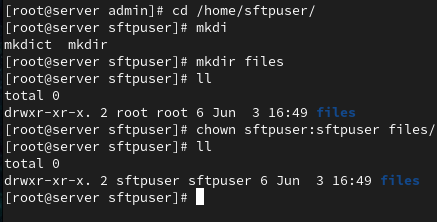
* **Match User sftpuser**  
  This line starts a configuration block that only applies to the user named **sftpuser**.
* **ForceCommand internal-sftp**  
  Forces the user to only use **SFTP** for file transfers. They will **not** get terminal (shell) access.
* **PasswordAuthentication yes**  
  Allows the user to log in using a **password** (instead of or in addition to SSH keys).
* **ChrootDirectory /home/sftpuser**  
  Locks the user inside the **/home/sftpuser** directory. They will not be able to see or access any files outside this folder. This improves security by isolating users. U can change the folder, just make sureu made it before
* **PermitTunnel no**  
  Disables SSH **tunneling** (no port forwarding or VPN-like behavior).
* **AllowAgentForwarding no**  
  Prevents **SSH agent forwarding**, which stops the server from accessing the user's private SSH key from their local machine.
* **AllowTcpForwarding no**  
  Blocks the user from using **TCP forwarding** (they can’t create custom network tunnels through SSH).
* **X11Forwarding no**  
  Disables **X11 forwarding**, so the user cannot run graphical Linux applications (like GUI apps) remotely through SSH.



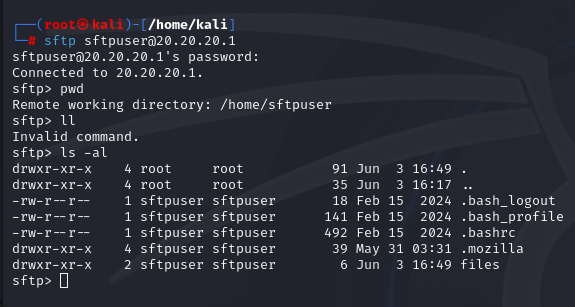
1. Change the owner of /home/sftpuser to the root, so sftpuser will kept confine inside the chroot jail and cant change anything about his /home/sftpuser “I mean his jail”, also change the permissions so the sftpuser wont delete, add, update that folder



1. Make a directory where the sftp users/clients, will manage files, and change the ownership to the sftpuser

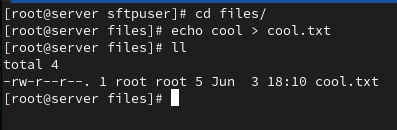


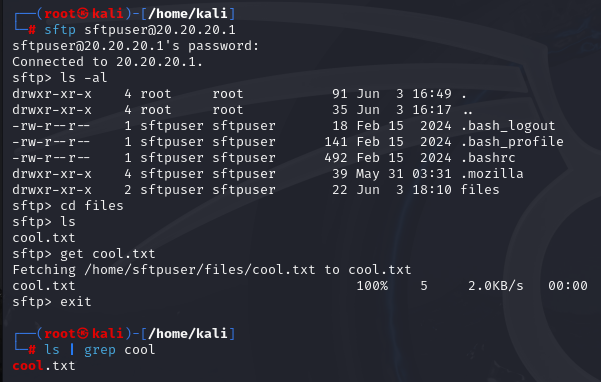
1. Now with the kali machine lets try to get into that sftp service



NOTE: we could get into it, but look how we cant exec eve ry command, only commands that the internal-sftps subsys lets us execute.

1. Lets create some files in the files directory and see if we can upload them from the kali.





1. TIPS: you can install filezilla client and use it for managing files instead of using the command line manner