GETTING STARTED

WITH

REMOTE ACCESS SERVICES

➢ REMOTE ACCESS SERVICES:

- Walk into the server room and install RedHat on your new server. After the installation is complete, you need to configure numerous packages and then begin setting up users.
- All these tasks take time, and if you work in data center, chances are office is located somewhere else. Remote access to any system that work on makes managing and troubleshooting much easier because you don't physically need to be in the same location as your systems.

❖ TELNET:

- Telnet is a network protocol which is used to connect remote computers over **TCP/IP** network.
- Telnet in **public network (WAN)** is very bad idea. It transmits login data in the **clear format.** Everything will be sent in **plain text.**
- It is a valuable tool for **troubleshooting and manipulation of various services.**
- The Telnet port number is **23**.

→ **Installing telnet-server package:**
#yum install telnet-server -y

→ **To start and enable telnet server:**
#systemctl start telnet.socket
#systemctl enable telnet.socket
#systemctl status telnet.socket

→ **To verify the port number:**
#netstat -pantl

CONNECTING REMOTE SERVER USING TELNET:

→ **Installing client package:**
#yum install telnet -y

SYNTAX: #telnet <server_ipaddress>

#telnet 192.168.10.254
Login:
Password:

❖ SSH (SECURE SHELL):

- SSH (Secure Shell) is a program for logging into a remote machine and executing commands on that machine.
- The SSH protocol provides secure encrypted communications between two untrusted hosts over an insecure network.
- OpenSSH is an implementation of the SSH protocol supported by Linux, UNIX, and similar operating systems.
- SSH running with port number 22.
- The OpenSSH suite consists of the following user-space tools:
  - **ssh** is a remote login program (SSH client).
  - **sshd** is an OpenSSH SSH daemon.
  - **scp** is a secure remote file copy program.
  - **sftp** is a secure file transfer program.
  - **ssh-agent** is an authentication agent for caching private keys.
  - **ssh-add** adds private key identities to ssh-agent.
  - **ssh-keygen** generates, manages, and converts authentication keys for ssh.
  - **ssh-copy-id** is a script that adds local public keys to the authorized_keys file on a remote SSH server.
  - **ssh-keyscan** gathers SSH public host keys.

INSTALLING & CONFIGURING OPENSSH SERVER:

→ **Installing openssh-server package:**
#yum install openssh-server -y

→ **To verify the installed package:**
#rpm -q openssh-server

→ **Start the sshd service at boot time:**
#systemctl start sshd
#systemctl enable sshd

→ **To verify the sshd service status:**
#systemctl status sshd

CONNECTING SSH-SERVER:

#ssh username@Ip-address / Hostname

#ssh root@192.168.10.254

SETTING AN OPENSSH SERVER CONFIGURATION:

→ **Open the /etc/ssh/sshd_config configuration in a text editor:**

→ **Changing port number, access root and password authentication.**

```
# vim /etc/ssh/sshd_config
    Port 22
    LoginGraceTime 2m
    PermitRootLogin yes
    MaxAuthTries 3
    PubkeyAuthentication yes
    PasswordAuthentication yes
```

→ **Reload the systemd configuration and restart sshd to apply the changes:**

```
#systemctl daemon-reload
#systemctl restart sshd
```

→ **Allow Users to connect ssh-server:**

```
    AllowUsers ram raju

#systemctl daemon-reload
#systemctl restart sshd
```

LOG FILE VERIFICATION:

- Login records are usually in **/var/log/secure**.

```
#tail -f /var/log/secure
      (or)
#journalctl -u sshd
```

GENERATING SSH KEY PAIRS:

- This is also be called as password less login.
- You can log in to an OpenSSH server without providing a password by generating an SSH key pair on a local system and copying the generated public key to the OpenSSH server.
- In this mechanism to setup an SSH key-based authentication and connect to your Linux servers without entering a password.

→ **Generate keys of the SSH protocol:**
#ssh-keygen

NOTE: You can also generate an RSA key pair by using the -t rsa option with the ssh-keygen command

→ **By default, keys are located in the user home directory .ssh location:**
#cd ~/.ssh
#ls

→ **Copy the public key to a remote machine:**
#ssh-copy-id <username>@<ip-address>
#ssh-copy-id root@192.168.10.100

→ **Log in to the OpenSSH server without providing any password:**
#ssh <username>@<ip-address>
#ssh root@192.168.10.100

SECURE COPY (SCP):

- Moving files between systems is one of a Linux system administrator's regular activity.
- On Red Hat Enterprise Linux, SFTP (Secure File Transfer Protocol) and SCP (secure copy) are handy commands to move files between systems securely.
- As part of the OpenSSH suite, these tools rely on Secure Shell (SSH) to transfer the files.
- The scp command line tool uses the SFTP protocol for file transfers by default.
- To transfer files with SCP, specify the remote server's IP address or hostname and the destination path where you want it to copy the file or directory.

→ **General syntax to transfer a local file to a remote system is as follows:**

#scp <options> <local_file> <username@tohostname:<remotefile>

#scp -r script.sh root@192.168.10.254:/opt

NOTE: Go to remote server and verify the script.sh file in /opt location

→ **syntax to transfer a remote file to the local system is as follows:**

#scp <options> <username@tohostname:remotefile> <newlocal_file>

#scp -r root@192.168.10.254:/opt/script.sh /root

#ls /root

Copy file with non-standard port number:

- It is used to Securely Copy File to a Remote Machine on a Non-Standard SSH Port and specify the port to connect on the remote host.
- It is useful when our SSH server is listening on a non-standard port.

SYNTAX: #scp -P port source_file user@hostname:destination_file
#scp -P 2024 devops root@192.168.10.254:/opt

WINSCP:

- It is an open source free SFTP client, FTP client, WebDAV client, S3 client & SCP client for windows.
- Its main function is **file transfer** between a local and a remote computer.
- It is a Graphical user interface.

PUTTY:

- Putty is a free and open-source terminal emulator, network file transfer application, and serial console for Windows platforms.
- It allows you to connect to remote computers or devices using various protocols such as secure socket shell (SSH), Telnet, login, and more.
- Putty is widely used by system administrators, network engineers, and developers for managing and controlling remote systems.