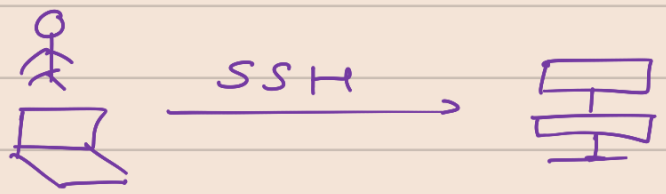
SSH - Secure - Shell (lesson 20)

Some use-cases:

- copy file to remote server
- Install software on new server

- SSH is a network protocal that gives users a secure way to access a computer over the internet.
- SSH refers also to the suite of utilities that implement that protocal



✓ encrypted data communication

2 ways to authenticate.

1.) username and passwd

→ admin creates user on remote server.

→ user can then connect with username and password

2.) SSH key pair.

a.) Client creates an SSH key pair.

Key pair = private key + public key

Private key = Secret key . is stored securely on the client machine

Public key : Public . can be shared , eg. with the remote server.

→ Client machine for that Public key can safely connect.

- Client can unlock the public key

with his private key.

If a public key of a person is not registered on the remote server, he/she cannot connect to it

* SSH for services:

Services, like Jenkins often need to connect to another server via ssh

→ Create jenkins user on app server

→ create ssh key pair on jenkins servers.

→ add public SSH key to authorized_keys on application server.

* firewall and port 22.

by default, SSH server listens on port 22

SSH is powerful and needs to be restricted to specific ip address.

* SSH in action.

1.) Create a remote server on cloud platform

2.) Generate SSH key pair on our laptop

3.) copy Bash script file to the remote server.

4.) execute script file on Remote server.

Create a virtual server on cloud platform (Digital ocean)

Create Droplet → Distribution (Ubuntu) → plan (Basic) cpu options (Regular Intel with SSD) → $5/m → Datacenter (Bengaluru)

→ Authentication (Password) root password : _____ → Create Droplet.

This will give us server on Digital ocean.

change server name → (remote-server)

ipv4: _____ (public) private ip: _____

* connect via SSH (password authentication)

open linux terminal

ssh root@159.69.73.21

↓ ip address of remote server.

'yes'

password: _____

'connection established.'

* generate SSH key pair.

open new terminal (yash@yash :~$)

ls .ssh/

→ known_hosts

ssh-keygen -t rsa

→ cryptographic algo used to encrypt keys.

enter, enter, enter.

Key pair created

ls .ssh/

→ id_rsa id_rsa.pub known_hosts

↓ private key ↓ public key

* add public keys to authorized_keys

(on root remote server terminal)

ls .ssh

→ authorized_keys → this is the file where we add the public keys.

cat .ssh/authorized_Keys

→ empty

vim .ssh/authorized_keys

(go to local terminal)
yash@yash

cat .ssh/id_rsa.pub

copy that whole text

(go to the root remote server terminal)

vim .ssh/authorized_keys

paste that text.

:wq

exit

(back to local machine)

ssh root@159.69.83.54

(connection done)

exit

(multiple id_rsa

ssh -i .ssh/id_rsa2 root@159.69.84.32

* copy Bash script and execute

(local computer) Terminal.

vim test.sh

#!/bin/bash

echo "I am executed on the remote server"

:wq

scp test.sh root@169.59.49.32:/root

scp → secure copy; test.sh → file; root@169.59.49.32 → server; /root → root directory.

ssh root@169.59.49.32

ls

ls -l

chmod u+x test.sh

./test.sh

"I am executed on remote server".

* Wrap up.

We will use lot of ssh.