

Lab 14 – How to SSH into a server from a Linux machine

Task 1: Connect to a Remote Server via SSH

Logic / Purpose:

This task introduces you to the basic usage of SSH (Secure Shell), which allows you to remotely access and manage a device over a secure network connection. SSH ensures encrypted communication between your local machine and the remote server.

Detailed Steps:

1. **Open the terminal** on your Linux system.

Use the SSH command to initiate a connection to the remote server:

```
ssh osboxes@192.168.1.221
```

- 2.

- **ssh**: The command to start the Secure Shell connection.
- **osboxes**: The username you want to log in as on the remote server.
- **192.168.1.221**: The IP address of the remote server.

Accept the server's fingerprint when prompted with:

The authenticity of host '192.168.1.221 (192.168.1.221)' can't be established.
Are you sure you want to continue connecting (yes/no)?

- 3.

- Type **yes** and press Enter. This adds the server's key to your list of known hosts.

Enter the password when prompted. Use:

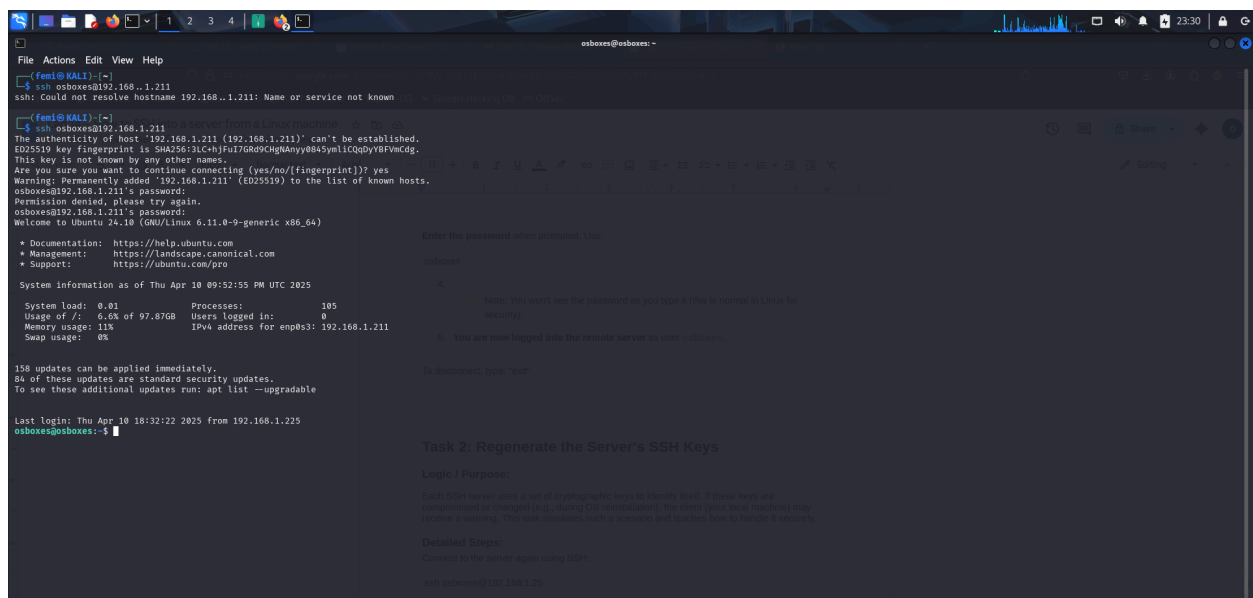
osboxes

4.

Note: You won't see the password as you type it (this is normal in Linux for security).

5. You are now logged into the remote server as user **osboxes**.

To disconnect, type: "exit"



```
(femi@KALI) ~$ ssh osboxes@192.168.1.221
ssh: Could not resolve hostname 192.168.1.221: Name or service not known

(femi@KALI) ~$ ssh osboxes@192.168.1.221
The authenticity of host '192.168.1.221 (192.168.1.221)' can't be established.
ED25519 key fingerprint is SHA256:3lChjFul7GR9CHgNany0845ym1CQqdyYBFVmcDg.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.221' (ED25519) to the list of known hosts.
osboxes@192.168.1.221's password:
osboxes@192.168.1.221 ~$
Welcome to Ubuntu 24.10 (GNU/Linux 6.11.0-9-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Thu Apr 10 09:52:55 PM UTC 2025

System load: 0.02          Processes:              185
Usage of /:  6.6% of 97.87GB Users logged in: 0
Memory usage: 11%          IPv4 address for enp8s3: 192.168.1.221
Swap usage:  0%

158 updates can be applied immediately.
84 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Last login: Thu Apr 10 18:32:22 2025 from 192.168.1.225
osboxes@osboxes:~$
```

Task 2: Regenerate the Server's SSH Keys

Logic / Purpose:

Each SSH server uses a set of cryptographic keys to identify itself. If these keys are compromised or changed (e.g., during OS reinstallation), the client (your local machine) may receive a warning. This task simulates such a scenario and teaches how to handle it securely.

Detailed Steps:

Connect to the server again using SSH:

```
ssh osboxes@192.168.1.221
```

Gain root privileges:

```
sudo su -
```

2.

🌐 Enter the user password if prompted.

Delete existing SSH keys:

```
rm -v /etc/ssh/ssh_host_*
```

3.

- 🌕 **-v**: Makes **rm** verbose, showing which files are deleted.

```
root@osboxes:~# sudo su-
osboxes@osboxes:~$ sudo su-
[sudo] password for osboxes:
osboxes@osboxes:~$ sudo su-
sudo: su- command not found
osboxes@osboxes:~$ sudo su -
osboxes@osboxes:~$ ls
total 0
root@osboxes:~# ls -la
total 20
drwxr-xr-x  3 root root 4096 Oct  7 2024 .
drwxr-xr-x  1 root root 3136 Oct  7 2024 ..
-rw-r--r--  1 root root 161 Oct  7 2024 .profile
drwxr-xr-x  2 root root 4096 Oct 27 03:57 .ssh
root@osboxes:~# vi femi
root@osboxes:~# cd .ssh
root@osboxes:~/.ssh# ls
authorized_keys
root@osboxes:~/.ssh# cat authorized_keys
root@osboxes:~/.ssh# vi authorized_keys
root@osboxes:~/.ssh# cd .profile
-bash: cd: .profile: No such file or directory
root@osboxes:~/.ssh# cd ..
root@osboxes:~# cd .profile
-bash: cd: .profile: Not a directory
root@osboxes:~# vi .profile
root@osboxes:~# vi .bashrc
root@osboxes:~# ls
femi
root@osboxes:~# cat femi
viNCrypt@031e.cchsec[?***e]@ecucw@jB...
root@osboxes:~# vi femi
root@osboxes:~# pwd
/root
root@osboxes:~# rm -v /etc/ssh/ssh_host_*
removed '/etc/ssh/ssh_host_ecdsa_key'
removed '/etc/ssh/ssh_host_ecdsa_key.pub'
removed '/etc/ssh/ssh_host_ed25519.key'
removed '/etc/ssh/ssh_host_ed25519.key.pub'
removed '/etc/ssh/ssh_host_rsa_key'
removed '/etc/ssh/ssh_host_rsa_key.pub'
root@osboxes:~#
```

Regenerate new keys for the server:

`dpkg-reconfigure openssh-server`

4.

🟡 This command generates a fresh set of SSH host keys.

```
root@osboxes:~# dpkg-reconfigure openssh-server
Creating SSH2 RSA key; this may take some time ...
1072 SHA256:2015aW63uy32/syaxxz7+0G9y2f7j130WkK/njc root@osboxes (RSA)
Creating SSH2 ECDSA key; this may take some time ...
256 SHA256:zxi2pJ26YexC6U4VCH6cJAARt83fMUSM80xq7IA root@osboxes (ECDSA)
Creating SSH2 ED25519 key; this may take some time ...
256 SHA256:a0uj+8mzLymL1e6g5745e1AZ/2N3lWn1oQid2H+gAQ root@osboxes (ED25519)
root@osboxes:~#
```

Regenerate new keys for the server.

`dpkg-reconfigure openssh-server`

4.

This command generates a fresh set of SSH host keys.

Restart the SSH service to apply the changes.

Restart the SSH service to apply the changes:

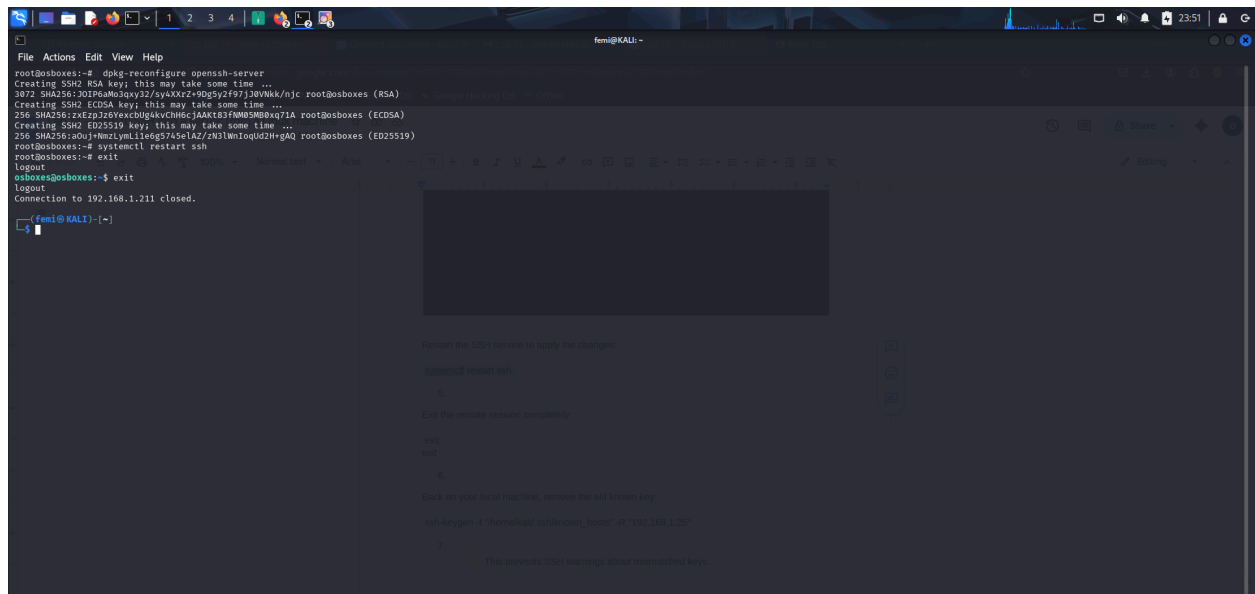
```
systemctl restart ssh
```

5.

Exit the remote session completely:

```
"exit"
```

```
"exit"
```



```
File Actions Edit View Help
root@osboxes:~# dpkg-reconfigure openssh-server
Creating SSH2 RSA key; this may take some time ...
3072 SHA256:30IP6aM03qy32/sy4XXrz+90G5y2F97j30VnkK/njc root@osboxes (RSA)
Creating SSH2 ECDSA key; this may take some time ...
256 SHA256:zxEZpJ2GyexcB0g4kvChH6cJAAt83fM05MB0xq7IA root@osboxes (ECDSA)
Creating SSH2 ED25519 key; this may take some time ...
256 SHA256:z0nJ+Mwzsyml1e6g57A5clA2/z81Wn1oq02hngQ root@osboxes (ED25519)
root@osboxes:~# systemctl restart ssh
root@osboxes:~# exit
logout
osboxes@osboxes:~$ exit
logout
Connection to 192.168.1.211 closed.

(femi@kali) ~$
```

6.

🟡 This prevents SSH warnings about mismatched keys.

Reconnect using SSH and accept the new key:

```
ssh osboxes@192.168.1.221
```

It would throw a security warning: “WARNING : REMOTE IDENTIFICATION HAS CHANGED”

```
(femi@KALI)-[~]
$ ssh osboxes@192.168.1.221
Warning: REMOTE HOST IDENTIFICATION HAS CHANGED!
It is possible that a host key has just been changed.
The fingerprint for the ED25519 key sent by the remote host is
SHA256:a0Uj+MazLymL1e6g5745e1AZ/zN3lwnIoQd2H+gAQ.
Please contact your system administrator.
Add correct host key in /home/femi/.ssh/known_hosts to get rid of this message.
Offending ECDSA key in /home/femi/.ssh/known_hosts:3
remove with:
ssh-keygen -f /home/femi/.ssh/known_hosts -R '192.168.1.221'
Host key for 192.168.1.221 has changed and you have requested strict checking.
Host key verification failed.

Task 3: Passwordless SSH Login Using Key-Based Authentication

Logic / Purpose:
Passwordless login via SSH keys enhances security and convenience. Your system uses cryptographic keys instead of relying on typed passwords.

Detailed Steps:
Navigate to your .ssh directory.
```

To fix this;

Back on your local machine, remove the old known key:

```
ssh-keygen -f "/home/kali/.ssh/known_hosts" -R "192.168.1.221"
```

```
osboxes@osboxes: ~
File Actions Edit View Help

(femi@KALI)-[~]
$ ssh-keygen -f /home/kali/.ssh/known_hosts -R 192.168.1.221
Cannot stat /home/kali/.ssh/known_hosts: No such file or directory

(femi@KALI)-[~]
$ ssh-keygen -f /home/femi/.ssh/known_hosts -R 192.168.1.221
# Host 192.168.1.221 found: line 1
# Host 192.168.1.221 found: line 2
# Host 192.168.1.221 found: line 3
/home/femi/.ssh/known_hosts updated.
Original contents retained as /home/femi/.ssh/known_hosts.old

(femi@KALI)-[~]
$ ssh osboxes@192.168.1.221
The authenticity of host '192.168.1.221 (192.168.1.221)' can't be established.
ED25519 key fingerprint is SHA256:a0Uj+MazLymL1e6g5745e1AZ/zN3lwnIoQd2H+gAQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.221' (ED25519) to the list of known hosts.
Connection closed by 192.168.1.221 port 22

(femi@KALI)-[~]
$ ssh osboxes@192.168.1.221
osboxes@192.168.1.221's password:
Welcome to Ubuntu 24.10 (GNU/Linux 6.11.0-9-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Thu Apr 10 11:18:22 PM UTC 2025

System load: 0.0          Processes: 103
Usage of /: 6.6% of 97.87GB Users logged in: 0
Memory usage: 12%        IPv4 address for enp0s3: 192.168.1.221
Swap usage: 0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.
   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

158 updates can be applied immediately.
6% of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Last login: Thu Apr 10 21:52:56 2025 from 192.168.1.76
osboxes@osboxes:~$
```

Try SSH again:

```
ssh osboxes@192.168.1.221
```

You'll be asked to trust the new key — type **yes**, and then log in as before.

Task 3: Passwordless SSH Login Using Key-Based Authentication

Logic / Purpose:

Passwordless login via SSH keys enhances security and convenience. Your system uses cryptographic keys instead of relying on typed passwords.

Detailed Steps:

Navigate to your `.ssh` directory:

```
cd ~/.ssh/
```

1.

Generate an SSH key pair:

```
ssh-keygen
```

2.

- Press Enter to accept default filename.
- Press Enter again to leave the passphrase blank (or enter a passphrase for extra security).

3. Upload your public key to the remote server by Manually copyng your public key to the remote machine:

```
ssh-copy-id -i ~/.ssh/femi.pub osboxes@192.168.1.211
```

Now, the remote machine knows to **trust your private key** when you log in.

4. Login using your private key

You tried to log in using:

```
ssh -i /home/femi/.ssh/femi osboxes@192.168.1.21
```

```
osboxes@osboxes:~$ ssh -i ~/.ssh/femi osboxes@192.168.1.211
osboxes@192.168.1.211:~$ ls
femi  femi.pub  known_hosts  known_hosts.old  known_hosts.backup

osboxes@192.168.1.211:~$ ssh-copy-id -i ~/.ssh/femi.pub osboxes@192.168.1.211
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/femi/.ssh/femi.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
Enter passphrase for key "/home/femi/.ssh/femi.pub":
/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist on the remote system.
(if you think this is a mistake, you may want to use -f option)

osboxes@192.168.1.211:~$ ssh -i ~/.ssh/femi osboxes@192.168.1.211
Enter passphrase for key "/home/femi/.ssh/femi.pub":
Welcome to Ubuntu 24.10 (GNU/Linux 6.11.0-9-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Fri Apr 11 10:11:45 PM UTC 2025

System load: 0.0          Processes: 102
Usage of /:  7.0% of 97.87GB   Users logged in: 0
Memory usage: 17%          IPv4 address for enp0s3: 192.168.1.211
Swap usage:  0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
just raised the bar for easy, resilient and secure K8s cluster deployment.
https://ubuntu.com/engage/secure-kubernetes-at-the-edge

74 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

** System restart required **
Last login: Fri Apr 11 22:04:04 2025 from 192.168.1.76
osboxes@osboxes:~$
```

(Optional) Disable password authentication for added security:

Disable SSH Password Login for Better Security

Purpose:

To enhance the security of your server by disabling password logins and requiring SSH key-based authentication only. This prevents brute-force password attacks and ensures only devices with your private key can log in.

Step 1: SSH into your server using your SSH key

bash

CopyEdit

```
ssh -i ~/.ssh/femi osboxes@192.168.1.211
```


Why?

You need to log into the server before you can change its configuration. This command logs you in using your SSH private key (**femi**), not a password.

Step 2: Open the SSH server configuration file

```
sudo nano /etc/ssh/sshd_config
```

Why?

This file controls how the SSH server behaves — what it allows and disallows. We'll change it so the server no longer accepts passwords for login.

sudo gives you administrator rights, and **nano** is the text editor we're using.

Step 3: Find the # **Authentication:** section

Scroll until you find this (or something similar):

```
# Authentication:
```

```
#LoginGraceTime 2m
```

```
#PermitRootLogin prohibit-password
```

```
#StrictModes yes
```

```
#MaxAuthTries 6
```

```
#MaxSessions 10
```

Right below this section, add the following line (or find and modify it if it already exists):

`PasswordAuthentication no`

Why?

This tells the SSH server:

“Do not accept password logins anymore. Only allow people with a valid SSH key.”

Make sure this line is not commented out (no `#` at the beginning).

Step 4: Save and exit the file

In `nano`, do this:

- Press `Ctrl + O` → saves the file
- Press `Enter` → confirms the filename
- Press `Ctrl + X` → exits the editor

Why?

You’ve made a change to the SSH configuration — now you need to save and close it so you can apply it.

Step 5: Restart the SSH service

`bash`

`sudo systemctl restart ssh`

Why?

This restarts the SSH server so that your new settings (disabling password login) take effect immediately.

✓ Step 6: Test the new login behavior from your Kali machine

✓ This should still work (because it uses your private key):
bash

```
ssh -i ~/.ssh/femi osboxes@192.168.1.211
```

✗ This should now fail (because password login is disabled):
bash

```
ssh osboxes@192.168.1.211
```

Task 4: SSH Port Forwarding (Tunneling)

Logic / Purpose:

SSH tunneling allows you to securely access a network service (e.g., a web server) on another machine through the remote server.

Detailed Steps:

On your local machine, run this command:

```
ssh -L8080:192.168.1.1:80 osboxes@192.168.1.211
```

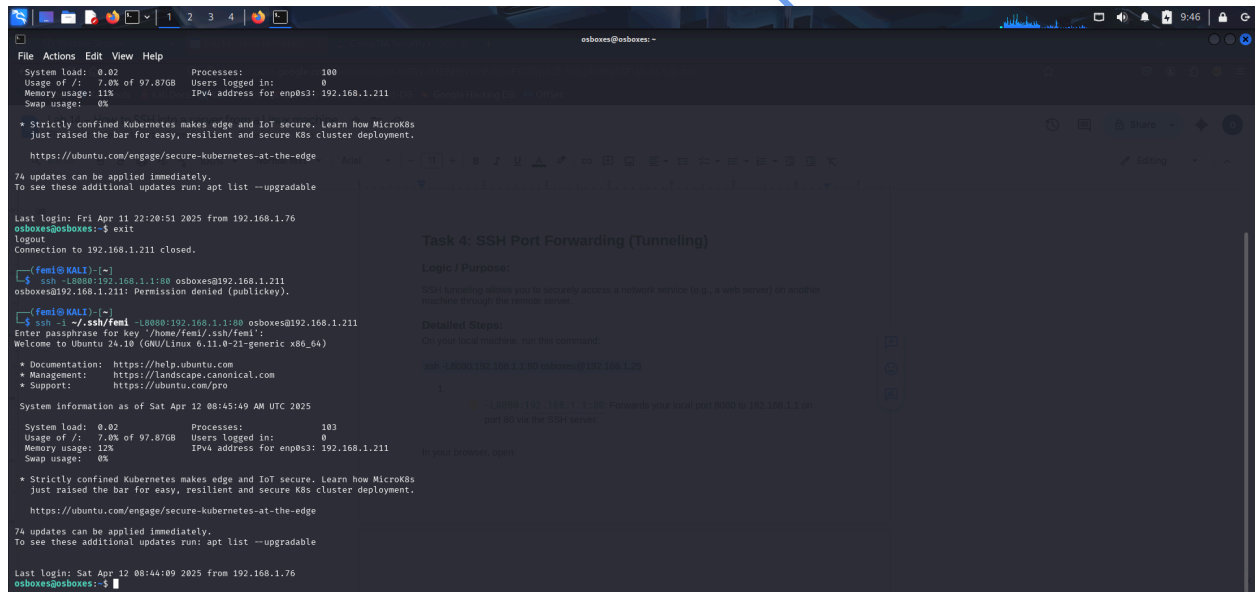
1.

🟡 **-L8080:192.168.1.1:80**: Forwards your local port 8080 to 192.168.1.1 on port 80 via the SSH server.

Note: In case you already disabled SSH password for login, the the above command won't work. So instead, use:

```
ssh -i ~/.ssh/femi -L8080:192.168.1.1:80 osboxes@192.168.1.211
```

N/A



```
osboxes@osboxes:~$ ssh -L8080:192.168.1.1:80 osboxes@192.168.1.211
osboxes@192.168.1.211: Permission denied (publickey).

(femi@KALI)~$ ssh -i ~/.ssh/femi -L8080:192.168.1.1:80 osboxes@192.168.1.211
Enter passphrase for key '/home/femi/.ssh/femi':
Welcome to Ubuntu 24.10 (GNU/Linux 6.11.0-21-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Sat Apr 12 08:45:49 AM UTC 2025

System load: 0.02          Processes: 103
Usage of /:  7.0% of 97.87GB Users logged in: 0
Memory usage: 12%         IPv4 address for enp0s3: 192.168.1.211
Swap usage:  0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

74 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

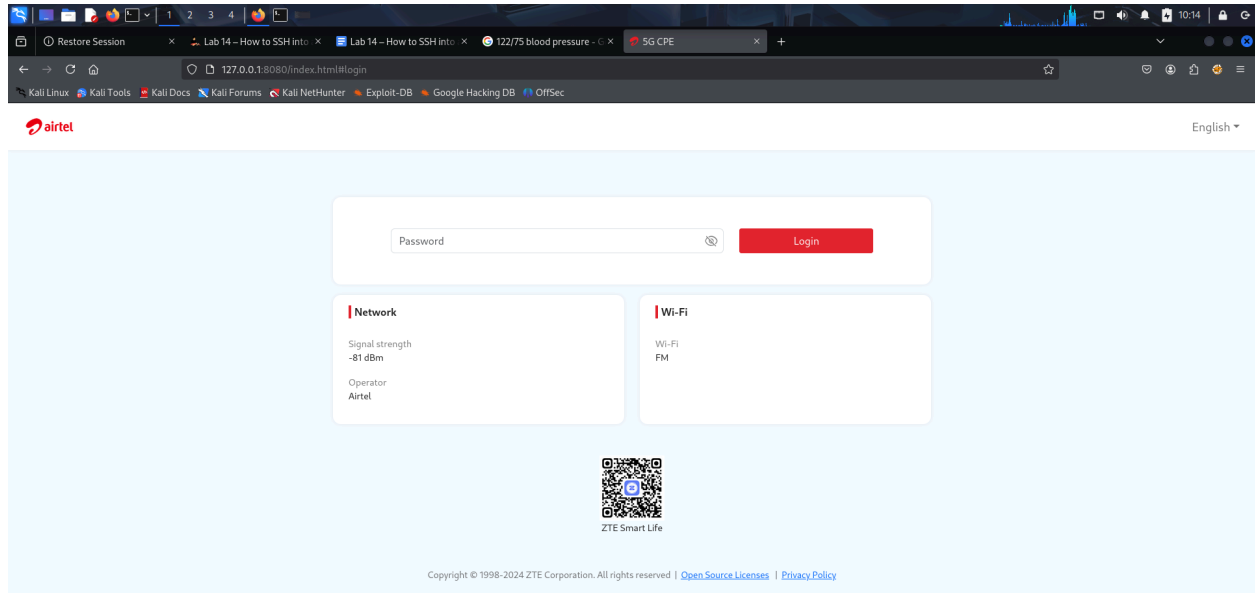
Last login: Sat Apr 12 08:44:09 2025 from 192.168.1.76
osboxes@osboxes:~$
```

In your browser, open:

`http://127.0.0.1:8080`

2.

🟡 This accesses the internal service via the SSH tunnel.



Task 5: Secure File Copy Using SCP

Logic / Purpose:

SCP (Secure Copy) allows you to copy files from your local system to a remote server (or vice versa) over an SSH connection.

Detailed Steps:

Copy a local file to the remote server:

```
scp /etc/hosts osboxes@192.168.1.25:/tmp/
```

1.

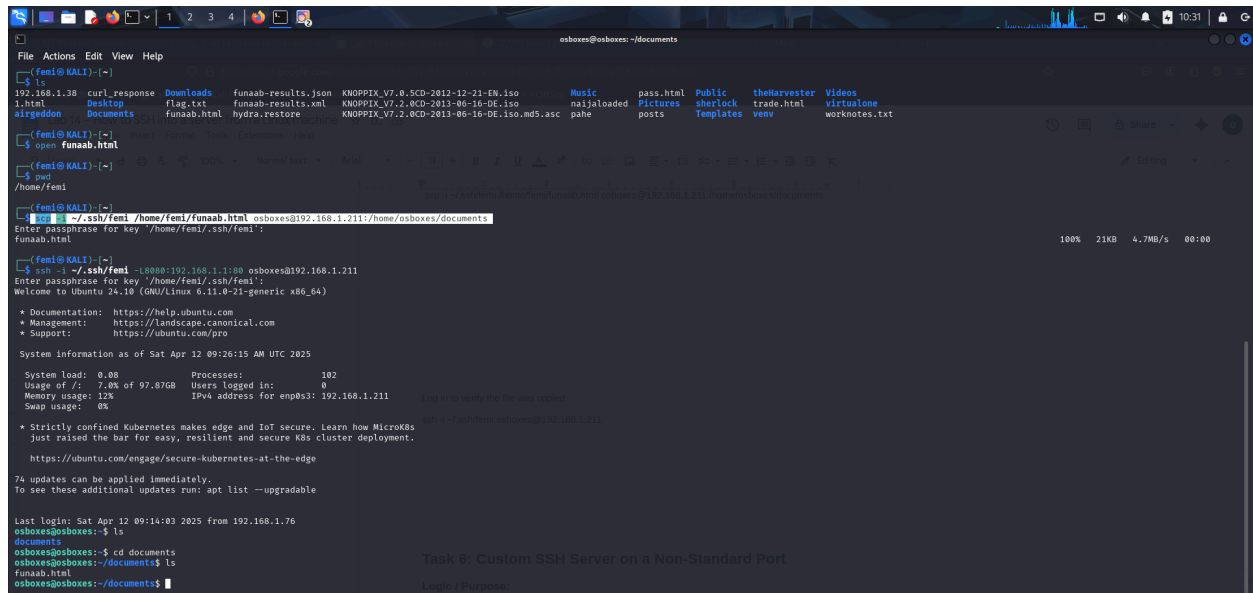
🟡 This command sends the `/etc/hosts` file to `/tmp/` on the remote server.

Note: In case you already disabled SSH password for login, the above command won't work. So instead, use:

```
scp -i ~/.ssh/femi /home/femi/funaab.html osboxes@192.168.1.211:/home/osboxes/documents
```

Log in to verify the file was copied:

```
ssh -i ~/.ssh/femi osboxes@192.168.1.211
```



```
(femi@KALI)~$ ls
192.168.1.38  curl_response  Downloads  funaab-results.json  KNOPPIX_V7.8.5CD-2012-12-21-EN.iso  Music  pass.html  Public  theHarvester  Videos
1.html      Desktop      flag.txt   funaab-results.xml  KNOPPIX_V7.2.0CD-2011-06-16-DE.iso  naja10loaded  Pictures  sherlock  trade.html  virtualone
airgeddon   Documents    funaab.html  hydra.restore       KNOPPIX_V7.2.0CD-2011-06-16-DE.iso.md5.asc  pahe         posts    Templates  venv        worknotes.txt

(femi@KALI)~$ open funaab.html
(femi@KALI)~$ pwd
/home/femi
(femi@KALI)~$ ssh -i ~/.ssh/femi /home/femi/funaab.html osboxes@192.168.1.211:/home/osboxes/documents
Enter passphrase for key '/home/femi/.ssh/femi':
funaab.html
(femi@KALI)~$
osboxes@osboxes: ~/documents
$ ssh -i ~/.ssh/femi -L8080:192.168.1.1:80 osboxes@192.168.1.211
Enter passphrase for key '/home/femi/.ssh/femi':
welcome to Ubuntu 24.10 (GNU/Linux 6.11.0-21-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Sat Apr 12 09:26:15 AM UTC 2025

System load: 0.08          Processes:              102
Usage of /:  7.0% of 97.87GB Users logged in:        0
Memory usage: 12%          IPv4 address for enp0s3: 192.168.1.211
Swap usage:  0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.
   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

74 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Sat Apr 12 09:14:03 2025 from 192.168.1.76
osboxes@osboxes:~$ ls
documents
osboxes@osboxes:~$ cd documents
osboxes@osboxes:~/documents$ ls
funaab.html
osboxes@osboxes:~/documents$
```

Task 6: Custom SSH Server on a Non-Standard Port

Logic / Purpose:

This demonstrates how to run a custom SSH server on a different port without needing root privileges—useful in restricted environments or for testing.

Detailed Steps:

Generate a new SSH key pair:

```
ssh-keygen -t rsa -N "" -f ~/.ssh/id_rsa <<< y
```

1.

 **-t rsa**: Specifies RSA key type.



```
femi@KALI: ~/ssh
File Actions Edit View Help

(femi@KALI)-[~]
$ ssh-keygen -t rsa -W "" -f ~/.ssh/id_rsa ooc y
Generating public/private rsa key pair.
Your identification has been saved in /home/femi/.ssh/id_rsa
Your public key has been saved in /home/femi/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:maahDZ7595SPeb75t/72mCdr+XqlQiv38Qz8WfajY femi@KALI
The key's randomart image is:
+-----+
| o. |
| ... |
| o. o |
| o... o . + |
| .o5. f * + |
| . . + o + +o |
| o . + o . + |
| . + o + + |
| oB=O = + |
+-----+
[SHA256]
(femi@KALI)-[~]
$ cd ~/.ssh/

(femi@KALI)-[~/ssh]
$ ls
femi  femi.pub  id_rsa  id_rsa.pub  known_hosts  known_hosts.old

(femi@KALI)-[~/ssh]
$
```

Create a custom SSH server configuration:

```
nano ~/.ssh/sshd_config
```

2.

Add the following lines:

Port 2222

UsePAM yes

HostKey ~/.ssh/id_rsa

Start the SSH server with this config:

```
/usr/sbin/sshd -f ~/.ssh/sshd_config
```

3.

Connect using the custom port:

ssh -p 2222 osboxes@192.168.1.211

```
osboxes@osboxes:~$ ssh -p 2222 osboxes@192.168.1.211
(osboxes@192.168.1.211) Password:
(osboxes@192.168.1.211) Password:
Welcome to Ubuntu 24.10 (GNU/Linux 6.11.0-21-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Sat Apr 12 10:34:26 AM UTC 2025

System load: 0.00      Processes:           180
Usage of /:   7.8% of 97.87GB   Users logged in:       0
Memory usage: 12%      IPv4 address for enp0s3: 192.168.1.211
Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.
   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

74 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Sat Apr 12 10:33:20 2025 from 192.168.1.76
osboxes@osboxes:~$ ls
documents
osboxes@osboxes:~$ ssh -p 2222 osboxes@192.168.1.211

End of Queue
```

End