SSH and Webserver Essentials

This handout covers essential concepts and practical steps for deploying a website using SSH, managing Linux services, and transferring files securely to your server.

---

1. Introduction

The main goal is to deploy a simple HTML website to a Linux web server. Key steps:

- Connect remotely using **SSH**
- Secure your server with **SSH keys**
- Manage your server remotely
- Transfer files to the server via **SCP/SFTP**
- Install & manage an Apache web server
- Publish a website online

---

2. Understanding SSH (Secure Shell)

SSH is a cryptographic protocol for secure remote login to Linux servers.

- **Server-side** (usually pre-installed): Accepts incoming connections.
- **Client-side**: Connects to the SSH server.

Basic connection:

```
ssh user@server_ip
```

If SSH runs on a non-default port (default is `22`):

```
ssh –p 222 user@server_ip
```

---

3. Server Options

- **Dedicated Server** → Entire physical server rented (high security/performance).
- **Virtual Server** → Shared server slices (cost-effective; minimal isolation risk).
- **Managed Server →** Provider manages updates/security (less control).
- **Unmanaged Server →** User manages updates/security (more control).

---

4. Renting a Server (Example: *DigitalOcean*)

Steps:

1. Create a *DigitalOcean* account.
2. Rent a **Droplet** (Virtual Server):
    - Select a server region, OS (Ubuntu), and plan.
3. Set authentication method (password initially, later SSH keys).
4. Connect via SSH (e.g., `ssh root@server_ip`).

---

5. Alternatives to Renting

Local Virtual Machine (VirtualBox)

- Set networking to **Bridged Adapter**.
- Install SSH Server:

```
sudo apt install openssh-server
```

- Connect using VM's IP address from host machine.

Localhost SSH Server (for practice)

- Install SSH Server locally:

```
sudo apt install openssh-server
```

- Connect to yourself:

```
ssh user@localhost
```

---

6. Changing SSH Default Port

SSH on port `222` for security (fewer automated attacks):

Edit /etc/ssh/sshd_config file:

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

Change/add line:

```
Port 222
```

Reload & Restart SSH:

```
sudo systemctl daemon-reload
sudo systemctl restart sshd
```

Connect with new port:

```
ssh –p 222 user@server_ip
```

---

7. Securing SSH (Public/Private Keys)

Secure server authentication using keys (instead of passwords)

Generate key pair:

```
ssh-keygen -t ed25519
```

- **Private key**: `~/.ssh/id_ed25519` (keep secret ❗)
- **Public key**: `~/.ssh/id_ed25519.pub` (upload to server)

Upload public key to server's `~/.ssh/authorized_keys` .

Disable password login:

Edit `/etc/ssh/sshd_config` and set:

```
PasswordAuthentication no
KbdInteractiveAuthentication no
UsePAM no
```

Restart SSH:

```
sudo systemctl restart sshd
```

---

8. Installing Apache Web Server

Install Apache:

```
sudo apt install apache2
```

- Default served directory → `/var/www/html`
- Default config → `/etc/apache2/apache2.conf`
- Website available at → `http://server_ip`

Apache logs:

- Access logs → `/var/log/apache2/access.log`
- Error logs → `/var/log/apache2/error.log`

---

9. Managing Linux Services (`systemctl`)

Control server processes like Apache via `systemctl` :

Command	Description
`sudo systemctl status apache2`	Check service status
`sudo systemctl start apache2`	Start service
`sudo systemctl stop apache2`	Stop service
`sudo systemctl restart apache2`	Restart service
`sudo systemctl enable apache2`	Enable auto-start at boot
`sudo systemctl disable apache2`	Disable auto-start at boot

`sudo systemctl reload apache2`	Reload configuration without restart

---

10. Uploading Files to Server (SCP & SFTP)

Using SCP (secure copy)

Syntax:

```
scp -rp -P port local_path user@server_ip:/remote_path
```

- `r` → recursive (folders/files)
- `p` → preserve modification timestamps
- `P` → custom SSH port (optional)

Example:

```
scp -rp -P 222 ./mywebsite root@123.456.78.90:/var/www/html
```

Using SFTP (graphical)

- Linux → Open File Browser and enter URL:

```
sftp://root@123.456.78.90:222
```

- Drag-and-drop files.
- **Recommended graphical tools**:
  - **Cyberduck**: cyberduck.io (Windows, Mac)
  - **FileZilla**: filezilla-project.org (cross-platform)

📌 Summary of Key Commands

Command	Description
`ssh user@server`	Connect via SSH
`ssh-keygen -t ed25519`	Generate SSH key pair
`scp -rp -P port local remote`	Securely copy files via SCP
`sudo apt install apache2`	Install Apache Web Server
`sudo nano /etc/ssh/sshd_config`	Edit SSH configuration
`sudo systemctl restart sshd`	Restart SSH daemon
`sudo systemctl start apache2`	Start Apache service
`sudo systemctl stop apache2`	Stop Apache service
`sudo systemctl restart apache2`	Restart Apache service

`sudo systemctl enable apache2`	Enable Apache to start at boot
`sudo systemctl disable apache2`	Disable Apache from starting at boot
`sudo systemctl status apache2`	Check Apache service status