

Forum:[Hack Forums](#)

Thread: [Harden SSH & Secure Remote Access \(Linux\)](#)

Vectura - October 5, 2025, 10:02 am

Harden SSH & Secure Remote Access (Linux)

Introduction:

This practical, legal tutorial shows how to harden SSH on Linux servers: key-based auth, fail2ban, firewalls, user restrictions and basic 2FA. Test everything in your own lab VMs.

Prerequisites: A Linux server/VM (Ubuntu/Debian/CentOS)

A separate client machine to SSH from

Basic shell knowledge

Steps (do this in your lab):

Create recovery admin user — make sure you have local/console access before locking SSH.

Generate an SSH key on client and copy it to server

Code # on client

```
ssh-keygen -t ed25519 -C "you@example.com"
ssh-copy-id -i ~/.ssh/id_ed25519.pub user@server
```

Edit SSH server config — open /etc/ssh/sshd_config and set:

Code PermitRootLogin no

PasswordAuthentication no

PubkeyAuthentication yes

AllowUsers youruser

ChallengeResponseAuthentication no

UsePAM yes

Code sudo systemctl reload sshd

Install and configure fail2ban to block brute-force:

Code sudo apt update && sudo apt install -y fail2ban

minimal /etc/fail2ban/jail.d/local.conf

[sshd]

enabled = true

maxretry = 5

bantime = 3600

Limit source IPs with firewall (optional)

Code sudo ufw allow from 203.0.113.0/24 to any port 22 proto tcp

sudo ufw enable

Optional: Add 2FA for SSH (Google Authenticator PAM) — test thoroughly in lab before deploying.

Lab exercises: Attempt a wrong password 6 times from attacker VM and verify fail2ban blocks it.

Verify only key-based login works and password login is rejected.

Notes: Always keep a recovery console and do not lock yourself out. Do this only on machines you own or are permitted to manage.

purplerain84 - *October 9, 2025, 11:57 am*

Hit me up for ssh/rdp access pm