# Server security

To add basic security to your node, we've provided a guide covering 2 simple tools.

- Uncomplicated Firewall (UFW)
- Key-based SSH authentication.
-

Raising The Security Floor Of The Cosmos Ecosystem

Within the #Cosmos, conversations around node security tend to start with whether or not you use backup servers, sentries, and a remote-signing key management solution. This does not see the forest for the trees. While those steps are certainly important, they are *final* security steps. We should instead be discussing the first steps you make when setting up a new Tendermint node; raise the floor of security, rather than the ceiling, if you will.

This is intended to be a very basic guide on Linux security practices. If you want to more in-depth information you can read about it here. The following topics will be covered: 1.SSH Key Setup

2.Server Configuration

3.Setting up a Basic Firewall

1. Using Local CLI Machinesbas

When you receive your server, you will be provided a root user login, and a password. You'll be inclined to log in with that login and password, but we have steps before we do that! We first want to create out ssh key as we'll be disabling password login shortly.

SSH Key Setup

Create SSH Key

An SSH (Secure Shell) key is a way to identify yourself as a user without using a password. It has 2 parts: the pubkey and private key. When you create the SSH key, you give your pubkey to a computer you wish to log into. You can then "show" the server your private key and it will admit you automatically. This makes it far more secure than a password, as then only you will have access to the server via your key.

This document assumes you're using a Mac. If you need instructions for Linux or Windows, see the Github instruction for generating an SSH key.

1. Open Terminal
2. Generate the SSH key:
3.

```

Copy ssh-keygen-ted25519-C"your_email@example.com"

```

Generate SSH key

1. When you're prompted to "Enter a file in which to save the key," press Enter. This accepts the default file location.
2. At the prompt, type a secure passphrase. For more information, see "Working with SSH key passphrases ."

```

Copy

> Enter passphrase (emptyfornopassphrase): [Type a passphrase]>Enter same passphrase again: [Type passphrase again]

```

Your SSH key is now created, but we have to add it to the agent for it to be usable.

Adding your SSH key to the ssh-agent

1. Start the ssh-agent in the background

2.

```

Copy eval"(ssh-agent-s)">Agentpid59566

```

1. Open your SSH config file

```

Copy open~/.ssh/config

```

Open ~/.ssh/config

1. Add the following text block to your file

```

Copy Host*AddKeysToAgentyesUseKeychainyesIdentityFile~/.ssh/id_ed25519

```

1. Add your SSH key to the ssh-agent

```

Copy ssh-add-K~/.ssh/id_ed25519

```

Your SSH key is now set up! This only has to happen once, so you can skip this if you need to refer back to this document.

Setting Up A Basic Firewall

Uncomplicated Firewall (UFW) is a program for managing a netfilter firewall designed for easy use. It uses a command-line interface (CLI) with a small number of simple commands and is configured withiptables . UFW is available by default in all Ubuntu installations after 18.04 LTS, and features tools for intrusion prevention which we will cover in this guide.

1. Start by checking the status of UFW.
2.

```

Copy sudoufwstatus

```

Check UFW status

1. Enable SSH

```

Copy sudoufwallowssh/tcp

```

1. Enable p2p

This is the default p2p port for Tendermint systems, but if you've changed the port, you'll need to update the ufw setting.

```

Copy sudo ufw allow 26656

```

1. Enable UFW

```
```

Copy sudoufwenable

```

1. Confirm UFW is enabled

```

Copy sudoufwstatus

```

Confirm UFW is enabled

Note that at any time you can disable ufw by doing:

```

Copy sudoufwdisable

```

Last updated5 months ago On this page ** * Raising The Security Floor Of The Cosmos Ecosystem * SSH Key Setup * * Setting Up A Basic Firewall *

Was this helpful? Edit on GitHub Export as PDF
```