

1.0 Setup Ubuntu Linux Base Server

- Introduction
- Initial Setup
 - Download
 - Your Own Home System
 - Hosting Company
 - Installing from Scratch in a Virtual Environment
 - Starting from a Server Hosted Solution
 - Server Overview
- Login
- Security First
- Synchronize Server Time
- Select a Locale
- Update Repositories
- Update Server
 - General Upgrades

Next Steps

Introduction

Describes production proven Bonsai Framework approach to setup of a Ubuntu server.

[Bash script version](#) is also maintained for quick setup.

Initial Setup

To follow through these tutorials you should have basic understanding of networks and know how to edit files on Ubuntu through terminal or sftp.

For security reasons may want to replace the account names in this tutorial especially setupadmin with something else of your choosing.

Download

If you are working for a large organization, it is recommended to use the LTS (Long Term Support) version,

<https://www.ubuntu.com/download/server>

Your Own Home System

Before paying for hosting, [practice on your desktop using virtualization technology](#). If you have minimal needs, you might want to [host from home](#).

Hosting Company

When using a hosting company, currently KVM based solutions the most cost effective where 1GB of RAM is required. See the [Bonsai Framework discussion on Hosting](#) for further details.

Cloud

You may also consider using Cloud with instructions here,

- [Azure](#) (to write though it's pretty point and click)
- [AWS](#)

Installing from Scratch in a Virtual Environment

If you are installing Ubuntu from scratch within a Virtual environment Ubuntu has a special Virtual option to [optimize performance](#).

These instructions are currently written and then updated for Ubuntu 12.10, 13.x, 14.04.2 (April 2015), 16.04.1 (Oct 2016)

You may use these instructions for other versions, but there may be slight differences in the order of events.

1. Select your language and press Enter.
2. On the install screen hit **F4**
 - a. Select Install a [minimal virtual machine](#) and press enter.
 - b. There will be no apparent change on install screen, but don't worry it worked.
3. Install Ubuntu Server should be highlighted by default (if not select it), press enter.
4. You will be prompted for Language, choose English.
5. When asked for Country, choose your country, in our case we use Canada.
6. When asked to configure the keyboard, provided you are using English,
 - a. Choose **No** to "Detect keyboard layout"?
 - b. Select English (US) to "Country of origin for the keyboard".
 - c. **English (US)** to "Keyboard layout".
7. For hostname, applying the [Bonsai framework naming standards](#) we will use, bonsaih0v00a
8. The first account is privileged. Use for Full name: **Setup Admin**
9. Username for your account: **setupadmin**
10. Password for you account. The [Bonsai Framework password standard](#) should suffice. By default we will not be enabling any services like ssh.
11. "Encrypt your home directory?" **No** (you can choose yes here, but in a closed server environment this will probably cause more head-ache then add to security)
12. When asked for your time zone, pick your time zone.
13. On Partitions disk choose,
 - a. **Ubuntu 18** - Choose entire normal and **do not go for LVM**. There appears to be a bug or change in install (I need to investigate) where LVM does not auto-partition for you.
 - b. **Ubuntu 17 and lower** - Choose **Guided - use entire disk and set up LVM** (Here's what [LVM](#) does).
14. When asked to select disk partition, usually you will only have one. By default that will be selected. Just push **Enter**.
15. "Write the change to disks and configure LVM?" Choose **Yes**.
16. For partition prompts choose **defaults**, how big for swap or root ect, just go with defaults. This used to be important but now with [LVM](#) and modern Linux systems this is no longer that important.
17. "Amount of volume group to use for guided partitioning:" **Continue**.
18. One final confirmation, "Write the changes to disk?" **Yes**.
19. HTTP proxy information. Unless you are behind a proxy (unlikely unless you on a company network), leave blank, **Continue**.
20. "How do you want to manage to upgrades on this system?" For maximum control, use, **No automatic updates**. For large enterprise environments consider [Ubuntu Landscape](#).
21. "Choose software to install:" do not select anything. If **OpenSSH server** is required, make sure to use a very very strong password and [switch to SSH key authentication](#) as soon as possible. Systems can be compromised within 1 hour.
22. Assuming this is a new server, **Yes** to "Install the GRUB boot loader to the master boot record?".
23. Installation complete. Choose **Continue** to reboot the system.

Starting from a Server Hosted Solution

With most [Virtual Server Hosting solutions](#) you will be asked to select a virtual server image operating system which is then built as defined by the hosting provider. Some providers also allow you to install from a linked iso image which provides the most fined grained control and security.

Some hosting providers modify the Ubuntu Setup or only let you load a pre-configured image. In that case they are just asking for credentials.

With older hosting provider they may ask for a root password. Make sure to select a VERY complex password for the root account. It might be a little annoying, but as you walk through the article you will find that we actually stop using root altogether.

Modern Cloud servers like Microsoft Azure will create a default user account or provide option to provide your public ssh keys. For now choose password. [SSH keys](#) with how to load them into your OS will be covered later.

Server Overview

In this example, the server is a virtual machine provided by a hosting company and the following attributes,

Server Name - tinman
Memory - 1024 MB
Disk Space - 40 GB

Our current [preferred](#) Virtual Server Hosting solution is [vServer Center](#) which also allows users to install Ubuntu from scratch. See the [Bonsai Framework discussion on Hosting](#) for further details.

Login

The very first thing to do is to log into the server. There are two main types of login, remote and console.

With a remote login, you are not physically at the machine. This is often accomplished using SSH. By default following these instructions SSH login will not be enabled.

With a console login, you are for all intents and purposes physically at the server. Most hosted solution provide a web based interface to simulate this type of behaviour.

Security First

I usually immediately install fail2ban,

```
sudo apt-get install fail2ban
```

You don't need to know what it does right away (other than it blocks a user from logging for a period of time after too many incorrect passwords) but it makes your box much safer.

If your system is on the Internet and you only have SSH login with a user name and password, make sure it is strong. Your password can be broken in within minutes.

You also want to go straight to [SSH RSA Key Based Authentication](#).

Synchronize Server Time

The ntp daemon calculates the drift of your system clock and continuously adjusts it.

Determine the systems current timezone,

```
date # determine current timezone
```

If you did a fresh setup, you should have the correct time zone and you can **skip** this step. If your timezone is incorrect, change it,

```
sudo dpkg-reconfigure tzdata # sets the timezone
```

Before proceeding to below instructions, please do an update on ubuntu setup

```
sudo apt-get update # updates your basic installation
```

Install the ntp daemon,

```
sudo apt-get install ntp
```

If you do not want to spare the processing power and memory, consider using other approaches [documented by the Ubuntu Community](#) or the [traditional cron approach](#).

Select a Locale

If you used a hosting company like Slice or Rackspace chances are the server image provided is minimal with no locale set. This will cause problems with Perl (used to install quite a few things) and in my case, it caused me head-aches setting up PostgreSQL.

```
locale # Look at your current locale.
LANG=en_CA.UTF-8
LC_CTYPE="en_CA.UTF-8"
LC_NUMERIC="en_CA.UTF-8"
LC_TIME="en_CA.UTF-8"
LC_COLLATE="en_CA.UTF-8"
LC_MONETARY="en_CA.UTF-8"
LC_MESSAGES="en_CA.UTF-8"
LC_PAPER="en_CA.UTF-8"
LC_NAME="en_CA.UTF-8"
LC_ADDRESS="en_CA.UTF-8"
LC_TELEPHONE="en_CA.UTF-8"
LC_MEASUREMENT="en_CA.UTF-8"
LC_IDENTIFICATION="en_CA.UTF-8"
LC_ALL=
setupadmin@tinman:~$
```

If the results show an error or the locale is not set up for UTF-8 as shown above, then run the following commands,

If you are not from Canada, you can choose another language, just make sure to use UTF-8.

```
sudo locale-gen en_CA.UTF-8
sudo update-locale LANG=en_CA.UTF-8
```

If you changed the locale, log off and then back on.

Finally, verify the locale changes took effect,

```
setupadmin@tinman:~$ locale
LANG=en_CA.UTF-8
LC_CTYPE="en_CA.UTF-8"
LC_NUMERIC="en_CA.UTF-8"
LC_TIME="en_CA.UTF-8"
LC_COLLATE="en_CA.UTF-8"
LC_MONETARY="en_CA.UTF-8"
LC_MESSAGES="en_CA.UTF-8"
LC_PAPER="en_CA.UTF-8"
LC_NAME="en_CA.UTF-8"
LC_ADDRESS="en_CA.UTF-8"
LC_TELEPHONE="en_CA.UTF-8"
LC_MEASUREMENT="en_CA.UTF-8"
LC_IDENTIFICATION="en_CA.UTF-8"
LC_ALL=
setupadmin@tinman:~$
```

Update Repositories

As of Ubuntu 8.10 it this has been simplified. We used to have to edit the sources.list to point your server to the universe and multi verse repository in order to install more popular packages. Now, even the minimal Server install already has what you need. So we can go straight to updating the indexes,

```
sudo apt-get update
```

If you are behind a proxy you may run into a [problem with how the proxy is caching](#). To resolve the GPG errors follow [my instructions on resolving proxy caching issues with Ubuntu Updates](#).

Update Server

General Upgrades

Next perform a dist-upgrade.

```
sudo apt-get dist-upgrade # upgrade to the newest kernel
```

Once you have applications and software running and stable, it is advisable to only perform incremental upgrades and judiciously perform dist-upgrades as appropriate.

During the update, all the update installers are downloaded to your system and kept there for future reference in case you need to re-install. With fast connections, this is not necessary. Clean up,

```
sudo apt-get autoclean # use this if you only want to clean out no longer
used packages
sudo apt-get clean # clean out all downloaded packages - I usually use this
one
sudo apt-get autoremove # cleans out unused packages
```

[Read Upgrading Ubuntu Packages](#) for more in depth details.

Since this is still a fresh system, restart the system,

```
sudo reboot
```

Next Steps

Continue to [Ubuntu Firewall Software](#).