Why You Should Not Use Arch

A new users guide to highly personalized, low maintenance operating system.

Artur Frącek

Intro

Arch is a very good Linux distribution so it is not a surprise that it has gained a lot of users in the last couple of years. That being said it seems to me like there are a lot of people installing Arch for the wrong reasons, choosing this distro for features that are not unique to Arch and ignoring some that are detrimental to how they are going to use it. I'm writing this guide to clear up a lot of common misconceptions about Arch while walking the reader through setting up Ubuntu in a highly personalized, possibly lightweight and simple way.

Why not Arch?

I'm not writing this document to discourage anyone from using Arch. I see Linux distributions as tools and think that you should just pick the right one for the job. I've talk to quite a few people recently who use Arch for tasks that could be accomplished more efficiently with other distributions. As reasons for choosing Arch I often hear that it's highly configurable and lightweight, these are important factors when choosing a distribution but not features unique to Arch. On the other hand Arch comes with some complications a lot of users seem to ignore.

One of the most common misconceptions about Arch is that it's hard to install, it isn't. Granted, there is no installation tool that holds your hand through it like there is with most distributions but the process is relatively simple and more importantly very well documented in the Arch Wiki. With most hardware configurations it means that no special skills or extensive knowledge are required, just some patience to read through the wiki and maybe to research some terminology that you don't understand if you are completely new to Linux.

From my observations what seems to pose a much bigger challenge to a lot of users is maintaining their newly installed operating system. This is because Arch developers chose to make their distribution user-centric rather than user-friendly which means that the user has complete control over the system but also all the responsibility to maintain it. This includes controlling and maintaining a lot of components of the system that some users may not be very familiar with or not wanting to spend the time on.

One of the best things about Arch is that it provides the users with the newest software in a form of binary packages, this includes the kernel, drivers and other basic components of the system. But it is up to the user to choose the rate of stagnation for these packages. This seems to be a common pitfall as a lot of people don't understand this and simply update all packages to the newest available version which often leads to problems. This is why a lot of people consider Arch to be an unstable operating system while the truth is it's as stable as you make it.

Why not Manjaro?

Manjaro is an Arch based distribution and the operating system of choice for a lot of users who are looking for something that requires less maintenance. The developers claim to provide all the benefits of Arch in a user-friendly environment but I don't see how that's possible since the biggest benefit of Arch is the complete control over the operating system and with Manjaro you trade off some of that control for ease of use. I find it hard to recommend this distribution to anyone because there are better, more stable distributions designed to be user friendly that use tools that were made to be user friendly in the first place.

Why Ubuntu?

What I'm trying to accomplish with this guide can be done with almost any distribution. I wanted to pick one that provides a stable base operating system that can be maintained with ease but still highly configurable and have a big range of packages available. I also considered Debian for it's semi rolling release model but in the end I settled on Ubuntu because of PPAs.

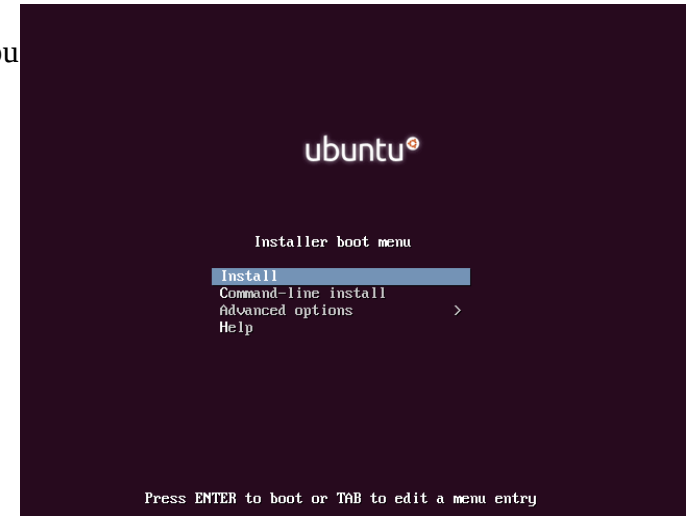
A lot of people don't think of Ubuntu as lightweight or highly configurable, especially since Unity became its default desktop environment. What you may not know is that you can install just the base Ubuntu with no display manager or desktop environment and then install your favorite software on top of it just like you would with Arch. It's called Ubuntu Minimal CD and this is what we will use as a starting point for this guide.

Preparation

To start with you will need the Ubuntu Minimal CD which you can download here [help.ubuntu.com] . Then simply create a bootable CD/DVD (How To [help.ubuntu.com]) or USB stick (How To [help.ubuntu.com]). Because the Minimal CD downloads all of the components of the OS from the Ubuntu repositories you will need to be connected to the internet during the installation process.

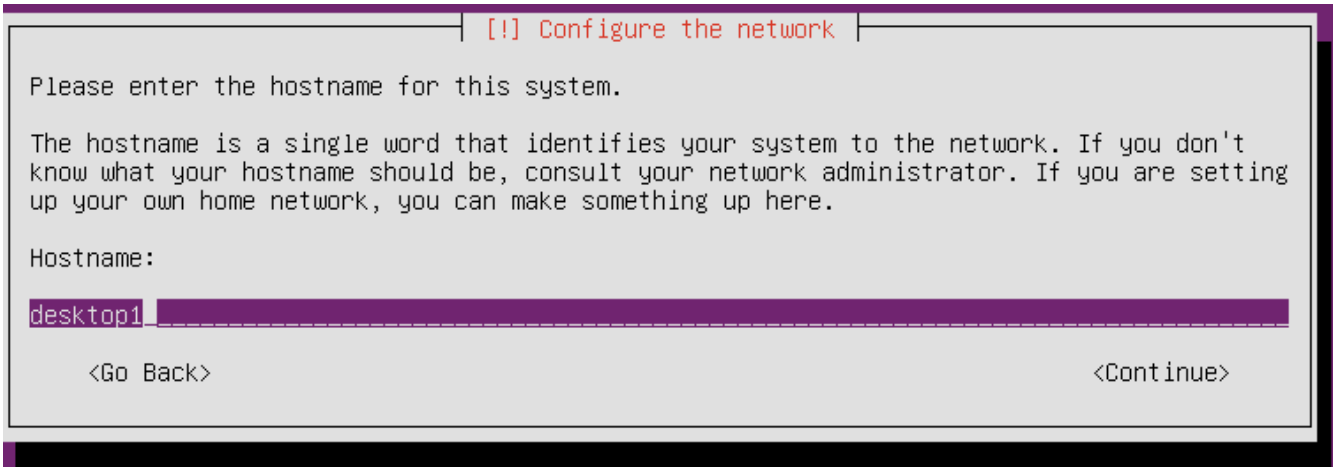
Installation

The installation of Ubuntu using the Minimal CD is relatively easy and it will look somewhat familiar if you have installed Arch while it was still using the Arch Installation Framework. I will still walk you through it so you know what to expect.



Once you select Install in the boot menu the installer will ask you for your preferred language, location, keyboard layout. If you don't know what what your keyboard layout is the installer can detect it by asking you a few questions about the characters on your keyboard.

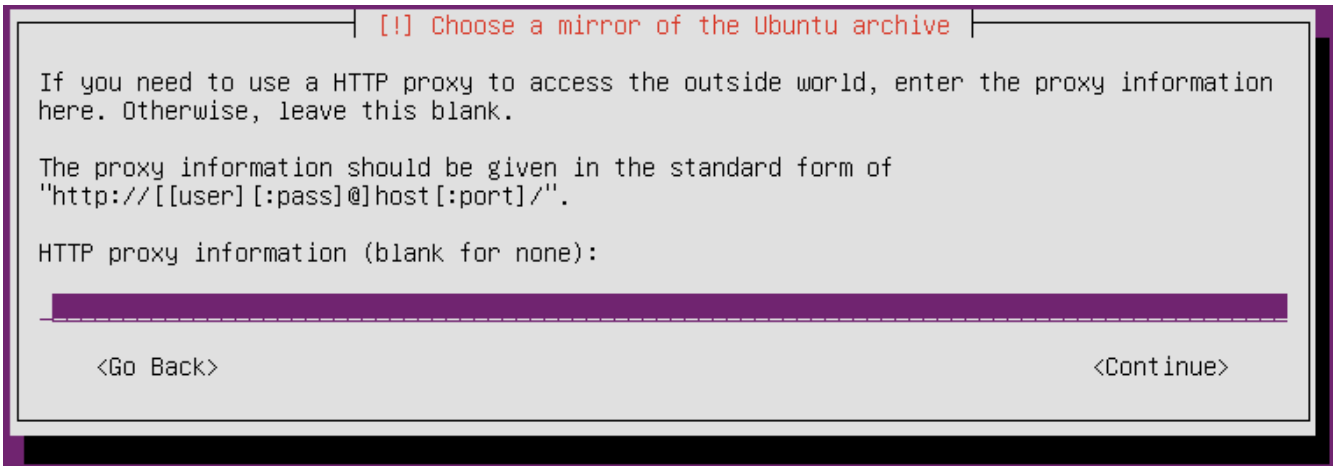
Next the installer will ask you for the hostname of the system. This is how your computer will identify on the network. You can call it anything but for convenience it's good to use a name that will make it easy to identify it on a list of devices, like desktop-office or laptop1.



Next you will have to choose the mirror you want to connect to. The Minimal CD is only ~30MB because it comes with none of the operating system components. It's just the installer and all the necessary files to make it boot. A mirror is a server that the installer will download all the packages from. Pick one that is closest to you since most of the time this is going to be the one that will allow the

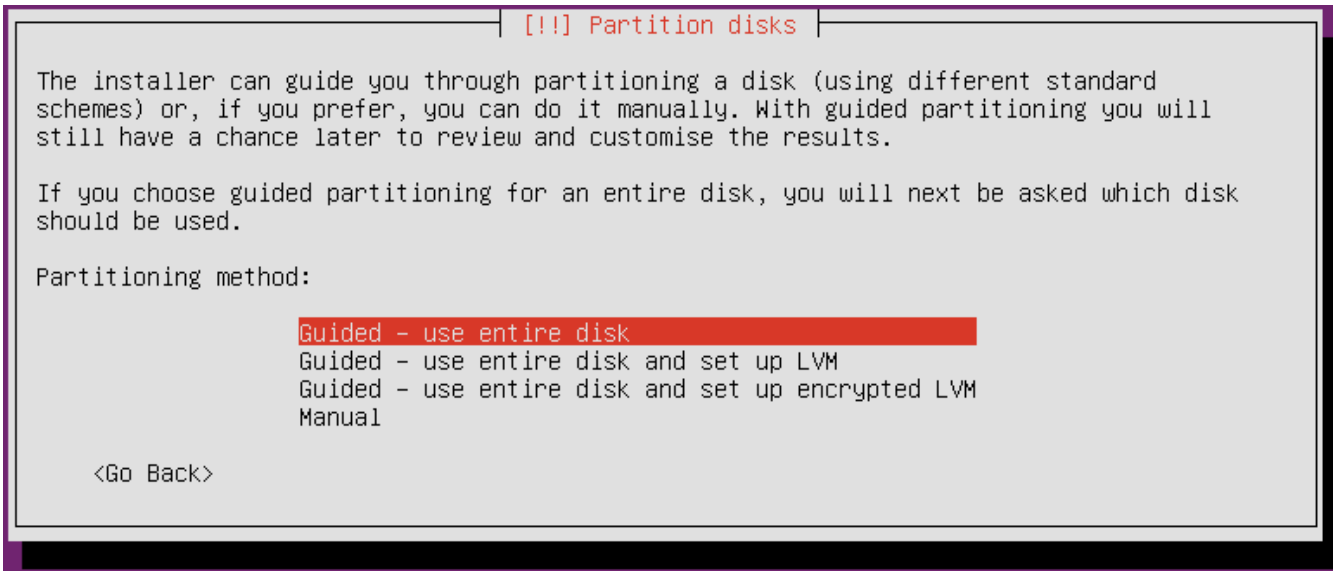
fastest download for you.

Then the installer will ask you for the HTTP proxy information. If you didn't need that information to connect any other devices to your network then you should just leave this field empty.



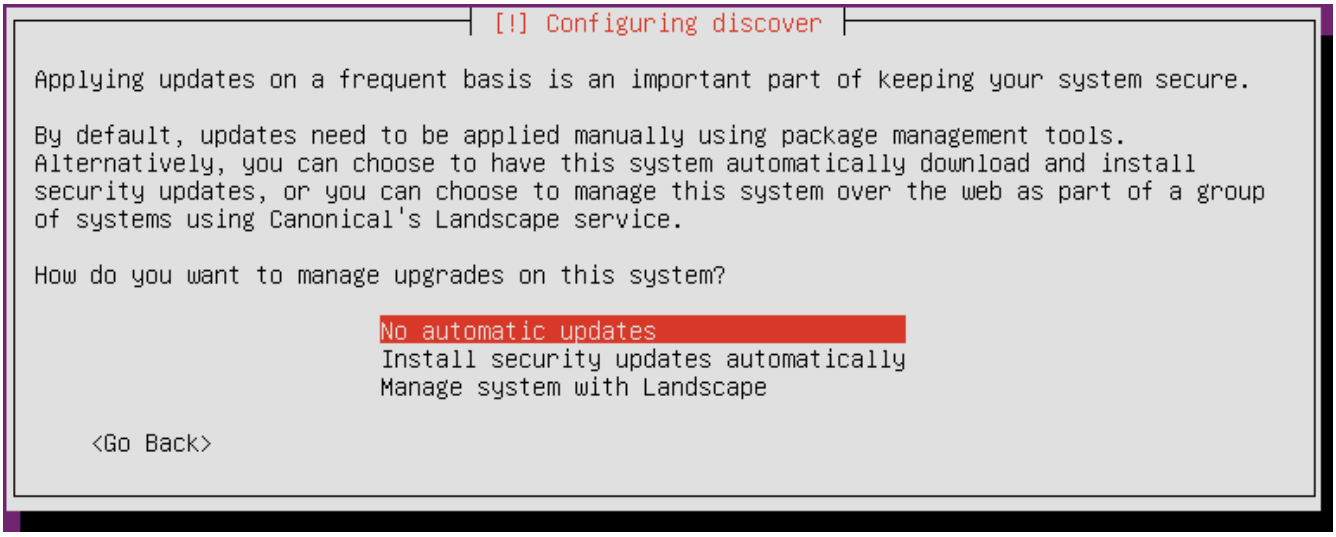
After you click Continue the installer will begin do download some additional tools from the mirror you selected and after that is done you will have to set up your user name and password and then the installer will ask you if you want to encrypt your home directory. Whether you want encryption or not depends on your personal preferences, you can read more about it here [askubuntu.com] if you need some information to decide.

Next you will be asked to select your time zone and after loading some additional components the installer will get to the disk partitioning portion of the installation. This is very straightforward if you want to use your whole disk for your Ubuntu installation, just select the Guided – use entire disk option. Otherwise things get a little complicated here and in order to be comprehensive I would have to write a whole separate guide. If you need some additional information for partitioning and how to set up partitions for dual booting with Windows I recommend you look here [askubuntu.com] and here [help.ubuntu.com].



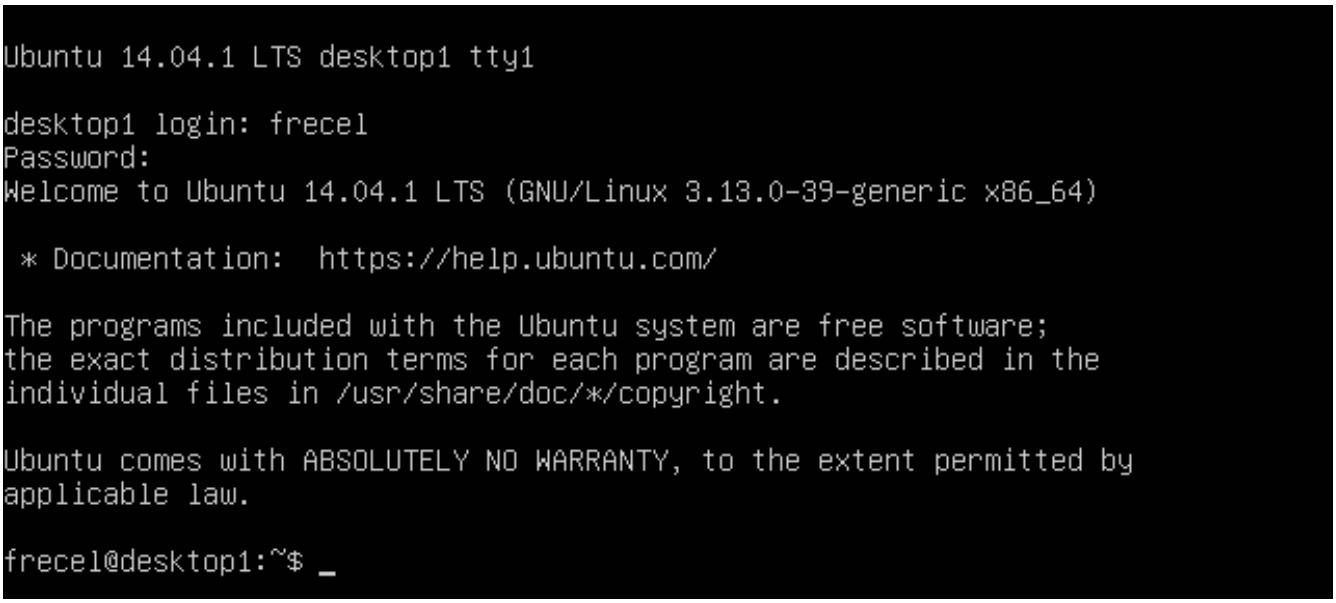
After that the actual installation of the base operating system will begin. After installing some packages you will have to set up how you want to configure your updates. For many it is very important to install security updates as soon as they are released but there are some risks involved in software updates without supervision. The decision between between automatic and manual updates is largely a matter of preference and can be changed later. The option to Manage system with Landscape is meant mostly for servers.

Next you will have an option to install some meta-packages, this is convenient if you already know what you want do with your operating system but these can also be manually installed after. For this guide I didn't select any and we will install everything manually.



The next step is the installation of a boot loader. This will vary depending on whether you have one or more operating systems installed. If you are not dual booting you can safely just select Yes and allow for the boot loader to be installed in the MBR.

After that you just have to tell the installer if your system clock is set to UTC (it generally is) and you will be ready to restart and run your newly installed operating system. You will be greeted by some gray text on black background just like you would on a brand new Arch installation.



Congratulations! You just completed the installation of your new operating system.

Creating a personalized experience

Now we get to the fun part of installing software to create a personalized operating system. This includes installation of a desktop environment [wikipedia.org] and/or window manager [wikipedia.org]. There are multiple DEs and WMs to choose from and multiple ways of doing this in ubuntu and the performance of your operating system will depend to some extend on these choices so I recommend that you spend some on researching what software you want to use.

It is important to keep in mind that Ubuntu repositories contain a lot of meta-packages which provide multiple ways of installing DEs and WMs. For example there are three ways to install KDE, one would be to install the *kde-plasma-desktop* package which contains just the the bare minimum needed to run KDE and basic applications like a terminal emulator, file manager and a text editor. The *kde-full* package contains all applications that are a part of the KDE project and the *kubuntu-desktop* installs all applications included in Kubuntu.

What you install now really comes down to personal preference, and the best advice I can give you here is just to have a basic plan of what you want your operating system to be. For example I want a lightweight system so I'm going to start by installing i3, a light tiling window manager. I can do this by using a command:

sudo apt install i3

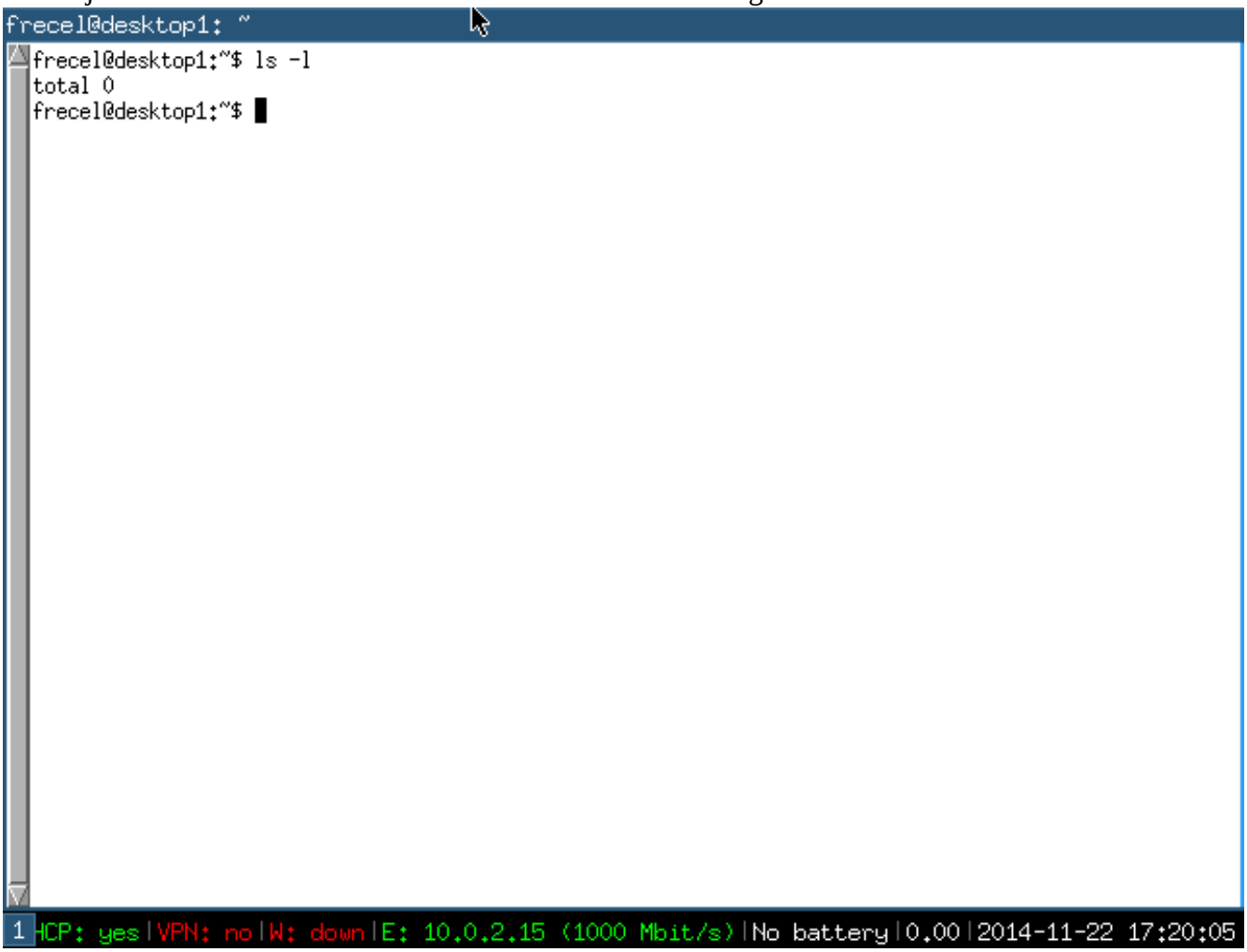
I can't get to the graphical interface yet I need a display server. There are multiple ways to install it, one of which is to install the *xinit* package and apt will install everything else we need as a dependency.

sudo apt install xinit

I'm also going to need a terminal emulator so I'm going to install urxvt

sudo apt install rxvt-unicode

Now I just need to run the command *startx* and I can start using i3.

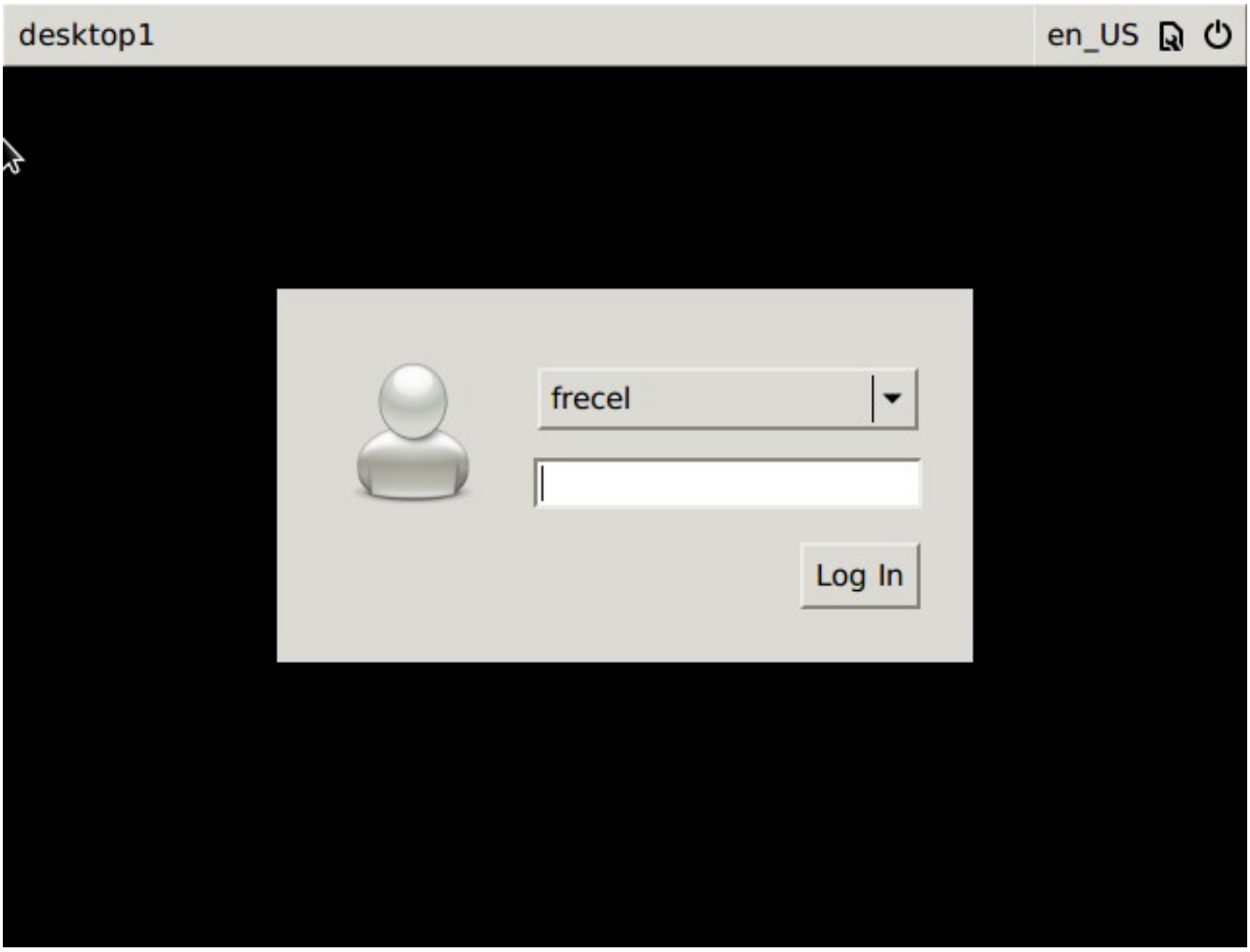


If I reboot now I will be greeted by text again because I have not installed a display manager. Since I'm sticking to lightweight software I'm going to install ligtdm with a gtk greeter.

sudo apt install ligtdm ligtdm-gtk-greeter

It will have to install quite a few packages but that's not an issue. A lot of it is part of GTK and it is a dependency for a lot of applications that I will use later like Firefox. We could shave of a few megabytes by not installing some of the recommended packages but I don't think thats necessary.

Now if I reboot I will be greeted by LightDM.



I could continue now and install other basic applications like a text editor, file manager and so on but I think that at this point you should know how to do this on your own.

PPAs

In order to improve the stability of the system packages in Ubuntu repositories are not updated as often as they are in Arch. This is because Arch assumes that the users will do all of the work of choosing which versions of the software works for them and Ubuntu package maintainers choose packages for Ubuntu users. This is very good for people who don't care about what version of the kernel or coreutils or other software that may not directly impact their work-flow they have installed and they just want it to work.

Sometimes you might still want to have a bleeding edge version of some software installed. This is where PPAs (Personal Package Archives) come in handy. PPAs are small non-standard repositories

published on launchpad. For example I like to use the nightly build of Firefox so I'm going to add the *ubuntu-mozilla-daily* ppa to my list of repositories. In order to easily manage PPAs we are going to need a *software-properties-common* package.

sudo apt install software-properties-common

Now we can just add the new PPA by running

sudo add-apt-repository ppa:ubuntu-mozilla-daily/ppa

Now all we have to do is update the package list and install the *firefox-trunk* package.

sudo apt update && sudo apt install firefox-trunk

This approach is great because I can still update my system with *sudo apt update && sudo apt upgrade* and Firefox will upgrade to the newest possible version while the rest of the system will stay at the versions that the maintainers decided will make it stable. I only have to take care of the things that are relevant to my work-flow and the rest is taken care of for me.

Where to go from here

If you installed Arch on a desktop before you probably already know what to do. Otherwise just use your new operating system, install more software and tweak it to your liking. If you have never installed any bare bones distribution before you might be surprised how many elements of the operating system that you might consider basic necessities are still missing. It might take some time for it to feel complete but it's worth the effort because when you are done you will have an operating system that is truly yours.

If you need some inspiration on how to tweak your system I recommend visiting reddit.com/r/unixporn.