Getting to Know Your System

Before we overwrite our Panics with a new OS, we need to get to know our machines. Using the command line, please determine the:

- Amount of memory, and
- IP address

of your Panic. Write these down because we will use them later. (You may find `man ifconfig` and the pseudo-file `/proc/meminfo` useful.)

Preparing to Install

Go to the Ubuntu download page and download Ubuntu 16.04 LTS. The download may take a while, so please read up on Ubuntu while you wait: Ubuntu is one of several Linux distributions. Ubuntu it is derived from Debian; you will often read references to its Debian roots, and they are still very similar in many ways. However Ubuntu is maintained by Canonical, so the communities have diverged.

Like many large-scale software projects, Ubuntu follows a regular release schedule. The release numbers do more than just denote a sequence—they have meaning.

```
> What is the significance of the Ubuntu release number?
    (What is a major number? A minor number? LTS?)
> How long is each relase supported?
> How is this schedule related to the Linux Kernel's release schedule?
```

Once your download is complete, you want to verify your download. Your OS is one of the most important pieces of software on your machine. We place a large amount of trust in our OS and we rely on its integrity. For this reason, we will use hashing to compare the file we downloaded to Canonical's posted hash. This should ensure that the file we receive was neither replaced by a malicious user nor corrupted during transfer or storage. Please follow the instructions at the bottom of the download page to verify your downloaded `.iso` file.

```
> How do we generate a hash of a file from the command line?
> What are the recommended hash functions for integrity checks?
> What is the birthday paradox, and how does it relate to hashing?
```

Now that you have downloaded and verified your live USB ISO, we will create a bootable drive. The easiest way is to use the Startup Image Creator GUI utility. A bootable drive is very useful. You can use it to run Ubuntu *without* installing it on a machine. You may choose to allocate space on your USB drive to hold extra data for this purpose. You can also use a bootable USB drive to perform many tasks, such as rescuing system, partitioning drives, resetting passwords, and many other administrative tasks.

Installing Ubuntu

Before we can install our OS, we need to understand how a system "boots". Please read about BIOS (and the newer UEFI) here. When a computer boots, it needs to figure out what hardware is available, and then load an OS from the appropriate media. The "appropriate media" is determined by a configurable priority order. (If the first option is present, the system boots from that media, if not, it checks for the next option.) To boot from our USB drive, we want it to be at the top of our priority list. Then, when we have successfully installed Ubuntu, we can remove the USB drive and BIOS should boot from the hard disk (or whatever is next in line. . .).

The Startup Image Creator formatted our USB drives so that they are "bootable". However, when we install Ubuntu, we will be overwriting our PC's hard drive with a new OS. Thus, we must make sure our hard drive is properly formatted and bootable. To do this we must (1) partition our hard drive, (2) format each partition with an

appropriate file system, and (3) select the location in the file system namespace hierarchy that each partition will be mounted.

```
> What are some of the reasons to partition a hard drive?
> What is the relationship between a partition and a file system?
> What is the default Linux file system?
> What is the difference between a Primary, Extended, and Logical partition?
```

Now that we understand the basic concepts behind partitioning, there are extra wrinkles to consider. After we divide a physical disk into logical, isolated units (partitions. . .), our OS needs to know where one partition ends and the next begins. This information is stored in a partition table in one (or more) locations on the disk so the OS knows how to find it. Of course, there are multiple ways to do this: MBR and GPT. GPT is newer and better.

We are almost ready to install Ubuntu. We can set our boot order in the BIOS so that our PC boots from our Live USB. We know the basics of file systems and disk partitioning. But knowing *how* to make a partition doesn't tell us *which* partitions to make.

You can find several "rules of thumb" for choosing your Linux partition scheme. Most of them made sense at the time they were written, but storage capacities and speeds are constantly changing. I would highly recommend:

- A small amount of space for an EFI partition at the start of your disk (> 36MiB required)
- A swap partition 1.5-2x the size of RAM
- A reasonable amount of space in an `ext4`-formatted partition mounted at `/`
- A reasonable amount of space in an `ext4`-formatted partition mounted at `/tmp`
- A large amount of space in an `ext4`-formatted partition mounted at `/home`

So please partition use your Live USB drive to install Ubuntu. When prompted with options to partition your drive, please select "something else" and manually create a reasonable partition scheme. The user that you create will have administrator privileges. Please choose an appropriate username and password, write them down, and give them to Mary. This is important because, although unlikely, it is possible that an attacker gains access to your system. Mary needs to be able to stop any ongoing attacks and diagnose their severity.

Minimal Configuration

Now that your Panic is up and running a fresh Ubuntu 16.04.1 LTS, it is time to do some basic setup. The first thing you need is the ability to modify files. Please choose your favorite text editor and install it. For many text editors, you may want to use `apt`, the Ubuntu package manager. We will be using `apt` very frequently. To install a package using `apt`, you use the command:

```
$ apt install <package>
```

There are packages for `emacs`, `vim`, and `nano`. To install other editors, like `atom` (download the .deb) or `sublime` (Ubuntu 64-bit), you may need to use your web browser.

Installing software requires administrator privileges. Luckily your user has those privileges. However, you must tell the system that you wish to execute a command as a super-user. At the command line, you do this with

```
$ sudo <command>
```

```
> What common tools/utilities that you use are installed by default?
> What are the names of additional packages that you can't live without?
```

Now that we have installed a text editor, we want to configure our Panic's network. To make sure that our Panics have a consistent IP address, we want to assign it a static IP. Unfortunately, the default network manager makes this difficult. We need to remove the network manager software, and then edit our OS's network configuration directly. Please open the file `/etc/network/interfaces` in your favorite text editor, and add a few new lines. It should look like this (with <my IP> replaced by the IP address you recorded at the very start):

```
auto lo
iface lo inet loopback
address <my IP>
netmask 255.255.252.0
network 137.165.8.0
broadcast 137.165.11.255
gateway 137.165.8.1
dns-search cs.williams.edu
dns-nameservers 137.165.8.3 137.165.8.149

> What are the different components of an IP address?
> What is a DNS server?
> How is your computer's hostname mapped to an IP address?
> Why might it be important for a machine to have a static IP?
```

Now that we have edited our network configuration, we can remove the network manager. If `apt get` installs a package, `apt remove` removes one. We can optionally use the `--purge` flag to also remove the configuration files associated with a given package. Please remove the network manager with:

```
$ sudo apt remove --purge network-manager*
```

The last thing we need to do is harden our system against intrusions. Since our Panics are publicly visible from the wider Internet, we need to make it hard for malicious users to get into our system. Bots will attempt to guess your password and fail repeatedly. The `denyhosts` utility will monitor log messages for multiple failed `ssh` attempts, and add any IP addresses that fail 3 times in a row to a blacklist. Please install `denyhosts` to prevent brute force `ssh` attacks.

```
$ sudo apt install denyhosts
```