Ubuntu

Download Ubuntu 18.04 and Burn ISO

1. Go to [Ubuntu Desktop download page and download Ubuntu 18.04](https://www.ubuntu.com/download/desktop).
2. After downloading the ISO file, burn the image on to a disk with [unetbootin tool](https://unetbootin.github.io/) (Mac users) or the [Rufus tool](https://rufus.ie/en_IE.html)(Windows) on to a USB thumb-drive.
3. Insert thumb-drive into mining-rig | super-computer.

Install Ubuntu on Mining Rig

1. After turning on the mining-rig, immediately hit the F2-key.
2. Click on the boot-menu.
3. Make sure the sandisk device containing the Ubuntu 18.04 iso file is selected.
4. Save and Exit

Updating Ubuntu

1. Hit the Option-key or Windows key on keyboard and type in “Software Updater” within the Ubuntu-OS search-field and then hit enter; that should prompt the Ubuntu-updater- upon completion, select the “Install Now” button and then enter system-password.
2. After running the Ubuntu-updater, restart the system.
3. For the sake of thoroughness, update Ubuntu 18.04 from the command-line by running the following commands in sequence.

* sudo apt-get update # Fetches the list of available updates
* sudo apt-get upgrade # Strictly upgrades the current packages
* sudo apt-get dist-upgrade
* sudo reboot

1. After running the above commands, restart Ubuntu or simply type sudo reboot from the command-line.

Ubuntu Preparation for SSH (Optional)

1. Download net-tools by running sudo apt install net-tools
2. Run ifconfig -a or ip addr show to identify the host-IP address or name to access via SSH.
3. Prepare Ubuntu-environment for SSH

* sudo apt-get install openssh-server
* sudo ufw allow 22
* sudo reboot

1. Download [nMap](https://nmap.org/download.html) (optional).
2. Run nmap -sn your-device-ip-address to ping Ubuntu instance; it should return the name of the Ubuntu-host or computer name in additon to the IP-address within the terminal output.

* e.g. nmap -sn 10.3.1.233

1. Run ssh -v username@your-device-ip-address or ssh -v username@computer-name

* e.g. ssh -v alexanderjsingleton@10.3.1.233 or ssh -v alexanderjsingleton@alexanderjsingleton-ubuntu

How to Install Nvidia GeForce GTX 1080 Graphics Drivers

Updating GPU Drivers from Graphical User Interface (GUI)

1. Hit the Activities icon within the bottom left corner of the screen and click-on “Settings” OR hit the Option-key and search for “Settings”.
2. Next click on “Details” within the “Settings” window.
3. Since we haven’t loaded any drivers yet, Ubuntu 18.04 pre-loaded the instnace with whatever your on-board CPU carried (e.g., on my rig, within the Settings > Details, the “Graphics” field displays “Intel HD Graphics 630 (Kaby Lake GT2)”.
4. After verifying the aforementioned, go back to the Activites-page within the lower left-corner of the desktop and select “Software & Updates” or hit Option-key and search for “Software & Updates”.
5. Within the Software & Updates window, click on the “Additional Drivers” tab.
6. After a few moments, Ubuntu will return a suggested driver to download- by default it will select the second option radio-button parenthetically described as (open source) but you want the radio-button option parenthetically described as “proprietary, tested” (e.g. in my case it was “Using NVIDIA driver metapackage from nvidia-driver-390 (proprietary, tested).
7. After selecting the previously mentioned radio-button option, select “Apply Changes” and then reboot the computer.
8. After rebooting the computer, go back to Settings > Details- verify that the new drivers were installed by checking Graphics description field (it should say (nvidia-driver-390).
9. Go back to the Activites-icon page within the lower left-corner of the desktop and select “Nvidia X Server Settings” from the application menu; click on the “Prime Profiles” tab on the left-pane, and then select Intel card on the right-pane.
10. Select the NVIDIA (Performance Mode) radio-button and then close the window.
11. Proceed to ‘Installing Hashchat’ Section.

Updating GPU Drivers from the Command-Line (Recommended)

1. Access the Terminal.
2. Run sudo lshw -c display OR sudo lshw -c video to display the Ubuntu 18.04 stock-drivers loaded for the Nvidia GeForce GTX 1080 GPU cards.
3. Since we haven’t installed any drivers from the command-line yet, the driver-detail within the configuration field should display “driver=nouveau”.
4. Run sudo ubuntu-drivers devices
5. After running the above command, only two drivers will be displayed (e.g. ‘nvidia-driver-390 – distro non-free recommended’ and ‘xserver-xorg-video-nouveau – distro free builtin’); we want the latter ‘non-free recommended’- so run sudo apt install nvidia-driver-version-number (e.g. sudo apt install nvidia-driver-390).
6. After installing the drivers from the command-line, reboot the computer by running sudo shutdown -r now.
7. If after rebooting your welcome-screen hangs, click on the settings-wheel icon, select the “Ubuntu on wayland” and then re-enter password.
8. Access the terminal and run sudo lshw -c display; the driver field should now display “nvidia” instead of “nouveau”.

Installing the Nvidia CUDA Toolkit 10.0

Installing the Nvidia CUDA Toolkit 10.0 for Ubuntu 18.04 – Option A (Recommended)

1. sudo apt update && apt dist-upgrade -y && reboot
2. lspci -v
3. sudo apt install -y ocl-icd-libopencl1 nvidia-driver nvidia-cuda-toolkit.

* e.g. sudo apt install -y ocl-icd-libopencl1 nvidia-driver-390 nvidia-cuda-toolkit

1. nvidia-smi
2. sudo reboot
3. Access terminal and enter the following commands in sequence:

* sudo add-apt-repository ppa:graphics-drivers/ppa
* sudo apt-get update
* sudo apt-get install mesa-common-dev
* sudo apt-get install freeglut3-dev

1. Reboot the computer by running sudo reboot

Installing the Nvidia CUDA Toolkit 10.0 for Ubuntu 18.04 – Option B

1. Go to [Nvidia CUDA Toolkit 10.0 for Ubuntu 18.04](https://developer.nvidia.com/cuda-downloads).
2. Select the following Select Target Platform Window: Linux > X86\_64 > Ubuntu > 18.04 > runfile (local)

* To check your OS architecture run uname -a it should return “x\_86\_64” within the terminal display.

1. Download the Base Installer (2.0 GB) by click on the Download button.
2. Access Terminal and [close X Server to avoid errors while updating nVidia driver](https://unix.stackexchange.com/questions/25668/how-to-close-x-server-to-avoid-errors-while-updating-nvidia-driver) by running sudo init 3; a black-screen will appear and prompt you to enter your Ubuntu username and password- please enter.
3. After logging-in, run cd ~/Downloads.
4. Run sudo sh cuda\_10.0.130\_410.48\_linux.run
5. Scroll down to accept the conditions by holding down the “Return” key or space-bar- upon scrolling to the bottom, type: accept; alternatively, you can hit ‘CTRL + D’ to bypass scrolling.
6. Install the CUDA 10.0 Toolkit by enter y
7. Do you want to Install the OpenGL Libraries y
8. Do you want to run nvidia-xconfig? Enter y
9. Install the CUDA Toolkit? Enter y
10. Unless otherwise preferred, hit the “Return” key to enter the specified default absolute-path (e.g. /usr/local/cuda-10.0).
11. Enter y to install a symbolic link at /usr/local/cuda.
12. Enter y to Install the CUDA 10.0 Samples
13. Hit the “Return” key to enter CUDA Samples default location-path (e.g. /home/alexanderjsingleton).
14. After the install, open the X Server by running sudo init 5; if for whatever reason you’re unable to reboot the computer after installing the CUDA 10.0 Toolkit, turn-off the computer and reboot- remember to click on the Settings icon and select “Ubuntu on wayland” before entering the username and password.
15. Access terminal and enter the following commands in sequence:

* sudo add-apt-repository ppa:graphics-drivers/ppa
* sudo apt-get update
* sudo apt-get install mesa-common-dev
* sudo apt-get install freeglut3-dev

1. Reboot the computer by running sudo shutdown -r now

Installing OpenCL™ Runtimes for Intel® Processors

Installing OpenCL™ Runtimes for Intel® Processors – Option A (Recommended)

1. Check your CPU system processor architecture by running sudo lscpu to identify the class of CPU e.g. “Intel (R) Core (TM) i7-7700 CPU @ 3.60Ghz”- alternatively, you can go to Settings > Details which will also reveal the CPU-family (e.g. Intel (R) HD Graphics 630 (Kaby Lake GT2)).
2. Go to the [Intel Developer Zone](https://software.intel.com/en-us/articles/opencl-drivers#graph-lin-ubuntu) and scroll down to the corresponding download link for your system; assuming you’re running Ubuntu 18.04 and have a Intel Core-class CPU, scroll down to “Linux\* OS Ubuntu\* 16.04.x (deb)” and click on the [Github link for 18.46.11837 Runtimes](https://github.com/intel/compute-runtime/releases) to access the runtime-packages to download.
3. There should be instructions within the aforementioned Github repo- or you can proceed with the following in the next steps.
4. cd ~/Desktop
5. mkdir neo
6. cd neo
7. sudo wget https://github.com/intel/compute-runtime/releases/download/18.46.11837/intel-gmmlib\_18.4.0.348\_amd64.deb
8. wget https://github.com/intel/compute-runtime/releases/download/18.46.11837/intel-igc-core\_18.44.1060\_amd64.deb
9. wget https://github.com/intel/compute-runtime/releases/download/18.46.11837/intel-igc-opencl\_18.44.1060\_amd64.deb
10. wget https://github.com/intel/compute-runtime/releases/download/18.46.11837/intel-opencl\_18.46.11837\_amd64.deb
11. Install the packages by running sudo dpkg -i \*.deb.
12. Restart the system by running sudo reboot.

Installing OpenCL™ Runtimes for Intel® Processors – Option B

1. Go to [Intel OpenCL Runtimes website](https://software.intel.com/en-us/articles/opencl-drivers) and scroll down to the bottom of the page to Download Intel CPU Runtime for OpenCL Applications 18.1 for LINUX OS (64-bit only).
2. After downloading the OpenCL Runtimes for Intel Processors, cd ~/Downloads
3. Run sudo tar -xzf l\_opencl\_p\_18.1.0.013.tgz to unpack the tgz file- please note, the OpenCL file version may update since the time as of this gist (ie as of this gist, the version is 18.1.0.013.)
4. cd l\_opencl\_p\_18.1.0.013
5. Run sudo ./install.sh
6. A welcome screen is prompted- hold-down the “Enter” key until able to input select “2” for “I do NOT consent to the collection of my Information”
7. Hit “1”
8. Hit “1”
9. Hit “Enter” and then restart the computer.
10. If any additional Intel OpenCL errors are encountered after testing hashcat, go to section ‘Installing OpenCL™ Runtimes for Intel® Processors – B’- if not, proceed with section “Testing Hashcat”

Hashcat

Downloading Hashcat Binaries – Option A (Recommended)

1. Download the latest version of [hashcat](https://hashcat.net/hashcat)– scroll down to the bottom and download the version at the top of the table (as of this gist, the latest version is v5.1.0.7- be sure to download the “binaries” version of Ubuntu); use the wgetcommand to download the latest version of hashcat from the hashcast website (e.g. wget https://hashcat.net/files/hashcat-5.1.0.7z– be sure to cd into the Downloads folder for the sake of consistency).
2. Access the download-directory: cd ~/Downloads
3. If you haven’t already, download p7zip: sudo apt install p7zip
4. Assuming you’re still in the Downloads folder, run sudo p7zip -d hashcat-5.1.0.7z to unzip (if that command didn’t work, simply right-click and extract file within the Downloads folder).
5. Access the unzipped hashcat folder by running cd hashcat-5.1.0.7z
6. Next run ls /usr/bin/ | grep -i hash
7. Run sudo cp hashcat64.bin /usr/bin/
8. sudo ln -s /usr/bin/hashcat64.bin /usr/bin/hashcat
9. sudo cp -Rv OpenCL/ /usr/bin/
10. sudo cp hashcat.hcstat2 /usr/bin/
11. sudo cp hashcat.hctune /usr/bin/

Downloading Hashcat from Ubuntu Repository – Option B

Alternatively, you may download hashcat from the Ubuntu repository by simply running sudo apt-get install hashcat; however, this will more than likely download a deprecated version of hashcat.

Testing Hashcat

1. Test hashcat, by running sudo hashcat --benchmark.
2. As of this gist, hashcat is reporting that Intel’s OpenCL runtime (GPU only) is currently broken and they are currently waiting for updated OpenCL drivers from Intel; if you encounter that error message try running sudo hashcat --benchmark --force that should successfully override any OpenCL issues.
3. After running sudo hashcat --benchmark or sudo hashcat --benchmark --force the terminal output will report hashrates for given hash-algorithms (eg. ‘MD5 560.4 KH/s’, ‘SHA1 149.4 MH/s’, ‘SHA2-256 72485.9 Mh/s’ etc.)
4. Hit CTRL + C to stop benchmark testing.

Creating Password Hashes

1. Go to the Desktop by running cd ~/Desktop
2. Create a new folder within the desktop (e.g. mkdir infosexy).
3. cd infosexy
4. Create a new folder within infosexy directory for the password dictionary (e.g. mkdir skull)
5. cd skull
6. Go to [skullsecurity](https://wiki.skullsecurity.org/Passwords) and download the ‘rockyou.txt’ password dictionary.
7. Assuming you downloaded the ‘rockyou.txt’ file go to the ‘Downloads’ folder: (e.g.cd ~/Downloads).
8. Run bzip2 -d rockyou.txt.bz2 to unpack the file-folder.
9. Copy the ‘rockyou.txt’ dictionary from the Downloads-folder to the destination skull-folder- easiest way to do this is launch two terminals by opening another terminal-window from Terminal (e.g. CTRL + ALT + T) and then enter pwdfrom both terminal-windows to identify their respective paths.
10. Copy source-folder file ‘rockyou.txt’ by running the cp command between file-paths (e.g. cp /home/alexanderjsingleton/Downloads/rockyou.txt /home/alexanderjsingleton/infosexy/skull).
11. Run ls to confirm file transfer.
12. Go back into the infosexy folder (e.g. cd ..)

Cracking the MD5 Hash Algorithm

1. Within the infosexy directory, run the following commands below in sequence to create password hashes encrypting the passwords with the MD5 hash-algorithm.
2. echo -n “Password” | md5sum | tr -d ” -” >> target\_hashes.txt
3. echo -n “HELLO” | md5sum | tr -d ” -” >> target\_hashes.txt
4. echo -n “MYSECRET” | md5sum | tr -d ” -” >> target\_hashes.txt
5. echo -n “Test1234″ | md5sum | tr -d ” -” >> target\_hashes.txt
6. echo -n “P455w0rd” | md5sum | tr -d ” -” >> target\_hashes.txt
7. echo -n “GuessMe” | md5sum | tr -d ” -” >> target\_hashes.txt
8. echo -n “S3CuReP455Word” | md5sum | tr -d ” -” >> target\_hashes.txt
9. Run cat target\_hashes.txt to display all of the encrypted password with the md5sum algorithm within the Terminal window.
10. Run sudo hashcat -m 0 -a 0 -o cracked.txt target\_hashes.txt skull/rockyou.txt
11. I have 8 \* Nvidia Asus GeForce GTX 1080 \* 8 GB GPUs on my mining-rig- cracking-time may vary per computer; I cracked this set within less than one second- needless to say that is really fast.
12. Run sudo cat cracked.txt to reveal the hash key-value-pairs corresponding to the encrypted passwords within the Terminal window.

Cracking the SHA256 Hash Algorithm

1. Within the infosexy directory, run the following commands below in sequence to create password hashes encrypting the passwords with the SHA256 hash-algorithm.
2. echo -n “Password” | sha256sum | tr -d ” -” >> target\_hashes-2.txt
3. echo -n “HELLO” | sha256sum | tr -d ” -” >> target\_hashes-2.txt
4. echo -n “MYSECRET” | sha256sum | tr -d ” -” >> target\_hashes-2.txt
5. echo -n “Test1234″ | sha256sum | tr -d ” -” >> target\_hashes-2.txt
6. echo -n “P455w0rd” | sha256sum | tr -d ” -” >> target\_hashes-2.txt
7. echo -n “GuessMe” | sha256sum | tr -d ” -” >> target\_hashes-2.txt
8. echo -n “S3CuReP455Word” | sha256sum | tr -d ” -” >> target\_hashes-2.txt
9. Run cat target\_hashes-2.txt to display all of the encrypted passwords with the SHA256 algorithm within the Terminal window.
10. Run sudo hashcat -m 1400 -a 0 -o cracked-2.txt target\_hashes-2.txt skull/rockyou.txt
11. I have 8 \* Nvidia Asus GeForce GTX 1080 \* 8 GB GPUs on my mining-rig- cracking-time may vary per computer; I cracked the first hash for “Password” within less than one second- needless to say that is really fast, considering SHA256 is the same encryption-grade protecting the Bitcoin network.
12. Run sudo cat cracked-2.txt to reveal the hash key-value-pairs corresponding to the encrypted password within the Terminal window.