CANONICAL

MAAS 2 Advanced NUC Installation and Configuration — Scripted

(The MANIACS Document)

Contents

Purpose

This document describes how to install MAAS on a computer so that you can deploy systems in a test environment as well as install the certification tools and perform certification testing. Consult the Ubuntu Certified Hardware Self-Testing Guide (available from https://certification.canonical.com) for detailed information on running the certification tests themselves.

In this document, the MAAS server is referred to generically as a “portable computer” because the intent is that the MAAS server (such as an Intel NUC or laptop) be portable for field technicians; however, you can deploy a desktop computer or server in exactly the same way.

A computer configured as described here is not intended for general Internet use. Some settings relax security in the interest of ease of use, so you should limit use of the portable computer on the Internet at large.

This document begins with information on the required hardware and then moves on to a general description of Ubuntu installation, details on how to install and configure MAAS, and how to test your MAAS installation. Appendixes cover more esoteric or specialized topics, including how to add support for i386 (32-bit) images and how to set up advanced network configurations.

Figure 1 illustrates the overall configuration that this document will help you create. This document describes configuration of the Portable Computer device in the figure. It presupposes the existence of a local LAN that the portable computer can use for external connections, as well as the availability of at least one SUT for testing at the end of the process. (Note that the Internet connection is required for initial setup, but a properly-configured MAAS server does not need this connection to bring up SUTs.) Once configured, you will be able to move the portable computer from one site to another, repopulating the MAAS LAN at each site.

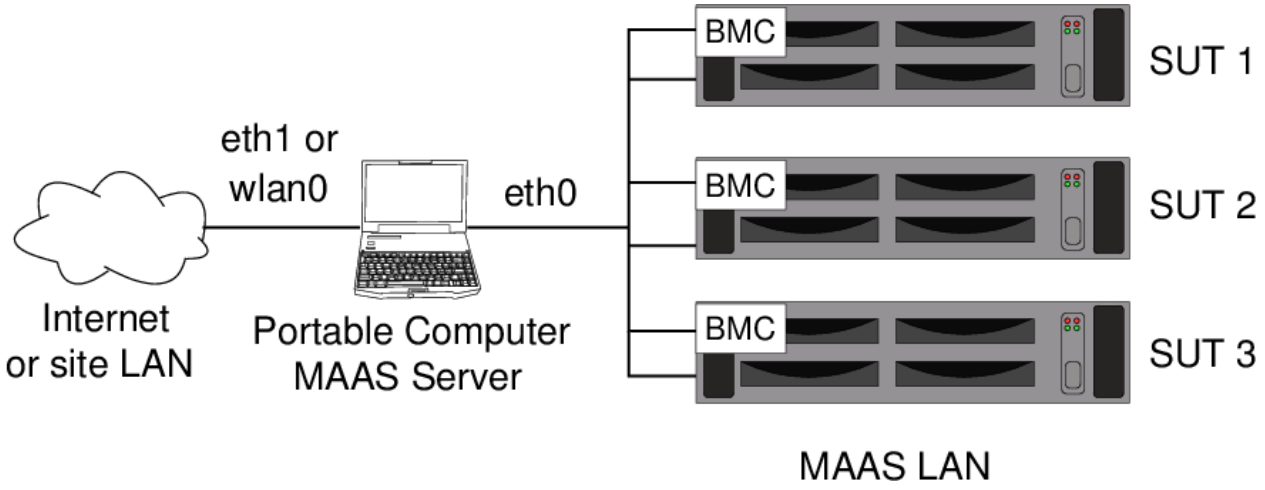


Figure 1: Network structure in which the portable computer will reside

WARNING: The configuration described in this document leaves several server programs running on the portable computer, including a proxy server, a web server (that can control MAAS), and an SSH server. Thus, it is unwise to expose the portable computer directly to the Internet. You should either secure it with strict local firewall rules or place it behind a strong firewall running on a router between it and the Internet.

Hardware Required

Before beginning, you should ensure that you have the following hardware:

- Portable computer
  - Ensure that the portable computer has two network interfaces. A laptop with both Ethernet and wi-fi should suffice; or you can use a USB network dongle to provide a second interface.
  - Because testing sessions can last for hours, ensure that you have a power brick; you should *not* run on battery power!
  - You can install on a virtual machine in a more general-purpose computer, but you'll have to pay careful attention to the network and disk settings.
- System Under Test (SUT) that provides one of the power control types MAAS supports:
  - American Power Conversion (APC) PDU
  - Cisco UCS Manager

    - Digital Loggers, Inc. PDU
    - Facebook's Wedge
    - HP Moonshot - iLO Chassis Manager
    - HP Moonshot - iLO (IPMI)
    - IBM Hardware Management Console (HMC)
    - IPMI
    - Intel AMT
    - Microsoft OCS - Chassis Manager
    - OpenStack Nova
    - Rack Scale Design
    - SeaMicro 15000
    - Sentry Switch CDU
    - VMWare
    - Virsh (virtual systems)
- Gigabit or faster switch (we recommend 8 ports minimum)
    - For laptop with Wi-Fi: one Ethernet cable
    - For NUC or laptop with dongle: two Ethernet cables
    - For each SUT: one Ethernet cable for each NIC port including the BMC
    - Please see the Self-Test Guide for further information on network requirements for certification testing.
- Monitor and keyboard for SUT (helpful, but not strictly required)
- Monitor, keyboard, and mouse for the MAAS system (a laptop's built-in devices should be sufficient)
- At least 1 TB of disk space with which to mirror the Ubuntu archives, if desired. (An external USB3 hard disk may be used for this, if necessary.)

Note that these hardware requirements are geared toward a typical testing environment. You may need to expand this list in some cases. For instance, if you test multiple servers simultaneously, you may need additional Ethernet ports and cables.

Installing and Configuring Ubuntu

Once you've assembled the basic hardware for your portable system, you can begin preparing it. The initial steps involve installing Ubuntu and setting up its most basic network settings:

1. Install Ubuntu 18.04 (Bionic Beaver) to the portable system.
    - The version of Ubuntu Server 18.04 described here can be obtained from https://www.ubuntu.com/download/server.
    - This guide assumes the use of Ubuntu Server 18.04 and MAAS 2.4. Although other versions of Ubuntu and MAAS may work, some details will differ. Some notable variants include:
        - If your MAAS server requires use of LVM or other exotic disk configurations, you may need to install using the older Debian-based installation medium, which you can obtain at http://cdimage.ubuntu.com/releases/18.04/release/, rather than by using the "live server" installation medium that's described here; however, some installation details differ from what's described in this document.
        - **When you boot the installation medium, you should select the "Install Ubuntu" option, not either of the "Install MAAS bare-metal cloud" options.** The procedure in this document involves installing MAAS later.
    - On the *Network connections* screen, configure your network ports:

- Configure your *external* network port:
  - If you need to use both a built-in Ethernet port and an Ethernet dongle, it's best to use the latter as your external port.
  - Use DHCP or a static IP address, as required by your environment.
  - If you use a static configuration, provide a gateway and DNS server, if possible.
  - In most cases, no explicit configuration of the external port is necessary because the Ubuntu Server installer will have set it up to use DHCP, which is appropriate. You can adjust it if necessary, though.
  - If you intend to use WiFi for your external network, you may be best served by installing NetworkManager, which NetPlan can use to manage the WiFi link. See https://djanotes.blogspot.com/2018/03/connecting-to-wifi-network-with-netplan.h for an example.
- Configure your *internal* network port:
  - If possible, configure the computer's built-in Ethernet port, rather than a plug-in dongle, as the internal port.
  - This guide assumes use of a static IP address of 172.24.124.1/22 on this port; however, you can use a different network address, if desired or necessary.
  - Using a /22 or wider network is advisable for the internal network, for reasons described in Appendix C: MAAS Network Ranges.
  - If your portable computer will move from one *external* network to another, be sure to consider all its likely *external* addresses when deciding on its *internal* address and netmask.
  - Avoid the 10.0.3.0/24 address range, because Ubuntu server uses this address range for its LXC container tool.
  - *Do not* set a gateway or DNS server on the *internal* network port.
- If you can't easily differentiate the two ports during installation, you can configure one or both of them after completing the Ubuntu installation. Note that Ubuntu 17.10 and later use NetPlan for network configuration; see https://wiki.ubuntu.com/Netplan/Design and https://netplan.io for details.

- The network configuration screen resembles Figure 2. In this example, `enp0s8` is the internal port and `enp0s3` is the external port.

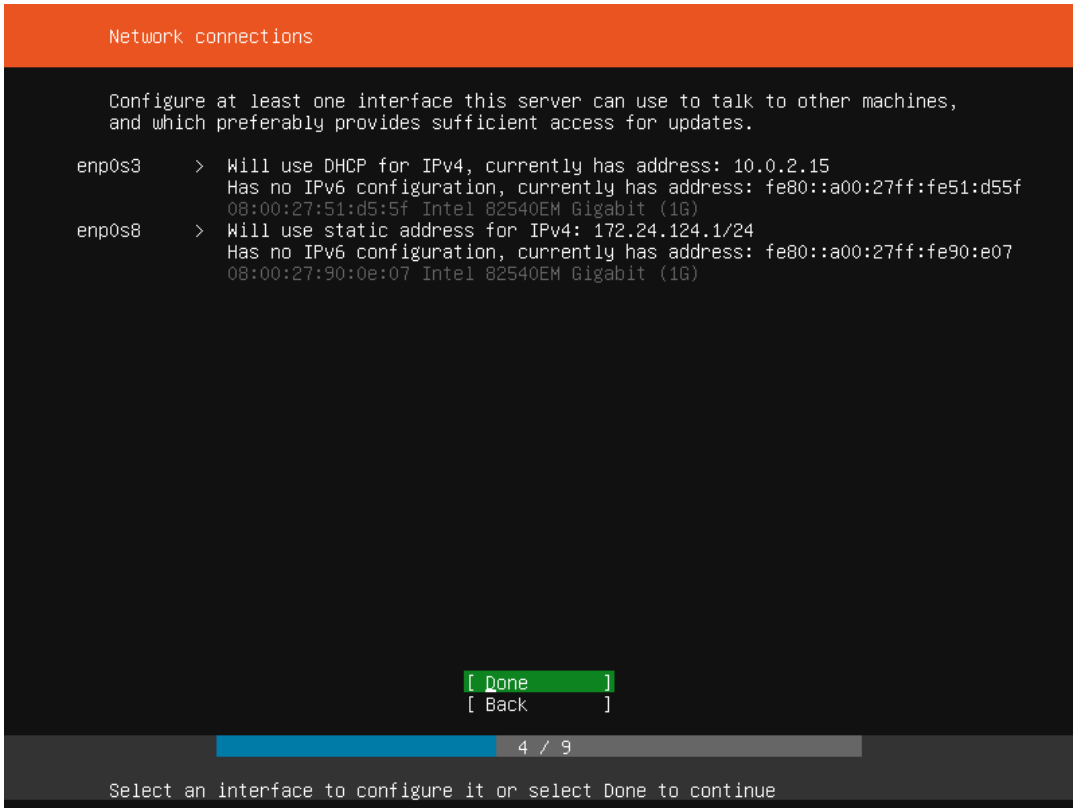


Figure 1: The network can be configured during installation.

- Configure the disk storage and other options as you see fit.
- If you plan to mirror the Ubuntu archives locally, ensure you have enough space in the `/srv` directory to hold your mirrors. As a general rule of thumb, you should set aside about 200 GiB per release. In most cases, a 1 TB disk dedicated to this task works well. If necessary, mount an extra disk at `/srv` to hold your repository mirror. (You can do this after installing Ubuntu, if you like.)

2. When the installation is complete, boot the portable computer and log in.
3. Type `ifconfig` to verify your network configuration. If either network port is not properly configured, edit the configuration file in `/etc/netplan/`. This file may be

called `50-cloud-init.yaml`, `01-netcfg.yaml`, or something else; the name depends on the installation method. A typical configuration should look like this, although likely with different network device names (`enp0s3` and `enp0s8` here) and possibly different IP addresses:

```
network:
  version: 2
  ethernets:
    enp0s8:
      addresses:
      - 172.24.124.1/22
      nameservers:
        search: [ ]
        addresses: [ ]
      optional: true
    enp0s3:
      addresses: [ ]
      dhcp4: true
      optional: true
```

If you need to change the network configuration, type `sudo netplan apply` or reboot the computer to apply the changes.

4. Update the software on your system to the latest versions available:

```
$ sudo apt update
$ sudo apt dist-upgrade
```

5. If desired, install X11 and your preferred desktop environment. This will enable you to use the portable computer itself to access the MAAS web UI. You can skip this step if your MAAS server will be accessed remotely. If in doubt, don't install X11 and a desktop environment. You can always install it later if you discover it's necessary. In most cases, you can install X11 and the desktop environment with a single command, such as the following to install Ubuntu 18.04's GNOME:

```
sudo apt install ubuntu-gnome-desktop
```

6. Reboot the computer. This enables you to begin using your updated kernel (if it was updated) and ensures that your network settings will survive a reboot.

Installing and Configuring MAAS

Installing MAAS on the computer is quite straightforward; you simply use APT. With MAAS installed, you can run the `maniacs-setup` script to configure MAAS for use in an Ubuntu certification environment.

Installing MAAS

Configuring MAAS is described in generic terms at http://maas.ubuntu.com/docs/install.html. The more specific procedure for using MAAS in certification testing is:

1. Several scripts and configuration files are available in the `maas-cert-server` package in the hardware certification PPA. You can install the scripts and configuration files as follows:

   ```
   $ sudo apt-add-repository ppa:hardware-certification/public
   $ sudo apt install maas-cert-server
   ```

   The `maas-cert-server` package includes a dependency on MAAS, so installing `maas-cert-server` will also install MAAS, as well as all of MAAS's dependencies.

2. Verify that you've installed MAAS 2.4.0-beta2 or later, rather than some earlier version:

   ```
   $ dpkg -s maas | grep Version
   ```

   If the wrong version is installed, fixing the problem (presumably an out-of-date mirror repository) and upgrading may work. If you upgrade from an earlier version of MAAS, be sure to select the option to upgrade all the configuration files when the package manager asks about this.

3. Edit the `/etc/maas-cert-server/config` file to be sure that the variables it contains are correct. Specifically:

- `INTERNAL_NET` must point to your *internal* network device (`eth0` in the below examples).
- `EXTERNAL_NET` must point to your *external* network device (`eth1` in the below examples).
- Do not adjust other values without consulting with the Server Certification Team.
- Note that there must *not* be spaces surrounding the equal signs (=) in the assignments!

4. Optionally create an `/etc/maas-cert-server/iperf.conf` file to identify your `iperf` server(s). This file should consist of a single line that contains a comma-delimited list of IP addresses, each identifying a different `iperf` (or `iperf3`) server. If this file is absent, SUTs will configure themselves to use their network gateways (normally the MAAS server) as the `iperf` target. If `/etc/maas-cert-server/iperf.conf` is present, though, MAAS will tell SUTs to use the specified system(s) instead. You might use this feature if your `iperf` server is not the SUTs' network gateway or if you have multiple `iperf` servers. The SUTs will attempt to use each `iperf` target in series until the network test passes or until the list is exhausted. This setting can be overridden on SUTs by editing the `/etc/xdg/canonical-certification.conf` file on the SUT. See Appendix B: Network Testing Options for more on advanced network testing configurations.

Running the Setup Script

The MAAS configuration script is called `maniacs-setup`, and was installed as part of the `maas-cert-server` package. Running this script will set up the MAAS server with reasonable defaults for certification work; however, the script will also ask you a few questions along the way:

```
$ sudo maniacs-setup

********************************************************************************
* Identified networks:
*   INTERNAL: 172.24.124.1 on eth0
*   EXTERNAL: 192.168.1.27 on eth1
```

```
*
* Is this correct (Y/n)?
```

Be sure your network assignments are correct at this point! If the script complains about a problem, such as an inability to identify an IP address or a default route being present on your internal network, go back and review both your network settings and the contents of your `/etc/maas-cert-server/config` file to identify the cause and correct the problem.

If you approve the settings, the script will display additional messages as it begins to configure the MAAS server. Some of these messages are the output of the programs it calls. For the most part this output can be ignored, but if a problem occurs, be sure to report it in detail, including the script's output.

Note that at all prompts for a "Y/N" response, the default value is capitalized; if you press Enter, that default will be used.

The next question acquires a password for the MAAS administrative account, which will have the same name as your default login name:

```
*******************************************************************************
* A MAAS administrative account with a name of ubuntu is being
* created.
*
* Please enter a password for this account:
* Please re-enter the password for verification:
```

In most cases, you should enable NAT on your MAAS server; however, if official policy at the site where the server will be used forbids the use of NAT, you may opt to leave it disabled:

```
*******************************************************************************
* NAT enables this computer to connect the nodes it controls to the Internet
* for direct downloads of package updates and to submit certification results
* to C3.
*
* You can configure this computer to automatically start NAT. The following
* commands can start or stop NAT:
*  sudo systemctl enable certification-nat -- Start NAT on the next reboot
```

```
*  sudo systemctl disable certification-nat -- Stop NAT on the next reboot
*  sudo service certification-nat start -- Start NAT until the next reboot
*  sudo service certification-nat stop -- Stop NAT until the next reboot
*
* Do you want to set up this computer to automatically enable NAT (Y/n)?
```

The `service certification-nat start` and `service certification-nat stop` commands run the `startnat.sh` and `flushnat.sh` scripts, respectively. Both of these scripts come with the `maas-cert-server` package. The `systemctl` commands set up or remove symbolic links in the systemd configuration directories to enable (or not) NAT when the computer boots. You can check whether NAT is running by typing `sudo iptables -L`, which should show a pair of `ACCEPT` rules in the `FORWARD` chain if NAT is enabled and no such rules if it's not running; and by typing `cat /proc/sys/net/ipv4/ip_forward`, which should return `1` if NAT is enabled and `0` if it's not enabled.

If your work site has poor Internet connectivity or forbids outgoing connections, you must create a local mirror of the Ubuntu archives on your MAAS server. These archives will be stored in the `/srv/mirrors/` directory, but creating them takes a long time because of the amount of data to be downloaded — about 200 GiB per release. For comparison, HD video consumes 1-8 GiB per hour — usually on the low end of that range for video streaming services. As should be clear, the result will be significant network demand that will degrade a typical residential DSL or cable connection for hours, and possibly exceed your monthly bandwidth allocation. The download will occur in the background, though, so you can continue with MAAS setup as the download proceeds. If you want to defer creating a mirror, you should respond `N` to the following prompt, then re-launch `maniacs-setup` with the `--mirror-archives` (or `-m`) option later. In any event, you make your selection at the following prompt:

```
******************************************************************************
* Mirroring an archive site is necessary if you'll be doing testing while
* disconnected from the Internet, and is desirable if your test site has
* poor Internet connectivity. Performing the mirroring operation takes
* time and disk space, though -- about 150 GiB per release mirrored.
* To defer this task, respond 'N' to the following question.
*
* Do you want to mirror an archive site for local use (y/N)? Y
```

If you opt to mirror the archive, the script will ask you to verify the upstream mirror site:

```
* Identified upstream archive is:
*  http://us.archive.ubuntu.com/ubuntu/
*
* Is this correct (Y/n)? y
```

If you respond `n` to this question, the script asks you to specify another archive site. The script then asks you which Ubuntu releases to mirror:

```
* Do you want to mirror trusty (Y/n)? y
* Do you want to mirror xenial (Y/n)? y
* Do you want to mirror artful (Y/n)? n
* Do you want to mirror bionic (Y/n)? y
* Do you want to mirror cosmic (Y/n)? n
```

The list of releases changes as new versions become available and as old ones drop out of supported status. When the mirror process is done, you'll be asked if you want to configure the computer to automatically update its mirror every day, by modifying the `/etc/cron.d/apt-mirror` file. If you do not opt for automatic daily updates, you can update your mirror at any time by typing `sudo apt-mirror`.

```
* Set up cron to keep your mirror up-to-date (Y/n)? y
* Cron should update your mirror every morning at 4 AM.
* You can adjust /etc/cron.d/apt-mirror manually, if you like.
```

Note that `maniacs-setup` configures the system to mirror AMD64, i386, and source repositories because all three are required by the default MAAS configuration. If you want to tweak the mirror configuration, you can do so by editing the `/etc/apt/mirror.list` file — but do so *after* finishing with the `maniacs-setup` script, and then type `sudo apt-mirror` to pull in any new directories you've specified. You can also configure the computer to use its own local mirror, if you like:

```
* Adjust this computer to use the local mirror (Y/n)? y
```

The script then gives you the option to retrieve images used for virtualization testing. If your site has good Internet connectivity, you may not need these images; but it's not a

bad idea to have them on hand just in case. Although downloading the cloud images isn't nearly as time-consuming as mirroring the archives, it can take long enough that you may want to defer this action. You can download the cloud images later by launching `maniacs-setup` with the `--download-virtualization-image` (or `-d`) option.

```
********************************************************************************
* An Ubuntu cloud image is required for virtualization tests. Having such
* an image on your MAAS server can be convenient, but downloading it can
* take a while (each image is about 250MiB). This process will import cloud
* images for whatever releases and architectures you specify. If you select
* 'Y', logs will be stored at $MCS_DATA/cloudimg-*-dl-*.log;
* monitor them if you suspect download problems.
*
* To defer this task, respond 'N' to the following question.
*
* Do you want to set up a local cloud image mirror for the virtualization
* tests (Y/n)?
```

If you respond `Y` to this question, the script proceeds to ask you what Ubuntu versions and architectures to download:

```
* Cloud Mirror does not exist. Creating.
* Do you want to get images for trusty release (y/N)? n
* Do you want to get images for xenial release (Y/n)? y
* Do you want to get images for artful release (y/N)? n
* Do you want to get images for bionic release (y/N)? y
* Do you want to get images for cosmic release (y/N)? n
*
* Do you want to get images for amd64 architecture (Y/n)? y
* Do you want to get images for i386 architecture (y/N)? n
* Do you want to get images for arm64 architecture (y/N)? n
* Do you want to get images for armhf architecture (y/N)? n
* Do you want to get images for ppc64el architecture (y/N)? n
* Do you want to get images for s390x architecture (y/N)? n
* Downloading cloud images. This may take some tiime.
```

```
*
* Downloading image for xenial on amd64 in the background....
```

These downloads proceed in the background, with logs stored in `~/.maas-cert-server`, so you can check there if you suspect problems. To monitor the downloads, use `top` or `ps` to look for instances of `wget`.

You can customize the site that MAAS tells nodes to use for their repositories. If you mirrored a repository, the script points nodes to itself (via its internal IP address); but if you did not mirror a repository, the script should point your nodes to the same site used by the MAAS server itself. In either case, you can press the Enter key to accept the default or enter a new value:

```
******************************************************************************
* MAAS tells nodes to look to an Ubuntu repository on the Internet. You
* can customize that site by entering it here, or leave this field blank
* to use the default value of http://172.24.124.1/ubuntu.
*
* Type your repository's URL, or press the Enter key:
```

At this point, the script gives you the option of telling MAAS to begin importing its boot resources — images it uses to enlist, commission, and deploy nodes. This process can take several minutes to over an hour to complete, so the script gives you the option of deferring this process:

```
******************************************************************************
* MAAS requires boot resource images to be useful, and they will now be
* imported in the background. This can take a LONG time, but will not
* significantly slow down subsequent configuration steps.
* Beginning import of boot resources
```

If the download of boot resources fails and you want to initiate it manually later, you can use the MAAS web UI or launch `maniacs-setup` with the `--import-boot-resources` (or `-i`) option.

Sometimes this process hangs. Typically, the boot images end up available in MAAS, but the script doesn't move on. If this happens, you can kill the script and, if desired, re-launch it to finish the installation.

After the download of boot resources is begun, the script configures personal package archives (PPAs), in which the latest certification software is stored. You'll want to configure PPAs for whatever architectures you intend to test:

```
************************************************************************
* Now we will set up the PPAs necessary for installing the certification
* tools when deploying the SUT.
*
* Do you want to set PPAs for amd64 architecture (Y/n)? Y
* Do you want to set PPAs for i386 architecture (y/N)? n
* Do you want to set PPAs for arm64 architecture (y/N)? n
* Do you want to set PPAs for armhf architecture (y/N)? n
* Do you want to set PPAs for ppc64el architecture (y/N)? n
*
* Adding PPAs for the following architectures: amd64
*
* Hardware Certification Stable PPA
* Firmware Test Suite Stable PPA
* Hardware Certification Development PPA (Disabled by default)
*
* PPA Setup Complete
```

Finally, the script announces it's finished its work:

```
************************************************************************
* The maniacs-setup script has finished!
```

In addition to setting the options for which it prompts, `maniacs-setup` adjusts some other details of which you should be aware:

- SSH keys are generated for your user account and added to the MAAS server. These keys enable you to log in to nodes that MAAS deploys from your regular account on the MAAS server.
- Any keys in your `~/.ssh/authorized_keys` file on the portable computer are also added to the MAAS setup. Again, this simplifies login.
- The portable computer's SSH server configuration is relaxed so that changed host keys do not block outgoing connections. This change is *insecure*, but is a practical

necessity because your internal network's nodes will be redeployed regularly. You should keep this setting in mind and minimize your use of this computer to SSH to external sites.

- MAAS is configured to tell nodes to install the Canonical Certification Suite whenever they're deployed. This detail increases deployment time compared to a generic MAAS installation.
- The default storage layout setting is changed from "LVM" to "flat." Some certification tests assume a flat layout, which is the default (and only) option in MAAS 1.8 and earlier.

Checking the MAAS Configuration

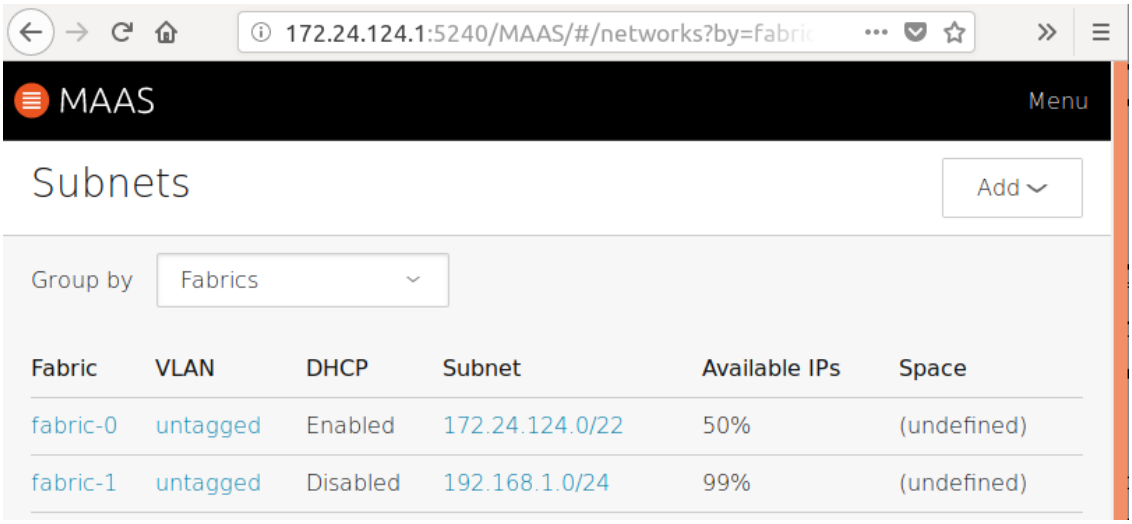
At this point, MAAS should be installed and configured; however, it's worth verifying the most important options in the MAAS web UI. You may also want to modify a few settings. To do so, follow these steps:

1. Verify you can access the MAAS web UI:
    - Launch a browser and point it to http://172.24.124.1:5240/MAAS (changing the IP address as necessary).
    - Note that MAAS 2.3 and earlier accepted connections on port 80, but starting with MAAS 2.4, you *must* use port 5240.
    - You should be able to access the server on either its internal or external network address, although at this point, the only computer on the internal network may be the portable computer itself.
    - If you provide the computer with a hostname in DNS or `/etc/hosts`, you should be able to access it via that name, as well.
    - You should see a login prompt.
2. Log in to the web UI using your regular username and the password you gave to the setup script.
3. Once you log in, MAAS presents an overview screen.
    1. Review these settings for sanity. Some show options that were set earlier in this process. Most others should be self-explanatory. The standard MAAS images are among the items shown. Be sure the Ubuntu versions and

architectures you need are checked, and click Save Selection to import anything you need.

2. When you're done, click Continue at the bottom of the page.
3. The next page shows SSH keys. You can add more keys, taken from Github, Launchpad, or copied-and-pasted from your local computer. When you're done adding SSH keys, click Go to Dashboard at the bottom of the page.

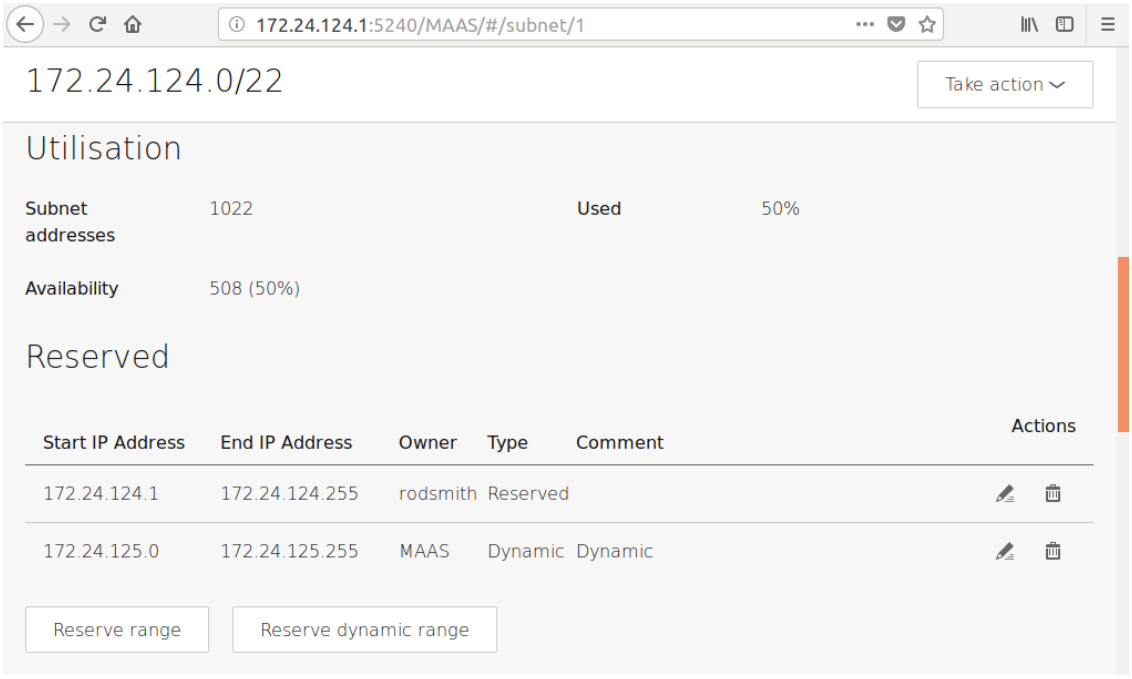
4. On the Network Discovery page, you may want to disable the Device Discovery feature; you can do this with the slider near the top right of the page. In theory, this feature should passively detect devices and should cause no problems. In practice, it may trigger security alerts on the external network.
5. Click the Subnets link near the top of the web page so you can review the DHCP options:
    1. Click the subnet range for the *internal* network (172.24.124.0/22 in this example):



    Your network, of course, may be different from this example, particularly if you have unused network devices, which will show up as additional "fabrics."

    2. On the page for your internal network, scroll down about halfway to view the Utilisation and Reserved sections. At this point, about half the addresses will be classified as "used" because `maniacs-setup` set them aside as reserved or as managed by DHCP. The "available" addresses are those that do not belong to either of these categories; MAAS assigns them to nodes that are deployed

using its standard settings. (See Appendix C: MAAS Network Ranges for details of how MAAS manages its network addresses.)



3. If the various ranges (reserved, dynamic, or the implicit available addresses) are not appropriate, you can edit them as follows:
   - Click the edit icon near the right side of the page in the row for the range you want to delete or modify. You can then change the start and end addresses, and then click Save to save your changes.
   - If you want to completely delete the range, click the trash can icon instead of the edit icon.
4. You can optionally reserve additional ranges for machines not managed by MAAS (using the Reserve Range button) or for DHCP addresses (using the Reserve Dynamic Range button).

6. Click Settings near the top of the page to load the MAAS Settings page, where you review several miscellaneous MAAS details. This page is broken into several subsections, each of which has a tab near the top of the page — Users, General, User Scripts, and so on. If you change any settings, be sure to click the associated “Save” button within that section.

Testing the MAAS Server

At this point, your MAAS server should be set up and configured correctly. To test it, follow these steps:

1. Prepare a computer by configuring it to boot via PXE. This computer need not be a computer you plan to certify; anything that can PXE-boot should work, although to fully test the MAAS server, the test system should provide IPMI or some other power-control tool that MAAS supports.
2. Connect the test computer to the portable computer's *internal* network and power it on.
    - The test computer should PXE-boot from the portable MAAS computer.
    - This first boot should be to the enlistment image, which provides some very basic information to the MAAS server.
    - Once the node powers itself off you should see it listed in the MAAS machines list (http://172.24.124.1:5240/MAAS/#/machines/) with a Status field of "New." If it doesn't appear, try refreshing the page. Also, be sure you check the *Machines* tab, not the *Devices* tab.
3. Click on the node's hostname to view the node's summary page.
4. If desired, click the node's hostname near the upper-left corner of the page. This will enable you to change the hostname to something descriptive, such as the computer's model number. Click "Save" when you've made your changes.
5. You may need to make a few changes in the Configuration area:
    - If necessary, click "Edit" in the Machine Configuration section to change the architecture of the machine. Click "Save Changes" when you're done.
    - For non-IPMI machines, you will most likely have to enter power control details by clicking Edit next to the Power Configuration heading. This may necessitate setting an IP address, MAC address, password, or other information, depending on the power control technology in use. Click "Save Changes" when you're done.
6. Click "Take Action" near the top-right corner of the page, followed by "Commission" from the resulting drop-down menu. You must then click "Commission machine."
    - The node should power on again. This time you'll see it PXE-boot the commissioning image. Note that if your test system lacks a BMC or other means to remotely control its power, you must manually power it on.

- The node should do a bit more work this time before powering off again.
- Once it's done, the UI will show a Status of "Ready."

7. Once the system powers off after commissioning, click "Take Action" followed by "Deploy." You must then click "Deploy machine" to confirm this action.
   - The node should power on again (or you will have to control it manually if it lacks a BMC). This time it will take longer to finish working, as MAAS will install Ubuntu and the certification suite on the system.
   - Once it's done, the computer will reboot into its installed image.
   - Log into the node from the MAAS server by using SSH, as in `ssh testnode` if you've given the node the name `testnode`.
   - In the node, type `canonical-certification-precheck`. The certification suite's precheck script should run to verify that the system is ready for testing. Don't be too concerned with the results; the point of this operation is to check that the certification suite was properly installed, and at this point, additional steps are needed for the precheck script to say the node's ready for testing. These details are described in the Self-Testing Guide, which is available from your MAAS server itself, such as http://172.24.124.1.

If any of these steps fail, you may have run into a MAAS bug; your test computer may have a buggy PXE, IPMI, or other subsystem; or you may have misconfigured something in the MAAS setup. You may want to review the preceding sections to verify that you configured everything correctly. To help in debugging problems, the node status page includes sections entitled Commissioning, Logs, and Events with various pieces of diagnostic information related to commissioning and deployment.

Appendix A: Adding i386 Support

By default, the `maniacs-setup` script supports only AMD64 (64-bit, x86-64) nodes. (If you created a local mirror, it includes i386/x86 binaries because they're needed by some 64-bit packages.) If you expect to run certification tests on i386 (32-bit, x86) computers, though, you must add support for such systems in MAAS:

1. In the MAAS web UI, click the Images tab.
2. Select "i386" in the "Architecture" column.
3. Click "Apply changes." The standard MAAS i386 images will download. This process can take several minutes, and perhaps over an hour on a slow Internet connection.

That's it. You can add support for ppc64el, ARM64, or other architectures in a similar way these architectures. Please consult the Server Certification Team if you need to certify systems using these CPUs.

Appendix B: Network Testing Options

A key part of certification is testing your SUT's network cards. This document is written with the assumption of a fairly basic configuration; however, some labs may have more advanced needs. Differences also exist between network configuration on Ubuntu 18.04 and earlier LTS releases. Important variables include:

- **Multiple simultaneous network tests** — A single server takes about 60 minutes per network port to run its network tests — long enough that testing multiple SUTs simultaneously is likely to result in contention for access to the `iperf3` server. This is especially true if SUTs have multiple network ports — a server with four ports will tie up an `iperf3` server for four hours. An `iperf3` server will refuse multiple connections, which should at least enable one SUT's network tests to pass; but if the `iperf3` server has a sufficiently fast NIC, it will then be under-utilized.
- **Advanced network interfaces** — A portable computer configured as described here will likely have a 1 Gbps link to the internal LAN. If you're testing systems with faster interfaces, you will need a separate computer to function as an `iperf3` server.
- **Network configuration methods** — Two network configuration tools have been used in recent versions of Ubuntu Server:
  - Versions through 17.04 used a system built around the `/etc/network/interfaces` file and the `ifup` and `ifdown` commands.
  - Versions starting with 17.10 use the new NetPlan system (https://netplan.io).

  Deploying a SUT via MAAS automatically configures its network ports — or at least, those ports that are configured via the MAAS web UI. Thus, you needn't be concerned with these differences on your SUTs. You may need to learn the new NetPlan tools for advanced configuration of your MAAS server, though.

If your `iperf3` target system has a fast NIC and want to test multiple slower SUTs, you can configure the fast NIC with multiple IP addresses. An `/etc/network/interfaces` entry for Ubuntu 17.04 or earlier to do this might look like this:

```
# The 10Gbps network interface
auto eno2
iface eno2 inet static
      address 172.24.124.2
      netmask 255.255.252.0
```

```
      broadcast 172.24.127.255
auto eno2:1
iface eno2:1 inet static
      address 172.24.124.3
      netmask 255.255.252.0
      broadcast 172.24.127.255
auto eno2:2
iface eno2:2 inet static
      address 172.24.124.4
      netmask 255.255.252.0
      broadcast 172.24.127.255
```

An equivalent NetPlan entry, for Ubuntu 17.10 or later, is simpler; you can specify multiple IP addresses, rather than just one, in `/etc/netplan/50-cloud-init.yaml` (or another file; the name varies depending on how the system was installed). For example:

```
eno2:
    addresses:
    - 172.24.124.2/22
    - 172.24.124.3/22
    - 172.24.124.4/22
```

Note that you do not explicitly set separate names for each interface, as in the `eno2:1` of the `/etc/network/interfaces` example.

In either case, you must activate the changes after making them. In theory, you can do this without rebooting by using commands such as `sudo ifup eno2:1` or `sudo netplan apply`; however, you may find it's necessary to reboot to reliably apply an advanced configuration like this one. With either network configuration system, you can verify the network settings with `ip addr show eno2` (changing the interface name as necessary):

```
$ ip addr show eno2
3: eno2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
   UNKNOWN group default qlen 1000
  link/ether 08:00:27:90:0e:07 brd ff:ff:ff:ff:ff:ff
  inet 172.24.124.2/22 brd 172.24.127.255 scope global eno2
```

```
    valid_lft forever preferred_lft forever
inet 172.24.124.3/22 brd 172.24.127.255 scope global secondary eno2
    valid_lft forever preferred_lft forever
inet 172.24.124.4/22 brd 172.24.127.255 scope global secondary eno2
    valid_lft forever preferred_lft forever
inet6 fe80::a00:27ff:fe90:e07/64 scope link
    valid_lft forever preferred_lft forever
```

This example shows `eno2` up with all three of its IP addresses. Note that the older `ifconfig` tool will show only the first IP address for any device configured via NetPlan.

You would then launch `iperf3` separately on each IP address:

```
iperf3 -sD -B 172.24.124.2
iperf3 -sD -B 172.24.124.3
iperf3 -sD -B 172.24.124.4
```

On the MAAS server, you can enter all of the `iperf3` target addresses in `/etc/maas-cert-server/iperf.conf`:

```
172.24.124.2,172.24.124.3,172.24.124.4
```

The result should be that each of your SUTs will detect an open port on the `iperf3` server and use it without conflict, up to the number of ports you've configured. Past a certain point, though, you may over-stress your CPU or NIC, which will result in failed network tests. You may need to discover the limit experimentally.

Furthermore, if you want to test a SUT with a NIC that meets the speed of the `iperf3` server's NIC, you'll have to ensure that the high-speed SUT is tested alone — additional simultaneous tests will degrade the performance of all the tests, causing them all to fail.

If the `iperf3` server has multiple interfaces of differing speeds, you may find that performance will match the *lowest-speed* interface. This is because the Linux kernel arbitrarily decides which NIC to use for handling network traffic when multiple NICs are linked to one network segment, so the kernel may use a low-speed NIC in preference to a high-speed NIC. Two solutions to this problem exist:

- You can disable the lower-speed NIC(s) (permanently or temporarily) and rely exclusively on the high-speed NIC(s), at least when performing high-speed tests.

- You can configure the high-speed and low-speed NICs to use different address ranges — for instance, 172.24.124.0/22 for the low-speed NICs and 172.24.128.0/22 for the high-speed NICs. This approach will require additional MAAS configuration not described here. To minimize DHCP hassles, it's best to keep the networks on separate physical switches or VLANs, too.

If your network has a single `iperf3` server with multiple physical interfaces, you can launch `iperf3` separately on each NIC, as just described; however, you may run into a variant of the problem with NICs of differing speed — the Linux kernel may try to communicate over just one NIC, causing a bottleneck and degraded performance for all tests. Using multiple network segments or bonding NICs together may work around this problem, at the cost of increased configuration complexity.

If your lab uses separate LANs for different network speeds, you can list IP addresses on separate LANs in `/etc/maas-cert-server/iperf.conf` on the MAAS server or in `/etc/xdg/canonical-certification.conf` on SUTs. The SUT will try each IP address in turn until a test passes or until all the addresses are exhausted.

If you want to test multiple SUTs but your network lacks a high-speed NIC or a system with multiple NICs, you can do so by splitting your SUTs into two equal-sized groups. On Group A, launch `iperf3` as a server, then run the certification suite on Group B, configuring these SUTs to point to Group A's `iperf3` servers. When that run is done, reverse their roles — run `iperf3` as a server on Group B and run the certification suite on Group A. You'll need to adjust the `/etc/xdg/canonical-certification.conf` file on each SUT to point it to its own matched server.

You may find the `iftop` utility helpful on the `iperf3` server system. This tool enables you to monitor network connections, which can help you to spot performance problems early.

Appendix C: MAAS Network Ranges

As noted earlier, in Installing and Configuring Ubuntu, a /22 or wider network on the internal port is desirable, because this provides more addresses that are assigned more flexibly than with smaller networks. Specifically, MAAS splits the internal network into three parts:

- A reserved space, from which you can assign addresses manually. The MAAS server itself should be in this space. You might also use this space for other permanent infrastructure on the network, such as switches or other necessary servers. If you assign static IP addresses to your BMCs, their addresses would either come out of this space or be on another network block entirely.
- A DHCP space, which MAAS manages so that it can temporarily address servers when enlisting and commissioning them. Depending on your needs, your BMCs and even deployed nodes may be assigned via DHCP, too.
- An automatic space, which is a range of addresses that MAAS assigns to servers once they've been deployed in the default manner. (You can reconfigure nodes to use DHCP once deployed, if you prefer.)

In the MAAS subnet configuration page, the reserved and DHCP spaces are explicitly defined. Any address that does not fall into either of those spaces is part of the automatic space.

The following table shows how the `maniacs-setup` script described in this document splits up a /22, a /23, and a /24 network, starting with 172.24.124.1, between these three purposes. You can adjust the ranges after they've been set up by using the MAAS web UI, as described earlier, in Checking the MAAS Configuration, should the need arise. If you use a network block starting at something other than 172.24.124.1, the exact IP addresses shown in the table will be adjusted appropriately.

Purpose	*/22 network*	*/23 network*	*/24 network*
Reserved	172.24.124.1 - 172.24.124.255	172.24.124.1 - 172.24.124.50	172.24.124.1 - 172.24.124.9
Assigned via DHCP	172.24.125.0 - 172.24.125.255	172.24.124.51 - 172.24.124.255	172.24.124.10 - 172.24.124.127
Assigned Automatically	172.24.126.0 - 172.24.127.254	172.24.125.0 - 172.24.125.254	172.24.124.128 - 172.24.124.254

Appendix D: Glossary

The following definitions apply to terms used in this document.

1 Gbps

1 Gigabit - Network speed for Gigabit Ethernet (1000 Mbps).

10 Gbps

10 Gigabit - Network speed for 10 Gigabit Ethernet (10,000 Mbps).

BMC

Baseboard Management Controller — A device in many server models that allows remote in- and out-of-band management of hardware.

DHCP

Dynamic Host Control Protocol — method for providing IP addresses to the SUTs.

IPMI

Intelligent Platform Management Interface — A technology for remotely connecting to a system to perform management functions.

LAN

Local Area Network — the network to which your SUTs are connected. The LAN does not need to be Internet accessible (though that is preferable if possible).

MAAS

Metal as a Service — a Canonical product for provisioning systems quickly and easily.

NIC

Network Interface Card — the network device(s).

NUC

A small form-factor PC product from Intel.

PXE

Pre-boot Execution Environment — A technology that allows you to boot a system using remote images for easy deployment or network-based installation.

SUT

System Under Test — The machine you are testing for certification.