CANONICAL

MAAS 2 Advanced NUC Installation and Configuration — Scripted

(The MANIACS Document)

www.canonical.com

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 update the fixed-point-release images, how to add support for i386 (32-bit) images, and how to mirror repositories not configured for mirroring by the main setup procedure.

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.

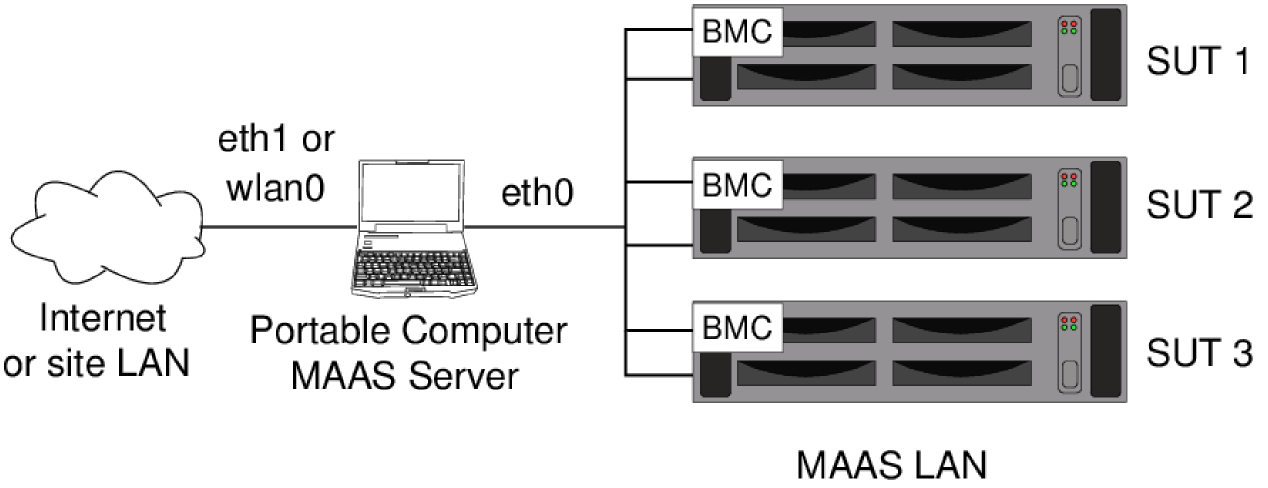


Figure 1: Network structure in which the portable computer will reside

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.

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 16.04 (Xenial Xerus) to the portable system
   - The Desktop version of Ubuntu is recommended because it enables you to easily access the MAAS web UI locally without needing a third system.
   - If you choose to use the Server version, you will probably want to install the X server and a desktop environment on top of that as it simplifies MAAS access.
   - This guide assumes the use of Ubuntu 16.04 and MAAS 2.2. Although other versions of Ubuntu and MAAS may work, some details will differ. Note that MAAS 2.0 or later is required for certification.
2. Boot the portable computer and log in.
3. Configure your *external* network port:
   - On a laptop, you can use the Wi-Fi (usually `wlan0`) port as the external 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.
   - In most cases, no explicit configuration of the external port is necessary because the Ubuntu Desktop system will have set it up to use DHCP, which is appropriate. You can adjust it if necessary, though.
   - You can configure the external network either via `/etc/network/interfaces` or by using the Desktop's Network Manager tool.
4. Configure your portable computer's *internal* port.
   - This guide assumes use of `eth0` and a static IP address of 172.16.0.1/22 on this port; however, your actual port name is likely to be `enp0s1`, `p4p1`, or something else.
   - If possible, configure the computer's built-in Ethernet port, rather than a plug-in dongle, as the internal port.
   - You can either edit `/etc/network/interfaces` or use the GUI Network tool in the System Settings panel to configure the internal port. An example `/etc/network/interfaces` configuration resembles the following:

```
auto eth0
iface eth0 inet static
    address 172.16.0.1
    broadcast 172.16.3.255
    netmask 255.255.252.0
```

- If necessary or desired, you may use a different IP address on the *internal* port. (Be sure to use an address within the MAAS server's reserved range, as described shortly!) 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 16.04 server uses this address range for its LXC container tool.
- Do not specify a gateway for the private internal LAN; doing so will create confusion when trying to access the Internet via the external port.
- If you have issues installing packages, check `route -n` and make sure you don't have a gateway route to the private LAN.
- Using a /22 or wider network is advisable for the internal network, for reasons described in Appendix E: MAAS Network Ranges.
- Once you've finished configuring this network port, be sure to activate it. If you configured it by editing `/etc/network/interfaces`, type `sudo ifup eth0` to activate it. (Depending on your starting configuration, you might need to type `sudo ifdown eth0` or bring it down via your GUI tools before bringing it up with its changed configuration.)

5. 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 150 GiB per release. If necessary, mount an extra disk at `/srv` to hold your repository mirror.
6. Update the software on your system to the latest versions available:

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

7. 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-get update
   $ sudo apt-get 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.

   Most of the `maas-cert-server` files will be installed in subdirectories of `/usr/share/maas-cert-server`, although a few appear outside of that directory tree. (Subsequent steps describe how to use these files.)

2. Verify that you've installed MAAS 2.1.2 or later, rather than some earlier version:

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

   If the wrong version is installed, fixing the problem (presumably a misconfigured PPA) 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 this document).
    - `EXTERNAL_NET` must point to your *external* network device (`eth1` in this document, but this is likely to be some other value for you).
    - 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 D: 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: 172.16.0.1 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 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. If you do so, you
* can disable it temporarily by using the 'flushnat.sh' script or permanently
* by removing the reference to /usr/sbin/startnat.sh from /etc/rc.local.
*
* Do you want to set up this computer to automatically enable NAT (Y/n)?
```

Note that you can enable NAT on a one-time basis by running the `startnat.sh` script and disable it by running the `flushnat.sh` script. Both of these scripts come with the `maas-cert-server` package.

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` directory, but creating them takes a long time because of the amount of data to be downloaded — about 150 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. 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 precise (Y/n)? n
* Do you want to mirror trusty (Y/n)? y
* Do you want to mirror xenial (Y/n)? y
* Do you want to mirror yakkety (Y/n)? n
* Do you want to mirror zesty (Y/n)? y
```

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 APT configuration. If you want to tweak the mirror configuration, you can do so by editing the `/etc/apt/mirror.list` file, as described in Appendix C: Mirroring Additional Repositories — 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 a 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.
*
* 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 yakkety release (y/N)? n
* Do you want to get images for zesty release (y/N)? y
* Do you want to get images for artful 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....
```

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://172.16.0.1/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 start 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; however, importing them
* can take a LONG time. You can perform this task now or defer it until
* later (or do it manually with the MAAS web UI).
*
* Do you want to import boot resources now? (Y/n)
```

If you choose to defer this process, MAAS may begin it automatically in the background. If this 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 with the `--update-point-releases` (or `-u`) option to finish the installation.

Certification may be done using either the default MAAS images or custom point-release images. In the past, the latter method was preferred; but the default MAAS images are now the preferred method. If you want to have the custom images available, `maniacs-setup` will help you import them:

```
***************************************************************************
* Ubuntu hardware certification is done using point-release images. These
* can take a LONG time to download. You can do so now or defer this task.
*
* Do you want to import point-release images now (Y/n)? y
```

```
*
* Do you want to import 17.04 (1 image) (y/N)? y
* Do you want to import 16.10 (1 image) (y/N)? n
* Do you want to import the 16.04 series (3 images) (Y/n)? y
* Do you want to import the 14.04 series (6 images) (Y/n)? y
```

Whenever you respond `Y` to a question about a particular version or series, the script proceeds to download and register the images. (The relevant output has been omitted from the preceding example.) If an image is already installed, `maniacs-setup` skips that image. Certification uses only LTS images; however, non-LTS images, such as 17.04, may be made available for testing and as a way to "preview" the features of future LTS series. `maniacs-setup` registers the most recent point-release image in any series you download as the default OS for deployments.

Again, this process can take a while. If you want to skip this step for now and return to it, you can; you should re-launch `maniacs-setup` with its `--update-point-releases` (or `-u`) option when you're ready to download these images.

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.

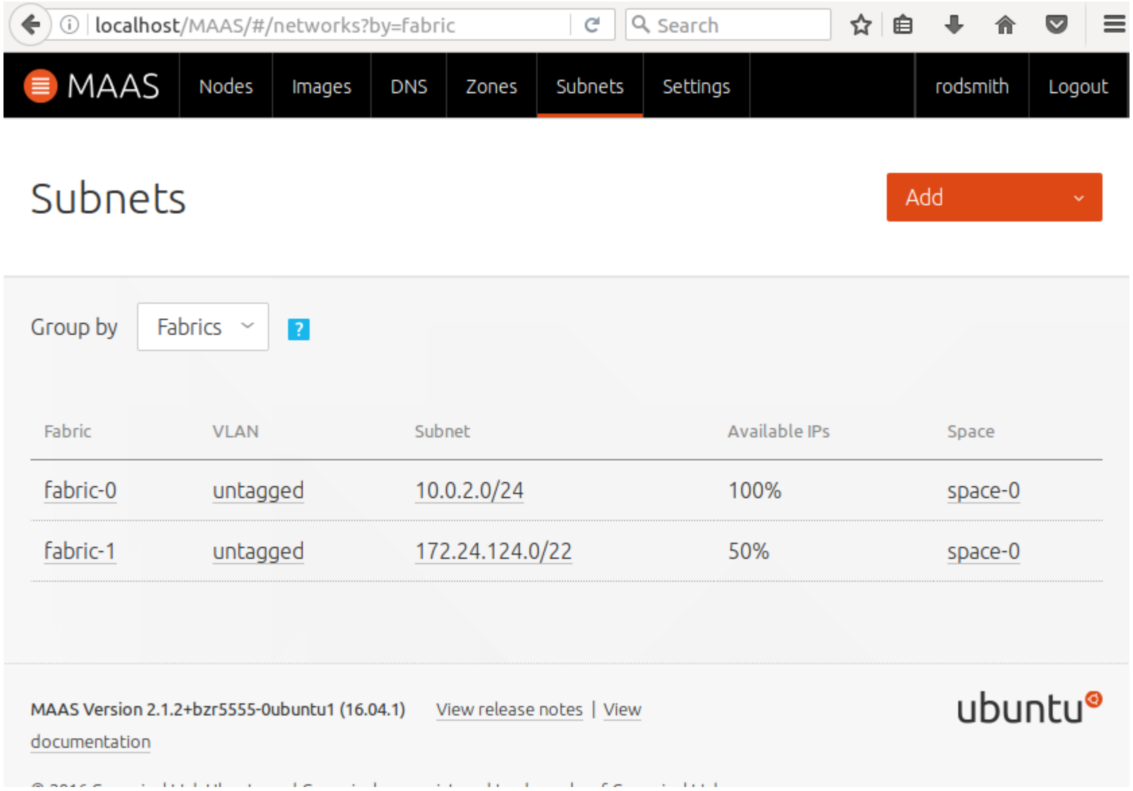
- MAAS is configured to tell nodes to install the Canonical Certification Suite whenever they're deployed. This detail increases deployment time compared to a generic MAAS installation.
- The default storage layout setting is changed from "LVM" to "flat." Some certification tests assume a flat layout, 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://172.16.0.1/MAAS (your internal port).
   - You should also be able to access this by default on the external port.
   - 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 (accessible later at `http://localhost/MAAS/#/intro`, or equivalent on the appropriate address).
   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. If you want to certify systems using these images rather than the custom images, 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 Dashboard page (aka Getting Started), you may optionally want to disable the Device Discovery feature. In theory, this feature should passively detect devices and should cause no problems. In practice, it may detect devices on your external network interface that shouldn't interact with MAAS.
5. Click Images near the top of the MAAS web page. This page will show the Ubuntu images that are available on the server. The setup script imports a 16.04 image for AMD64, as well as whatever custom point-release images you specified. You may need to take additional actions in some cases:
   - If you see a blue circle next to an image, it did not import correctly, or it is still importing.
   - If you need to support an architecture other than AMD64, you must check that architecture and click Apply Changes. This process will probably take several minutes to complete.
   - If you need to import i386 point-release images, you must import them manually, as described in Appendix B.
6. 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):



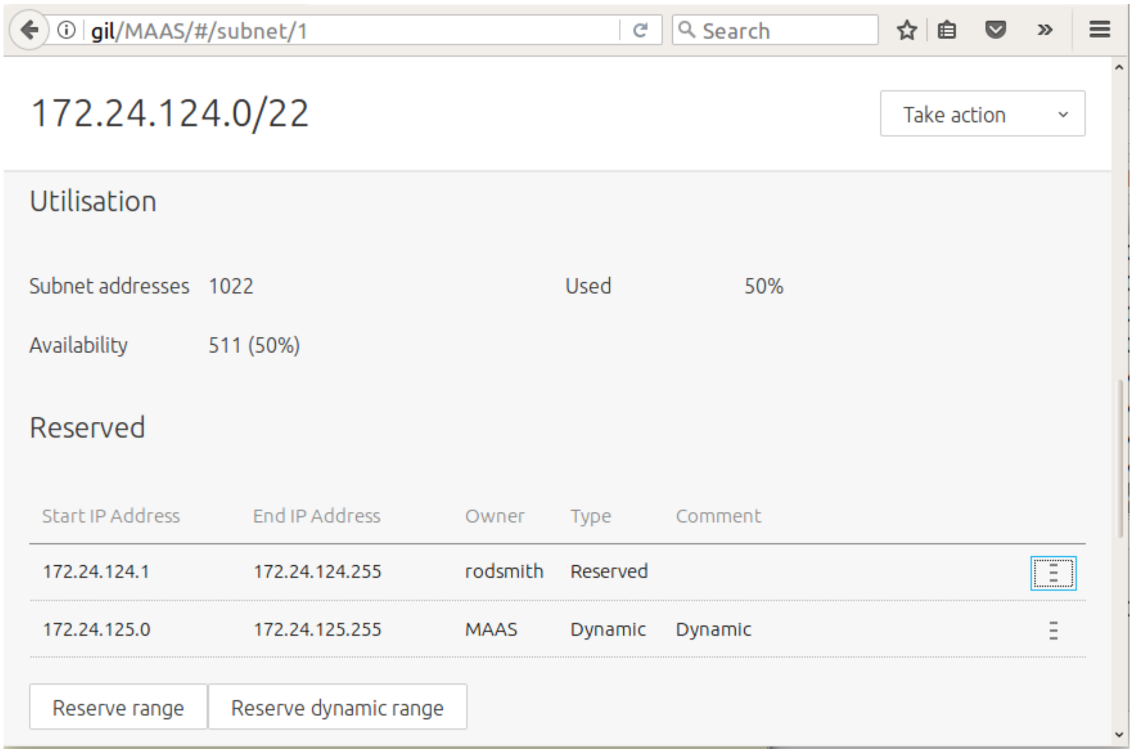
Subnets

Fabric	VLAN	Subnet	Available IPs	Space
fabric-0	untagged	10.0.2.0/24	100%	space-0
fabric-1	untagged	172.24.124.0/22	50%	space-0

MAAS Version 2.1.2+bzr5555-0ubuntu1 (16.04.1) View release notes | View documentation

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 E: MAAS Network Ranges for details of how MAAS manages its network addresses.)



172.24.124.0/22

Utilisation

Subnet addresses 1022 Used 50%

Availability 511 (50%)

Reserved

Start IP Address	End IP Address	Owner	Type	Comment
172.24.124.1	172.24.124.255	rodsmith	Reserved	
172.24.125.0	172.24.125.255	MAAS	Dynamic	Dynamic

3. If the various ranges (reserved, dynamic, or the implicit available addresses) are not appropriate, you can edit them as follows:
    1. Click the three horizontal lines on the right side of the screen in the row for the range you want to delete or modify.
    2. If you want to completely delete the range, click Remove in the resulting pop-up menu; otherwise, click Edit.
    3. If you click Edit, you can change the start and end addresses, then click Save to save your changes.
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).

7. Click Settings near the top of the page to load the MAAS Settings page, where you review several miscellaneous MAAS details. If you change any settings, be sure to click the associated “Save” button within that section. Unfortunately, each save button is section-specific and won’t save changes made in other sections of that page.

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 nodes list (http://localhost/MAAS/#/nodes/) with a Status field of "New." If it doesn't appear, try refreshing the page.
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. If necessary, click "Edit" in the Machine Summary section to change the architecture of the machine. Click "Save Changes" when you're done.
6. If necessary, click Power to open the Power tab to change the Power Type. (You must click Edit to make any actual changes.) 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.

7. Click “Take Action” near the top-right corner of the page, followed by “Commission Node” from the resulting drop-down menu. You must then click “Go.”
   - 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.”
8. Once the system powers off after commissioning, click “Take Action” followed by “Deploy.” You must then click “Go” 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-server`. The certification suite software should run. You can type Ctrl+C to exit; at this point, it’s sufficient to know that it installed correctly.

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 a section entitled Latest Node Events with a summary of the last few events related to the node. (You may have to refresh the page to see new events.)

Appendix A: Updating Fixed Point Release Images

From time to time, Canonical updates the LTS versions of Ubuntu with new point releases, such as 16.04.1, 16.04.2, and so on. Because you must run the certification tests on the latest point release, updating your MAAS server with the latest point releases will become necessary, sooner or later. This task can be accomplished by installing custom images.

The `maniacs-setup` script automatically downloads and installs all the available point-release images at the time you first run it. After a new point release is made, you can re-run the script with the `--update-point-releases` (or `-u`) option to have the script install the new release:

```
$ sudo apt-get install maas-cert-server
$ sudo maniacs-setup --update-point-releases
```

The script will skip most of the setup steps and proceed to asking you which point-release images to download and install. When you select a series, only those point-release images that have not yet been installed from that series will be downloaded.

If a particular point release is giving you problems, you can delete it using the MAAS web UI and then update your point-release images as just described; this will refresh the image to the latest available version. Similarly, if the Server Certification Team releases new images, you should first delete the old ones using the web UI and then update your point releases.

Note that before you can install a custom image for any given architecture, you must have first imported at least one image for that architecture via the conventional means. The `maniacs-setup` script detects whether you've installed the standard i386 images and will install i386 point-release images if and only if the standard i386 images are already installed. Currently, only AMD64 and i386 point-release images are available.

Appendix B: 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.
4. Re-run the `maniacs-setup` script, but add the `--update-point-releases` option to the script, as described in Appendix A. This will cause the script to download and add the point-release images for i386 systems.

That's it. You can add support for ppc64el, ARM64, or other architectures in a similar way; however, there are currently no point-release images for these architectures. Please consult the Server Certification Team if you need to certify systems using these CPUs.

Appendix C: Mirroring Additional Repositories

You can mirror repositories or Ubuntu versions beyond those configured by `maniacs-setup` by editing the `/etc/apt/mirror.list` file. This feature may be helpful to mirror a PPA you need locally or to mirror a pre-release version of Ubuntu. In most cases, the easiest way to do this is to cut-and-paste a similar block of configuration lines and modify them to suit your needs. The source may be lines in the original `/etc/apt/mirror.list` file or an example in the documentation for whatever site you want to mirror.

As an example, consider setting up a mirror of a pre-release version of Ubuntu. In the weeks leading up to the release of Ubuntu 16.04 (Xenial Xerus), the `maniacs-setup` script would not offer to mirror this archive, so if you wanted to mirror it, you had to copy and then modify the configuration for a working version, such as 14.04 (Trusty Tahr). (Note that this should *not* be necessary any longer; however, the process is covered here because it's one of the more complex types of mirrors you might want to manually add.) The `/etc/apt/mirror.list` file created by `maniacs-setup` includes three blocks for Trusty Tahr, each of which looks something like this:

```
## trusty on amd64 archives

deb-amd64 http://archive.ubuntu.com/ubuntu/ trusty main restricted universe \
          multiverse
deb-amd64 http://archive.ubuntu.com/ubuntu/ trusty-security main restricted \
          universe multiverse
deb-amd64 http://archive.ubuntu.com/ubuntu/ trusty-backports main restricted \
          universe multiverse
deb-amd64 http://archive.ubuntu.com/ubuntu/ trusty-updates main restricted \
          universe multiverse

deb-amd64 http://ppa.launchpad.net/hardware-certification/public/ubuntu trusty \
          main
deb-amd64 http://ppa.launchpad.net/firmware-testing-team/ppa-fwts-stable/ubuntu \
          trusty main
```

Two other blocks have lines beginning with `deb-i386` and `deb-src`, respectively, but are otherwise similar. Your file may refer to an archive site other than `archive.ubuntu.com`, as well. Also, this example wraps long lines to fit on the page, but you should not wrap long lines in this way.

Once you've found these blocks, duplicate *all three of them* and then modify them to suit your needs — in this example, you should change `trusty` on each line to `xenial`. If you're mirroring from something other than an official Ubuntu repository (`archive.ubuntu.com` or a regional `*.archive.ubuntu.com` site), you may need to change to an official Canonical repository to mirror pre-release software. If in doubt, check your preferred source with a web browser to see if the pre-release version is available on it. The result, for the block shown earlier, looks something like this:

```
## xenial on amd64 archives

deb-amd64 http://archive.ubuntu.com/ubuntu/ xenial main restricted universe \
          multiverse
deb-amd64 http://archive.ubuntu.com/ubuntu/ xenial-security main restricted \
          universe multiverse
deb-amd64 http://archive.ubuntu.com/ubuntu/ xenial-backports main restricted \
          universe multiverse
deb-amd64 http://archive.ubuntu.com/ubuntu/ xenial-updates main restricted \
          universe multiverse

deb-amd64 http://ppa.launchpad.net/hardware-certification/public/ubuntu xenial \
          main
deb-amd64 http://ppa.launchpad.net/firmware-testing-team/ppa-fwts-stable/ubuntu \
          xenial main
```

If you're mirroring something other than an Ubuntu release, you may be able to get by with mirroring just one architecture — or perhaps you want to mirror an architecture other than those described in this guide. Feel free to experiment, but remember that some archives are quite large, so you may end up consuming considerable network bandwidth and disk space creating your mirror.

Another key point to remember when modifying the `/etc/apt/mirror.list` file created by `maniacs-setup` is that re-running that script may overwrite your changes. Thus, you should back up your customized `/etc/apt/mirror.list` file so that you can cut-and-paste any changes you make back into the file that `maniacs-setup` regenerates, should the need arise.

Appendix D: 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 Ubuntu 14.04 and 16.04 testing. Important variables include:

- **Network test software** — The certification suite for Ubuntu 14.04 relies on `iperf` (version 2), but this has changed to `iperf3` for Ubuntu 16.04. Thus, you may need to be prepared to run both programs.
- **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 `iperf` (2) or `iperf3` server. This is especially true if SUTs have multiple network ports — a server with four ports will tie up an `iperf` server for four hours. An `iperf` 2 server will permit multiple connections, which will result in failed tests if the server's network hardware is not fast enough to handle the connections. 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 `iperf` or `iperf3` server.

If you have a fast (10 Gbps or faster) NIC and want to test multiple slower (say, 1 Gbps) SUTs, you can configure the fast NIC with multiple IP addresses. An `/etc/network/interfaces` entry to do this might look like this:

```
# The 10Gbps network interface
auto eth2
iface eth2 inet static
      address 172.16.0.2
      netmask 255.255.252.0
      broadcast 172.16.3.255
auto eth2:1
iface eth2:1 inet static
      address 172.16.0.3
      netmask 255.255.252.0
      broadcast 172.16.3.255
auto eth2:2
iface eth2:2 inet static
      address 172.16.0.4
      netmask 255.255.252.0
      broadcast 172.16.3.255
```

This example configures the `iperf3` server with three addresses. You can enter them all in `/etc/maas-cert-server/iperf.conf`:

```
172.16.0.2,172.16.0.3,172.16.0.4
```

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

```
iperf3 -sD -B 172.16.0.2
iperf3 -sD -B 172.16.0.3
iperf3 -sD -B 172.16.0.4
```

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, 172.16.0.0/22 for the low-speed NICs and 172.16.4.0/22 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` or on SUTs in `/etc/xdg/canonical-certification.conf`. 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. 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 E: 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, as explained later. Depending on your needs, your BMCs and even deployed nodes may be assigned via DHCP, too.
- 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.)

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 172.16.0.1, between these three purposes. You can adjust the ranges after they've been set up by using the MAAS web UI, should the need arise. If you use a network block starting at something other than 172.16.0.1, the exact IP addresses shown in the table will be adjusted appropriately.

Purpose	*/22 network*	*/23 network*	*/24 network*
Reserved	172.16.0.1 - 172.16.0.255	172.16.0.1 - 172.16.0.50	172.16.0.1 - 172.16.0.9
DHCP	172.16.1.0 - 172.16.1.255	172.16.0.51 - 172.16.0.255	172.16.0.10 - 172.16.0.127
Assigned Automatically	172.16.2.0 - 172.16.3.254	172.16.1.0 - 172.16.1.254	172.16.0.128 - 172.16.0.254

Appendix F: 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.