Maas deployment system configuration

PW Owned by Paul Wilson

Oct 27, 2022 • 4 min read • 3 people viewed

Metal As A Service (MAAS) Deployment configuration

MAAS is a bare-metal deployment system that allows JUJU to deploy .yml markup files to bare-metal spinning up clouds and software in clouds through automation instead of manual configuration. MAAS is also a bare-metal configuration and enrollment system. you can perform tests, burn-in, and verification of hardware resources pre-deployment.

Prepping for MAAS

You will need to generate and save SSH keys for MAAS. these will be used not just to access MAAS but additionally to access deployed hosts from MAAS.

Requirements for the production environment are as follows:

System	Memory	CPU (GHZ)	Disk (GB
Region controller (minus PostgreSQL)	2048	2.0	5
PostgreSQL	2048	2.0	20
Rack controller	2048	2.0	20

So, based on the above, the approximate requirements for this scenario are:

1. A region controller (including PostgreSQL) installed on one host, with 4.5 GB memory, 4.5 GHz CPU, and 45 GB of disk space.
2. A duplicate region controller (including PostgreSQL) on a second host, also with 4.5 GB memory, 4.5 GHz CPU, and 45 GB of disk space.
3. A rack controller installed on a third host, with 2.5 GB memory, 2.5 GHz CPU, and 40 GB of disk space.
4. A duplicate rack controller on a fourth host, also with 2.5 GB memory, 2.5 GHz CPU, and 40 GB of disk space.

Network configuration

Setup MAAS with bonded links, this way you can connect it to many different VLANs that it will need to access for PXE and deployments of OpenStack.

First, install ifupdown

```
sudo apt -y install ifupdown
```

Copy

then modify the /etc/networks/interfaces file to allow bonds and the bond connections

Interfaces Example:

```
# interfaces(5) file used by ifup(8) and ifdown(8)
# Include files from /etc/network/interfaces.d:
#source-directory /etc/network/interfaces.d

#network bond
auto bond0
 iface bond0 inet manual
 bond-slaves none
 bond-mode 6
 bond-miimon 200
up ip link set dev $IFACE up
```

```
12  down ip link set dev $IFACE down
13
14  #bond1
15  auto %INTERFACE0%
16  iface %INTERFACE0% inet manual
17  bond-master bond0
18  bond-primary eno1
19
20  #bond2
21  auto %INTERFACE1%
22  iface %INTERFACE1% inet manual
23  bond-master bond0
24
25  #Add any additional interfaces to the bond above
26
27  #vlan 5 maint
28  #Subnet 5.0.5.0/24
29  auto bond0.12
30  iface bond0.12 inet manual
31  up ip link set dev $IFACE up
32  down ip link set dev $IFACE down
33
34  #vlan 6 PXE
35  #Subnet 10.0.6.0/24
36  auto bond0.6
37  iface bond0.6 inet static
38  address %STATIC-IP%
39  netmask 255.255.255.0
40  gateway 10.0.6.1
41   dns-nameservers 8.8.8.8 %SELF%
42
43  #vlan 7 MGMT
44  #Subnet 10.0.7.0/24
45  auto bond0.7
46  iface bond0.7 inet manual
47  up ip link set dev $IFACE up
48  down ip link set dev $IFACE down
49
50  #vlan 8 Prov
51  #Subnet 10.0.8.0/24
```

```
auto bond0.8
iface bond0.8 inet manual
up ip link set dev $IFACE up
down ip link set dev $IFACE down
```

Copy

then configure Ifenslave:

```
sudo apt install ifenslave
sudo modprobe bonding
```

Copy

add bonding to etc/modules

```
sudo nano /etc/modules
Append "bonding"
```

Copy

Enable Ifupdown and /etc/network/interfaces config

```
sudo ifdown --force eno1 lo && ifup -a
sudo systemctl unmask networking
sudo systemctl enable networking
sudo systemctl restart networking
```

Copy

Remove Netplan.io

```
sudo apt -y purge netplan.io
```

Copy

Configure DNS in resolved.conf

```
sudo nano /etc/systemd/resolved.conf
```

Copy

Add the DNS search zone and DNS servers

```
DNS=%STATIC-IP% 8.8.8.8 (if you want for google)
```

Copy

Setup Postgres

Postgres clustering is simplified in Juju deployment, for this model we will install Postgres on MAAS initially and convert it to HA Postgres after the JUJU controllers are bootstrapped.

Initial installation of Postgres:

```
sudo apt update -y
sudo apt install -y postgresql
```

Copy

Make the Postgres user:

```
sudo -u postgres psql -c "CREATE USER \"$MAAS_DBUSER\" WITH ENCRYPT
```

Copy

Create the DB:

```
sudo -u postgres createdb -O "$MAAS_DBUSER" "$MAAS_DBNAME"
```

Copy

you will need to edit the config file located at: /etc/postgresql/10/main/pg_hba.conf

Add the following:

```
host    $MAAS_DBNAME    $MAAS_DBUSER    0/0     md5
```

Copy

Terminology

Region: Used to create a new region controller

Rack: Used to locate the region controller

Installing MAAS

Check for the proper version of MAAS this is important when rolling another MAAS server later for HA cluster Redundancy

check available versions:

```
sudo snap info maas
```

Copy

Once you determine the required version change the version number in the below command to the one you want. or leave the --channel part off to install latest.

Install MAAS with required version:

```
sudo snap install --channel=3.1/stable maas
```

Copy

Setting up MAAS after MAAS and Postgre and configured:

```
sudo maas init region+rack --database-uri "postgres://$MAAS_DBUSER:
```

Copy

MAAS will give a URL for access:

```
http://${API_HOST}:5240/MAAS
```

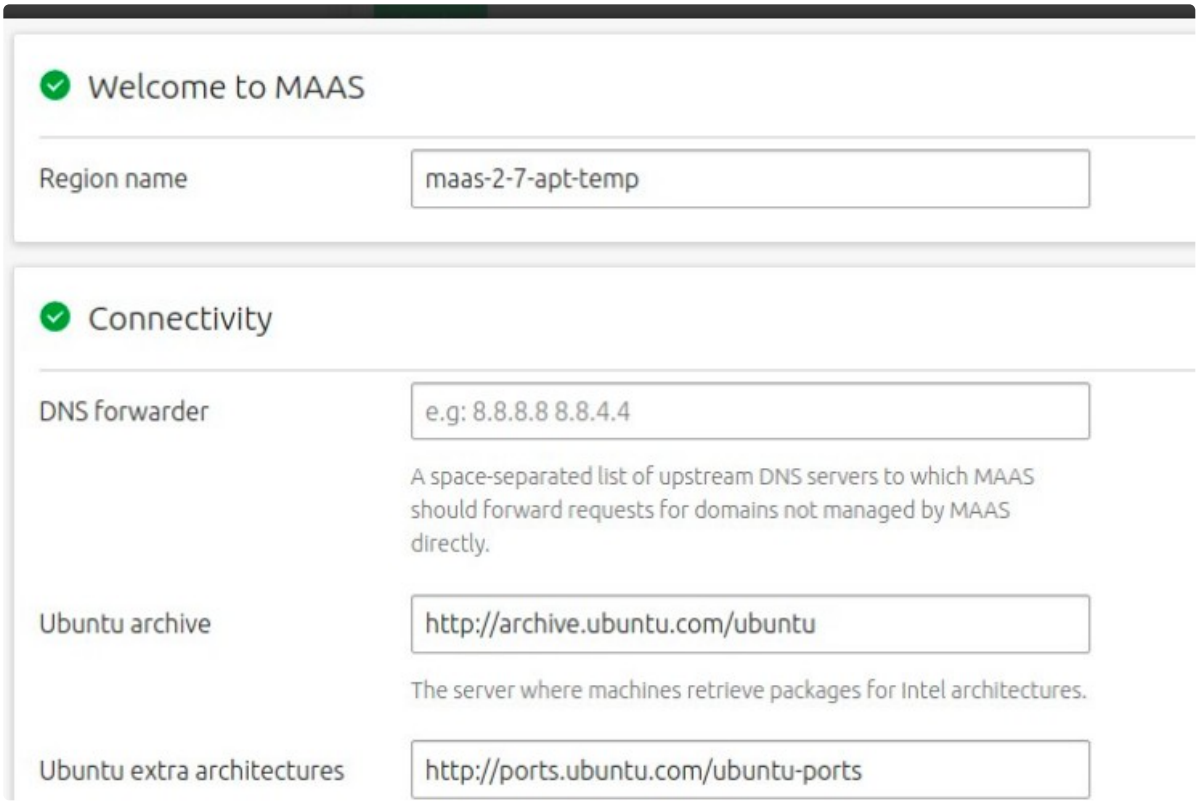
Copy

Follow the instructions to configure MAAS on the initial login

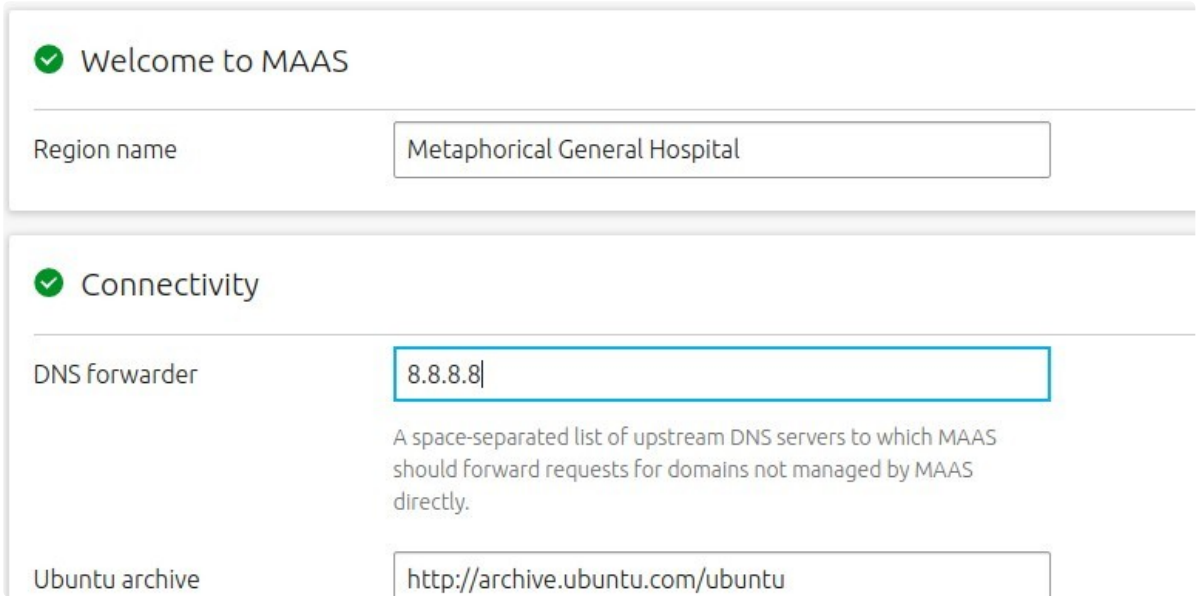
Log in at the prompts, with the login information you created when initializing MAAS.

Configuration

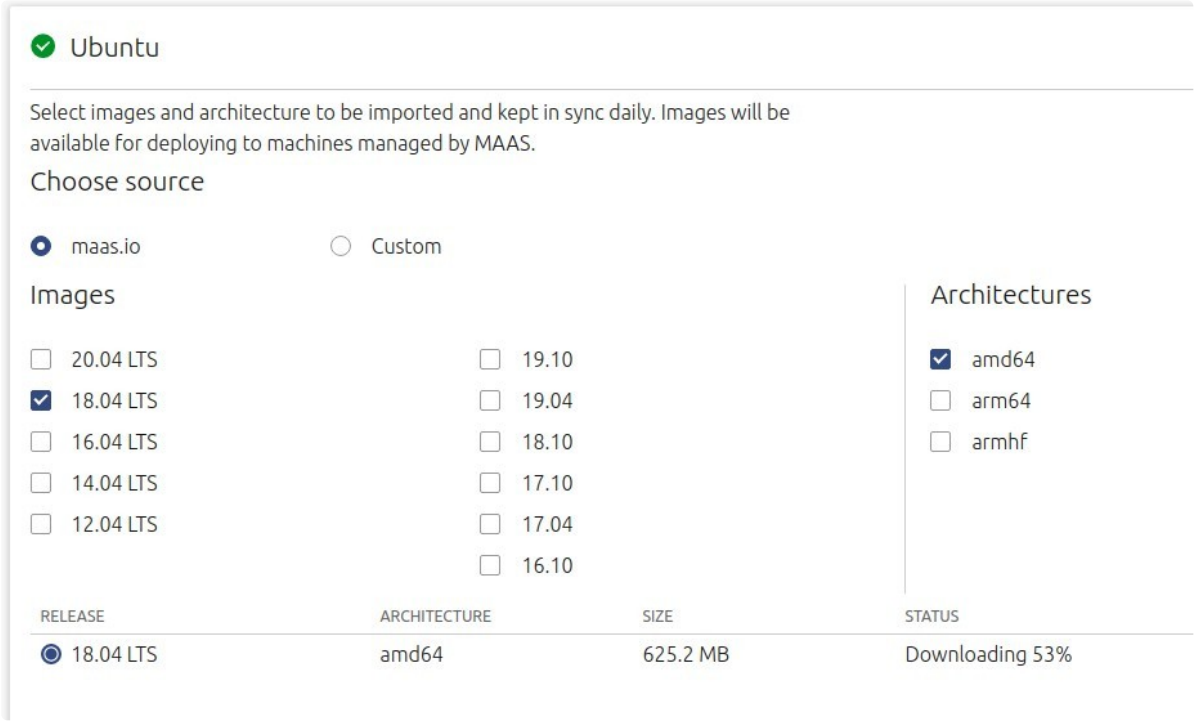
After a fresh MAAS installation, the web UI presents a couple of welcome screens. From these screens, you can set many system-wide options, including connectivity, image downloads, and authentication keys.



Your main concerns for this experiment are the DNS forwarder, the Ubuntu image import section, and the SSH public key, though you might want to set the region name to something memorable since this text will appear at the bottom of every MAAS screen in this install domain. Set the DNS forwarder to something obvious, e.g., `8.8.8.8`, Google's DNS server. Set this to your own internal DNS server if you know the IP address.

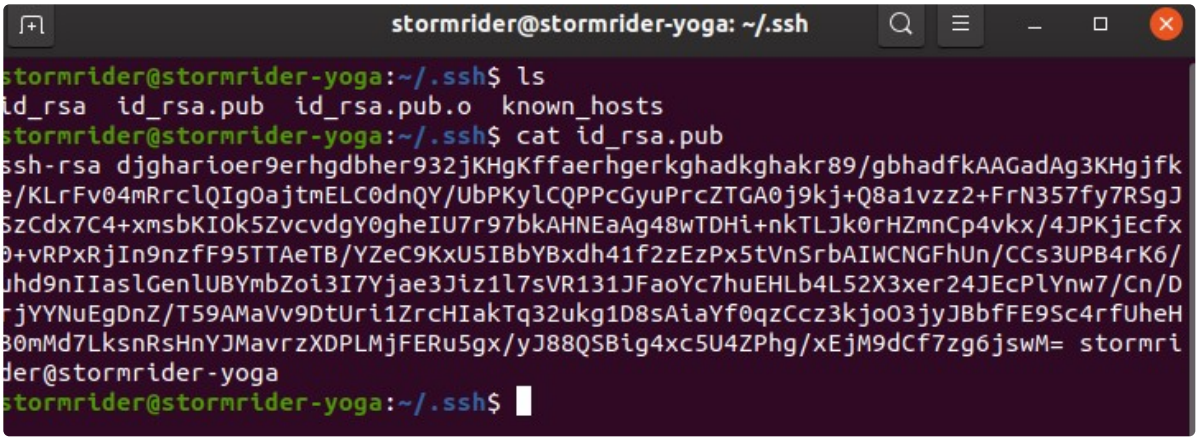


Select an Ubuntu image to import, noting that you may be required to select at least one LTS version, depending upon the version of MAAS that snap installed. In this example, we've already chosen an image, and downloading is partially complete.

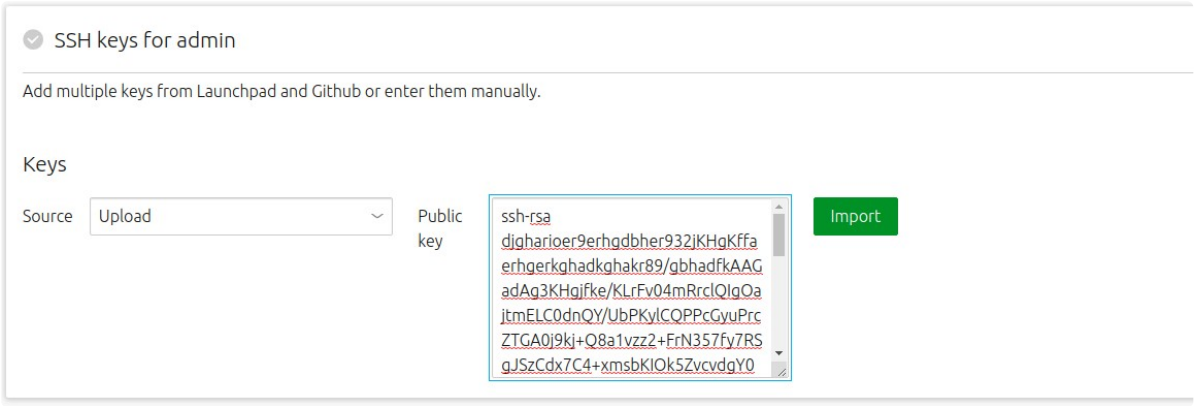


When you click on "Continue," the screen will shift to a screen labeled, "SSH keys for admin:"

If you want to use your existing public key from your home directory, you can select "Upload" and then copy your entire public key from `.ssh/id_rsa.pub` (or wherever you may have stored the key):



and paste it into the block labeled "Public key." Finally, press the "Import" button to import this key:



With this complete, you'll see that MAAS has been successfully set up. Click 'Go to the Dashboard' to proceed.

+ Add label

Be the first to add a reaction