



Azure Marketplace Image

Contents

Azure Marketplace Image User Guide.....	3
Configuring Azure.....	4
Selecting an Azure machine type.....	4
Security groups.....	4
Launching the ARM template.....	4
Connecting agents.....	5
Certificates and DNS configuration.....	6
Changing the master hostname and regenerating certificates.....	6
Tuning and configuration.....	8
Running commands on nodes.....	8
Migrating to a larger instance type.....	8
Upgrading.....	8
Troubleshooting.....	9
PE password hasn't been set, or I don't have the password.....	9
Agents can't connect because PE configuration isn't yet complete.....	9
SSH user name or credentials don't work when connecting to the EC2 instance.....	9
Unsigned certificates page in the console has an internal URL for the curl.....	9
Public and private hostnames are incorrectly configured in PE.....	10
Puppet agent run won't work when started by a non-root user.....	10

Azure Marketplace Image User Guide

Thank you for choosing Puppet Enterprise (PE). System administrators use PE to programmatically provision, configure, and manage servers, network devices, and storage, whether in a data center or in the cloud.

This guide covers how to install, configure, and use the Puppet Enterprise Azure Marketplace template for Azure Resource Manager (ARM).

For additional information about Puppet Enterprise, see these pages in the PE documentation:

- [PE Overview](#)
- [Architecture overview](#)
- [Supported operating systems and devices](#)

Overview

This Puppet Enterprise Azure Marketplace Image contains a [monolithic installation](#) of PE, comprised of a collection of PE services running on a single Azure VM—the Puppet master. The [PE Architecture Overview](#) describes PE's major services and components.

After you launch the VM with the ARM template, PE is automatically and securely configured during operating system boot. Unique keys and certificates are generated, PE services are started, and the local node is brought under PE management.

Next, install Puppet agent on systems that will be managed by the master. Agent packages are included for all [supported platforms](#), including *nix distributions, Windows, and macOS, so you can quickly deploy and provision a PE-managed virtual network.

- [Configuring Azure](#) on page 4

Before installing PE, select an appropriate Azure machine type and configure a security group to accommodate PE's required network ports.

- [Launching the ARM template](#) on page 4

The Puppet Enterprise ARM starting template launches the Puppet Enterprise Azure Image.

- [Connecting agents](#) on page 5

To connect agents to the master, this image is configured to use a private virtual network.

- [Certificates and DNS configuration](#) on page 6

Using the master's private VM hostname, PE generates certificates which include the master's public fully qualified domain name and `puppet` as alternate DNS names.

- [Tuning and configuration](#) on page 8

The PE installation contained in this image is pre-configured and tuned for use with up to 4,000 nodes.

- [Running commands on nodes](#) on page 8

When running these Puppet commands on nodes in Azure environments, you must first switch to the superuser role by running `sudo su`.

- [Migrating to a larger instance type](#) on page 8

As your PE-managed infrastructure grows, moving to a larger Azure VM size improves your system performance.

- [Upgrading](#) on page 8

There is no automated process for upgrading to newer versions of the PE Azure ARM Image. As with any PE installation, however, you can manually upgrade PE to the latest release from the Puppet master's command line.

- [Troubleshooting](#) on page 9

When using the Puppet Enterprise Azure Marketplace Image and ARM template, you might encounter some problems that this troubleshooting section can address.

Configuring Azure

Before installing PE, select an appropriate Azure machine type and configure a security group to accommodate PE's required network ports.

Selecting an Azure machine type

The VM you're running PE on must have sufficient memory and processing power.

To fulfill the [hardware requirements](#), use at least a Standard D3 v2 VM as a minimum baseline. The PE installation contained in this image does not include additional Puppet compile masters and therefore should not manage more than 4,000 nodes.

Security groups

The PE ARM template defines the ports that are required for PE VM to be configured and accessible. The security group is created in the resource group associated with your account.

TCP ports

TCP Port	Description	VPC Access
22	SSH	Externally visible
433	Puppet Enterprise console (HTTP)	Externally visible
8140	Puppet master	Visible inside virtual network only
8142	Orchestration services	Visible inside virtual network only
8143	Orchestration services	Visible inside virtual network only
61613	MCollective	Visible inside virtual network only

For more information about PE's required network availability, see its [documentation](#).

Launching the ARM template

The Puppet Enterprise ARM starting template launches the Puppet Enterprise Azure Image.

1. Launch a Microsoft Azure PE VM from the [Azure Portal's templates](#), [Powershell](#), or SDKs: [Ruby](#), [Python](#), [Go](#), or [Java](#). To configure your Puppet master, select or create each of the following options:
 - **Resource Group** (required): Create a new group or reuse an existing group.
 - **Location** (required): If you use an existing resource group, the location must match the resource group's location.
 - **Admin Password** (required): The password for the Admin user. If you choose `sshPublicKey` as the authorisation for the VM, this password is used as a fallback.
 - **VM Size**: The minimum VM size capable of running Puppet Enterprise is `Standard_D3_v2`.
 - **Admin User Name**: The user name to use when logging into the VM with SSH.
 - **Authentication Type**: The supported types are `sshPublicKey` or `password`.
 - **SSH Public Key**: If your authentication type is `sshPublicKey`, this should contain your SSH public key.
 - **Virtual Network New Or Existing**: If this is set to `existing`, you do not need to enter an address prefix or subnet.
 - **Public IP Address New Or Existing** : Set to `new` and specify a static IP address. If the IP is dynamic and the VM is restarted, you won't be able to access the console, because the console uses the initial public IP address.
 - **Public IP Address Domain Name Label**: This label forms the prefix of the VM's fully qualified domain name.
 - **Storage Account New Or Existing**: If this is set to `existing`, you do not need to enter an account type.
2. Connect to the Azure VM using the key pair created in step 1 and the user name `puppetadmin` by running:

```
ssh -i ~/.ssh/id_rsa.pem puppetadmin@<Azure VM FQDN or public IP>
```

SSH keys are automatically provisioned based on the specified authentication type.

Or, when connecting enter the admin password you defined previously:

```
ssh puppetadmin@<Azure VM FQDN or public IP>
```

3. Wait for PE configuration, which begins automatically while booting the Azure VM, to complete. This takes about 8 minutes, and must finish before you connect and manage nodes. To determine when the PE services are fully configured, use the `check_status.sh` script.

```
/opt/puppetlabs/azure/bin/check_status.sh --wait
```

4. As root, set the console password. Console access is disabled until the password is set.

```
sudo /opt/puppetlabs/azure/bin/set_console_password.sh
Puppet Enterprise console password: *****
```

Note: You can run this script at any time to reset the console password.

5. Using a web browser, connect to the console, accept the console's certificate, and login with user name `admin` and the password set in step 4.

Once you are logged in, the console indicates that the master is actively managed by showing “1 Nodes run in enforcement”; the node it refers to is the master itself.

For help accessing the console, see the [PE documentation](#).

6. Optional. Configure [PE certificate auto-signing](#).

The Puppet master is now ready to manage nodes, and for agents to connect to it.

Connecting agents

To connect agents to the master, this image is configured to use a private virtual network.

Before you begin

To manage nodes outside of Azure or across Azure Virtual Networks, configure the master by running the included `update_agent_repos.sh` script, which can be obtained from the [Azure portal](#) in the "Virtual Machines" section.

```
sudo /opt/puppetlabs/azure/bin/update_agent_repos.sh public
[puppetmaster_public_fqdn]
```

This image contains agent packages for all [supported platforms](#). These instructions are a quick reference to the agent installation and configuration process. For detailed information, see the [PE documentation](#).

To manage a new *nix node:

1. Launch the node, then connect to it over SSH.
2. Install the agent package for a typical *nix platform by running:

```
curl -k https://<PUPPET-MASTER-HOST>:8140/packages/current/install.bash |
sudo bash
```

Use the Azure internal hostname or IP address of the Puppet master .

During installation, the agent submits a certificate signing request to the master.

3. If certificate auto-signing is not enabled on the master, manually sign the agent certificate request using the console or command line by running:

```
sudo puppet cert list

sudo puppet cert sign <NAME>
```

Certificates and DNS configuration

Using the master's private VM hostname, PE generates certificates which include the master's public fully qualified domain name and `puppet` as alternate DNS names.

For more information about EC2 hostnames, see the troubleshooting section.

Managing Azure ARM nodes by their private hostname (rather than the public hostname) keeps their hostnames consistent if, for example, the node is resized or changed to a different VM machine type. This requires less work when administering a virtual network.

Related information

[Troubleshooting](#) on page 9

When using the Puppet Enterprise Azure Marketplace Image and ARM template, you might encounter some problems that this troubleshooting section can address.

Changing the master hostname and regenerating certificates

While not recommended, you can change the hostname of your master and use this hostname to generate new PE certificates. Do this before you connect any agents to the master.

Note: These instructions are specific to the PE installation contained in this image and include some minor variations and simplifications of the instructions outlined in the [PE documentation](#).

1. Connect to the master:

```
ssh -i ~/.ssh/<private key>.pem puppetadmin@<Public IP or FQDN of PE
Master>
```

2. Wait for PE configuration to complete, and use the `check_status.sh` script to confirm its status.

3. Stop all PE services:

```
sudo /usr/local/bin/puppet resource service puppet ensure=stopped
sudo /usr/local/bin/puppet resource service pe-puppetserver ensure=stopped
sudo /usr/local/bin/puppet resource service pe-activemq ensure=stopped
sudo /usr/local/bin/puppet resource service mcollective ensure=stopped
sudo /usr/local/bin/puppet resource service pe-puppetdb ensure=stopped
sudo /usr/local/bin/puppet resource service pe-postgresql ensure=stopped
sudo /usr/local/bin/puppet resource service pe-console-services
ensure=stopped
sudo /usr/local/bin/puppet resource service pe-nginx ensure=stopped
sudo /usr/local/bin/puppet resource service pe-orchestration-services
ensure=stopped
sudo /usr/local/bin/puppet resource service pxp-agent ensure=stopped
```

4. Copy the SSL certificate directory `/etc/puppetlabs/puppet/ssl/` to a backup location:

```
sudo mv /etc/puppetlabs/puppet/ssl /etc/puppetlabs/puppet/ssl.backup
```

Should anything go wrong during this process, you can restore certificates and your PE installation.

5. Delete the local cached catalog, which will be invalidated by the new hostname:

```
sudo rm -f /opt/puppetlabs/puppet/cache/client_data/catalog/*
```

6. Set the Puppet master's new hostname. This depends on your configuration, and could be as simple as following these instructions, or this might entail configuring a DNS service like [AWS's Route 53](#).

- a. Set the hostname: `sudo hostnamectl set-hostname <NEW-MASTER-HOSTNAME>`
- b. Add the hostname to `/etc/hosts`
- c. Add `preserve_hostname: true` to the main section of `/etc/cloud/cloud.cfg`, for example, immediately below `disable_root: 1`

7. Verify that the master and agents can resolve the new hostname. Puppet must be able to contact this hostname to connect to PE services and complete the certificate generation process. Use the current admin user name if it has been changed from the default:

```
ssh puppetadmin@<NEW-MASTER-HOSTNAME>
```

8. Edit the master's `/etc/puppetlabs/puppet/puppet.conf` file and set the `certname` parameter in both the `[main]` and `[master]` sections to the new hostname.

Note: For best compatibility, limit the `certname` to letters, numbers, periods, underscores, and dashes.

9. Optional. To also include alternate DNS names, edit `/etc/puppetlabs/enterprise/conf.d/pe.conf` and set `pe_install::puppet_master_dnsaltnames` to a list of desired alternate hostnames.

Note: If you want to change the alternate DNS names on the master later, you must repeat all of these steps.

10. Remove the contents of the config files so Puppet can regenerate them with the new hostname:

```
echo '' > /etc/puppetlabs/nginx/conf.d/proxy.conf
echo '' > /etc/puppetlabs/nginx/conf.d/http_redirect.conf
echo '' > /etc/puppetlabs/puppetdb/certificate-whitelist
echo '' > /etc/puppetlabs/console-services/rbac-certificate-whitelist
echo '<beans></beans>' > /etc/puppetlabs/activemq/activemq.xml
```

11. Remove the old hostname from `/etc/puppetlabs/puppet/autosign.conf`.

12. Use the Puppet Enterprise module to regenerate certificates and restart PE services:

```
sudo /usr/local/bin/puppet infrastructure configure --no-recover --
modulepath /opt/puppetlabs/server/data/enterprise/modules
```

Should anything go wrong during this process, you can restore certificates and your PE installation.

13. Remove the former master hostname from the list of managed nodes:

```
sudo /usr/local/bin/puppet node purge <FORMER-MASTER-HOSTNAME>
```

14. Start a local agent run on the master:

```
sudo /usr/local/bin/puppet agent -t
```

15. To confirm the master's certname at any time, run this command:

```
sudo /usr/local/bin/puppet config print certname
```

Tuning and configuration

The PE installation contained in this image is pre-configured and tuned for use with up to 4,000 nodes.

For a list of default settings, and instructions for changing these settings and reconfiguring PE services, see the [Tuning monolithic installations](#) docs.

Running commands on nodes

When running these Puppet commands on nodes in Azure environments, you must first switch to the superuser role by running `sudo su`.

- `puppet agent -t`
- `puppet ca`
- `puppet cert`
- `puppet enterprise support`
- `puppet infrastructure`
- `puppet license`
- `puppet lookup`
- `puppet node`
- `puppet plugin`
- `/opt/puppetlabs/server/bin/set_console_admin_password.rb`

Migrating to a larger instance type

As your PE-managed infrastructure grows, moving to a larger Azure VM size improves your system performance.

To add disks through the Azure Portal or the command line, see [Add a data disk to a VM through the Portal](#) in the Azure documentation.

Upgrading

There is no automated process for upgrading to newer versions of the PE Azure ARM Image. As with any PE installation, however, you can manually upgrade PE to the latest release from the Puppet master's command line.

For information, see [Upgrading Puppet Enterprise](#).

Troubleshooting

When using the Puppet Enterprise Azure Marketplace Image and ARM template, you might encounter some problems that this troubleshooting section can address.

You can get help with Puppet from us and the rest of the Puppet community. If you've purchased Puppet Enterprise, you can also access our knowledge base and open a support ticket.

You can also find troubleshooting information in the PE docs:

- [Troubleshooting connections between components](#)
- [Troubleshooting Code Manager](#)
- [Troubleshooting the databases](#)
- [Troubleshooting Puppet core](#)
- [Troubleshooting MCollective](#)
- [Troubleshooting Puppet on Windows](#)

PE password hasn't been set, or I don't have the password

Access to the console is disabled until you set the console password.

To create or reset a password, use the `set_console_password.sh` script described in the [Launching the ARM template](#) section.

Related information

[Launching the ARM template](#) on page 4

The Puppet Enterprise ARM starting template launches the Puppet Enterprise Azure Image.

Agents can't connect because PE configuration isn't yet complete

When first booting the Puppet Master Azure VM, PE configuration must complete before you can connect and manage agents.

Run the `check_status.sh` script to determine whether configuration has finished:

```
/opt/puppetlabs/azure/bin/check_status.sh --wait
```

SSH user name or credentials don't work when connecting to the EC2 instance

This image is created and configured using Azure ARM templates. You can provision the VM to use SSH key pair access, or a user name and password.

Note: The private key must be the pair of the `sshPublicKey` that was entered as a parameter for the ARM template.

```
ssh -i ~/.ssh/<id_rsa>.pem puppetadmin@<Public IP or VM FQDN>
```

Alternatively, you can set a password to log in using the admin password you entered as a parameter to the ARM template.

Related information

[Launching the ARM template](#) on page 4

The Puppet Enterprise ARM starting template launches the Puppet Enterprise Azure Image.

Unsigned certificates page in the console has an internal URL for the curl

The `curl` command on the console's Unsigned Certificates page contains a URL that uses your master's internal DNS name, which won't work for nodes that cannot resolve that name.

For example, the command might look like:

```
curl -k https://
puppetmasterv2.liweionmsdnnwoe.xx.internal.cloudapp.net:8140/packages/
current/install.bash | sudo bash
```

To use the curl command, change the internal DNS name to the VM's external qualified domain name, which is available from the [Azure Portal](#) in the Virtual Machines section.

Public and private hostnames are incorrectly configured in PE

This PE installation and recommended security group settings are configured to use the private (internal) VM hostname for communicating from managed nodes to the master.

To find the local hostname on the master, use Factor.

```
$ factor networking.fqdn
puppetmdns.westus2.cloudapp.azure.com

$ factor networking.dhcp
168.63.129.16

$ factor networking.ip
10.0.0.4

$ factor networking.domain
puppetmasterv2.iwsmirleafiednzwua4ghulbc.xx.internal.cloudapp.net
```

Then follow the steps in Certificates and DNS configuration section.

Related information

[Certificates and DNS configuration](#) on page 6

Using the master's private VM hostname, PE generates certificates which include the master's public fully qualified domain name and puppet as alternate DNS names.

Puppet agent run won't work when started by a non-root user

An agent run initiated by puppetadmin or any other non-root user will fail when attempting to access certificates, packages, and services.

Always start manual Puppet agent runs with super-user privileges:

```
sudo /usr/local/bin/puppet agent -t
```