Test Plan for XenServer Fuel Plugin

Release 3.0-3.0.0-1

John Hua (john.hua@citrix.com)

April 19, 2016

Contents

1 XenServer Fuel Plugin

XenServer Fuel Plugin will help to deploy Mirantis OpenStack using the XenServer hypervisor to host virtual machines, making all the necessary changes to the Mirantis OpenStack to use the xenapi Nova compute driver.

2 Developer's Specification

See developers specification in the source code repository at https://git.openstack.org/openstack/fuel-plugin-xenserver

2.1 Limitations

This version of XenServer Fuel Plugin has not been certified to work with the Ceilometer, MongoDB or Murano additional services. Future versions of the plugin will relax these restrictions.

3 Test strategy

3.1 Acceptance criteria

All tests that do not depend on additional services must pass.

3.2 Test environment, infrastructure and tools

All tests need to be run under a cluster of at least 4 XenServer machines with 3 physical NICs. As HA and multihost are enabled, a topology of 3 Controller Nodes + 3 Compute Nodes + 1 Storage Node will be recommended to be created as VMs on XenServer machines. Easy setup and management of those XenServers and VM Nodes can be achieved using XenCenter and a plugin, described below, to add an internal management network to VMs.

To simplify setup, the fuel master is also installed on the XenServer hosts (so XenServer hosts can fully control the network setup), but this is not required.

While many networking setups are expected to work, the following setup is used by this test plan:

- eth0 / "Access network": Used to access the XenServer hosts and the Fuel Master's web interface

- eth1 / "Control network": OpenStack control plane (management and storage), the PXE network and the public network; all separated by VLAN tags. The public network is also on this network, and if a VLAN is required this is applied by the switch for untagged traffic.

- eth2 / "VLAN network": This version of the plugin only supports VLAN segmentation for Neutron networking. This device carries all of the VLANs to be used by Neutron for VM traffic.
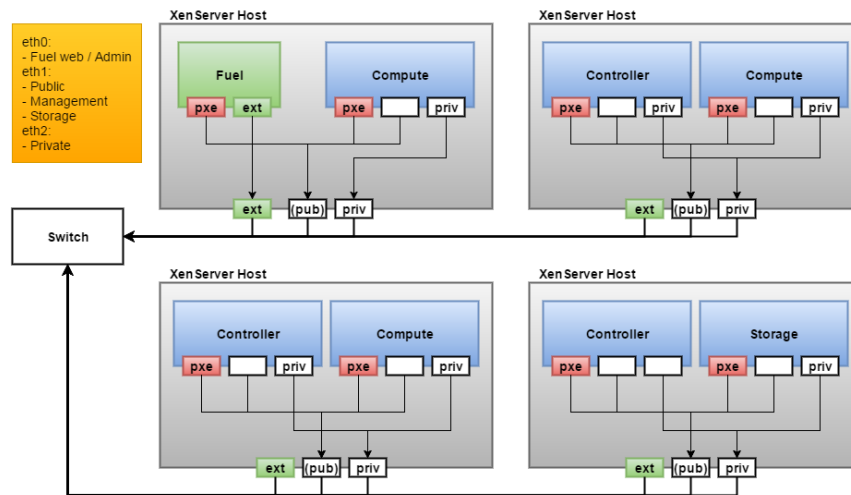
3.3 Product compatibility matrix

The plugin is compatible with MOS 8.0 and XenServer 6.5 SP1, with all hotfixes applied (Especially hotfix XS65ESP1013).

4 Prerequirements

4.1 Prepare XenServers

1. Install and start XenCenter on your Windows PC

2. Add new servers with a common root password in XenCenter

3. Plug three physical NIC to each of all XenServer machines, make sure the cabling of all NIC 0 are attached to the 'access' network, all NIC 1 to the 'public' network and NIC 2 are attached to the isolated, 'VLAN network'. It is recommended to rename these networks using XenCenter to make the network topology clear.

4. Add a further network, with a vlan tag that will be used for PXE.

4.2 Prepare Fuel Master

1. Upload Fuel ISO to a NFS/Samba server and make it accessible to your XenServer hosts.

2. Select a XenServer and click "New Storage" button, in the popup window check on CIFS/NFS ISO library and input NFS/Samba server path.

3. Create a new VM in XenCenter using the "Other Install Media" template (to ensure a HVM domain is created) with and PXE network as eth0 and 'access' network as eth1. In the Console Tab, insert Fuel ISO and install.

4. In fuel menu, enable eth1 with DHCP so the fuel master can be accessed over the 'access' network.

5. Select Fuel Master in XenCenter and switch to Console tab, login with prompted user and password

6. Visit http://ip_of_fuel_master:8000 in browser.

5 Type of testing

5.1 Install XenServer Fuel Plugin

Test Case ID	insall_xfp
Description	Verify that XenServer Fuel Plugin can be installed into Fuel Master, and the new OpenStack release is registered.

Steps	<pre>[root@fuel-master ~]# fuel plugins
	id | name | version | package_version
	9 | fuel-plugin-xenserver | 3.0.0 | 3.0.0</pre>

Expected Result	<pre>fuel plugins
	id | name | version | package_version
	2 | fuel-plugin-xenserver | 3.0.0 | 3.0.0
	fuel rel
	id | name | state | operating_system | version
	2 | Liberty on Ubuntu 14.04 | available | Ubuntu |
	liberty-8.0
	3 | Liberty+Citrix XenServer on Ubuntu 14.04 | available |
	Ubuntu | liberty-8.0
	1 | Liberty on CentOS 6.5 | unavailable | CentOS |
	liberty-8.0</pre>

5.2 Prepare Nodes

Test Case ID	prepare_nodes
Description	Verify all controller/compute/storage nodes are ready for PXE install.
Steps	1. Create 3 new VMs in XenCenter in different XenServers and name them Controller1, Controller2, Controller3 2. Create 3 new VMs in XenCenter in different XenServers and name them Compute1, Compute2, Compute3 3. Create 1 new VM in XenCenter and name it Storage1 4. Add 'PXE' network as eth0, Public/Management/Storage network as eth1 and 'VLAN network' as eth2 to each of new VMs created above.
Expected Result	All nodes are shown in XenCenter with 'PXE network' as eth0 and 'VLAN network' as eth1.

5.3 Install XenCenter HIMN plugin

Test Case ID	install_xcp
Description	Verify XenCenter HIMN plugin is installed to Windows.
Steps	1. Download SetupHIMN from http://ca.downloads.xensource.com/OpenStack/Plugins/ 2. Install MSI to your XenCenter 3. Restart XenCenter
Expected Result	Right click on any selected VMs, there will be a menu item "Manage internal management network".

5.4 Add Host Internal Management Network to Compute Nodes

Test Case ID	add_himn
Description	Verify (or add) Host Internal Management Network is added to all Compute Nodes.
Steps	1. Select Compute1, Compute2, Compute3 in XenCenter 2. Right click on above nodes and select "Manage internal management network" menu. 3. In the popup window, after status detection, make sure all selected Compute nodes are checked on. Click on "Manage internal management network" button. 4. After processing, the status column should be shown as management network is added with new generated MAC address 5. Close the management network window
Expected Result	The wizard will report success, however the networks may not be visible in XenCenter.

5.5 Create an OpenStack environment with XenServer Fuel Plugin

Test Case ID	create_env
Description	Verify that an OpenStack environment created with XenServer Fuel Plugin can have XenServer options and options of hypervisor/network/storage/additional services are disabled.
Steps	1. Create new OpenStack environment Fuel Web UI and select "Liberty+Citrix XenServer on Ubuntu 14.04" in the OpenStack release dropdown list 2. Hypervisor is default to QEMU, Network is default to "Neutron with VLAN segmentation" and Storage is default to Cinder. Other options are disabled. 3. In Nodes Tab, add all 3 Controller Nodes, 3 Compute Nodes and 1 Storage Node. 4. Select all Compute Nodes and click "Configure Interfaces", drag Storage/Management network from default eth0 to eth1, Private network to eth2. Leave PXE on eth0. No networks should be assigned to the final interface. 5. Select all Controller and Storage Nodes and click "Configure Interfaces", drag Storage/Management network from default eth0 to eth1, Private network to eth2. Leave PXE on eth0. 6. In Networks Tab, set the vlan tags according to your network interfaces previous set and make sure network range will not be conflicting with other systems in the same lab. Then click "Verify Networks" button. 7. In the Settings Tab under the side tab "Others", input the credential applied to all your XenServer hosts. 8. Click "Deploy Changes" button
Expected Result	Deploy of nodes all succeed

5.6 Verify hypervisor type

Test Case ID	verify_hypervisor
Description	Verify that all hypervisors are identified by OpenStack as 'xen'.
Steps	1. Login to Horizon with admin user when OpenStack deployment is finished. 2. Enter into Admin->Hypervisors
Expected Result	The Type column should show xen for all hypervisors.

5.7 Create guest instances

Test Case ID	create_instances
Description	Verify that new environment can create guest instances.
Steps	1. Create an instance with image of TestVM and flavor of m1.tiny in either of Horizon or Controller Node. 2. Find the instance in XenCenter and switch to Console Tab. 3. Login with the username and password that prompted in the terminal screen. 4. Ping out to 8.8.8.8
Expected Result	Guest instances can ping out.

5.8 Verify Fuel Health Checks

Test Case ID	verify_health_checks
Description	Ensure that all applicable health checks pass
Steps	1. Within the Fuel Master, select the appropriate environment 2. Run all health checks and wait for completion
Expected Result	All health checks, except those requiring additional services (e.g. Ceilometer) and those where the configuration has not been changed from the defaults, pass

6 Mandatory Tests

6.1 Install plugin and deploy environment

Covered above.

6.2 Modifying env with enabled plugin (removing/adding compute nodes)

Test Case ID	modify_env_compute_nodes
Description	Adding/removing compute nodes to an existing environment
Steps	1. Create one more compute following the procedure in step prepare_nodes 2. Add compute node to an existing environment 3. Redeploy cluster 4. Run Health Check 5. Remove a compute node 6. Redeploy cluster 7. Run Health Check
Expected Result	Removing a compute node will cause "Sanity tests-Check that required services are running" fail. "Some nova services have not been started.. Please refer to OpenStack logs for more details."

6.3 Modifying env with enabled plugin (removing/adding controller nodes)

Test Case ID	modify_env_controller_nodes
Description	Adding/removing controller nodes to an existing environment
Steps	1. Create one more controller following the procedure in step prepare_nodes 2. Add controller node to an existing environment 3. Redeploy cluster 4. Run Health Check 5. Remove a compute node (not the primary controller node) 6. Redeploy cluster 7. Run Health Check
Expected Result	(Probably caused by previous test modify_env_compute_nodes)Removing a compute node will cause "Sanity tests-Check that required services are running" fail. "Some nova services have not been started.. Please refer to OpenStack logs for more details."

6.4 Uninstall of plugin with deployed environment

Test Case ID	uninstall_plugin_with_deployed_env
Description	Verify XenServer Fuel Plugin cannot be uninstalled before all dependant environments are removed.
Steps	`fuel plugins --remove fuel-plugin-xenserver==3.0.0`
Expected Result	400 Client Error: Bad Request (Can't delete plugin which is enabled for some environment.)

6.5 Uninstall of plugin

Test Case ID	uninstall_plugin
Description	Verify XenServer Fuel Plugin can be uninstalled as well as XenServer OpenStack release after all dependant environments are removed.

Steps	<pre>fuel plugins --remove fuel-plugin-xenserver==3.0.0
	fuel rel
	id | name | state | operating_system | version
	2 | Liberty on Ubuntu 14.04 | available | Ubuntu |
	liberty-8.0
	3 | Liberty+Citrix XenServer on Ubuntu 14.04 | available |
	Ubuntu | liberty-8.0
	1 | Liberty on CentOS 6.5 | unavailable | CentOS |
	liberty-8.0</pre>

Expected Result	Both of plugin and release are removed.

7 Appendix

- XenServer Fuel Plugin GitHub: https://git.openstack.org/openstack/fuel-plugin-xenserver
- XenCenter HIMN Plugin GitHub: https://github.com/citrix-openstack/xencenter-himn-plugin
- Plugin download server: http://ca.downloads.xensource.com/OpenStack/Plugins/

8 Revision history

Version	Revision Date	Editor	Comment
1.0	18.09.2015	John Hua (john.hua@citrix.com)	First draft.
2.0	18.11.2015	John Hua (john.hua@citrix.com)	Revised for Fuel 7.0
3.0	22.03.2016	John Hua (john.hua@citrix.com)	Revised for Fuel 8.0