Deploying ESX Cloud

Contents

Deploying ESX Cloud

This page describes the procedure to deploy ESX cloud using input model.

Important

Before you start your ESX cloud deployment ensure that you read the following instructions carefully.

- Refer to ***Important Notes*** in the Installation Guide.
- Prepare your baremetal hardware on all nodes. For instruction, please refer to ***Before You Start***.

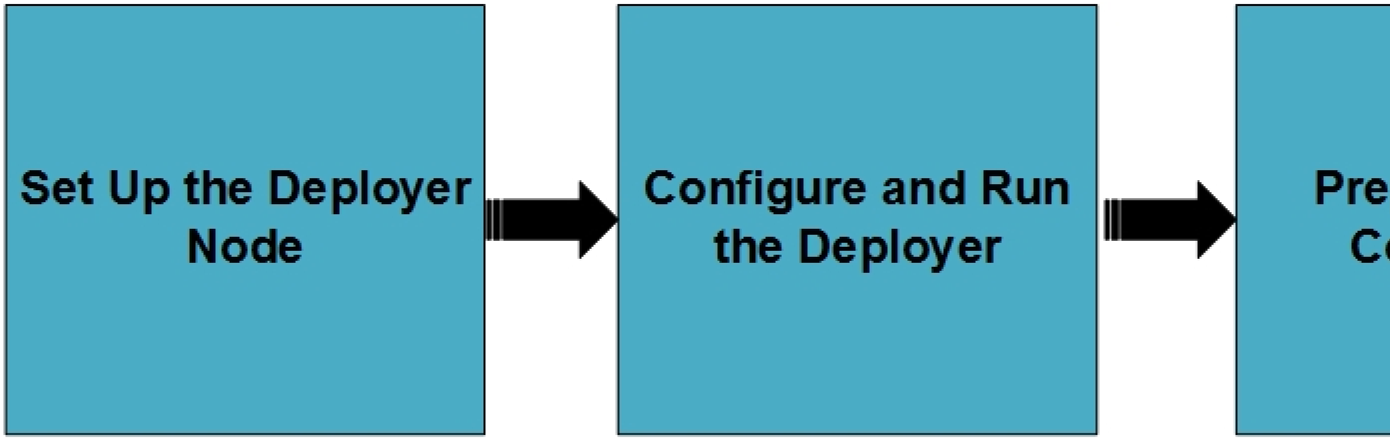
Prerequisite

ESX/vCenter integration is not fully automatic, vCenter administrators are advised of the following responsibilities to ensure secure operation:

- The VMware administrator is responsible for administration of the vCenter servers and the ESX nodes using the VMware administration tools. These responsibilities include:
    - Installing and configuring vCenter Server
    - Installing and configuring ESX server and ESX cluster
    - Installing and configuring shared datastores
    - Establishing network connectivity between the ESX network and the HP Helion management network
- The VMware administration staff is responsible for the review of vCenter logs. These logs are not automatically included in Helion centralized logging.
- Logging levels for vCenter should be set appropriately to prevent logging of the password for the Helion message queue.
- The vCenter cluster and ESX Compute nodes must be appropriately backed up.
- Backup procedures for vCenter should ensure that the file containing the Helion configuration as part of Nova and Cinder volume services is backed up and the backups are protected appropriately.
- Since the file containing the Helion message queue password could appear in the swap area of a vCenter server, appropriate controls should be applied to the vCenter cluster to prevent discovery of the password via snooping of the swap area or memory dumps

Deploy ESX Cloud

At a high level, here are the steps to configure and deploy ESX cloud:



Procedure to Deploy ESX cloud

The following topics in this section explain how to deploy ESX cloud.

1. **Set up the Deployer Node**

   For the detailed instructions on setting up the deployer node, refer to *Set up the Deployer node* in the installation guide.
2. **Configure and Run the Deployer**

   For the detailed instructions on configuring and running deployer node, refer to *Configure and Run the Deployer* in the installation guide.
3. **Prepare Cloud Controllers**

   1. See a sample set of configuration files in the `~/helion/examples/one-region-poc-with-esx` directory. The accompanying README.md file explains the contents of each of the configuration files.
   2. Perform the steps **b** and **c** from *Configure your Environment* in the installation guide.
4. **Provision Controller Nodes**

   For the detailed instructions on provisioning the controller nodes, refer to *Deploy Cobbler* and *Provision the Nodes* in the installation guide.
5. **Temp: (Run the Configuration Processor)**

   For the detailed instructions on running the configuration processor, refer to steps **1** and **2** (**a** and **b**) from *Run the configuration Processor* in the installation guide.
6. **Deploy Cloud Controllers**

   Perform the following steps to deploy cloud controllers:

   1. Modify the `./helion/hlm/ansible/hlm-deploy.yml` to uncomment the line containing `eon.yml`.
   2. Run the following command:

      ```
      ansible-playbook -i hosts/verb_hosts site.yml
      ```
7. **Prepare and Deploy ESX Computes and OVSvAPP**

   The following sections describe the procedure to install and configure ESX compute and OVSvAPP on vCenter.

   - *Deploy Helion Linux Shell VM Template*
   - *Preparation for ESX Cloud Deployment*

Deploy Helion Linux Shell VM Template

The first step in deploying the ESX compute proxy and OVSvApps is to create a VM template that will make it easier to deploy the ESX compute proxy for each Cluster and OVSvApps on each ESX server.

Perform the following steps to deploy a template:

1. Import the `hlm-shell-vm.ova` in the vCenter using the vSphere client.
2. In the vSphere Client, click **File** and then click **Deploy OVF Template**.
3. Follow the instructions in the wizard to specify the data center, cluster, and node to install. Refer to the VMWare vSphere documentation as needed.

Preparation For ESX Cloud Deployment

Manage vCenters and Clusters

This section describes the procedures to prepare the ESX cloud for deployment.

Note: Source `service.osrc` before you perform the following procedure.

Register a vCenter Server

vCenter provides centralized management of virtual host and virtual machines from a single console.

1. Add a vCenter using EON CLI.

```
eon vcenter-add --name <vCenter Name> --ip-address <vCenter IP address> --
username <vCenter Username> --password <vCenter Password> --port <vCenter
 Port>
```

where:

- vCenter Name - the name of the vCenter server where the service is deployed.
- vCenter IP address - the IP address of the vCenter server where the service is deployed.
- vCenter Username - the username for the vCenter administrator.
- vCenter Password - the password for the vCenter administrator.
- vCenter Port - the vCenter server port.

Sample Output:

```
+------------+--------------------------------------+
| Property   | Value                                |
+------------+--------------------------------------+
| created_at | 2015-08-20T12:08:09.000000           |
| deleted    | False                                |
| deleted_at | None                                 |
| id         | BC9DED4E-1639-481D-B190-2B54A2BF5674 |
| ip_address | 10.1.200.41                          |
| name       | vc01                                 |
| password   | <SANITIZED>                          |
| port       | 443                                  |
| type       | vcenter                              |
| updated_at | 2015-08-20T12:08:09.000000           |
| username   | administrator@vsphere.local          |
+------------+--------------------------------------+
```

Register ESX Cloud Network Configuration

This involve getting a sample network information template. Fill the details of the template and use that template to register cloud network configuration for the vCenter.

1. Execute the following command to get the network information template:

```
eon get-network-info-template --filename <NETWORK_CONF_FILENAME>
```

For example:

```
eon get-network-info-template --filename net_conf.json
```

Sample file of `net_conf.json` is shown below:

```
{
    "network": {
        # Deployer Network details
        "deployer_network": {
            #Deployer Portgroup Name.
            "deployer_pg_name": "hlm-Deployer-PG",

            #VLAN id for Deployer Portgroup
            "deployer_vlan": "1702",

            #Enable DHCP for Deployer N/W
            "enable_deployer_dhcp": "no",

            #CIDR and gateway for deployer network
            "deployer_cidr": "172.170.2.0/24",
            "deployer_gateway_ip": "172.170.2.1",
            "deployer_node_ip": "172.170.1.10"
        },

        #Management Network details
        "management_network": {
            #Mgmt DVS name.
            "mgmt_dvs_name": "hlm-Mgmt",

            #Physical NIC name for Mgmt DVS
            "mgmt_nic_name": "vmnic1",

            #Mgmt Portgroup Name.
            "mgmt_pg_name": "hlm-Mgmt-PG",

            #Interface order: Example eth1
            "mgmt_interface_order": "eth1"
        },

        "data_network": {
            #Tenant network type
            "tenant_network_type": "vlan",

            "data_dvs_name": "hlm-Data",

            "data_nic_name": "vmnic2",

            #Data Portgroup Name.
            "data_pg_name": "hlm-Data-PG",

            #Interface order: Example eth2
            "data_interface_order": "eth2",

            #If more than one mgmt_nic_name are specified then NIC teaming 
will be enabled by default
            #Active uplink NICs for NIC teaming(If this field is empty, 
first data_nic_name will be "Active" and the rest will be in "Standby")
            "active_nics": "vmnic2",

            #Load Balancing. Please choose the corresponding number
            #    1 -> Route based on the originating virtual port
            #    2 -> Route based on IP hash
            #    3 -> Route based on source MAC hash
            #    4 -> Route based on physical NIC load
            #    5 -> Use explicit failover order
            "load_balancing": "1",

            #Network Failover Detection. Please choose the corresponding 
number
            #    1 -> Link Status
            #    2 -> Beacon Probing
            "network_failover_detection": "1",

            #Notify Switches(yes/no)
            "notify_switches": "yes"
        },

        "hpvcn_trunk_network": {
            #Trunk DVS name
            "trunk_dvs_name": "hlm-Trunk",

            #Trunk Portgroup Name.
            "trunk_pg_name": "hlm-Trunk-PG",
            #Interface order: Example eth3
            "trunk_interface_order": "eth3"
        },

        #VLAN Range for Data & Trunk port group. Please provide the range
 separated by a hyphen(vlan-vlan).
        #Multiple vlan or vlan ranges has to be a comma separated
 value(*OPTIONAL)
        "vlan_range": "1-4094"
    },

    "template": {
        #Provide the template/appliance name that will be used for cloning
        "template_name": "hlm-shell-vm"
    },

    "vmconfig": {
        #Number of CPUs for OVSvApp/Computeproxy VM
        "cpu": "4",

        #Amount of RAM in MB
        "memory_in_mb": "4096",

        #SSH public key content for OVSvAPP/Computeproxy password less
 login.
        "ssh_key": "ssh-rsa
 AAAAB3NzaC1yc2EAAAADAQABAAABAQCbSGs9OeJofAp7oHrztyAWX5LKK8ZSyLjRbmPwDls0qu
+obWxRi7vJF9SdRgOB44zoLyRT2i5DC9Vz3sg4zshygLdg9qwtyTKS5N0Qi
+R8D5rnbCAPGiU7eTu3jpgVy/
xJOuCo6u1TQm2zA8epsSisqjtg6o36gZNvVcOE8XBYr92Dc3wFncjCh+Ej
+X2WsKQHiais2fgCME1g4bj2r2E4+8oTiL/
g5bhrhl1fSwQZPAMc2WO18Eyum3ItHpD9stxr3OgEpR0sqk2piUasgT5lc4x9NGqa0RZgtbrEVjATBCdF6EOr
qWzx9SDeD4dyDa4Y+P+Letp stack@hlm"
    },
        # Do you want to skip inactive or maintenance mode hosts ?
    "skip_inactive_hosts": "yes",
        # Provide new host mo ids when you are adding a new host in an
 activated cluster.
    "new_host_mo_ids": ""
}
```

2. Modify the template (json file) as per your environment.

   ```
   vi <NETWORK_CONF_FILENAME>
   ```

   For example:

   ```
   vi net_conf.json
   ```

3. Use the template to register Cloud Network Configuration. This sets the network information for a vCenter which is used to deploy and configure compute proxy and OVSvAPP VMs during the cluster activation.

   ```
   eon set-network-info --vcenter-id <vCenter ID> --config-json
    <NETWORK_CONF_FILENAME>
   ```

   For example:

   ```
   eon set-network-info --vcenter-id <vCenter ID> --config-json net_conf.json
   ```

   Note: The vcenter ID is generated when you execute the above (**step 2**) command.

4. Execute the following command to view the list of clusters for the given vCenter.

   ```
   eon cluster-list --vcenter-id <vCenter ID>
   ```

   Sample Output

   ```
   +------------+----------+------------+---------------+
   | MOID       | Name     | Datacenter | Import Status |
   +------------+----------+------------+---------------+
   | domain-c21 | Cluster1 | DC1        | not_imported  |
   +------------+----------+------------+---------------+
   ```

Import Cluster

You can use one or more ESX clusters for ESX Cloud Deployment. When a Import Cluster is invoked, required ESX Compute Proxy and OVSvApp nodes are deployed.

1. Import the cluster for the EON database under the given vCenter.

   ```
   eon cluster-import --vcenter-id <vCenter ID> --cluster-name <Cluster Name>
    --cluster-moid <Cluster Moid>
   ```

   where:

   - vCenter ID - ID of the vcenter containing the cluster.
   - Cluster Name - the name of the cluster that needs to be imported.
   - cluster Moid - Moid of the cluster that needs to be imported.

   Sample Output

   ```
   +--------------+-----------+
   | Property     | Value     |
   +--------------+-----------+
   | cpu_free     | 83071.73  |
   | cpu_total    | 83072     |
   | cpu_used     | 0.27      |
   | datacenter   | DC1       |
   | disk_free    | 1022.79   |
   | disk_total   | 1023.75   |
   | errors       | []        |
   | memory_free  | 496.82    |
   | memory_total | 511.76    |
   | memory_used  | 14.94     |
   | name         | Cluster1  |
   | state        | importing |
   | switches     | []        |
   +--------------+-----------+
   ```

   One vCenter can have multiple clusters. But it allows you to import only one cluster at a time.

Activate Clusters

When you execute the active cluster command, the `server.yml` of the input model is updated with IP Addresses of compute proxy and OVSvApp.

1. Activate the cluster for the selected vCenter.

   ```
   eon cluster-activate --vcenter-id <vCenter ID> --cluster-moid <Cluster
    Moid>
   ```

Sample Output

```
+---------------
+------------------------------------------------------------------------------------
+
| Property      | Value



                      |
+---------------
+------------------------------------------------------------------------------------
+
| node_info     | {u'computeproxy': {u'pxe-mac-addr':
 u'00:50:56:b6:ce:1b', u'pxe-ip-addr': u'172.170.2.4', u'name':
 u'COMPUTEPROXY_Cluster1', u'cluster-moid': u'domain-c21'},
 u'network_driver': {u'cluster_dvs_mapping': u'DC1/host/Cluster1:hlm-
Trunk', u'Cluster1': [{u'host-moid': u'host-29', u'pxe-ip-addr':
 u'172.170.2.3', u'esx_hostname': u'10.1.200.33', u'ovsvapp_node':
 u'ovsvapp-10-1-200-33', u'pxe-mac-addr': u'00:50:56:b6:5e:9a'}, {u'host-
moid': u'host-25', u'pxe-ip-addr': u'172.170.2.2', u'esx_hostname':
 u'10.1.200.66', u'ovsvapp_node': u'ovsvapp-10-1-200-66', u'pxe-mac-addr':
 u'00:50:56:b6:56:e6'}]}} |
| resource_moid | domain-c21



                      |
| resource_name | Cluster1



                      |
| state         | activated



                      |
+---------------
+------------------------------------------------------------------------------------
+
```

2. Execute the following command to view the list of clusters for the given vCenter.

```
eon cluster-list --vcenter-id <vCenter ID>
```

Sample Output

```
+------------+----------+------------+---------------+
| MOID       | Name     | Datacenter | Import Status |
+------------+----------+------------+---------------+
| domain-c22 | Cluster2 | DC1        | imported  |
+------------+----------+------------+---------------+
```

Modify the Volume Configuration File

Once the cluster is activated you must configure the volume.

Perform the following steps to modify the volume configuration files:

1. Execute the following command:

```
cd /home/stack/helion/hos/ansible/roles/_CND-CMN/templates
```

2. Modify the `cinder.conf.j2` as follows:

```
 # Start of section for VMDK block storage
#
# If you have configured VMDK Block storage for cinder you must
# uncomment this section, and replace all strings in angle brackets
# with the correct values for vCenter you have configured. You
# must also add the section name to the list of values in the
# 'enabled_backends' variable above. You must provide unique section
# each time you configure a new backend.
#
#[<unique-section-name>]
#vmware_api_retry_count = 10
#vmware_tmp_dir = /tmp
#vmware_image_transfer_timeout_secs = 7200
#vmware_task_poll_interval = 0.5
#vmware_max_objects_retrieval = 100
#vmware_volume_folder = cinder-volumes
#volume_driver = cinder.volume.drivers.vmware.vmdk.V
MwareVcVmdkDriver
#vmware_host_ip = <ip_address_of_vcenter>
#vmware_host_username = <vcenter_username>
#vmware_host_password = <password>
#
#volume_backend_name: <vmdk-backend-name>
#
# End of section for VMDK block storage
```

Modify the Neutron Service Configuration File

Perform the following steps to modify the neutron service configuration files:

1. Execute the following command:

```
cd /home/stack/helion/hos/ansible/roles/neutron-common/templates
```

2. Modify the `neutron.conf.j2` with the following values:

```
router_distributed = False
modify ml2_conf.ini.j2
with the below values
[ml2]
mechanism_drivers = ovsvapp, openvswitch, l2population
[agent]
enable_distributed_routing = False
```

Commit your Cloud Definition

1. Add the cloud deployment definition to the git :

```
cd /home/stack/helion/hos/ansible;
git add -A;
git commit -m 'My config';
```

2. Prepare your environment for deployment:

```
ansible-playbook -i hosts/localhost config-processor-run.yml;
ansible-playbook -i hosts/localhost ready-deployment.yml;
cd /home/stack/scratch/ansible/next/hos/ansible;
```

Deploy ESX Compute Proxy and OVSvApps

Execute the following command to deploy a esx compute and OVSvApps:

Note: The variable **esx-ovsvapp** and **esx-compute** must be taken from the **name** key in the `resource-nodes` section in the `/data/control_plane.yml` file (/home/stack/helion/my_cloud/definition/data/control_plane.yml).

```
ansible-playbook -i hosts/verb_hosts guard-deployment.yml
ansible-playbook -i hosts/verb_hosts osconfig-run.yml --limit '*esx-
ovsvapp:*esx-compute'
ansible-playbook -i hosts/verb_hosts hlm-deploy.yml --limit NOV-ESX:NEU-
OVSVAPP
```