Contents

System Requirements for Cisco OpenStack Installer

Supported Hardware and Software

Cisco OSI has been tested against a system that includes Cisco UCS servers. Systems with nonsupported servers might encounter issues.

Cisco OpenStack Installer has been tested on the following infrastructure:

- Cisco UCS C-Series and B-Series Servers serve as physical compute and storage hardware.
- Cisco switches provide physical networking.
- Ubuntu 14.04 LTS (64-bit PC edition) serves as a base operating system.
- KVM serves as the hypervisor.
- OpenStack Neutron provides the network services for the OpenStack cloud. You can select a variety of Neutron setup options, including support for OVS in GRE tunneling mode, OVS in VLAN mode, the Cisco Nexus plugin, and provider networks.

Recommended Release Levels

The following release levels are recommended for any server that is part of an OpenStack cluster:

- For blade servers and integrated rack-mount servers, Cisco UCS Manager, Release 2.1(1) and later releases
- For standalone rack-mount servers, Cisco Integrated Management Controller, Release 1.5 and later releases
- If Cobbler is used for bare-metal Linux installs, Cobbler 2.4 is recommended.

Proxy Configurations

If your network uses proxies, they must be configured properly in order to download the packages used by the installer.

How to configure your proxy is discussed in Creating the Build Node, on page 18.

Minimum Server Requirements

The following table lists the minimum requirements for the Cisco UCS servers that you use for the nodes in your OpenStack cluster:

Server/Node	Recommended Hardware	Notes
Build node	Processor: 64-bit x86	The build node must also have Internet connectivity to be able to download Cisco OSI modules and Puppet

	Server or VM with:	manifests.
	• Memory: 4 GB (RAM) • Disk space: 20 GB	To ensure that the build node can build and communicate with the other nodes in your cluster, it must also have a network interface on the same network as the management interfaces of the other OpenStack cluster servers. A minimal build node (for example, a VM with 4 GB of RAM and a 20-GB disk) is sufficient for a test install. However, because the build node acts as the puppet master, caches client components, and logs all installation activity you might need a more powerful machine with more disk space for larger installs.
Control node	Processor: 64-bit x86 Memory: 12 GB (RAM) Disk space: 1 TB (SATA or SAS) Network: Two 1-Gbps network interface cards (NICs)	A quad core server with 12 GB of RAM is sufficient for a minimal control node. These are minimum requirements. Memory, disk, and interface speed will be greater for larger clusters.
Compute node	Processor: 64-bit x86 Memory: 128 GB (RAM) Disk space: 300 GB (SATA) Volume storage: Two disks with 2 TB (SATA) for volumes attached to the compute nodes Network: Two 1-Gbps NICs	These are minimum requirements. Memory, disk, and interface speed will be greater for larger clusters.
HA proxy (load balance) node	Processor: 64-bit x86 Memory: 12 GB (RAM) Disk space: 20 GB (SATA or SAS) Network: One 1-Gbps NIC	These are minimum requirements. Memory, disk, and interface speed will be greater for larger clusters.
Swift storage proxy node	Processor: 64-bit x86 Memory: 12 GB (RAM) Disk space: 300 GB (SATA or SAS) Network: Two 1-Gbps NICs	These are minimum requirements. Memory, disk, and interface speed will be greater for larger clusters.
Swift storage node	Processor: 64-bit x86 Memory: 32 GB (RAM) Disk space: 300 GB (SATA) Volume storage: • For rack-mount servers, either 24 disks with 1 TB (SATA) or 2 disks with 3 TB (SATA) depending upon the model • For blade servers, two disks with 1 TB (SATA) for combined base OS and storage	Three or more storage nodes are needed. These are minimum requirements. Memory, disk, and interface speed will be greater for larger clusters.

Network: Two 1-Gbps NICs

Choosing a Deployment Scenario

The following table lists the deployment scenarios that are currently available with Cisco OSI:

Scenario Name	Node Count	Roles	Description	Typical Use Case
all_in_one	1	all_in_one 1 all_in_one, compute	A single node that combines the control services and compute services with the build server. Optionally, you can add more compute-only nodes.	Evaluating OpenStack without tying up a lot of hardware. Providing a fully functional (though not scalable or redundant) OpenStack cloud with minimal resources.
2_role	3	build, control, compute, swift_proxy (optional), swift_storage (optional)	Separate nodes (two in addition to the build node) for control and compute. Optionally, one or more nodes for swift storage services. High availability is not possible in this scenario.	Evaluating a multi-node installation with services separated to different machines without the added complexity of messaging, HA, and other production features.
3_role	4	build, control, compute, network_control, swift_proxy (optional), swift_storage (optional)	Same as 2_role, but adds a separate node for network control services that are handled by the control node in 2_role. Optionally, one or more nodes for Swift storage services. High availability is not possible in this scenario.	Evaluating a simplified multi-node installation similar to a 2-role scenario, but with separate networking services.
full_ha	14	build, control, compute, swift_proxy, swift_storage, load_balancer	Similar to the 2_role scenario, but includes a load balancer for high-availability deployment.	Deploying production environments that provide separation of services including dedicated load balancing nodes, storage nodes, compute nodes,and an active/active highly available control

compressed_ha	4	build, compressed_ha, compressed_ha_cephall, compressed_ha_cephosd, compressed_ha_cephmon compressed_ha	A high-availability deployment on three nodes with each node serving all functions. If Ceph is not used, the compressed_ha_cephall, compressed_ha_cephosd and compressed_ha_cephmon roles can be omitted.	plane. Deploying an active/active highly available control plane with a limited number of nodes and limited scalability.

Note: The node count includes the build node.

Note: "Roles" refers to the roles defined in the data model. Each role defines a package of resources required to function in that role.

The following figures illustrate the deployment scenarios listed in the previous table.
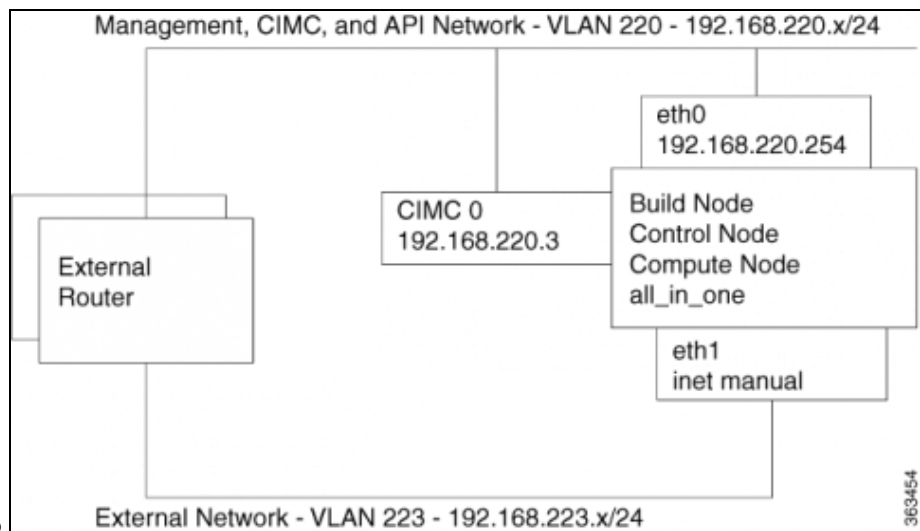
Figure 1: All-In-One Scenario
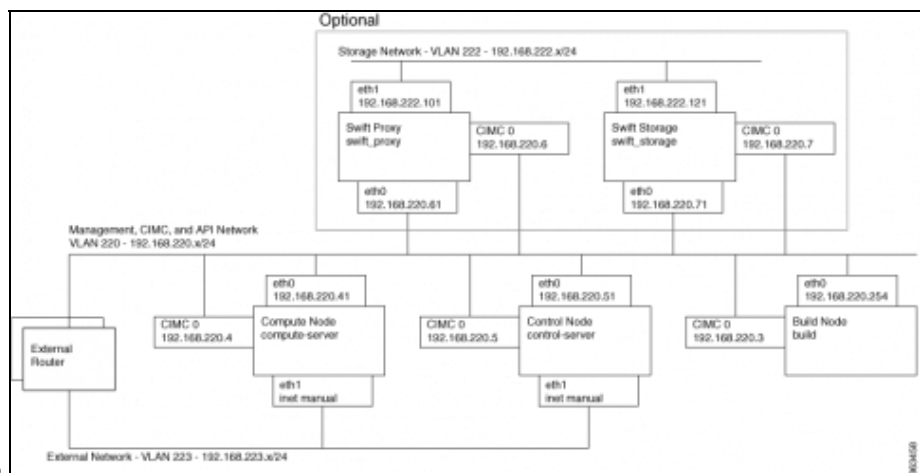
Figure 2: Two-Role Scenario
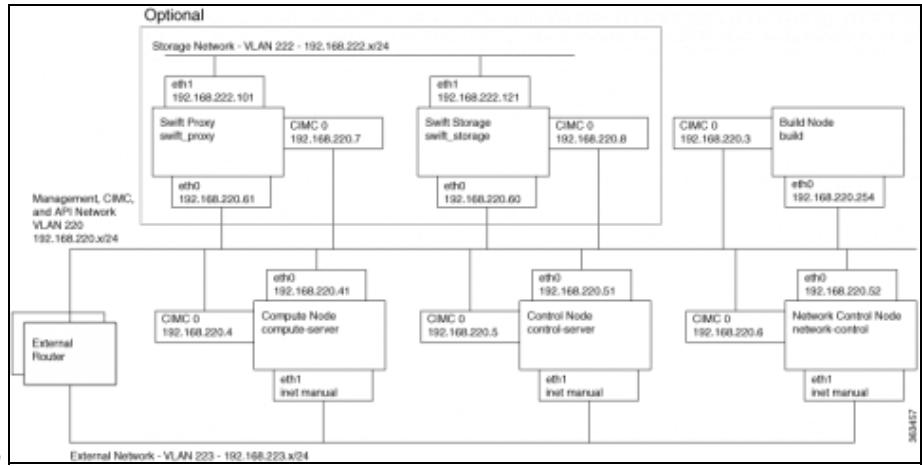
Figure 3: Three-Role Scenario

Figure 4: Compressed High-Availability Scenario
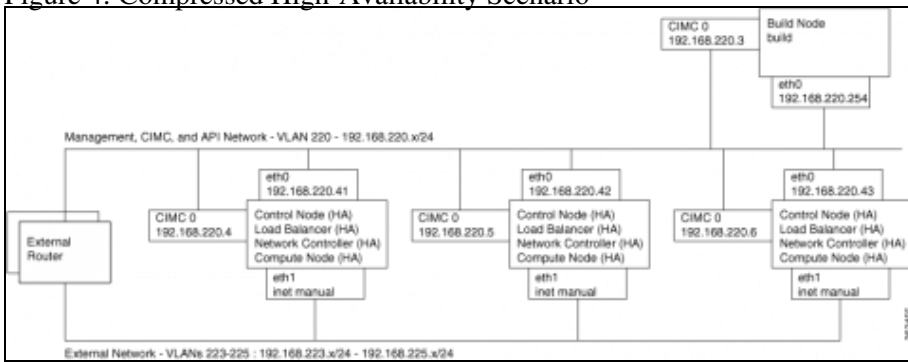
Figure 5: Full High-Availability Scenario

~=~