PHAROS
Release brahmaputra.1.0 (6c30e99)

OPNFV

July 19, 2016

OPNFV COMMUNITY LAB INFRASTRUCTURE

1.1 Pharos Project Information

1.1.1 Introduction

The Pharos Project deals with developing an OPNFV lab infrastructure that is geographically and technically diverse. This will greatly assist in developing a highly robust and stable OPNFV platform. Community labs are hosted by individual companies and there is also an OPNFV lab hosted by the Linux Foundation that has controlled access for key development and production activities. The **Pharos Specification** defines a "compliant" deployment and test environment. Pharos is responsible for defining lab capabilities, developing management/usage policies and process; and a support plan for reliable access to project and release resources. Community labs are provided as a service by companies and are not controlled by Pharos however our goal is to provide easy visibility of all lab capabilities and their usage at all-times.

A requirement of Pharos labs is to provide *bare-metal* for development, deployment and testing. This is resource intensive from a hardware and support perspective while providing remote access can also be very challenging due to corporate IT policies. Achieving a consistent *look and feel* of a federated lab infrastructure continues to be an objective. Virtual environments are also useful and provided by some labs. Jira is currently used for tracking lab operational issues as well as for Pharos project activities.

Future lab capabilities are currently focussed on 1) Deployment automation 2) Dashboards (for capability and usage) 3) *Virtual Labs* for developer on-boarding.

- Pharos page: https://www.opnfv.org/developers/pharos
- Pharos project Wiki: https://wiki.opnfv.org/pharos
- Pharos Planning

1.1.2 Project Communication

- Jira
- Weekly Pharos meeting
- Weekly coordination meeting for Test related projects
- IRC: freenode.net #opnfv-pharos http://webchat.freenode.net/?channels=opnfv-pharos
- Mailing List: use opnfv-tech-discuss and tag your emails with [Pharos] in the subject for filtering

1.1.3 Project Release Artifacts

- Project Repository: https://gerrit.opnfv.org/gerrit/#/q/pharos
- Continuous Integration https://build.opnfv.org/ci/view/pharos/
- Documentation: http://artifacts.opnfv.org/pharos/docs/

1.1.4 Pharos Lab Process

- Process for requesting lab access and support https://wiki.opnfv.org/pharos_rls_b_support
- Pharos Lab Governance and Policies https://wiki.opnfv.org/pharos_policies
- Status of Community labs https://wiki.opnfv.org/pharos_rls_b_labs

1.1.5 Current Labs

An interactive map of OPNFV lab locations, lab owners and other lab information is maintained on the Pharos Wiki

#	Hosted by	Home page	Location
1	**Linux** Foundation	https://wiki.opnfv.org/get_started_lf_hosting	Portland, Oregon
2	Spirent	https://wiki.opnfv.org/pharos/spirenttestlab	Nephoscale, CA
3	China Mobile	https://wiki.opnfv.org/lab2_chinamobile_hosting	Beijing, China
4	Dell	https://wiki.opnfv.org/dell_hosting	Santa Clara, CA
5	Enea	https://wiki.opnfv.org/enea-pharos-lab	Kista, Sweden
6	Ericsson	https://wiki.opnfv.org/get_started_ericsson_hosting	Montreal, Canada
7	Huawei	https://wiki.opnfv.org/lab4_huawei	Xi'an, China
8	Huawei	https://wiki.opnfv.org/get_started_huawei_sc_hosting	Santa Clara, CA
9	Intel	https://wiki.opnfv.org/get_started_intel	Hillsboro, Oregon
10	Orange	https://wiki.opnfv.org/opnfv-orange	Lannion, France
11	Orange	https://wiki.opnfv.org/opnfv-orange	Paris, France
12	ZTE	https://wiki.opnfv.org/zte-nj-testlab	Nan Jing, China
13	**Okinawa** Open Lab	https://wiki.opnfv.org/display/pharos/OOL	Okinawa OOL+TestLab

1.1.6 Pharos project Key Facts

Key Project Facts are maintained in the Pharos INFO file in the project repo

- Can be viewed on the project Wiki INFO
- Project key facts in repo: pharos/INFO

1.2 Pharos Specification

The Pharos Specification provides information on Pharos hardware and network requirements
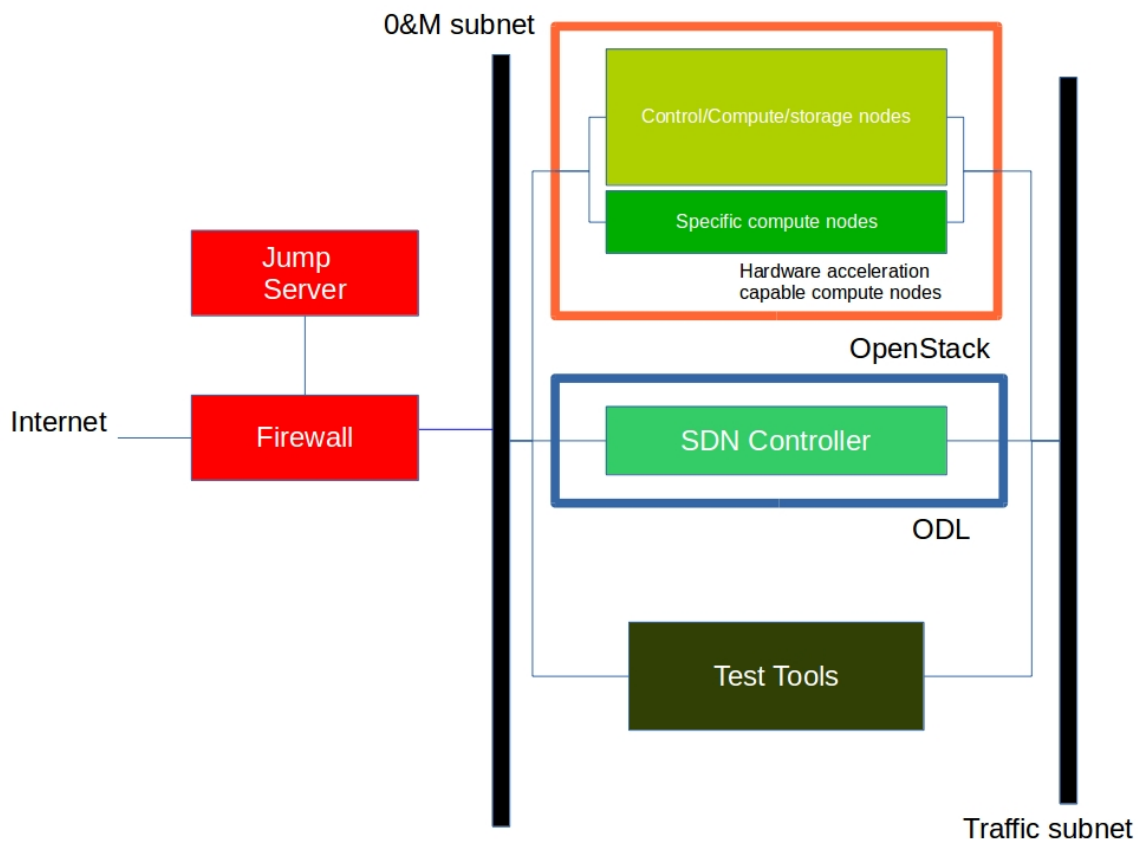
1.2.1 Pharos Compliance

The **Pharos Specification** defines a hardware environment for deployment and testing of the **Brahmaputra** platform release. The **Pharos Project** is also responsible for defining lab capabilities, developing management/usage policies and process; and a support plan for reliable access to project and release resources. Community labs are provided as a service by companies and are not controlled by Pharos however our objective is to provide easy visibility of all lab capabilities and their usage at all-times.

Pharos lab infrastructure has the following objectives: - Provides secure, scalable, standard and HA environments for feature development - Supports the full Brahmaputra deployment lifecycle (this requires a **bare-metal** environment) - Supports functional and performance testing of the Brahmaputra release - Provides mechanisms and procedures for secure remote access to Pharos compliant environments for OPNFV community

Deploying Brahmaputra in a Virtualized environment is possible and will be useful, however it does not provide a fully featured deployment and realistic test environment for the Brahmaputra release of OPNFV.

The high level architecture is outlined in the following diagram:

1.2.2 Hardware

A pharos compliant OPNFV test-bed provides:

- One CentOS 7 jump server on which the virtualized Openstack/OPNFV installer runs

- In the Brahmaputra release you may select a variety of deployment toolchains to deploy from the jump server.

- 5 compute / controller nodes (BGS requires 5 nodes)

- A configured network topology allowing for LOM, Admin, Public, Private, and Storage Networks

- Remote access as defined by the Jenkins slave configuration guide

http://artifacts.opnfv.org/brahmaputra.1.0/docs/opnfv-jenkins-slave-connection.brahmaputra.1.0.html

Servers

CPU:

- Intel Xeon E5-2600v2 Series or newer

Local Storage:

Below describes the minimum for the Pharos spec, which is designed to provide enough capacity for a reasonably functional environment. Additional and/or faster disks are nice to have and mayproduce a better result.

- Disks: 2 x 1TB HDD + 1 x 100GB SSD (or greater capacity)

- The first HDD should be used for OS & additional software/tool installation

- The second HDD is configured for CEPH object storage

- The SSD should be used as the CEPH journal

- Performance testing requires a mix of compute nodes with CEPH (Swift+Cinder) and without CEPH storage

- Virtual ISO boot capabilities or a separate PXE boot server (DHCP/tftp or Cobbler)

Memory:

- 32G RAM Minimum

Power Supply

- Single power supply acceptable (redundant power not required/nice to have)

1.2.3 Networking

Network Hardware

- 24 or 48 Port TOR Switch

- NICs - Combination of 1GE and 10GE based on network topology options (per server can be on-board or use PCI-e)

- Connectivity for each data/control network is through a separate NIC. This simplifies Switch Management however requires more NICs on the server and also more switch ports

- BMC (Baseboard Management Controller) for lights-out mangement network using IPMI (Intelligent Platform Management Interface)
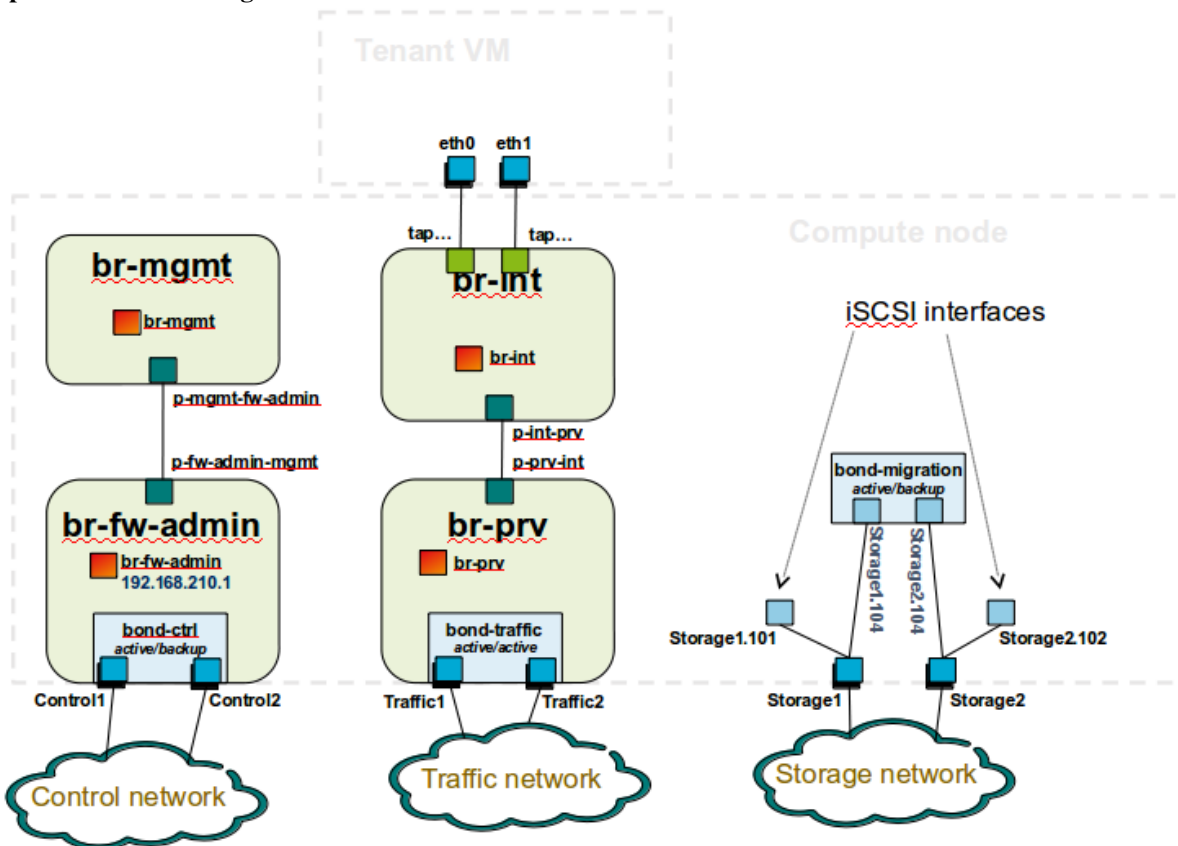
Network Options

- Option I: 4x1G Control, 2x10G Data, 48 Port Switch

  - 1 x 1G for lights-out Management

  - 1 x 1G for Admin/PXE boot

  - 1 x 1G for control-plane connectivity

  - 1 x 1G for storage

  - 2 x 10G for data network (redundancy, NIC bonding, High bandwidth testing)
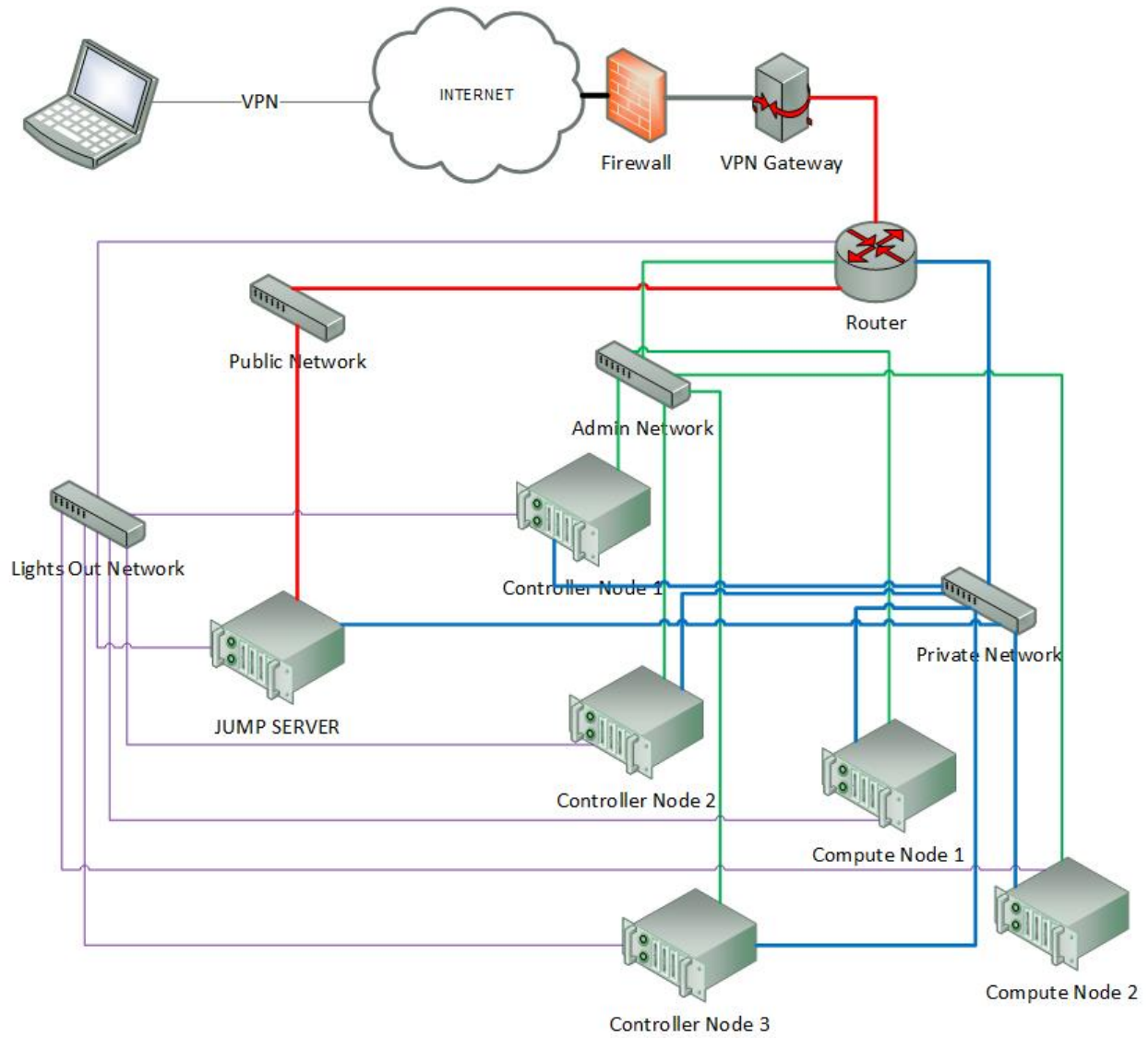
- Option II: 1x1G Control, 2x 10G Data, 24 Port Switch

  - Connectivity to networks is through VLANs on the Control NIC

  - Data NIC used for VNF traffic and storage traffic segmented through VLANs

- Option III: 2x1G Control, 2x10G Data, 2x10G Storage, 24 Port Switch

  - Data NIC used for VNF traffic

  - Storage NIC used for control plane and Storage segmented through VLANs (separate host traffic from VNF)

  - 1 x 1G for lights-out mangement

  - 1 x 1G for Admin/PXE boot

  - 2 x 10G for control-plane connectivity/storage
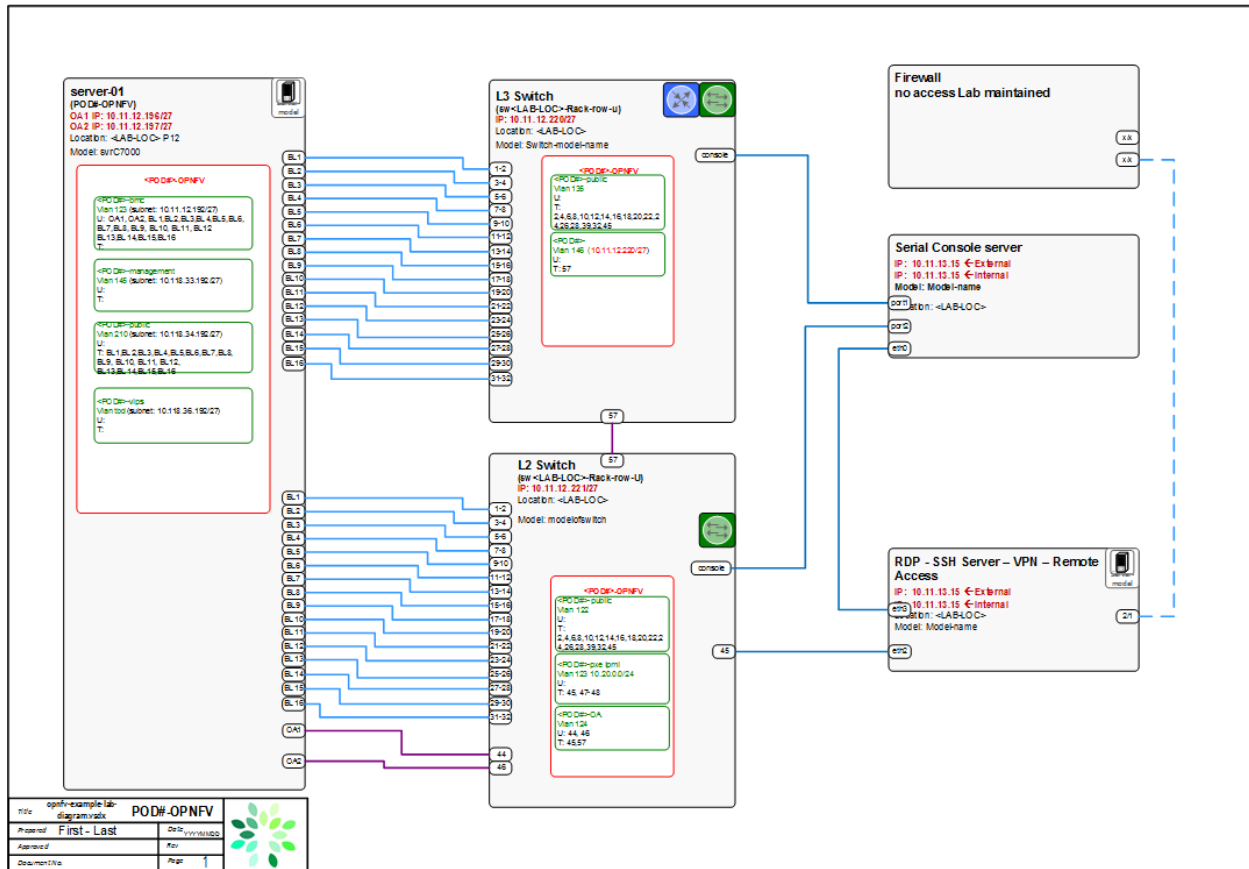
  - 2 x 10G for data network

Documented configuration to include:

- Subnet, VLANs (may be constrained by existing lab setups or rules)

- IPs

- Types of NW - lights-out, public, private, admin, storage

- May be special NW requirements for performance related projects

- Default gateways

Sample Network Drawings

Download the visio zip file here: opnfv-example-lab-diagram.vsdx.zip

1.2.4 Remote Management

Remote access is required for ...

- Developers to access deploy/test environments (credentials to be issued per POD / user)
- Connection of each environment to Jenkins master hosted by Linux Foundation for automated deployment and test

OpenVPN is generally used for remote however community hosted labs may vary due to company security rules. For POD access rules / restrictions refer to individual lab documentation as each company may have different access rules and acceptable usage policies.

Basic requirements:

- SSH sessions to be established (initially on the jump server)
- Packages to be installed on a system (tools or applications) by pullig from an external repo.

Firewall rules accomodate:

- SSH sessions
- Jenkins sessions

Lights-out management network requirements:

- Out-of-band management for power on/off/reset and bare-metal provisioning

- Access to server is through a lights-out-management tool and/or a serial console
- Refer to applicable light-out mangement information from server manufacturer, such as ...
    - Intel lights-out RMM
    - HP lights-out ILO
    - CISCO lights-out UCS

Linux Foundation Lab is a UCS-M hardware environment with controlled access *as needed*

- Access rules and procedure are maintained on the Wiki
- A list of people with access is maintained on the Wiki
- Send access requests to infra-steering@lists.opnfv.org with the following information ...
    - Name:
    - Company:
    - Approved Project:
    - Project role:
    - Why is access needed:
    - How long is access needed (either a specified time period or define "done"):
    - What specific POD/machines will be accessed:
    - What support is needed from LF admins and LF community support team:
- Once access is approved please follow instructions for setting up VPN access ... https://wiki.opnfv.org/get_started/lflab_hosting
- The people who require VPN access must have a valid PGP key bearing a valid signature from LF
- When issuing OpenVPN credentials, LF will be sending TLS certificates and 2-factor authentication tokens, encrypted to each recipient's PGP key

1.3 Pharos Templates and Configuration Files

Lab and POD templates are provided to help lab owners document capabilities, configurations and network topologies. Compute, network and storage specifications with network topology details are required to help developers use lab resources efficiently while minimizing support needs. This also greatly assists with troubleshhoting. It is the responsibility of the lab owner to keep individual lab documents updated and determine appropriate level of detail that is exposed publically through the Wiki or maintained in a secure Pharos repo with controlled access.

While human readable configuration files are needed, the goal is for full automation of deployments. This requires a common machine readable format for POD configurations as input to every installer. This is the "POD inventory" common format file.

1.3.1 Lab Specification Template

Introduction

Add an summary of what your lab hosts, its focus areas and purpose

Lab Resources

POD Name	Project(s)	Project Lead(s)	Email(s)	POD Role	Status	Notes
POD1	project name	John Doe	john@abc.com	CI: stable	Active	

- **POD Name:** Use consistent naming / numbering to avoid confusion. Hyperlinked to POD description.

- **POD Role:** CI stable, CI latest, Dev/test, Stand-alone, Virtual, ...

- **Status:** Assigned, Configuring, Active, Troubleshooting, Available, ...

Acceptable Usage Policy

Define lab user policies and expectations

Remote Access Infrastructure

Describe lab remote access setup (typically VPN, also link speed, any known restrictions, etc.)
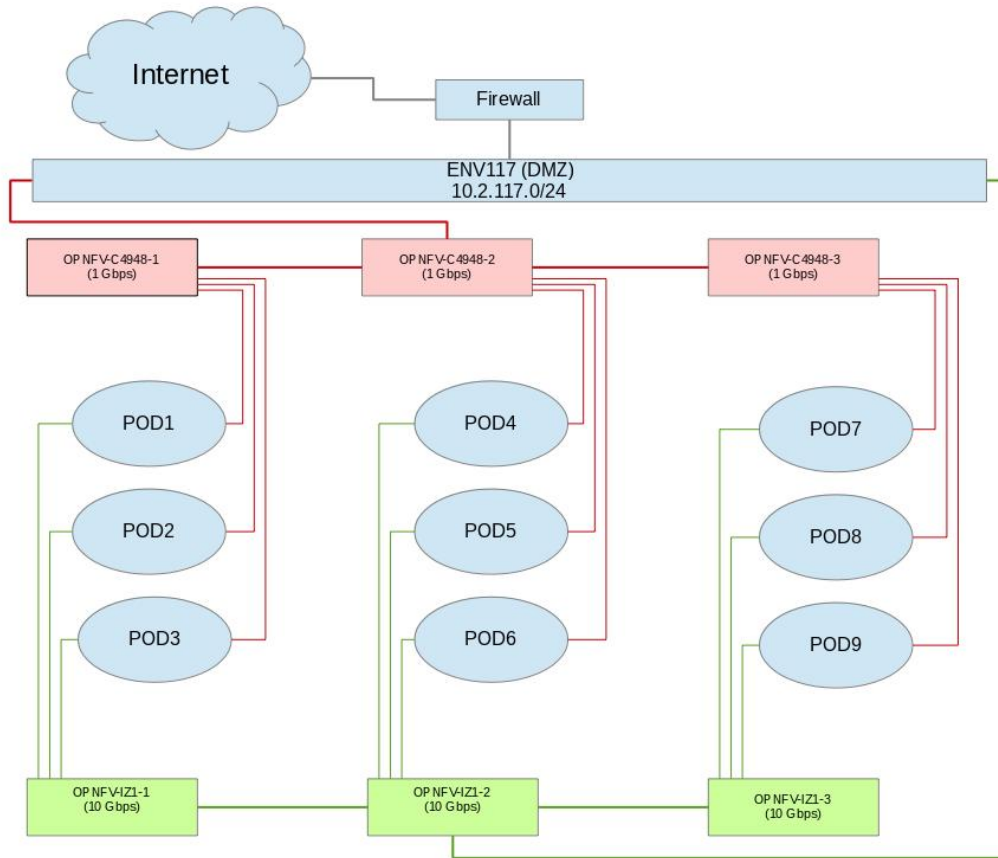
Remote Access Procedure

Define lab process for requesting access to the lab (e.g. VPN guide, how to modify BIOS settings, etc.)

Lab Documentation

List lab specific documents here

Lab Topology

Provide a diagram showing the network topology of lab including lights-out network. Any security sensitive details should not be exposed publically. The following diagram is an example only.

1.3.2 POD Specification Template

Introduction

Add an summary of the POD usage (Project, CI stable, CI latest, dev/test, stand-alone servers, etc.)

Additional Requirements

Describe any addional POD requirements beyond a standard Pharos compliant POD e.g. test equipment, shared usage, ...

Server Specifications

Jump Host

Host-name	Ven-dor	Model	Serial Number	CPUs	Mem-ory	Lo-cal stor-age	Lights-out network (IPMI): IP/MAC, U/P	1GbE: NIC#/IP MAC/VLAN/Network	10GbE: NIC#/IP MAC/VLAN/Network	Notes
pod2-jp-jump	Dell	R730	ABCDEF0054	2x E5-2699	64 GB	240GB SSD 1 TB SATA	10.10.10.10 00:1E:67:D4:06:EA root/root	IF0: 10.2.117.36 00:1E:67:4F:B7:B1 VLAN 984 Public IF1: 10.2.1.1 00:1E:67:4F:B7:B2 VLAN 201 Admin	IF2: 10.2.12.1 00:1E:67:4F:B7:B4 VLAN 202 Private IF3: 10.2.13.1 00:1E:67:4F:B7:B5 VLAN 203 Storage	

Compute Nodes

Host-name	Ven-dor	Model	Serial Number	CPUs	Mem-ory	Lo-cal stor-age	Lights-out network (IPMI): IP/MAC, U/P	1GbE: NIC#/IP MAC/VLAN/Network	10GbE: NIC#/IP MAC/VLAN/Network	Notes
pod2-jp-node1										
pod2-jp-node2										
pod2-jp-node3										
pod2-jp-node4										
pod2-jp-node5										

VPN Users

Name	Email	Project	Role	Notes
joe user	ju@gmail.com	Pharos	contributer	CI support

Firewall Rules

Port(s)	Service	Note
22, 43, 80	Jenkins CI	

POD Topology

Provide a diagram showing the network topology of the POD. Any security sensitive details should not be exposed publically and can be stored in the secure Pharos repo. The following diagram is an example only.

1.3.3 POD Inventory File

The Common Inventory File provides a template for machine reabable input into every installer. For convenience the YAML file template currently resides in the Genesis Project repo. Refer to the following patch for further infomation https://gerrit.opnfv.org/gerrit/#/c/4079.

1.4 Pharos Configuration

OPNFV development, test and production activities rely on Pharos resources and support from the Pharos community. Lab owners and Pharos project committers/contributors will evolve the vision for Pharos as well as expand lab capabilities that are needed to help OPNFV be highly successful.

Pharos configuration documents provide information on how to setup hardware and networks in a Pharos compliant lab. Jira is used to track Pharos activities including lab operations. PODs are connected to Jenkins and generally available 24/7 other than scheduled maintenance and troubleshooting. Lab resources are declared as either for *Development (bare-metal or virtual)*, *Production latest (bare-metal)* or *Production stable (bare-metal)*

1.4.1 Lab Setup Guide

Provides an overview for setting up a Pharos lab. A full set of *PHAROS* documents are maintained in the *pharos* repo.
When setting up an OPNFV community lab ...

- Provide the Pharos community with details of the intended setup, including ...
    - Overview of resources are being offered to the community, intended purpose and known limitations
    - Lab owner name with contacts
    - Timelines for availablity for development, test, release production, ...
- Update the Pharos Wiki with lab details
    - Lab map, organization, contacts, status, location, resources, role, etc.
    - https://wiki.opnfv.org/pharos#community_labs
    - *Updating Pharos Documents*
- Update the Pharos project information file "Current Labs"
    - *Pharos Project Information*
- Create new Wiki pages for lab and POD specific information
    - Access procedures
    - Usage guidelines for developers
    - Update infomtation as PODs are re-assigned or usage/availability changes
- Fill Lab and POD templates ... *Lab Specification Template* ... *POD Specification Template*
    - Note that security sensitive lab information should be stored in the secure Pharos repo
- Connect PODs to Jenkins/CI

1.4.2 Updating Pharos Documents

Details about each Community Lab is found in 3 places:

- Summary of lab including location, contacts, status, etc. on the Pharos Project Wiki page
- Lab specific details are provided with dedicated Wiki pages, see this Example Lab
- Pharos repo docs ...
    - docs/information/pharos.rst ... project information file
    - docs/labs/ ... Lab documents (includes lab specific capabilities, usages and policies; POD information)
    - docs/labs/images/ ... Lab and POD toplogies

Update Pharos repo

Clone the Pharos Git repository

- Make the changes to Pharos project information file (docs/information/pharos.rst)
- After code gets merged http://artifacts.opnfv.org/pharos/docs/information/pharos.html will contain your change

Update Pharos Wiki

Edit Wiki page

- https://wiki.opnfv.org/pharos

- Look for {{scrape>http://artifacts.opnfv.org/pharos/docs/information/pharos.html}}

- Click "Preview" and see if your change is shown; if shown add a short "Edit summary" and click "Save" (Wiki does not auto update content)

You will see a section of code as shown below. Add your page to the bullet list with wiki link, nice name, and location summary

Update the map info on the Pharos Project Page https://wiki.opnfv.org/pharos?&#community_labs

- You will see a section of code as shown below. Add your lab infomation to the list with a comma separated list as follows:

    - longitude

    - latitude

    - .8 <- for size

    - marker color png ([[marker-green.png|marker-green.png]], [[marker-blue.png|marker-blue.png]], [[marker-red.png|marker-red.png]], [[marker-gold.png|marker-gold.png]])

    - Nice Format Lab Name

    - ';'

    - Location Summary

    - ''\'' <– for a new line

    - external link: <– optional

MAP:

```
<olmap id="olMapOne" width="877px" height="200px" lat="45.0" lon="0.0" zoom="3" statusbar="1" toolbar
poihoverstyle="0" baselyr="OpenStreetMap" gpxfile="" kmlfile="">
45.52,-122.67,60,.8,marker-red.png,Linux Foundation;Portland, Oregon \\ external link: [[http://www.t
39.7392,-104.9902,60,.8,marker-red.png,Cable Labs;Denver, CA \\ external link: [[http://www.test.com
37.333685,-121.891272,60,.6,marker-green.png,[[pharos/spirentvctlab|Spirent VCT Lab]] \\ San Jose, Ca
39.90,116.35,60,.8,marker-red.png,China Mobile Labs;Beijing, China \\ external link: [[http://www.tes
37.413137,-121.977975,-180,.6,marker-red.png,Dell Labs;Santa Clara, California \\ link: [[https://wi
59.41,17.95,60,.8,marker-red.png,Enea Pharos Lab;Kista, Sweden \\ external link: [[http://www.enea.co
45.50,-73.66,60,.8,marker-blue.png,Ericsson Labs;Montreal, Canada \\ external link: [[http://www.test
34.26,108.97,60,.8,marker-green.png, Huawei Labs;Xi an, China \\ external link: [[http://www.test.com
37.373424,-121.964913,60,.8,marker-green.png, Huawei Labs;Santa Clara, USA \\ external link: [[http:/
45.53,-122.97,60,.8,marker-green.png,Intel Labs;Hillsboro, Oregon \\ link: [[https://wiki.opnfv.org/g
48.75867,-3.45196,60,.8,marker-gold.png,Orange Labs;Lannion, France \\ external link: [[http://www.te
48.825786,2.274797,-60,.8,marker-gold.png,Orange Labs;Paris, France \\ external link: [[http://www.te
31.97,118.79,60,.8,marker-red.png,ZTE Labs;Nan Jing, China \\ link:[[zte-nj-testlab|ZTE, Nan Jing]]
[[http://test.com|test.com]] \\ internal link: [[::start]]\\ **DW Formatting**
</olmap>
```

1.4.3 Jump Server Configuration

Jump server install procedures are maintained by each installer project. Addional Jump server configuraton BKMs will be maintained here. The below install information was used for Fuel however may be outdated (please refer to Fuel Installer documents).

Procedure

1. Obtain CentOS 7 Minimal ISO and install

   ```
   wget http://mirrors.kernel.org/centos/7/isos/x86_64/CentOS-7-x86_64-Minimal-1503-01.iso
   ```

2. Set parameters appropriate for your environment during installation

3. Disable NetworkManager

   ```
   systemctl disable NetworkManager
   ```

4. Configure your /etc/sysconfig/network-scripts/ifcfg-* files for your network

5. Restart networking

   ```
   service network restart
   ```

6. Edit /etc/resolv.conf and add a nameserver, for example 8.8.8.8

   ```
   echo nameserver 8.8.8.8 >> /etc/resolv.conf
   ```

7. Install libvirt & kvm

   ```
   yum -y update   yum -y install kvm qemu-kvm libvirt   systemctl enable
   libvirtd
   ```

8. Reboot:

   ```
   shutdown -r now
   ```

9. Configure SSHD

   If you wish to avoid annoying delay when use ssh to log in, disable DNS lookups:

   When **UseDNS** is existed in the config file, update it:

   ```
   sed -i -e 's/^#*UseDNS\ \+yes/UseDNS no/' /etc/ssh/sshd_config
   ```

   or append the setting when not existed:

   ```
   echo UseDNS no >> /etc/ssh/ssd_config
   ```

   Disable Password Authenticaion for security:

   ```
   sed -i -e 's/^#PasswordAuthentication\ \+yes/PasswordAuthentication
   no/' /etc/ssh/sshd_config
   ```

   If you want to disable IPv6 connections, comment IPv6 ListenAddress and change AddressFamily to inet:

   ```
   sed -i -e 's/^ListenAddress\ \+::/#ListenAddress ::/'
   /etc/ssh/sshd_config  sed -i -e 's/^AddressFamily\ \+any/AddressFamily
   inet/' /etc/ssh/sshd_config
   ```

10. Restart sshd

    ```
    systemctl restart sshd
    ```

11. Install virt-install

    ```
    yum -y install virt-install
    ```

12. Visit artifacts.opnfv.org and D/L the OPNFV Fuel ISO

13. Create a bridge using the interface on the PXE network, for example: br0

    ```
    brctl addbr br0
    ```

14. Make a directory owned by qemu:

```
mkdir /home/qemu; mkdir -p /home/qemu/VMs/fuel-6.0/disk

chown -R qemu:qemu /home/qemu
```

15. Copy the ISO to /home/qemu

```
cd /home/qemu

virt-install -n opnfv-2015-05-22_18-34-07-fuel -r 4096
--vcpus=4 --cpuset=0-3 -c opnfv-2015-05-22_18-34-07.iso
--os-type=linux --os-variant=rhel6 --boot hd,cdrom --disk
path=/home/qemu/VMs/mirantis-fuel-6.0/disk/fuel-vhd0.qcow2,bus=virtio,size=50,format=q
-w bridge=br0,model=virtio --graphics vnc,listen=0.0.0.0
```

16. Temporarily flush the firewall rules to make things easier:

```
iptables -F
```

17. Connect to the console of the installing VM with your favorite VNC client.

18. Change the IP settings to match the pod, use an IP in the PXE/Admin network for the Fuel Master

TWO

INDICES

- search