



Deploying Cisco Smart PHY

This chapter provides information about deploying the Cisco Smart PHY software product package in an offline environment (without Internet connectivity).

- [Offline Deployment Overview, on page 1](#)
- [Prerequisites for Deployment, on page 3](#)
- [Preparing the Staging Environment, on page 9](#)
- [Preparing Cluster Configuration File, on page 11](#)
- [Deploying the Deployer VM and Cisco Smart PHY Cluster, on page 19](#)
- [Auto Deployer User Interface, on page 23](#)
- [Configuring Dual Stack on External Cluster Interfaces, on page 32](#)
- [Deployment Limitations, on page 33](#)
- [Smart PHY Client Requirements, on page 33](#)

Offline Deployment Overview

Feature History

Feature Name	Release information	Description
Support for SMI path based URL routing in the Deployer VM.	Cisco Smart PHY 22.3	Deployer VM supports path based URL. Smart PHY only needs two DNS entries as a prerequisite for installation. <ul style="list-style-type: none">• cluster.example.com (DNS entry for Cluster)• deployer.example.com (DNS entry for Deployer)

Cisco Smart PHY supports deployment in an offline operator-managed vSphere virtualization environment.

You can download the Smart PHY software package as a compressed file from the Cisco.com website. The software package contains instructions, sample cluster configuration files, a cluster deployment tool, and the Smart PHY software.

Deployment Components

- **Deploy tool**—A deployment automation tool that controls the deployment of an Smart PHY cluster.

- [Staging Environment, on page 3](#)—A desktop operating system or virtual machine that meets the requirements to run the deploy tool.
- [Cluster Configuration, on page 4](#) file—An YAML-formatted text file that contains the Smart PHY cluster configuration, including the vSphere configuration, and “Deployer” VM configuration. An 'admin' user creates the configuration file.
- [Deployer VM](#)—The deploy tool instantiates the Deployer VM. This virtual machine hosts the software, container image, and VM image repositories needed to complete an offline Smart PHY cluster deployment.

Deployment Overview

At a high level, deploying the Cisco Smart PHY cluster consists of the following steps:

1. (Optional) Configuring UCS Servers for Hosting Smart PHY. This step is only required when deploying on to Cisco UCS servers dedicated to the cluster.
2. Preparing the Staging Environment
3. Creating the cluster Configuration File
4. Executing the deploy tool. For more information, see [Deploying the Deployer VM and Cisco Smart PHY Cluster, on page 19](#). To deploy another cluster, repeat creating the configuration file and deploying the cluster procedures.

The deploy tool, which is run from your staging environment, reads the Smart PHY cluster configuration from the specified cluster configuration file. After validating the values from the cluster configuration file, the deploy tool instantiates a “Deployer” VM in your designated vSphere environment.

Once the “Deployer” VM boots completely, the deploy tool syncs the Smart PHY cluster configuration to a software agent running on the “Deployer” VM. The agent executes the following Smart PHY cluster operations:

- Copying cluster VM images to the vSphere datastore
- Instantiating cluster VM
- Configuring Guest OS
- Installing and configuring container orchestrations software
- Finally, launching Smart PHY’s containerized micro-services

Deployment Types

The deploy tool can create two types of Cisco Smart PHY clusters:

- All-in-one (AIO) cluster—Runs as a single VM on an ESXi host.
 - AIO clusters are best suited for labs and small production environments where high availability is not required.
- Multinode cluster—Consists of 12 VMs deployed across three ESXi hosts.
 - Each ESXi host runs one instance of these four VMs: control-plane, etcd, Infra, and Operations.

- Multinode clusters provide high availability and continues to operate even after a failure of one ESXi host.
- Two multimode cluster sizes are available:
 - Small—Best suited for labs and small production environment. This is the default deployment size when no value is specified in the cluster configuration file.
 - Normal—Best suited for large production environment.

Prerequisites for Deployment

This section provides details on prerequisites that must be met before deploying a Cisco Smart PHY cluster. The following resources are required to deploy, operate, and manage the Cisco Smart PHY cluster.

Related Topics

[Staging Environment](#), on page 3

[Domain Name System](#), on page 3

[Cluster Configuration](#), on page 4

[VMware vSphere](#), on page 6

[VMware ESXi Host Running the Deployer VM](#), on page 7

[VMware ESXi Hosts Running Smart PHY Cluster VMs](#), on page 7

[Connectivity](#), on page 8

[NTP Server](#), on page 9

Staging Environment

The staging environment is any Operating System or virtual machine with:

- High-speed, low latency connectivity to the vSphere environment
- At least 50GB of free disk

Prerequisites for Staging Environment

The following software must be installed:

- UNIX compatible shell
- Docker 18.09.7 or later
- Python 3.6 or later

Domain Name System

You can assign a Fully Qualified Domain Names (FQDN) to both the Smart PHY cluster and Deployer VM so that these resources function properly.

Two types of FQDN are available:

1. User-Specified FQDNs (Recommended)
2. Autogenerate FQDNs

User-Specified FQDN

User-Specified FQDN enables you to specify the hostname for both the Smart PHY cluster and deployer VM. You can specify your preferred cluster and deployer VM FQDNs using the “ingress-hostname” key-value pair (optional parameter) in the cluster configuration file.

Prerequisites for User-Specified FQDN

- Supports only alphanumeric characters in the FQDN.
- Unique FQDNs must be assigned to the cluster and deployer VM.

Ensure that you configure the corresponding DNS records (listed below) correctly in your authoritative DNS server before conducting a deployment with your specified FQDNs. If the records are not resolved correctly, then your cluster deployment fails.

Required DNS Records by entity:

- Operations Hub cluster:
 <cluster-fqdn>
- Deployer VM:
 <deployer-vm-fqdn>

For example, if the `ingress-hostname` value in the **clusters** section of your configuration file is `opshub.example.com`, then the required DNS records is `opshub.example.com`.

Autogenerate FQDN

You can trigger autogeneration of an FQDN by omitting the optional parameter “ingress-hostname” key-value pair from the relevant sections of the cluster configuration file.

The deploy tool autogenerates FQDNs by combining the entities’ management IP address, specified in the cluster configuration file, with the “nip.io” domain name.

For example, if 10.0.22.22 is assigned to the cluster management VIP and the `ingress-hostname` key-value pair is omitted, the autogenerated cluster FQDN will be `10.0.22.22.nip.io`.

Prerequisites for Autogenerate FQDN

- Ensure that your DNS servers resolve the nip.io domain properly. If nip.io resolution is blocked, then your cluster deployment fails.

Cluster Configuration

Feature History

Feature Name	Release Information	Description
vSphere Data Center Folder Path support	Cisco Operations Hub, Release 22.4	Operations Hub’s Autodeployer supports the ability to deploy Ops Hub clusters into a defined vSphere datacenter folder path.

You need the following information to prepare the cluster configuration file.

vSphere Environment

Collect or prepare the following information:

- vCenter server hostname or IPv4 address
- vCenter credentials (username and password)
- vCenter Datacenter
- vCenter Datacenter folder path (optional)
- vCenter cluster name
- vCenter networks
- Datastore name
- Datastore folder path (optional)
- ESXi hostnames
- DNS hostname or IPv4 addresses
- Search domains
- NTP hostnames or IPv4 addresses
- HTTPS Proxy Server IPv4 address (if required)
- Environment name (This name is referenced in the cluster configuration file)

Deployer VM

Collect or prepare the following information:

- Guest OS management network:
 - One IPv4 Address
 - Subnet mask in CIDR notation
 - Gateway address
- Ingress Hostname (Optional, but recommended. For more information, see [Domain Name System, on page 3](#).)
- Username (you'll need to choose a username)
- Deployer name (This name is referenced in the cluster configuration file)

Smart PHY Cluster

Collect or prepare the following information:

- Cluster & Guest OS Management network:
 - Cluster
 - Virtual Router Redundancy Protocol (VRRP) ID

- One IPv4 Virtual IP Address (VIP)
- Subnet mask in CIDR notation
- Gateway address
- Guest OS (Must be in the same IPv4 subnet as the cluster management network):
 - AIO: One IPv4 Address
 - Multinode: 12 IPv4 Addresses. (1 address for the Guest OS on each of the 12 cluster VMs)
- Ingress Hostname (Optional, but recommended. For more information, see [Domain Name System, on page 3](#).)
- Deployment size (small or normal)
- Username (you must choose an username)

To prepare the cluster configuration file, see [Cluster Configuration File, on page 11](#).

VMware vSphere

VMware vSphere is the only virtualization environment where the Smart PHY clusters are supported.

Supported Hypervisors

- VMware ESXi 7.0
- VMware ESXi 8.0

Supported ESXi Host Management

- VMware vCenter Server 7.0.3
- VMware vCenter Server 8.0.1

If VMware ESXi 7.0 is installed on the host, then ensure that the VMware vCenter Server version is also 7.0. If VMware ESXi 8.0 is installed on the host, then ensure that the VMware vCenter Server version is also 8.0.1.



Note We recommend that you use VMware vCenter Server 7.0 with VMFS 6 Datastore type.

Datastore Cleanup in ESXi Hosts

The following tasks are applicable for a new deployment of Smart PHY.

- Before deploying SmartPHY, if there are any existing instances of inception server and cluster VM instances, then power off those VMs and use the **Delete from disk** option to delete them. When it is deleted from the host, go to the datastore folder of the host or hosts, and delete the data that is related to the inception and cluster.

- Starting with Cisco Smart PHY 23.1 Release, delete the **smi-base-image** folders in the datastore before deployment, as the Kubernetes version is upgraded.

Limitation of VM Movement Post Cluster Installation

Post the Cluster installation, if VMs are moved to a different VMware environment, then it leads to an improper cluster configuration in the deployer VM. Hence, upgrading the cluster or RMA actions on the cluster won't work through the deployer VM.

VMware ESXi Host Running the Deployer VM

One ESXi host is required to run the “Deployer” VM. The “Deployer” VM must not be co-located on the ESXi hosts running cluster VMs.

Prerequisites for VMware ESXi Host Running the Deployer VM

Ensure that the VMware ESXi host has the recommended capacity for compute, storage, and networking which are listed in the following table:

Table 1: Minimum System Requirements - ESXi Hosts

Parameter	Value
Processor	8 vCPUs
Memory	16 GB
Storage	320 GB Minimum 50,000 IOPS (Input/output operations per second) Latency of < 5 ms
NIC	10G vNIC

VMware ESXi Hosts Running Smart PHY Cluster VMs

Three ESXi hosts are required to run a Cisco Smart PHY multinode cluster. Cluster VMs must not be co-located on the ESXi host running the “Deployer” VM.

Prerequisites for VMware ESXi Hosts Running Smart PHY Cluster VMs

Ensure that the VMware ESXi host has the recommended capacity for compute, storage, and networking which are listed in the following table:

Table 2: Minimum System Requirements - ESXi Hosts

Cluster Size	Small	Normal
Processor	20 vCPUs	34 vCPUs
Memory	160 GB	304 GB

Cluster Size	Small	Normal
Storage	1640 GB Minimum 50,000 IOPS (Input/output operations per second) Latency of < 5 ms	2440 GB Minimum 50,000 IOPS (Input/output operations per second) Latency of < 5 ms
NIC	2x 10G vNIC	2x 10G vNIC

The following tables show the minimum requirements for AIO cluster:

AIO

VM Type	CPU Cores	RAM Size (GB)	SSD Storage Size (GB)
AIO	18	96	1541

The following tables show the minimum requirements for each of the VM types deployed in a multimode cluster:

Small Multimode Cluster

VM Type	CPU Cores	RAM Size (GB)	SSD Storage Size (GB)
Control Plane	2	16	125
etcd	2	16	125
infra	8	64	1000
ops	8	64	320

Normal Multimode Cluster

VM Type	CPU Cores	RAM Size (GB)	SSD Storage Size (GB)
Control Plane	2	16	125
etcd	2	16	125
infra	14	96	1500
ops	16	176	620

Connectivity

From the Staging environment, the deploy tool must have connectivity to the following resources:

- Local DNS server
- vCenter server
- NTP server
- All ESXi hosts

- IPv4 subnet that is assigned to the Guest OS management network on the “Deployer” VM
- IPv4 subnet that is assigned to the Cluster VIP and Guest OS management network on the cluster VMs

The “Deployer” VM, when created, must have connectivity to the following resources:

- Local DNS server
- vCenter server
- IPv4 subnet assigned to the Cluster VIP and Guest OS management network on the cluster VMs

NTP Server

Ensure that the clocks for the staging server, Deployer VM, and Cluster VM are in sync, preferably pointing to the same NTP server.

Preparing the Staging Environment

This section provides details on how to prepare your staging environment for a Smart PHY cluster deployment.

Preparing the staging environment involves the execution of the following procedures

- [Transferring the Smart PHY release package to your Staging Environment, on page 9](#)
- [Extracting Installation files and Performinng Signature Verification, on page 9](#)

Transferring the Smart PHY release package to your Staging Environment

Step 1 Download the Smart PHY release package (`smartphy-installer-<version>.SPA.tgz`) from Cisco.com. The package is approximately 15 GB.

Step 2 Securely copy the release package to your Staging Environment.

Extracting Installation files and Performinng Signature Verification

The commands listed in this procedure should be executed from the shell in your Staging Environment.

Step 1 Run the `tar` command against the `smartphy-installer-<version>.SPA.tgz` release package to extract the installation files.

Example:

```
tar -zxovf smartphy-installer-<version>.SPA.tgz
```

The following files are extracted:

- `smartphy-installer-<version>.tgz`
- `smartphy-installer-<version>.tgz.signature`

- cs-verify.sh
- SMART_PHY_REL_KEY-CCO_RELEASE.cer
- signed_files

Step 2 Run the `cs-verify.sh` script to verify the signature of the `smartphy-installer-<version>.tgz` installer image.

Example:

```
./cs-verify.sh SMART_PHY_REL_KEY-CCO_RELEASE.cer smartphy-installer-<version>.tgz
```

Example output:

```
Verifying signature
```

```
Signature verification succeeded
```

If the signature verification fail, then delete the previously extracted installation files and the installer image. Re-download the Smart PHY release package from Cisco.com and start this procedure again.

Step 3 Run the `tar` command against the extracted `smartphy-installer-<version>.tgz` installer image to create the staging directory.

Example:

```
tar -zxovf smartphy-installer-<version>.tgz
```

A new directory named `smartphy-installer-<version>` has been created. This directory is known as the staging directory.

Step 4 Navigate to the `smartphy-installer-<version>` staging directory.

Example:

```
cd smartphy-installer-<version>
```

The staging directory `smartphy-installer-<version>` contains the following files and folders:

```
:
```

```
smartphy-installer-<version>
├── README.md
├── cluster-deployer-<version>.tar
├── cluster-deployer-<version>.tar.signature
├── deploy
├── deploy.signature
├── docker-images
│   ├── ccmts-customization_<version>.tar
│   └── ccmts-customization_<version>.tar.signature
├── examples
│   ├── aio-smartphy-config.yaml
│   ├── aio-smartphy-standby-config.yaml
│   ├── deployer-sample-config.yaml
│   ├── multinode-smartphy-config.yaml
│   └── multinode-smartphy-standby-config.yaml
├── offline-products
│   ├── cee-<version>.tar
│   ├── cee-<version>.tar.signature
│   ├── opshub.tar
│   ├── opshub.tar.signature
│   ├── smartphy-<version>.tar.signature
│   └── smartphy-<version>.tar
└── smi-install-disk.iso
```

```
|— smi-install-disk.iso.signature
|— upgrade-prep
|— upgrade-prep.signature
|— utility-images
   |— autodeploy_<version>.tar
   |— autodeploy_<version>.tar.signature
   |— cluster-manager-docker-deployer_<version>.tar
   |— cluster-manager-docker-deployer_<version>.tar.signature
```

Preparing Cluster Configuration File

This section provides info on how to create the Cluster Configuration file required by Smart PHY's deploy tool. This configuration file stores all of the information the deploy tool needs in order to deploy your Smart PHY cluster.

Sample Configuration Files

During the extraction of the Smart PHY release package an `example` directory is created under the staging directory. The `example` directory contains the following sample configuration files:

- `deploy-sample-config.yaml`—A configuration file with a deployer, but no cluster.
- `aio-smartphy-config.yaml`—A configuration file with a deployer and a single-node Smart PHY cluster.
- `multinode-smartphy-config.yaml`—A configuration file a deployer and multinode Smart PHY cluster.
- `aio-smartphy-standby-config.yaml`—A configuration file with a deployer and a single-node Smart PHY cluster that is configured for standby (without CIN configuration).
- `multinode-smartphy-standby-config.yaml`—A configuration file with a deployer and a multinode Smart PHY cluster configured standby (without CIN configuration).

Cluster Configuration File

Place the configuration file in the staging directory. This configuration file is in the standard YAML language format, with the following three sections:

- Environments
- Deployers
- Clusters (Smart PHY multi-node/single-node)

Each section can contain multiple items. Replace `<...>` with actual values.



Note Starting with Cisco Smart PHY 23.3,

- The values for Boolean parameters such as **enable-http-redirect** and **ipv6-mode**, must be enclosed in double quotes. The accepted values are **"true"** or **"false"**.
- IPv6 address parameter values must be enclosed in double quotes. For example: **"2001:18:208::/64"**.

In previous Cisco Smart PHY releases, the usage of double quotes is optional.

Environment Configuration

The `environments` section defines a vSphere deployment domain. This environment is referenced in the `deployers` and `clusters` sections, which you define shortly.

```
environments:
  <env-name>:
    server: <value>           # vCenter Server IPv4 Address or name
    username: <value>        # vCenter username, user will be prompted for
                             # the password
    datacenter: <value>      # vCenter Datacenter Name
    datacenter-path: <value> # vCenter Datacenter folder path (optional). When
                             # omitted cluster VMs are created under root of the
                             #
                             # Datastore. For nested folders, the folder name
                             # should be separated by / symbol
    cluster: <value>         # vCenter Cluster Name
    nics: [ <list> ]         # List of vCenter nics (port groups)
    nameservers: [ <value> ] # List of DNS Server IPv4 Addresses
    search-domains: [ <value> ] # List of Search domains
    ntp: [ <list> ]         # List of NTP Server IPv4 Addresses or name
    https-proxy: <value>    # Optional HTTPS Proxy
                             # (Ex: http://proxyhost.domain.tld:port)
    no-proxy: <value>       # Optional HTTPS Proxy bypass
```

Guidelines for Defining an Environment

- The environment name can have only lowercase letters, digits, and hyphens (-).
- The `nics` list must contain only one network, although the `nics` configuration allows multiple networks. This network is used as the management network by the deployer or cluster that refers to this environment.
- Create multiple `environments` if your vCenter has more than one network that serves as a management network. Create one `environments` for each network. In addition, refer to the corresponding `environments` in the deployer or cluster based on the management network it uses.
- Make sure the `nics`, `nameservers`, and `search-domains` values are structured as YAML lists.

Deployer Configuration

The `deployers` section defines the configuration of a Deployer VM.

```
deployers:
  <deployer-vm-name>:
    environment: <value> # Environment name defined in the 'Environments'
                         # section.
```

```

address: <value>           # Deployer VM IPv4 Address in CIDR format
gateway: <value>           # Deployer VM's Gateway IP Address
ingress-hostname: <value>  # Optional FQDN (Ex. "deployer1.example.com")
username: <value>          # Deployer VM username, you are prompted for
                           # the password
private-key-file: <value>  # Optional SSH private-key-file (.pem) with
                           # path relative to the staging directory. Key is
                           # auto-generated, if one if not provided

host: <value>              # IPv4 Address of the ESXi Host
datastore: <value>         # Datastore for the Deployer VM
datastore-folder: <value> # Optional Datastore folder path. When omitted,
                           # deployer VM is created under root of the
                           # Datastore. For nested folders, the folder
                           # name should be separated by / symbol

docker-subnet-override:   # 'docker-subnet-override' and its values are
                           # optional. It should only be used when you need
                           # to customize the deployer VM's Docker
                           # IP Addressing. When omitted, the Docker bridge
                           # defaults to 172.17.0.0/16
- pool-name: <value>      # Docker bridge address pool name
  base: <value>           # Docker bridge subnet in CIDR format
  size: <value>          # Docker bridge subnet size: 8-24

```

Guidelines for Defining a Deployer

- The `deployer-vm-name` can have only lowercase letters, digits, and hyphens (-).
- The `private-key-file`, when present, must refer to the SSH private key file (.pem). This file must be in the staging directory and must not be accessible (read/write/execute) to other users.
- If the `private-key-file` line is omitted, the deploy tool generates an SSH private key for the deployer and places it in the `.sec` subdirectory under the staging directory. The filename is `<deployer-vm-name>_auto.pem`.
- The values associated with `docker-subnet-override` are optional. Those values should only be included in the configuration when you need to customize the deployer VM's Docker IP addressing. When omitted, the Docker bridge on the Deployer VM defaults to 172.17.0.0/16.
- To avoid resource contention, do not host the deployer VM on the same ESXi hosts running any of the cluster VMs.
- When you configure a custom `ingress-hostname`, ensure that the following entries are in the DNS:

```

<host.domain.tld>
charts.<host.domain.tld>
files-offline.smi-cluster-deployer.<host.domain.tld>
deployer-ui.smi-cluster-deployer.<host.domain.tld>
cli.smi-cluster-deployer.<host.domain.tld>
restconf.smi-cluster-deployer.<host.domain.tld>
docker.<host.domain.tld>

```

Cluster Configuration

Table 3: Feature History

Feature Name	Release Information	Description
Single NIC for NB and SB communication for Multi-node cluster	Cisco Smart PHY, Release 24.1.1	You can optionally deploy an All in one (AIO) or Multi-node cluster with a single vNIC.
Single NIC for NB and SB communication for AIOs	Cisco Smart PHY, Release 23.2	You can optionally deploy an AIO cluster with a single vNIC.

The `clusters` section defines the type and configuration of the `cluster`. At least one `environment` and one `deployer` must be defined in the cluster configuration file in order for a `cluster` to be deployed.

Clusters can be deployed on a single ESXi host or across three ESXi hosts. The single host deployment is known as a single-node deployment, or an All-in-one (AIO), while the three-node deployment is known as a multi-node deployment.

By default, All-in-one and Multi-Node clusters are deployed with two vNICs. One vNIC must be logically connected to a management network, while the other must be logically connected to the CIN. If you have a shared management and CIN network, you can optionally deploy an AIO/MN cluster with a single vNIC. For Multi Node cluster with Single NIC deployment, a Virtual IP (VIP) will be associated as the primary interface for all ops nodes.

The following `clusters` configuration below shows the mandatory and optional key-value pairs required for a multi-node deployment.

```
clusters:
  <cluster-name>:
    type: <value>           # Cluster name
    size: <value>          # Cluster type must be 'opshub'
    'normal'.              # Optional cluster size: 'small' or
                           # Defaults to 'small' when not
                           # specified.
    environment: <value>   # Environment name defined in the
                           # 'Environments' section.
    gateway: <value>       # Cluster Gateway IPv4 Address
    ingress-hostname: <value> # Optional FQDN (Ex.
    "smartphy.example.com")
    username: <value>     # Cluster username, User is
                           # prompted for cluster password
    private-key-file: <value> # SSH private-key-file (.pem) including
                           # path relative to the staging
    directory              # Key is auto-generated, if not
    provided.
    primary-vip: <value>   # Management Virtual IPv4 Address
    in CIDR format
    vrouter-id: <value>    # Management Keepalived Virtual
    Router ID,
                           # value must be between 0-255
    enable-http-redirect: value # Optional. Defaults to false when
    not specified.
                           # Set to "true" to redirect HTTP
    requests to HTTPS.
                           # The next three key-value pairs are
```

```

optional.
need to
IP Addressing
  pod-subnet: <value>
  service-subnet: <value>
  docker-bridge-subnet: [ <addr-list> ]
  CIDR format.
  additional-vip:
  additional-vip config,
  deployment
    - vip: "172.22.18.192/25"
  provision nic
  information should
  additional IP is
  and
    vip-ipv6: "2001:DB8:54FF:24:0000:0000:655:0017/112" # not a mandatory field. The
  CIDR value for 'vip-ipv6'
  range of 1 to 112.
    vrouter-id: 188
  master so that can't
  in range 1 to 255
    vrrp-routers:
      - "ops-1":
        routes:
          - {dest: ["10.208.0.0/16"], nhop: "10.208.0.1"}
          - {dest: ["10.117.0.0/16"], nhop: "10.208.0.1"}
      - "ops-2":
        routes:
          - {dest: ["10.208.0.0/16"], nhop: "10.208.0.1"}
          - {dest: ["10.117.0.0/16"], nhop: "10.208.0.1"}
      - "ops-3":
        routes:
          - {dest: ["10.208.0.0/16"], nhop: "10.208.0.1"}
          - {dest: ["10.117.0.0/16"], nhop: "10.208.0.1"}
    nodes:
      - host: <value>
        addresses: [ <addr-list> ]
  addr assigned to
  Ops VMs respectively
    datastore: <value>
  vCenter Datastore
    - host: <value>
      addresses: [ <addr-list> ]
  addr assigned to
  and Ops VMs respectively
    datastore: <value>

```

They should only be used when you customize the cluster's internal # K8s Pod subnet in CIDR format. If omitted, defaults to: 192.168.0.0/16 # K8s Service subnet in CIDR format. If omitted, defaults to 10.96.0.0/12 # List of Docker bridge subnets in CIDR format. If omitted, defaults to 172.17.0.0/16 # Optional, this complete its needed only for Single NIC MN # If additional-vip is defined then is not allowed. No additional NIC be present in apps section. This required for southbound communication load balancing for Ops nodes. param should fall within the # vrouter_id, 51, is reserved for be used, use different vrouter_id # ESXi Host 1: IPv4 Address # ESXi Host 1: List of Mgmt IPv4 control-plane, etcd, infra, and # ESXi Host 1: IPv4 Address for # ESXi Host 2: IPv4 Address # ESXi Host 2: List of Mgmt IPv4 control-plane, etcd, infra, # ESXi Host 2: IPv4 Address for

```

vCenter Datastore
  - host: <value> # ESXi Host 3: IPv4 Address
    addresses: [ <addr-list> ] # ESXi Host 3: List of Mgmt IPv4
addr assigned to
    control-plane, etcd, infra,
and Ops VMs respectively
  datastore: <value> # ESXi Host 3: IPv4 Address for
vCenter Datastore
  apps:
  - smartphy: #
are Smart PHY specific
  nodes: #
  - host: <value> # ESXi Host 1: IPv4 Address (Same
address as # ESXi Host 1: IPv4 Address (Same
used earlier in the nodes
section.)
  nics: [ <list> ] # ESXi Host 1: vCenter list of
Network for CIN # -- The following parameters
  ops: # Ops VM 1: CIN Interface
apply to Ops VM 1. --
  interfaces: # Ops VM 1: CIN Interface
configuration:
  - addresses: [ <addr-list> ] # Ops VM 1: List of CIN IPv4 or
IPV4 # Ops VM 1: List of CIN Virtual
  vip: [ <vip-list> ] # Ops VM 1: List of CIN Virtual
or v6 Addresses in CIDR format
  vrouter-id: <value> # Ops VM 1: CIN Keepalived Virtual
Router ID, # Ops VM 1: CIN Keepalived Virtual
value must be between 0-255
  routes: # Ops VM 1: Optional Route
configuration:
  - { dest: [ <list> ], nhop: <value> } # Ops VM 1: Optional list of
Destination # Ops VM 1: Optional list of
mask> Subnets in <IP address>/<subnet
format; Next-Hop IP Address
  - { dest: [ <list> ], nhop: <value> } # Ops VM 1: Optional list of
Destination # Ops VM 1: Optional list of
mask> Subnets in <IP address>/<subnet
format; Next-Hop IP Address
  - host: <value> # ESXi Host 2: IPv4 Address (Same
address # ESXi Host 2: IPv4 Address (Same
as used earlier in the nodes
section.) # ESXi Host 2: vCenter list of
  nics: [ <list> ] # ESXi Host 2: vCenter list of
Network for CIN # -- The following parameters
  ops: # Ops VM 2: CIN Interface
apply to Ops VM 2. --
  interfaces: # Ops VM 2: CIN Interface
configuration:
  - addresses: [ <addr-list> ] # Ops VM 2: List of CIN IPv4 or
IPV4 or # Ops VM 2: List of CIN Virtual
  vip: [ <vip-list> ] # Ops VM 2: List of CIN Virtual
IPV6 Addresses in CIDR format
  vrouter-id: <value> # Ops VM 2: CIN Keepalived Virtual
Router # Ops VM 2: CIN Keepalived Virtual
ID, value must be between 0-255

```



```

configuration:
    routes:
        # Ops VM 2: Optional Route
        - { dest: [ <list> ], nhop: <value> } # Ops VM 2: Optional list of
        Destination
        Subnets in <IP address>/<subnet
        mask>
        format; Next-Hop IP Address
        - { dest: [ <list> ], nhop: <value> } # Ops VM 2: Optional list of
        Destination
        Subnets in <IP address>/<subnet
        mask>
        format; Next-Hop IP Address
        - host: <value> # ESXi Host 3: IPv4 Address (Same
        address as used earlier in the nodes
        section.)
        nics: [ <list> ] # ESXi Host 3: vCenter list of
        Network for CIN
        ops: # -- The following parameters
        apply to Ops VM 3. --
        interfaces: # Ops VM 3: CIN Interface
        configuration:
            - addresses: [ <addr-list> ] # Ops VM 3: List of CIN IPv4 or
            IPv6 Addresses in CIDR format
            vip: [ <vip-list> ] # Ops VM 3: List of CIN Virtual
            IPv6 Addresses in CIDR format
            vrouter-id: <value> # Ops VM 3: CIN Keepalived Virtual
            Router ID, value must be between
            0-255
        routes: # Ops VM 3: Optional Route
        configuration:
            - { dest: [ <list> ], nhop: <value> } # Ops VM 3: Optional list of
        Destination
        Subnets in <IP address>/<subnet
        mask>
        format; Next-Hop IP Address
            - { dest: [ <list> ], nhop: <value> } # Ops VM 3: Optional list of
        Destination
        Subnets in <IP address>/<subnet
        mask>
        format; Next-Hop IP Address

```



Note In the preceding `clusters` configuration, the syntax for configuring all instances of `routes` (as shown below) is only applicable for Cisco Smart PHY, Release 23.2 and later. See [Install and Upgrade Guides](#), to view the `clusters` configuration for previous releases of Cisco Smart PHY.

```

routes:
    # Ops VM VM Number:Optional Route configuration:
    - { dest: [ <list> ], nhop: <value> } # Ops VM VM Number: Optional list of Destination
    Subnets in <IP address>/<subnet mask>
    format; Next-Hop IP Address
    - { dest: [ <list> ], nhop: <value> } # Ops VM VM Number: Optional list of Destination
    Subnets in <IP address>/<subnet mask>

```

Guidelines for Defining a Cluster

- The `cluster-name` can have only lowercase letters, digits, and hyphens (-).

- When you specify a FQDN in the `ingress-hostname`, ensure that the corresponding entries are configured in your DNS servers:

```
<host.domain.tld>
opscenter.<host.domain.tld>
```

- If you do not specify an FQDN in the `ingress-hostname`, the cluster's `primary-vip` (also known as the Management Virtual IP address) is used to generate an FQDN leveraging `nip.io` as the domain and top-level domain (TLD). For example, if the `primary-vip` is `10.0.0.2`, the generated FQDN is `10.0.0.2.nip.io`. Your DNS servers must allow the resolution of the `nip.io` domain. If resolution of `nip.io` is blocked, you cannot access the cluster.
- The `private-key-file`, when present, must refer to the SSH private key file (`.pem`). This file must be in the staging directory and must not be accessible (read/write/execute) to other users.
- If the `private-key-file` line is omitted, the deploy tool generates an SSH private key for the cluster and places it in the `.sec` subdirectory under the staging directory. The filename is `<cluster-name>_auto.pem`.
- If multiple clusters share the same management subnet, the management `vrouter-id` (VRRP ID) of each cluster must be unique.

Cisco Smart PHY CIN Configuration

Configure Converged Interconnect Network (CIN) for the Cisco Smart PHY cluster. One or more CIN networks can be present. Configure CIN under each node.

Guidelines for Defining Smart PHY's CIN Interfaces

- The CIN Virtual IP addresses (`vip`) field is mandatory. You can configure up to one IPv4 and one IPv6 addresses per CIN network.
- The CIN Virtual IP addresses (`vip`) and the VRRP ID (`vrouter-id`) fields are used only in multi-node cluster deployments. They are configured on the first node.
- If multiple Smart PHY clusters share a CIN subnet, the VRRP ID (`vrouter-id`) should be unique for each cluster.
- For multi-node clusters, all Ops VMs must have the same number of CIN interfaces. The `nics` or `route` fields must be explicitly mentioned on all Ops VM nodes.
- You can setup a Smart PHY cluster as a backup cluster. To do so, do not include any CIN configuration. The configuration should not have `ops` and `interfaces` under `nodes`.



Note Cisco Smart PHY clusters can connect to multiple CIN networks using multiple network interfaces configured during deployment.

Adding CIN Configuration Without Cluster Reboot

After deploying Smart PHY, you can add new CIN Configuration without restarting the cluster by using the following steps:

1. In the Day-0 config, add additional CIN Configurations details and provide the necessary details of CIN, interface address, vip, and vrouter-id config.
2. Run cluster deployment with the `-np` argument. Example: `./deploy -c day0.yaml -np`

Once deployment is done successfully, the cluster is updated with CIN Configuration information. The cluster does not reboot.

Deploying the Deployer VM and Cisco Smart PHY Cluster

Table 4: Feature History

Feature Name	Release Information	Description
Deploying Clusters on Network Based Storage Devices Using vSAN	Cisco Smart PHY, Release 23.2	You can deploy clusters to remote network-attached storage devices using vSAN.

This section explains how to use the `deploy` tool to deploy the Deployer VM and Cisco Smart PHY cluster.

From the staging environment, run the `deploy` tool to deploy the clusters using the following command:

```
$ ./deploy
Usage ./deploy -c <config_file> [-v]
  -c <config_file>      : Configuration File, <Mandatory Argument>
  -v                   : Config Validation Flag [Optional]
  -f                   : Cluster config_file: Force VM Redeploy Flag [Optional]
  -u                   : Cluster Upgrade Flag [Optional]
  -s                   : Update the Inception VM, and upload the CNF tar files to the VM.
                        It can be used with -u, -p, or -r flags to reduce the maintenance
                        window before the actual operation [Optional]
  -sc                  : Skip Compatibility check during upgrade, patch, or rollback
                        operation. It can be used with -u, -p, or -r flags [Optional]
  -D                   : Enable Debug Logs [Optional]
  -np                  : Provision Additional Nic [Optional]
  -sv                  : Skip Vcenter-obj Validation [Optional]
  -gui                 : Trigger and monitor auto deployer actions through Browser [Optional]

  --maintenance       : To trigger planned or unplanned cluster maintenance [Optional]
  --resume-maintenance : To resume nodes from maintenance mode [Optional]
  -p <path/patch_file> : Cluster patch operation [Optional]
  -r                   : Revert the CNF to the version that is packaged
                        in the installer [Optional]
```

The following options are available in the `deploy` tool:

- `-c <config_file>`: Configuration file (Mandatory Argument). This option is the first option in the command.
- `-u`: Cluster chart Update Flag. This is for updating CNF/charts in cluster. [Optional]
- `-v`: Config Validation Flag, [Optional]
- `-f`: Redeploy the cluster. If you redeploy the cluster, cluster VMs are rebooted, and the data persisted on disk is retained. You can use this option to modify some of the cluster parameters. [Optional]

- `-s`: Staging flag. This is used to upload a CNF tar file to the deployer without running SMI sync. It should be used in conjunction with `-u`, `-p` or `-r` [Optional]
- `-np`: Nic Provision flag. This is used to provision additional NICs.
- `-sv`: Skip validation flag. This is used to skip all the vcenter object validations. It can be used with any of the flags [Optional]
- `-gui`: Autodeployer browser flag. This is used to trigger and monitor auto deployer actions through a browser interface.
- `--maintenance`: Cluster Maintenance flag. This is used to trigger planned or unplanned cluster maintenance.
- `resume-maintenance`: Nodes Maintenance flag. This is used to resume nodes from maintenance mode.
- `-p <patch-file>`: Patch flag. This is used to patch/update the CNF files in cluster.
- `-r`: Rollback flag. This is used to revert the CNF to the version that is packaged in the installer.

The `-u` flag is for updating CNF/charts in cluster.

The `deploy` tool triggers the `docker` command that requires root permission to run. Depending on your setting, you can use the `sudo` to the `deploy` command.

The `deploy` tool does the following operations:

- If you're running the `deploy` tool for the first time, it prompts you to enter all passwords required for installation.
 - For vCenter environment: vCenter password for the user specified in the environment configuration.
 - For deployer: SSH password of the user `admin` for the deployer's Operation Center.
 - For Cisco Smart PHY cluster: SSH password for all VMs in the cluster (or user-specified in the cluster's configuration file). Also, the SSH passwords for the three Operation Centers (Cisco Smart PHY, Operations Hub, and CEE); for user `admin`.

You're prompted twice to enter each password. The password is saved inside the staging directory in encrypted form for future use.

- Passwords for the deployer, the cluster, and the Operation Centers must be eight characters long, and must have a lowercase letter, uppercase letter, a digit, and a special character.
- The `deploy` tool generates an SSH key pair when the `private-key-file` line is missing for the deployer or the cluster in the configuration file. The generated private key files are in the `.sec` sub directory under the staging directory, with `<cluster-name>_auto.pem` filename.
- The root user owns the generated private keys. When logging in using SSH and these private key files, make sure that you run it with `sudo`.
- If the deployer VM isn't running, the `deploy` tool installs the deployer VM.
- The `deploy` tool checks if the deployer VM is missing any of the product packages that are found in the `offline-images` directory, and if it finds any missing, it uploads them to the deployer VM.
- The tool also generates the configuration for each cluster and pushes them to the deployer VM.

- The deploy tool triggers the deployer VM to perform the sync operation for the cluster. The sync operation applies the configuration to the cluster. If you haven't set up the cluster, it installs the cluster. Or the sync operation updates the cluster with the configuration.
- If the sync operation times out, the deploy tool triggers the sync operation again. The tool waits for the sync operation to complete, and then continues to monitor the cluster to make sure that all helm charts are deployed and all pods are created.

You can repeat the deploy tool to deploy more than one cluster by providing the corresponding configuration files. Alternatively, you can run this command appending a `-v` flag. The `-v` flag forces the deploy tool to skip the synchronizing operation. Use this option to push the configuration of a cluster to the deployer without deploying or updating the cluster.

Wait for the installation process to complete. Following is a sample output after the process is complete:

```
Friday 22 October 2021 07:53:52 +0000 (0:00:00.123) 0:12:22.518 *****
install-cm-offline : Extract cluster manager file into /data ----- 545.16s
vm-vsphere-iso : Wait for ssh ----- 88.51s
install-cm-offline : Deploy cluster manager ----- 85.14s
install-ntp-iso : force_time_sync ----- 7.34s
vm-vsphere-iso : Create VM ----- 3.85s
vm-vsphere-iso : Get VM Update needed ----- 1.65s
install-ntp-iso : Cleaning cache ----- 1.53s
Gathering Facts ----- 1.34s
vm-vsphere-iso : Check if ISO file exists ----- 0.79s
vm-vsphere-iso : Test vCenter credentials are valid ----- 0.60s
install-ntp-iso : apt_update ----- 0.55s
vm-vsphere-iso : Create user data ISO ----- 0.52s
install-ntp-iso : Remove "ntp" package ----- 0.47s
install-cm-offline : Ensure /data/cm-install folder NOT exists ----- 0.36s
install-ntp-iso : Install offline APT repo GPG key ----- 0.34s
install-cm-offline : Ensure /data folder exists ----- 0.33s
install-ntp-iso : restart_chrony ----- 0.28s
install-ntp-iso : enable_chrony_ntp ----- 0.28s
download-iso : download base image ISO file ----- 0.28s
vm-vsphere-iso : Create netplan Template ----- 0.18s
```

Create deployers completed

Deploying Clusters on Network Based Storage Devices Using vSAN

The Operations hub platform and the applications running over it can be deployed on a vSAN cluster. vSAN stands for Virtual Storage Area Network. It is a software-defined storage (SDS) solution that is offered by VMware that virtualizes and abstracts storage resources in a vSphere environment. vSAN combines local storage devices such as hard drives and solid-state drives (SSDs) from multiple hosts into a single shared storage pool.

Prerequisites for Deploying Clusters on vSAN

1. All the ESXi hosts used for the cluster deployment in the day-0 configuration yaml should be part of the same vSAN cluster.
2. The network speed between ESXi hosts in a vSAN cluster should be at least 10 Gbps.
3. On VCenter 7.0, proactive tests are available to ensure the vSAN cluster setup is proper. We recommend that you run proactive tests before installation to ensure the vSAN cluster is healthy. Use the following steps to run the Proactive tests using the VCenter GUI:

- Access **Cluster > Monitor > vSAN** and select **Proactive tests**.

- You can run the **VM Creation Test**, **Network Performance Test**, and **Storage Performance Test**. These tests must pass.
4. We recommend that you enable vSphere HA if there is a host failure. VMs should be moved to the active host. If vSphere HA is not configured, then it is recommended to use **VM/Host groups** and **VM/Host rules** to ensure the **VM-to-host** allocation model is mentioned in the day-0 configuration yaml.
 - ESXi-1 should host Control-plane-1, ops-1, infra-1, etcd-1
 - ESXi-2 should host Control-plane-2, ops-2, infra-2, etcd-2
 - ESXi-3 should host Control-plane-3, ops-3, infra-3, etcd-3

Verifying Installation

After successfully deploying the Cisco Smart PHY application using the deploy tool, the console shows a success message.

Log in to one of the control-plan nodes and make sure that all the pods are in the Running state.

```
kubectl get pod --all-namespaces
```

A few internal services and pods may need more time to complete the startup tasks and successfully establish communication with other services within the cluster. After a few minutes, you can initiate all operations from the Cisco Smart PHY web UI page.

Options That Can Be Passed With Other Arguments

`./deploy -c <config_file> -v`: This command validates the config file and vCenter objects.

`./deploy -c <config_file> -D`: This command is used to install the cluster in debug mode to capture more logs. The `-D` option can also be used during any stage of the deployment process.

`./deploy -c <config_file> -sv`: This command is used to skip the vCenter object validation during the cluster Installation/Upgrade/Patch/Rollback.

`./deploy -c <config_file> -np`: This command is used to perform the CIN activation on the running cluster without any VM reboot.

`./deploy -c <config_file> -u -s`: This command is used to perform the staging operation which updates the CNF files to the deployer during upgrade.

`./deploy -c <config_file> -p -s`: This command is used to perform the staging operation which updates the CNF files to the deployer during patch operation.

Patch Installation Command

`./deploy -c <config_file> -p <path/patch-file>.tgz`: This is used to perform the cluster patch with the provided patch.tgz file.

Verifying Installation

After successfully deploying the Cisco Smart PHY application using the deploy tool, the console shows a success message.

Log in to one of the control-plan nodes and make sure that all the pods are in the `Running` state.

```
kubectl get pod --all-namespaces
```

A few internal services and pods may need more time to complete the startup tasks and successfully establish communication with other services within the cluster. After a few minutes, you can initiate all operations from the Cisco Smart PHY web UI page.

Auto Deployer User Interface

Table 5: Feature History

Feature Name	Release Information	Description
Auto Deployer User Interface Enhancements	Cisco Smart PHY, Release 24.1	You can use guided flows in the Operations Hub Deployer GUI to <ul style="list-style-type: none"> • Install a new cluster. • Upgrade, patch, and provision NIC on existing clusters. • Validate a cluster configuration file before using it.
Auto Deployer User Interface	Cisco Smart PHY, Release 23.3	With this release, you can deploy a Smart PHY and Operations Hub cluster using the Operations Hub Deployer Graphical User Interface(GUI). This method simplifies the process of installation and this method provides an enhanced validation framework. CLI-based installation remains supported.

Operations Hub Deployer GUI

You can install, upgrade, patch a cluster using the auto deployer user interface. When the prerequisites are met and the staging environment is set properly, initiate cluster deployment through the Operations Hub Deployer GUI.

Use the following steps to launch the Operations Hub Deployer GUI.

1. Enter the **./deploy -gui** command in the Autodeployer CLI and capture the URL and user credentials that are displayed.

Example:

```
$ ./deploy -gui
check if the signature file is present locally.
Software components are not signed. skipping signature verification step.

Running autodeployer...

2023-10-10 14:09:56. 724-INFO: Running on autodeployer GUI at location
https://{autodeployer-URL}/
Credentials for auto deployer GUI user: {user} password {password}
```



Note The URL and user credentials remains active as long as the autodeployer CLI session is active. If you close the autodeployer CLI session, you must run the `./deploy -gui` command to generate a new Autodeployer URL with new credentials.

2. Open a browser, enter the URL and login to the Operations Hub Deployer GUI using the credentials provided in the previous step.
3. After a successful login, you can choose one of the following options:
 - Deploy a Cluster
 - Upgrade a Cluster
 - Patch a Cluster
 - Validate Cluster Configuration File
 - Provision CIN Interfaces



-
- Note**
- When you log in to the Operations Hub Deployer GUI, a session with a validity of 360 minutes is created. If the operation takes more than 360 minutes to complete, you are logged out of the session and must log in to the Operations Hub Deployer GUI again using the same credentials.
 - If the Autodeployer is triggered through an SSH console session, the **SSH timeout** can lead to the termination of the Autodeployer process. We recommended you to configure a higher value for **SSH timeout** to avoid the termination of the Autodeployer process.
 - If you close the browser session before the deployment is complete, then the container continues to run after the deployment, and it must be stopped manually. Use one of the options below to manually stop the container.
 1. Access the terminal where the Autodeployer process is running and press the **Ctrl+C** keys.
 2. Find the docker container with the container name **autodeployer** and kill this container.
 - When an operation triggered from the Operations Hub Deployer GUI is complete, click **Next** to view the Operations Hub Deployer GUI Landing Page. Post this, the container is terminated in 5 seconds by the Operations Hub Deployer GUI. You can now close the Operations Hub Deployer GUI web page.
-

Deploy a Cluster

You can use the **Deploy a Cluster** option to initiate a new deployment of a cluster.

1. Access the landing page of the Operations Hub Deployer GUI and click **Deploy a Cluster**. The **Deploy a Cluster - Overview** panel displays.
2. Review the **Deploy a Cluster - Overview** panel and click **Next** to display the **Validate Cluster Configuration File - Deploy a Cluster** panel.
3. Upload the Operations Hub cluster configuration file using one of the options below:

- **My Computer:** Click **Choose a file** and select a valid cluster configuration file from your computer.
 - **Staging Environment:** In the **File Name** field, enter the name of the cluster configuration file that is saved in the staging directory in `.yaml` format. For Example, `testconfig.yaml`. Ensure that the filename is the same as the cluster configuration file stored in the staging directory. If the cluster configuration file is stored in a subfolder under the staging directory, then enter the complete path of the cluster configuration file. For Example `downloads/testconfig.yaml`. You can optionally download a sample cluster configuration file by clicking the **Download a sample cluster configuration file** hyperlink.
4. Click **Validate Config File** to validate the file content. In case of any errors, you can view and fix the errors before moving to the next step. If the validation is successful, then click **Next** to display the **Credential Validation - Deploy a Cluster** panel.
 5. Enter the vCenter, Deployer, Cluster, and OpsCenter passwords in the respective password fields and click **Validate Credentials**. In case of any you have entered incorrect credentials, you can correct the credentials before moving to the next step.
 6. If all the passwords are successfully validated, then click **Start Deployment** to start the deployment. You can view the real time progress of each stage of the deployment process. The deployment process contains the following stages:
 - Configuration Validation
 - Inception VM Creation
 - Configure Clusters
 - Cluster Sync
 - Post Check
 - Configure Log Forwarding

You can view the real time progress of each stage of the deployment process in the UI. The status of each stage is also indicated as mentioned below:

Icon	Status of the Stage
Green icon with a check mark	Stage is complete
Blue icon with spinning arrows	Stage is in progress
Gray icon with a crescent moon	Stage is yet to start
Red icon with a cross	Stage has stopped due to some errors. You must start the operation all over again. For more information, check the error logs.

Cluster deployment may take several minutes to complete. When all the stages are completed without any errors, you can close the Operations Hub Deployer GUI and login to the Operations Hub cluster. See [Cisco Operations Hub User Guide](#) for details.

If you wish to cancel the deployment at any point (even if there are no errors), click **Cancel**, and then confirm by clicking **Yes, Cancel**.



Note If you cancel the deployment operation during the **Cluster sync** or **Inception VM creation** stage, then you must clean up the cluster (i.e. cluster and inception VM) before proceeding with the next installation. You must perform the cleanup even if **Cluster sync** or **Inception VM creation** fail due to an error.

Logs for each stage are displayed in the **Deployer Logs** section. If there are any errors in any stage, they are captured in the logs. You can download (*logs.json* format) or copy the logs by clicking the **Download** or the **Copy** options respectively.

Upgrade a Cluster

You can initiate a Cluster upgrade through the Operations Hub Deployer GUI.

1. Access the landing page of the Operations Hub Deployer GUI and click **Upgrade a Cluster**. The **Upgrade a Cluster - Overview** panel displays.
2. Review the **Upgrade a Cluster - Overview** panel and click **Next** to display the **Validate Cluster Configuration File - Upgrade a Cluster** panel.
3. Upload the Operations Hub cluster configuration file using one of the options below:
 - **My Computer**: Click **Choose a file** and select a valid cluster configuration file from your computer.
 - **Staging Environment**: In the **File Name** field, enter the name of the cluster configuration file that is saved in the staging directory in *.yaml* format. For Example, **testconfig.yaml**. Ensure that the filename is the same as the cluster configuration file stored in the staging directory. If the cluster configuration file is stored in a subfolder under the staging directory, then enter the complete path of the cluster configuration file. For Example **downloads/testconfig.yaml**. You can optionally download a sample cluster configuration file by clicking the **Download a sample cluster configuration file** hyperlink.
4. Click **Validate Config File** to validate the file content. In case of any errors, you can view and fix the errors before moving to the next step. If the validation is successful, then click **Next** to display the **Credential Validation - Deploy a Cluster** panel.
5. Enter the vCenter, Deployer, Cluster, and OpsCenter passwords in the respective password fields and click **Validate Credentials**. In case of any you have entered incorrect credentials, you can correct the credentials before moving to the next step.
6. If all the passwords are successfully validated, then click **Start the Upgrade** to start the upgrade process. You can view the real time progress of each stage of the upgrade process. The Upgrade process contains the following stages:
 - Configuration Validation
 - Initializing the Operation
 - Performing Integrity Check
 - Cluster Health Check
 - Backing up the Database
 - Performing pre-update Actions
 - Updating Inception VM

- Cluster sync (for Platform and Application upgrade) or Performing application update (for Only application upgrade)
- Performing Post Check
- Verifying Log Forward Configuration

You can view the real time progress of each stage of the upgrade process in the UI. The status of each stage is also indicated as mentioned below;

Icon	Status of the Stage
Green icon with a check mark	Stage is complete
Blue icon with spinning arrows	Stage is in progress
Gray icon with a crescent moon	Stage is yet to start
Red icon with a cross	Stage has stopped due to some errors. Check the error logs for more information.

Cluster upgrade may take several minutes to complete. It is not recommended to cancel the ongoing upgrade process.

When all the stages are completed without any errors, you can close the Operations Hub Deployer GUI and login to the Operations Hub cluster. See [Cisco Operations Hub User Guide](#) for details.

Logs for each stage are displayed in the **Deployer Logs** section. If there are any errors in any stage, they are captured in the logs. You can download (*logs.json* format) or copy the logs by clicking the **Download** or the **Copy** options respectively.

Patch a Cluster

Auto deployer allows you to apply patches to a base installation of a cluster.

Prerequisites



Note You can't apply a patch without downloading the base installer image.

- Download the following tarball Images from <https://software.cisco.com/download/home/286319080/type>.
 1. Smart PHY **Base** Release Signed tarball Image : This package contains the necessary tools for deploying, updating, and patching of the cluster and application software in tarball format. A base installer is provided with every stable major release of Smart PHY.
 2. Smart PHY **Patch** Release Signed tarball Image : This file only contains the application software. A Patch installer is released when a software bug fix or security fix needs be installed over the base installation of Smart PHY. This package does not contain the necessary tools for deploying, updating, and patching of the cluster. A patch tarball Image is different from an upgrade tarball Image.
- Extract the base installer. The extracted installer directory is called the staging directory. Do not use the same staging directory for applying multiple patches or reuse it to apply the same patch on a different cluster.
- Place the cluster and deployer VM SSH private keys and patch the tarball Image that is downloaded earlier inside the extracted staging directory. The name of the private key file must be the same as that of private key name mentioned in the cluster configuration YML.

Follow the step below to install a Patch:

1. Access the landing page of the Operations Hub Deployer GUI and click **Patch a Cluster**. The **Patch a Cluster - Overview** panel displays.
2. Review the **Patch a Cluster - Overview** panel and click **Next** to display the **Validate Cluster Configuration File - Patch a Cluster** panel.
3. Upload the Operations Hub cluster configuration file using one of the options below:
 - **My Computer**: Click **Choose a file** and select a valid cluster configuration file from your computer.
 - **Staging Environment**: In the **File Name** field, enter the name of the cluster configuration file that is saved in the staging directory in *.yaml* format. For Example, **testconfig.yaml**. Ensure that the filename is the same as the cluster configuration file stored in the staging directory. If the cluster configuration file is stored in a subfolder under the staging directory, then enter the complete path of the cluster configuration file. For Example **downloads/testconfig.yaml**. You can optionally download a sample cluster configuration file by clicking the **Download a sample cluster configuration file** hyperlink.
4. Click **Validate Config File** to validate the file content. In case of any errors, you can view and fix the errors before moving to the next step. If the validation is successful, then click **Next** to display the **Credential Validation - Patch a Cluster** panel.
5. Enter the vCenter, Deployer, Cluster, and OpsCenter passwords in the respective password fields and click **Validate Credentials**. In case of any you have entered incorrect credentials, you can correct the credentials before moving to the next step.
6. If all the passwords are successfully validated, then click **Start the Upgrade** to display the **Patch File Name** panel. In the provided **Patch File Name** field, enter the name of the Smart PHY **Patch** Release

Signed tarball Image, which is saved in the staging directory. For Example, **smartphy-installer-v2X.X.X-XXXXXX-XXXX.SPA.tgz**

7. Click **Apply the patch** to start the patch process. You can view the real time progress of each stage of the patch process. The patch process contains the following stages:
 - Configuration Validation
 - Initializing the Operation
 - Performing Integrity Check
 - Cluster Health Check
 - Backing up the Database
 - Performing pre-update Actions
 - Updating Inception VM
 - Performing Application Update
 - Performing Postcheck
 - Verifying log forward configuration

Icon	Status of the Stage
Green icon with a check mark	Stage is complete
Blue icon with spinning arrows	Stage is in progress
Gray icon with a crescent moon	Stage is yet to start
Red icon with a cross	Stage has stopped due to some errors. Check the error logs for more information.

Cluster patch may take several minutes to complete. When all the stages are completed, you can close the Operations Hub Deployer GUI and login to the Operations Hub cluster. See [Cisco Operations Hub User Guide](#) for details.

It is not recommended to cancel the ongoing patch process. If you wish to cancel the deployment at any point (even if there are no errors), click **Cancel**, and then confirm **Yes, Cancel**. After Cancellation, you can retry Patch a cluster.

Logs for each stage are displayed in the **Deployer Logs** section. If there are any errors in any stage, they are displayed in the logs. You can download (*logs.json* format) or copy the logs by clicking the **Download** or the **Copy** options respectively.

Provision CIN Interfaces

If you chose not to provision CIN interfaces during your cluster's initial deployment, you can do so when you are ready from the Operations Hub Deployer GUI.

1. Access the landing page of the Operations Hub Deployer GUI and click **Provision CIN Interfaces**. The **Provision CIN Interfaces - Overview** panel displays.
2. Review the **Provision CIN Interfaces - Overview** panel and click **Next** to display the **Validate Cluster Configuration File - Provision CIN Interfaces** panel.

3. Upload the Operations Hub cluster configuration file using one of the options below:
 - **My Computer:** Click **Choose a file** and select a valid cluster configuration file from your computer.
 - **Staging Environment:** In the **File Name** field, enter the name of the cluster configuration file that is saved in the staging directory in *.yaml* format. For Example, **testconfig.yaml**. Ensure that the filename is the same as the cluster configuration file stored in the staging directory. If the cluster configuration file is stored in a subfolder under the staging directory, then enter the complete path of the cluster configuration file. For Example **downloads/testconfig.yaml**. You can optionally download a sample cluster configuration file by clicking the **Download a sample cluster configuration file** hyperlink.
4. Click **Validate Config File** to validate the file content. In case of any errors, you can view and fix the errors before moving to the next step. If the validation is successful, then click **Next** to display the **Credential Validation - Provision CIN Interfaces** panel.
5. Click **Start Provisioning** to provision CIN Interfaces. You can view the real time progress of each stage of the provisioning process. The provisioning process contains the following stages:
 - Configuration Validation
 - Merge Cluster Configuration
 - Configure Additional Network Interface
 - Inception VM creation
 - Cluster Sync
 - Postcheck

Icon	Status of the Stage
Green icon with a check mark	Stage is complete
Blue icon with spinning arrows	Stage is in progress
Gray icon with a crescent moon	Stage is yet to start
Red icon with a cross	Stage has stopped due to some errors. For more information, check the error logs.

If you wish to cancel the deployment at any point (even if there are no errors), click **Cancel**, and then confirm by clicking **Yes, Cancel**. After Cancellation, you can retry provisioning the CIN Interfaces.

CIN Provisioning may take several minutes to complete. When all the stages are completed, you can close the Operations Hub Deployer GUI and login to the Operations Hub cluster. See [Cisco Operations Hub User Guide](#) for details.

Logs for each stage are displayed in the **Deployer Logs** section. If there are any errors in any stage, they are displayed in the logs. You can download (*logs.json* format) or copy the logs by clicking the **Download** or the **Copy** options respectively.

Validate Cluster Config File

You can validate a Cluster Configuration before using it. Validating a cluster configuration file ensures that:

- The YAML is formatted properly.

- All mandatory keys and values are present.
- All values meet data type requirements.
- All resource credentials are valid.

Validation occurs in two phases:

- Input Validation (Syntactical and Semantic)
- Credential Validation

Validate Cluster Config File can be triggered through the Operations Hub Deployer GUI.

1. Access the landing page of the Operations Hub Deployer GUI and click **Validate Cluster Config File**. The **Validate Cluster Config File - Overview** panel displays.
2. Review the **Validate Cluster Config File - Overview** panel and click **Next** to display the **Validate Cluster Config File - Input Validation** panel.
3. Validate the Operation Hub cluster configuration file by uploading a cluster configuration file using one of the options below:
4. Upload the Operations Hub cluster configuration file using one of the options below:
 - **My Computer**: Click **Choose a file** and select a valid cluster configuration file from your computer.
 - **Staging Environment**: In the **File Name** field, enter the name of the cluster configuration file that is saved in the staging directory in *.yaml* format. For Example, **testconfig.yaml**. Ensure that the filename is the same as the cluster configuration file stored in the staging directory. If the cluster configuration file is stored in a subfolder under the staging directory, then enter the complete path of the cluster configuration file. For Example **downloads/testconfig.yaml**. You can optionally download a sample cluster configuration file by clicking the **Download a sample cluster configuration file** hyperlink.
5. Select the appropriate cluster operation option from the **Intended cluster operation(Optional)** drop-down list. There are two options:
 - **Initial Cluster Deployment**: This is the default option which does not perform a ping check on the cluster and the deployer VM. You can use this option when you are validating the cluster configuration for a new cluster installation.
 - **Modification (Upgrade/Patch/CIN) of existing cluster**: This option performs a ping check on both the cluster and the deployer VM. Use this option for scenarios other than a new cluster installation.
6. Click **Validate Config File** to validate the file content. In case of any errors, you can view and fix the errors before moving to the next step. If the validation is successful, then click **Next** to display the **Credential Validation - Validate Cluster Config File** panel.
7. Enter the vCenter, Deployer, Cluster, and OpsCenter passwords in the **Password Entry** panel and click **Validate Passwords**. If all the passwords are entered correctly, a Success message displays, mentioning that all passwords are valid and that the configuration file is valid.
8. Click **Done**. The landing page of the Operations Hub Deployer GUI displays.

Configuring Dual Stack on External Cluster Interfaces

Table 6: Feature History

Feature Name	Release Information	Description
IPv6 Dual Stack on support on external cluster interfaces	Cisco Smart PHY, Release 22.2	Support that is extended to configure IPv6 dual stack configuration parameters such as mode, gateway, subnet address, and so on, on an external cluster interface.

Prerequisites

1. Dual stack must be configured at the time of cluster creation. It cannot be added to a previously created Smart PHY cluster.
2. The ESXi hosts must be connected to dual stack enabled networks.

To configure dual stack on an external cluster interface, perform the following steps:

1. Navigate to the cluster configuration file in the staging environment.
2. Add the following parameters in the configuration file.

Table 7: IPv6 Dual Stack Parameters

Parameter	Description	Value to set
ipv6-mode (Optional)	Specifies to set whether the IPv6 mode is true or false.	<ul style="list-style-type: none"> • "true"—To enable the dual stack on the cluster interface. • "false"—To disable the dual stack on the cluster interface.
primary-vip-ipv6	Specifies the IPv6 address where the Nginx ingress controller binds to.	This parameter is mandatory if the ipv6-mode is set as "true".
ipv6-gateway	Specifies the IPv6 gateway address.	This parameter is mandatory if the ipv6-mode is set as "true".
pod-subnet-ipv6	Specifies IPv6 subnet address.	This parameter is optional. If no value is specified, then by default the value "fd20::0/112" is assigned.
service-subnet-ipv6	Specifies the IPv6 service subnet address.	This parameter is optional. If no value is specified, then by default the value "fd20::0/112" is assigned. This parameter is valid only if the ipv6-mode is set as "true".

Parameter	Description	Value to set
addresses-v6	Specifies the virtual machine SSH IPv6 address.	This parameter is mandatory if the ipv6-mode is set as "true".



Note Starting with Cisco Smart PHY 23.3,

- The values for Boolean parameters such as **enable-http-redirect** and **ipv6-mode**, must be enclosed in double quotes. The accepted values are **"true"** or **"false"**.
- IPv6 address parameter values must be enclosed in double quotes. For example: **"2001:18:208::/64"**.

In previous Cisco Smart PHY releases, the usage of double quotes is optional.

Deployment Limitations

- Modification of cluster parameters such as server names, NTP server configuration details, data store file path, subnets, IP addresses of VMs, and so on, requires a VM restart. You can restart VM using the **deploy-f** command.
- Autodeployer only supports Application Product chart and docker image upgrades. The modification of cluster configuration is not supported as part of the upgrade process.
- When you enable dual-stack, you must redeploy the cluster.
- The Deployer VM must be saved so that you can use the VM while upgrading the cluster.
- Manual provisioning of NIC interfaces on VMs must be performed through vCenter during the SmartPHY installation on top of the running Operations Hub platform.
- Removal of exiting NICs requires a VM restart. You can restart VM using the **deploy-f** command.
- Data store folders must be created in vCenter manually before starting the Smart PHY installation.

Smart PHY Client Requirements

Smart PHY supports the following Operating Systems (OS) and Browsers.

Supported Operation Systems

- Mac OS 12 and later.
- Windows 10 and later.

Supported Browsers

- Mozilla Firefox Version 122 and later.
- Google Chrome Version 120 and later.

- Microsoft Edge Version 120 and later.

Windows System Resolution

Smart PHY UI screens are best seen when Windows display resolution is set to 1920 x 1080 and the size of the text and apps is set to 100%.