CPS vDRA Installation Guide for VMware, Release 20.2.0

First Published: 2020-08-27

Last Modified: 2020-08-27

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

CONTENTS

Preface

About This Guide

This document is a part of the Cisco Policy Suite documentation set.

For information about available documentation, see the *CPS Documentation Map* for this release at Cisco.com.

Audience

This guide is best used by these readers:

- Network administrators
- Network engineers
- Network operators
- System administrators

This document assumes a general understanding of network architecture, configuration, and operations.

Additional Support

For further documentation and support:

- Contact your Cisco Systems, Inc. technical representative.
- Call the Cisco Systems, Inc. technical support number.
- Write to Cisco Systems, Inc. at support@cisco.com.

- Refer to support matrix at https://www.cisco.com/c/en/us/support/index.html and to other documents related to Cisco Policy Suite.

Conventions (all documentation)

This document uses the following conventions.

Conventions	Indication
bold font	Commands and keywords and user-entered text appear in **bold** font.
italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
[ ]	Elements in square brackets are optional.
{x | y | z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
courier font	Terminal sessions and information the system displays appear in courier font.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Note Means reader take note. Notes contain helpful suggestions or references to material not covered in the manual.

Caution Means reader be careful. In this situation, you might perform an action that could result in equipment damage or loss of data.

Warning IMPORTANT SAFETY INSTRUCTIONS.

Means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.

SAVE THESE INSTRUCTIONS

Note Regulatory: Provided for additional information and to comply with regulatory and customer requirements.

Communications, Services, and Additional Information

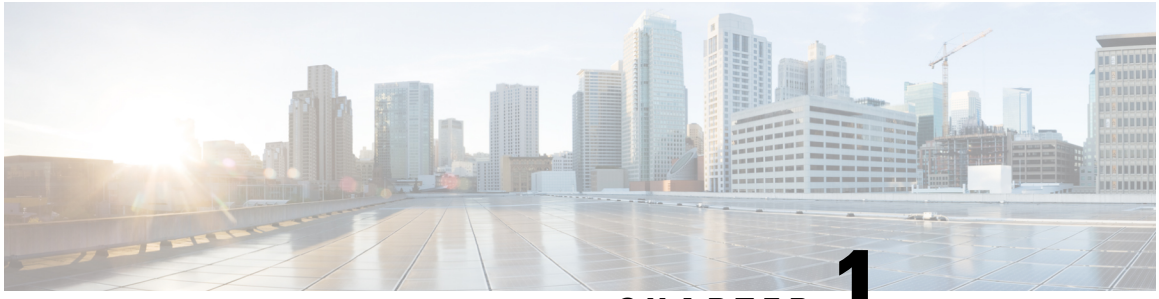
- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Important Notes

Important Any feature or GUI functionality that is not documented may not be supported in this release or may be customer specific, and must not be used without consulting your Cisco Account representative.



CHAPTER 1

Pre-Installation Requirements

Installation Overview

The vDRA vSphere installer launches vDRA VMs as specified in the User Input structure. Once the VMs are launched, all VMs must be registered with the master as displayed using the command `show running-config docker | tab`. Also, the system percent-complete must reach 100% as displayed using the command `show system status`.

Once the VMs are registered, the installer is done and you can proceed with configuring the vDRA system.

VMware ESXi 6.5 must be installed on all the blades that are used to host the vDRA system. For more details, see http://www.vmware.com/products/esxi-and-esx/overview.html.

Installing vDRA on vSphere includes the following:

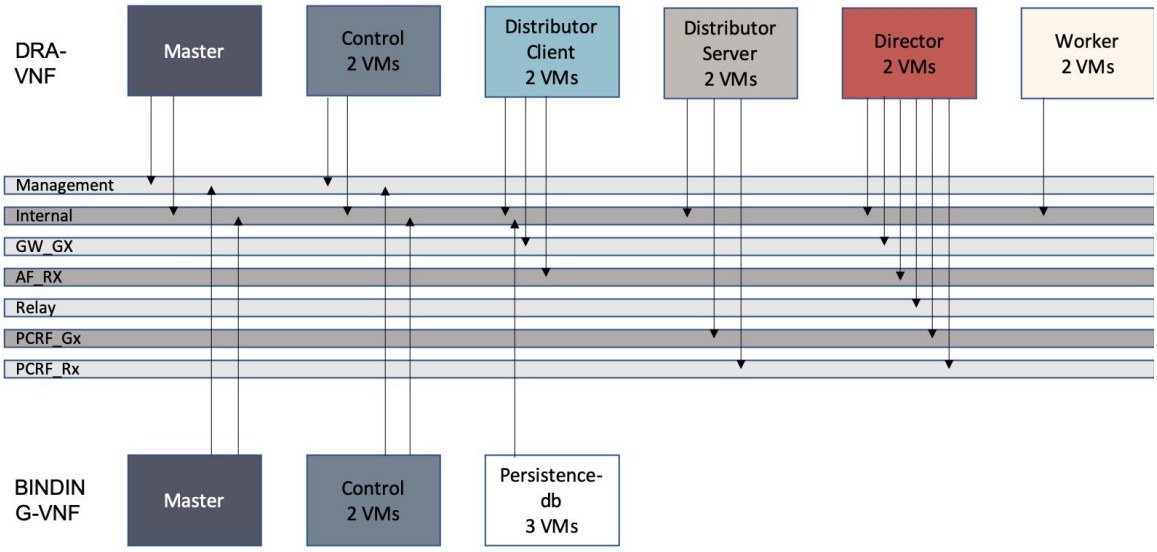
- Create a vDRA installer VM in vSphere using the vDRA Deployer Host VMDK.
- Create the artifacts that describe the VM roles, CPS ISO (dra-vnf or binding-vnf), IP addresses, hostnames, target ESXi servers, and so on.
- Run the `cps install <vnf directory>` command.

Note The ESXi servers must be configured to use the Network Time Protocol (NTP) to synchronize their clocks.

Sample vDRA System

The following network diagram, configuration and VM layout are for illustration purposes only. Contact Cisco Account representative for your specific vDRA requirements.

Figure 1: Sample vDRA System



Installation Order

The following installation order should be used:

1. Binding VNF
2. DRA VNF

Note VMs per VNF must be installed in parallel. There are no VM ordering requirements while installing a vDRA VNF.

Requirements

Note For blade requirements, contact your Cisco Account representative.

Virtual Machine (VM)

The table list the VM requirements for vDRA:

Table 1: VM Requirements

Role	vCPU	RAM (GB)	Primary Disk (GB)	Data Disk (GB)
master	16	64	100	200

Role	vCPU	RAM (GB)	Primary Disk (GB)	Data Disk (GB)
control	16	64	100	200
dra-director	40	128	100	-
dra-distributor	16	32	100	-
dra-worker	16	128	100	-
persistence-db	8	64	100	-
Installer	8	32	100	-

vSphere

vSphere 6.5

ESXi Servers

- UCSB-B200-M5
- 512 GB RAM
- 2 SSD Drivers
- 2 CPUs with 28 cores each
- NTP Enabled

VMware Interface Name and Order

In VMware, the NETWORK definition from the `env` files map to the following Linux interface names:

Table 2: Network Definition Mapping to Linux Interface Name

NETWORK_	Linux Interface Name
0	ens160
1	ens192
2	ens224
3	ens256
4	ens161
5	ens193
6	ens225
7	ens257
8	ens162

NETWORK_	Linux Interface Name
9	ens194

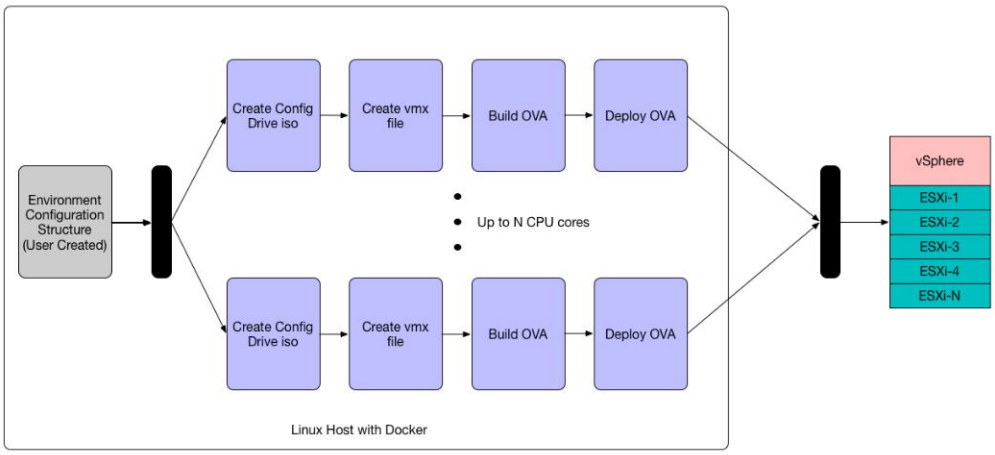
Environment Artifacts

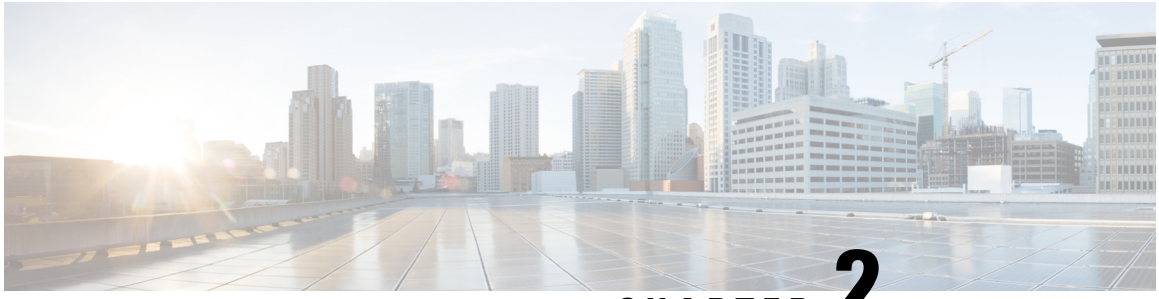
You can specify the test bed configuration settings for global, role, and VM in increasing precedence using a directory structure and files containing key-value environment variables.

The Jinja2 templates are used to create user_data files for cloud-init, ovftool options, and VMware Virtual Machine VMX configuration files. The environment variables are applied to the various Jinja2 template files using `envtpl`.

The installer loops over the directory structure sourcing global environment, role environment, and finally VM environment settings. Once at the VM level, the installer applies the environment variables to the Jinja2 templates to create the cloud-init configuration drive files (meta_data.json, user_data, and interfaces file (content/0000)), the VMX files for creating OVAs, and `ovftool` command line options. The VM artifacts are stored in `data/vmware/<vm name>`.

Figure 2: Installer Flow





CHAPTER 2

Installing CPS vDRA

Create Installer VM in vSphere

Create the installer VM in VMware vSphere.

Download the vDRA deployer VMDKs and base image VMDKs.

Upload the VDMK File

Upload the VDMK file as shown in the following example:

```
ssh root@my-esxi-1.cisco.com
cd /vmfs/volumes/<datastore>
mkdir cps-images
cd /vmfs/volumes/<datastore>/cps-images
wget http:/<your_host>/cps-deployer-host_<version>.vmdk
```

Convert CPS Deployer VMDK to ESXi Format

Convert the CPS deployer host VMDK to ESXi format as shown in the following example:

```
ssh root@my-esxi-1.cisco.com
cd /vmfs/volumes/<datastore>/cps-images
vmkfstools --diskformat thin -i cps-deployer-host_<version>.vmdk
cps-deployer-host_<version>-esxi.vmdk
```

Create CPS Installer VM

Using the vSphere client, create the CPS Installer VM.

Step 1 Log into vSphere and select **Hosts and Clusters**.

Step 2 Select the target EXSi host.

Step 3 Select **Actions** > **New Virtual Machine**.

Step 4 Select **Create a new virtual machine** and click **Next**.

Step 5 Enter a name for the virtual machine (for example, cps-installer) and select the location for the virtual machine. Click **Next**.

Step 6 Select blade IP address from **Select a compute resource** window and click **Next** to open **Select storage** window.

Step 7 From **Select storage** window, select *datastorename* and click **Next** to open **Select compatibility** window.

Step 8 From **Compatible with:** drop-down list, select **`ESXi 6.0 and later`** and click **Next** to open **Select a guest OS** window.

Step 9 From **Guest OS Family:** drop-down list, select **`Linux`** and from **Guest OS Version:** drop-down list, select **`Ubuntu Linux (64-bit)`**.

Step 10 Click **Next** to open **Customize hardware** window.

Step 11 In **Virtual Hardware** tab:

a) Select 4 CPUs.
b) Select **Memory** size as **`32 GB`**.
c) Delete **New Hard Disk** (VM will use the existing disk created earlier with vmkfstools command).
d) Select **Management** network from the **`New Network`** drop-down list and check **Connect At Power On**.

Step 12 Create hard disk.

a) Select **Existing Hard Disk** from **New Device:** drop-down list and click **`ADD`**.
b) Navigate to **`cps-deployer-host_<version>-esxi.vmdk`** file created earlier with the vmkfstools command and click **OK**.
c) Click **Next**.
d) Click **Finish**.

Step 13 Adjust hard disk size.

a) Press **Ctrl + Alt +2** to go back to **Hosts and Clusters** and select the VM created above (*cps-installer*).
b) Right-click and select **Edit Settings..**. **Virtual Hardware** tab is displayed as default.
c) In the **Hard disk 1** text box enter **`100`** and click **OK**.

Step 14 Power ON the VM and open the console.

Configure Network

Step 1 Log into the VM Console as user: cps, password: cisco123.

Step 2 Create the `/etc/network/interfaces` file using `vi` or using the `here document` syntax as shown in the example:

```
cps@ubuntu:~$ sudo -i
root@ubuntu:~# cat > /etc/network/interfaces <<EOF
auto lo
iface lo inet loopback

auto ens160
iface ens160 inet static
address 10.10.10.5
netmask 255.255.255.0
gateway 10.10.10.1
dns-nameservers 192.168.1.2
dns-search cisco.com
```

```
EOF
root@ubuntu:~#
```

Step 3 Restart networking as shown in the following example:

```
root@ubuntu:~# systemctl restart networking
root@ubuntu:~# ifdown ens160
root@ubuntu:~# ifup ens160
root@ubuntu:~# exit
cps@ubuntu:~$
```

What to do next

You can log in remotely using the SSH login cps/cisco123.

Binding-VNF

The process for installing the binding-vnf is the same as the dra-vnf. Create the configuration artifacts for the binding-vnf using the same VMDK. But use the binding ISO instead of DRA ISO. Similar to the dra-vnf, add a 200 GB data disk to the master and control VMs.

Artifacts Structure

```
cps@installer:/data/deployer/envs/binding-vnf$ tree
.
|-- base.env
|-- base.esxi.env
|-- user_data.yml
|-- user_data.yml.pam
`-- vms
    |-- control-0
    |   |-- control-binding-0
    |   |   |-- interfaces.esxi
    |   |   |-- user_data.yml
    |   |   |-- user_data.yml.pam
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- role.env
    |   `-- role.esxi.env
    |-- control-1
    |   |-- control-binding-1
    |   |   |-- interfaces.esxi
    |   |   |-- user_data.yml
    |   |   |-- user_data.yml.pam
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- role.env
    |   |-- role.esxi.env
    |   `-- user_data.yml.disk
    |-- master
    |   |-- master-binding-0
    |   |   |-- interfaces.esxi
    |   |   |-- user_data.yml
    |   |   |-- user_data.yml.functions
    |   |   |-- user_data.yml.pam
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
```

```
    |   |-- role.env
    |   `-- role.esxi.env
    `-- persistence-db
        |-- persistence-db-1
        |   |-- interfaces.esxi
        |   |-- vm.env
        |   `-- vm.esxi.env
        |-- persistence-db-2
        |   |-- interfaces.esxi
        |   |-- vm.env
        |   `-- vm.esxi.env
        |-- persistence-db-3
        |   |-- interfaces.esxi
        |   |-- vm.env
        |   `-- vm.esxi.env
        |-- role.env
        `-- role.esxi.env

11 directories, 38 files
cps@installer:/data/deployer/envs/binding-vnf$
```

cps Installer Commands

Command Usage

Use the `cps` command to deploy VMs. The command is a wrapper around the docker command required to run the deployer container.

Example:

```
function cps () {
     docker run \
         -v /data/deployer:/data/deployer \
         -v /data/vmware/:/export/ \
         -it --rm dockerhub.cisco.com/cps-docker-v2/cps deployer/deployer:latest  \
         /root/cps "$@"
}
```

To view the help for the command, run the following command: `cps -h`

```
cps@installer:~$ cps -h
usage: cps [-h] [--artifacts_abs_root_path ARTIFACTS_ABS_ROOT_PATH]
           [--export_dir EXPORT_DIR] [--deploy_type DEPLOY_TYPE]
           [--template_dir TEMPLATE_DIR]
           [--status_table_width STATUS_TABLE_WIDTH] [--skip_create_ova]
           [--skip_delete_ova]
           {install,delete,redeploy,list,poweroff,poweron,datadisk}
           vnf_artifacts_relative_path [vm_name [vm_name ...]]

positional arguments:
  {install,delete,redeploy,list,poweroff,poweron,datadisk}
                        Action to perform
  vnf_artifacts_relative_path
                        VNF artifacts directory relative to vnf artifacts root
                        path. Example: dra-vnf
  vm_name               name of virtual machine

optional arguments:
  -h, --help            show this help message and exit
  --artifacts_abs_root_path ARTIFACTS_ABS_ROOT_PATH
                        Absolute path to artifacts root path. Example:
```

```
                      /data/deployer/envs
  --export_dir EXPORT_DIR
                      Abosolute path to store ova files and rendered
                      templates
  --deploy_type DEPLOY_TYPE
                      esxi
  --template_dir TEMPLATE_DIR
                      Absolute path to default templates
  --status_table_width STATUS_TABLE_WIDTH
                      Number of VMs displayed per row in vm status table
  --skip_create_ova   Skip the creation of ova files. If this option is
                      used, the ova files must be pre-created. This if for
                      testing and debugging
  --skip_delete_ova   Skip the deletion of ova files. If this option is
                      used, the ova files are not deleted. This if for
                      testing and debugging
```

List VMs in Artifacts

Use the following command to list VMs in artifacts:

```
cps list example-dra-vnf
```

where, *example-dra-vnf* is the VNF artifacts directory.

Deploy all VMs in Parallel

Use the following command to deploy all VMs in parallel:

```
cps install example-dra-vnf
```

Deploy one or more VMs

The following example command shows how to deploy dra-director-2 and dra-worker-1:

```
cps install example-dra-vnf dra-director-2 dra-worker-1
```

Delete one or more VMs

The following command is an example for deleting dra-director-1 and dra-worker-1 VMs:

Note VM deletion can disrupt services.

```
cps delete example-dra-vnf dra-director-1 dra-worker-1
```

Redeploy all VMs

Redeploying VMs involves deleting a VM and then redeploying them. If more the one VM is specified, VMs are processed serially. The following command is an example for redeploing all VMs:

Note VM deletion can disrupt services.

```
cps redeploy example-dra-vnf
```

Redeploy one or more VMs

Redeploying VMs involves deleting a VM and then redeploying them. If more the one VM is specified, VMs are processed serially. The following command is an example for redeploing two VMs:

Note VM deletion can disrupt services.

```
cps redeploy example-dra-vnf dra-director-1 control-1
```

Power down one or more VMs

The following command is an example for powering down two VMs:

Note Powering down the VM can disrupt services.

```
cps poweroff example-dra-vnf dra-director-1 dra-worker-1
```

Power up one or more VMs

The following command is an example for powering up two VMs:

Note Powering Up the VM can disrupt services.

```
cps poweron example-dra-vnf dra-director-1 dra-worker-1
```

Validate Deployment

Use the CLI on the master VM to validate the installation.

Connect to the CLI using the default user and password (admin/admin).

```
ssh -p 2024 admin@<master management ip address>
```

show system status

Use `show system status` command to display the system status.

Note System status percent-complete should be 100%.

```
admin@orchestrator[master-0]# show system status
system status running     true
system status upgrade     false
system status downgrade   false
system status external-services-enabled true
system status debug       false
```

```
system status percent-complete 100.0
admin@orchestrator[master-0]#
```

show system diagnostics

No diagnostic messages should appear using the following command:

```
admin@orchestrator[master-0]# show system diagnostics | tab | exclude pass
NODE       CHECK ID                        IDX  STATUS   MESSAGE
---------------------------------------------------------------

admin@orchestrator[master-0]#
```

show docker engine

All DRA-VNF VMs should be listed and in the CONNECTED state.

```
admin@orchestrator[master-0]# show docker engine
                              MISSED
ID                 STATUS     PINGS
-------------------------------------
control-0          CONNECTED  0
control-1          CONNECTED  0
dra-director-1     CONNECTED  0
dra-director-2     CONNECTED  0
dra-distributor-1  CONNECTED  0
dra-distributor-2  CONNECTED  0
dra-worker-1       CONNECTED  0
dra-worker-2       CONNECTED  0
master-0           CONNECTED  0

admin@orchestrator[master-0]#
```

show docker service

No containers should be displayed when using the exclude HEAL filter.

```
admin@orchestrator[master-0]# show docker service | tab | exclude HEAL
                                                          PENALTY
MODULE  INSTANCE NAME  VERSION  ENGINE  CONTAINER ID  STATE  BOX     MESSAGE
-------------------------------------------------------------------------------

admin@orchestrator[master-0]#
```

APPENDIX A

Installation Examples

DRA-VNF Example

This section provides an example for configuring the installer with a dra-vnf test bed. The dra-vnf example includes the following roles and VMs:

- master:

  master-0

- control:

  control-0

  control-1

- DRA Director:

  dra-director-1

  dra-director-2

- DRA Worker:

  dra-worker-1

  dra-worker-2

- DRA Distributor:

  dra-distributor-1

  dra-distributor-2

  dra-distributor-3

  dra-distributor-4

Artifacts Structure Example

```
cps@installer:/data/deployer/envs/dra-vnf$ tree
.
```

```
|-- base.env
|-- base.esxi.env
|-- user_data.yml
|-- user_data.yml.pam
`-- vms
    |-- control-0
    |   |-- control-0
    |   |   |-- interfaces.esxi
    |   |   |-- user_data.yml
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- role.env
    |   `-- role.esxi.env
    |-- control-1
    |   |-- control-1
    |   |   |-- interfaces.esxi
    |   |   |-- user_data.yml
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- role.env
    |   `-- role.esxi.env
    |-- dra-director
    |   |-- dra-director-1
    |   |   |-- interfaces.esxi
    |   |   |-- user_data.yml
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- dra-director-2
    |   |   |-- interfaces.esxi
    |   |   |-- user_data.yml
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- role.env
    |   `-- role.esxi.env
    |-- dra-distributor
    |   |-- dra-distributor-1
    |   |   |-- interfaces.esxi
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- dra-distributor-2
    |   |   |-- interfaces.esxi
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- dra-distributor-3
    |   |   |-- interfaces.esxi
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- dra-distributor-4
    |   |   |-- interfaces.esxi
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- role.env
    |   |-- role.esxi.env
    |   `-- user_data.yml
    |-- dra-worker
    |   |-- dra-worker-1
    |   |   |-- interfaces.esxi
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- dra-worker-2
    |   |   |-- interfaces.esxi
    |   |   |-- vm.env
    |   |   `-- vm.esxi.env
    |   |-- role.env
```

```
    |   `-- role.esxi.env
    `-- master
        |-- master-0
        |   |-- interfaces.esxi
        |   |-- user_data.yml
        |   |-- vm.env
        |   `-- vm.esxi.env
        |-- role.env
        `-- role.esxi.env

18 directories, 55 files
cps@installer:/data/deployer/envs/dra-vnf$
```

Top Level Directory

```
/data/deployer/envs/example-dra-vnf/base.env
/data/deployer/envs/example-dra-vnf/base.esxi.env
/data/deployer/envs/example-dra-vnf/user_data.yml
/data/deployer/envs/example-dra-vnf/base.esxi.env
/data/deployer/envs/example-dra-vnf/esxi
/data/deployer/envs/example-dra-vnf/vms
```

base.env

All the settings in the `base.env` file can be overridden in `vms/`*`role`*`/role.env` and `vms/`*`role/vm_name`*`/vm.env` files.

```
MASTER_IP=192.169.21.10
INTERNAL_NETWORK=192.169.21.0/24
WEAVE_PASSWORD=cisco123
CLUSTER_ID=test-cluster
SYSTEM_ID=test-system
```

MASTER_IP: Internal address of master VM.

base.esxi.env

All the settings in the `base.esxi.env` file can be overridden in the `vms/`*`role`*`/role.esxi.env` and `vms/`*`role/vm_name`*`/vm.esxi.env` files.

```
VMDK="cps-docker-host_18.0.1.dra.vmdk"
VMDK_DISK_TYPE="thick"
VSPHERE_HOST="example-vsphere.cisco.com"
VSPHERE_USER="administrator@vsphere.local"
VSPHERE_PASSWORD="foo123"
VSPHERE_DISABLE_SSL_VERIFICATION="True"
VSPHERE_RESERVE_MEMORY="True"
DATACENTER="Microservices"
```

- VMDK: Place the VMDK file at the top level directory of your VNF environment structure `example-dra-vnf/`*`microservices.vmdk_file_name`*.

  Another option is to specify the full path such as `/data/deployer/envs/images/`*`microservices.vmdk_file_name`*

  Replace *microservices.vmdk_file_name* with the actual VMDK file name.

- VMDK_DISK_TYPE: VMDK disk type. See the link for a list of supported disk types.

- VSPHERE_HOST: DNS name or IP address of the vSphere host.

- VSPHERE_USER: (Optional) Login user for vSphere. If the user name is not specified, installer prompts user for vSphere login user name.
- VSPHERE_PASSWORD: (Optional) vSphere password. If the password is not specified, installer prompts user for password
- VSPHERE_DISABLE_SSL_VERIFICATION: (Optional) Disable verification of vSphere SSL Certificate. This is necessary if your vSphere server is using a Self Signed Certificate
- VSPHERE_RESERVE_MEMORY: (Optional) Reserve VM's memory before starting the VM
- DATACENTER: Datacenter for VM placement.

user_data.yml

Use the Jinja2 template to create the user data file for cloud-init.

Cloud-init user data template: This file is for reference only. You need to create cloud-init file based on your requirements.

```
#cloud-config
debug: True
output: {all: '| tee -a /var/log/cloud-init-output.log'}

users:
  - name: cps
    sudo: ['ALL=(ALL) NOPASSWD:ALL']
    groups: docker
    ssh-authorized-keys:
      - ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDzjJjndIvUiBta4VSIbd2gJmlMWcQ8wtejg
        AbiXtoFZdtMdo9G0ZDEOtxHNNDPwWujMiYAkZhZWX/zON9raavU8lg cps@root-public-key

resize_rootfs: true

write_files:
  - path: /root/swarm.json
    content: |
     {
        "role": "{{ ROLE }}",
        "identifier": "{{ IDENTIFIER }}",
        "master": "{{ MASTER_IP }}",
        "network": "{{ INTERNAL_NETWORK }}",
        {% if WEAVE_PASSWORD is defined %}"weavePw": "{{ WEAVE_PASSWORD }}", {% endif %}
        "zing": "{{ RUN_ZING | default(1) }}",
        "cluster_id": "{{ CLUSTER_ID }}",
        "system_id": "{{ SYSTEM_ID }}"
     }
    owner: root:root
    permissions: '0644'
  - path: /home/cps/.bash_aliases
    encoding: text/plain
    content: |
      # A convenient shortcut to get to the Orchestrator CLI
      alias cli="ssh -p 2024 admin@localhost"
    owner: cps:cps
    permissions: '0644'
runcmd:
 - [vmware-toolbox-cmd, timesync, enable ]
```

example-dra-vnf/vms/*role*

```
example-dra-vnf/master/role.env
example-dra-vnf/master/role.esxi.env
example-dra-vnf/master/master-0
```

role.env

All settings in the `role.env` file can be overridden in the `vms/role/vm_name/vm.env` file. In non-master roles the `role.env` file is empty.

```
CPS_ISO="cisco-policy-dra.iso"
```

where, *CPS_ISO* is the CPS ISO file. This is required for master virtual machines.

Not used in non-master virtual machines. It is possible to specify this with a full path `/data/deployer/envs/images/cisco-policy-dra.iso`.

role.esxi.env

All settings in the `role.esxi.env` file can be overridden in the `vms/vm_name/vm.esxi.env` file.

```
CPU=16
RAM=65536
NETWORK_0=Management
NETWORK_1=Internal
# Data disk size in GB
VM_DATA_DISK_SIZE="200"
VM_DATA_DISK_TYPE="thick"
```

- CPU: Number of CPUs.
- RAM: Memory in megabytes (65536/1024 = 64 GB)
- NETWORK_0: The name of the first network assigned to the VM. Name is case sensitive and must match the network name configured in vSphere. Network interface names are defined using the scheme in "Interface Numbering" section.

  Add a NETWORK_*N* setting for each network required.
- VM_DATA_DISK_SIZE: Data disk size in GB for master and control VMs.
- VM_DATA_DISK_TYPE: VM data disk type. See the link for a list of supported disk types.

Data Disk

A data disk is a separate disk for the control and master virtual machines and is configured in the artifacts environment files before installing a CPS system. The data has a `/data` partition and a `/stats` partition. Perform the following steps to add a data disk to master and control VMs.

- Specify VM_DATA_DISK_SIZE and VM_DATA_DISK_TYPE in `example-env/vms/<role>/role.esxi.env` file.
- Specify VM_DATA_VMDK_ROOT_PATH and VM_DATA_DISK_NAME in `example-env/vms/<role>/role.esxi.env` file.
- Specify disk file system and mount point in `example-env/vms/<role>/<vm_name>/user_data.yml` file.

The installer checks for an existing data disk in `VM_DATA_VMDK_ROOT_PATH/<disk_name>`. If a data disk exists, the disk is attached to the target VM. If a data disk does not exist, the installer creates a new VMDK disk and attaches it to the VM. Cloud init is responsible for formatting the disk and mounting it. If the data disk has an ext-4 file system, cloud-int does not reformat the disk, preserving existing data.

If a VM is deleted with the deployer container's `cps delete example-dra control-0` command, the data disk is detached before the VM is deleted. Detached disks are not deleted when the VM is deleted.

master-0

The master-0 directory is the name of a VM. This directory name must match the hostname of the VM.

```
example-dra-vnf/vms/master/vm_name
```

Directory containing configuration information for a VM

```
example-dra-vnf/vms/master/master-0/interfaces.esxi
example-dra-vnf/vms/master/master-0/vm.env
example-dra-vnf/vms/master/master-0/vm.esxi.env
```

interfaces.esxi

The contents of the `interfaces.esxi` file are placed in `/etc/network/interfaces` file on the VM. Any valid content for the ubuntu `/etc/network/interfaces` file can be placed in `interfaces.esxi`.

```
auto lo
iface lo inet loopback

auto ens160
iface ens160 inet static
address 10.10.10.155
netmask 255.255.255.0
gateway 10.10.10.1
dns-nameservers 172.10.5.25 172.11.5.25 172.12.5.25

auto ens192
iface ens192 inet static
address 192.169.21.10
netmask 255.255.255.0
```

vm.env

```
HOSTNAME=master-0
FQDN=master-0.local
```

vm.esxi.env

```
ESXI_DNS_NAME="example-esxi-1.cisco.com"
DATASTORE="datastore1"
VM_DATA_VMDK_ROOT_PATH="[datastore1] data-disks"
VM_DATA_DISK_NAME="master-0-data.vmdk"
```

- ESXI_DNS_NAME: DNS name of the VM's target ESXi server.
- ESXI_IP: IP address of ESXi server. This can be used instead of ESXI_DNS_NAME. If both, ESXI_DNS_NAME and ESXI_IP are specified, ESXI_DNS_NAME is used.

vCenter always directs the API client to the DNS name of the target ESXi server regardless if the EXSi host's IP address or DNS name is specified. The installation fails if the deployer VM cannot resolve the ESXi's DNS

name. To avoid this, update the "cps" bash function in the file `/etc/bash.aliases` and add `--add-host <esxi dns name>:<ip address>` for each ESXi server. Use `sudo` to modify the file.

```
/etc/bash.aliases
function cps () {
     docker run \
         --add-host esxi-1.example.com:10.0.0.1 \
         --add-host esxi-2.example.com:10.0.0.2 \
         -v /data/deployer:/data/deployer \
         -v /data/vmware/:/export/ \
         -it --rm dockerhub.cisco.com/cps-docker-v2/cps-deployer/deployer:latest \
         /root/cps "$@"
}
```

- DATASTORE: Case sensitive name of the vSphere datastore used to store the VM.
- VM_DATA_VMDK_ROOT_PATH: Root path to store the master or control VM's data disk.
- VM_DATA_DISK_NAME: Name of the VMDK disk.

VM Level user_data.yml for Data Disks

Place this file at the VM level for master and control VMs when using a separate data disks.

Note This file is for reference only. You need to create `user_data.yml` file based on your requirements.

```
#cloud-config
# ESC velocity escape variable during deployment
#set ( $DS = "$" )
debug: True
output: {all: '| tee -a /var/log/cloud-init-output.log'}

users:
  - name: cps
    sudo: ['ALL=(ALL) NOPASSWD:ALL']
    groups: docker
    ssh-authorized-keys:
      - ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDzjJjndIvUiBta4VSIbd2g
        JmlMWcQ8wtejgAbiXtoFZdtMdo9G0ZDEOtxHNNDPwWujMiYAkZhZWX/zON9raav
        U8lgD9+YcRopWUtujIC71YjtoxIj   EWEaj/50jegN cps@root-public-key

resize_rootfs: true

write_files:
  - path: /root/swarm.json
    content: |
     {
        "role": "{{ ROLE }}",
        "identifier": "{{ IDENTIFIER }}",
        "master": "{{ MASTER_IP }}",
        "network": "{{ INTERNAL_NETWORK }}",
        {% if WEAVE_PASSWORD is defined %}"weavePw": "{{ WEAVE_PASSWORD }}", {% endif %}
        "zing": "{{ RUN_ZING | default(1) }}",
        "cluster_id": "{{ CLUSTER_ID }}",
        "system_id": "{{ SYSTEM_ID }}"
     }
    owner: root:root
    permissions: '0644'
  - path: /home/cps/.bash_aliases
    encoding: text/plain
```

```
    content: |
      # A convenient shortcut to get to the Orchestrator CLI
      alias cli="ssh -p 2024 admin@localhost"
      alias pem="wget --quiet http://171.70.34.121/microservices/latest/cps.pem ; chmod 400
 cps.pem ; echo 'Retrieved \"cps.pem\" key file'"
    owner: cps:cps
    permissions: '0644'

disk_setup:
  /dev/sdb:
      table_type: 'gpt'
      layout:
          - 35
          - 65
      overwrite: False
fs_setup:
  - label: DATA
    device: /dev/sdb
    filesystem: 'ext4'
    partition: auto
    overwrite: False
  - label: STATS
    device: /dev/sdb
    filesystem: 'ext4'
    partition: auto
    overwrite: False

mounts:
 - [ "LABEL=DATA", /data, "ext4", "defaults,nofail", "0", "2" ]
 - [ "LABEL=STATS", /stats, "ext4", "defaults,nofail", "0", "2" ]
runcmd:
 - [vmware-toolbox-cmd, timesync, enable ]
```