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	Login to the vSphere Web Client and select the blade where you want to create a new VM to install the cluster manager VM.
Step 2	Right-click on the blade and select **New Virtual Machine**. **New Virtual Machine** window opens up.
Step 3	Select **Create a new virtual machine** and click **Next** to open **Select a name and folder**.
Step 4	Enter a name for the virtual machine (for example, CPS Cluster Manager) and select the location for the virtual machine. Click **Next**.
Step 5	Select blade IP address from **Select a compute resource** window and click **Next** to open **Select storage** window.
Step 6	From **Select storage** window, select *datastorename* and click **Next** to open **Select compatibility** window.
Step 7	From **Compatible with:** drop-down list, select `ESXi 6.7 and later` and click **Next** to open **Select a guest OS** window.

> **Note** Support for VMX11 is added only for fresh install. For upgrade flow (option 2/option 3), upgrade of VMX is not supported.

Step 8	From **Guest OS Family:** drop-down list, select `Linux` and from **Guest OS Version:** drop-down list, select `Ubuntu Linux (64-bit)`.
Step 9	Click **Next** to open **Customize hardware** window.
Step 10	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) Expand **New SCSI controller** and from **Change Type** drop-down list, select **VMware Paravirtual**.
e) 2 NICs are required (one for eth1 as internal and second for eth2 as management). One NIC already exists as default under **New Network**.

   Under **New Network**, check **Connect At Power On** is selected.

f) To add another NIC, click **ADD NEW DEVICE** and from the list select **Network Adapter**.

   Under **New Network**, check **Connect At Power On** is selected.

g) Click **Next** to open **Ready to complete** window.

Step 11	Review the settings displayed on **Ready to complete** window and click **Finish**.
Step 12	Press **Ctrl** + **Alt** +**2** to go back to **Hosts and Clusters** and select the VM created above (*CPS Cluster Manager*).

a) Right-click and select **Edit Settings...**. **Virtual Hardware** tab is displayed as default.
b) Click **ADD NEW DEVICE** and from the list select **Existing Hard Disk** to open **Select File** window.
c) Navigate to `cps-deployer-host_<version>-esxi.vmdk` file created earlier with the vmkfstools command and click **OK**.

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 Cluster Manager*).
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 that is 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
```

Deploy all VMs with or without a Hypervisor Flag

Use the following command to install all VMs that are tagged with a ESXIHOST value matching hypervisor name as esxi-host-1 in their *vm.esxi.env* file:

```
cps install dra-vnf --hypervisor esxi-host-1
```

The following cps install command allows you to perform activities on more than one artifiactory files, which are tagged with or without *--hypervisor* flag.

```
cps install —addartifact artifact-env-2
--hypervisor hypervisor-name
```

Health Checks

Using the **--hypervisor** option that you can perform health check of docker engine and consul status of other VMs before making changes on the requested VM.

For example, if you run cps install **--hypervisor** *esxi-host-1*, then any VMs that are tagged with *esxi-host-1* are excluded and the remaining set of VMs from the artifact file is considered for health check.

VM Name	ESXiHOST
vm01	esxi-host-1
vm02	esxi-host-2
vm03	esxi-host-2

This is done to ensure that VM's on other blades are stable before performing the requested changes on their partner blade VMs. The health check fetches details of the master VM automatically from the artifiactory file and performs SSH to master, to check if the docker engine and consul status of vm02 and vm03 are in a proper state. If the state is proper, then *cps* command starts the requested operation such as install, power on, or redeploy and so on.

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
```

Upgrading VMs using Diagnostics and Redeployment Health Check

Diagnostics of VMs

Use the following command to perform system diagnostics on VMs from vDRA to DB VNFs.

```
cps diagnostics dra-vnf
```

Redeployment Health Check for VMs

Use the following command to perform the redeployment health check on VMs.

```
cps redeploy dra-vnf --healthcheck yes --sysenv dra
```

Ranking Details

To upgrade the VMs, create a group of specific VMs from artifact files and place it under
`/data/deployer/envs/upgradelist.txt`. It is a one-time creation process and the file has a ranking
mechanism.

Based on ranking, separate the contents with a comma(,) as given.

```
Example:

cat /data/deployer/envs/upgradelist.txt
1,sk-master0
2,sk-control0,sk-dra-worker2
3,sk-control1,sk-dra-worker1
4,sk-dra-directo1,sk-dra-director2
```

The pre and postchecks for Master and Control VMs vary from other VMs.

Ranking Details		
Rank 1	Master VM Example: 1,sk-master0	If there is no master VM, then remove Rank1(1,sk-master0) from the upgradelist.txt file not to disturb the other ranks.
Rank 2 **Rank 3**	Control VM Example: 2,sk-control0, sk-dra-worker2 3,sk-control1, sk-dra-worker1	• Declare the control VMs for Ranks 2 and 3 and add one or more VMs. • If you do not redeploy control VMs, do not declare any values in the upgradelist.txt file starting with Rank 2 and 3.
Rank 4	Other VMs Example: 4,sk-dra-directo1,sk-dra-director2	Do not contain either master or control VMs.

The differentiation between Rank 1(Master) and Rank2(Control) VMs is because the pre and postchecks for Master and Control VMs varies withing themselves.

Resume Redeployment

The resume option starts the VM redeployment from the last successful completion.

Consider the following scenario where the deployment occurs until site2-binding-control-0. For some reason, the VMs after site2-binding-control0 faces a problem and the automation feature terminates the execution.

```
root@ubuntu:~# cat /data/deployer/envs/upgradelist.txt
1,site2-binding-master-1
2,site2-binding-control-0,site2-persistence-db-1
3,site2-binding-control-1,site2-persistence-db-2
```

Use the **cps redeploy /data/deployer/envs/dba-vnf/ --healthcheck yes --sysenv dba** command to resume the redeployment.

Configuration and Restriction:

• The diagnostics and redeployment of VMs with the health check works only if the Master VM is active.

• For a proper health check, copy the cps.pem key used for connecting to the Master VM to the `/data/deployer/envs` folder.

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]#
```

Redeploy VMs during the ISSM Operation

To redeploy VMs during In-Service Software Migration (ISSM) , use the following procedure:

Step 1   Find the consul container that is having a consul leader role:

a) To find the consul leader use the following command:

 # docker exec consul-1 consul operator raft list-peers

For example, in the following output consul-3 is the leader.

```
admin@orchestrator[an-master]# docker exec consul-1 "consul operator raft list-peers"
==========output from container consul-1==========
Node                ID                                    Address           State    Voter
RaftProtocol
consul-2.weave.local  52d5b25c-77fc-1163-0304-493b117096cd  10.46.128.2:8300  follower  true  3
consul-4.weave.local  fe68543b-ef72-66a7-7830-1c0405fd06a0  10.32.128.1:8300  follower  true  3
consul-5.weave.local  21539d8a-7d55-9cdb-c3e0-7680b448b5d5  10.32.160.1:8300  follower  true  3
consul-3.weave.local  f7a87957-a129-a12e-eb44-03bc3b385ec1  10.46.160.2:8300  leader    true  3
consul-1.weave.local  2d14416d-cc22-bcbd-e686-04bdc860332d  10.32.0.3:8300    follower  true  3
consul-7.weave.local  a3b0ba51-a8d4-68b4-b899-c20ede286e09  10.47.160.1:8300  follower  true  3
consul-6.weave.local  36d06c94-2ec5-094d-7acf-7ea190b36825  10.46.224.1:8300  follower  true  3
admin@orchestrator[an-master]#
```

Step 2   Use the following command to find the VM in which the consul leader is running:

show docker service | tab | include consul

For example, in the following output the consul leader is running in the director-0 vm.

```
admin@orchestrator[an-master]# show docker service | tab | include consul
consul            1         consul-1               23.2.0-release  an-master        consul-1
                  HEALTHY  false    -
consul            1         consul-2               23.2.0-release  an-control-0     consul-2
                  HEALTHY  false    -
consul            1         consul-3               23.2.0-release  an-control-1     consul-3
                  HEALTHY  false    -
consul-dra        1         consul-4               23.2.0-release  an-dra-director-0 consul-4
                  HEALTHY  false    -
consul-dra        1         consul-5               23.2.0-release  an-dra-director-1 consul-5
                  HEALTHY  false    -
consul-dra        1         consul-6               23.2.0-release  an-dra-worker-0   consul-6
                  HEALTHY  false    -
consul-dra        1         consul-7               23.2.0-release  an-dra-worker-1   consul-7
                  HEALTHY  false    -
admin@orchestrator[an-master]#
```

Step 3   Perform consul leader failover in the consul leader container using **docker exec <consul-leader-container> "supervisorctl stop consul-server"** command .

Example: If the consul leader VM is same as the VM to be redeployed, then stop the consul-server in the consul leader container to perform consul leader failover.

```
admin@orchestrator[an-master]# docker exec consul-3 "supervisorctl stop consul-server"
==========output from container consul-3==========
consul-server: stopped
admin@orchestrator[an-master]#
```

Step 4 Verify the consul leader failover with another VM that will not be redeployed. Use the **docker exec consul-1 "consul operator raft list-peers"** command to verify the details as shown in the sample configuration.

```
admin@orchestrator[an-master]# docker exec consul-1 "consul operator raft list-peers"
==========output from container consul-1==========
Node                 ID                                    Address            State      Voter
RaftProtocol
consul-2.weave.local  52d5b25c-77fc-1163-0304-493b117096cd  10.46.128.2:8300   follower   true   3
consul-4.weave.local  fe68543b-ef72-66a7-7830-1c0405fd06a0  10.32.128.1:8300   leader     true   3
consul-5.weave.local  21539d8a-7d55-9cdb-c3e0-7680b448b5d5  10.32.160.1:8300   follower   true   3
consul-3.weave.local  f7a87957-a129-a12e-eb44-03bc3b385ec1  10.46.160.2:8300   follower   true   3
consul-1.weave.local  2d14416d-cc22-bcbd-e686-04bdc860332d  10.32.0.3:8300     follower   true   3
consul-7.weave.local  a3b0ba51-a8d4-68b4-b899-c20ede286e09  10.47.160.1:8300   follower   true   3
consul-6.weave.local  36d06c94-2ec5-094d-7acf-7ea190b36825  10.46.224.1:8300   follower   true   3
admin@orchestrator[an-master]#
```

Step 5 Start the consul server in the consul container stopped in Step 3.

Step 6 Verify the health of the consul using the **show docker service | tab | include consul** command to ensure that the consul containers are healthy after consul leader failover.

```
admin@orchestrator[an-master]# show docker service | tab | include consul
consul              1         consul-1                 23.2.0-release  an-master         consul-1
                    HEALTHY  false    -
consul              1         consul-2                 23.2.0-release  an-control-0      consul-2
                    HEALTHY  false    -
consul              1         consul-3                 23.2.0-release  an-control-1      consul-3
                    HEALTHY  false    -
consul-dra          1         consul-4                 23.2.0-release  an-dra-director-0 consul-4
                    HEALTHY  false    -
consul-dra          1         consul-5                 23.2.0-release  an-dra-director-1 consul-5
                    HEALTHY  false    -
consul-dra          1         consul-6                 23.2.0-release  an-dra-worker-0   consul-6
                    HEALTHY  false    -
consul-dra          1         consul-7                 23.2.0-release  an-dra-worker-1   consul-7
                    HEALTHY  false    -
admin@orchestrator[an-master]#
```

Step 7 Redeploy the VM.