



Pre-Installation Requirements

- [Installation Overview, on page 1](#)
- [Sample vDRA System, on page 2](#)
- [Installation Order, on page 2](#)
- [Requirements, on page 2](#)
- [Environment Artifacts, on page 4](#)

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 6.7/7.0 must be installed on all the blades that are used to host the vDRA system. For more details see [link](#).

Installing vDRA on vSphere includes the following:

- Create a vDRA installer VM in vSphere using the latest 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.



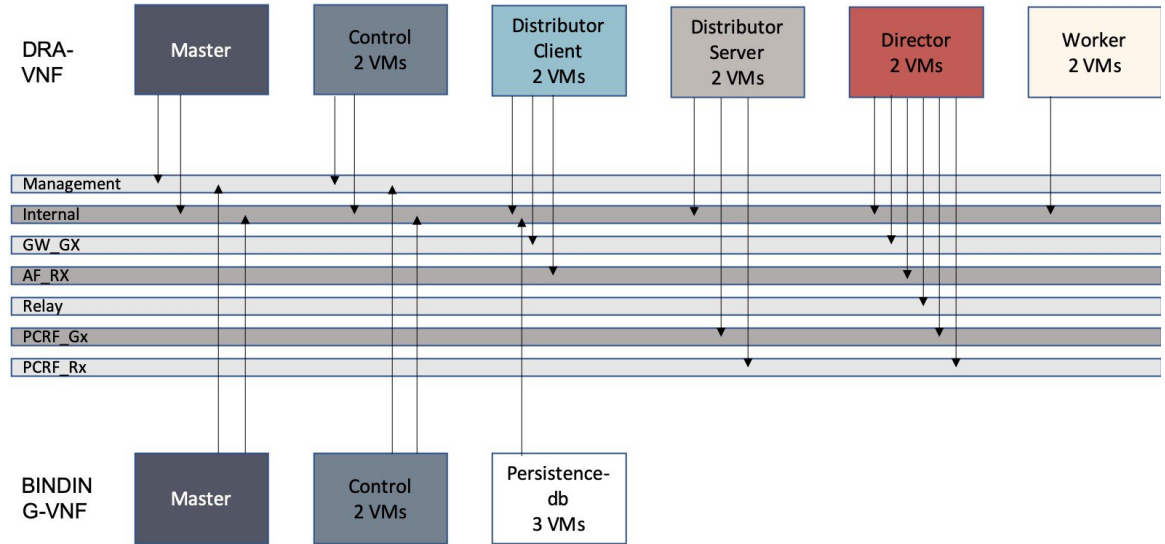
Note The ESXi servers must be configured to use the Network Time Protocol (NTP) to synchronize their clocks.

In vSphere 6.7/7.0 and later, the vSphere Web Client is installed as part of the vCenter Server on Windows or the vCenter Server Appliance deployment.

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 |
| 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 6.7/7.0

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 |

| NETWORK_ | Linux Interface Name |
|----------|----------------------|
| 7 | ens257 |
| 8 | ens162 |
| 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

