HPE OS Deployment Automation (OSDA) Tool

Contents

[Release Notes 1](#_Toc47393329)

[System Requirements 1](#_Toc47393330)

[Installation 1](#_Toc47393331)

[Pre-requisites 1](#_Toc47393332)

[Setup 1](#_Toc47393333)

[Configuration 1](#_Toc47393334)

[Enabling OSDA Service 1](#_Toc47393335)

[Install OSDA Web Client 1](#_Toc47393336)

[Adding Synergy Composer connectivity settings to OSDA UI 1](#_Toc47393337)

[Creating OS Package 1](#_Toc47393338)

[Creating Custom Kickstart 1](#_Toc47393339)

[Deploying OS on Synergy server 1](#_Toc47393340)

[Deploying OS using iLO access to DL and Apollo servers 1](#_Toc47393341)

[OSDA RESTful API Documentation 1](#_Toc47393342)

[Deployment for Synergy servers 1](#_Toc47393343)

[Deployment for servers using HPE iLO interface 1](#_Toc47393344)

[JSON Parameters Mapping to kickstart variables 1](#_Toc47393345)

[SLES Support 1](#_Toc47393346)

[Prerequisites 1](#_Toc47393347)

[Known caveats 1](#_Toc47393348)

## Release Notes

**HPE OSDA version beta 3 (August 3, 2020):**

1. SLES 15 SP1 is now supported operating system for running OSDA.
2. Un-deploy server feature with REST API for powering off the servers as a means to un-deploy the server.
3. Teaming configuration for Ethernet interfaces supported for SLES. This feature can be added to other supported operating systems by creating kickstart files with NIC bonding handling.
4. Ethernet interfaces configuration with VLAN is supported.
5. Supports configuring upto two Ethernet networks.
6. Fixes for deployment task details to show overall status accurately.
7. Validation checks added for input deployment JSON before proceeding with the deployment
8. Several bug fixes and improvements to both server and web client.

**HPE OSDA version beta 2 (July 8, 2020):**

1. Added support for deploying SLES 15 SP1.
2. Major changes made to deploy JSON file which is input data for OS deployment.
3. Deploy JSON allows following additional OS settings:
   1. HTTP, HTTPS proxy and no proxy settings.
   2. Configure up to 2 NICs for Ethernet network (for Ethernet NIC bonding).
   3. Add up to 2 Ethernet network connection settings per host.
   4. User specified SSH Keys to add the deployed host.
4. Choice of kickstart files available for supported operating system types to suite different requirements
5. Allows users to add custom kickstart files to the tool.
6. Error handling improvements.
7. GUI now shows latest task on top of the list under ‘Activity’ menu. REST API for tasks returns tasks ordered by time.
8. User can delete OS packages from GUI or using REST API
9. User can delete OneView connection from GUI or REST API
10. Accepts DNS servers as comma separated entries.

**HPE OSDA version beta 1 (May 18, 2020):**

1. Added support for ESXi 7
2. Support for Gen9 servers
3. Example scripts included
4. Can create logical RAID drive before installing OS
5. Modifies boot-order to place the OS drive on top of the boot order
6. Can deploy OS on Synergy blades on existing server profiles.

**HPE OSDA version alpha\_0.10:** This version includes the following updates:

1. Error handling changes
2. Synergy deployment JSON now takes server profile connection name for the NIC instead of port name.
3. iLO deployment JSON now allows NIC specified using Ethernet adapter and port numbers as enumerated by iLO.

**HPE OSDA version alpha\_0.9:** This version includes the following updates:

1. Modifies boot-order for DL servers to place OS drive on the top.
2. Tracks the deployment progress till the server is booted with installed OS with 30 min timeout.
3. Deployment progress display fixes.

**HPE OSDA version alpha\_0.6**: This version includes the following updates:

1. Introduced Networks section to predefine network settings for the OS
2. Deployment Progress screen introduced to show the deployment progress. This will be launched on clicking “Deploy” button or on clicking the tasks item.
3. Add OS image screen shows the progress of file upload correctly.
4. The tool setup script “setup.sh” changed completely to fix Python installation issue. The tool now installs and uses Python 3.6 version.
5. The deployment task name can be specified in the deploy servers screen.

**HPE OSDA version alpha\_0.2**: This version includes the following features:

1. Supports ESXi 6.7, RHEL/CentOS 7.x operating systems for OS deployment
2. OS deployment modes supported: Synergy Composer using Server Profile Templates (SPTs)
3. Direct iLO based deployment for DL servers.
4. The tool configures the server with user provided IP address and enables SSH for remote access to the server.

**Disclaimer:** This version of tool is for evaluation only as it has many limitations and unhandled error scenarios.

## System Requirements

**Operating System**

CentOS 7.6, RHEL 7.6, SLES 15 SP1

**Storage**

50 GB or more. Storage requirement for OS image ISO files.

**Ports to be allowed by Firewall**

5000 – Incoming traffic on this port for HPEOSDA server side component

80 – Web-UI and HTTP File server

22 – SSH connectivity to the deployed hosts

443 – HTTPS connectivity to the deployed hosts and HPE OneView and iLO

**Software Dependencies**

Python version 3.6 64-bit version

**HPE OneView**

HPE OneView for Synergy version 5.0

**HPE iLO**

Supports iLO 5 and iLO 4 with iLO Advanced License.

## Installation

The software for HPE OSDA is available as installable package. It can be installed on a virtual machine or physical machine running the supported operating system.

After the installation it runs as systemd service with service name **osda**.

### Pre-requisites

Following minimum requirements:

|  |  |
| --- | --- |
| Resource | Minimum requirements |
| CPU | 8 cores (vCPUs) |
| Memory | 8 GB |
| Storage | 50 GB  (Storage requirement for OS image ISO files) |

Storage can be increased based on the requirement for storing ISO images.

### Setup

The installation can be performed using the following steps:

1. Download/Extract the OSDA.tar or git clone the OSDA repository to the local directory in the CentOS/RHEL host. This will be install directory for the tool.
2. Initialize and install the python code by running the following command.

# python3 setup.py install --ip <*host’s IP address*>

python setuppy install

Note: The argument ‘--ip’ should be included with IP address of the OSDA server

This command installs all the required pre-requisites from requirements.txt, creates the directory hierarchy and enables OSDA as a systemd service.

### Configuration

If the setup is successful, the following directories are created.

|  |  |
| --- | --- |
| Directory | Description |
| /opt/hpe/osda | Default path for all OSDA related files |
| /opt/hpe/osda/etc | Configuration files of OSDA |
| /opt/hpe/osda/lib | All python libraries related to OSDA |
| /opt/hpe/osda/bin | Binaries of OSDA |
| /opt/hpe/osda/data | All data files of OSDA – This would include all the kickstarts, osimages, etc., |
| /opt/hpe/osda/log | Logging files for OSDA |
| <Python site-packages>/osda\_server-<release>.egg | OSDA Python module |
| /opt/hpe/osda/data/kickstarts/esxi67 | Path for kickstart templates for VMWare ESXi 6.x |
| /opt/hpe/osda/data/kickstarts/esxi70 | Path for kickstart templates for VMWare ESXi 7.0 |
| /opt/hpe/osda/data/kickstarts/rhel76 | Path for kickstart templates for RHEL 7.x, CentOS 7.x |
| /opt/hpe/osda/data/kickstarts/sles15 | Path for kickstart templates for SUSE Linux Enterprise Server 15 SP1 |

The OSDA configuration is stored in the path /opt/hpe/osda/etc/config.ini.

The following table describes various parameters in this configuration file:

|  |  |  |
| --- | --- | --- |
| Parameter | Default | Description |
| server | **-** | IP address of the OSDA server |
| port | 5000 | Port to run the REST API server on the OSDA server. |
| log\_path | /opt/hpe/osda/log | Path for the OSDA log directory |
| log\_level | INFO | Log level (set to DEBUG for troubleshooting) |

### Enabling OSDA Service

Ensure OSDA server IP address is updated in the configuration file /opt/hpe/osda/etc/config.ini

**This IP address setting is essential for OSDA functionality**

1. Load the OSDA service by reloading the system

# systemctl daemon-reload

1. Start the osda server using systemd service

# systemctl start osda.service

The OSDA server backend will be started on server address and host specified in the config file.

### Install OSDA Web Client

Run the below command to install and configure web server:

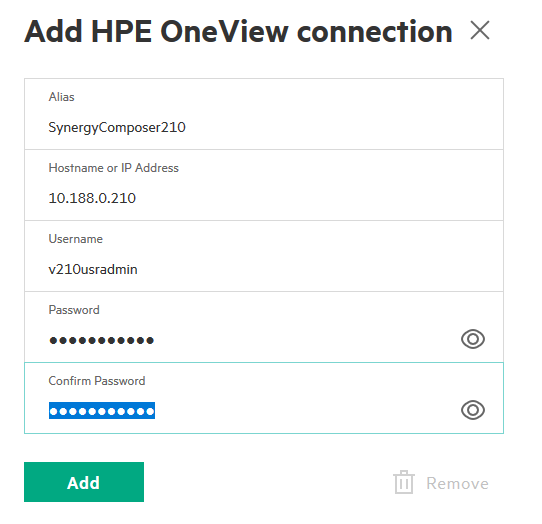
# ./configureWeb.sh

**IMP: The above script can install the Web Client only on Apache HTTPD server running local to machine where OSDA service is running**

Now the tool should be ready. To access the Web-UI from browser, use the url http:/// where “host-IP address in the IP address of the host that is running the HPE OSDA tool.

## Adding Synergy Composer connectivity settings to OSDA UI

Go to sidebar menu item “HPE OneView Appliances” and click on ‘+’ to add new Synergy composer connection. Here is the screenshot for Add Synergy Composer connection:

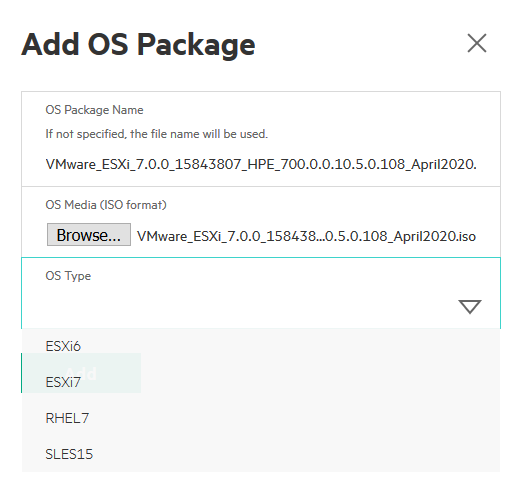


**Note:** HPE OneView appliance managing C7000 and DL servers is not supported by this version of the tool.

## Creating OS Package

OS package contains the OS installation media in ISO-9660 format along with meta-data. The tool performs OS deployment using these precreated OS packages.

To create a new OS package, go to sidebar menu item “OS Packages” and click on ‘+’ to launch add OS package screen. Here is the screen shot:



When uploading the ISO, please wait for the progress to reach **100%** for post upload processing to complete in the server side.

The current version of the tool supports VMWare ESXi 6.5/6.7, RHEL 7.x, CentOS 7.x., SLES 15 SP1.

## Creating Custom Kickstart

OSDA generates the kickstart files for automated OS deployment by using user inputs through JSON and kickstart templates. These templates have variable fields which will be substituted by the user inputted values at the time of the deployment.

The tool is packaged with several kickstart templates for all the supported operating systems.

The location for the template files is:

/opt/hpe/osda/data/kickstarts/

Additional custom kickstart templates can be added to an existing installation by manually copying the custom kickstart templates to respective directory based on the OS type. The following table shows the directories for various OS types.

|  |  |  |
| --- | --- | --- |
| **OS Type** | **Valid Operating Systems** | **Kickstart Template Directory** |
| ESXI6 | VMWare ESXi 6.5 and 6.7 | /opt/hpe/osda/data/kickstarts/esxi67 |
| ESXI7 | VMWare ESXi 7.0 | /opt/hpe/osda/data/kickstarts/esxi70 |
| RHEL7 | RHEL 7.x or later, CentOS 7.x | /opt/hpe/osda/data/kickstarts/rhel76 |
| SLES15 | SLES 15 SP1 | /opt/hpe/osda/data/kickstarts/sles15 |

## Deploying OS on Synergy server

This tool allows the user to deploy OS on Synergy servers using Server Profile Templates (SPTs). The tool uses the user input for the server profile template to request Synergy composer for a server that matches the server-hardware-type associated with the SPT. This eliminates the need for user to choose particular server hardware explicitly.

The SPTs should have the following defined:

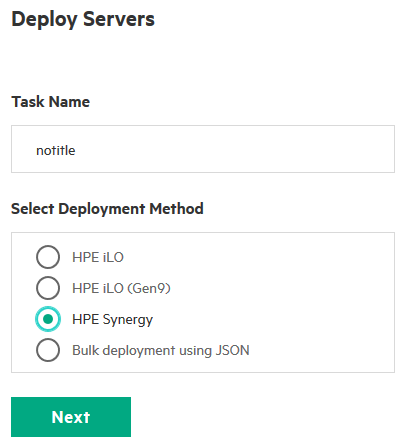
* Network connections
* Logical drive for OS filesystem, with RAID1 using the server’s local drives.

The tool considers the servers that satisfy the following rules:

* The server-hardware-type matches with the SPT
* No server profile exists
* Server is powered-off
* Server health is green in the OneView.

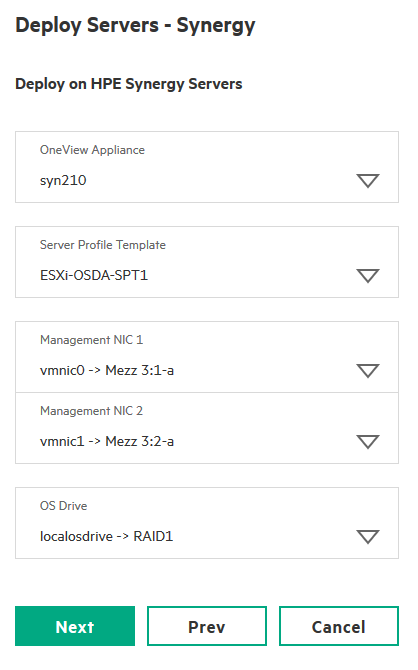
**IMP:** The root password for installed OS is hardcoded to “**Welcome#123**”.

To deploy the servers go to sidebar menu item “Deploy”. This will launch the deployment wizard. The first screen allows you to choose deployment method. Choose “HPE Synergy” as in the following screenshot:



**Next:** In the next screen, you need to select the Synergy Composer that was added to OneView connections. This will automatically load the Server Profile Templates found in the specified Synergy Composer.

On selecting the desired SPT, the form automatically loads the network connections and storage drives (logical drives only) defined in the SPT. You need to choose the correct network connection for assigning IP address by this tool.

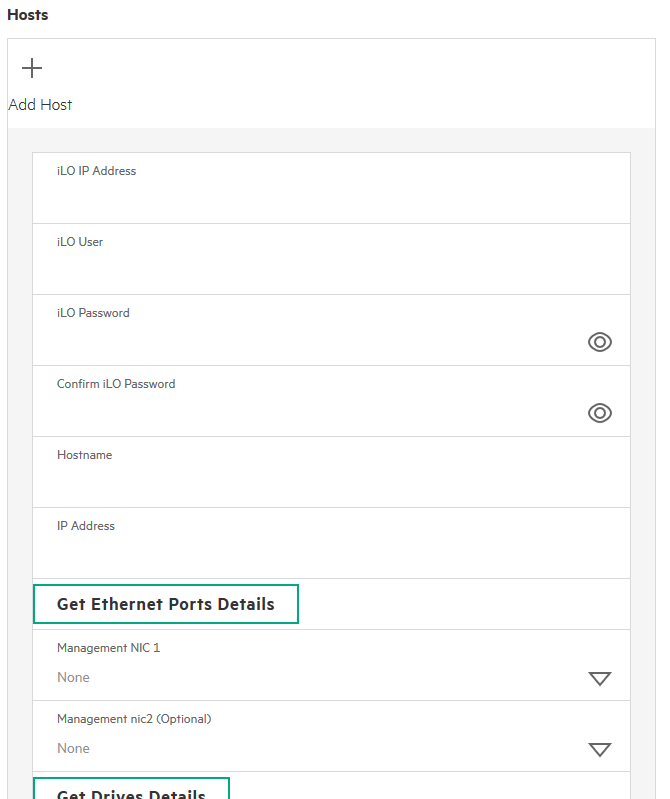


## Deploying OS using iLO access to DL and Apollo servers

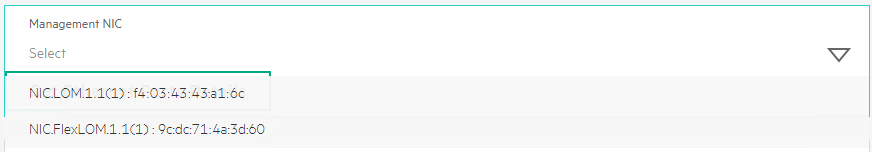
This tool deploys the operating system on DL and Apollo servers by direct iLO access through Redfish API.

**IMP:** The root password for installed OS is hardcoded to “**Welcome#123**”.

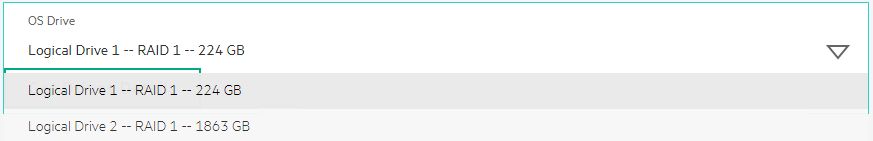
For inputting the network connection and target OS drive, the tool lets the user query iLO using the buttons in the deploy servers form. The below screenshot shows the host entry dialog with buttons to query network connections and storage drives:



The following is an example screenshot showing the network ports listed by the tool after the query. Please note that the tool lists only the connected network ports of the server.



The following is an example screenshot showing the storage drives listed by the tool after the query. Please note that the tool lists only the logical drives.



## OSDA RESTful API Documentation

### Deployment for Synergy servers

url = 'http://<OSDA host IP address>:5000/rest/deploy'

The JSON body for deploy server REST call should be like the following example:

{

"taskName": "Task1",

"hosts": [

{

"serverProfile": "sp1",

"hostName": "host1",

"osPackage": "VMware-ESXi-6.7.0-Update3-15160138-HPE-Gen9plus-670.U3.10.5.0.48-Dec2019.iso",

"kickstartFile": "ks.cfg",

"networks": [

{

"ipAddr": <IP address>,

"netmask": "255.255.255.0",

"gateway": "10.188.210.1",

"dns": <DNS1,DNS2>,

"bootProto": "static",

"nic1": {

"connectionName": "vmnic0"

},

"nic2": {

"connectionName": "vmnic1"

},

"bondingType": "",

"vlans": "210"

},

{}

],

"osDrive": {

"driveName": "localosdrive"

}

}

],

"osPackage": "VMware-ESXi-6.7.0-Update3-15160138-HPE-Gen9plus-670.U3.10.5.0.48-Dec2019.iso",

"deploymentMode": "hpesynergy",

"createServerProfile": true,

"oneviewDetails": {

"ovName": "syn210",

"ovSPT": "ESXi-OSDA-SPT1"

}

}

**Description of various JSON parameters:**

|  |  |  |
| --- | --- | --- |
| **Parameter name** | **Description** | **Value Type** |
| taskName | This is name assigned to the task. | String |
| deploymentMode | Mode of the deployment | String  **“hpesynergy”** – for deployment on Synergy servers. “oneViewDetails” section is mandatory with this option.  **“hpeilo”** – For deployment through HPE iLO. The hosts section should include iLO IP address and credentials for each server.  **“hpeilo\_gen9”** – For deployment through HPE iLO 4 (version 2.30 and above). Supports DL Gen9 servers only. |
| oneviewDetails | This section contains name the OneView connection details. Applicable for HPE Synergy only. | JSON |
| oneviewDetails[‘ovName’] | Alias name of the OneView connection entry in OSDA settings, for the target OneView appliance. | String |
| oneviewDetails[‘ovSPT’] | Name of the Server Profile Template to be used for server delployment. For a given deployment task, all the hosts are expected to be deployed using same SPT. | String |
| createServerProfile | Specify whether the server profile should be created by OSDA or created prior to the OS deployment task. | Boolean  **True** – OSDA creates the server profile using the name specified in the hosts section. If the server profile already exists in OneView then the deployment fails. This is the DEFAULT value if this option is not specified.  **False** – OSDA deploys on the host using the user specified server profile name in the hosts details. If the server profile is not found or in powered state the deployment will fail. |
| hosts | This section is array of hosts details. Each host item specifies the details of target server hardware, OS settings etc.  The limit of host entries is not determined yet but should support up to 10 hosts at a time for concurrent deployment. | JSON |
| hosts[][‘serverProfile’] | Name of the server profile associated with the target server. This value should be unique to the OneView appliance. | String |
| hosts[][‘hostName’] | Host name (FQDN) | String |
| host[][‘networks’][][‘bootProto’] | Specify IP assignment for the installed host. | String  ‘static’ – static assignment of IP address. This option requires the host IP address along with the other network settings.  ‘dhcp’ – for dhcp assignment of IP address. This option expects DHCP service running in the network. |
| host[][‘networks’][][‘ipAddr’] | Specify host’s IP address | String in IP address format |
| host[][‘networks’][][‘netmask’] | Subnet mask | String in IP address format |
| host[][‘networks’][][‘gateway’] | Network gateway IP address | String in IP address format |
| host[][‘networks’][][‘dns’] | DNS servers (comma separated list of DNS servers up to three items) | String in IP address format |
| host[][‘networks’][][‘nic1’] | Details of the first Ethernet NIC. | JSON |
| host[][‘networks’][][‘nic1’][‘connectionName’] | Name of the network connection as in the server profile template definition in the OneView. In case of redundant connections created for the network, specify only one connection name. | String |
| host[][‘networks’][][‘nic2’] | Details of the second Ethernet NIC. | JSON |
| host[][‘networks’][][‘nic2’][‘connectionName’] | Name of the network connection as in the server profile template definition in the OneView. In case of redundant connections created for the network, specify only one connection name. | String |
| host[][‘networks’][][‘bondingType’] | Network bonding type | String |
| host[][‘networks’][][‘vlans’] | Network VLAND ID | String |
| host[][‘osDrive][‘ driveName’] | Name of the logical drive for the OS installation destination. Name should be same as in the server profile template definition. RAID level 1 is recommended. | String |
| host[‘osPackage’] | Name of the OS package that should be used for OS installation. This package should be pre-created in the OSDA. | String |

### Deployment for servers using HPE iLO interface

url = 'http://<OSDA host IP address>:5000/rest/deploy'

The Ethernet NICs are identified using the Network Adapter number and the port numbers with-in the adapter.

The following screenshot shows the physical network adapters enumerated by iLO. The network port that corresponds to the desired NIC for the management network should be specified using the adapter number and port number shown by iLO (also the order returned by Redfish API “/redfish/v1/Systems/1/BaseNetworkAdapters”).

Please note that the servers need to be similarly configured in order to participate in the bulk deployment using the OSDA tool.

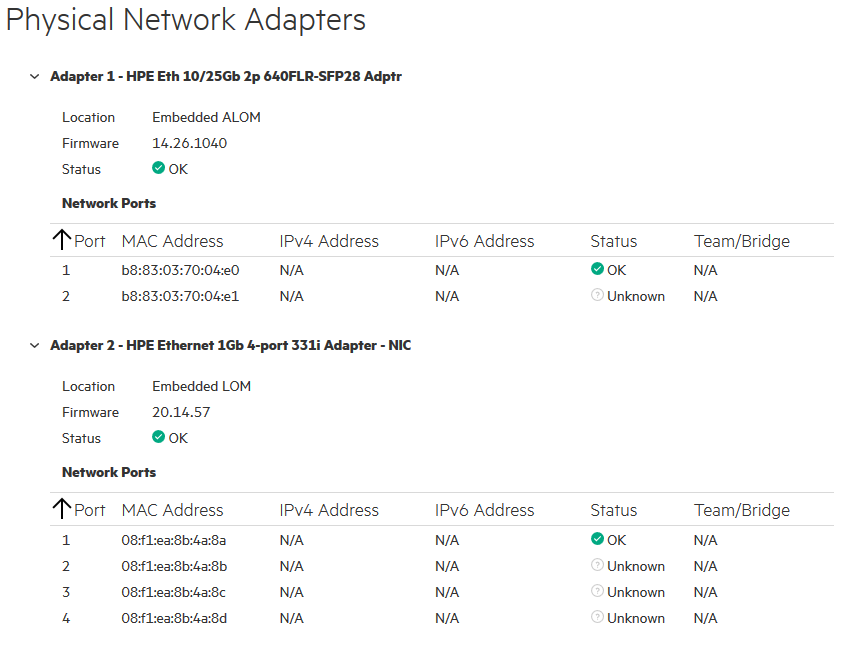


Figure 1: Physical Network Adapters enumerated by the iLO

The JSON body for deploy server REST call should be like the following example:

{

"taskName": "notitle",

"hosts": [

{

"iloIPAddr": "10.188.175.116",

"iloUser": "v0175usradmin",

"iloPassword": "HP!nvent123",

"hostName": "host90",

"osPackage": "HPE-SLES15.iso",

"osLicenseKey": "",

"sshKey": "",

"kickstartFile": "",

"httpProxy": "",

"httpsProxy": "",

"noProxy": "",

"networks": [

{

"ipAddr": "10.188.175.90",

"netmask": "255.255.255.0",

"gateway": "10.188.175.1",

"dns": "10.188.0.2",

"bootProto": "static",

"NIC1": {

"adapterId": "1",

"portId": 1

}

}

],

"osDrive": {

"logicalDrive": {

"driveTechnology": "SATA SSD",

"capacity": "200",

"capacityUnit": "GB",

"raidLevel": "Raid1",

"operation": "DELETE\_ALL\_AND\_CREATE"

}

}

}

],

"deploymentMode": "hpeilo",

"oneviewDetails": {}

}

**Description of various JSON parameters:**

|  |  |  |
| --- | --- | --- |
| **Parameter name** | **Description** | **Value Type** |
| taskName | This is name assigned to the task. | String |
| deploymentMode | Mode of the deployment | String  **“hpesynergy”** – for deployment on Synergy servers. “oneViewDetails” section is mandatory with this option.  **“hpeilo”** – For deployment through HPE iLO. The hosts section should include iLO IP address and credentials for each server.  **“hpeilo\_gen9”** – For deployment through HPE iLO 4 (version 2.30 and above). Supports DL Gen9 servers only. |
| Hosts | This section is array of hosts details. Each host item specifies the details of target server hardware, OS settings etc.  The limit of host entries is not determined yet but should support up to 10 hosts at a time for concurrent deployment. | JSON |
| hosts[][‘iloIPAddr’] | The IP address of the iLO for the target server. | String |
| hosts[][‘iloUser’] | iLO local user name | String |
| hosts[][‘iloPassword’] | iLO local user password | String |
| hosts[][‘hostName’] | Host name (FQDN) | String |
| hosts[][‘sshKey’] | SSH Key | String (SSH Key) |
| hosts[][‘kickstartFile’] | Custom Kickstart File  The custom kickstart file is present in the OSDA data kickstart config files | String |
| hosts[][‘httpProxy’] | HTTP Proxy to be assigned to the server | String |
| hosts[][‘httpsProxy’] | HTTPS Proxy to be assigned to the server | String |
| hosts[][‘noProxy’] | NO Proxy to be assigned to the server | String |
| hosts[][‘networks’][][‘bootProto’] | Specify IP assignment for the installed host. | String  ‘static’ – static assignment of IP address. This option requires the host IP address along with the other network settings.  ‘dhcp’ – for dhcp assignment of IP address. This option expects DHCP service running in the network. |
| hosts[][‘networks’][][‘ipAddr’] | Specify host’s IP address | String in IP address format |
| hosts[][‘networks’][][‘netmask’] | Subnet mask | String in IP address format |
| hosts[][‘networks’][][‘gateway’] | Network gateway IP address | String in IP address format |
| hosts[][‘networks’][][‘dns’] | Primary DNS server | String in IP address format |
| hosts[][‘networks’][][‘nic1’][‘adapterId’] | The network adapter number | Number |
| hosts[][‘networks’][][‘nic1’][‘portId’] | The port number with in the adapter. This port should be connected to the desired network. | Number |
| hosts[][‘networks’][][‘nic1’][‘macAddress] | MAC Address (Optional)  Combination of adapterId/portId or macAddress are used | Hexadecimal MAC address format |
| hosts[][‘networks’][][‘nic2’][‘adapterId’] | Network adapter number of second interface  (nic2 is optional here and can be skipped if only single interface and no bonding is required) | Number |
| hosts[][‘networks’][][‘nic2’][‘portId’] | The port number with in the adapter. This port should be connected to the desired network. | Number |
| hosts[][‘networks’][][‘nic2’][‘macAddress] | MAC Address of second interface (Optional)  Combination of adapterId/portId or macAddress are used | Hexadecimal MAC address format |
| hosts[][‘networks’][][‘bondingType’] | Network bonding type | String |
| hosts[][‘networks’][][‘vlans’] | Network VLAN ID | String |
| host[][‘osDrive][‘logicalDrive’] | Logical drive details for OS installation destination | JSON |
| host[][‘osDrive][‘logicalDrive’][‘driveTechnology’] | Physical drives drive technology. Supported values “SAS SSD, SAS HDD, SATA SSD, SATA HDD”.  Make sure this value matches with the available physical drives in the server. | String |
| host[][‘osDrive][‘logicalDrive’][‘capacity’] | Capacity of the logical drive. | String |
| host[][‘osDrive][‘logicalDrive’][‘capacityUnit’] | GB or TB | String |
| host[][‘osDrive][‘logicalDrive’][‘raidLevel’] | RAID1 is the only value supported | String |
| host[][‘osDrive][‘logicalDrive’][‘operation’] | Operation specifies operation by OSDA for the logical drive. | String  **DELETE\_ALL\_AND\_CREATE** – This value specifies that the logical drive with specified settings should be created after all existing logical drives are deleted from the server.  **USE\_EXISTING** – This value specifies that the OS should be installed on existing logical drive matching specified settings. |
| host[][‘osDrive][‘logicalDrive’][‘logicalDriveNumber’] | The logical drive number assigned by iLO. This option is valid only with “operation”=”USE\_EXISTING” | Number |
| Host[][‘osPackage’] | Name of the OS package that should be used for OS installation. This package should be pre-created in the OSDA. | String |

### JSON Parameters Mapping to kickstart variables

Custom user generated kickstarts are supported from Beta 2.

End users using OSDA can have their custom kick start file and OSDA will automatically selects the custom configuration mentioned in those kickstarts.

Few parameters of custom kickstarts can be dynamically selected using input parameters passed through json.

This section gives a detailed overview of mapping between the input parameters and the kickstart variables.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| JSON Variables | | | | Kickstart  Variables |
| taskName |  |  |  | NA |
| hosts |  |  |  | NA |
|  | iloIPAddr |  |  | NA |
|  | iloUser |  |  | NA |
|  | iloPassword |  |  | NA |
|  | hostName |  |  | **%HOSTNAME%** |
|  | osPackage |  |  | NA |
|  | osLicenseKey |  |  | NA |
|  | sshKey |  |  | **%SSH\_KEY%** |
|  | kickstartFile |  |  | NA |
|  | httpProxy |  |  | **%HTTP\_PROXY%** |
|  | httpsProxy |  |  | **%HTTPS\_PROXY%** |
|  | noProxy |  |  | **%NO\_PROXY%** |
|  | networks |  |  |  |
|  |  | ipAddr |  | **%IPADDR1%** |
|  |  | netmask |  | **%NETMASK1%** |
|  |  | gateway |  | **%GATEWAY1%** |
|  |  | dns |  | **%DNS11%** |
|  |  | bootProto |  | NA |
|  |  | nic1 |  | **%MAC11%** |
|  |  |  | adapterId |
|  |  |  | portId |
|  |  |  | macAddress |
|  |  | nic2 |  | **%MAC12%** |
|  |  |  | adapterId |
|  |  |  | portId |
|  |  |  | macAddress |
|  |  | vlans |  | **%VLANS1%** |
|  |  | ipAddr |  | **%IPADDR2%** |
|  |  | netmask |  | **%NETMASK2%** |
|  |  | gateway |  | **%GATEWAY2%** |
|  |  | dns |  | **%DNS21%** |
|  |  | bootProto |  | NA |
|  |  | nic1 |  | **%MAC21%** |
|  |  |  | adapterId |
|  |  |  | portId |
|  |  |  | macAddress |
|  |  | nic2 |  | **%MAC22%** |
|  |  |  | adapterId |
|  |  |  | portId |
|  |  |  | macAddress |
|  |  | vlans |  | **%VLANS2%** |
|  | osDrive |  |  |  |
|  |  | logicalDrive |  | NA |
|  |  | driveTechnology |  | NA |
|  |  | capacity |  | NA |
|  |  | capacityUnit |  | NA |
|  |  | raidLevel |  | NA |
|  |  | operation |  | NA |
| deploymentMode |  |  |  | NA |
| oneviewDetails |  |  |  | NA |
|  | ovName |  |  | NA |
|  | ovSPT |  |  | NA |

## SLES Support

Starting Beta 2.0, SLES 15 SP1 can be deployed using OSDA. AutoYast control file (equivalent to kickstart) is used for automation of SLES installation.

### Prerequisites

* AutoYast2 is used for SUSE installation
* SLES ISO should contain the packages of autoyast2, autoyast2-installation, yast2

### Known caveats

* Custom autoyast files should be used for advanced operations like VLAN, bonding, etc.,