

Lenovo Networking Plug-in for VMware vRealize Orchestrator

Deployment and User Guide

Version 1.2

LenovoTM

Note: Before using this information and the product it supports, read the general information in the *Safety information and Environmental Notices and User Guide* documents on the *Lenovo Documentation CD* and the *Warranty Information* document that comes with the product.

First Edition (August 2016)

© Copyright Lenovo 2016

LIMITED AND RESTRICTED RIGHTS NOTICE: If data or software is delivered pursuant a General Services Administration "GSA" contract, use, reproduction or disclosure is subject to restrictions set forth in Contract No. GS-35F-05925.

Lenovo and the Lenovo logo are trademarks of Lenovo in the United States, other countries or both.

Contents

Chapter 1.. Overview	.5
Requirements	6
VMware vRealize Orchestrator	6
Memory and CPU Utilization	6
Supported Lenovo Networking Products	6
Licensing	7
Chapter 2.. Plug-in Installation	.9
Installation and Update Procedures	9
Prerequisites	9
Plug-in Installation	9
Package Installation	.11
Plug-in Uninstallation	.15
Chapter 3.. Using the Plug-in	17
Actions and Workflows	.18
Actions	.18
Advanced Workflows	.24
Chapter 4.. Troubleshooting	25
Chapter 5.. Known Issues	29
Appendix A. Getting help and technical assistance	31
Appendix B. Notices	33
Trademarks	.34

Chapter 1. Overview

The *Lenovo Networking Plug-in* leverages the open plug-in architecture of vRealize Orchestrator product to support the management of Lenovo Networking products. Through the use of Actions and Workflows, it allows management of key features in Lenovo switches, such as:

- VLANs
- Static & LACP portchannels
- Ports
- Connectivity to server adapters
- vLAGs
- UFP
- Server/uplink ports
- Firmware updates
- Switch reload

Requirements

VMware vRealize Orchestrator

This version of the plug-in is supported on the following VMware vRealize Orchestrator releases:

- VMware vRealize Orchestrator 6.0
- VMware vRealize Orchestrator 7.0

Memory and CPU Utilization

The memory and CPU utilization of this plug-in have been characterized and has been determined that none of the workflows or actions cause significant usage of these resources.

Supported Lenovo Networking Products

The following Lenovo Networking products are supported by the current version of the *Lenovo Networking Plug-in*:

- Lenovo Flex System Interconnect Fabric
- Lenovo Flex System Fabric EN4093R 10Gb Scalable Switch
- Lenovo Flex System CN4093 10Gb Converged Scalable Switch
- Lenovo Flex System SI4093 System Interconnect Module
- Lenovo Flex System SI4091 10Gb System Interconnect Module
- Lenovo RackSwitch G7028
- Lenovo RackSwitch G7052
- Lenovo RackSwitch G8052
- Lenovo RackSwitch G8124-E
- Lenovo RackSwitch G8264
- Lenovo RackSwitch G8264CS
- Lenovo RackSwitch G8272
- Lenovo RackSwitch G8296
- Lenovo RackSwitch G8332

Licensing

The *Lenovo Networking Plug-in* for VMware vRealize Orchestrator comes in two forms:

- Non-warranted version that is free to anyone and downloadable from the [VMware Solution Exchange website](#)
- Warranted version that is purchased under the vRealize Subscription and Support Package and is downloadable by the customer from [IBM Passport Advantage](#) system

Although the functionality of the two plug-ins is identical, each will contain a different End User License Agreement (EULA). The EULA is presented to the user upon import, where they must accept the terms.

- Non-warranted will display Lenovo's ILAN license. First line of license: *International License Agreement for Non-Warranted Programs*
- Warranted will display Lenovo's IPLA license. First line of license: *International Program License Agreement*

Chapter 2. Plug-in Installation

Installation and Update Procedures

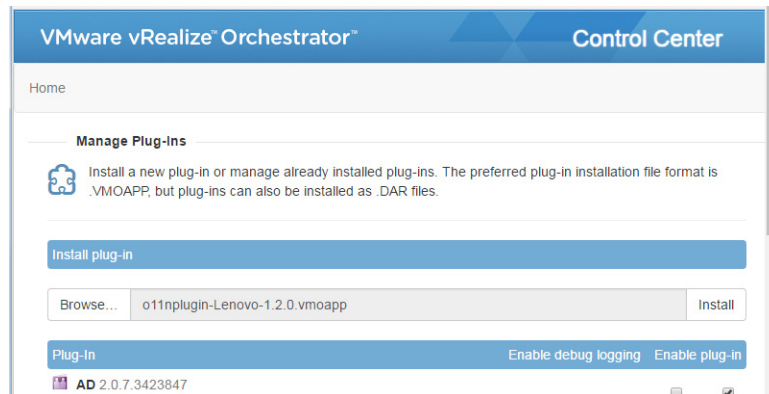
To install or update the plug-in follow the steps described below:

i. Prerequisites

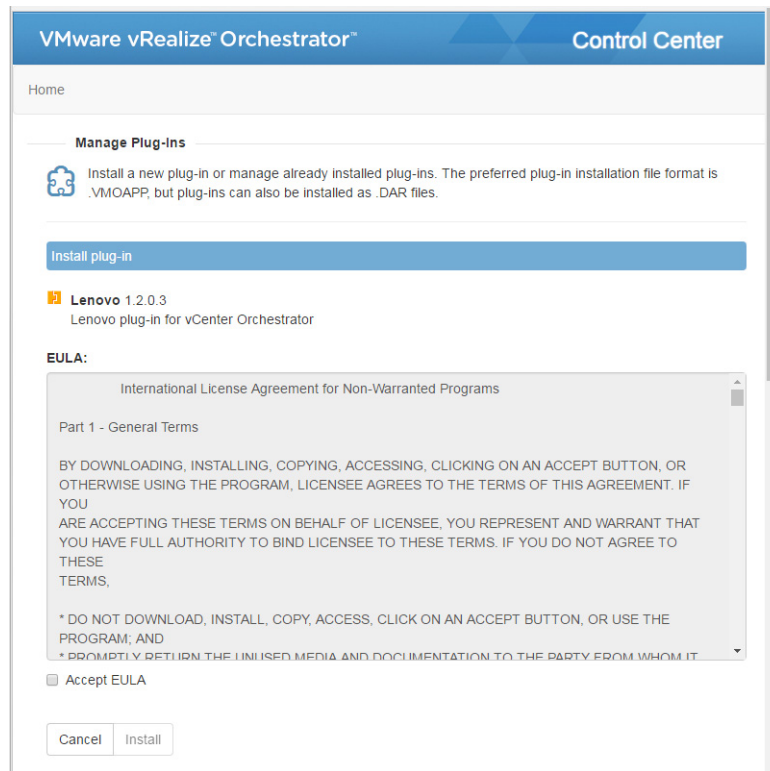
1. Install VMware vRealize Orchestrator 6.0 or 7.0.
2. Download the Lenovo Networking Plug-in for VMware vRealize Orchestrator from any of the following:
 - o [VMware Solution Exchange \(VSX\) website](#)
 - o [IBM Passport Advantage](#), if purchased via the “Lenovo Networking Bundle for vRealize” product

ii. Plug-in Installation

1. Logon to the vRealize Orchestrator web Configuration page.
2. On the left pane, click **Plug-ins**.
3. In the Install new plug-in area, click on the **browse icon**.
4. Navigate to the folder where you have saved the .vmoapp file and select the .vmoapp file.
5. Click **Open**.
6. Click **Upload and Install**.



7. Agree to the license terms. Depending whether you are installing the free, non-warranted plug-in or the for-fee, warranted plug-in the license that is displayed will be different.

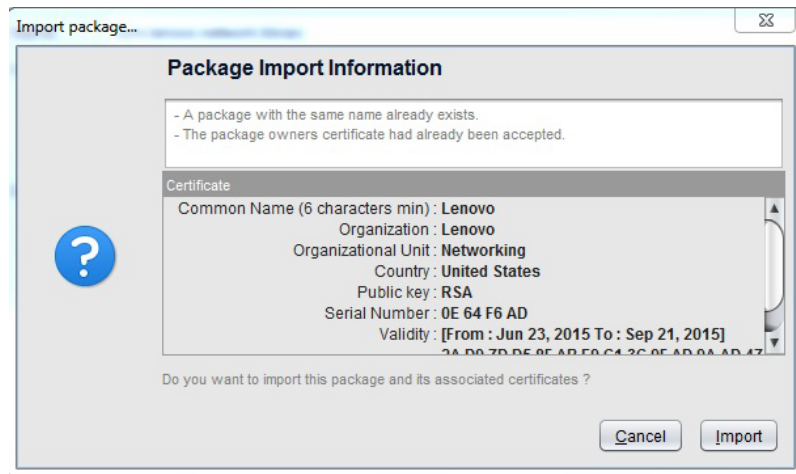


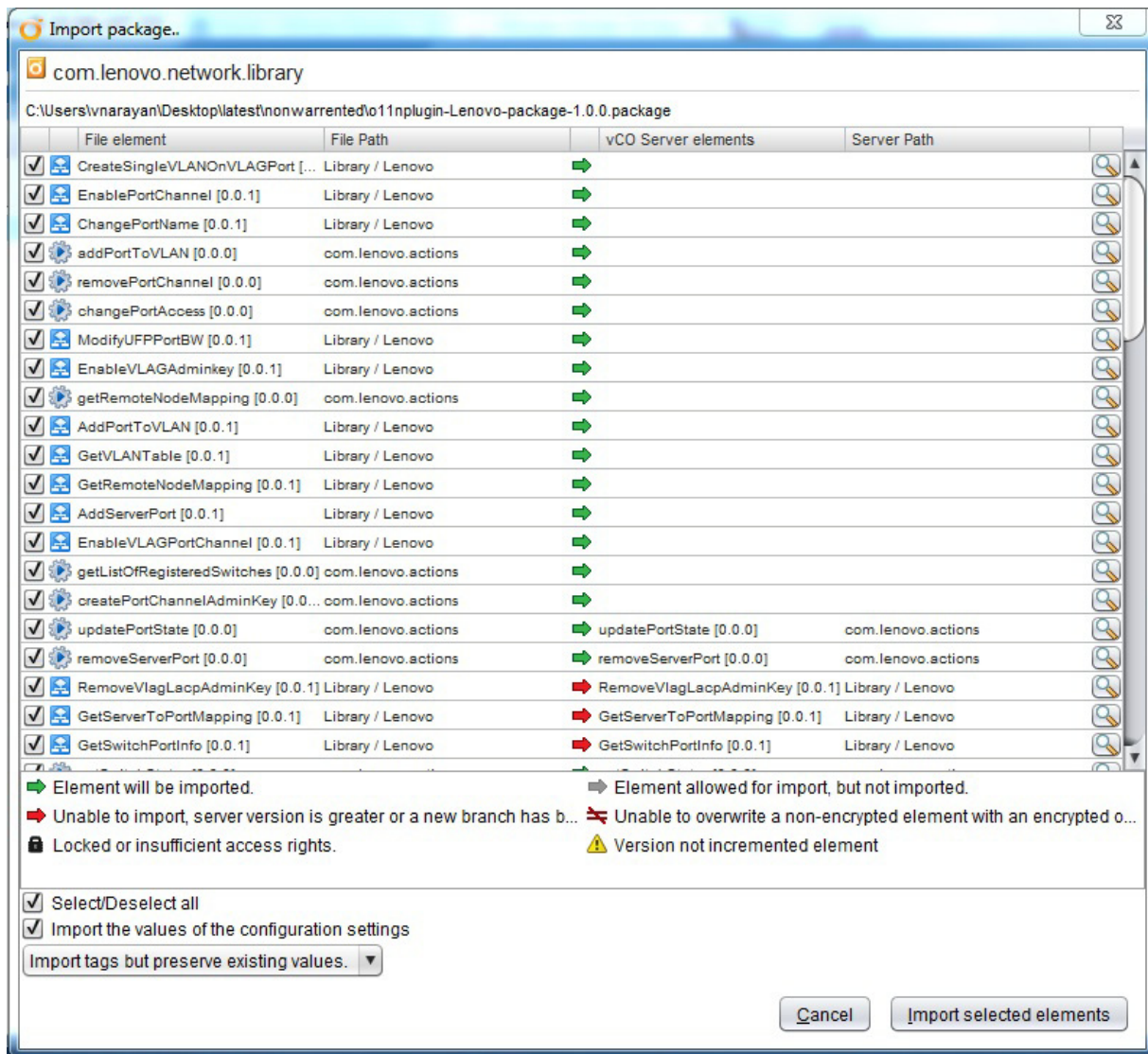
The plug-in is installed.

iii. Package Installation

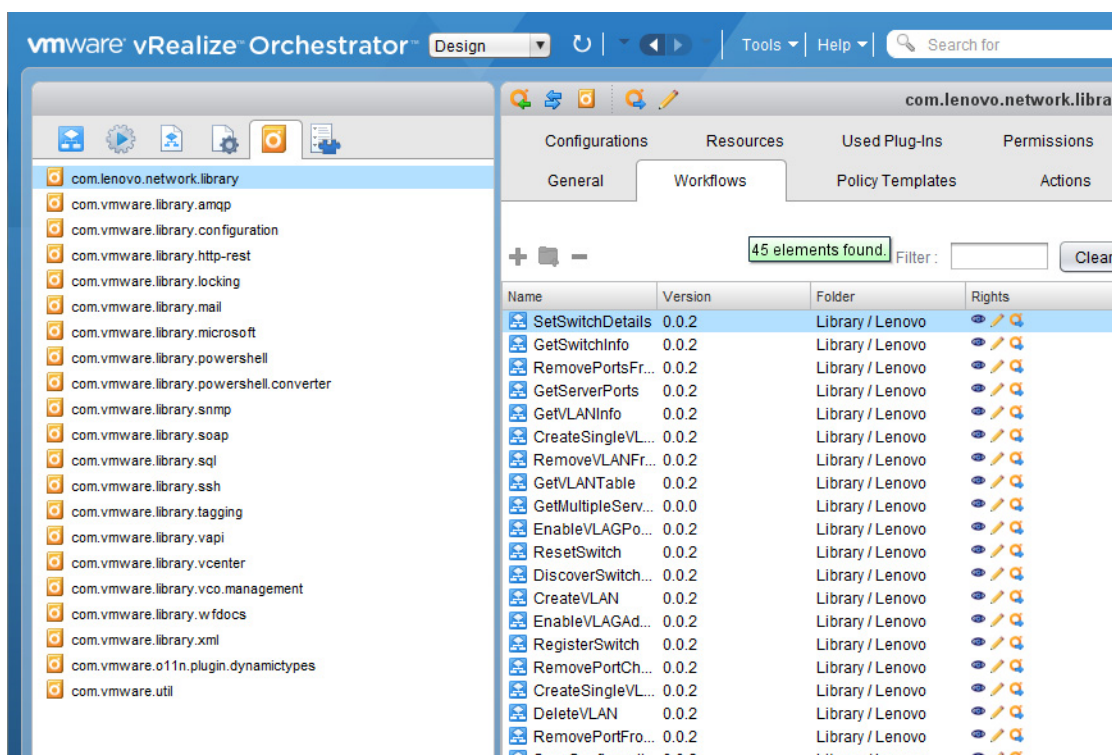
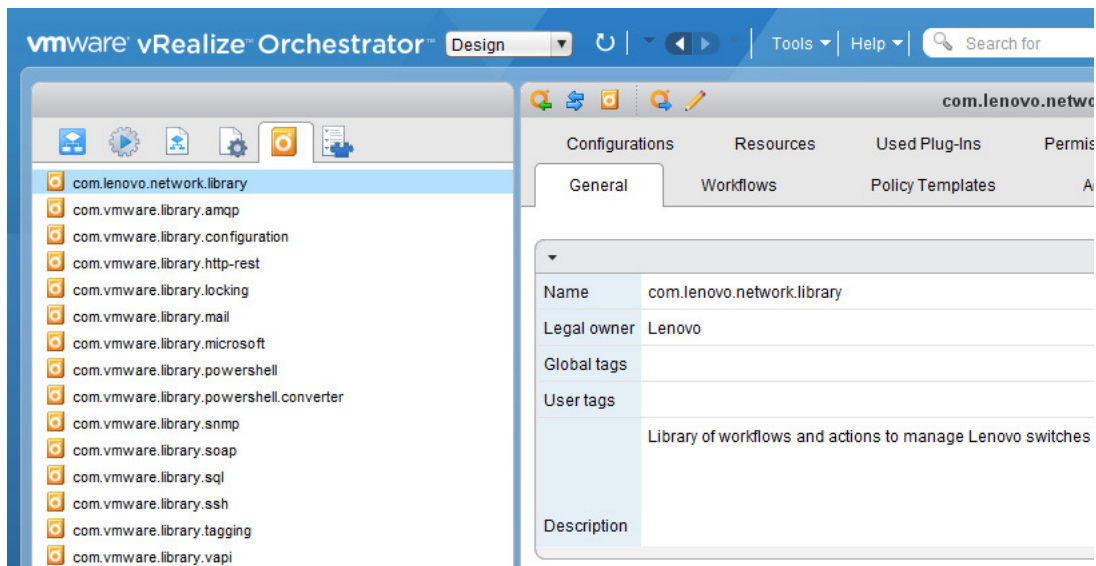
The procedure below has to be followed only the first time when the vRealize Orchestrator (vRO) plug-in is installed on the vRO Virtual Machine (VM) so that the certificate is registered. Subsequently, the package will be installed automatically when updating the plug-in.

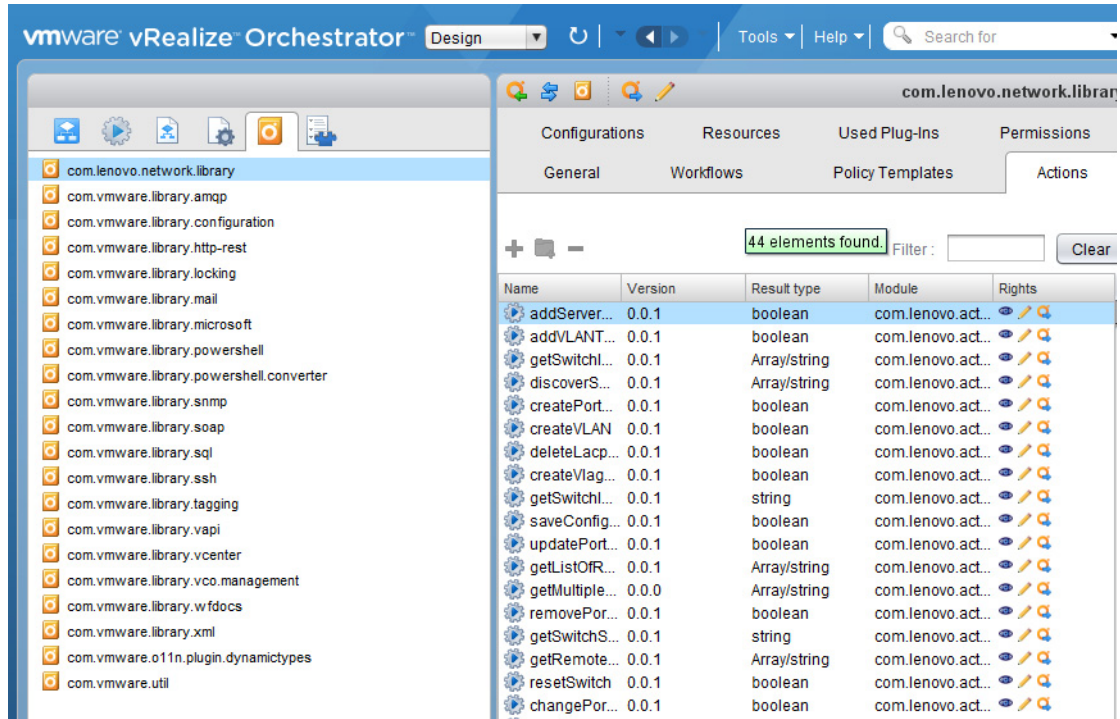
1. Log into the vRO Client.
2. Go to the **Design** mode.
3. Go to **Packages** tab.
4. Right-click on the open space present on the left side of the tab. A menu is then displayed.
5. Click on the **import package** option. Choose the **.package** file included with the plug-in and click **Open** button and then import all the actions and workflows into vRO.



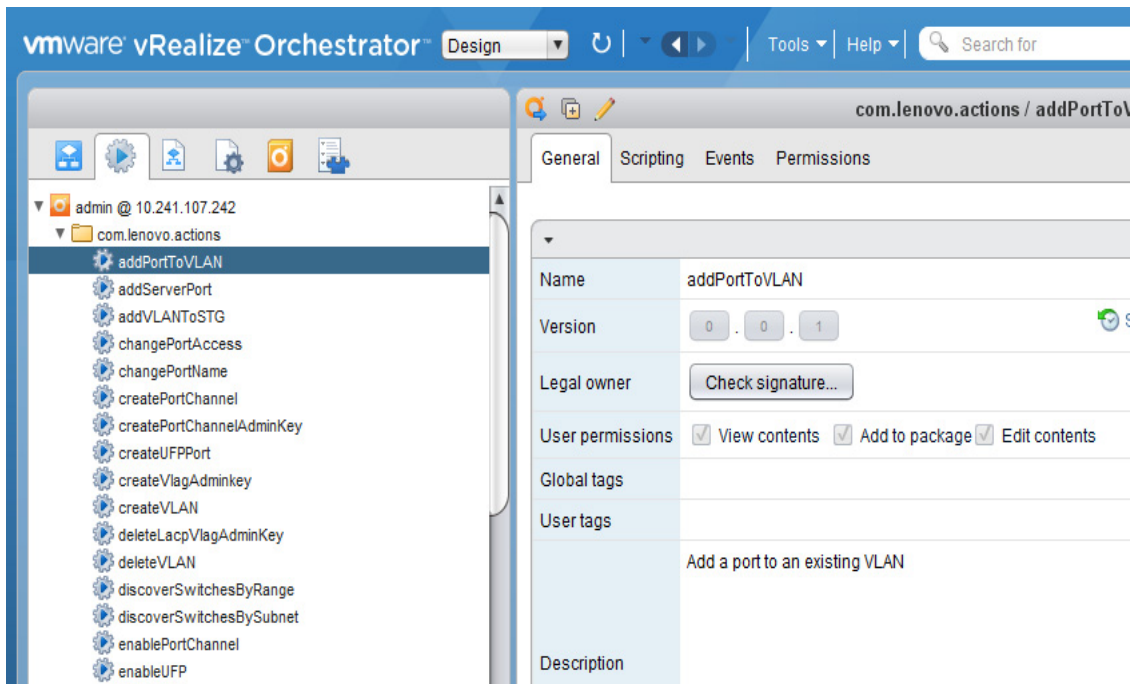


- After this step, the package that is imported will be shown in the packages tab. Click on the package **com.lenovo.network.library** and see the actions and workflows listed on the right side. Verify if the package contents are proper after the import is over:

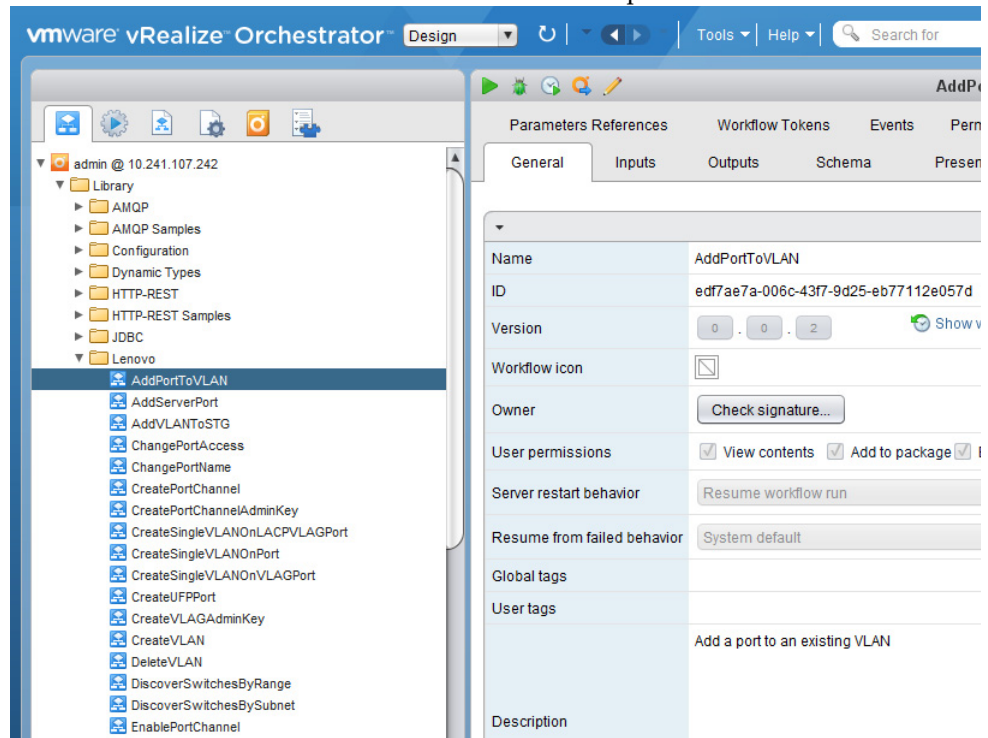




- a. Click on the **Actions** tab in the left panel and expand the **com.lenovo.actions** folder to view the imported actions.



- b. Click on the **Workflows** tab in the left panel and expand the **com.lenovo.actions** folder to view the imported workflows.



iv. Plug-in Uninstallation

To uninstall the plug-in follow the steps described below:

1. Go to **/usr/lib/vro/app-server/plugins** directory.
2. Remove the **o11nplugin-Lenovo.dar** file.
3. Restart the servers.

Note: Instructions for general plug-in uninstallation provided by the VMware Knowledge Base can be found here:

[Uninstalling a plug-in from VMware vRealize Orchestrator 5.5 and later](#)

Chapter 3. Using the Plug-in

To start using the plug-in, run and log into the vRealize Orchestrator (vRO) Client. The client allows you to run and schedule workflows, manage user permissions etc. The client also enables you to develop workflows and actions.

For more information about using the vRO Client, refer to the following document on the [vRealize Orchestrator Documentation](#) page:

- Using the VMware vRealize Orchestrator Client.

The client has three views:

- **Run** - Provides features that enable you to run and schedule workflows.
- **Design** - Provides features that enable you to develop actions and workflows.
- **Administer** - Provides features that enable you to manage users, packages etc.

Actions and Workflows

This section describes the Actions and Workflows provided by the plug-in. Actions typically are individual tasks that have a single result and can be used to build Workflows. Workflows typically provide a task or process that may involve many actions, decisions and results.

Actions

The following table lists all the Actions implemented by the plug-in. In addition, a corresponding Workflow for each Action is provided. An Action begin with a lower case letter, whereas its corresponding Workflow begins with an upper case letter.

Table 1. *Actions*

Action name	Workflow name	Description	Input	Output	Preconditions
registerSwitch	RegisterSwitch	Add switches to the VRO management domain using its IPv4 address and credentials.	String switchIPAddress, String version, String readCommunity, String writeCommunity, String userName, String authenticationProtocol, String authenticationPassword, String privacyProtocol, String privacyPassword	Boolean	SNMP enabled on switch.
getListOfRegisteredSwitches	GetListOfRegisteredSwitches	Gets the list of registered switches within the vRO management domain.	None	String[] - array of switch IP addresses	None
getServerToPortMapping	GetServerToPortMapping	Determine which switch and port has specific server connection.	String serverMacAddress	String[] - Switch IP and Switch Port	Switch is registered. Switch has valid connection to an adapter in a server. LLDP is enabled on the switch.
getRemoteNodeMapping	GetRemoteNodeMapping	Return the remote system information for a given port.	String switchIPAddress, String switchPort	String[] - Remote system's chassis ID and port name	Switch is registered. Switch has valid connection to an adapter in a server. LLDP is enabled on the switch.

Table 1. Actions

Action name	Workflow name	Description	Input	Output	Preconditions
getMultipleServertoPortMapping	GetMultipleServertoPortMapping	Determine the switch and port to which each of the of the server MAC addresses that are input is connected.	String serverMACAddress, [serverMACAddress], ...	String[] – serverMACAd dress, Switch IP address, Switch Port, [serverMACA ddress, Switch IP address, Switch Port], ...	Switch is registered. Switch has valid connection to an adapter in a server. LLDP is enabled on the switch.
getVLANTable	GetVLANTable	List of VLANs Configured on a switch.	String switchIPAddress	String[] - Array of VLANs and their information	Switch is registered.
getVLANInfo	GetVLANInfo	Get detailed information about a specific VLAN.	String switchIPAddress, String vlanNumber	String[] - Name, Status, Port list	Switch is registered.
getSwitchStatus	GetSwitchStatus	Return the value of the Global Health Status of the switch.	String switchIPAddress	String Status: ok(1), noncritical(2), critical(3), unreachable(4)	Switch is registered.
createVLAN	CreateVLAN	Create new VLAN on a switch.	String switchIPAddress, String vlanName, number vlanId	boolean: Status	Switch is registered.
deleteVLAN	DeleteVLAN	Remove existing VLAN from a switch.	String switchIPAddress, String vlanId	boolean: Status	Switch is registered.
addPortToVLAN	AddPortToVLAN	Add a port to an existing VLAN.	String switchIPAddress, String vlanId, String portNum	boolean: Status	Switch is registered.
removePortFromVLAN	RemovePortFromVLAN	Remove a port from a VLAN.	String switchIPAddress, String vlanId, String portNum	boolean: Status	Switch is registered.
updatePortState	UpdatePortState	Enable/Disable a switch port administratively.	String switchIPAddress, String portNum, number enable(2)/disable(3)	boolean: Status	Switch is registered.
getSwitchPortInfo	GetSwitchPortInfo	Return detailed information about a specific port.	String switchIPAddress, String portNum	String[] - Speed, Mode, State, Description, Type Note: The port type will be null on some devices.	Switch is registered.

Table 1. Actions

Action name	Workflow name	Description	Input	Output	Preconditions
changePortName	ChangePortName	Configure the user-defined name for a port on a switch for easy reference.	String switchIPAddress, String portNum, String name	boolean: Status	Switch is registered.
changePortAccess	ChangePortAccess	Change the port to Access or Trunk Mode.	String switchIPAddress, String portNum, number access (tagged(2)/untagged(3))	boolean: Status	Switch is registered.
addVLANToSTG	AddVLANToSTG	Add the VLAN to a specific spanning tree group.	String switchIPAddress, number spanningTreeGroup, number vlanId	boolean: Status	Switch is registered. Spanning tree group is already created.
removeVLANFromSTG	RemoveVLANFromSTG	Remove the VLAN from a specified spanning tree group.	String switchIPAddress, number spanningTreeGroup, number vlanId	boolean: Status	Switch is registered. Spanning tree group is already created.
createPortchannel	CreatePortchannel	Create a vLAG portchannel on a set of ports.	String switchIPAddress, number PortchannelNum, String ports (comma separated list of port numbers)	boolean: Status	Switch is registered.
enablePortchannel	EnablePortchannel	Enable the portchannel.	String switchIPAddress, number portChannelNum, number enable(1)/disable(2)	boolean: Status	Switch is registered. Portchannel is already created.
createPortChannelAdminKey	CreatePortChannelAdminKey	Create LACP portchannel.	String switchIPAddress, number adminKey	boolean: Status	Switch is registered.
removePortChannelAdminKey	RemovePortChannelAdminKey	Removes the static ID assignment from a LACP portchannel.	String switchIPAddress, number portChannelNum	boolean: Status	Switch is registered.
enableVLAGPortChannel	EnableVLAGPortChannel	Enable the vLAG portchannel.	String switchIPAddress, number portChannelNum, number vlag enable(1)/disable(2)	boolean: Status	Switch is registered. Portchannel is already created.
enableVLAGAdminKey	EnableVLAGAdminKey	Enable vLAG Adminkey.	String switchIPAddress, number adminKeyofthePortChannel, number vlag enable (1)/disable(2)	boolean: Status	Switch is registered. Portchannel is already created.
createVLAGAdminKey	CreateVLAGAdminKey	Create a new LACP on a given port.	String switchIPAddress, number portNum, number adminKeyofthePortChannel, number enable (off(1), active (2), passive (3))	boolean: Status	Switch is registered. The LACP trunk group is created with the required admin key.

Table 1. Actions

Action name	Workflow name	Description	Input	Output	Preconditions
removePortsFromPortchannel	RemovePortsFromPortchannel	Remove ports from the portchannel.	String switchIPAddress, number PortchannelNum, String ports (comma separated list of port numbers)	boolean: Status	Switch is registered.
removePortchannel	RemovePortchannel	Remove the LAG.	String switchIPAddress, number portChannelNum	boolean: Status	Switch is registered.
deleteLacpVlagAdminkey	RemoveVlagLacpAdminkey	Set port to default LACP.	String switchIPAddress, String portNum, number setToDefault (delete(2))	boolean: Status	Switch is registered.
enableUFP	UFPEnable	Enable Global UFP on the switch.	String switchIPAddress	boolean: Status	Switch is registered.
createUFPport	CreateUFPport	Enable a specific UFP port.	String switchIPAddress, String portNum, String vlanId, String ufpType, String vPortNum	boolean: Status	Switch is registered. Action is applicable only on: - G8264CS SIF internal ports. - RackSwitch ports that have been configured as server ports.
modifyUFPportBW	ModifyUFPportBW	Modify parameters on UFP port.	String switchIPAddress, String portNum, String vPortNum, String minBW, String maxBW	boolean: Status	Switch is registered.
saveConfiguration	SaveConfiguration	Save the running configuration to the start-up configuration.	String switchIPAddress	boolean: Status	Switch is registered.
addServerPort	AddServerPort	Designate a port on a RackSwitch as an server port. Not applicable to Flex devices.	String switchIPAddress, String portNum	boolean: Status	Switch is registered.
removeServerPort	RemoveServerPort	Remove a port on a RackSwitch as an server port. Not applicable to Flex devices.	String switchIPAddress, String portNum	boolean: Status	Switch is registered.
getServerPorts	GetServerPorts	Return the list of active ports that are server ports. Not applicable to Flex devices.	String switchIPAddress	String[] - port list	Switch is registered.

Table 1. Actions

Action name	Workflow name	Description	Input	Output	Preconditions
setSwitchDetails	SetSwitchDetails	Set the SNMP credentials used for the communication with the switch.	String switchIPAddress, String version, String readCommunity, String writeCommunity, String userName, String authenticationProtocol, String authenticationPassword, String privacyProtocol, String privacyPassword	Boolean	Switch is registered.
getSwitchInfo	GetSwitchInfo	Return various information about the switch.	String switchIPAddress	String[] – Next Boot Image Location, Next Boot Config Block, Image 1 Version, Image 2 Version, Boot Version, Active Image Version,	Switch is registered.
updateSwitch	UpdateSwitch	Update the switch with a new firmware image.	Number location (valid values are image1(2), image(3), boot(4), onie(5)), String filename, Number protocol (valid values are SFTP(1), TFTP(2), FTP(3)) String fileServerIPAddress Number fileServerPort String username (required for SFTP/FTP transfers) String password (required for SFTP/FTP transfers) Number switchPort (valid values are data(1) and mgt(2))	boolean: Status	Switch is registered.
getLastTransferStatus	GetLastTransferStatus	Retrieve the human readable string of the results of the last transfer action.	String switchIPAddress	String [] - lastTransferStatus	Switch is registered.
resetSwitch	ResetSwitch	Reload/reboot the switch.	Number configForNextReset (valid values are: noChange(1), active(2), backup(3), default(4)) Number imageForNextReset (valid values are: noChange(1), image1(2), image2(3), onie-install(4), onie-rescue(5), onie-uninstall(6), onie-update(7)) Number Reset the switch (valid values are: reset(2), resetStaggered(3), resetNoDump(4))	boolean: Status	Switch is registered.

Table 1. Actions

Action name	Workflow name	Description	Input	Output	Preconditions
discoverSwitchesBySubnet	DiscoverSwitchesBySubnet	<p>Discover devices in the given subnet. Return their IP addresses, sysObjectID, sysDesc, and sysName if accessible via SNMP.</p> <p>Version should be specified as snmpv1, snmpv2, snmpv3 only. readCommunity and writeCommunity should be public and private only. Range must have no more than 256 IP addresses otherwise an error status will be returned.</p>	<p>String subnetIPAddress, String subnetSubnetMask, String version, String readCommunity, String writeCommunity, String userName, String authenticationProtocol, String authenticationPassword, String privacyProtocol, String privacyPassword</p>	<p>String [] - IPAddress, status, sysObjectID, sysDesc, sysName</p> <p>Note: The value of status will be "true" if a device was discovered at that IP address. It will be "false" if there was no response from a device at that IP address.</p>	
discoverSwitchesByRange	DiscoverSwitchesByRange	<p>Discover devices in the given IP address range. Return their IP addresses, sysObjectID, sysDesc, and sysName if accessible via SNMP.</p> <p>Version should be specified as snmpv1, snmpv2, snmpv3 only. readCommunity and writeCommunity should be public and private only. Range must have no more than 256 IP addresses otherwise an error status will be returned.</p>	<p>String startIPAddress, String endIPAddress, String version, String readCommunity, String writeCommunity, String userName, String authenticationProtocol, String authenticationPassword, String privacyProtocol, String privacyPassword</p>	<p>String [] - IPAddress, status, sysObjectID, sysDesc, sysName</p> <p>Note: The value of status will be "true" if a device was discovered at that IP address. It will be "false" if there was no response from a device at that IP address.</p>	

Advanced Workflows

The following table lists Advanced Workflows that combine multiple Actions to perform a task.

Table 2. *Advanced Workflows*

Workflow name	Description	Input	Output	Preconditions	Notes
CreateSingleVLAN OnPort	This will create a VLAN on a switch based on server MAC address and VLAN number.	String serverMacAddress, String VLANNum, String VLANName	Status message on the console log and the highlighted green end point in the workflow path.	Switch is registered. LLDP is enabled on the switch.	Use actions: 1. Get Server Port Mapping 2. Verify Active Switch 3. Create VLAN 4. Apply VLAN to Port 5. Apply Configuration
CreateSingleVLAN OnVLAGPort	This will create a VLAN on switch based on server MAC address and VLAN number with the server port on the switch belonging to a Static Portchannel.	String serverMacAddress, String VLANNum, String VLANName, String portChannelNumber	Status message on the console log and the highlighted green end point in the workflow path.	Switch is registered. LLDP is enabled on the switch.	Use actions: 1. Get Server Port Mapping 2. Verify Active Switch 3. Create Port Channel on the server port. 4. Enable Port Channel 5. Create VLAN 6. Apply VLAN to Static Port Channel vLAG Port 7. Apply Configuration
CreateSingleVLAN OnLACPVLAGPort	This will create a VLAN on switch based on server MAC address and VLAN number with the server port on the switch belonging to a LACP Portchannel.	String serverMacAddress, String VLANNum, String VLANName, String portChannelNumber	Status message on the console log and the highlighted green end point in the workflow path.	Switch is registered. LLDP is enabled on the switch.	Use actions: 1. Get Server Port Mapping 2. Verify Active Switch 3. Create LACP Port Channel. 4. Create LACP Port using LACP Port Channel adminKey. 5. Create VLAN 6. Apply VLAN to LACP Port Channel Port 7. Apply Configuration

Chapter 4. Troubleshooting

vRealize Orchestrator provides an extensive logging facility for troubleshooting issues. Refer to the following document on the [vRealize Orchestrator Documentation](#) page for details on how to enable logging, change log levels and where to access the log files:

- o Installing and Configuring VMware vRealize Orchestrator

The Lenovo Networking Plug-in for VMware vRealize Orchestrator supports the following log levels:

- o INFO
- o DEBUG
- o ERROR

The following table lists the various log levels that are supported:

Table 3. *Logging Messages*

LOG_INFO	2015-05-25 07:15:23.237+0000 [WorkflowExecutorPool-Thread-18] INFO {vcoadmin:RegisterSwitchUsingAction:8a71eb5b4d89bd6d014d89ed0373009c:3d3ebb73-6413-42e6-858a-539fed85e849:[3d3ebb73-6413-42e6-858a-539fed85e849]} [RegisterSwitch] sysInfo is [1, Discovered device info... IP Address = 10.241.105.239 sysDescr = Lenovo Flex System Fabric EN4093R 10Gb Scalable Switch sysObjectID = 1.3.6.1.4.1.20301.1.18.18 sysName = compassr SNMP Version = 1 SNMP Port = 161 Security Model = v1v2 Read Community = public Write Community = private]
LOG_INFO	2015-05-25 10:28:35.430+0000 [WorkflowExecutorPool-Thread-1] INFO {vcoadmin:RegisterSwitchUsingAction:8a71eb7b4d8a9ac2014d8a9de41f0004:3d3ebb73-6413-42e6-858a-539fed85e849:[3d3ebb73-6413-42e6-858a-539fed85e849]} [SCRIPTING_LOG] [RegisterSwitchUsingAction (5/25/15 10:28:32)] Registration of the switch successful
LOG_INFO	2015-05-25 10:39:43.801+0000 [WorkflowExecutorPool-Thread-7] INFO {vcoadmin:CreateVLAN:8a71eb7b4d8a9ac2014d8aa8034b0034:5ec1d57a-3e00-4b86-a025-96c0741d1fa7:[5ec1d57a-3e00-4b86-a025-96c0741d1fa7]} [SCRIPTING_LOG] [CreateVLAN (5/25/15 10:39:36)] VLAN creation has been successful

Table 3. Logging Messages

LOG_INFO	2015-05-25 10:46:42.213+0000 [WorkflowExecutorPool-Thread-10] INFO {vcoadmin:AddPortToVLAN:8a71eb7b4d8a9ac2014d8aae6c7a0050:26de7650-eab1-4cfa-b0b6-2f2acf60f5e2:[26de7650-eab1-4cfa-b0b6-2f2acf60f5e2]} [SCRIPTING_LOG] [AddPortToVLAN (5/25/15 10:46:36)] Adding port to VLAN successful
LOG_INFO	2015-05-25 10:46:42.169+0000 [WorkflowExecutorPool-Thread-10] INFO {vcoadmin:AddPortToVLAN:8a71eb7b4d8a9ac2014d8aae6c7a0050:26de7650-eab1-4cfa-b0b6-2f2acf60f5e2:[26de7650-eab1-4cfa-b0b6-2f2acf60f5e2]} [AddPortToVLAN] Applying configuration after SET DONE
LOG_INFO	2015-05-25 10:50:20.175+0000 [WorkflowExecutorPool-Thread-11] INFO {vcoadmin:GetListOfRegisteredSwitches:8a71eb7b4d8a9ac2014d8ab1d1710058:3a9b700c-2bb5-4323-b060-0f1eaa97fa29:[3a9b700c-2bb5-4323-b060-0f1eaa97fa29]} [SCRIPTING_LOG] [GetListOfRegisteredSwitches (5/25/15 10:50:18)] Got the registered switches
LOG_INFO	2015-05-25 10:59:08.787+0000 [WorkflowExecutorPool-Thread-15] INFO {vcoadmin:GetMarsSwitchStatus:8a71eb7b4d8a9ac2014d8ab9d8da007e:52711401-2600-45af-94b7-7255f1a3a250:[52711401-2600-45af-94b7-7255f1a3a250]} [SCRIPTING_LOG] [GetMarsSwitchStatus (5/25/15 10:59:05)] getting the switch health status successful
LOG_ERROR	2015-05-25 07:24:30.955+0000 [WorkflowExecutorPool-Thread-21] ERROR {vcoadmin:CreateVLAGAdminKey:8a71eb5b4d89bd6d014d89f562ce00b3:9fc0ef8d-3b9c-419b-8063-41ef6b12f5c8:[9fc0ef8d-3b9c-419b-8063-41ef6b12f5c8]} [CreatePortChannel] ip address is not valid
LOG_ERROR	2015-05-25 10:32:06.735+0000 [WorkflowExecutorPool-Thread-3] ERROR {vcoadmin:UFPEnable:8a71eb7b4d8a9ac2014d8aa122cf0013:bfbeba47-d593-496e-af1f-156da77ccbc9:[bfbeba47-d593-496e-af1f-156da77ccbc9]} [UFPEnable] ip address is not valid
LOG_ERROR	2015-05-25 10:33:40.354+0000 [WorkflowExecutorPool-Thread-4] ERROR {vcoadmin:CreateUFPPort:8a71eb7b4d8a9ac2014d8aa29095001b:6c95e358-8d71-4434-a5f7-e33ea164c55e:[6c95e358-8d71-4434-a5f7-e33ea164c55e]} [UFPPortEnable] UFP port parameters are invalid

Table 3. Logging Messages

LOG_ERROR	2015-05-25 10:35:47.032+0000 [WorkflowExecutorPool-Thread-5] ERROR {vcoadmin:RemovePortChannel:8a71eb7b4d8a9ac2014d8aa47f8f0024:63150e59-36cf-4e07-b193-68d24dc4c085:[63150e59-36cf-4e07-b193-68d24dc4c085]} [RemovePortChannel] port channel number is invalid
LOG_ERROR	2015-05-25 10:37:18.799+0000 [WorkflowExecutorPool-Thread-6] ERROR {vcoadmin:ModifyUFPPortBW:8a71eb7b4d8a9ac2014d8aa5e613002c:7cae3d5e-6084-4398-a555-c6cab36c58b0:[7cae3d5e-6084-4398-a555-c6cab36c58b0]} [ModifyUFPPortBW] UFP port params are invalid
LOG_ERROR	2015-05-25 10:44:47.502+0000 [WorkflowExecutorPool-Thread-9] ERROR {vcoadmin:GetRemoteNodeMapping:8a71eb7b4d8a9ac2014d8aacbe9d0048:c36a11ae-6d12-4547-978b-24fcf659c075:[c36a11ae-6d12-4547-978b-24fcf659c075]} [GetRemoteNodeMapping] switch port is not valid
LOG_DEBUG	2015-05-25 10:41:38.751+0000 [WorkflowExecutorPool-Thread-8] DEBUG {vcoadmin:GetVLANInfo:8a71eb7b4d8a9ac2014d8aa9dce5003c:73f2b9dd-1df3-4b9e-893e-7d622dc17349:[73f2b9dd-1df3-4b9e-893e-7d622dc17349]} [WorkflowHandler] getAttributeFromCache WorkflowTokenAttribute [name=array, type=Array/string, value=##string#key = vlanInfoStatus.1300,value = 2#;#string#key = vlanInfoPorts.1300,value = 13;43-44#;#string#key = vlanInfoName.1300,value = VLAN 1300#]#]
LOG_DEBUG	2015-05-25 10:46:37.495+0000 [WorkflowExecutorPool-Thread-10] DEBUG {vcoadmin:AddPortToVLAN:8a71eb7b4d8a9ac2014d8aae6c7a0050:26de7650-eab1-4cfa-b0b6-2f2acf60f5e2:[26de7650-eab1-4cfa-b0b6-2f2acf60f5e2]} [WorkflowHandler] getAttributeFromCache WorkflowTokenAttribute [name=portNum, type=string, value=6] 2015-05-25 10:46:37.495+0000 [WorkflowExecutorPool-Thread-10] DEBUG {vcoadmin:AddPortToVLAN:8a71eb7b4d8a9ac2014d8aae6c7a0050:26de7650-eab1-4cfa-b0b6-2f2acf60f5e2:[26de7650-eab1-4cfa-b0b6-2f2acf60f5e2]} [WorkflowScriptRunner] Fetching portNum

Table 3. Logging Messages

LOG_DEBUG	2015-05-25 10:50:20.150+0000 [WorkflowExecutorPool-Thread-11] DEBUG {vcoadmin:GetListOfRegisteredSwitches:8a71eb7b4d8a9ac2014d8ab1d1710058:3a9b700c-2bb5-4323-b060-0f1eaa97fa29:[3a9b700c-2bb5-4323-b060-0f1eaa97fa29]} [WorkflowHandler] getAttributeFromCache WorkflowTokenAttribute [name=returnVals, type=Array/string, value=#{#string#switchIp=10.241.105.239,switchType=compassr# }#]
LOG_DEBUG	2015-05-25 10:52:06.365+0000 [WorkflowExecutorPool-Thread-12] DEBUG {vcoadmin:GetSwitchPortInfo:8a71eb7b4d8a9ac2014d8ab36cb20061:9028ba38-6619-4399-9deb-7311036b35da:[9028ba38-6619-4399-9deb-7311036b35da]} [WorkflowHandler] getAttributeFromCache WorkflowTokenAttribute [name=retArray, type=Array/string, value=#{#string#key = portInfoType.5,value = 7#;#string#key = portInfoMode.5,value = 2#;#string#key = portInfoPhyIfDescr.5,value = INTA5#;#string#key = portInfoSpeed.5,value = 5#;#string#key = portInfoPhyIfOperStatus.5,value = 2#}#]
LOG_DEBUG	2015-05-25 10:56:07.479+0000 [WorkflowExecutorPool-Thread-13] DEBUG {vcoadmin:GetSwitchPortInfo:8a71eb7b4d8a9ac2014d8ab71e8c006d:9028ba38-6619-4399-9deb-7311036b35da:[9028ba38-6619-4399-9deb-7311036b35da]} [WorkflowHandler] getAttributeFromCache WorkflowTokenAttribute [name=retArray, type=Array/string, value= __NULL__]
LOG_DEBUG	2015-05-25 10:59:08.785+0000 [WorkflowExecutorPool-Thread-15] DEBUG {vcoadmin:GetMarsSwitchStatus:8a71eb7b4d8a9ac2014d8ab9d8da007e:52711401-2600-45af-94b7-7255f1a3a250:[52711401-2600-45af-94b7-7255f1a3a250]} [WorkflowHandler] getAttributeFromCache WorkflowTokenAttribute [name=result, type=string, value=critical]

Chapter 5. Known Issues

Problem:

The UpdateSwitch workflow does not support the RackSwitch G7028 and RackSwitch G7052 devices on versions 8.1.3 and prior.

Cause:

This is due to a limitation in the switch firmware which will be rectified in a future firmware release.

Workaround:

None.

Problem:

The EnableVLAGAdminkey workflow fails on the RackSwitch G8332 when enabling an LACP port channel group.

Cause:

Presence of a VLAN on the trunk port which is part of the port channel.

Workaround:

Remove the VLAN on the trunk port.

Appendix A. Getting help and technical assistance

If you need help, service, or technical assistance or just want more information about Lenovo products, you will find a wide variety of sources available from Lenovo to assist you.

Use this information to obtain additional information about Lenovo and Lenovo products, and determine what to do if you experience a problem with your Lenovo system or optional device.

Note: This section includes references to IBM web sites and information about obtaining service. IBM is Lenovo's preferred service provider for the System X, Flex System, and NeXtScale System products.

Before you call, make sure that you have taken these steps to try to solve the problem yourself.

If you believe that you require warranty service for your Lenovo product and you have purchased the plug-in through the “Lenovo Networking Bundle for vRealize”, the service technicians will be able to assist you more efficiently if you prepare before you call.

- Go to the [IBM Support portal](#) to check for information to help you solve the problem.
- Gather the following information to provide to the service technician. This data will help the service technician quickly provide a solution to your problem and ensure that you receive the level of service for which you might have contracted.
 - Pertinent information such as error messages and logs
- Start the process of determining a solution to your problem by making the pertinent information available to the service technicians. The IBM service technicians can start working on your solution as soon as you have completed and submitted an Electronic Service Request.

You can solve many problems without outside assistance by following the troubleshooting procedures that Lenovo provides in the online help or in the Lenovo product documentation. The Lenovo product documentation also describes the diagnostic tests that you can perform. The documentation for most systems, operating systems, and programs contains troubleshooting procedures and explanations of error messages and error codes. If you suspect a software problem, see the documentation for the operating system or program.

Appendix B. Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area.

Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service.

Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.
1009 Think Place - Building One
Morrisville, NC 27560
U.S.A.

Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties.

Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary.

Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk.

Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Trademarks

Lenovo, the Lenovo logo, Flex System, System x, NeXtScale System, and X-Architecture are trademarks of Lenovo in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.