ScienceLogic

Cisco: Viptela PowerPack Release Notes

Version 200

Overview

Version 200 of the "Cisco Viptela" PowerPack converts the PowerPack and all Dynamic Applications to support Python 3 using the Snippet Framework, adds a universal credential and guided discovery workflow, and adds proxy support to the PowerPack.

- **Minimum Required SL1 Version**: 11.3.0

This document covers the following topics:

Before You Install or Upgrade 2
Installation or Upgrade Process 2
Features 3
Enhancements and Issues Addressed 3
Known Issues 4

Before You Install or Upgrade

Ensure that you are running version 11.3.0 or later of SL1 before installing the "Cisco: Viptela" PowerPack.

NOTE: For details on upgrading SL1, see the relevant ***SL1 Platform Release Notes***.

Installation or Upgrade Process

TIP: By default, installing a new version of a PowerPack will overwrite all content in that PowerPack that has already been installed on the target system. You can use the ***Enable Selective PowerPack Field Protection*** setting in the **Behavior Settings** page (System > Settings > Behavior) to prevent the new version of the PowerPack from overwriting local changes for some commonly customized fields.

To install version 200 of the "Cisco Viptela" PowerPack:

1. See the ***Before You Install or Upgrade*** section. If you have not done so already, upgrade your system to the minimum required platform version or later release.
2. Download the PowerPack from the Support Site to a local computer.
3. Go to the **PowerPack Manager** page (System > Manage > PowerPacks). Click the **[Actions]** menu and select *Import PowerPack*. When prompted, import the PowerPack.
4. After importing/upgrading the PowerPack, you will be prompted to install the PowerPack. Click the **[Install]** button to install the PowerPack.

5. To ensure all apps are collecting data after the upgrade, go to the **Credentials** page (Manage > Credentials). Click the name of the current credential to edit the credential. Add *%silo_token=X-XSRF-TOKEN* as a header in the **HTTP Headers** section if it is not present already, and click the **[Close]** button.
6. Go to the **Device Manager** page (Devices > Device Manager). Identify the vManage virtual root device and click the Device Properties icon. In the Collections tab, select the **[Actions]** menu and click the **[Add Dynamic Application]** button. Select the "Cisco: Viptela Token Configuration" Dynamic Application with the current credential and save it. Perform these same steps to align the "Cisco: Viptela vManage IP Address Configuration" Dynamic Application to the root device. Follow these steps if the two Dynamic Applications involved are not aligned already. Otherwise, skip this step.
7. See the manual ***Monitoring Cisco Viptela*** for instructions on using the new PowerPack.

NOTE: To discover vEdge devices with only "valid" statuses, modify the "Cisco: Viptela vEdge Discovery" Dynamic Application snippet arguments to filter by "valid" by changing the `[*]` section to use `[?validity=='valid']`. For example, the `serialNumber` snippet argument uses the following: `rest://dataservice/system/device/vedges&silo_args=jpath=data[?validity=='valid'].serialNumber|serialNumber`

Features

This release includes the following features:

- Dynamic Applications that enable SL1 to discover, model, and monitor Cisco: Viptela devices.
- Device Classes for each type of Cisco: Viptela device monitored.
- Event Policies and the corresponding alerts that are triggered when Cisco: Viptela devices meet certain status criteria.
- An example credential that you can use as a template to create a SOAP/XML credential to connect to the Cisco: Viptela component devices that you want to monitor.
- Dashboards that display information about Cisco: Viptela component devices.
- "The Cisco: Viptela vManage Template" is available for aligning all Dynamic Applications to the root component device.

Enhancements and Issues Addressed

The following enhancements and addressed issues are included in this release:

- Added the new "Cisco: Viptela Environment v200" Execution Environment to support conversion to the Snippet Framework.
- Converted the PowerPack and all Dynamic Applications to support Python 3 using the Snippet Framework.
- Added the new "Cisco SDWAN/Viptela" universal credential type and guided discovery workflow.
- Made a number of updates to the "Cisco: Viptela Events" Dynamic Application:

    - Consolidated functionality from the "Cisco: Viptela Events Caching" Dynamic Application into this Dynamic Application.
    - Added the new "API Alarms" collection object to trigger the API request.
    - Changed page size to 40 to reduce the number of requests made to the API.
    - Added a query parameter to reduce the number of alarms retrieved from the API to only those from the last hour. This can be customized in the snippet by adjusting the polling frequency.

> **NOTE:** Upon upgrading from a version of this PowerPack using Python 2, duplicate alerts may be created due to incompatibility of the cached values from Python 2 to Python 3.

- Deprecated the following Dynamic Applications:
    - Cisco: Viptela Token Configuration
    - Cisco: Viptela Events Caching
- Added proxy support to the PowerPack.

> **NOTE:** The HTTPS proxy scheme is only available using the new universal credential . Existing SOAP/XML credentials do not support switching the scheme.

- You can now change the logout method in the universal credential for API versions that require a POST request upon log out instead of a GET request. (Case: 00474177)
- You can now configure the CA path and SSL certificate peer verification in the universal credential.
- Both device and site alarms/events are now processed correctly for the "Cisco: Viptela Events" Dynamic Application. (Case: 00339100)

Known Issues

The following known issues affect version 200 of the "Cisco: Viptela" PowerPack:

- If you are monitoring vManage version 20.12 or later, you will need to update the GetAlarms API call parameter due to a change in the API from Viptela. (Case: 00443285) (Jira ID: SOL-27385)

  On the **[Snippets]** tab of the "Cisco: Viptela Events Caching" Dynamic Application, select **create_cache.py** and replace the existing code:

  ```
  path = "alarms/page?count=20"
  ```

  with the following workaround code:

  ```
  path = "alarms"
  ```

- There might be a collection gap in the "Cisco: Viptela Events Caching" Dynamic Application when the "Cisco: Viptela Token Configuration" Dynamic Application logs out of a previous session. This could

happen approximately every 12 hours.

- "Cisco: Viptela Tunnel Statistics Performance" Dynamic Application might present collection gaps if the API is not returning data in the period of time configured in this Dynamic Application.
- "Cisco: Viptela TLOC Aggregation Configuration" Dynamic Application might not display data in its configuration report if the API is not returning data in the period of time configured in this Dynamic Application.
- If there is a large amount of data to analyze on the database for the "Cisco: Viptela TLOC Aggregation Configuration" Dynamic Application, you might experience a 500 (Internal Server Error) error, which will be displayed in the inspector web console. As a troubleshooting method, go to /etc/php.d/sl1_memory_limit.ini to increase the memory_limit and restart services with systemctl restart em7 php-fpm nginx.

© 2003 - 2025, ScienceLogic, Inc.

All rights reserved.

LIMITATION OF LIABILITY AND GENERAL DISCLAIMER

ALL INFORMATION AVAILABLE IN THIS GUIDE IS PROVIDED "AS IS," WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED. SCIENCELOGIC™ AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT.

Although ScienceLogic™ has attempted to provide accurate information on this Site, information on this Site may contain inadvertent technical inaccuracies or typographical errors, and ScienceLogic™ assumes no responsibility for the accuracy of the information. Information may be changed or updated without notice. ScienceLogic™ may also make improvements and / or changes in the products or services described in this Site at any time without notice.

Copyrights and Trademarks

ScienceLogic, the ScienceLogic logo, and EM7 are trademarks of ScienceLogic, Inc. in the United States, other countries, or both.

Below is a list of trademarks and service marks that should be credited to ScienceLogic, Inc. The ® and ™ symbols reflect the trademark registration status in the U.S. Patent and Trademark Office and may not be appropriate for materials to be distributed outside the United States.

- ScienceLogic™
- EM7™ and em7™
- Simplify IT™
- Dynamic Application™
- Relational Infrastructure Management™

The absence of a product or service name, slogan or logo from this list does not constitute a waiver of ScienceLogic's trademark or other intellectual property rights concerning that name, slogan, or logo.

Please note that laws concerning use of trademarks or product names vary by country. Always consult a local attorney for additional guidance.

Other

If any provision of this agreement shall be unlawful, void, or for any reason unenforceable, then that provision shall be deemed severable from this agreement and shall not affect the validity and enforceability of any remaining provisions. This is the entire agreement between the parties relating to the matters contained herein.

In the U.S. and other jurisdictions, trademark owners have a duty to police the use of their marks. Therefore, if you become aware of any improper use of ScienceLogic Trademarks, including infringement or counterfeiting by third parties, report them to Science Logic's legal department immediately. Report as much detail as possible about the misuse, including the name of the party, contact information, and copies or photographs of the potential misuse to: legal@sciencelogic.com. For more information, see https://sciencelogic.com/company/legal.

800-SCI-LOGIC (1-800-724-5644)

International: +1-703-354-1010