SDAVCConnector

A demo connector for pushing application rules to Cisco SD-AVC

V1.0.0 supports reading domains from a CSV file or Infoblox DNS server, writing to Cisco SD-AVC external connector API. A configuration file used for configuration variables

The code is given as an example for an SD-AVC connector and can be used or modifiled by any party. The code owners do not responsible for any issue, bug or damage caused by using this code or any of the associated files.

Dependent packages: The connector depend on the following 3rd party artifacts, user is required to follow their license requirements:

Artifact name	Group ID	Version
json-simple	com.googlecode.json-simple	1.1.1
jersey-client	org.glassfish.jersey.core	2.26-b03
junit	junit	4.12
log4j	log4j	1.2.13
commons-cli	commons-cli	1.4

Running:

An execution jar exist in Connectors-1.0.0-SNAPSHOT.jar

Run using: java -jar Connectors-1.0.0-SNAPSHOT.jar -t [infoblox|csv] -c config_file

Examples:

CSV file: java -jar Connectors-1.0.0-SNAPSHOT.jar -t csv -c file.cfg.json

Infoblox DNS server: java -jar Connectors-1.0.0-SNAPSHOT.jar -t infoblox -c infoblox.cfg.json

Logging:

Depend on log4j. A sample log4j.xml file exist with "info" level logging. To disable logging, modify the log4j.xml file with "off" instead of "info".

Configuration file:

Sample configuration files for csv and infoblox respectively could be found at file.cfg.json and infoblox.cfg.json The configuration files are in JSON format. The first section include:

- the reading poll time (poll_time)
- IP address of SDAVC (sdavc_ipv4addr)
- username/password to log into SD-AVC (sdavc_login) and (sdavc_passwd).

The configuration file can have a list of entities to read from:

CSV file entity:

Include:

- The connector name (name), this name will be presented in the external connectors screen in SD-AVC
- The file name (filename)
- Segment (segment). The segment is the SD-AVC segment rules are provisioned for. The CSV file is comma separated CSV without a header that include the application name and the application FQDN. Multiple FQDNs could be provisioned for the same application.

Infoblox entity:

Include:

- The connector name (name), this name will be presented in the external connectors screen in SD-AVC.
- The infoblox server ipv4 address (ipv4addr)
- The vrf name is currently not in use (vrf_name)
- The infoblox API version (wapi_version)
- Whether to read the FQDNs from the TXT DNS records or Extensible attributes stored in the DNS A record (application_info). Set to "TXT" or "A" respectively.
- User and password for the infoblox to read the FQDNs (user) and (passwd)
- The connector reads FQDNs in a specific zone listed by the (zone_filters) filed. Multiple entities can be used for multiple zones
- The segment (segment) is the SD-AVC segment rules are provisioned for For either TXT records or Extensible attributes, the application information uses the following fields:

Field	Meaning
app-name	Application Name
app-set	Application Set
app-class	Traffic Class
business	Business Relevance

Field	Meaning
app-category	Category
app-sub-category	Sub-Category

The TXT format starts with "CISCO-CLS=" and following with the fields values separated by a vertical bar "|". For example, the following provisions application MyApp with traffic-class=Bulk-Data and business-relevance=Yes: CISCO-CLS=app-name:MyApp|app-class:BULK-DATA|business:YES When using Extensible attributes, each of the fields should be a separate extensible attributes ("CISCO-CLS=" is not used).

SD-AVC External Connector API:

Description can be found here

V1.0.0 limitations:

- Application attributes are not supported for CSV file reader
- Only FQDN rules supported, IP/Tuple based rules are not supported
- No certification validation, both access to Infoblox and SD-AVC skip certificate validation