Juniper Secure Analytics

Reference Data Collections

Release 2014.4
Reference Data Collections
Release 2014.4
Copyright © 2015, Juniper Networks, Inc.
All rights reserved. Printed in USA.
Revision History
March 2015 — Reference Data Collections
The information in this document is current as of the date listed in the revision history.

END USER LICENSE AGREEMENT
The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement (“EULA”) posted at http://www.juniper.net/support/eula.html, as modified by the following text, which shall be treated under the EULA as an Entitlement Document taking precedence over any conflicting provisions of such EULA as regards such software:

As regards software accompanying the STRM products (the “Program”), such software contains software licensed by Q1 Labs and is further accompanied by third-party software that is described in the applicable documentation or materials provided by Juniper Networks.
For the convenience of Licensee, the Program may be accompanied by a third party operating system. The operating system is not part of the Program, and is licensed directly by the operating system provider (e.g., Red Hat Inc., Novell Inc., etc.) to Licensee. Neither Juniper Networks nor Q1 Labs is a party to the license between Licensee and the third party operating system provider, and the Program includes the third party operating system "AS IS", without representation or warranty, express or implied, including any implied warranty of merchantability, fitness for a particular purpose or non-infringement. For an installed Red Hat operating system, see the license file: /usr/share/doc/redhat-release-server-6Server/EULA.

By downloading, installing or using such software, you agree to the terms and conditions of that EULA as so modified.
CONTENTS

1 REFERENCE DATA COLLECTIONS
About Reference Data Collections ........................................... 7
Creating a Reference Data Collection ....................................... 8
Utility Command Reference ..................................................... 10
Use the ReferenceDataUtil.sh utility to make complex reference data collections, such as a Reference Map, Reference Map of Sets, and Reference Map of Maps.

This technical note applies to Juniper Secure Analytics (JSA) and Log Manager.

Reference data collections enable the storage, retrieval and testing of complex data structures. You can create the following reference data collection types:

- **Reference Map** - In a Reference Map, data is stored in records that map a key to a value. For example, to correlate user activity on your network, you can create a reference map that uses the Username parameter as a key and the user’s global ID as a value.

- **Reference Map of Sets** - In a Reference Map of Sets, data is stored in records that map a key to multiple values. For example, to test for authorized access to a patent, you can create a Map of Sets that uses a custom event property for **Patent ID** as the key and the **Username** parameter as the value to populate a list of authorized users.

- **Reference Map of Maps** - In a Reference Map of Maps, data is stored in records that map one key to another key, which is then mapped to single value. For example, to test for network bandwidth violations, you can create a Map of Maps that uses the **Source IP** parameter as the first key, the **Application** parameter as the second key, and the **Total Bytes** parameter as the value.

For each reference data collection, you specify what the element type is for the collection. You can specify the following element types:

- Alphanumeric
- Alphanumeric Ignore Case
- IP
- Numeric
- Port
- Date
Creating a Reference Data Collection

Using the ReferenceDataUtil.sh utility, you can create a reference data collection.

Before you begin

If you plan to load an external file containing data elements, ensure that the file is in Comma Separated Value (CSV) format. Also ensure that you have copied the file to your JSA system.

The file must follow the format in the following examples reference data collections:

Example 1

```
# # ReferenceMap
# key1,data
key1,value1
key2,value2
```

Example 2

```
# # ReferenceMapOfSets
# key1,data
key1,value1
key1,value2
```

Example 3

```
# # ReferenceMapOfMaps
# key1,key2,data
map1,key1,value1
map1,key2,value2
```

The # symbol in the first column indicates a comment line. The first non-comment line is the column header and identifies the column name (ie., key1, key2, data). Then each non-commented line after that is a data record that gets added to the map. Keys are alphanumeric strings.

About this task

See Utility Command Reference for a list of all commands and parameters you can use to manage your reference data collections. You can also type `./ReferenceDataUtil.sh` and press Enter to access a list these commands
Creating a Reference Data Collection

Procedure

Step 1  Using SSH, log in to JSA as the root user:
Username: root
Password: <password>

Step 2  Create a reference data collection:
   a  To change to the /opt/qradar/bin directory, type the following command:
      cd /opt/qradar/bin
   b  To create the reference data collection, type the following command:
      ./ReferenceDataUtil.sh create <name> <count> [MAP | MAPofSETS | MAPofMAPS] [ALN | NUM | IP | PORT | ALNic | DATE] [timeout_type] [timeToLive]

Step 3  To populate the map with data from an external file, type the following command:
   ./ReferenceDataUtil.sh load <name> <filename> [-encoding=...] [-sdf="..."]

What to do next

Log in to the JSA user interface to create rules that add data to your reference data collections or rule tests that detect activity from elements in your reference data collection. For more information on creating rules and rule tests, see the Juniper Secure Analytics Users Guide.
Use the following commands to manage your reference data collections:

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameters</th>
</tr>
</thead>
</table>
| create  | <name> is the name of the reference data collection.  
          |           
          | <count> is the maximum number of elements that the reference data collection can contain.  
          |           
          | [timeout_type] specifies whether the timeToLive is from the time the element was inserted (0) or last seen (1).  
          |           
          | [timeToLive] specifies the amount of time reference data collection elements remain in the collection.  
          |           
          | [MAP | MAPofSETS | MAPofMAPS] specifies the type of reference data collection.  
          |           
          | [ALN | NUM | IP | PORT | DATE] is the type of data in the reference set, where:  
          |           
          | • ALN specifies a reference data collection of alphanumeric values. This data type supports IPv4 and IPv6 addresses.  
          |           
          | • ALNic specifies a reference data collection of alphanumeric values but tests ignore the case. This data type supports IPv4 and IPv6 addresses.  
          |           
          | • NUM specifies a reference data collection of numeric values.  
          |           
          | • IP specifies a reference data collection of IP addresses. This data type supports only IPv4 address.  
          |           
          | • PORT specifies a reference data collection of PORT addresses.  
          |           
          | • DATE specifies a reference data collection of DATE values.  
| update  | <name> is the name of the reference data collection.  
          |           
          | <count> is the maximum number of elements that the reference data collection can contain.  
          |           
          | [timeout_type] specifies whether the timeToLive is from the time the element was inserted (0) or last seen (1).  
          |           
          | [timeToLive] specifies the amount of time reference data collection elements remain in the collection.  
| add     | <name> is the name of the reference data collection.  
          |           
          | <value key1 [key2]> specifies the values you want to add. Key1 is required for MAP, MAPofSETS and key2 is required for MAPofMAPS. Keys are alphanumeric strings.  |
Table 1-1 Command Reference

<table>
<thead>
<tr>
<th>Command</th>
<th>Parameters</th>
</tr>
</thead>
</table>
| delete  | `<name>` is the name of the reference data collection  
          `<value key1 [key2]>` specifies the values you want to delete. Key1 are required for MAP, MAPofSETS and key2 is required for MAPofMAPS  
          `[-sdf=" ... "]` specifies the Simple Date Format string used to parse the date data. |
| remove  | `<name>` is the name of the reference data collection. |
| purge   | `<name>` is the name of the reference data collection. |
| get     | `<name>` is the name of the reference data collection  
          `[loadElements]` displays all elements in the specified reference data collection. |
| getall  | `[loadElements]` displays all elements in all reference data collections. |
| load    | `<name>` is the name of the reference data collection  
          `<filename>` is a fully qualified filename to be loaded, with each line in the file representing a record to be added to the reference data collection  
          `[-encoding=...]` specifies encoding to use when reading the file.  
          `[-sdf=" ... "]` specifies the Simple Date Format string used to parse the date data. |