Add-on for Regional Internet Registry Consolidated Extended Statistics Tables

TA-rirstats

Version 3

Frank Wayne

Table of Contents

Overview

About

The Add-on for Regional Internet Registry Consolidated Extended Statistics Tables (TA-rirstats) creates and maintains a lookup containing all the network ranges documented by the five regional Internet registries (RIRs). Splunk queries can use this lookup to determine with what subnet[1] a particular public IP address is associated. The lookup also includes type address type, the registry responsible for the subnet, the date the registration was changed, the status, the country the registration is in, and a registration ID for the party to whom the registration was made.

Source types

Source type	Description
rirstats:log	Information and error messages from the Python module responsible for collecting and processing data from the RIRs.

This source type appears only in the _internal index. The add-on puts no data into non-internal indexes.

Release notes

Features

Version 3 adds the *type* column to the lookup. The Python code has been completely rewritten using the requests HTTP library.

Known issues

There are no known issues at this time.

Third-party software attributions

This add-on uses original Python code with libraries distributed with Splunk. It does not incorporate any third-party software.

Installation overview

1. Download the add-on from Splunkbase.
2. Install the add-on.
3. Run the lookup refresh script for the first time manually.

[1] The IPv4 space is divided by the RIRs into ranges, not subnets. Thus, an individual allocation or assignment can include several adjacent subnets. For example (using private address space), a range might start at 10.0.0.0 and include 384 addresses. This range includes 10.0.0.0 through 10.0.1.127. This cannot be expressed as a single subnet, but rather as 10.0.0.0/24 plus 10.0.1.0/25. When a RIR describes a range this way, the add-on breaks it up into the minimal number of subnets required to describe that range.

Installation

Hardware and software requirements

Splunk admin requirements

To install and configure the add-on, you must be a member of the admin role.

Internet access

This add-on refreshes the RIR registration data automatically using HTTP. The data refresh can only work if the search heads that have the add-on installed can access the Internet on port 80/TCP.

Splunk platform requirements

This add-on runs on Splunk Enterprise. All the requirements for Splunk Enterprise apply.

The add-on performs CIDR lookups, which requires that the table reside in memory during the process. The size of the lookup table exceeds the default Splunk memory table limits. You must manually increase **`max_memtable_bytes`** for the **`[lookup]`** stanza in **limits.conf** so that the lookup can run. This may increase search head memory utilization as a result. I recommend increasing the limit to 64MiB like this in $SPLUNK_HOME/etc/system/local/limits.h:

```
[lookup]
max_memtable_bytes = 67108864
```

If you do not increase this value, your searches will be "indexed on disk" according to the documentation. In practice (on version 9.0), your lookup will produce **no results** and produce no errors.

In case you are wondering, using the KV Store for these lookups is not practical. On my Xeon-based test machine running Splunk Enterprise 8.0.3 on Red Hat, a CSV-based lookup took 17.8 µs per event, whereas a KV Store lookup with an accelerated subnet field took 7,782.3 µs per event.

Install

1. Get the add-on from Splunkbase.
2. Determine where to install the add-on in your deployment.
3. Perform the installation.
4. Run the lookup refresh script for the first time manually.

Distributed deployments

The search tier needs this add-on in order to perform the lookup. You may need to install it in more than one place, depending on your search environment.

You need to set the **limits.conf** requirement on the indexers if you have a distributed environment with discrete tiers. You can do this by adjusting **limits.conf** on the server directly or via a deployment server (for stand-alone indexers) or from the cluster master (for indexer clusters). Do not install the add-on on the indexers.

Splunk Instance type	Supported	Required	Comments
Search Heads	Yes	Yes	Install this add-on on each search head that needs to perform the lookup. Change **limits.conf**.

Indexers	No	No	Only a **limits.conf** change is required.
Heavy Forwarders	No	No	Not applicable.
Universal Forwarders	No	No	Not applicable.
Light Forwarders	No	No	Not applicable.

Distributed deployment feature compatibility

Distributed deployment feature	Supported	Comments
Search Head Clusters	Yes	You can install the add-on on a search head cluster.
Indexer Clusters	No	Only a **limits.conf** change is required.
Deployment Servers	Yes	You can deploy the add-on to stand-alone search heads.

Other installation considerations

If you have already overridden `[lookup]/max_memtable_bytes` in **limits.conf**, be sure that your setting is high enough to provide enough memory for the lookup table (35MiB as of 2024-03), and keep in mind that the table will grow over time. If the setting is too small, your lookups will not generate results.

Initial Lookup Refresh

Some of the lookup data distributed with the add-on will be stale by the time you install it. To get the latest data immediatenly after installation, go to the UI of the search head (or one of the members of the search head cluster) and run the **TA-rirstats Refresh Lookup** search to perform a refresh.

Upgrade

Upgrading From Version 2 or 1

Version 3 of this add-on places the lookup in its own lookup directory. Previous versions stored the lookup in the system/lookups directory.

To make sure the new version works as intended, do the following:

1. Uninstall any previous version of the add-on by removing the TA-rirstats directory from $SPLUNK_HOME/etc/apps/ or, for a search head cluster, removing it from $SPLUNK_HOME/etc/shcluster/apps/ on the deployer and pushing the bundle to the members.
2. Restart Splunk on the search head(s).
3. Remove the rirstats.csv lookup file manually. Using the UI, go to Settings, Lookups, Lookup table files. Find any lookup files or lookup file indexes called “rirstats.csv”. Delete them using the delete action on the right of the table.
4. Install the new add-on.
5. (Optional) Run the “TA-rirstats Refresh Lookup” report to get the latest data.

Configuration

Search head configuration

The add-on does not *require* configuration for the search tier. The only requirement after installation is the creation of the lookup table, as described **Error! Reference source not found.**.

Distributed indexer configuration

However, if there is a separate indexer tier, you must change the **`max_memtable_bytes`** value as described **Error! Reference source not found.**. Otherwise, if a search head pushes a knowledge bundle to the indexer tier that includes the **`rirstats`** lookup, the indexer will report an error because the table size exceeds the default memory limit. (The add-on changes this limit for the search head tier, which is why no special configuration is required there.)

A work-around for running the lookup without changing limits on the indexer tier is to use **`local=true`** as an option for the lookup command. For example:

```
… | lookup local=true rirstats subnet AS src_ip OUTPUT subnet
```

Changing the automatic table refresh schedule

By default, the lookup table is refreshed with the latest RIR data every Sunday at 03:00, with a 1 hour window, using the scheduled search **`TA-rirstats Refresh Lookup.`** An administrator can change this schedule by creating a **local/savedsearches.conf** file and overriding the search stanza. For example, to run the refresh every day at midnight, use the following:

savedsearches.conf

```
[TA-rirstats Refresh Lookup]
cron_schedule = 0 0 * * *
schedule_window = 15
```

Each RIR updates their registration data daily. Refreshing the lookup more frequently is unlikely to be advantageous.

Using the add-on

Examples

Example 1: Getting subnet information

To use the add-on, find public IP addresses that you want to look up. Many inputs produce events containing a src_ip field, for example. To get counts of errors by IP address and include subnet information, you might do this:

```
error src_ip=*
| stats count by src_ip
| lookup rirstats subnet AS src_ip OUTPUT subnet country registry
```

Unfortunately, the subnet field is also the lookup field, which means that the lookup command does not return it by default. You must include it in the OUTPUT expression to get it.

Example 2: Using subnet information for the **`stats`** command

To count events by subnet, use output from the lookup:

```
error src_ip=*
| stats count BY src_ip
| lookup rirstats subnet AS src_ip OUTPUT subnet country registry date status
| stats sum(count) AS count first(country) AS country first(registry) AS registry BY
subnet
```

Example 3: A simple test with ad-hoc data

If you have no data ready, but want to test the lookup with addresses that should work, try:

```
| makeresults 1 | eval src_ip = "2620:10D:2000::1"
| append [ makeresults 1 | eval src_ip = "129.105.136.70" ]
| lookup rirstats subnet AS src_ip OUTPUT subnet country registry date status
| eval date = strftime(date, "%F")
| table src_ip subnet country registry date status
```

This should generate two results with details.

Available lookup fields

Field name	Data type	Comments
country	string	The ISO 3166-1 alpha-2 country code of the organization to which the range is allocated or assigned.
date	String	The date of the allocation or assignment in ISO extended format.
reg_id	string	A unique organization identifier.[2]
registry	string	The regional registry ID, e.g.: 'ripencc', or 'arin'.
status	string	One of 'available', 'allocated', 'assigned', or 'reserved'.
subnet	string	A subnet in CIDR format.
type	string	The IP address type, e.g.: 'ipv4' or 'ipv6'.

[2] "All records in the file with the same reg-id are registered to the same resource holder." (Extended Allocation and Assignment Report of RIRs, version 2.3.)

Troubleshooting

General troubleshooting

See the Splunk resource for add-on troubleshooting for helpful information.

Error messages

Informational and error messages are logged to `rirstats.log` in Splunk's log directory. You can find these records in the `_internal` index (look for `source=*rirstats.log`).

When I run a search, I get the error: `[instance.domain.tld] Streamed search execute failed because: Error in 'lookup' command: Error using lookup table 'rirstats': CIDR and wildcard matching is restricted to lookup files under the in-memory size limit.`

> You did not change limits.conf on the search heads or the indexers to provide an override for the **`max_memtable_bytes`** value.

In index _internal, I find the error: `ERROR TA-rirstats get_rirstats.py download of http://...`, or I find the error `<timestamp> ERROR TA-rirstats get_rirstats.py download encountered exception http://....`

> The search head tried to download from of the RIR websites, but failed. The reason follows the URL in the event. This is usually caused by failure to resolve the name (DNS), a network problem, a firewall policy, or a problem with the target site.

In index _internal, I find the error: `<timestamp> ERROR TA-rirstats get_rirstats.py parsing encountered exception....`

> Data from one of the registries could not be parsed. Something is wrong with the data or something has changed. Contact the author with details.

About the author

General

Frank Wayne is a Senior Systems Engineer and Splunk Core Consultant working at Northwestern University in Evanston, Illinois. He has a Master of Science and Bachelor of Philosophy, both in Information Systems, from that university and has been working in IT for over twenty years.

Contact information

Email: frank.wayne@northwestern.edu