Flow Indexing

Making queries go faster

FloCon

11 January 2012

John McHugh
RedJack LLC

How is large scale flow data used?

“Selection” is the most generic and most popular query type. Selection queries specify a source IP address, a destination IP address, or both”

- Publication NetSA-2011-19

- When flow data is being searched to corroborate information from other sources, entry is often via an IP address or set of IP addresses.
- While other information (flowtype primarily) may limit the scope of the data pull, it is common to read through many files containing no relevant records.

Motivation: Alert Processing

- Alerts from IDS or similar sensors are often corroborated (or refuted) with flow data.
- The alert typically has a time window (stime / etime)and one or more IP addresses (or {src/dst} pairs}.
- Data pull for the alert examines all the data within the time window (+/- an error margin) for all of the IPs.
  - Many of the examined files have no relevant data
  - Sometimes there is no data satisfying the pull criteria (differences in sensor placement and source visibility).
- Avoid fruitless work whenever possible

Can we short circuit the data pull?

- Indices are the traditional mechanism for short circuiting sequential searches.
  - A book index usually points to a specific page.
  - SiLK flow files were not designed for random access, and the index terms (IP addresses) often take up 25% or more of each record so a traditional *term -> page* index is not a solution.
- On the other hand, absence of an index entry means that we don’t have to read the book.
  - If there are lots of missing entries, our reading list is limited.

The SiLK file repository (refresher)

- For the data set we are using, files are stored in a directory hierarchy

```
 <data_rootdir>/
     class/
          type/
              year/
                  month/
                      day/
```

- day/ directory contains per sensor hourly files with names of the form: <flowtype>-<sensor>_YYYYMMDD.HH
- May be up to (#class X #type X #sensor) files to search

```
rwfglob –start-date=YYYY/MM/DD:HH –flowtype=all/all
```

provides a complete list for a given hour

IPsets as indices

- For each flow file, we create IPsets for its source and destination addresses and a union containing both.
- Now, questions like

  “Which, if any, of the hourly files contains data with:

  a) Source IP == aaa.bbb.ccc.ddd or
  b) Destination address == mmm.nnn.ooo.ppp or
  c) Either source or destination address == www.xxx.yyy.zzz”

  and IPset queries can be answered from the indices.
- Index files are much much smaller than the flow files. Querying indices is much faster than filtering flows, especially when there are a small number of hits and we can avoid reading some flow files.

Automagic index creation

- We observe that flow data usually settles about 30 – 45 minutes after the close of the flow hour.
  - Run a script to create indices on a timed basis.
    - Can be done within the existing framework
    - Refresh if index is out of date (again timed) or
    - Refresh on attempt to use stale index (in access scripts)
  - Make index creation part of the packing process
    - Saves a pass over the data.
    - Index updated with late additions to the flow file.
    - Ensures consistency
    - Requires changes to packing process

Benefits

- Substantial savings in data pulls.
  - No need to pass files without selected IPs
  - Negative answers fast, no data, no waiting
  - Reduced processing and analyst workloads
- Potential for long term studies from indices alone
  - When did we first see www.xxx.yyy.zzz
  - How often are we visited by aaa.bbb.ccc.ddd
  - What sites were active in fff.ggg.0.0/16 in May 09
- Can even aid in “tuple” searches
  - If there is src -> dst traffic, it will be in files where src is in the sIPset and dst is in the dIPset.

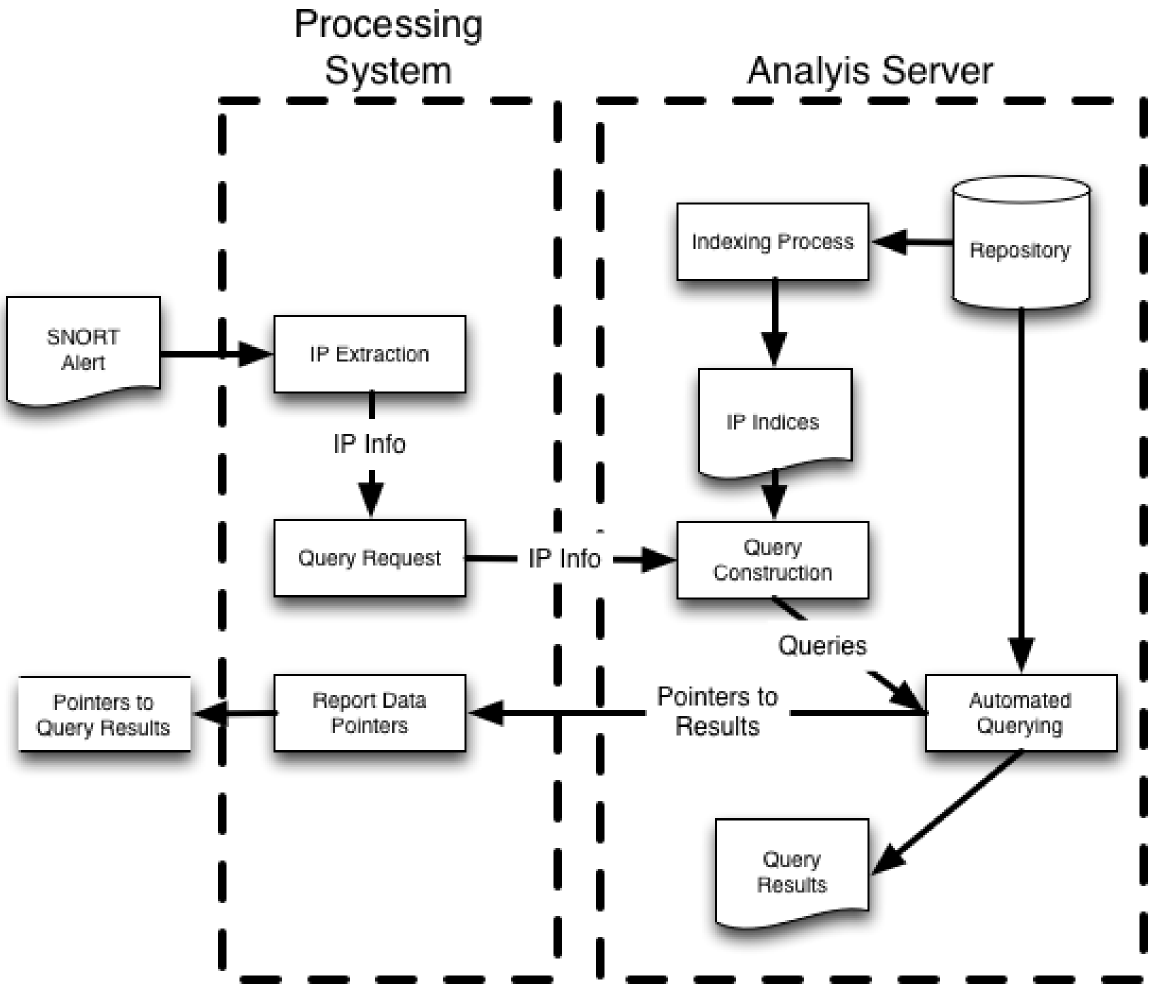
Possible Improvements / Extensions

- External IP set representation can be made much more compact allowing faster queries.
  - Set / Bag union / intersection can be streamed. Avoid building structures in memory.
- Consolidated hourly indices can be formed using bags. Up to 64 hourly files searched in 1 stream op.
- Dedicated index servers using large ram / SSD for processing recent indices.
- Index hierarchy with hourly, monthly, 5 year indices
  - hourly index detail for protocols, volumes, etc. if deemed useful.

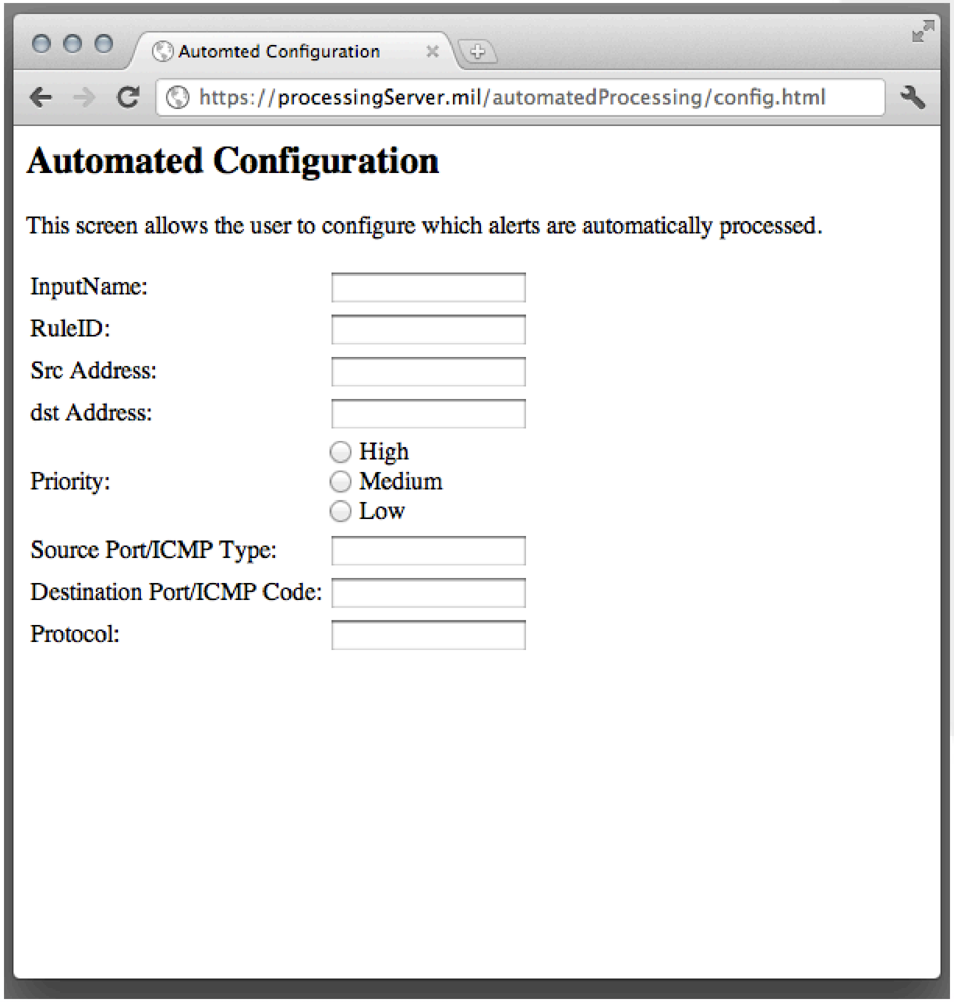
Current Status

- Currently have
  - An indexing script available on the analysis servers
  - A script for constructing repository queries using indexes
- We have 9 sensors, 4 flowtypes = 36 files/hour
  - being able to skip 2 files pays for the index search
  - big payoff most of the time based on limited trial
- Near-term goals
  - Deploy an automated tool for parsing IP address information out of SNORT alerts

Parsing Snort Alerts Automatically



Configuration Capability


Automted Configuration
https://processingServer.mil/automatedProcessing/config.html

Automated Configuration

This screen allows the user to configure which alerts are automatically processed.

InputName:

RuleID:

Src Address:

dst Address:

Priority: ○ High ○ Medium ○ Low

Source Port/ICMP Type:

Destination Port/ICMP Code:

Protocol:

- Tuning to what the analyst actually cares about
  - System ignores all alerts it is not explicitly configured to handle
- Automation will be configurable through a web interface
  - Filtering based on fields within the alert

Conclusions

- Indexing of flow is effective and inexpensive.
- For a large class of queries, it can significantly reduce query time by eliminating files from consideration.
- Everything can be done within the SiLK framework

but

- Some reorganization of the data and improvements in the tools (more efficient data structures, tool approaches, multi-key sets) could make things much better.
- Routine indexing of {sip,dip}, ports, protocols and of size, rate, frequency, etc., distributions with graphical presentation (RDD tool?) would be a big plus.

Questions

Talk to me!

- I’ll be around through Thursday noon
- You can reach me through RedJack