Flexible Web Visualization for Alert-Based Network Security Analytics

Lihua Hao¹, Christopher G. Healey¹, Steve E. Hutchinson²

¹North Carolina State University, ²U.S. Army Research Laboratory

lhao2@ncsu.edu

VizSec’13 October 14, 2013, Atlanta, GA, USA
Introduction

• Building a visualization tool for Army Research Laboratory (ARL) network security analysts

• Driven by analysts
  - Our approach does not focus explicitly on network security data, but rather on network security analysts
  - “Don’t fit our problem to your tool. Build a tool to fit our problem.”

• We must balance
  1. Meeting needs of the analysts.
  2. Applying knowledge and best practices from visualization.
Design Constraints

1. Mental models
   - “Fit” the mental models the analysts use to investigate problems

2. Working environment
   - Integrate into the analyst’s current working environment (web browser for ARL analysts)

3. Configurability
   - Static, pre-defined presentations of the data are typically not sufficient

4. Accessibility
   - The visualizations should be familiar to analysts, to avoid steep learning curves

5. Scalability
   - Support query and retrieval from large data sources

6. Integration
   - Augment the analyst’s current problem-solving strategies with useful support
Existing Visualization Techniques

- **Node-link graphs**
  - Portall, HoNe, LinkRank

- **Treemaps**
  - NetVis, NFlowVis

- **Timelines and Event Plots**
  - Aggregate value over events
  - Capture patterns of individual events

- **Basic Charts**
  - Snorby, NVisionIP

- **Zooming, Multivariate**
  - NVisionIP: galaxy, small multiple, and machine views
  - VisFlowConnect: global, domain, internal, and host views
**Data Management**

- MySQL & PHP running on a remote server
  - Provide reasonable *scalability*, efficient data filtering and projection
- No pre-defined table formats
  - Analysts choose columns to visualize, define table correlations and data filtering
  - Provide flexibility and *configurability*
- Cache results of current query in memory
  - Generate queries to retrieve the new data on demand
- Full SQL is available on demand to the analyst
  - System suggests visualization with automatically generated SQL queries
  - Analysts can manually *configure* system suggestions
Web-Based Visualization

- ARL analysts work in a browser
  - “Fit” analysts’ working environment

- HTML5 canvas element
  - No external plug-ins required
  - Run in any modern web browser

- Use 2D charts
  - Common in other security visualization systems
  - Effective for presenting values, trends, patterns and relationships our analysts want to explore
  - Provides accessibility
Analyst-Driven Charts

• RGraph for basic chart visualizations
  - Open source library for visualization with 2D charts
  - Choose charts commonly used in network data visualization

• Assisted chart selection based on data and task (*accessibility*)
  - Pie/bar: proportion and frequency comparison
  - Bar: value comparison over a secondary attribute
  - Scatterplots: correlation between two attributes
  - Gantt: range value comparison

• Suggested chart properties
  - Backgrounds, grids, glyph size, color and type

• Free to change the initial choices
Interaction

- Intelligent zoom
  - Redraw chart to include only the selected chart elements
  - Rescale the visual attributes of chart elements
- Tooltips for value query
  - Display the exact attribute values encoded in a chart element
  - Provide access to quantitative data on demand
- Toolbars
  - Customize glyph size, color, size
  - Change chart title, size, label width, and so on
Correlated Views

- A sequence of visualizations to track an ongoing investigation
  - Correlate multiple data sources
  - Explore data at multiple levels of details

- Correlated charts
  - Select sub-regions of a chart as input for a following, correlated chart
  - Generate constraints to extract data of interest
  - Add additional constraints, tables, or attributes

- Raw data spreadsheets for value examination
  - Text-based examination: a conventional approach
  - “Fit” the analyst’s working environment, mental models
Trap Data

• Need real world data to test the system
• For security reasons, not possible to use data from ARL for testing
• The trap server
  - Data collected by network security researchers at NCSU
  - Real world network traffic in Computer Science building
  - Transmitted to a Snort sensor to perform: (1) intrusion detection and (2) extraction of network packets
  - Stores two types of data: (1) NetFlow data and (2) Snort alerts
• An example file for 24 hours of data
  - 17.4GB of packet headers
  - 938K unique source IPs, 168K unique destination IPs
  - 1.6M flows with 615K alerts
Example Tables

- Tables queried in the visualization
  - **event**: alert signature id and timestamp
  - **flows**: network flow sources and destination IP, port, start and end time
  - **iphdr**: source and destination IP and other information of packet headers
  - **tcphdr**: TCP related information such as source and destination port

- One of our research colleagues acted as the “analyst” in our scenario
Aggregate Alerts on Destination IPs

- Visualize number of alerts for each destination IP
- Pie chart, proportion of alerts by destination IP
- Bar chart, absolute numbers of alerts by destination IP
- The majority of the alerts are sent to destination IP 172.16.79.134
- “Sub Canvas” in the tooltip to create correlated chart for target destination IP
Focus on High-Alert Destination IP

- Focus on the destination IP with the maximum number of alerts (i.e., 172.16.79.134)
- Scatterplot of an analyst-chosen source IP versus the target destination IP and port
- Sizes of scatterplot glyphs indicate number of alerts from the source to the destination/port
- Analyst requests a text table detailing the exact IPs, ports, and alert counts
- Most alerts are sent to port 21 (894 alerts), so follow-on analysis will focus on this port
NetFlows for Target Destination IP and port

- Visualize netflow traffic related to the target destination IP on port 21
- Zoom to examine details in left and right flow clusters

- Right flow contains only one alert, does not look suspicious
- Most alerts happened in left flow, may contain attack
- Analyst decides to perform further analysis of traffic associated with left flow
  - E.g., include more tables and attributes to perform deeper analysis

Flows are distributed over two time ranges

Majority of alerts occur in left flow. Look suspicious

Right flow has single alert
Summarization of the Example

• Major steps supported by our visualization tool:
  - High level aggregation to highlight destination IPs with numerous alerts
  - Scatterplots to examine relationship between source IP and suspicious destination IP’s ports
  - Correlated netflow visualization to examine timeline of alerts
  - Further analysis will focus on traffic related with the left flow

• Analysts focus on the data they are interested in at a given point in an investigation
• Easy to request follow-on visualizations and modify them to pursue new hypotheses and investigate new findings as they are uncovered
Future work

• Analysis Sandbox
  - Individual analyses can be performed, stored, reviewed and compared
  - Improve an analyst’s “working memory” capacity

• Analysis Preferences
  - Track an analyst’s actions to better anticipate their strategies for specific types of tasks
  - Use preference elicitation algorithms to track an analyst’s interest within a visualization session

• Real-world Validation
  - Not allowed to speak directly with the analysts
  - Coordinate with IT staffs who support the analysts
Contact Information

Christopher G. Healey
Department of Computer Science
North Carolina State University

healey@ncsu.edu

Special Thanks

Peng Ning, CS Department, NC State University
Doug Reeves, CS Department, NC State University
Cliff Wang, Army Research Office