IVACS - Interactive Visual Analytics for Cyber Security

James Elder*
Centre for Vision, Speech and Signal Processing
University of Surrey

Dr. Eng-Jon Ong
Centre for Vision, Speech and Signal Processing
University of Surrey

Prof. Richard Bowden
Centre for Vision, Speech and Signal Processing
University of Surrey

Abstract

The ability to efficiently analyse large network datasets is extremely important for cyber security. We present our tool IVACS (Interactive Visual Analytics for Cyber Security) to aid an analyst in the discovery of cyber security threats. IVACS features database connectivity for efficient storage of "Big Data" and is capable of automating the discovery of frequent patterns using frequent itemset mining. The results are then visualised through multiple cross-linked visualisations. A number of interaction methods are available in addition to a parallel co-ordinate representation of the raw data, for context and further detail. In addition, IVACS provides a textual search panel allowing the user to search for specific values to be highlighted along with other co-occurring events.

IVACS has been tested using the firewall log dataset from the VAST Challenge 2012 and has been proven to provide both a more extensive, interpretable and natural view of the data and a more efficient method for the discovery of behavioural patterns within the data.

Keywords: Network security and intrusion, data filtering, coordinated and multiple views, multidimensional data.

1 System Overview

IVACS is composed of 3 modules (Figure 1) written primarily in Python (using the matplotlib library created by Hunter [2]) and OpenGL.

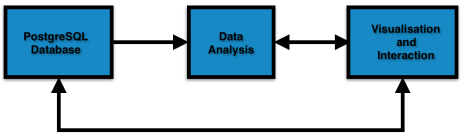

Figure 1: The system modules with the data flow shown as arrows.

The indexed PostgreSQL database allows for efficient storage and querying of "Big Data". This is coupled with a data analysis module which performs frequent itemset mining, using the apriori implementation by Borgelt [1], in order to both summarise and extract patterns from the data. The resulting mined patterns and the original raw data are then passed to the visualisation and interaction module, which consists of a series of panels to visualise and allow interaction from differing perspectives. This method does not restrict the tool to a single purpose and thus can aid in the detection of both new and existing attacks of varying behavioural signatures.

The first 4 panels visualise the results of the frequent itemset mining through a bar chart, a clock glyph, a pixel map and a line plot, with the ability to fullscreen a single visualisation into a separate panel. To provide context, the fifth visualisation provides a parallel co-ordinate plot of the raw data with user selectable axis ordering. Each of these 5 visualisations feature cross-interaction, whereby user interaction in one panel will filter and update the other visualisations. A textual output panel presents the user with the results of their interactions. These 6 panels are displayed in Figure 2. Additionally, a textual search panel allows for highlighting of user specified values with relevant co-occurences, as shown in Figure 3. This search window also allows the use of boolean operations in order to focus or expand the search to additional dimensions or values of the same dimension.

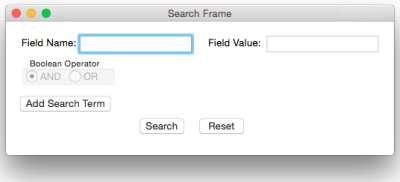

Figure 3: The search window for the user to enter specific values to highlight within the visualisations.

2 Results

For the user testing we used the firewall logs dataset from the VAST Challenge 2012. To perform the testing we tasked 5 users of varying levels of experience with 3 simple tasks. They were asked to perform the tasks using an Excel spreadsheet of the raw dataset and subsequently through the IVACS visualisation. For both methods, the user was provided with a practice session whereby they were familiarised with the functions and methods required.

The Excel spreadsheet provided to the users composed an anonymised representative subset of the dataset, consisting of 100,000 transactions (approximately 0.42% of the full dataset). This restriction is due to the size constraints of the data which can be opened in Excel. For the IVACS test the entire dataset consisting of approximately 23,000,000 transactions was used. In order to identify as many patterns as possible, the support for the frequent itemset mining was set sufficiently low at 1%. The mining was performed on hourly subsets, set to only output maximal itemsets and any itemsets not present in at least 5 hours were discarded.

The three tasks are as follows:

- Find a list of unique destination IP addresses corresponding to the ports: 21, 22, 23, 6667.
- Find the total number of rows containing one of these IP addresses.
- Find the total number of rows containing one of the above 4 ports.

Every 30 seconds, for each of the tests, the data discovered by the user was recorded. The average total completion time, shown in Figure 4, for the Excel method is 375s compared to 235s for the IVACS method. Remember, the IVACS test was performed on the complete dataset (approximately 230 times larger than that used for the Excel test).

*e-mail: j.elder@surrey.ac.uk

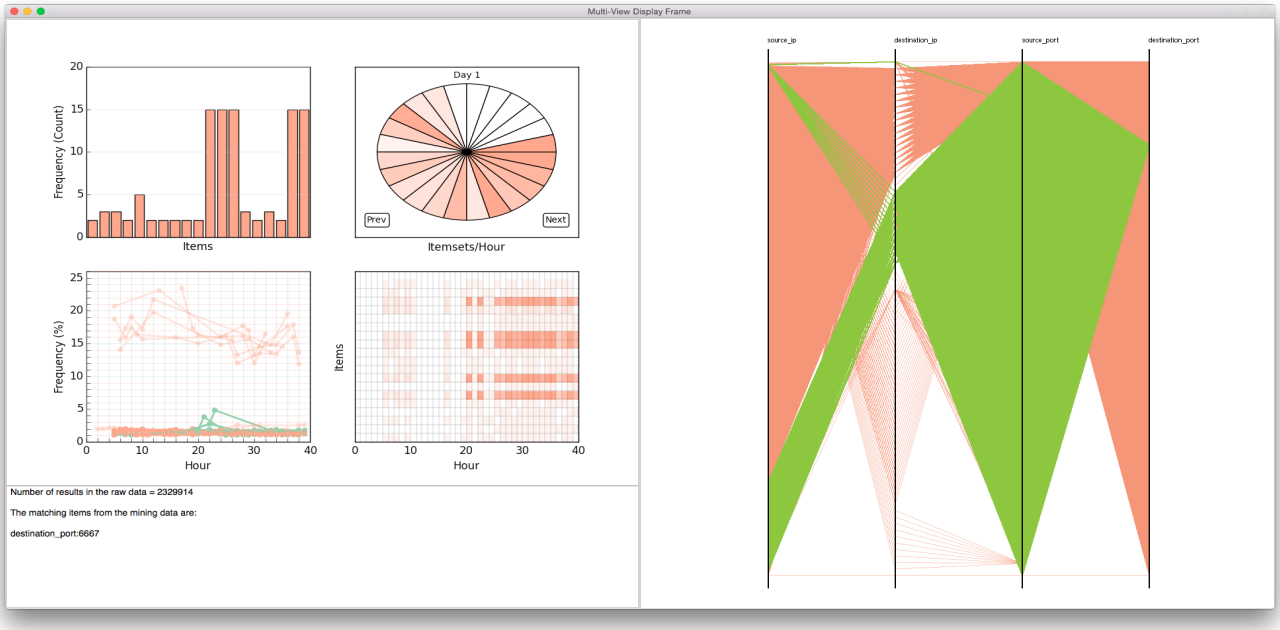

Figure 2: The main visualisation window showing the 5 visualisation methods and the textual output panel. The highlighted data corresponds to a search performed for a destination port of 6667 within the firewall log dataset of the VAST Challenge 2012.

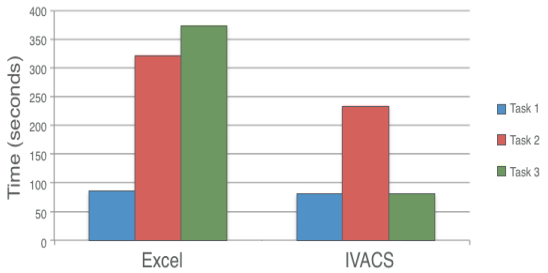

Figure 4: The average time to task completion. The average total completion time using Excel is 375s vs. only 235s using IVACS.

Figure 5 shows the percentage completion recorded every 30 seconds. Using IVACS, tasks 1 and 3 complete simultaneously at the point 1 with task 2 completing at point 2. Most of this time is due to database queries requiring no user effort. However for the Excel method tasks must be performed individually and in order.

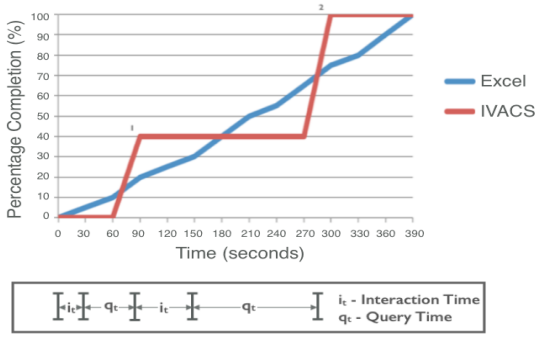

Figure 5: Completion percentage over 30 second intervals. For the IVACS method tasks 1 and 3 occur simultaneously. Most of the completion time for the IVACS method is due to database queries.

3 Conclusion

To conclude, in this paper we have presented our tool IVACS which comprises a database backend for efficient storage of "Big Data", frequent itemset mining for automatic pattern detection and cross-linked visualisations of the mining results and the raw data.

We aimed to prove the benefit, in terms of the reduction of the temporal overhead and the user effort, available through combining these methods in the discovery process of cyber attacks within large datasets.

We have performed user testing of our tool using the firewall log dataset of the VAST 2012 challenge comparing it to a manual method using Excel. Our tool not only allows the user to discover the embedded attack faster but also requires less user effort. Our tool is also able to cope with much larger dataset sizes than is otherwise possible.

Although the user testing confirms that there is a clear benefit we would like to perform further development introducing additional visualisation and interaction methods. We would also like to perform further user testing trials and to test the tool on different datasets.

References

[1] Borgelt, C. (2012). Frequent item set mining. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(6):437–456.

[2] Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing In Science & Engineering*, 9(3):90–95.