SIGRAD 2015
L. Kjelldahl and C. Peters (Editors)

WOW-A-Cluster! A Visual Similarity-Based Approach to Log Exploration

J. E. Twellmeyer[1] A. Kuijper[2] and J. Kohlhammer[1,2]

[1]Fraunhofer IGD, Germany
[2]TU Darmstadt, Germany

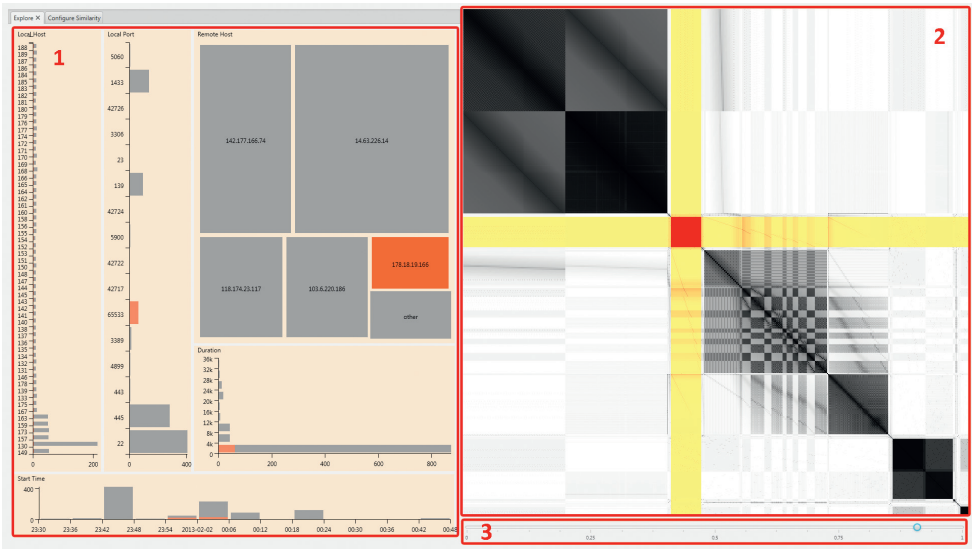

Figure 1: *The WOW-A-Cluster! prototype. The log values for each entry are shown in appropriate charts on the left (1). A matrix sorted by cluster label displays the log entry similarities on the right (2). The clustering threshold can be adjusted with the slider on the bottom right(3). Brushing a cluster in the matrix causes its values in the charts to be highlighted.*

Abstract

We present our work on a visual, similarity-based approach to log file exploration. The use of similarity rather than simple aggregation schemes empowers users to focus on the high-level events behind log entries, rather than the entries themselves. We make use of an accelerated version of TRIAGE to determine the similarity coefficients for each pair of log entries. The model is embedded in an interactive visualization system which enables the fluid interpretation of similarities with the help of a simple clustering approach.

Categories and Subject Descriptors (according to ACM CCS): I.5.3 [Pattern Recognition]: Clustering—Similarity measures

1. Introduction

Although logs or log files play an important role in fields, such as auditing, administration, security and forensics, there are few visual approaches to their exploration. In this paper we present our progress on a similarity-based approach for visual log exploration.

A log is a sequence of machine-generated information entries. Examples of entries include security alerts, TCP connections and events in a process. Each entry has a time stamp and may have other important features. This definition corresponds to those given in the literature [Kre14, CS12].

Many logs are generated at a high level of granularity (such as TCP connections). Forensic analysts and network administrators are generally not interested in individual TCP connections, but rather in the high-level events which caused those TCP connections (such as a port scan). Multiple high-level events occurring simultaneously often have overlapping sets of log entries, which may obscure one another, making interpretation difficult. In order to solve this problem, state-of-the-art tools apply feature-based or index-based aggregation schemes, combined with small-multiples views on log entries. While these approaches do provide users with useful overviews, high-level events may not be revealed by simple aggregation schemes.

Thonnard et al. proposed a clustering approach to the analysis of security events called TRIAGE [TMD10, Tho10]. Clustering partitions the log into sets of entries, such that similar entries are in the same set and dissimilar entries are in different sets [KR09]. Obtaining a measure of similarity for log entries is a challenge. The TRIAGE approach assumes that features (or attributes) can be extracted from the log entries. Examples of features for a TCP connection include the source IP address, the start time and the target port. Similarity coefficients are calculated for the entries based on each feature. These similarity coefficients are then combined to form a unified similarity (see Figure 2). A key innovation of the TRIAGE approach was the use of aggregation functions, such as Ordered Weighted Averaging [Yag88] for this step. These aggregation functions enable users to integrate domain knowledge into the similarity-modeling process.

While Thonnard's approach is effective, it was not interactive; each run required expert parameterization and delivered results after minutes or hours. The aim of our prototype is to enable users in the field to apply TRIAGE to their logs on the fly. To achieve this, users must be able to configure the similarity model and receive immediate visual feedback. In addition, they must be able to interactively explore the resultant similarities in order to interpret high-level events.

In our initial prototype we have chosen to use the WOWA aggregation function [Tor96], due to its relative simplicity and effectiveness in practice. The WOWA function is parametrized with two weight vectors as shown in Figure 2. The first vector weights similarity coefficients based on their source feature. The second vector weights similarity coefficients based on their magnitude.

2. Related Work

Our presentation of related work focuses on two key areas; log file exploration and the visualization of similarities.

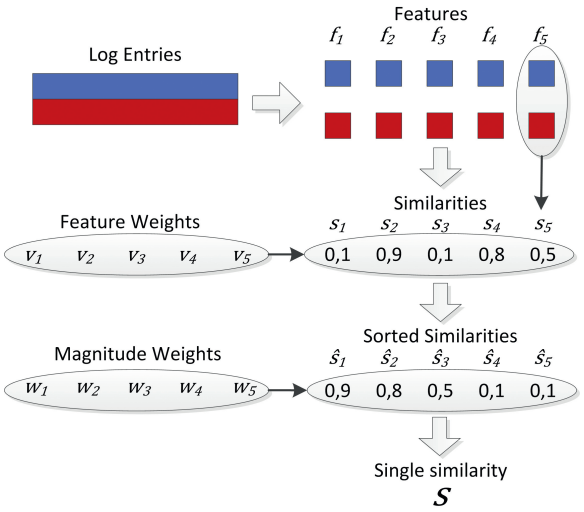

Figure 2: *Schematic illustration of the components of the WOWA function.*

While there is a lot of related work dealing with specific log file types, there are very few published general approaches to log file exploration. Progress has been made in specific domains, such as process mining [Aal11]. However, no general theory of log file analysis has emerged to date [CS12].

A recent system for the visual exploration of log files is ELVIS [HPBM13], which provides a set of feature types and corresponding visualizations for their exploration. The software vendor splunk offers an index-based, rather than a feature-based approach to log file exploration [Spl]. Both ELVIS and the splunk system include set of linked charts, as well as methods for interactive searching and filtering, and feature-based (ELVIS) or index-based (splunk) aggregation. However, neither of these approaches employ clustering to aggregate log entries into semantically meaningful groups, with respect to a multidimensional similarity model.

Numerous approaches exist for the exploration of clusters, such as parallel coordinates plots (PCPs), scatterplot matrices and sankey diagrams, which all rely on visualizing entries and their feature values. In contrast, matrices provide a view on the similarities between entries, and were proposed as a visual aid for exploratory data analysis by Jacques Bertin in 1967 [BB10]. Two prominent recent examples of matrix-based visualizations are MatrixExplorer [HF06] and NodeTrix [HFM07]. These approaches both combine matrices with node-link diagrams; MatrixExplorer provides the user with two coordinated views on the same data and NodeTrix combines the views to a hybrid visualization. Ghoniem et al. and Keller et al. conducted user studies comparing matrices and node-link diagrams [GFC05, KEC06] and concluded that matrices were a better choice for large, dense graphs in information retrieval tasks. The usefulness of ma-

Remote host	Remote port	Local host	Local port	Start time	Duration
10.20.69.98	14067	174	3389	2013-02-01T01:31:22	1
10.214.32.177	1854	165	445	2013-02-01T01:31:23	712
10.229.0.51	43	130	42540	2013-02-01T01:31:28	1

Table 1: *A sample of the dataset considered in our usage scenario*

trix visualizations is highly dependent on the applied seriation (or ordering) [MML07]. CLUSION [SG03] uses a coarse seriation algorithm to provide users with a quick, compact overview of similarities with respect to a given clustering. The authors compared their approach with PCP and projection techniques to illustrate the usefulness of matrices in cluster assessment. Twellmeyer et al. presented a linked, matrix-based visualization for the exploration of security event logs clustered with the help of TRIAGE [THB*15]. The feature-based similarity matrices are displayed alongside the aggregated matrix to enable exploration. The result was an abstract view on the data and the prototype was based on static clustering results and did not enable users to modify similarity or clustering parameters.

We use a matrix as a primary means for the exploration of log entry similarities and combine this with appropriate visualizations for the feature values. Our pipeline is designed for the interactive exploration of logs; enabling parameter adjustments with almost immediate feedback.

3. Approach and Usage Scenario

In this section we present our approach and illustrate it with a typical usage scenario. For the usage scenario we used a log obtained from a HoneypotMe [GGP] instance installed in a real network. Each entry in the log represents a malicious TCP connection with a host in the monitored network and consists of the features start time, duration (in milliseconds), the source IP address, the target host (anonymised) and the source and target ports (see Table 1).

The WOW-A-Cluster! pipeline is illustrated in Figure 3. Once the user has selected a data subset, the full pipeline is executed to display the data. The user is first presented with an overview of the data produced with the help of a default parametrization. Thereafter any changes to the parametrization of the aggregation function or clustering algorithm lead to updates. Only those parts of the pipeline are executed, which are required for the update.

The prototype has three panels; a configuration panel to configure the WOWA function, the matrix view and an exploration panel containing views of the feature values. In our usage scenario, the user configures the WOWA aggregation function in the configuration pane to reduce noise and sharpen the clusters in the data. The user then switches from the configuration to the exploration panel (see Figure 1).

The entries are clustered using graph-based clustering methods, because the WOWA function does not guarantee that the aggregated similarity values fulfill the triangle inequality (a prerequisite for distance and density based methods). We chose the search for connected components [HT73] for our prototype, because it runs in linear time based on the number of log entries. A slider enables the user to specify the minimum aggregated similarity value for which two entries are considered connected. A variant of the coarse seriation proposed by Strehl and Gosh [SG03] is used to give the matrix its characteristic block-diagonal form. The clustering and the corresponding seriation are updated when the slider is moved. Using the threshold slider, the user is able to increase or reduce the granularity of the clustering to examine groups and subgroups.

The clusters are selectable in the matrix view, which is linked with the exploration panel. Selecting an entry cluster highlights its values in the exploration panel. By selecting salient clusters the user is able to identify clear blocks of connections from specific IP-addresses (high-level events), but also the steps of an attacker (sub-events). Examples of prominent sub-events identified in the usage scenario include port scans (an attacker testing each port of a specific host), horizontal scans (an attacker testing the same port on every host) and sustained periods of communication between the attacker and a specific host on a specific port. Examining the similarities and clusters in this way enables the user to identify the high-level events recorded in the log, and to explore the processes and phases involved in these events.

4. Conclusions & Future Work

We presented a visual, similarity-based approach to log exploration. Our prototype is an accelerated version of TRIAGE, which enables the interactive exploration of log files. The TRIAGE model is embedded in an interactive visualization system which enables the fluid interpretation of similarities with the help clustering. We demonstrated our prototype in a usage scenario based on a real-world log. We were able to show the effective identification and interpretation of high-level events in the log.

At present it is possible to interactively explore 1 000 log entries with our prototype. We are currently exploring approaches to increase this number to around 5 000 entries, which would provide a reasonable compromise between the detail available to users and the level of interactivity. The WOWA function produces good results, however it requires some explanation before users can use it effectively. In ad-

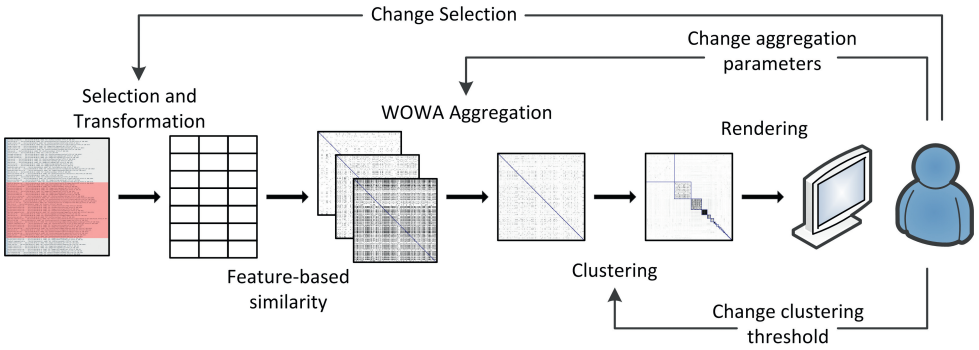

Figure 3: *The WOW-A-Cluster! pipeline. A subset of the log data is selected and mapped to features. Feature-based similarities are calculated and then aggregated using the WOWA function. The data are then clustered. Users have three leverage points in the pipeline: changes to the selection, the aggregation parameters and the clustering threshold.*

dition, other aggregation functions enable more flexibility in modeling entry similarity. Thus there are two further avenues of future work; integration of other aggregation functions and research into more intuitive methods of parameterizing these (e.g. through direct manipulation). Finally, we are currently acquiring a group of appropriate end users for involvement in a field study to evaluate our prototype.

References

[Aal11] AALST W. V. D.: *Process Mining: Discovery, Conformance and Enhancement of Business Processes*, 2011 edition ed. Springer, New York, Apr. 2011. 2

[BB10] BERTIN J., BERG W. J.: *Semiology of graphics: Diagrams, networks, maps*, 1st ed ed. ESRI Press and Distributed by Ingram Publisher Services, Redlands and Calif, 2010. 2

[CS12] CHUVAKIN A. A., SCHMIDT K. J.: *Logging and Log Management: The Authoritative Guide to Understanding the Concepts Surrounding Logging and Log Management*, 1 edition ed. Syngress, Amsterdam, Dec. 2012. 2

[GFC05] GHONIEM M., FEKETE J.-D., CASTAGLIOLA P.: On the readability of graphs using node-link and matrix-based representations: a controlled experiment and statistical analysis. *Information Visualization 4*, 2 (2005), 114–135. 2

[GGP] GASSEN J., GERHARDS-PADILLA E.: HoneypotMe. https://bitbucket.org/fkie_cd_dare/honeypotme, retrieved on 17/04/2015. 3

[HF06] HENRY N., FEKETE J.: MatrixExplorer: a Dual-Representation System to Explore Social Networks. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (Sept. 2006), 677–684. 2

[HFM07] HENRY N., FEKETE J.-D., MCGUFFIN M. J.: NodeTrix: a Hybrid Visualization of Social Networks. *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (2007), 1302–1309. 2

[HPBM13] HUMPHRIES C., PRIGENT N., BIDAN C., MAJORCZYK F.: ELVIS: Extensible Log VISualization. In *Proceedings of the Tenth Workshop on Visualization for Cyber Security* (New York, NY, USA, 2013), VizSec '13, ACM, pp. 9–16. 2

[HT73] HOPCROFT J., TARJAN R.: Algorithm 447: Efficient Algorithms for Graph Manipulation. *Commun. ACM 16*, 6 (June 1973), 372–378. 3

[KEC06] KELLER R., ECKERT C. M., CLARKSON P. J.: Matrices or node-link diagrams: which visual representation is better for visualising connectivity models? *Information Visualization 5*, 1 (2006), 62–76. 2

[KR09] KAUFMAN L., ROUSSEEUW P. J.: *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, 2009. 2

[Kre14] KREPS J.: *I Heart Logs: Event Data, Stream Processing, and Data Integration*, 1 edition ed. O'Reilly Media, Oct. 2014. 2

[MML07] MUELLER C., MARTIN B., LUMSDAINE A.: A comparison of vertex ordering algorithms for large graph visualization. In *Asia-Pacific Symposium on Visualisation 2007* (2007), pp. 141–148. 3

[SG03] STREHL A., GHOSH J.: Relationship-Based Clustering and Visualization for High-Dimensional Data Mining. *INFORMS Journal on Computing 15*, 2 (2003), 208–230. 3

[Spl] SPLUNK INC.: Operational Intelligence, Log Management, Application Management, Enterprise Security and Compliance. http://www.splunk.com/, retrieved on 17/04/2015. 2

[THB*15] TWELLMEYER J., HUTTER M., BEHRISCH M., KOHLHAMMER J., SCHRECK T.: The Visual Exploration of Aggregate Similarity for Multi-dimensional Clustering. In *Proceedings of International Conference on Information Visualization Theory and Applications* (Mar. 2015), pp. 40–50. 3

[Tho10] THONNARD O.: *A Multi-Criteria Clustering Approach to Support Attack Attribution in Cyberspace*. PhD thesis, Ecole Nationale Supérieure des Télécommunications, Paris, 2010. 2

[TMD10] THONNARD O., MEES W., DACIER M.: On a multi-criteria clustering approach for attack attribution. *ACM SIGKDD Explorations Newsletter 12*, 1 (2010), 11. 2

[Tor96] TORRA V.: Weighted OWA operators for synthesis of information. In *IEEE 5th International Fuzzy Systems* (1996), pp. 966–971. 2

[Yag88] YAGER R. R.: On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Transactions on Systems, Man, and Cybernetics 18*, 1 (1988), 183–190. 2