Hierarchical Clustering and Visualization of Aggregate Cyber Data

Robert M. Patton, Justin M. Beaver, Chad A. Steed, Thomas E. Potok, Jim N. Treadwell
Applied Software Engineering Research
Oak Ridge National Laboratory
Oak Ridge, USA
{pattonrm, beaverjm, steedca, potokte, treadwelljn }@ornl.gov

***Abstract*—Most commercial intrusion detections systems (IDS) can produce a very high volume of alerts, and are typically plagued by a high false positive rate. The approach described here uses Splunk to aggregate IDS alerts. The aggregated IDS alerts are retrieved from Splunk programmatically and are then clustered using text analysis and visualized using a sunburst diagram to provide an additional understanding of the data. The equivalent of what the cluster analysis and visualization provides would require numerous detailed queries using Splunk and considerable manual effort.**

Keywords-hierarchical clustering, sunburst visualization, IDS analysis, vector space model

I. INTRODUCTION

Cyber attacks on enterprise IT systems are growing in complexity and remain very difficult to detect due to the enormous volumes of traffic data that is available for analysis. While there are a variety of commercial intrusion detections systems (IDS) available that produce alerts when a signature or condition is found in a network packet, these systems can produce a very high volume of alerts, and are typically plagued by a high false positive rate. Consequently, many organizations have resorted to using IT data management, such as Splunk [1], to handle the aggregation and management of the volumes of alert or message data as well as assisting in operational intelligence of the network. Although the storage, management, and retrieval capabilities of IT data management systems are very mature, they lack advanced analytic capabilities beyond simple statistical analysis.

There are many available IDS tools that address niche areas of detection. These tools are relatively mature and stable, and while weak at reliably detecting actual threats, they perform well in terms of highlighting potential concerns about anomalous behavior. A core concept for cyber security event correlation is to focus on leveraging the available suite of IDS tools, and to provide an additional layer of analysis that can more reliably identify attacks, minimize the false positives, and detect anomalous behaviors. The work described here is not intended to replace or recreate the set of commercially or freely available IDS software components, or their capabilities, but is focused on analyzing the data produced by those systems to improve their collective reliability.

A significant challenge in the correlation of IDS outputs is the varied format that each tool produces. In a correlation engine, all data formats would need to be normalized into a common format in order to be compared and aggregated effectively. Creating a common format that accounts for the fields associated with each niche area and tool would be a very meticulous and time-consuming task. Furthermore, a common format would suffer from a lack of flexibility because the addition of any new IDS output would likely require an adjustment to the underlying data structures. We address this issue by focusing on the raw text as the unifying mechanism for IDS correlation. A consistent property of IDS tools is that they produce raw text outputs for analysis by a human being. Common terms and phrases, therefore, are the means to link events across niche IDS tools. Our work leverages extensive research, experience and tools in processing raw text to more effectively correlate cyber security events, and bypass the need for complex normalized data structures.

Our approach to monitoring network traffic uses multiple tools such as Snort [2] or Bro [3] dispersed throughout the enterprise network. These tools then feed their textual output to a Splunk server. Splunk is the mechanism for aggregating the IDS alerts, but does not correlate results based on raw text and has no concept of textual similarity. In our approach, the aggregated IDS alerts are retrieved from Splunk programmatically and are then clustered using text analysis to provide an additional understanding of the data. The equivalent of what the cluster analysis and visualization provides would require numerous detailed queries using Splunk and considerable manual effort.

The following sections provide a brief background, a description of the clustering and visualization approach, and future work.

978-1-4577-9538-2/11/$26.00 ©2011 IEEE

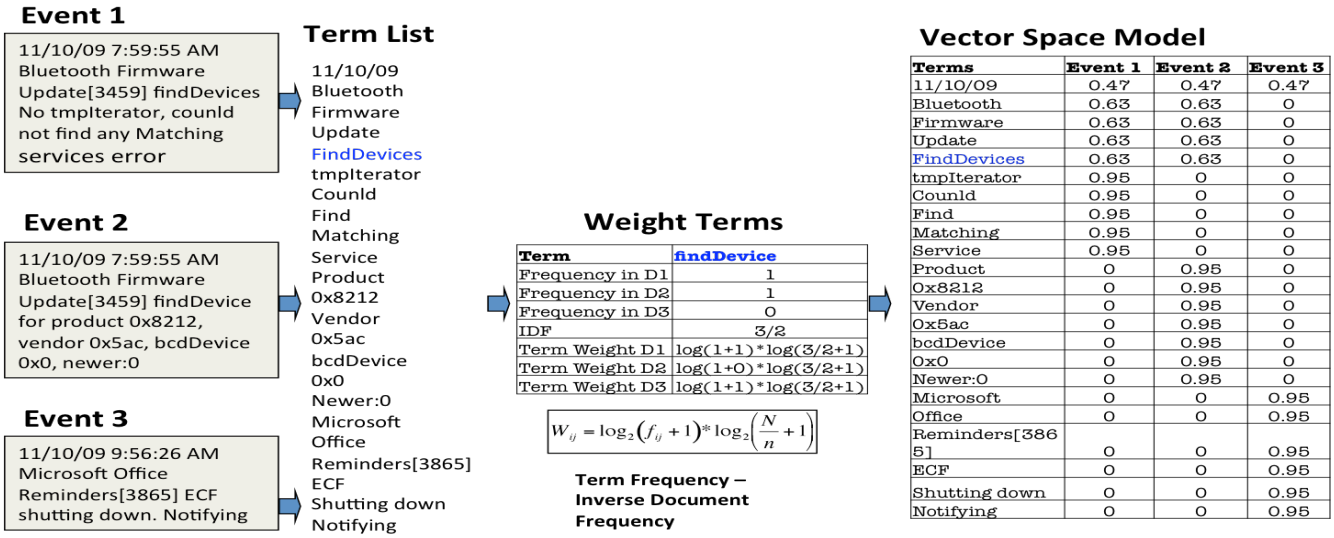

Figure 1. Vector space model representation

II. BACKGROUND

The application of clustering techniques for anomaly detection in cyber security has been well researched. Hendry [4] applied the Simple Log-file Clustering Tool (SLCT) density-based clustering algorithm to the 1999 DARPA Intrusion Detection Evaluation Set [5], and investigated its performance. The author concluded that while SLCT was too sensitive to user parameters and percentage of malicious content in clusters to serve as a stand-alone IDS, the high cluster integrity is evidence that clustering can serve as an efficient means to summarize a large collection of security events. In contrast to [4], the work described here uses hierarchical clustering that does not require user parameters, and is based on raw text similarity comparisons.

Portnoy, et al., [6] also used the DARPA data to perform anomaly detection through clustering. His team performed normalization by transforming the data in the feature space from a collection of raw values for each measure to a collection of relative values with respect to the mean and standard deviation of each measure. A Euclidean distance metric was used as the basis for similarity between feature vectors. The results indicated the clustering performance was overly dependent on the unlabeled training set used to establish clusters. The authors also found that the false positive rate for identifying malicious packets in the KDD data was unacceptable for operational use. In contrast to [6], the work described here, works directly with the raw text data, and does not require a training set.

III. CLUSTERING

As discussed previously, each IDS provides a common mode of output: raw text intended for processing by a human. By applying advanced methods of text analysis, this raw text can be leveraged to provide an effective means of information fusion across IDSs.

Our approach is to cluster alerts based only on the text produced through the various IDSs. By comparing IDS outputs in terms of the raw text, the format is inherently normalized. However, the unstructured nature of the raw text makes organizing and comparing the data a challenge. We convert the raw text into a collection of terms and associated weights using the vector space model method shown in Fig. 1. In this method, the IDS alert text is converted into vectors, which provides a mathematical representation of the text. The vectors can then be compared to each other in order to organize (clustering) and group (categorizing) the various alerts.

The Vector Space Model (VSM) is a recognized approach to document content representation [7] in which the text in a document is characterized as a collection (vector) of unique terms/phrases and their corresponding normalized significance weight. Developing a VSM is a multi-step process.

The first step in the VSM process is to create a list of unique terms and phrases. This involves parsing the text and analyzing each term/phrase individually for uniqueness. The weight associated with each unique term/phrase is the degree of significance that the term or phrase has, relative to the other terms/phrases. For example, if the term “plan” is common across all or most documents, it will have a low significance, or weight value. Conversely, if “strategic” is a fairly unique term across the set of documents, it will have a higher weight value. The VSM for any document is the combination of the unique term/phrase and its associated weight as defined by a term weighting scheme.

In our approach, the term frequency-inverse document frequency (TF-IDF) is used as the term weighting scheme [7]. The equation for this scheme is shown in Fig. 1. In that equation, f_{ij} represents the frequency of occurrence of a term i in document j. In addition, N represents the total number of documents in the set of documents, and n represents the number of documents in which term i occurs. For a given frequency f_{ij}, the weight, W_{ij}, increases as the value of n decreases, and vice versa. Terms with a very high weight will have a high frequency f_{ij}, and a low value of n.

Once a vector representation is created for each IDS alert, similarity comparisons can be made. In our approach, a cosine similarity is used to compare two vectors A and B, as shown in (1).

$$\text{Similarity} = (A \cdot B) / (\|A\| \|B\|) \quad (1)$$

Similarity values ranges between 0 and 1, inclusive. A value of 1 means that vectors A and B are identical; while a value of 0 means that they are not alike at all.

Using the cosine similarity measure, clustering can now be performed. There are several forms of clustering. Our approach uses hierarchical clustering [8][9]. As the name implies, this technique creates a hierarchy of clusters where a cluster is composed of sub-clusters until the final sub-cluster contains the actual data elements. This technique enables an analyst to not only observe similarities between individual data elements but also similarities between clusters. This is particularly useful for IDS alerts. These alerts tend to be very similar but slightly different as shown in the example of Fig. 1. In that example, event 1 and 2 contain similar but slightly different text. If there were multiple events like these, then a hierarchical clustering would form a cluster of "Bluetooth Firmware Update". This cluster would then contain two sub-clusters with each one representing event 1 and 2, respectively.

In our clustering approach, the clustering begins with a cluster representing a 0 similarity threshold. From here, any new documents are compared to composite vectors representing the sub-clusters within the 0 similarity cluster. A composite is the summation of all vectors contained within the cluster. The sub-cluster that matches best is selected, and the new document is then compared to its sub-clusters. This continues until a sub-cluster representing the 1.0 similarity threshold is found. Sub-clusters are defined at 0.05 increments of similarity (e.g., 0, 0.05, 0.10, 0.15, etc.).

IV. VISUALIZATION

In order to visualize the cluster results, a graphical user interface was developed using Adobe® Flex [10] and Flare [11]. A sunburst diagram is used to represent the cluster tree with the root of the tree being the center of the diagram and each concentric ring representing a child node (i.e., sub-cluster) of the tree.

The sunburst diagram utilizes a radial space-filling (RSF) tree layout as implemented in the prefuse flare library [12]. This layout is from the radial graph-drawing family of visualization techniques, which have previously appeared in the visualization research literature. In particular, the approach is related to the semi-circular RSF hierarchies of Information Slices [13] and the focus+context interaction techniques for fully circular Starburst visualizations [14]. By adding support for brushing and interactive radial distortion, the InterRing [15] visualization expands RSF tree interaction techniques. More recently, Docuburst [16] combines RSF visualization and said interaction capabilities to represent text document content for drill down analysis.

Examples of the cluster visualization is shown in Fig. 2 through Fig. 7. These diagrams show the results of the cluster analysis, with the center being the point of 0 similarity and the leaves being more similar as you approach the outer edge. Each gray band encountered outside of the center circle represents a sub-cluster of alerts that have a similarity greater than the minimum threshold value of that sub-cluster. The gray bands may be toggle-selected by the operator to include or exclude the details of a sub-cluster from the visualization. The colored bands are actual IDS alerts that have been pre-categorized based on severity.

The intent of the visualization is to provide a frequency-based view of the IDS alerts– that is, show clusters of similar alerts across the spectrum of retrieved alerts from Splunk. This view allows an operator to see the distribution of alerts across categories and in terms of alert text. Alerts that are dissimilar to the rest of the alerts are easily and quickly highlighted.

In addition, the visualization provides analytic interaction with the underlying data. A "mouse over" of both alerts and alert clusters provide summary statistics. The operator may select different clusters to highlight or eliminate them from the view. Also, the operator may bring up the details on a specific cyber alert. Fig. 6 displays the same clustering of data shown in Fig. 5, but shows the result of an operator interacting with the cluster diagram to highlight the most significant and relevant data. This demonstrates how the cluster view allows for the rapid exploration of thousands of alerts to identify a significant subset for deeper inspection.

Fig. 2. shows the sunburst diagram along with the pre-defined categories that we used to color code the alerts in the cluster. This enables the operator to quickly identify clusters of significant alerts and of particular categories.

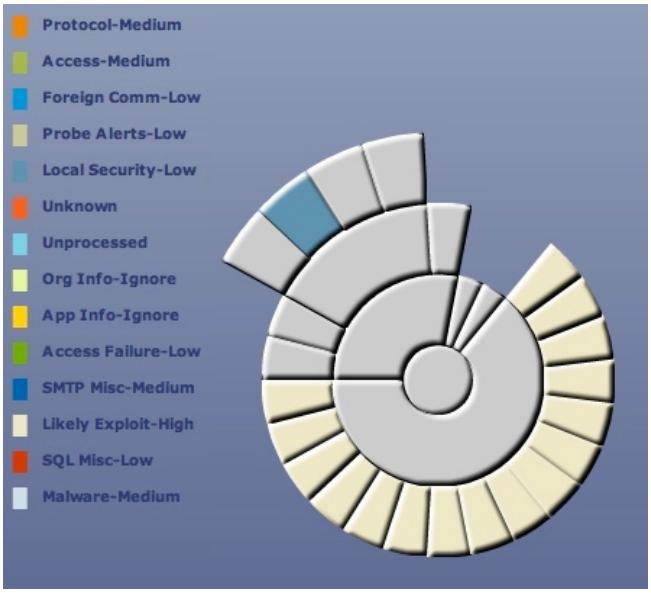

Figure 2. Sunburst diagram of cluster results shown with corresponding categories of alerts

Fig. 3 shows a the results of a "mouse over" one of the sub-clusters. Information about the particular sub-cluster is provided such as the similarity threshold that sub-cluster represents, the number of sub-clusters it contains, the number of alerts it contains, and the most significant terms in those alerts.

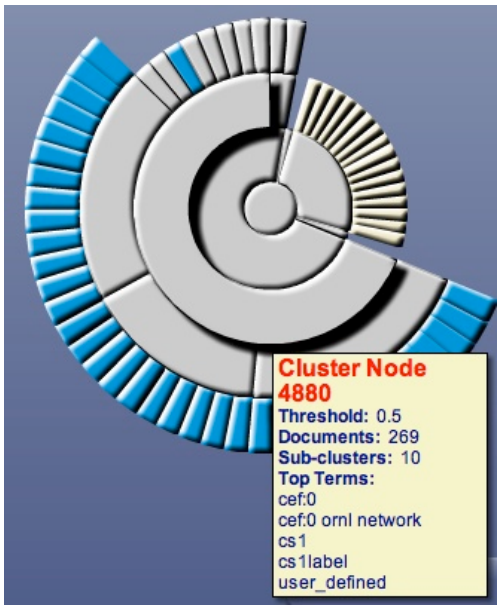

Figure 3. Cluster node information

Fig. 4 through Fig. 6 show different "mouse over" results for different alerts in the cluster. If available, these results will display what category the alert is associated, the source and destination IP addresses, as well as the top terms in the alert message. This allows the operator to quickly move through different alerts in order to gain an understanding of the overall data.

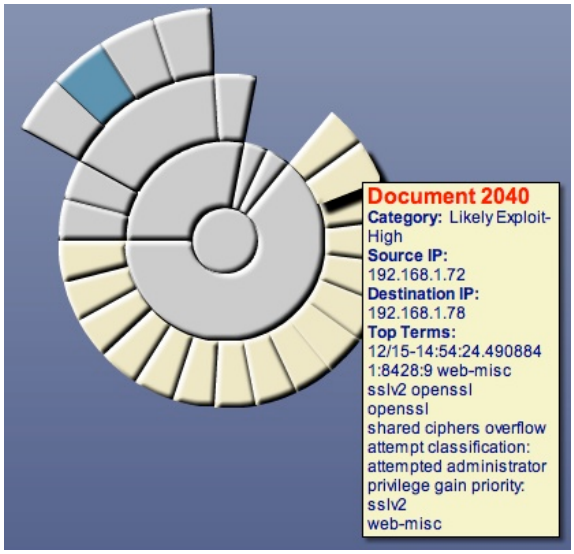

Figure 4. Alert information showing details of source and destination IP as well as top terms of the alert message

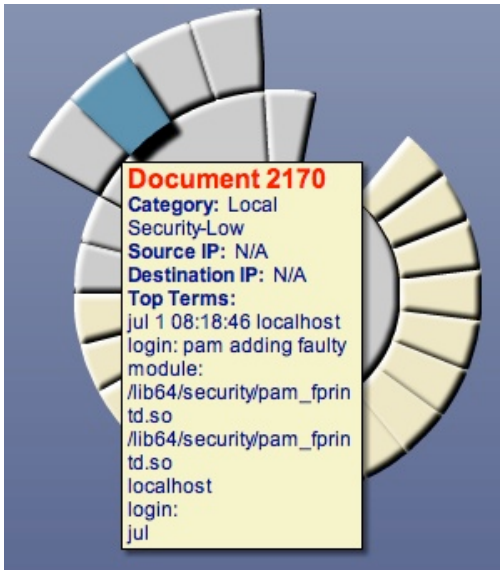

Figure 5. Alert information

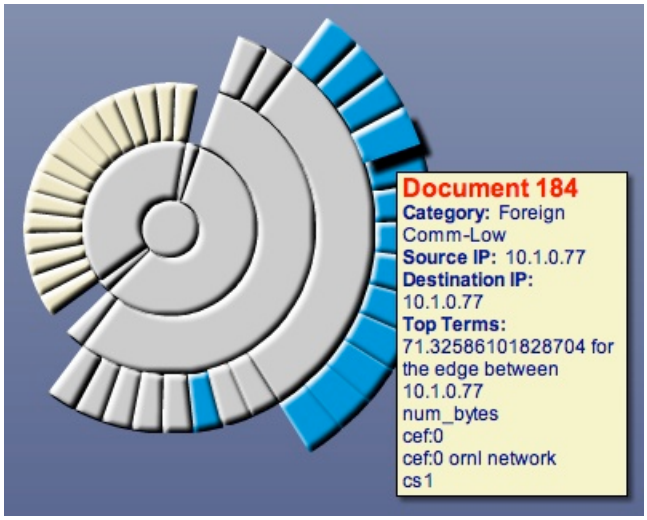

Figure 6. Alert information

Finally, Fig. 7 shows a clustering of alerts that failed to match any of the predefined categories. However, these alerts do cluster with other alerts that are categorized. This enables an operator to quickly discover any new alerts that are very similar to previously categorized alerts. The operator could then decide if these alerts represent a new category, or are part of the already defined category of the alerts that they are clustering.

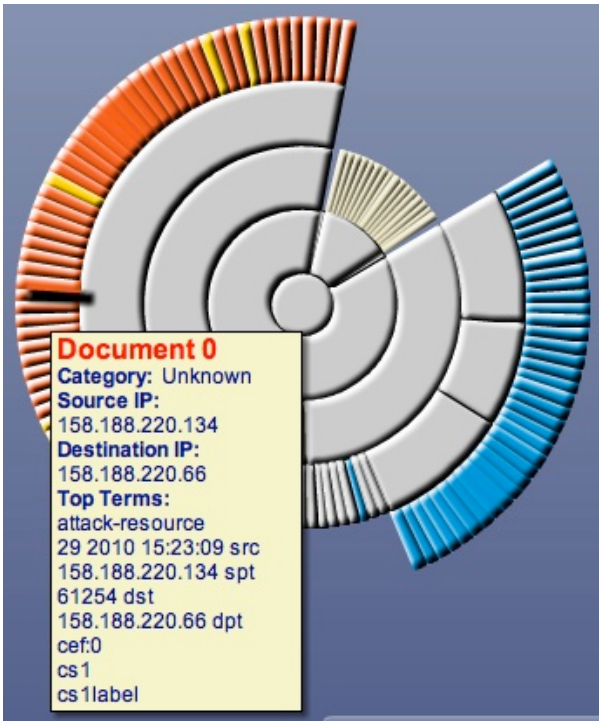

Figure 7. Alert message categorized as "unknown"

V. Future Work

The hierarchical clustering of alert text enables an analyst to quickly navigate thousands of IDS alert events in a matter of seconds. In this capacity, it is a powerful decision support tool that allows for rapid identification of the most significant alert events and diagnosis of probable attack vector, such that a timely response can be invoked.

Future work in applying hierarchical clustering to cyber security alert text will focus in two areas: alert fusion and IDS veracity. Both of these areas center on using the similarity in both the text and the timing of IDS alerts to determine whether independent IDS tools have alerted on the same event. Inferring whether a single attack vector causes multiple IDSs alerts leads to several advanced capabilities. For alert fusion, it allows for an alternate alerting stream to be produced that consolidates duplicate alerts and, leveraging a text classifier, filters out insignificant alerts. For IDS veracity, we plan to explore a means to score the detection power of IDSs based on the extent to which their alerts are corroborated with other IDS tools. These research directions will provide value in reducing the quantity of alerts to be processed, and provide organizations with insight into the reliability of the toolset that comprises their computer network defense.

Acknowledgment

Prepared by Oak Ridge National Laboratory, P.O. Box 2008, Oak Ridge, Tennessee 37831-6285; managed by UT-Battelle, LLC, for the U.S. Department of Energy under contract DE-AC05-00OR2225. This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 for the U.S. Department of Energy. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains non-exclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes.

References

[1] Splunk, current January 2011, http://www.splunk.com/

[2] Snort, current January 2011, http://www.snort.org/

[3] Bro, current January 2011, http://www.bro-ids.org/

[4] G.R. Hendry. "Applicability of clustering to cyber intrusion detection." Master's Thesis, Rochester Institute of Technology, Rochester, New York, August 2007.

[5] MIT Lincoln Laboratory. 1999 DARPA intrusion detection evaluation data set, 1999. Available online: http://www.ll.mit.edu/IST/ideval/data/1999/1999_data_index.html.

[6] L. Portnoy E. Eskin, and S. Stolfo. "Intrusion detection with unlabeled data using clustering." In *Proceedings of ACM CSS Workshop on Data Mining Applied to Security,* 2001.

[7] G. Salton, A. Wong, and C. S. Yang (1975), "A Vector Space Model for Automatic Indexing," Communications of the ACM, vol. 18, nr. 11, pages 613–620.

[8] King, B. Step-wise clustering procedures. *Journal of the American Statistical Association*, *69*, 86-101, 1967.

[9] Han, J. and Kamber, M. *Data Mining –Concepts and Techniques*. Morgan Kaufmann, 2001.

[10] Adobe® Flex, current January 2011, http://www.adobe.com/products/flex/

[11] Flare, current January 2011, http://flare.prefuse.org/

[12] Jeffrey Heer, Stuart K. Card, and James A. Landay. "prefuse: A toolkit for interactive information visualization." In *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 421-430, Apr. 2005, ACM Press.

[13] Keith Andrews and Helmut Heidegger. "Information slices: visualizing and exploring large hierarchies using cascading, semi-circular discs." In *Proc. of IEEE Symposium on Information Visualization, Late Breaking Hot Topic*, pp. 9-12, 1998, IEEE Computer Society.

[14] John Stasko and Eugene Zhang. "Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualization." In *Proc. of the IEEE Symposium on Information Visualization*. pp. 57-65, 2000, IEEE Computer Society.

[15] Jing Yang, Matthew O. Ward, Elke A. Rundensteiner. "InterRing: An interactive tool for visually navigating and manipulating hierarchical structures." In *Proc. of the IEEE Symposium on Information Visualization*. pp. 77-84, 2002, IEEE Computer Society.

[16] Christopher Collins, Sheelagh Carpendale, and Gerald Penn. "DocuBurst: visualizing document content using language structure." In *Proc. of Eurographics/IEEE-VGTC Symposium on Visualization (EuroVis 09)*, pp. 1039-1046, June 2009.