Beehive: Large-Scale Log Analysis for Detecting Suspicious Activity in Enterprise Networks

Sowmya Jayaram Iyer
jayarami@purdue.edu
Date: 09/17/2021

Summary

The paper presents two key contributions: (1) A novel automated approach, Beehive, which uses clustering to analyze very large volumes of disparate or dirty log data collected, by a variety of network devices, in large enterprises to detect malicious activity such as malware infections. (2) The notion of using behavioral analysis to flag suspected security incidents such as enterprise policy violation which were undetectable for the existing state- of-the-art security tools and personnel. The author gives a clear picture of the threats that can arise in different scenarios within an enterprise. Also, a detailed picture of the various challenges faced in "big data" at various parts of the paper was given and attacked by presenting a detailed step-by-step approach to reduce the noise and inconsistencies in data. Though the paper convinces the need for an automated system, it fails to give a detailed comparison of methods and the presentation of results can be more general and varied. The way static and dynamic IP addresses were segregated and represented differently from a pool of IP addresses was nice. The chosen feature vector appeared to address the spectrum of the domain well. Overall, with suggested improvements along the lines of giving detailed comparisons of results, this paper can make a good case on why Beehive is best suitable for detection of suspicious activity in all enterprise networks. Comments and suggestion are listed in the next section.

Detailed Comments

The novel contribution of this paper is presenting an automated analytic approach to analyze logs stored in SIEM systems produced by an enterprise which were previously done by state-of-art security tools or personnel. The critical components in this method are: (1) the data-reduction algorithms and strategies to focus on security-relevant information from the massive number of events logged for timely detection. (2) identifying meaningful security incidents in the face of a significant semantic gap between the logs collected by the SIEM system and the information that security analysts require to identify suspicious host behavior. (3) evaluating its effectiveness accurately in spite of the lack of ground truth.

The pre-processing phase (Section 3.1) of the system has clearly addressed all the disparities in the large raw data and is followed by various techniques like timestamp normalization, determination of statically and dynamically assigned IP addresses of hosts, construction of bindings between hosts and IP addresses, attribution of the logged event to a specific host, which makes the data more understandable and readable. Though the pre-processing works for this data, it appears like the paper has only taken problems local to this particular enterprise and makes us question whether it can be generalized. Moreover, the paper uses data only from one enterprise which makes this approach very policy-specific and though the amount of data is huge, it is only taken for a period of two weeks. Also, while determining the dedicated hosts, the author mentions using authentication logs collected over a period of three months to build an accurate history of user activity and gives no evidence to support the mentioned accuracy. Since the Beehive focuses on monitoring the behavior of dedicated hosts, a detailed explanation for the chosen criteria for determining a host as "dedicated" is important.

Section 3.2 gives a neat description of the feature vector and how they have been grouped into four feature types based on new and unpopular destinations contacted by the host, features related to the host's software configuration, features related to the company policy, and features based on traffic volume and timing. In host-based features, using Levenshtein distance to compare UA strings and using the number of "new" UA strings from the host to detect software updates or new software installation is a good approach. Figure 4 supports the choice for threshold value for the connection spikes and domain spikes well, however, author fails to mention what CDF stands for in the figure.

Section 3.3 is not in par with the clarity seen in the previous sections. The author fails multiple times to explain his reasons for the method chose. For example, it is understandable that the lack of ground truth calls for unsupervised learning but why K-means is the best among the various clustering techniques is not mentioned. "However, the features may be related or dependent upon one another; e.g., a domain spike also triggers a connection spike." (Page 6) does not give a clear picture of how the data is actually co-related. A simple covariance matrix to show the need for PCA and support that claim is required. Additionally, "the top m principal components are selected, permitting projection of the original vectors down to dimensionality m. In Bee- hive, we select the top m components that capture at least 95% of the data variance." statement seemed abrupt without any background on why and how this choice was arrived at. Overall, this paper lacks in depth while explaining the technical aspect of the model.

The way Section 4 and Section 5 have been organized gives a quantitative clarity to the claim of Beehive to have surpassed the performance of existing state-of-the art security tools, demonstrating Beehive's ability to identify previously unknown anomalous behaviors. In section 5, author claims that this work is "the first exploration of the challenges of "big data" security analytics at the scale of real-world enterprise log data". Though a contrast between the previous methods and this method has been presented, why the existing automated methods would fail for large-scale enterprise data has not been clearly explained.

Recommendations

The paper makes a good effort in trying to use unsupervised learning for analyzing and categorizing not only malicious attacks but also policy violations. All the challenges mentioned in the paper were clearly addressed and resolved using Big Data and Clustering. This paper has proved its efficiency and feasibility against anti-virus softwares, state-of-the art security tools and even enterprise SOCs.

A comparison of the performance of various clustering techniques against K-means or elaborating more on why K-means was chosen as the most suitable model can better convince the readers. Providing more evidence for this model's efficiency in handling large-scale data of longer periods consisting more unknown IP destinations can make the model seem less specific to EMC enterprise alone. Also, more evidence that the model can handle more diverse threats and the flexibility of policies within enterprises is required. Also, mentioning a few drawbacks that could arise while using previous works on large-scale data can help give this paper more technical depth and better highlight its contribution.