Visualizing the Impact of Cyberattacks on Web-Based Transactions on Large-Scale Data and Knowledge-Based Systems

Mani Manavalan[1*], Nur Mohammad Ali Chisty[2]

[1]Technology Architect, Cognizant Technology Solutions, Teaneck, New Jersey, **USA**
[2]Additional Superintendent of Police, Police Headquarters, Dhaka, Bangladesh Police, **BANGLADESH**

[*]Corresponding Contact:
Email: manimanavalan47@gmail.com

ABSTRACT

Manual approaches rely on the abilities and knowledge of individual human administrators to detect, analyze, and interpret attacks. Intrusion Detection Systems (IDS) are systems that can automatically detect and warn the appropriate persons when an attack occurs. Despite the fact that individual attacks can be useful, they are frequently insufficient for understanding the entire attacking process, as well as the attackers' talents and objectives. The attacking stage is usually merely a component of a larger infiltration process, during which attackers gather information and set up the proper conditions before launching an attack, after which they clear log records in order to conceal their footprints and disappear. In today's assault scenarios, the pre-definition of cause-and-effect links between events is required, which is a tough and time-consuming task that takes considerable effort. Our technique for creating attack scenarios is based on the linking nature of web pages, and it does not require the pre-definition of cause and effect links, as demonstrated in previous work. Constructed situations are displayed in spatial and temporal coordinate systems to make viewing and analyzing them more convenient. In addition, we develop a prototype implementation of the concept, which we utilize to test a number of assault scenario scenarios.

Key words:
Cyberattacks, Coordinate Systems, Intrusion Detection Systems, Web-Based Transactions, Security Administrators

9/30/2019 **Source of Support:** None, **NoConflict of Interest:** Declared

INTRODUCTION

Despite the fact that traditional Intrusion Detection Systems (IDSs) may identify individual assaults and notify the proper parties in a short period of time, they usually lack the ability to synthesize and present connected attacks (or related events) to human users in a user-

friendly fashion. We believe that monitoring individual attacks does not aid in the comprehension of the overall attack process by responsible individuals. This comprehensive picture includes not only the attack phase, but also the data collecting and preparation operations that take place prior to the assault phase, as well as the exploitation and identity masking processes that take place after the attack phase (Bynagari, 2018; Bynagari & Fadziso, 2018). As a result, security administrators require a tool that provides them with a comprehensive view of the situation. Because of this technology, security administrators are able to monitor the entire incursion process and gain insight into the attackers' objectives, plans, and talents. This knowledge is critical not only for launching a counterattack as early as possible, but also for effectively planning future defense measures.

Two challenges must be overcome in order to forward with the development of the tool indicated above. The first step is figuring out how to group together occurrences that are associated with a given assault type. The second challenge is figuring out how to present these occurrences in a natural and understandable way. The first difficulty is referred to as "attack scenario construction," while the second problem is referred to as "attack scenario visualization" in this work. Although some research has employed information visualization methodologies to depict assault scenarios, the vast majority of studies have concentrated on portraying each attack individually. To put it another way, despite the fact that the attacks are displayed at the same time and on the same screen, there is usually no information about the linkages to the assaults displayed on the screen (and other relevant events). Furthermore, to the best of our knowledge, no documented work on attack scenario visualization for web applications has been made available. Therefore, we believe that the contributions of this work will provide some early viewpoints and conceptions on which future research may be based, which will be useful for future research. For the purpose of making our work more persuasive, we also develop a prototype implementation that serves to demonstrate our concept and allows us to experiment with a variety of assault situations. The results of the study indicate that our approach is capable of revealing information that would be difficult to discover using a conventional web application intrusion detection system.

Extensive research has been done on the topic of designing attack scenarios by network security experts in the past. Based on cause-and-effect correlations, previous studies have recommended strategies for developing assault scenarios. This study tested those methods. As a result, they specify some requirements (known as pre-conditions) that must be met in order for a specific attack to be valid, and they presume that if the attack is successful, it will also result in the fulfillment of some further requirements (known as consequences). These efforts resulted in attack scenarios, which are a collection of independent but connected events that occur in a specific order. In order to use these systems, managers must first define the cause and effect relationships between events in advance, which is a complex and time-consuming process. This article, on the other hand, is concerned with the security of web applications, and as a result, we are able to exploit the linking nature of web pages in order to provide an attack scenario generation technique that does not require the pre-definition of cause and effect linkages by a human user.

This paper attempted to address the following issues:

- To provide a technique for displaying attack scenarios to security administrators that blends information visualization with user involvement.
- Develop a mechanism for generating cyberattack scenarios that take advantage of website linking in order to demonstrate your understanding.

LITERATURE REVIEW

Information system attacks can be classified into three groups based on the level of human engagement required by the security administrator: automated, manual, and semi-automatic. Automatic approaches require no human interaction at all. Automatic solutions do not necessitate the involvement of human operators on a regular basis, and their outputs are typically straightforward, such as the generation of warnings when something unusual is detected. To determine whether or not warnings are real, as well as their sources and repercussions, it is up to security managers to conduct thorough investigations. Snort (Rosech, 1999) and Bro (Rosech, 1999) are two popular tools in this category (Paxson, 1999). Because it involves the least amount of human intervention, the automatic technique has the greatest number of advantages over manual procedures. Administrators, on the other hand, are unable to distinguish between the forest and the trees because the method's output is straightforward.

Those strategies that employ a manual approach, on the other hand, rely on the talents and experience of particular human administrators in order to recognize, analyze, and interpret offensive actions. In addition to being extremely limited, this method needs a significant amount of human labor. Furthermore, because the effectiveness of this method is based on the performance of the administrators, it is not considered a major research topic in the field.

Those techniques that employ a manual approach, on the other hand, rely on the talents and experience of particular human administrators in order to identify, analyze, and interpret threats and attacks. In addition to being extremely limited, this method needs a significant amount of human labor. Furthermore, because the effectiveness of this method is based on the performance of the administrators, it is not considered a major research topic in the field. Semi-automatic solutions function in the middle of the data processing spectrum, conducting some initial data processing, presenting the results to humans (typically in visual form), eliciting their reactions, and then repeating the entire data processing cycle. Here, the most important point to note is that the human user is considered to be a significant component of this strategy. Because administrators must interact with systems in order to receive the results they seek, this technique assists administrators in better understanding the security events that occur in their environments. As an alternative to individual warnings, the findings from automatic intrusion detection systems are sometimes used as part of the input to these semiautomatic systems and processed to provide a high-level representation of the overall security state (Livnat et al., 2005; Nkhoma and Dang, 2013).

Some studies in the field of network security propose the use of correlation algorithms on intrusion alerts in order to aggregate them in a meaningful manner in order to give managers a more comprehensive view of the security condition of their networks. One of the primary benefits of alerts correlation is that it reduces the amount of time spent examining alerts by condensing a large number of separate warnings into a smaller number of connected alerts, allowing administrators to evaluate these alerts at a more in-depth level as a result. Another advantage is that it provides administrators with a more understandable big picture by grouping warnings together in a logical way. According to Debar and Wespi (2001), some associations between alerts are generally established in advance and based on these relationships, alerts are grouped together in a logical manner. In a separate study, Ning et al. utilize the terms "prerequisites" and "consequences" to bind alerts together (Ning et al., 2002). When we talk about "prerequisites" and "consequences," we're talking about the conditions that must be satisfied in order for an attack to succeed,

and we're talking about the different results of a successful assault when we talk about "consequences." Visual directed graphs are used by the authors to present the correlation data in a more understandable manner than previously. According to the correlation techniques employed by Debar and Wespi (2001) and Ning et al. (2001), security administrators must establish the criteria that will be utilized to aggregate alerts in advance (2002). It is difficult and time-consuming to fulfill the demand for medium- to large-scale information system implementations. Furthermore, when changes occur in the monitored system, it is necessary to update the rules in place.

Clickstream analysis is a study field that is similar to ours in that it visualizes customers' online actions in a similar way to what we do. The primary difference between our work and that of the rest of this study field is that ours is concerned with security occurrences, whereas the other's is concerned with regular user behaviors. For webmasters of e-commerce sites, clickstream analysis is now a helpful tool to have on hand. In order to understand more about their website visitors, merchants use cookies to collect information such as where they came from, how they used the site, which pages they exited on, which sites they decided to buy goods from, and so on. Clickstream is often presented in visual style and allows users to communicate with the website's administrator. Lee et al. (2015) propose two visualization approaches for clickstream analysis: the parallel coordinate graph and the starfield graph. For example, the parallel coordinate is used to represent the sequence of user actions on a website such as searching for something on a website, clicking on something, purchasing something, and so on. It also shows the number of people that have dropped out as a result of each action taken. The performance of products is shown by a starfield graph, which shows how many times they have been viewed and how many times they have been clicked after they have been viewed. According to Kawamoto and Itoh (2010), another research aims to integrate and display users' access patterns with existing website link structures in order to obtain a better knowledge of users' behaviors. Google Analytics and Web trends are two examples of commerce solutions that may be found on the internet.

This component collects information on users' actions and behaviors from a number of sources, including the Internet. For each request, information such as the time of access, the URI of the accessible page, the IP address, the user agent, and the query string should be logged. The data it collects from a web application IDS and an Apache web server access log file is all that is included in the current prototype version. We intend to expand the number of input sources available in future editions. Immediately following the gathering of data, a preprocessing step is carried out in order to ensure that all of the data is standardized and structured in the same way throughout. As an added bonus, because the obtained data is scattered over several databases, this preprocessing component gathers it all into a single database for subsequent extraction and processing (Manavalan & Donepudi, 2016).

For the purpose of creating attack scenarios, this component brings together data from a number of sources. The two input sources in this prototype are HTTP requests to an Apache server and warnings given by a web application intrusion detection system. Generally, we presume that a security administrator is more interested in alerts than in regular requests. It produces two lists: a pre-events list and a post-events list, both of which are based on a certain alert and some user input criteria, respectively. Specifically, the pre-events list comprises events that occur before the given alert (for example, HTTP requests and IDS alarms), whereas the post-events list contains events that occur after the provided alert (for

example, HTTP requests). When events are divided into pre-events and post-events lists, administrators can have a better understanding of the attack's preparation phase (which contains pre-events data) and consequence phase (which contains post-events data) for a specific attack than they could otherwise. Sessions reconstruction (Spiliopoulou et al., 2003), which can be found in the online usage mining study area, is similar to the compilation of pre-events and post-events lists (which can be found in the online usage mining study area). Both of them are attempting to build a list of related events from a single event. Neither of them has succeeded. In contrast to the session reconstruction task, there is a key difference between our work and that of the session reconstruction task: in the session reconstruction task, users' actions are believed to be normal, but in our work, they are thought to be aberrant.

This assumption leads to the obvious conclusion that users in our scenario would go to great lengths to disguise their traces, making the building of attack scenarios more challenging than session reconstruction. Even in the case of the problem of session reconstruction, it has been noted that there are certain difficulties in overcoming the obstacles that arise (Srivastava et al., 2000). In order to overcome these difficulties, we do not use pre-defined rules and parameter values to build attack scenarios, but rather delegate this responsibility to security managers. This has two advantages: first, the administrator is the one who is most familiar with her system, and second, she will get valuable experience by tweaking these settings herself.

In this component, system administrators can interact with the system as well as modify the way scenario development and visualization are carried out. A second panel containing input elements can be used by administrators to modify the way the scenario building process is carried out. A common example is the ability of administrators to modify a threshold that governs the relatedness of two events in terms of time (Manavalan & Bynagari, 2015; Manavalan, 2016; Manavalan & Donepudi, 2016; Amin & Manavalan, 2017). On the primary visual interface, we immediately incorporate mouse actions such as hovering, selecting, and so on to allow administrators to interact with the scenario visualization. If an administrator clicks on a page, he or she can obtain extra particular information about it that isn't provided by default, such as the number of times it has been accessed by different users. How many warnings are generated on that particular page?

RESULTS AND DISCUSSION

A manual attack by a human is launched.

A human security specialist has been asked to analyze and attack the target web application on a computerized network. In this case, we utilize different values for the parameters used in the development of the attack scenario since it is apparent that the time between requests would be larger when the requests are done by a human user than when they are not. This implies that our technique is rather effective at distinguishing between attacks carried out by automated programs and attacks carried out by human users. Although we believe it is capable of distinguishing between attacks by professional hackers and assaults by beginner hackers, we have certain doubts about it.

Designing Visualizations

The major aim of the visualization component is to ensure that attack scenarios are correctly displayed to security administrators. When an assault (for example, an alarm issued by a

web application intrusion detection system) is selected, the events connected with that attack are captured. When a legitimate attack takes place, it is usually followed by a number of other dubious occurrences. When it comes to false assaults, on the other hand, they are more likely to occur spontaneously. We think (but do not have proof) that it is feasible to distinguish between fictitious and legitimate attacks by considering attacks and associated events in the same context. The primary goal of our visualization in this project is to represent how an assault scenario evolves through time (when the sequence of events takes place) and space (where the attack happens) (where the chain of events accesses to). Understanding what happens before and after an attack is launched (events in the preparation phase) by visualizing attack scenarios in space and time coordinate systems may be extremely beneficial to security administrators (events in consequence phase). As a result, administrators may get valuable information about an attacker's attacking process, such as what she does to prepare for an attack, what she does if the attack is successful, and how she cleans up after herself. Space-time visualization systems include two major visual areas, which we use to view assault situations. The time-oriented coordinate systems area and the space-oriented coordinate systems area are the two key visual areas we use to visualize assault scenarios (Fadziso & Manavalan, 2017; Manavalan, 2018).

System of Coordinates Oriented in Time

Whenever an event occurs, it is represented as a circle, with the center specified by the timestamp of the event's occurrence. Events that occur near together are combined into a single larger event in order to reduce the amount of clutter. In an event circle, the size of the circle is determined by the number of elementary events (i.e., events that are recorded in a single entry of the web server log or in a single IDS record) that are included inside it. The intensity of the influence that each event circle has on the web application is indicated by the color of the circle surrounding the event. In our application, the severity is determined by the number of alerts that have been generated in the event circle. This coordinate system enables security administrators to follow the evolution of an attack scenario over time using a variety of different sensors.

System of Coordinates Oriented in Space

When events access a common URI, this coordinate system is utilized to arrange them into a hierarchical structure. When comparing different pages, the aim is to illustrate how many events occur on each page and which pages attract the most attention from attackers in contrast to other sites. The size of each page is defined by the number of events that occur on it, and the color of each page is determined by the severity of the impact it has on the web application, in a manner similar to the time oriented coordinate system (Bynagari, 2016; Bynagari, 2017). On top of that, we classify pages according to their levels, which are defined as the minimum number of links required to reach the selected alert page from that specific page for pre-events pages, and the minimum number of links required to reach the selected alert page from that specific page for post-events pages.

CONCLUSION

As part of this research, we designed and implemented a mechanism for creating web application scenarios and visualizing them to aid security administrators in better understanding infiltration processes on their systems. To the contrary of previous work on the development of attack scenarios, our study takes advantage of the time restrictions between related web requests, in addition to the space constraints (linking relationships)

between pages on web applications, to build attack scenarios for web applications. This means that it does not need the manual specification of cause and effect links in advance, as is the case with certain alternative methods. Administrators must first grasp the process by which intrusions occur on their systems before they can appreciate the consequences of those incursions. To the contrary of previous work on the development of attack scenarios, our study takes advantage of the time restrictions between related web requests, in addition to the space constraints (linking relationships) between pages on web applications, to build attack scenarios for web applications. This means that it does not need the manual specification of cause and effect links in advance, as is the case with certain alternative methods. Especially in big online systems, this feature aids security administrators in significantly lowering their workload. It is the ability to display not just individual assaults or warnings, but also associated chains of events in a holistic way that is the most valuable feature of our method for security administrators. Event coordinates are assigned in this manner, based on the time (date) and the location (URL) at which the event takes place.

Following a specified alert in the attack scenario under design, we classify events into prevents (events that take place before the event that causes the selected alert) and post-events (events that occur after the event that causes the selected alert) (events happen after the event that raises the selected alert). Preparatory processes (pre-events) and cleaning procedures (post-events) will be more effective, we feel, if security managers have a clear understanding of what constitutes each (post events). The findings of our studies indicate the usefulness of our methodology in helping security administrators better comprehend web application threats. When we look at the visualization findings, we may acquire some insight into the attack tactics used by different tools and persons, which is tough to achieve through traditional approaches. Accordingly, we believe our proposed approach is a valuable addition to existing web application intrusion detection systems in that it offers security administrators with realistic attack scenarios based on individual assaults identified by other IDSs. As a result of learning from various assault scenarios, administrators may have a greater knowledge of the entire attack process. It is consequently possible to utilize this information both now and in the future to not just counter-attack, but also to develop defense strategies.

It is true that this attempt does not solve all of the constraints. In the first place, there are the tests in which we use automated tools and ask an expert to generate HTTP requests in order to attack a test web site. Each one of them has an air of fakery about it, and as such, it makes them appear untrustworthy in comparison with real-life circumstances. In order to circumvent this restriction, we want to put up a honey pot to attract genuine attackers from the Internet, which will give more realistic data for future research. Another issue is that certain feedback and assessments from real individuals who may use our prototype in their everyday job, such as security administrators, are lacking from our prototype. Their evaluation of our work will include its usability and usefulness. It is possible to use alternative ways to assess our work, but doing so will need a significant amount of additional effort (Plaisant, 2004). This work may be expanded in a second way by directly integrating other data into the visualization space, such as alert kind, severity, HTTP response code, and so on. This would be a significant step forward (currently they are displayed on demand via user interactions, i.e. mouse clicking). The amount of time spent studying data may be reduced, allowing administrators to gain more time and get fresh perspectives on their data. The use of animation to repeat attack scenarios (together with suitable time scaling methods to minimize or increase the amount of time spent viewing) is the third approach that might be used to further develop this study. It is our hope that by

adding animation into this tool, we would be able to provide more accurate information about assault scenarios to users. In addition, we feel that animating makes it more pleasant for administrators to collaborate with.

REFERENCES

Amin, R., & Manavalan, M. (2017). Modeling Long Short-Term Memory in Quantum Optical Experiments. *International Journal of Reciprocal Symmetry and Physical Sciences*, *4*, 6–13. Retrieved from https://upright.pub/index.php/ijrsps/article/view/48

Bynagari, N. B. (2016). Industrial Application of Internet of Things. *Asia Pacific Journal of Energy and Environment*, *3*(2), 75-82. https://doi.org/10.18034/apjee.v3i2.576

Bynagari, N. B. (2017). Prediction of Human Population Responses to Toxic Compounds by a Collaborative Competition. *Asian Journal of Humanity, Art and Literature*, *4*(2), 147-156. https://doi.org/10.18034/ajhal.v4i2.577

Bynagari, N. B. (2018). On the ChEMBL Platform, a Large-scale Evaluation of Machine Learning Algorithms for Drug Target Prediction. *Asian Journal of Applied Science and Engineering*, *7*, 53–64. Retrieved from https://upright.pub/index.php/ajase/article/view/31

Bynagari, N. B., & Fadziso, T. (2018). Theoretical Approaches of Machine Learning to Schizophrenia. *Engineering International*, *6*(2), 155-168. https://doi.org/10.18034/ei.v6i2.568

Debar, H. and Wespi, A. (2001). Aggregation and Correlation of Intrusion-Detection Alerts. In: Lee W., Mé L., Wespi A. (eds) Recent Advances in Intrusion Detection. RAID 2001. Lecture Notes in Computer Science, vol 2212. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-45474-8_6

Fadziso, T., & Manavalan, M. (2017). Identical by Descent (IBD): Investigation of the Genetic Ties between Africans, Denisovans, and Neandertals. *Asian Journal of Humanity, Art and Literature*, *4*(2), 157-170. https://doi.org/10.18034/ajhal.v4i2.582

Johnson, M. H., Dziurawiec, S., Ellis, H., & Morton, J. (1991). Newborns' preferential tracking of face-like stimuli and its subsequent decline. *Cognition, 40*(1-2), 1–19. https://doi.org/10.1016/0010-0277(91)90045-6

Kawamoto, M., and Itoh, T. (2010). A Visualization Technique for Access Patterns and Link Structures of Web Sites. 2010 14th International Conference Information Visualisation, 11-16. https://doi.org/10.1109/IV.2010.11

Lee, B., Riche, N. H., Isenberg, P. and Carpendale, S. (2015). More than Telling a Story: Transforming Data into Visually Shared Stories. *IEEE Computer Graphics and Applications, 35*(5), 84-90. https://doi.org/10.1109/MCG.2015.99

Livnat, Y. Agutter, J., Moon, S., Erbacher, R. F. and Foresti, S. (2005). A visualization paradigm for network intrusion detection. Proceedings from the Sixth Annual IEEE SMC Information Assurance Workshop, 92-99. https://doi.org/10.1109/IAW.2005.1495939

Manavalan, M. (2016). Biclustering of Omics Data using Rectified Factor Networks. *International Journal of Reciprocal Symmetry and Physical Sciences*, *3*, 1–10. Retrieved from https://upright.pub/index.php/ijrsps/article/view/40

Manavalan, M. (2018). Do Internals of Neural Networks Make Sense in the Context of Hydrology?. *Asian Journal of Applied Science and Engineering*, *7*, 75–84. Retrieved from https://upright.pub/index.php/ajase/article/view/41

Manavalan, M., & Bynagari, N. B. (2015). A Single Long Short-Term Memory Network can Predict Rainfall-Runoff at Multiple Timescales. *International Journal of Reciprocal Symmetry and Physical Sciences*, *2*, 1–7. Retrieved from https://upright.pub/index.php/ijrsps/article/view/39

Manavalan, M., & Donepudi, P. K. (2016). A Sample-based Criterion for Unsupervised Learning of Complex Models beyond Maximum Likelihood and Density Estimation. *ABC Journal of Advanced Research*, *5*(2), 123-130. https://doi.org/10.18034/abcjar.v5i2.581

Manavalan, M., & Donepudi, P. K. (2016). A Sample-based Criterion for Unsupervised Learning of Complex Models beyond Maximum Likelihood and Density Estimation. *ABC Journal of Advanced Research*, *5*(2), 123-130. https://doi.org/10.18034/abcjar.v5i2.581

Ning, P., Cui, Y., and Reeves, D. S. (2002). Constructing attack scenarios through correlation of intrusion alerts. In Proceedings of the 9th ACM conference on Computer and communications security (CCS '02). Association for Computing Machinery, New York, NY, USA, 245–254. https://doi.org/10.1145/586110.586144

Nkhoma, M. Z. and Dang, D. P. T. (2013). Contributing Factors of Cloud Computing Adoption: a Technology-Organisation-Environment Framework Approach. International Journal of Information System and Engineering, 1(1), 30-41. https://doi.org/10.24924/ijise/2013.04/v1.iss1/30.41

Paxson, V. (1999). Bro: A System for Detecting Network Intruders in Real-Time. *Computer Networks, 31*(23–24), 2435–2463. https://www.icir.org/vern/papers/bro-CN99.pdf

Plaisant, C. (2004). The challenge of information visualization evaluation. In Proceedings of the working conference on Advanced visual interfaces (AVI '04). Association for Computing Machinery, New York, NY, USA, 109–116. https://doi.org/10.1145/989863.989880

Roesch, M. (1999) Snort: Lightweight Intrusion Detection for Networks. LISA, 99, 229-238.

Spiliopoulou, M., Mobasher, B., Berendt, B., Nakagawa, M. (2003). A Framework for the Evaluation of Session Reconstruction Heuristics in Web-Usage Analysis. *INFORMS Journal on Computing 15*(2), 171-190. https://doi.org/10.1287/ijoc.15.2.171.14445

--0--

Archive Link:

https://abc.us.org/ojs/index.php/ei/issue/archive