Visualizing PHPIDS Log Files for Better Understanding of Web Server Attacks

Mansour Alsaleh
Computer Research Institute
King Abdulaziz City for
Science and Technology
Riyadh, KSA
maalsaleh@kacst.edu.sa

Abdullah Alqahtani
Computer Science
Department
King Saud University
Riyadh, KSA
alqahtani.a@live.com

Abdulrahman Alarifi
Computer Research Institute
King Abdulaziz City for
Science and Technology
Riyadh, KSA
aarifi@kacst.edu.sa

AbdulMalik Al-Salman
Computer Science Department
King Saud University
Riyadh, KSA
salman@ksu.edu.sa

ABSTRACT

The prevalence and severity of application-layer vulnerabilities increase dramatically their corresponding attacks. In this paper, we present an extension to PHPIDS, an open source intrusion detection and prevention system for PHP-based web applications, to visualize its security log. The proposed extension analyzes PHPIDS logs, correlates these logs with the corresponding web server logs, and plots the security-related events. We use a set of tightly coupled visual representations of HTTP server requests containing known and suspicious malicious content, to provide system administrators and security analysts with fine-grained visual-based querying capabilities. We present multiple case studies to demonstrate the ability of our PHPIDS visualization extension to support security analysts with analytic reasoning and decision making in response to ongoing web server attacks. Experimenting the proposed PHPIDS visualization extension on real-world datasets shows promise for providing complementary information for effective *situational awareness*.

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General—*Security and protection*; C.2.3 [**Computer-Communication Networks**]: Network Operations—*Network monitoring*; K.6.5 [**Management Of Computing And Information Systems**]: Security and Protection—*Invasive software*

Keywords

Security data visualization, log visualization, intrusion detection systems, network monitoring, web server attacks

VizSec '13 October 14 2013, Atlanta, GA, USA

1. INTRODUCTION

While the current trend of the criminal activity is a cash economy (e.g., spam), targeted attacks are in the rise serving a variety of purposes such as those attacks launched by organized groups or state-sponsored attacks to disrupt global critical infrastructures. Internet attacks are becoming increasingly hard to detect with the increasing complexity in Internet traffic dynamics and heterogeneity. New security vulnerabilities that enable a remote adversary to compromise a machine are discovered on a daily basis. Relative to other layers in the Internet protocol suite, the application layer is susceptible to a larger vector of security vulnerabilities due to complexity, variety, and rapid development of commodity applications. Application layer security is becoming a growing area of concern for security researchers. While the remedy seems in the prevention of security flaws and vulnerabilities during the software development process (i.e., the design, development, testing, and deployment of software), eliminating all security vulnerabilities during software development process seems infeasible.

Essential network services with Internet-addressable IP addresses (e.g., web, mail, and DNS services) represent attractive channels for adversaries who wish to reach other local hosts surreptitiously across network perimeter [21]. Therefore, securing services such as web servers is a critical security practice. Furthermore, web servers must also support server-side scripting (e.g., PHP and ASP), which in turn requires further security hardening measures. However, infection techniques that require user interaction such as pull-based infection techniques (e.g., drive-by downloads) and push-based techniques (e.g., opening malicious email attachments) are often not applicable to servers.

Intrusion Detection and Prevention Systems (IDPS) monitor network identify, log, and attempt to actively block malicious activity in the network. PHPIDS is an open source intrusion detection and prevention system for PHP-based web applications. By examining all input variables in the POST and GET request methods and in the HTTP cookies, PHPIDS has the capability to detect most conventional web attacks such as cross-site scripting (XSS), SQL injection, remote file execution, local file inclusion, Denial of Service

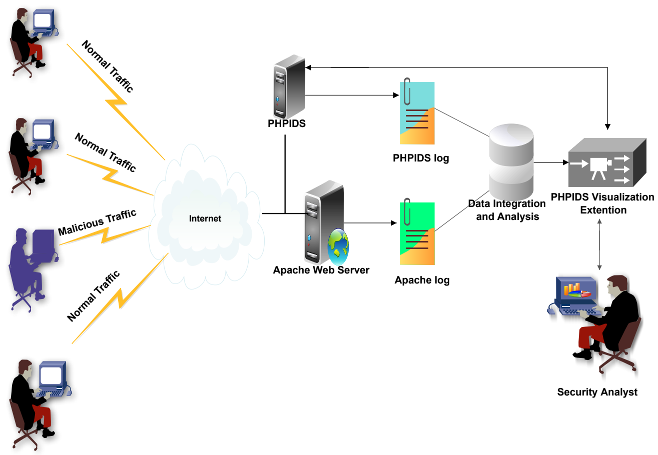


Figure 1: System Design.

(DoS), header injection, and directory traversal. Also, PHPIDS is able to detect unknown attack patterns with the *Centrifuge* component which does in depth string analysis and measurement of the input [22]. In fact, more than 80% of all websites use PHP as a server-side programming language, and most popular content management systems (e.g., WordPress, Joomla, and Drupal) are PHP-based [28].

Manual inspection of security logs is inevitable when processing IDPS alarms. This is particularly helpful for security analysts either for taking proactive measures or for forensic or root cause analysis. However, manual inspection of security logs is a time consuming and error-prone process, especially for mining large-scale attacks or stealthy activities. Security analysts often seek to answer two important questions: did adversaries exploit known or zero-day vulnerabilities to compromise local hosts in the network in question; and if yes, what operations did they perform in these local hosts? [29].

Visual representations of data carry much larger, richer information than text-based data. Data visualization techniques enable security analysts to browse large amounts of data, help in perceiving trends patterns, and provide a wealth of insights into the security-related events being inspected, unveiling relationships that are hard to be discovered otherwise. The usage of security data visualization is motivated by the fact that most security defense systems are mainly based on text-based logs for recording security-related events, which are difficult to analyze and correlate.

We argue that visualization of such security activities provides visibility of a class of cyber attacks that are otherwise difficult to detect in isolation. Accordingly, we present an extension to PHPIDS, an open source intrusion detection and prevention system for PHP-based web applications, to visualize its security log.

This work makes the following contributions:

1. Analysis and correlation of PHPIDS and web server logs. The log files of both PHPIDS and the Apache web server are analyzed and correlated to further enhance the security analysts' understanding of web security threats.

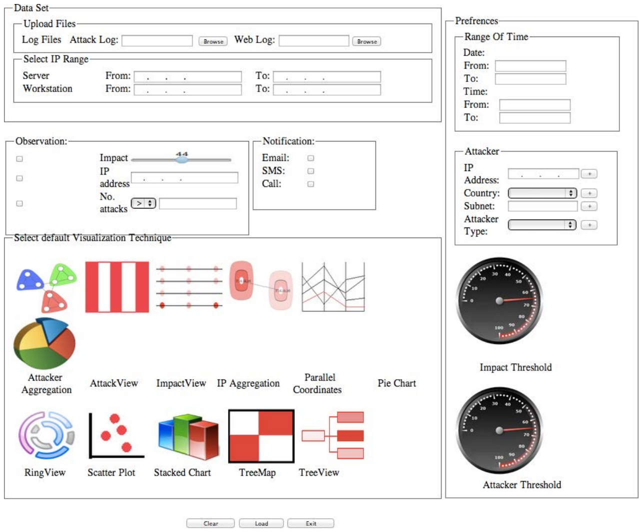


Figure 2: Control console of PHPIDS visualization extension.

2. PHPIDS visualization extension. We explore various data visualizations (including novel visualizations) and study their suitability in representing PHPIDS security logs. We then design an extension to PHPIDS to visualize its security log using a set of tightly coupled visual representations of HTTP server requests containing known and suspicious malicious content, to provide system administrators and security analysts with fine-grained visual-based querying capabilities.

3. Experimentation on real-world datasets. We demonstrate the utility of our proposed PHPIDS extension using real-world PHPIDS and Apache log files. We also present multiple case studies to demonstrate the ability of our PHPIDS visualization extension to support security analysts with analytic reasoning and decision making in response to the ongoing web server attacks.

2. DATASETS AND METHODOLOGY

We installed PHPIDS on a web server of an operational blog hosting site (startup site). The log files of both PHPIDS and Apache were gathered over the period of April 10 to May 10, 2013 (32 days). 148 web attacks were detected by PHPIDS from 58 distinct IP addresses.

We use Prefuse [8], a Java-based toolkit for building rich interactive data visualizations, to build visualizations for our PHPIDS extension. The extension web console takes advantage of Prefuse integration into Java Swing applications and web applets.

Based on the structure of PHPIDS and Apache log files, we first explore various data visualizations to find useful ones that seem to convey insights from security-related data that are hard to be extracted from text-based data. We then design an extension to PHPIDS to visualize its security log using a set of tightly coupled visualizations of web server requests, focusing on malicious content.

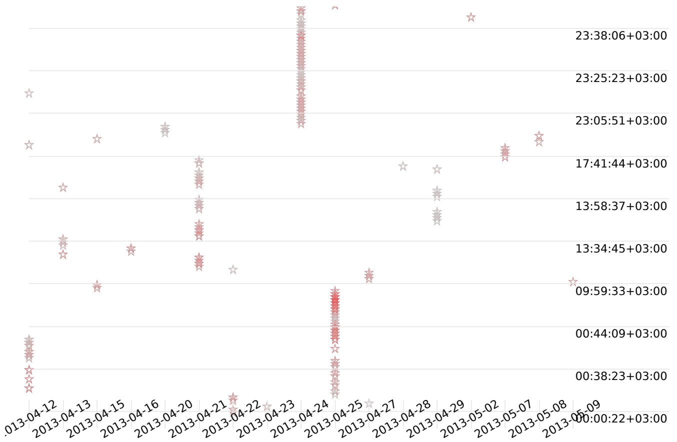


Figure 3: Scatter plot of web attacks based on date, time, and the type of attack.

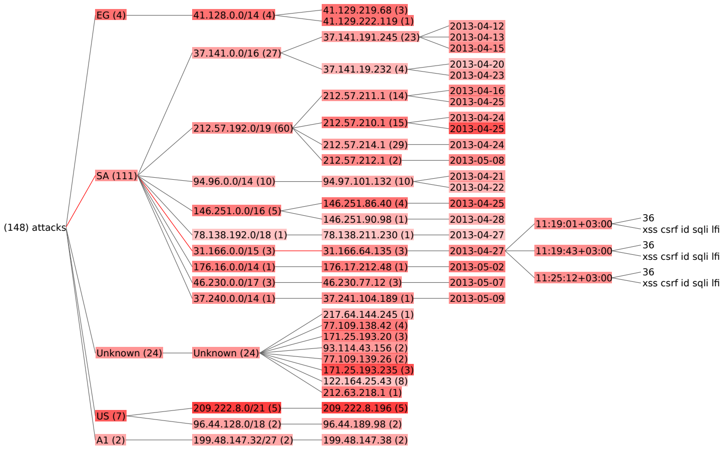


Figure 4: TreeView visualization of web attacks based on geolocation, subnet, and IP address.

3. PHPIDS VISUALIZATION EXTENSION

PHPIDS is based on a large set of regular expressions of known web attacks. It can also detect some unknown malicious patterns by applying a set of behavior heuristics. Once a web attack is detected, PHPIDS logs the attack and can be configured to drop the connection before it is processed by the web server. PHPIDS gives a weight for each detected attack that gives an indication of the attack severity (i.e., a higher weight means a higher risk). This helps security analysts to decide what sort of action should follow the hacking attempt.

We built a visualization extension on top of PHPIDS that visualizes the detected malicious web activity in real-time by passively integrating and analyzing PHPIDS and Apache log files (see Figure 1). The extension provides security analysts with a web interface of a visual-based querying system of malicious web activity for a given time period and within a specific IP address space (see Figure 2).

In the following, we illustrate our PHPIDS extension visualizations. This includes: (1) selected ones from known visual representations (e.g., scatter plot and parallel coordinate) that we found suitable for representing PHPIDS security logs; and (2) new visual representations designed specifically for providing an intuitive mean for detailed analysis of the detected web attacks.

3.1 Scatter Plot

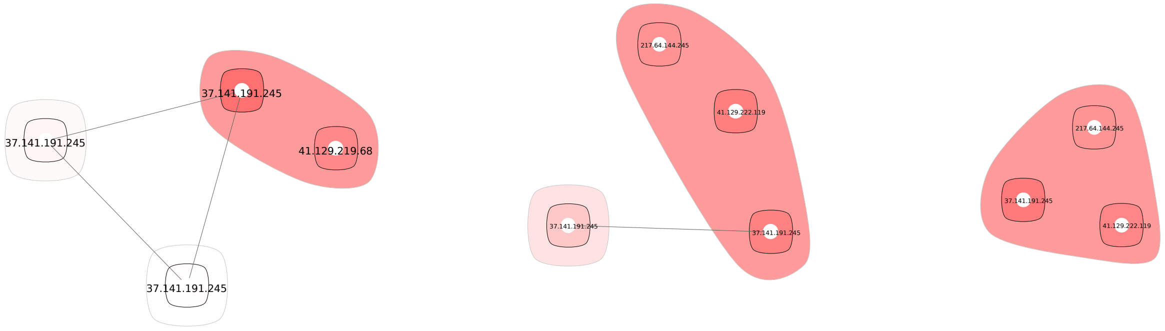
Scatter plot helps in visualizing large dataset with less overlapped values. Figure 3 is a basic scatter plot showing the date (plotted on the x axis) and the time (plotted on the y axis) of the web attacks selected by the analyst. The security analyst can set the impact threshold so that only those attacks with a higher or equal impact are displayed. When the analyst hovers the mouse over one of the plotted attacks, the attack detail is displayed on a label on the top of graph. Time period in a drop-down menu and the impact threshold in a slide are provided to enable the analyst to tweak the graph. Also, there are two search panels, one for searching by IP address and the other to search by attack type. When two or more attacks take place at the same time such that they overlap, the analyst can adjust the time menu to widen or zoom in until the two attacks are visible.

3.2 TreeView

When there are too many attacks in the log file, either over a short or a long period of time, it is sometimes hard for the analyst to identify similar IP addresses or those in the same subnet. Also, categorizing source IP addresses based on their geolocation provides additional information that could help in understanding relationships among different attack scenarios. A tree view visualization is plotted in Figure 4 showing attackers geolocation, subnets, and source IP addresses. Red color auto-scaling has been used to indicate attack's total impact for each node. The tree expand/collapse feature makes the graph scalable so that it presents large amount of data, while it enables the analyst to focus on attack sources of interest. By default, the graph shows four levels that give an overview of the attacks distribution. The subnet information is gathered by querying the database provided by *IpAddressLocation* [9].

3.3 IP Address Aggregator

In order to understand the distribution of the remote IP addresses the adversaries use to conduct web attacks, we design a new visualization in which these IP addresses are grouped into clusters. Each cluster contains IP addresses of the remote hosts used to conduct web attacks within n hours time frame. Inside each cluster, the IP addresses are also grouped based on the subnet (subnet information is gathered by querying the database provided by *IpAddressLocation* [9]). If a remote appears in more than one time cluster, its nodes will be connected with edges. A scale of red color is given to each cluster node and each subnet node (within a cluster) based on web attacks total impacts (i.e., the darker the red color the higher the impact). While the analysts can zoom in and out, drag, and position each node, to make the best use of the available display space, the auto zoom feature automatically scales the visualization. Moreover, by hovering the mouse over one of the plotted IP addresses, the web attack detail is displayed on a label on the top of the graph. Hovering the mouse over a time cluster displays the information of the time cluster including start/end date, start/end



(a) Time cluster of one hour period.

(b) Time cluster of two hour period.

(c) Time cluster of three hour period.

Figure 5: IP address aggregation graph in which each node contains adversaries IP addresses (i.e., IP addresses of the remote hosts launching the web attacks within a time period).

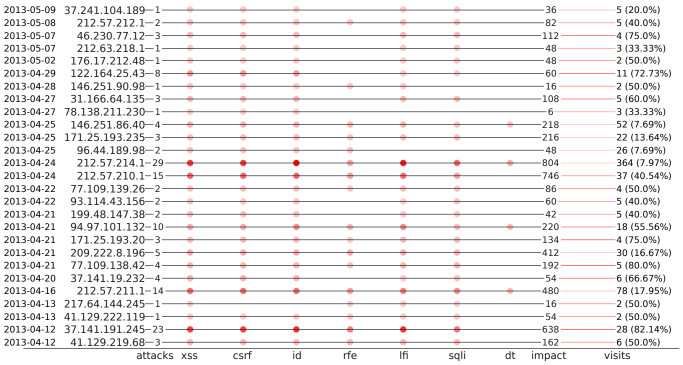


Figure 6: Attack frequency view.

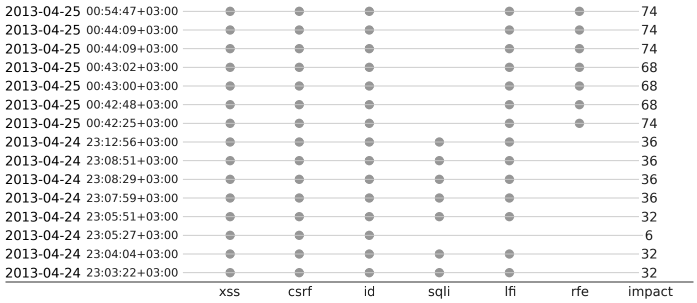


Figure 7: A zoom-in on one IP address in the attack frequency view shows the details of 29 web attacks.

time, time interval, and total impacts of all the IP addresses in the time cluster in question. Figures 5(a), 5(b), and 5(c) show the web attacks (during April 13th, 2013) for clusters of one-, two-, and three-hours time frames, respectively.

3.4 Attack Frequency View

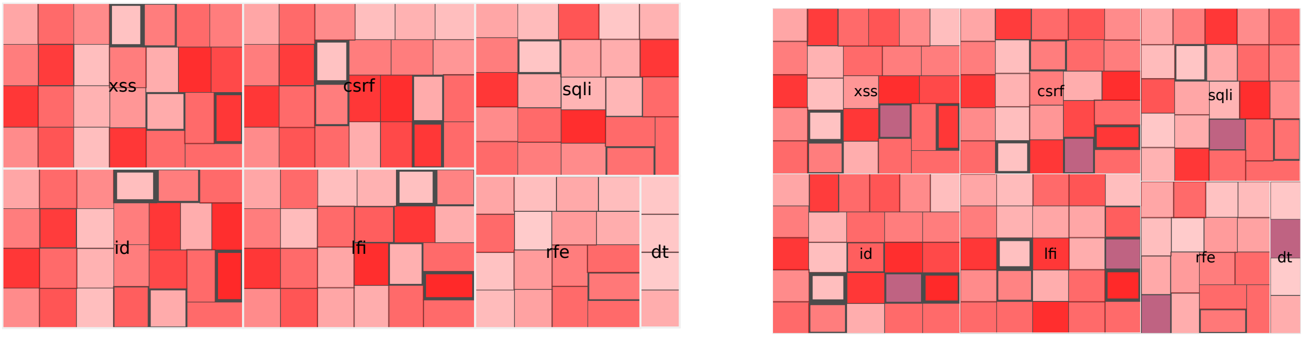
In order to help security analysts to easily find out frequent adversaries, Figure 6 shows the following information: (1) the date where the adversary IP address was firstly seen; (2) the adversary IP address; (3) number of attacks sent by the IP address; (4) the frequency of each of the general attack types (the darker the red color the higher the frequency); (5) the total impact of all the attacks sent by the IP address; (6) the number of HTTP requests the IP address made (extracted from the web server logs); and (7) the ratio of HTTP requests that have malicious contents to all the HTTP requests the IP address made.

If the number of IP addresses is large, the visualization can be scrolled down and up. Also, providing zoom feature makes this type of visualization scalable for visualizing large number of attacks and large number of IP addresses. When the analyst clicks on an IP address, the details of all the attacks corresponding to the IP address are shown on another visualization as in Figure 7. In this visualization, every horizontal line represents one attack (in this particular example, 15 attacks are launched by the IP address).

3.5 Tree Map

Tree map is a known visualization technique appropriate for visualizing large datasets in a hierarchical view. Rather than the conventional way of using tree map visualization for displaying the hierarchical pattern of IP addresses, we use tree map here to visually represent the hierarchy of the attacks types. That is, the tree map categorizes attacks according to their types such that every category displays IP addresses that have launched the corresponding type of attack in smaller squares. Figure 8(a) is an example of using tree map (over 30 days of the dataset). Scaling in red color is used to represent the percentage of attacks launched by the corresponding IP address (relative to the total number of the attack type in question). The border size of the IP address boxes (the smallest boxes in this visualization) is used to represent the number of web attacks launched by each IP address (i.e., thicker border indicates high number of web attacks).

By hovering the mouse over an IP address box, the attack information is displayed on the bottom of the graph. The attack information includes, among others, the IP address, the number of launched attacks by the IP address, and the total number of HTTP requests made by the IP address (extracted from the web server logs). A search box at the bottom of the graph enables the analyst to search by IP address and find all attack types launched by this IP



(a) Tree map visualization.

(b) Searching for an IP address in a tree map.

Figure 8: Tree map visualization for plotting IP addresses distribution based on the attack type.

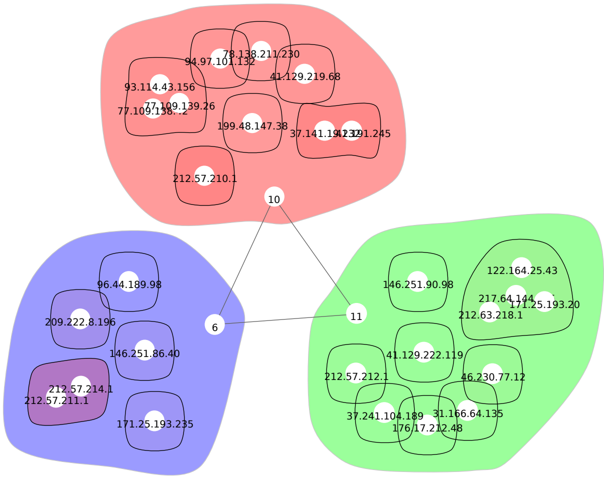


Figure 9: Attacker aggregation visualization presents three types of attackers in different clusters, malicious in the red cluster, suspicious in the green cluster and single malicious request in the blue cluster.

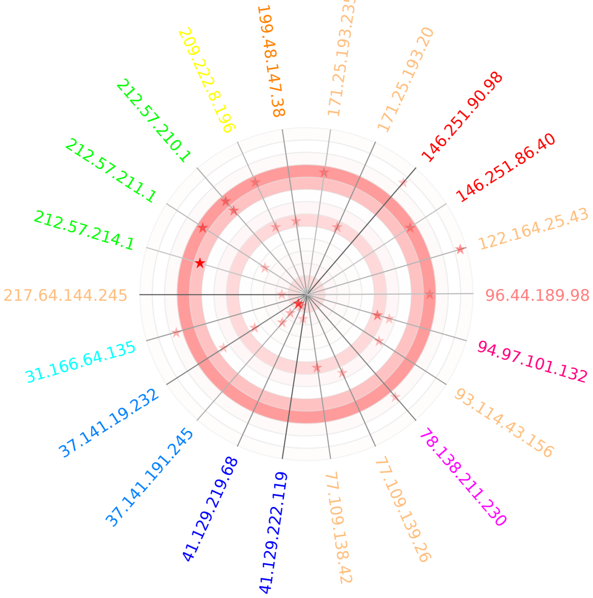


Figure 10: Ring visualization.

address. The searched IP address is colored in purple as in Figure 8(b). Also, number of matches is displayed in a label beside the search box.

3.6 Attacker Aggregation

In this new visualization, we divide IP addresses launching web attacks into three categories: (1) malicious IP addresses: those that most of their HTTP requests contain malicious content; (2) suspicious IP addresses: those that only some of their HTTP requests contain malicious content; and (3) single malicious request IP addresses: those that only send one HTTP request containing malicious content. Figure 9 shows these three types of IP addresses (category (1) in red color, category (2) in green, and category (3) in blue). Each category is divided into subnets so that all the IP addresses of one subnet are in the subnet cluster. Categories are connected with extra nodes that hold the number of IP addresses in each category.

When the mouse is moved over any IP address, further details of the IP address is displayed on a label at the bottom of the graph (e.g., the total number of attacks originated by the IP address and the total number of HTTP requests). Likewise, when the mouse is moved over a subnet, the subnet details are displayed (e.g., subnet country). Similar to the IP address aggregation visualization, zoom in/out, and drag/re-position features are provided which are crucial for the scalability of this visualization. However, it is important to note that this visualization might get cluttered in case of too many IP addresses, requiring the analyst to always zoom in to be able to see the details.

3.7 Ring View

In targeted attacks, the web server might observe several attempts of one or more attacks from the same remote host. On the other hand, for newly discovered web vulnerabilities or in case of attacks requiring a campaign of bots (e.g., dis-

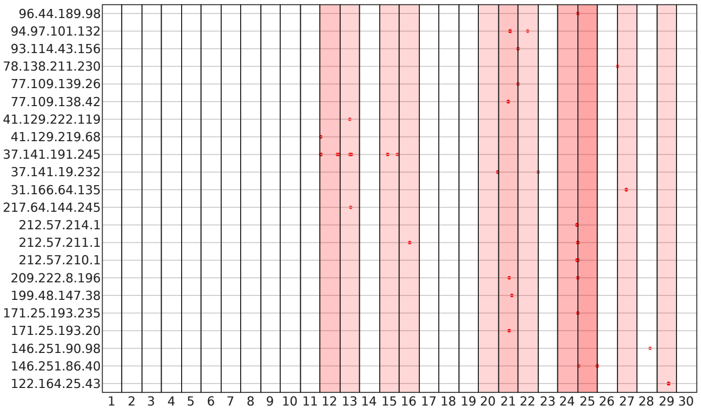


Figure 11: Bar view visualizing frequency of web attacks during 30 days.

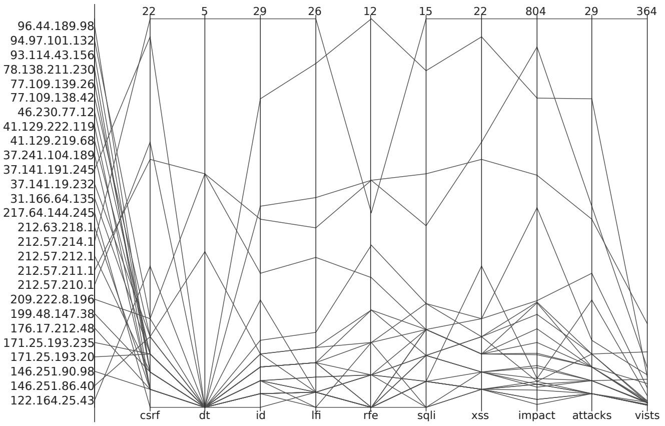


Figure 12: Parallel Coordinates showing types, impacts, and number of attacks originates from inbound traffic for a period of 30 days.

tributed denial of service attacks), multiple instances of the same attack are launched simultaneously by many remotes. In Figure 10, we leverage a ring view that displays all attacks during a particular range of time. In this visualization, each ring represents a day (17 days in total), whereas rings for days with no attacks are not plotted. The color of the ring is in a red scale such that the darker the red color the higher the total impact of the attacks launched during that day. Every remote IP addresses launching web attacks is displayed with a line intersecting all the rings. IP addresses in the same subnet are given the same color.

The star icon in the intersection indicates one or more attacks launched by the remote during the ring day whereas the red color of the star is scaled based on the total attacks impact during that day. The gray scale of an IP address line indicates the ratio of the remote HTTP requests containing malicious contents to the total number of the remote HTTP requests (a darker gray indicates a higher ratio). Hovering the mouse over the attack(s) star displays detailed information about the time and the type of the attacks. To display all the attacks launched by an IP address in a day, the analyst can click on the star to launch the visualization in Figure 7.

3.8 Bar View

In the bar view visualization (see Figure 11), two coordinates are used instead of the rings in the ring view visualization, one for plotting IP addresses activity (y axis) and one for displaying activities during days (x axis). To eliminate the need to browse other visualizations to find attacks detail, the web attacks within each day are plotted using multiple points instead of using one star as in the ring view. The position of an attack's point on the graph is relative to the day on which the attack was launched. Similar to the previous visualizations, hovering the mouse over the different objects in the visualization displays more detail and the red color scale indicates severity of the attack impact.

3.9 Parallel Coordinates

Parallel coordinates is a known useful multi-value visualization technique. For the purpose of comparison with the presented visualizations above, and in order to give security analysts more choices to visualize PHPIDS log files, we use parallel coordinates (see Figure 12) to show types, impacts, and number of attacks originates from inbound traffic. It also displays the number of HTTP requests the remote made and the ratio of HTTP requests that have malicious contents to all the HTTP requests of the remote (extracted from the corresponding web server logs).

4. LIMITATIONS AND FURTHER DISCUSSION

Among the several visualizations that are implemented in the proposed PHPIDS visualization extension, it is important to note that while some visualizations represent the same log data, their different visual representations convey different perceptions of the corresponding malicious events. In addition, this gives security analysts several options to visualize the log data.

A common problem in data visualization is cluttered plots. While Prefuse provides automatic clutter reduction techniques, the existence of many IP addresses (e.g., due to visualizing logs of a long period of time) might actually result in some cluttered visualizations. The zoom-in and search options available in most of the presented visualizations helps remedy this problem.

We believe further web server log data could be utilized to better understand some web attacks. Moreover, gathering and correlating security data from other security-related logs (e.g., firewall logs) could also help to understand the big picture. Thus, an interesting avenue for future work would be to integrate data from other security logs into the PHPIDS visualization extension.

We plan as a future work to experiment and evaluate the PHPIDS visualization extension on large logs. We also plan to gather feedback from administrators in the field.

5. RELATED WORK

Security visualization is both an art and a science that provides network admins and security analysts with a graphical representation of security-related data to provide a deeper insight into emerging threats. Security visualization can range from simple and familiar bar graphs to advanced and interactive graphs that help in seeing patterns and detecting malicious activities.

Since security visualization is user-centric, several literature have studied the definition of effective visualization and how to measure the effectiveness [33, 31, 6, 17, 30, 2]. However, security visualization is still a fairly new research field that needs to be explored [5, 15].

Security visualization has different potential data sources such as raw packets, netflow records, switch logs, router logs, server logs, firewall logs, intrusion detection system (IDS) logs, operating system log, and application logs [25, 15]. Incorporating different data sources can improve visualization tools and provide analysts with more information, enabling better understanding of the underlying security events [14, 7]. IDSs are one of the main sources for security-related data that provide network admins with useful information such as security policy violations, infections, information leakages, configuration errors, and unauthorized accesses. Log data from IDSs have been used in many visualization tools.

For the network layer, Koike et al. presented a new visualization tool for network-based IDS named SnortView which uses different visualization techniques for different data such as overlayed statistical information and source destination matrix [11].

Several tools were built to visualize network-related alerts that are generated from Snort [24, 20, 27, 13, 32]. Abdullah et al. used port-based overview of network activity to provide a better representation for detecting malicious activities [1]. NVisionIP uses Argus NetFlow to collect network traffic and visualize activities over different source and destination ports [12, 23]. Statistical profiling and visualization techniques have also been explored for host-based IDSs to filter false alarms [4].

Statistical analysis have been used intensively with visualization to extract existing patterns in the data. Min et al. have utilized correlation between intrusion detection alerts to built a better visualization tool [16]. The visualization shows all alerts as dots of different colors on time and location axis. This way, the correlation of alerts can be plotted as lines on the graphs which helps in analyzing and tracing alerts. A more advanced statistical analysis such as Bayesian classifier has been also used with visualization to help administrator in understanding IDS results [3].

Some three-dimensional visualizations for IDS log files has also been proposed [19]. Nyarko et al. proposal uses colored 3D graphs to evaluate impact of attacks [18]. Tudumi is another 3D visualization system used to monitor and audit system logs [26]. Khor et al. presented a visualization tool for web application administrators based on three authorization layers: public access layer, registered user layer, and administrator layer [10]. To the best of our knowledge, this paper is the first to visualize security logs of web-based IDS such as PHPIDS.

6. CONCLUSION

Security log visualization is an effective way for security analysts to protect IT infrastructure against security threats and vulnerabilities. In this paper, we have presented a visualization extension to PHPIDS that correlates data from different log files to provide better security situational awareness. Both the log files of PHPIDS and the Apache web server were analyzed and correlated to further enhance the security analysts' understanding of web security threats.

We demonstrated the utility of our proposed PHPIDS extension using real-world PHPIDS and Apache log files. We presented multiple case studies to demonstrate the ability of our PHPIDS visualization extension to support security analysts with analytic reasoning and decision making in response to web attacks.

We hope this work will encourage further research in this important area to help security analysts in making correct decisions and we hope this work will motivate developers of security tools to include visualization features.

7. ACKNOWLEDGEMENTS

We thank the anonymous reviewers for their comments which helped improve this paper to its present form. This work was supported in part by KACST.

8. REFERENCES

[1] K. Abdullah, C. Lee, G. Conti, and J. A. Copeland. Visualizing network data for intrusion detection. In *Information Assurance Workshop, 2005. IAW'05. Proceedings from the Sixth Annual IEEE SMC*, pages 100–108. IEEE, 2005.

[2] Z. Alshaikh, A. Alarifi, and M. Alsaleh. Christopher Alexander's fifteen properties: Toward developing evaluation metrics for security visualizations. In *Proceedings of the IEEE Intelligence and Security Informatics Conference*. IEEE Press, 2013.

[3] S. Axelsson and D. Sands. Combining a Bayesian classifier with visualization: Understanding the IDS. *Understanding Intrusion Detection Through Visualization*, pages 69–87, 2006.

[4] J. B. Colombe and G. Stephens. Statistical profiling and visualization for detection of malicious insider attacks on computer networks. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, pages 138–142. ACM, 2004.

[5] G. Conti. *Security Data Visualization*. No Starch Press, San Francisco, CA, USA, 2007.

[6] M. Dastani. The role of visual perception in data visualization. *Journal of Visual Languages and Computing*, 13:601–622, 2002.

[7] R. F. Erbacher, K. Christensen, and A. Sundberg. Designing visualization capabilities for IDS challenges. In *IEEE Workshop on Visualization for Computer Security (VizSec'05)*, pages 121–127. IEEE, 2005.

[8] J. Heer, S. K. Card, and J. A. Landay. Prefuse: a toolkit for interactive information visualization. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 421–430. ACM, 2005.

[9] IpAddressLocation. `http://www.ipaddresslocation.org/`. Accessed: May 2013.

[10] E. C. K.C. Khor, S.K. Lieong. Efficient information visualization for intrusion detection in web application. In *International Conference on Computer Graphics, Imaging and Visualization (CGIV 2005)*, pages 98–102, Beijing, China, 2005.

[11] H. Koike and K. Ohno. SnortView: visualization system of snort logs. In *Proceedings of the ACM*

workshop on Visualization and data mining for computer security, pages 143–147. ACM, 2004.

[12] K. Lakkaraju, W. Yurcik, and A. J. Lee. Nvisionip: netflow visualizations of system state for security situational awareness. In *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, pages 65–72. ACM, 2004.

[13] C. P. Lee, J. Trost, N. Gibbs, R. Beyah, and J. A. Copeland. Visual firewall: real-time network security monitor. In *IEEE Workshop on Visualization for Computer Security (VizSec'05)*, pages 129–136. IEEE, 2005.

[14] Y. Livnat, J. Agutter, S. Moon, R. F. Erbacher, and S. Foresti. A visualization paradigm for network intrusion detection. In *Information Assurance Workshop, 2005. IAW'05. Proceedings from the Sixth Annual IEEE SMC*, pages 92–99. IEEE, 2005.

[15] R. Marty. *Applied security visualization.* Addison-Wesley, 2009.

[16] B.-G. Min, J. Kim, and S.-J. Hong. Visualization of intrusion detection alerts with alert correlation. In *Second International Conference on Applied Cryptography and Network Security*, 2004.

[17] L. Nowell, R. Schulman, and D. Hix. Graphical encoding for information visualization: An empirical study. In *Proceedings of the IEEE Symposium on Information Visualization (InfoVis'02)*, pages 43–, Washington, DC, USA, 2002. IEEE Computer Society.

[18] K. Nyarko, T. Capers, C. Scott, and K. Ladeji-Osias. Network intrusion visualization with NIVA, an intrusion detection visual analyzer with haptic integration. In *Proceedings of the 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (HAPTICS'02)*, pages 277–284. IEEE, 2002.

[19] A. Oline and D. Reiners. Exploring three-dimensional visualization for intrusion detection. In *IEEE Workshop on Visualization for Computer Security, 2005 (VizSec'05)*, pages 113–120, 2005.

[20] Y. J. Park and J. C. Park. Web application intrusion detection system for input validation attack. In *Third International Conference on Convergence and Hybrid Information Technology (ICCIT'08)*, volume 2, pages 498–504. IEEE, 2008.

[21] V. Paxson, M. Christodorescu, M. J. J. Rao, R. Sailer, D. Schales, M. P. Stoecklin, K. T. W. Venema, and N. Weaver. Practical Comprehensive Bounds on Surreptitious Communication Over DNS. In *Proceedings of the in the Proceedings of the 22nd USENIX Security Symposium.* USENIX, 2013.

[22] PHPIDS. `http://www.phpids.org/`. Accessed: April 2013.

[23] L. QoSient. Argus NetFlow. Accessed: July 2013. `http://qosient.com/argus/argusnetflow.shtml`.

[24] A. E.-D. Riad, I. Elhenawy, A. Hassan, and N. Awadallah. Data visualization technique framework for intrusion detection. *International Journal of Computer Science Issues (IJCSI)*, 8(5), 2011.

[25] H. Shiravi, A. Shiravi, and A. A. Ghorbani. A survey of visualization systems for network security. *IEEE Transactions on Visualization and Computer Graphics*, 18(8):1313–1329, 2012.

[26] T. Takada and H. Koike. Tudumi: Information visualization system for monitoring and auditing computer logs. In *Proceedings of the Sixth International Conference on Information Visualisation*, pages 570–576. IEEE, 2002.

[27] R. R. UR. Intrusion detection with SNORT: advanced IDS techniques using SNORT, apache, mySQL, PHP and acid. Prentice Hall Publishers, 2003.

[28] Usage statistics and market share of PHP for websites. `http://w3techs.com/technologies/details/pl-php/all/all`. Accessed: June 2013.

[29] A. Vasudevan, N. Qu, and A. Perrig. Xtrec: Secure real-time execution trace recording on commodity platforms. In *Proceedings of the 44th Hawaii International Conference on System Sciences (HICSS)*, pages 1–10. IEEE, 2011.

[30] I. Vekiri. What is the value of graphical displays in learning? *Educational Psychology Review*, 14(3):261–312, 2002.

[31] M. Wattenberg and D. Fisher. Analyzing perceptual organization in information graphics. *Information Visualization Journal*, 3(2):123–133, June 2004.

[32] Y. Zhao, F. Zhou, and X. Fan. A real-time visualization framework for ids alerts. In *Proceedings of the 5th International Symposium on Visual Information Communication and Interaction*, pages 11–17. ACM, 2012.

[33] Y. Zhu. Measuring effective data visualization. In *Proceedings of the 3rd international conference on Advances in visual computing - Volume Part II*, ISVC'07, pages 652–661, Berlin, Heidelberg, 2007. Springer-Verlag.