DOCTORAL PROGRAM IN ENGINEERING SCIENCES AT ITESO

A NOVEL USER NETWORK PROFILE BASED ON HOST NETWORK TRAFFIC

Alvaro I. Parres-Peredo, Hugo I. Piza-Davila, and Francisco Cervantes-Alvarez

PhDEngScITESO-16-08-R.docx

July 14, 2016
Tlaquepaque, Mexico 45604

Doctoral Program in Engineering Sciences
ITESO (*Instituto Tecnológico y de Estudios Superiores de Occidente*)

A NOVEL USER NETWORK PROFILE BASED ON HOST NETWORK TRAFFIC

Alvaro I. Parres-Peredo, Hugo I. Piza-Davila, and Francisco Cervantes-Alvarez

July 14, 2016

Doctoral Program in Engineering Sciences
ITESO (*Instituto Tecnológico y de Estudios Superiores de Occidente*)

Tlaquepaque, Mexico 45604
Tel +52 33 3669 3598
E-mail: dci@iteso.mx

Keywords *computer network, network security, user profile, network traffic*

Abstract

To determine if the network traffic at a host in an organization corresponds to a specific user is a challenge in the field of computer network security. Some authors have built network user profiles based on traffic captured at the border of the organization. This work proposes a 4-view network profile constructed from capturing network traffic at the user host. As an example, we present the resulting network profile of a user, after capturing traffic at his host during four labor days; then, we compare the days with each other.

I. INTRODUCTION

Intrusion detections systems have been employed to detect new patterns of attacks using machine-learning algorithms. Most of these systems work at the border of the network, assuming all the attacks come from outside.

In large organizations, like universities, many users (students, employees, visitors) are connected to the campus area network for accessing intranet services or getting internet access, through different kinds of devices. The probability for an attack to occur from inside is high because of two main reasons: a) malicious behaviors of inexperienced users when putting in practice some hacking technologies, e.g., the script kiddies, and b) privileged users being careless when clicking links at e-mails and web pages from untrusted sources or being victims of social engineering attacks.

This work was supported in part by ITESO's Program for Academic Level Enhancement (*Programa de Superación de Nivel Académico*, PSNA) through an assistantship granted to A. I. Parres-Peredo.

We think that a viable way to prevent these security problems is by detecting when a user is having an abnormal network behavior or his/her behavior is similar to a well-known malicious behavior like the one of a script-kiddie. This involves building individual network profiles representing normal behaviors of every end user of the organization. To do this, real-time traffic has to be captured from the nearest point to each user access device.

In this work, we propose a 4-view network profile constructed from a dataset of 11 fields. The proposed dataset is built with real network traffic.

The frames are captured at user devices but they can be captured at either the network access switch, the wireless access point or the host's default gateway, depending on the network topology and configuration.

The present document is organized as follows. Section II presents the related work to user profiling with network data. Section III defines the 4-views network profile and the use of each one of the views. Section IV defines the fields that compose the dataset. Section V presents the results of an experiment using the proposed network profile. Finally, Section VI concludes the report indicating some future work.

II. RELATED WORK

Many research works about intrusion detection systems and computer network security validate their proposals using common datasets like KDD-CUP99 [1], which is an artificial dataset for testing intrusion detection systems, and NLS-KDD [2], built at the University of New Brunswick providing a more realistic scenario since the traffic is generated by agents based on real profiles [2]. The main problem of these datasets is that they have many fields that are calculated or specific to an application, e.g. "number of failed logins", such that they are not available or easy to calculate on raw real network traffic.

The few authors that work with real network traffic capture it at the border of the networks, typically at the gateway or firewall [3], or at internet backbone link [4].

Badea [3] says that for a network administrator it is very important to understand the user behavior in computer networks. The user behavior is defined by the analysis of logged events, and an event is defined by protocols and ports. Badea [3] proposes a system for detecting the abnormal behavior of users as a Security Information and Event Management (SIEM). It is a combination of two separate legacy products: Security Information Management (SIM) and Security Event

Manager (SEM). The former provides long term storage, analysis and reporting of recorded data; the latter deals with real-time monitoring, event correlation, notification and the possibility of supervision from the console.

The implementation of Badea [3] collects the events from a firewall where the packet is checked by the firewall rules and, after the blocking decision, sent to the abnormal user detection system which is an implementation of OSSIM[1] that can collect, normalize and correlate security events occurring within a local network.

Kuai [4] proposes a different approach for profiling traffic behavior: he identifies and analyzes cluster of hosts or applications that exhibit similar communication patterns. In this approach, he uses bipartite graphs to model network traffic at the internet-facing links of the border router; then, he constructs one-mode projections of bipartite graphs to connect source hosts that communicate with the same destination host(s) and to connect destination hosts that communicate with the same source host(s). This one-mode projection graphs enable to build similarity matrices of internet end-host, with similarity being characterized by the shared number of destinations or sources between two hosts. Based on these end-hosts matrices, at the same network prefixes, a simple spectral clustering algorithm is applied to discover the inherent end-host behavior cluster.

Kuai [4] carries out an analysis over a 200 GB dataset collected from an internet backbone of 8.6 GB/s bandwidth. The data was reduced by adding packet traces into 5-tuple network flows. The dataset was built using 24-bit network prefixes with timescale of 10 s, 30 s and 1 min; these timescales were chosen because they produced the highest percentages of hosts in the top cluster. Kuai concluded the following: 1) there is no correlation between the number of observed hosts and the number of behavior clusters, 2) the majority of end-hosts stay in the same behavior cluster over time, and 3) the profiling of network traffic in network prefixes detect anomalous traffic behaviors.

A similar approach was employed by Qin [5] but working only with traffic at port 80 (HTTP protocol), and integrating the destination URL, not only the IP address. One of the conclusions is that 93% of the hosts remain on the same behavior cluster.

However, not all the security problems occur at the network border, they also occur internally, e.g., 1) Arp Spoof Attack, 2) DNS Spoofing, 3) ICMP Redirect Attack, and 4) Wireless Replay Attack.

[1] Open Source Security Information Management - https://www.alienvault.com/products/ossim

III. 4-VIEW NETWORK USER PROFILE

In order to identify security issues, we propose a methodology for profiling normal user behavior, so that any network trace not following this profile can be considered as suspicious. The profile proposed is built upon the network traffic that the user generates, and is organized by the following levels of abstractions or views, each one having a specific security purpose, additional to the purpose of making the user profile:

a) Remote hosts communications or IP view

b) Remote services accessed or service view

c) Web hosts visited or host view, and

d) HTTP URLs accessed or URL view.

A. Remote Host Communications or IP View.

The main purpose of this view is to identify communications with remote devices which do not belong to the user profile.

The IP addresses of all the remote hosts that the user established connection with during a period of time is obtained. For each host, the total amount of incoming and outgoing bytes are calculated. This view includes only the top IP addresses according to the sum of incoming and outgoing amount of bytes transferred. The number of packets is ignored because of the uncontrolled packet segmentation allowed by IPv4.

B. Remote Services Accessed or Service View.

Helping to detect Trojans or malware installed at the user equipment is the main purpose of this view, which presents the top network services that the user reaches according to the sum of incoming and outgoing amount of bytes transferred. In this view, a network service is defined as a 3-tuple: <remote IP address, transport protocol, remote port>.

This view also helps to detect unauthorized services installed at internal hosts or servers. Most of the research works on intrusion detection consider this view, but located at the border of the organization network.

C. Web Hosts Visited or Host View

This view includes the top HTTP hosts visited by the user according to the number of times each host has been requested. This view is included in the user profile because of two main reasons: 1) since some networks use a HTTP Proxy, all the HTTP packets would have the same remote IP address; 2) nowadays, the HTTP servers support multiple domains and websites at the same IP address. These two reasons make the first two views less precise.

D. HTTP URLs Accessed or URL View

The last view provides the top visited URLs and HTTP methods employed according to the number of requests performed. This view has the aim of detecting JavaScript-based attacks that occur as a consequence of auto-execution of JavaScript programs by web browsers. The Cross-Site Scripting (XSS) attack [6] is an example of this. The BeEF2 project is a penetrating testing framework that focuses on web browsers providing tools to develop this kind of attacks.

IV. DATASET REQUIRED FOR NETWORK USER PROFILE.

In order to generate the four views proposed in the previous section, it is necessary to extract some specific data from the network traffic structured on the basis of the TCP/IP model.

A. Network Layer Model

The network traffic is comprised of packets; each packet encapsulates data organized in layers in accordance with the TCP/IP Network Model [7]. Fig. 1 depicts this encapsulation. The TCP/IP Network Model defines the following layers:

1. Application layer. It contains the logic required to support the various applications. For each different type of application, a particular internal structure is employed.
2. Transport layer. Also defined by Stalling as host-to-host layer [7], it provides an end-to-end delivery service that might have reliability mechanisms or not.
3. Internet layer. It provides procedures to allow data traverse multiple interconnected networks, using the Internet Protocol (IP).
4. Network access layer. Also called data link layer, it provides mechanisms for accessing and routing data across a network between two devices attached to the same network.
5. Physical layer. Covers the physical interface between a data transmission device and a transmission medium or network.

[2] BeFF – Browser Exploitation Framework Project - http://beefproject.com/

There are different protocols for each layer; some layers define many protocols (application layer) and others define few protocols (internet layer). Each protocol defines how data is organized inside a layer. Most of them define two sections: 1) a header, to store control information; and 2) a body, containing application payload or encapsulation of an upper layer. Fig. 2 depicts this generic structure.

B. Dataset Fields

The dataset proposed has eleven fields. Table I presents each of them with the following information:

a) Field: the name of the field in the dataset proposed

b) Layer: the TCP/IP Network layer where the data belongs

c) Header field: the name of the field in the packet captured

d) View: the name of the view(s) using this field.

The fields User and Timestamp are included with the purpose of allowing data analysis in a future work. Every captured packet provides all the data required to fill each of the remaining fields, taking into consideration that the HTTP-based fields might be empty if the packet captured is not HTTP. The Far and Local Hardware Address fields are extracted with the purpose of knowing the direction of the packet: incoming or outgoing. If incoming, the value for Far-IP-address field is taken from Header Field Source Address at the internet layer; otherwise, it is taken from the Destination Address, same layer. The same logic is applied for Far-Port field.

V. EXPERIMENTAL RESULTS

A computer application that captures the network traffic was developed using Java SE 1.8. JNetPcap4 library was employed for packet capture, decoding and data extraction. In order to allow offline analysis of the information, this application stores the traffic in one CSV file for each 10 MB of data, where each line of the file denotes a captured packet and contains only the corresponding values for each of the fields introduced in Section 4.B.

The campus area network used in this experiment has a 16-bit network; it has a Windows domain controller and uses a HTTP Proxy. The campus applications include web-apps and remote desktop apps. The email service is provided by Microsoft Exchange Server which is hosted outside of the campus network.

The traffic on one host was captured. Since the network allows mobility and the host was a laptop, in some periods of time it was attached to the network over a wired connection and in some others over a wireless connection. At the wired network, a gigabyte Ethernet connection was used and the host has a static IP address with a 24-bit mask. At the wireless network, it has a dynamic IP address with a 22-bit mask configured by DHCP.

The traffic was captured during four labor days. The size of the traffic captured was 2.56 gigabytes, involving 4’355,262 packets. Table II presents general statistics about the capture. Each of the views was constructed in the form of a table. Fig. 3 shows examples of the tables that represent each of the views.

A timeframe of 24 hours was selected to build the user network profile. In order to select the number of top elements for each view, we first selected the number of elements that represents around 90% of data, using the amount of bytes for remote host communication and remote services accessed views, and number of request for web hosts visited and HTTP URLs accessed views. The number of elements of each view at each timeframe are exposed at Table 3. Based on this information 53, 60 75, 3103 were the number of top elements selected for IP view, service view, host view and URL view respective, which represents the average.

A similarity matrix was constructed to compare the user network profile of each one of the four days. Tables IV to VII shows the similarity matrix of each view; the number represents the percentage of equal elements between two days. In average, the similarity between profiles is 26.59% percentage.

VI. CONCLUSIONS AND FUTURE WORK

In this work, we have presented a 4-view network user profile built from real-time network traffic captured at the end-user host.

The main differences from other approaches that generate user profiles based on network traffic are the following: 1) the network traffic is captured at the end-user host or at a near network point, and 2) the use of four different views to build the user profile: IP Address, Service reached, HTTP Host and URL.

We captured traffic at one host during four labor days and constructed a network profile of the same user for each day; then we compared the network profiles with each other to find similarities. We observed that the IP, Services and Hosts views showed all a similarity around

30%; however, the URL view showed less than 4% of similarity.

Future work includes: 1) analyze the pertinence of the URL view for profiling users; and, 2) test the proposed user profile with an algorithm capable of determining whether a given traffic sample corresponds to the same user or not.

REFERENCES

[1] K. Kandall, *A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems*, B.S. and M. Sc Thesis, Dept. of Electrical Eng. and Computer Science, MIT, Boston, MA, USA, 1999.

[2] A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, “Toward developing a systematic approach to generate benchmark datasets for intrusion detection,” *Comput. Secur.*, vol. 31, no. 3, pp. 357–374, May 2012.

[3] A. Badea, V. Croitoru, and D. Gheorghica, “Computer networks security based on the detection of user’s behavior,” in *9th International Symposium on Advanced Topics in Electrical Engineering (ATEE)*, Bucharest, Romania, May. 2015, pp. 55–60.

[4] K. Xu, F. Wang, and L. Gu, “Behavior analysis of internet traffic via bipartite graphs and one-mode projections,” *IEEEACM Trans. Netw.*, vol. 22, no. 3, pp. 931–942, Jun. 2014.

[5] T. Qin, X. Guan, C. Wang, and Z. Liu, “MUCM: multilevel user cluster mining based on behavior profiles for network monitoring,” *IEEE Syst. J.*, vol. 9, no. 4, pp. 1322–1333, Dec. 2015.

[6] S. Gupta and B. B. Gupta, “Cross-Site scripting (XSS) attacks and defense mechanisms: classification and state-of-the-art,” *Int. J. Syst. Assur. Eng. Manag.*, pp. 1–19, Sep. 2015.

[7] W. Stallings, *Data and Computer Communications*, Boston. Ma: Pearson, 2014.

TABLE I
FIELDS OF DATASET

Dataset Field	Layer	Header Field	View(s)
Far-Hardware-address	Network Access	Source Address / Destination Address	Remote Host and Remote Services
Local-Hardware-address	Network Access	Source Address / Destination Address	Remote Host and Remote Services
Far-IP-address	Internet	Source Address / Destination Address	Remote Host and Remote Services
Protocol	Internet	Protocol	Remote Services
Far-Port	Transport	Source Port / Destination Port	Remote Services
Bytes	Internet	Length	Remote Host and Remote Services
HTTP-Host	Application (HTTP)	Host	Web Host HTTP URLs
HTTP-URL	Application (HTTP)	URL	HTTP URLs
HTTP-METHOD	Application (HTTP)	Method	HTTP URLs

TABLE II
GENERAL STATISTICS OF CAPTURE TRAFFIC

	Day 1	Day 2	Day 3	Day 4	Total
Total Bytes (MB)	885 34%	718 27%	494 19%	532 20%	2,629
Count Packets	1,380,498 32%	1,248,241 29%	904,538 21%	821,985 19%	4,355,262
Different IPs	1,111 49%	1,033 46%	841 37%	658 29%	2,260
Different Services	8,078 25%	9,548 29%	10,806 33%	6,820 21%	32,641
Different HTTP Hosts	382 51%	338 45%	216 29%	209 28%	743
Different URLs	5,980 43%	5,115 37%	1,395 10%	1,778 13%	13,768

TABLE III
NUMBER OF ELEMENTS THAT REPRESENTS 90% OF DATA

	Day 1	Day 2	Day 3	Day 4	Avg.
IP View	56	65	56	33	53
Service View	66	73	64	37	60
Host View	89	64	76	69	75
URL View	5241	4403	1215	1553	3103

TABLE IV
IP VIEW - SIMILARITY MATRIX (%)
Average 37.20%

	Day 1	Day 2	Day 3	Day 4
Day 1	-	44.64	41.07	30.36
Day 2	44.64	-	44.64	26.79
Day 3	41.07	44.64	-	35.71
Day 4	30.36	26.79	35.71	-

TABLE V
SERVICE VIEW - SIMILARITY MATRIX (%)
Average 34.72%

	Day 1	Day 2	Day 3	Day 4
Day 1	-	43.33	38.33	30.00
Day 2	43.33	-	41.67	25.00
Day 3	38.33	41.67	-	30.00
Day 4	30.00	25.00	30.00	-

TABLE VI
HTTP HOST VIEW – SIMILARITY MATRIX (%)
Average 30.44%

	Day 1	Day 2	Day 3	Day 4
Day 1	-	33.33	33.33	26.67
Day 2	33.33	-	30.67	21.33
Day 3	33.33	30.67	-	37.33
Day 4	26.67	21.33	37.33	-

TABLE VII
HTTP URL VIEW - SIMILARITY MATRIX (%)
Average 3.98%

	Day 1	Day 2	Day 3	Day 4
Day 1	-	2.96	2.87	3.38
Day 2	2.96	-	1.48	2.26
Day 3	6.38	3.30	-	6.88
Day 4	5.91	3.94	5.40	-

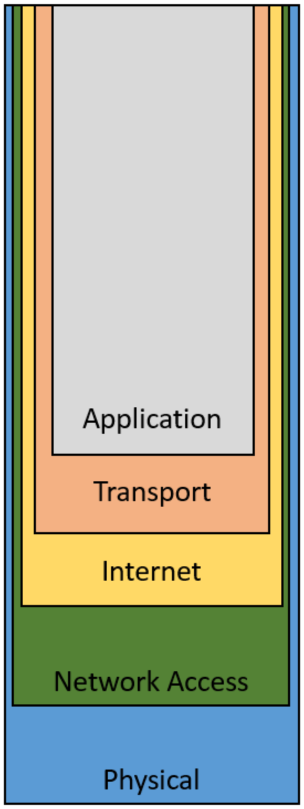


Fig. 1. Graphical representation of encapsulation and layers of the TCP/IP Network Model.

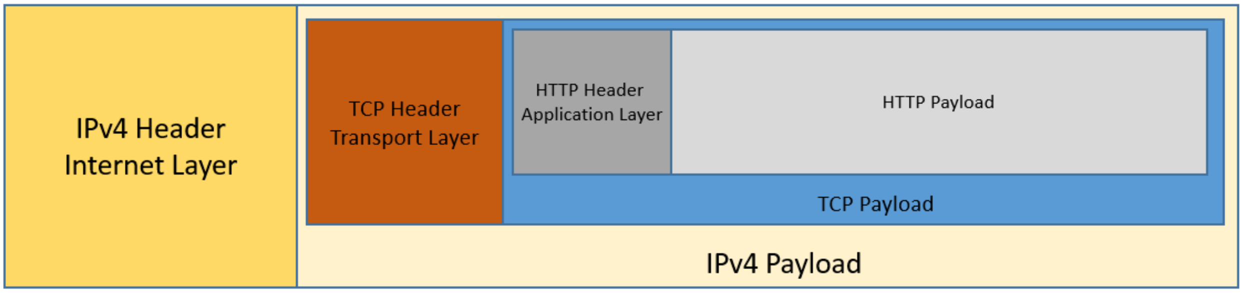
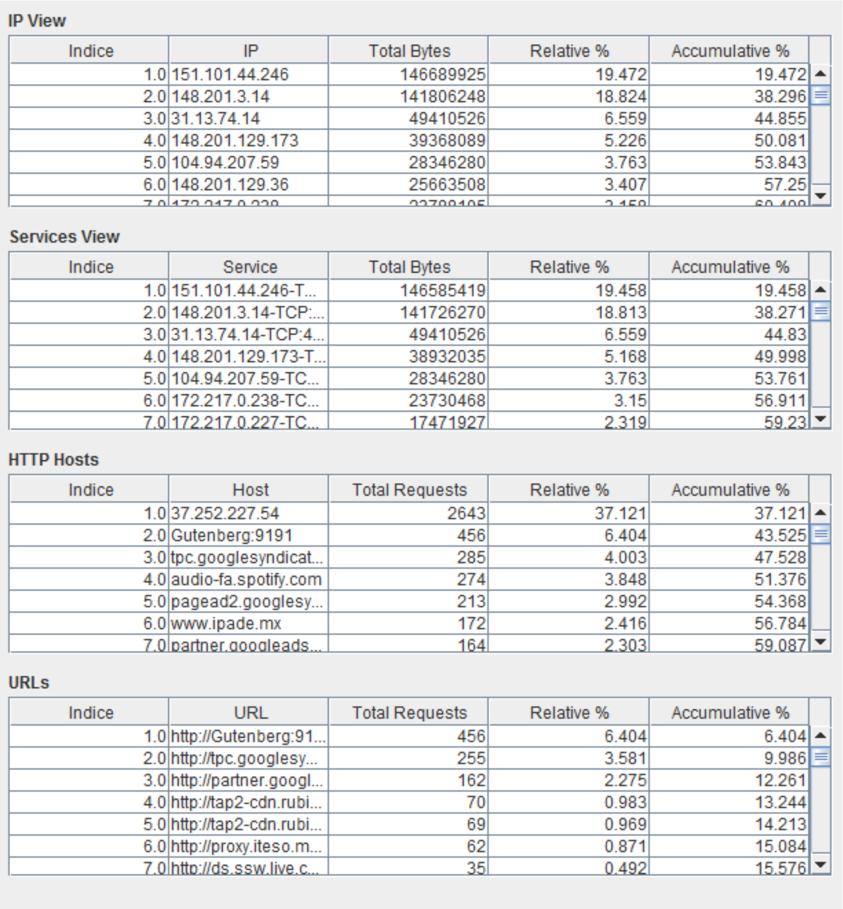


Fig. 2. Each TCP/IP protocol defines how data is organized inside a layer. Most of them define two sections: 1) a header, to store control information; and 2) a body, containing application payload or encapsulation of an upper layer.



IP View

Indice	IP	Total Bytes	Relative %	Accumulative %
1.0	151.101.44.246	146689925	19.472	19.472
2.0	148.201.3.14	141806248	18.824	38.296
3.0	31.13.74.14	49410526	6.559	44.855
4.0	148.201.129.173	39368089	5.226	50.081
5.0	104.94.207.59	28346280	3.763	53.843
6.0	148.201.129.36	25663508	3.407	57.25
7.0	172.217.0.238	23788105	3.158	60.408

Services View

Indice	Service	Total Bytes	Relative %	Accumulative %
1.0	151.101.44.246-T...	146585419	19.458	19.458
2.0	148.201.3.14-TCP:...	141726270	18.813	38.271
3.0	31.13.74.14-TCP:4...	49410526	6.559	44.83
4.0	148.201.129.173-T...	38932035	5.168	49.998
5.0	104.94.207.59-TC...	28346280	3.763	53.761
6.0	172.217.0.238-TC...	23730468	3.15	56.911
7.0	172.217.0.227-TC...	17471927	2.319	59.23

HTTP Hosts

Indice	Host	Total Requests	Relative %	Accumulative %
1.0	37.252.227.54	2643	37.121	37.121
2.0	Gutenberg:9191	456	6.404	43.525
3.0	tpc.googlesyndicat...	285	4.003	47.528
4.0	audio-fa.spotify.com	274	3.848	51.376
5.0	pagead2.googlesy...	213	2.992	54.368
6.0	www.ipade.mx	172	2.416	56.784
7.0	partner.googleads...	164	2.303	59.087

URLs

Indice	URL	Total Requests	Relative %	Accumulative %
1.0	http://Gutenberg:91...	456	6.404	6.404
2.0	http://tpc.googlesy...	255	3.581	9.986
3.0	http://partner.googl...	162	2.275	12.261
4.0	http://tap2-cdn.rubi...	70	0.983	13.244
5.0	http://tap2-cdn.rubi...	69	0.969	14.213
6.0	http://proxy.iteso.m...	62	0.871	15.084
7.0	http://ds.ssw.live.c...	35	0.492	15.576

Fig 3. Example of the tables that represents each of the views.

APPENDIX OF FILES USED

Filename	Author	Short Description