Towards a User Network Profiling for Internal Security using Top-K Rankings Similarity Measures

Alvaro Parres-Peredo, Ivan Piza-Davila, Francisco Cervantes
Department of Electronics, Systems and Informatics
ITESO - The Jesuit University of Guadalajara
Tlaquepaque, Mexico
parres@iteso.mx,hpiza@iteso.mx,fcervantes@iteso.mx

***Abstract*—A major goal of current computer network security systems is to protect the network from outside attackers; however, protecting the network from its own users is still an unattended problem. In campus area networks, the risk of having internal attacks is high because of their topologies and the amount of users. This work proposes a new approach to identify whether a network user is having or not a normal behavior, by analyzing host traffic using top-k ranking similarity measures. The result of this analysis could be an input of intrusion detection systems. The document presents an experiment where real-time traffic of different users in a campus area network is compared to a reference traffic that corresponds to one of them.**

***Keywords*—computer network, network security, user profile, network traffic, top-k rankings**

I. Introduction

On current computer networks, traditional security methods like firewalls, access control systems and simple Intrusion Detection Systems (IDS) are no longer enough to protect computer systems; day after day, intruders find new ways to attack computers and systems. This has given rise to the research and development of new security technology since 1980, when Anderson proposed the first IDS approach [1].

Nowadays, attackers use advanced techniques to go undetected by IDSs, including the following: IP address spoof, encrypted payload, or even social engineering techniques [2]. A common symptom of an attack using these techniques is that the host under attack is experiencing unexpected network behavior. This is why the use of profiles to determine whether the user is having the expected behavior or not has become necessary as a new way to detect intrusions.

Many authors have proposed using network profiles [3]–[5]. However, these works focus on the traffic at the border to build the profiles, losing visibility of internal attacks.

In this work, we propose to build the user profile using the traffic captured at the host or a nearby point, e.g., the switch at the access layer. The proposed profile is built upon the remote services accessed by the host in real time, and treated as a top-k ranking. In order to determine whether the user behaves as expected, an attack-free user profile -normal-behavior- is built a priori and compared with the current user profile by means of similarity measures for top-k rankings.

In order to validate our approach, we have built a normal-behavior profile of a selected user, and then calculated similarities with the same user and with two others.

The present document is organized as follows. Section II presents some works about user profiling and network traffic profiling. Section III explains the proposed methodology in detail. Section IV contains the experimental results. And finally, Section V concludes the report and suggests some future work.

II. Related Work

The definition of network user profiles to represent network behavior has been part of the research into computer network security.

Kihl et al. [6] present a study of traffic analysis and characterization of Internet users to help understand Internet usage and the demands on broadband access. They use a commercial tool for capturing and classifying traffic according to Internet protocols and applications. The authors conclude that Internet usage has changed from traditional WWW requests to a more complex use. Looking at Internet usage in 2010 they found that most of the traffic came from file-sharing protocols (74%), media streaming (7.6%), and web-traffic (5.5%). The traffic for this work was collected from a Swedish municipal FTTH network.

Sing et al. [3] present an intrusion detection technique using network-traffic profiling and an extreme online sequential machine-learning algorithm. The proposed methodology uses one profiling procedure called alpha profiling that creates profiles on the basis of protocol and service features, and a second profiling process, beta profiling, where the alpha profiles are grouped to reduce the number of profiles. The authors conducted three different experiments: 1) using all features and alpha profiling, 2) using only some features and alpha profiling, and 3) using only some features, alpha profiling and beta profiling. The best results were obtained from the last experiment using both profiling methods. The dataset used for this work was NSL-KDD.

A work that builds profiles of network prefixes instead of users is presented by Qin et al [4], who propose aggregating

This work was supported in part by ITESOs Program for Academic Level Enhancement (Programa de Superacin de Nivel Acadmico, PSNA) through an assistantship granted to A. Parres-Peredo

traffic based on network prefixes in order to reduce the amount of data to be processed, and then calculate clusters using a *k*-means algorithm. Qin found that similar users produce similar traffic; with this information, decisions about security and management can be taken. The traffic used for this work was captured at the CERNET backbone.

A similar work is presented by Xu et al. [5], who proposed a methodology that analyzes Internet traffic. This methodology first constructs bipartite graphs; after this, it generates one-mode projections; then, it builds a similarity matrix and generates clusters with a spectral clustering method; finally, it analyzes the clusters. The traffic used in this work was captured at the backbone of a large Internet service provider, aggregating the information using 24-bit length prefix networks, and the network 5-tuple.

As we can see, all the works presented here have used the traffic captured at a point far from the end-user host, even outside the users local network, leaving the internal network security unattended. On the other hand, the usage of profiles has proved to be viable to either identify or specify network behaviors.

III. METHODOLOGY PROPOSED

The proposed methodology has the goal of determining how similar the traffic captured in real time is to the traffic captured a priori in a controlled environment, i.e., the normal behavior.

The network traffic is captured at the host. For each packet the following data is collected: a) remote IP address, b) transport protocol, c) remote port and d) total length.

A. Building normal user behavior

This phase builds a user network profile called normal behavior, which will be used as a reference for calculating the similarity factor. Fig. 1 shows the overall process at this phase

Network traffic $P_{SE,x}$ is captured from user x's host during a period of time T in a secure environment in which we can guarantee that the host will be used only by the expected user and there are no malware, virus, Trojan or any other malicious software installed. The period of time T must be long enough to make sure that habitual tasks are registered.

From $P_{SE,x}$, a subset p that corresponds to the period of time $[t \cdots t+f]$ is selected. From traffic p, a top-*k* ranking of accessed services is calculated using the total transferred bytes as weight measure of each service. The ranking is added to K_x. This process is repeated while t is less than T. At the end of each iteration, t increases by Δt, a value smaller than f in order to produce overlaps in time frames.

An extraction of a top-*k* ranking is illustrated in Fig. 2, where $k = 10$ and five time frames are listed.

B. Capturing regular traffic

In this phase, regular traffic, $P_{RT,x}$ is captured in real time from the user x's host during a period of time $[t \cdots t+f]$. Using this traffic, a top-*k* ranking is built and the best similarity factor against every top-*k* ranking in K_x is calculated, as described in the next section. This factor will be useful to determine whether the current traffic corresponds to the expected users normal network behavior. This process is repeated the next time frame: $t + \Delta t$. Fig. 3 shows the flow chart of the process.

C. Calculating best similarity factor

The purpose of this phase is to find out if the real-time traffic captured during a period of time $l[t \cdots t+f]$ resembles any traffic within the records of the normal behavior captured during a period of time of length f. Thus, a similarity factor is calculated between the top-*k* ranking of the real-time traffic (κ_x) and each of the top-*k* rankings stored in K_x. According to this value, a decision might be taken about the correspondence

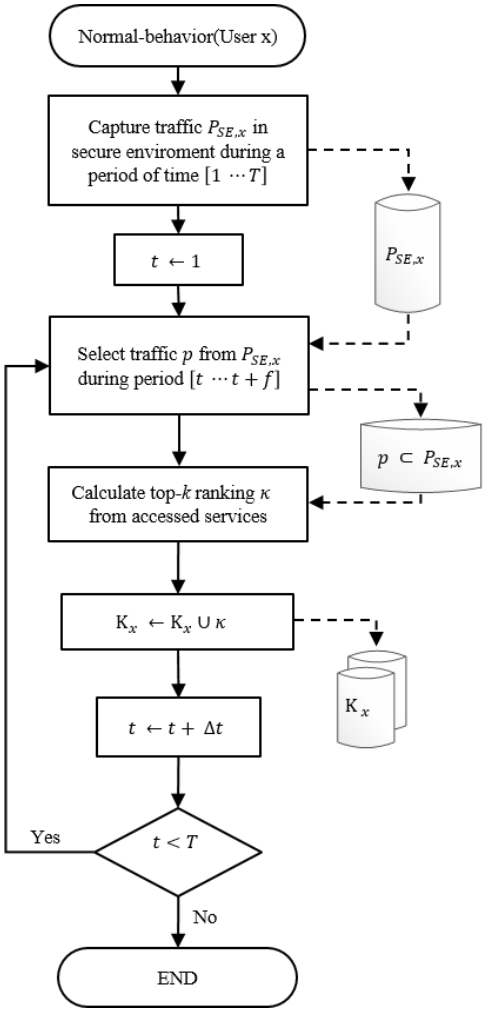

Fig. 1. Building normal user behavior.

Timeframe	*Top 1*	...	*Top 10*
$[t \cdots t+f]$	148.201.129.173:TCP80	...	148.201.140.50:TCP80
$[t+\Delta t \cdots t+\Delta t+f]$	148.201.129.173:TCP80	...	148.201.140.50:TCP80
$[t+2\Delta t \cdots t+2\Delta t+f]$	148.201.129.173:TCP80	...	148.201.140.98:TCP339
$[t+3\Delta t \cdots t+3\Delta t+f]$	132.245.44.2:TCP443	...	148.201.129.43:TCP80
$[t+4\Delta t \cdots t+4\Delta t+f]$	148.201.129.148:TCP443	...	148.201.129.43:TCP80

Fig. 2. Extraction of a top-*k* ranking

of the current network traffic to the expected traffic. Fig. 4 shows a diagram of this process.

To compare each pair of top-k rankings, it is necessary to use a ranking similarity measure, but with the peculiarity that these top-k rankings are non-conjoint rankings; this means that not all elements of one of them are present in the other. Thus, we have explored two different measures: 1) Spearmans rho [7] and 2) Average Overlap [8]. We decide to use the second measure because it produced better results. This measure, in general, calculates for each $d \in \{1 \cdots k\}$ the overlap at d, and then averages those overlaps to derive the similarity measure.

Formally, the average overlap between two top-k lists can be expressed as:

$$AO(S,T,k) = \frac{1}{k}\sum_{d=1}^{k} A_{S,T,d} \tag{1}$$

where S and T are top lists of k number of elements, and $A_{S,T,d}$ is defined as:

$$A_{S,T,d} = \frac{\mid S_{:d} \cap T_{:d} \mid}{d} \tag{2}$$

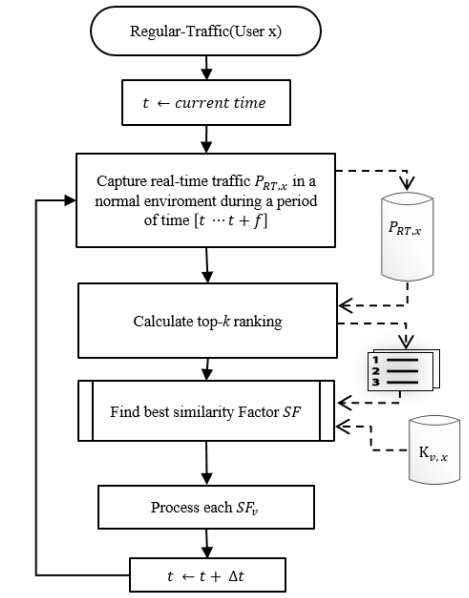

Fig. 3. Process of capture real-time traffic from user x.

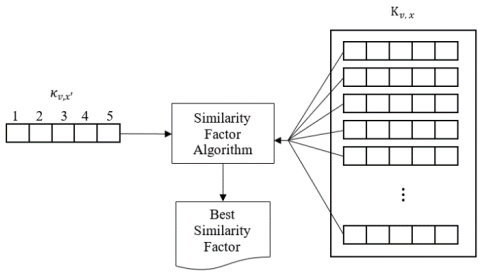

Fig. 4. Calculating best similarity factor.

Fig. 5 shows an example where the Average Overlap (AO) between two top-5 lists is calculated.

IV. Experiments and Results

A. Experiment setup

The experiment was carried out on a Campus Area Network (CAN) that has a 16-bit network; it has a Windows domain controller and uses a HTTP proxy. Campus applications include web-apps and remote desktop apps. The email service is provided by Microsoft Exchange Server which was hosted outside of the campus network.

The target users were full-time professors, who had a computer with two types of access to the network: a wired access with a static IP address and a wireless access with a dynamic IP address.

Five full-time professors (hereafter denoted as users) were selected for this experiment and, for each one, we generated his normal-behavior profile K_A to K_E, and then captured real-time traffic κ_A to κ_E.

Different values of the parameters were evaluated: $f = 1, 5, 10 minutes$, $\Delta t = 10, 30, 60 seconds$, the number of elements selected for the top-k rankings, $k = 10, 25, 50, 100$, and both measure functions.

The best tuple $< t, \delta t, k, Function >$ found was: $t = 5min$, $\delta t = 10sec$, $k = 10$, $AverageOverlap$, because it accentuated the differences between $AO(K_x, \kappa_x)$ and all $AO(K_x, \kappa_y)$ where y are the rest of users.

B. Capturing and processing traffic

During a labor week, the normal behavior was captured from each user's computer. Before starting to capture, we checked that no computers had any malicious software installed. During this period only the owner user had access to each computer. The average size of the traffic captured was 3 gigabytes per user, involving more than four million packets. All the packets were processed as described in Section 3.A.

On a different labor week, real-time traffic was captured from the same computers. This traffic was processed as described in Section 3.B and all the produced top-k were stored in a different collection for each user.

C. Similarity calculation and results

From each collection of top-k, two different 1000 top-k sets were selected randomly, κ_{A_1}, κ_{A_2}, $\cdots$, κ_{E_1}, κ_{E_2}. The best similarity factor with respect to K_A, $\cdots$, K_E was found and

d	$S_{:d}$	$T_{:d}$	$A_{S,T,d}$	$AO(S,T,d)$
1	$< a >$	$< x >$	0.0000	0.0000
2	$< ab >$	$< xc >$	0.0000	0.0000
3	$< abc >$	$< xcb >$	0.6667	0.2222
4	$< abcd >$	$< xcby >$	0.5000	0.2917
5	$< abcde >$	$< xcbye >$	0.6000	0.3534

Fig. 5. Similarity calculation of two top-5 lists using average overlap measure.

plotted. Fig. 6 and 7 show respectively the similarity factors corresponding to κ_{B_1} and κ_{C_1} against each normal-behavior $K_A \cdots K_E$. We can see that users B and C exhibited a higher similarity to their own normal behavior than the rest of the users.

Table I shows the average of the similarity factors of each evaluation set κ_{A_1},κ_{A_2}, $\cdots$, κ_{E_1},κ_{E_2} against all normal behavior sets $K_A \cdots K_E$. We can observe that the highest average of each evaluation set is found at the column that corresponds to the same user (main diagonal).

TABLE I. AVERAGE SIMILARITY FACTORS

Evaluation Set	K_A	K_B	K_C	K_D	K_E
κ_{A_1}	0.516	0.434	0.401	0.312	0.267
κ_{A_2}	0.516	0.434	0.401	0.312	0.267
κ_{B_1}	0.571	0.920	0.555	0.389	0.322
κ_{B_2}	0.571	0.914	0.548	0.391	0.330
κ_{C_1}	0.273	0.419	0.855	0.189	0.354
κ_{C_2}	0.279	0.417	0.847	0.197	0.354
κ_{D_1}	0.295	0.179	0.247	0.446	0.293
κ_{D_2}	0.296	0.176	0.245	0.447	0.289
κ_{E_1}	0.263	0.336	0.364	0.194	0.553
κ_{E_2}	0.256	0.335	0.362	0.198	0.552

V. CONCLUSION AND FUTURE WORK

In this work, we have proposed a methodology to determine how similar the traffic captured in real time at a user host κ_x, is to previously captured traffic, which we called normal behavior K_x. We present the results of the implementation of this methodology.

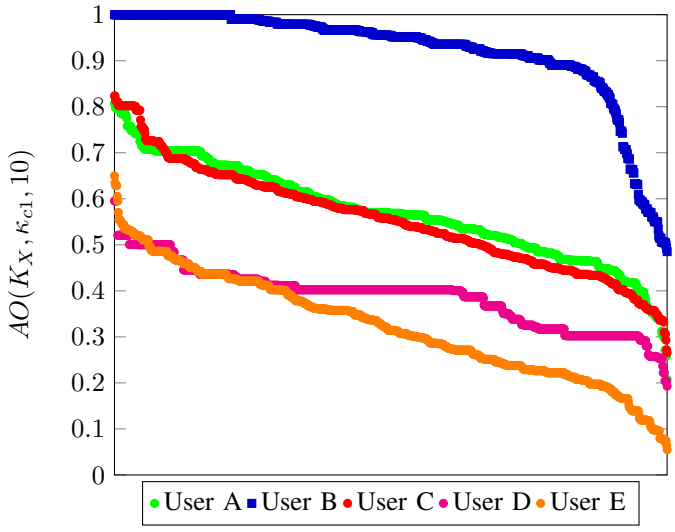

Fig. 6. Similarity Factors of κ_{B_1} against K_A to K_E ordered by value

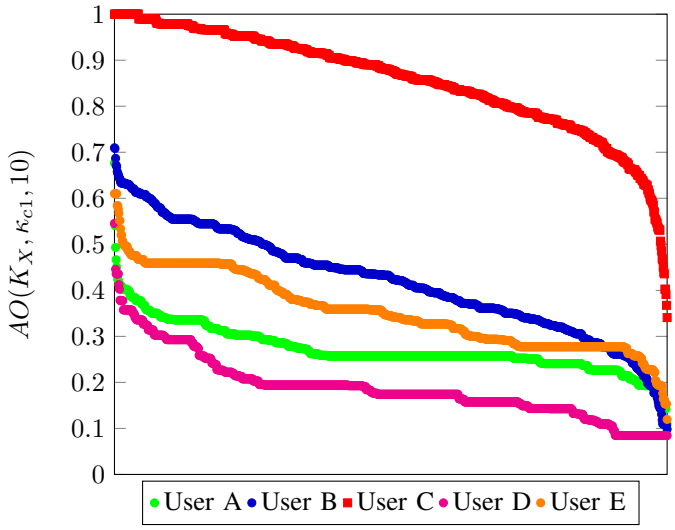

Fig. 7. Similarity Factors of κ_{C_1} against K_A to K_E ordered by value

An early conclusion from this work is that the proposed user-network profile allows us to determine whether the captured traffic corresponds to the expected user or not. In the results of the experiments we can see that a users real-time traffic has a higher similarity to his own normal-behavior than that of other users.

From the charts, we can see that only few top-k ranking from κ_{b_1} and κ_{c_1} are identical to any top-k from K_B and K_C, i.e., the similarity factor is 1.0. A possible reason for this is that multiple IP addresses can be configured by the same host or service, or the user does not actually do exactly the same thing all the time. But we consider that the average similarity factors obtained are good enough to differentiate between the expected user and the others.

In future work, the similarity factors obtained will be employed by a real-time process that determines how likely it is for the current traffic to belong (or not) to the expected user. More experiments are still necessary to determine the consistency of the proposed methodology.

REFERENCES

[1] J. P. Anderson, "Computer Security Threat Monitoring And Surveillance," NIST, Fort Washington, PA, Report Contract 79F296400, 1980.

[2] Paul Wood, Ed., *ISTR Internet Secuirty Threat Report*. Symantec, Apr. 2016.

[3] R. Singh, H. Kumar, and R. K. Singla, "An intrusion detection system using network traffic profiling and online sequential extreme learning machine," *Expert Systems with Applications*, vol. 42, no. 22, pp. 8609–8624, Dec. 2015.

[4] T. Qin, X. Guan, C. Wang, and Z. Liu, "MUCM: Multilevel User Cluster Mining Based on Behavior Profiles for Network Monitoring," *IEEE Systems Journal*, vol. 9, no. 4, pp. 1322–1333, Dec. 2015.

[5] K. Xu, F. Wang, and L. Gu, "Behavior analysis of internet traffic via bipartite graphs and one-mode projections," *IEEE/ACM Transactions on Networking*, vol. 22, no. 3, pp. 931–942, Jun. 2014.

[6] M. Kihl, P. dling, C. Lagerstedt, and A. Aurelius, "Traffic analysis and characterization of Internet user behavior," in *2010 International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, Moscow, Oct. 2010, pp. 224–231.

[7] R. Fagin, R. Kumar, and D. Sivakumar, "Comparing Top k Lists," *SIAM Journal on Discrete Mathematics*, vol. 17, no. 1, pp. 134–160, Jan. 2003.

[8] W. Webber, A. Moffat, and J. Zobel, "A Similarity Measure for Indefinite Rankings," *ACM Trans. Inf. Syst.*, vol. 28, no. 4, pp. 20:1–20:38, Nov. 2010.