A Novel Behavior-based Tracking Attack for User Identification

Xiaodan Gu*, Ming Yang*, Jiaxuan Fei , Zhen Ling*, Junzhou Luo*
*School of Computer Science and Engineering
Southeast University, Nanjing, P.R. China
{guxiaodan; yangming2002; zhenling; jluo }@seu.edu.cn
State Grid Smart Grid Research Institute: feijiaxuan@sgri.sgcc.com.cn

***Abstract*—Currently, people around the world daily use the Internet to access various services, such as, email and online**

In this paper, we study the behavior-based tracking techniques and propose a novel method to link multiple sessions of the same user. First, we utilize PFQ to capture the high-speed network traffic and construct two real world traffic datasets of different scale. Second, we perform the data preprocessing to the traffic, including picking out all GET requests for the html documents, converting the IP addresses to domain names, extracting primary domain names, and so on. Then we extract appropriate features from traffic to create behavior profiles relying on the users' activity patterns. Finally, we link multiple sessions with the Multinomial Naive Bayes Classifier. The results show that the behavior-based tracking attack is still feasible in the large-scale scenarios. We also discuss some countermeasures which can be used to resist the behavior-based tracking attack.

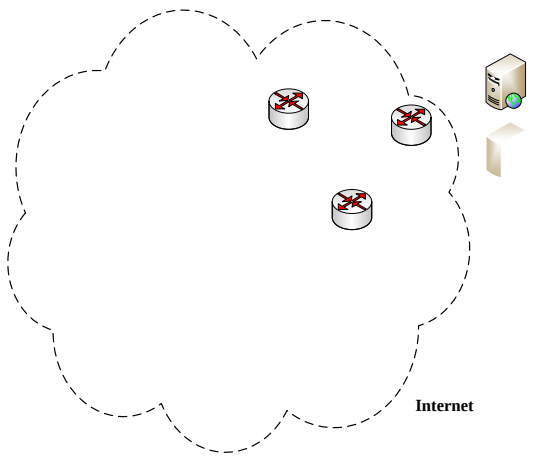
The rest of this paper is organized as follows. Section II presents the related work. In Section III, we describe the threat model and propose a novel behavior-based tracking method. We provide the results of our experiments in Section IV. We also discuss the countermeasures in Section V. Section VI concludes this paper and discusses the future work.

II. RELATED WORK

In the biometric field, the behavioural biometric is a mature technique based on users' skills, styles, preferences, knowledge, motor-skills or strategies. For example, researchers propose varieties of methods to identify users according to some subtle differences in patterns of mouse movements [2-4] or keystroke dynamics [5-7].

In the network application field, Padmanabha et al. [8] find that humans have unique clickprints when they browse the same Website. They extract the duration, number of pages viewed, average time spent per page, the starting time, and the starting day of the week per session to construct the clickprint pattern from real web browsing data. On this basis, Yang [9] proposes two profiling techniques and three additional criteria based on the concepts of support and lift. When the profiles are built, she evaluates the similarity by using the Euclidean distance. Besides, researchers believe that the email data also can be used to identify users. Vel et al. [10] use 170 style marker attributes and 21 structural attributes in the classification, including the greeting acknowledgment, farewell acknowledgment, signature text, number of attachments and so on. To solve the problem of high dimensional features, Lackner et al. [11] apply the concept of Activation Patterns to email header data, which based on the artificial intelligence and machine learning techniques.

Unlike the above scenarios, in the traffic analysis field, the behavior-based tracking techniques are carried out by passively sniffing traffic and extracting behavioural features to link multiple sessions of the same user. Kumpost et al. [12] believe that the websites visited by the user and the corresponding frequencies reflect his habits. They store the destination IP address, source IP address and the number of connections in a two-dimensional matrix based on the NetFlow logs. It means that the cell (i, j) contains the number of connections that are initiated from the source IP address i to the destination IP address j. Then they employ the inverse document frequency transformation and the cosine similarity metric to get a better result. The accuracy values for SSH, HTTP and HTTPS are 61.512%, 23.571% and 26.561% respectively. Similarly, Herrmann et al. [13] apply the Multinomial Naïve Bayes classifier to the destination host access frequencies. They evaluate their method on a real world dataset from 28 users and the accuracy is up to 73%. To further study the scalability in a real world setting, they [14] implement a scalable evaluation environment with a MapReduce framework on a large-scale dataset which contains more than 2100 concurrent users ′ DNS requests. By resolving ambiguous predictions with cosine similarity, 88.2% of all instances are linked correctly. They [15] also evaluate three techniques based on the criteria support and lift through a large number of experiments, including the 1-Nearest-Neighbor classifiers, the Multinomial Naive Bayes classifier9.92017 8.8(9.92.2(7.5(e)-0 .56088 0 0 .3(1e)-0 .56n)8.2(t)-12.1



the high requirements for the capability, the attacker also can set up a free VPN or hotspot to attract traffic as an alternative. However, he can't modulate the users' traffic or send probe packets.

Similar to the related works [13-15], we assume that the IP address of each user is not changed in a fixed period of time. And all traffic generated by the same IP address in this period can be aggregated as a session.

In the remainder of this paper, we set the time to 24 hours, beginning at 0:00 am each day. The attacker can extract features from one session's traffic to create an instance.

B. Data Pre-procession

Researchers believe that the websites visited by a user can express his web behaviors and habits. So they always extract the destination IP address and number of connections from traffic to create profiles. The number of connections means the access frequency of the corresponding website.

But nowadays, most websites display multiple ads belong to other domains in their pages. And one ad may appear in different websites. When a user visits some website, his browser establishes TCP connections to request all objects embed in the page. During HTML parsing, the browser doesn't distinguish the ads elements from other objects. Therefore the generated traffic contains the ads domains, which are not in line with the user's subjectively based demand. These ads domains are obfuscated data which may decreases accuracy of the classifier. Consider the following case scenario. The user *A* and *B* visit websites in the light of different list *L* and *M* respectively. There is no intersection between the list *L* and *M*. But most of the Ads displayed by L and M are the same. In common sense, the behavior patterns of the user *A* and *B* are completely different. Unfortunately, if we extract all domains according to the existing methods, including the ads domains, there is a possibility that the classifier identifies the user *A* and *B* as the same class. Moreover, the browser creates multiple TCP connections to the server to accelerate the data transmission in an interaction. And the specific number of concurrent connections is decided by the web server and parameter settings of the user's browser. It's not equal to the access frequency. If we extract all the destination domains and corresponding connections as features, we will make the profiles ambiguous and get lower accuracy in linking sessions. Therefore, we decide to pre-process the dataset to get the user's real behaviors.

In the data pre-procession, we first need to pick out all domains that the users really want to visit. According to the HTTP protocol, [17] just the first GET request retrieves the target HTML document when a user opens a new page in his browser. All the subsequent requests refer to the objects embed in the page. They may contain some resources in other domains, e.g., the ad images and videos. Therefore, we can pick out all GET requests for the html documents to represent users' accesses. Unfortunately, no explicit identifier can indicate the appearance of an html document request. To address this problem, we try to analysis the media types of resources based on the URL strings. But the result is frustrating. When we match the URL strings with some keywords, e.g., "html" and "jsp", we discard a lot of useful records. We find that most websites apply the URL rewriting techniques to improve the usability and search friendliness, which remove the type extensions of html documents. As an alternative, we filter HTTP traffic and retain GET requests whose Accept request-header field is filled with "text/html". Then, we use the geolocation Tool to obtain the approximate geographic location of the user based on his IP address. The result can help us infer the user's physical activity areas. When multiple sessions are linked, we even can depict the user's location trajectory. Finally, because a domain name may have multiple IP addresses, we convert the destination IP address to the domain name based on DNS records. If the Host field exits in the URL string, we can use the value as the destination domain. In addition, we replace all domains with the corresponding top-level domain name by using regular expressions to mitigate the curse of dimensionality.

C. Feature Extraction

In data mining and pattern classification, it is very crucial to extract features which can reflect the true characteristics, since the selected features have a direct and significant impact on the classification accuracy. Previous works only extract domains and frequency from the traffic which ignore the high level application related information. In this paper, we primarily focus on four applications to mine details about users' habits, including HTTP, IM, Email and P2P, and define additional features that help to improve the detection rate in the following:

- **Destination domain:** As mentioned before, we regard the GET request referring to the text/html resource as a website visit. We filter the HTTP Get packets with the Accept request-header field and collect different values of the top-level domain names. Suppose that there are D different top-level domain names in total, we can write them as a |D|-dimensional vector. And the *i-th*

component of the vector means the access frequency of the corresponding domain.

- **Access frequency of each domain:** We believe that the access frequency can't be simply equated with the number of connections between the client and the website. Because the latter is affected by the web server and parameter settings of the user's browser. In this paper, we assume 10 minutes is an epoch. If one interaction between the user and the website lasts within one epoch, we set the access frequency of the corresponding website to 1. And the value is plus 1 for every extra epoch. At last, we count the total numbers for each domain by aggregating all corresponding interactions in a day.
- **Geographic location of the user:** By using the open geolocation API, we can get the approximate geographic location of the user based on his IP address. This feature is useful in the scenario of setting up a free VPN or hotspot to attract traffic.
- **Mail domain:** The email service is so widespread that many service providers offer free mailboxes to their users. Besides, most companies and universities also offer the internal mail service. We believe that a user may have multiple email mailboxes and select one based on the different communication purpose every time. For example, a user uses the company mailbox to report the work progress to his leader. When he wants to greet his friend, he may select some private mailbox according his preferences. So we extract the combination of different email domains for each user.
- **Search engine:** Nowadays varieties of search engines are available and have their own advantages respectively. We believe that the user may make a choice based on the content type he wants to search. For instance, he likes to use Google Scholar to find scholarly articles. When he wants to search some Chinese keywords, he prefers to use Baidu. We extract the combination of search engines used by the user in a day by performing signature matching.
- **User-Agent strings of browsers:** We can get the details about the user's system version, browser version and the language preference by analyzing the URL strings. If a user has installed several operating systems and browsers, we even can identify him just with the User-Agent string.
- **Usage frequency of IM:** We count the number of connections established by QQ as the usage frequency of IM.
- **Usage frequency of P2P:** We count the number of connections established by the eDonkey as the usage frequency of P2P.

In summary, a user's traffic in one day can be aggregated and represented as a vector $\vec{x}$:

$$\vec{x} = < x_1^{f_{x_1}}, x_2^{f_{x_2}} \ldots x_n^{f_{x_n}} > \quad (1)$$

In the above, x_i represents the *i-th* feature, and f_{xi} represents the corresponding access frequency.

D. Classification

It is very important to select an appropriate classifier for the classification problem. Many researchers apply the support vector machine which is well-known for its high performance in terms of the classification accuracy. However, the high dimensional features have a devastating impact on the effectiveness of SVM [18]. As a result, we also apply the Multinomial Naive Bayes Classifier [19] to identify users like recent works.

The Multinomial Naive Bayes Classifier is a special model which implements the Naïve Bayes algorithm for multinomially distributed data. It also assumes that each feature is independent from one another given the class label. The Multinomial Naive Bayes Classifier estimates the probability of a vector $\vec{x} = < x_1^{f_{x_1}}, x_2^{f_{x_2}} \ldots x_n^{f_{x_n}} >$ belonging to a particular class C_i as:

$$\begin{aligned} p_{C_i}(x) &= p(C_i|x) \\ &= \frac{p(C_i)p(x|C_i)}{p(x)} \propto p(c_i)\prod_{j=1}^{n} p(x_j|C_i)^{f_{x_j}} \end{aligned} \quad (2)$$

IV. EVALUATION

In this section, we mainly describe the experiment methodology used in the real world environment to evaluate the performance. According to the previous works, we use the detection rate (accuracy) as evaluation criteria.

A. Data Collection

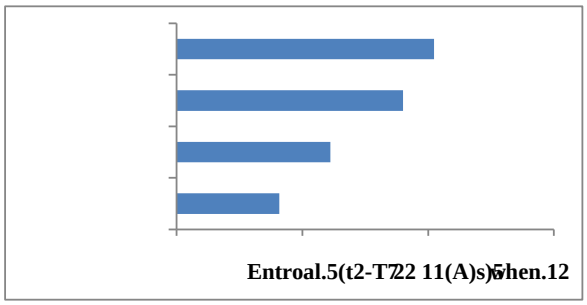
In order to evaluate the feasibility and scalability of our method, we collect large amounts of traffic in a real world setting. Considering the high speed network traffic, we capture packets with PFQ, which is highly optimized for the multi-core architecture, as well as for network devices equipped with multiple hardware queues. At last, we construct two datasets of different scale, including the *header* dataset and the *payload* dataset. These two datasets only contain IPV4 traffic.

The *header* dataset is generated by mirroring all traffic passing through the egress switch of our laboratory to a sniffing computer for 5 weeks. 55 users' traffic is captured but only 66 bytes of each packet are recorded. The *payload* dataset is obtained by sniffing all incoming and outgoing traffic of our college in cooperation with the network center of our university between May 19, 2015 and June 11, 2015. And the packets are completely recorded, including the payload of the transport layer. In total, we get an over 25TB dataset which contains 1200 individuals' traffic.

B. Experiment Results

Since the *header* dataset does not contain application data, we only extract destination domains and email domains as features to identify 55 users. In the classification phase, we use Weka [18] toolkit to implement the Multinomial Naive Bayes Classifier and all instances are labeled with the corresponding Mac address. The *header* dataset is divided into two parts: the training set and the testing set. The training set contains traffic of the first, third and fifth weeks, and the testing set contains the rest. There is no intersection between the training set and the testing set. When we identify users on the destination domains, the detection rate is up to 82.6%. By considering the email domains, we increase the detection rate to 100%. The satisfying results have preliminarily demonstrated the effectiveness of the behavior-based tracking attack. However, there is a significant deficiency that the scale of the *header* dataset is too small. To address this problem, we implement more experiments on the *payload* dataset to evaluate the scalability of our approach.

As described above, we collected 1200 individuals' traffic by sniffing all incoming and outgoing traffic of our college. But the total number of staff and students in our collage doesn't exceed 500. The excess part may be experiment machines or movable equipment.



9b)e F05 1 Entroal.5(t2-T722 11(A)s)6hen.12

To maintain a high data quality, we select 509 active individuals according to the total number of access frequencies which is more than 100. And the number of different domains visited by these active users is up to 25639. It means that the performance of our classifier will be seriously impacted by the high dimensional features. To evade this issue, we simply try selecting a subset of domains as an alternative. Fig. 2 illustrates the detection rates of 10-fold cross validation when we use top *K* most popular domains. As we can see from the Fig. 2, when *K* is varied from 10 to 50, there is an increase by 30%. But when *K* exceeds 800, there is a slightly drop in accuracy. We can conclude that redundant features will confuse the classifier and cause a decrease in accuracy. In subsequent experiments, we use top 800 domains as features.

To evaluate other features, we introduce the concepts of "surprisal" and entropy like Eckersley' work [22]:

$$S(i_{n,f}) = -\log_2(P(i_{n,f})) \tag{3}$$

$$H(I_f) = -\sum_{n=1}^{N} P(i_{n,f}) * \log_2(P(i_{n,f})) \tag{4}$$

In the above, i represents an instance and f represents a feature. $P(i_{n,f})$ is a discrete probability density function of the instance $i_{n,f}$. The surprisal S represents the amount of information associated with the value of a discrete instance, which is measured here in units of bits. $H(I_f)$ represents the entropy of the feature f in the overall sample space. Fig .3 illustrates the entropy of several features which are computed according to the above formulas. Among them, the Mac Address is used as a control. From the Fig .3, we can see that the User-Agent is a discriminational feature, especially when the user has multiple browsers.

Since the power of features cannot easily be predicted, we empirically tested a set of features. The results are illustrated by Fig. 4, which provides a visual impression of our recognition results and gives an overview of the features impact on the detection rate. The "Basic Approach" means that we extract destination domains, frequencies, geographic locations, usage frequencies of IM and P2P to create profiles. The training set contains traffic of the first 15 days, and the rest constitutes the testing set. When all features are used, we can identify average 78.93% instances correctly.

In order to future improve the accuracy of our approach, we use the term frequency (TF) and inverse document frequency (IDF) to process the raw data, which are the most common weighting methods in information retrieval and data Mining[24]. The formulas (5) – (7) describe the different transformations of the frequency values.

$$f_x^{tf} = \log(1 + f_x) \tag{5}$$

$$f_x^{idf} = \log(\frac{N}{n_x + 1}) \tag{6}$$

$$f_x^{tf-idf} = f_x^{tf} * f_x^{idf} \tag{7}$$

Fig. 5 illustrates the detection rates of three patterns with *G* guesses. As the name suggests, the pattern called "Raw" means that we identify users on the raw data. The "TF" and "TF-IDF" patterns use TF transformation and TF-IDF transformation to the raw data before the classification respectively. The value of *G* is increased from 1 to 10, which represents the size of the ranking list of candidates. In other words, if the real identity of a user is in the ranking list of candidates, we consider the classifier successful. As we can see from the figure, the average accuracy of our behavior-based tracking attack is raised to 85.61% when we use the TF-IDF transformation. And when *G* is set to 10, the accuracy exceeds 90%. All these results can prove the effectiveness and scalability of our approach in the large-scale scenario.

V. COUNTERMEASURES

In this section, we first discuss the effectiveness and feasibility of several countermeasures proposed by related work. Then we introduce a possible solution to resist the behavior-based tracking attack.

A. Discuss

Herrmann et al [14] carry out the behavior-based tracking attack by using DNS queries and propose 4 countermeasures to mitigate the effectiveness, including using anonymizers, changing IP address frequently, DNS cache, and range queries. We mainly pay attention to the first two general methods.

- **Using anonymizers:** Herrmann et al believe that the anonymizers like Tor can hide the user's communication patterns. Indeed, Tor enh(rst)8.3(a09 Tm[(c)-8.9(h)-1(a)-8.9(ngai)0.7)] TJET.3(a)-8.3(a(h)-19 Tc0.3376 Tw9.9

reaches a certain level, we think the real profile of Class A is hid.

VI. CONCLUSIONS

Recent works of the behavior-based tracking techniques always perform experiments on the small-scale datasets, which can't fully prove the feasibility of these methods. To address this problem, we propose a novel behavior-based tracking attack and extract features ranging from lower layer network packets to high level application related traffic. We execute experiments on a real world traffic dataset contains 509 active users. When all features are used, we can identify 85.61% instances correctly. Based on the results, we believe that the behavior-based tracking attack is feasible in the large-scale scenarios and should be carefully considered without contempt. For future research, we intend to look for more behavior features and construct a Spark Streaming framework to analysis the real time traffic.

ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China under Grants No. 61272054, No. 61320106007, No. 61502100, No. 61532013, No. 61572130, National High Technology Research and Development Program of China under Grants No.2013AA013503, Natural Science Foundation of Jiangsu Province under Grants No. BK20150637, State Grid Corporation of China - Research on Key Techniques of Integrated Security Audit for Massive Information Interaction on Intra/Extra Information Network Border, the Fundamental Research Funds for the Central Universities (No. 2242014R30010), Jiangsu Provincial Key Laboratory of Network and Information Security under Grants No. BM2003201, and Key Laboratory of Computer Network and Information Integration of Ministry of Education of China under Grants No. 93K-9.

REFERENCES

[1] JR Mayer and JC Mitchell, "Third-party web tracking: Policy and technology," in Proceedings of the 33th IEEE Symposium on Security and Privacy(IEEE S&P), SF, CA, USA, 2012.

[2] M Pusara and C Brodley, "User re-authentication via mouse movements," in Proceedings of CCS Workshop on Visualization and data mining for computer security, Washington, DC, USA, 2004.

[3] H Gamboa and A Fred, "A behavioral biometric system based on human computer interaction," in Proceedings of SPIE: Biometric Technology for Human Identification, Glasgow, Scotland, 2004.

[4] N Zheng, A Paloski and H Wang, "An efficient user verification system viamouse movements," in Proceedings of 18th ACM Conference on Computer and Communications Security, Chicago, Illinois, USA, 2011.

[5] F Monrose and A Rubin, "Authentication via keystroke dynamics," In Proceedings of the ACM Conference on Computer and Communications Security, Zurich, Switzerland, 1997.

[6] S Douhou and J Magnus, "The reliability of user authentication through keystroke dynamics," Statistica Neerlandica, vol4, pp. 432-449, 2009.

[7] J Ilonen, "Keystroke dynamics," http://www.it.lut.fi/kurssit/03-04/010970000/seminars/Ilonen.pdf.

[8] B Padmanabhan and YC Yang, "Clickprints on the web: Are there signatures in web browsing data," http://knowledge.wharton.upenn.edu/papers/1323.pdf, 2006.

[9] YC Yang, "Web user behavioral profiling for user identification," Decision Support Systems, vol 3, pp. 261-271, 2010.

[10] OD Vel, A Anderson, M Corney and G Mohay, "Mining e-mail content for author identification forensics," SIGMOD Record, vol 4, pp. 55-64aa9i[( )] TJETBT-0