International Journal of Security and Its Applications
Vol. 11, No. 7 (2017), pp.25-42
http://dx.doi.org/10.14257/ijsia.2017.11.7.03

Attacking HTTPS Secure Search Service through Correlation Analysis of HTTP Webpages Accessed

Qian Liping[1] and Wang Lidong[1,2*]

[1]*College of Electrical and Information Engineering, Beijing University of Civil Engineering and Architecture*
[2]*Information Security Laboratory, CNCERT/CC*
[1]*qianliping@bucea.edu.cn*, [2]*wld@cert.org.cn*

Abstract

It is very common for Internet users to query a search engine when retrieving web information. Sensitive data about search engine user's intentions or behavior can be inferred from his query phrases and the webpages he visits subsequently. In order to protect contents of communications from being eavesdropped, a search engine can adopt HTTPS-by-default to provide bidirectional encryption to protect its users' privacy. Since the majority of webpages indexed in search engine's results pages are still on HTTP-enabled websites and the contents of these webpages can be observed by attackers once the user click on the indexed web-links. We propose a novel approach for attacking secure search through correlating analysis of encrypted search with unencrypted webpages the user visits subsequently. We show that a simple weighted TF-DF mechanism is sufficient for selecting guessing phrase candidates. Imitating search engine users, by querying these candidates and enumerating webpages indexed in results pages, we can hit the definite query phrases and meanwhile reconstruct user's web-surfing trails through DNS-based URLs comparison and flow feature statistics-based network traffic analysis. In the experiment including 180 Chinese and English search phrases, we achieved 67.78% hit rate at first guess and 96.11% hit rate within three guesses. Our empirical research shows that HTTPS traffic can be correlated and de-anonymized through HTTP traffic and secure search of search engine is not always secure unless HTTPS-by-default enabled everywhere.

Keywords: *feature selection; Internet security; online privacy; secure search service; traffic identification; user activity identification*

1. Introduction

With the rapid development of Internet and its applications, our everyday activities have relied on them heavily. As more reports about information interception by outside hackers, deviant behavior of insiders within the same organization, and mass surveillance on the infrastructure by government having been published, users' concerns over breaches of information security and individual privacy continue to mount [1]. Some Internet service companies have taken up innovative technologies to protect the privacy of their users. Users also begin employing a variety of techniques, including proxy-based schemes, anonymity tools and encrypted tunnels, to safe their own communication security and data privacy. Even though the effectiveness of security measures to protect sensitive information is increasing, people remain susceptible to manipulation [2]. User's sensitive information could still be leaked even when privacy control mechanisms have been deployed and privacy rules are properly configured [3, 4].

Hyper-Text Transfer Protocol over Secure Socket Layer (HTTPS) is a mechanism for secure web browsing. Relying on Secure Socket Layer/Transport Layer Security (SSL/TLS) encryption, HTTPS protects communications from being eavesdropped,

ISSN: 1738-9976 IJSIA

intercepted or hijacked. After the revelation of widespread data collection by the US National Security Agency (NSA) by Edward Snowden in 2013, both Internet Engineering Task Force (IETF) and Internet Architecture Board (IAB) encouraged websites adopting HTTPS and encrypted communications by default. However, for a wide variety of reasons, ubiquitous encryption hasn't taken off. In January 2014, Naylor found that 27.6% of the Alexa top 500 websites completed a TLS handshake, while only 7.2% had HTTPS-by-default enabled [5]. By the end of 2015, according to Google's Transparency Report, over 75% of requests to Google's servers are using encrypted connections. Google also tracked the HTTPS state of the Top 100 non-Google sites on the Internet. These sites account for approximately 25% of all website traffic worldwide. Google found that 34 sites works on modern HTTPS, but only 22 runs modern HTTPS by default (https://www.google.com/transparencyreport/https/?hl=en).

Search engine plays an irreplaceable role in web information organizing and accessing. It has become the portal for Internet users to obtain information resources. According to STATISTICS BRAIN, total web searches per month in the U.S. are 11 billion. In order to protect communication from being eavesdropped, some search engines such as Google search and Chinese search engine giant Baidu adopt HTTPS by default so that data transferred between the search engine and their users are encrypted. If search engine enables secure search, attackers cannot successfully peep users' query phrases as well as the returned results pages. While due to a low proportion websites enabling HTTPS, most indexed webpages indexed in the returned results pages can only be accessed through plaintext Hyper-Text Transfer Protocol (HTTP). Once these indexed links are clicked, attackers can monitor the URLs and the returned webpage contents. Due to the strong correlations among the content of these pages, the query phrases might be guessed out. We propose an approach for attacking secure search and demonstrate that the query phrase can be reversed with high probability.

The purpose of this paper is to demonstrate that HTTPS traffic can be correlated and de-anonymized through HTTP traffic. Our research shows clearly that if pervasive encryption has not effected on Internet, privacy and security of web users' activity might be compromised. Our contributions include:

1) We propose an approach for attacking encrypted search phrases of secure search service and we show that a simple weighted Term Frequency-Document Frequency (TF-DF) method for feature selection on document set of goal-oriented content. Experimental result demonstrated that our attacking schema could efficiently reverse the secured search phrases. It hit the search phrases at first guess with probability 67.78%, and achieved 96.116% hit rate in the first three guesses.

2) We present a DNS-based URLs comparison method and a flow feature statistics-based network traffic analysis method to reconstruct user's web-surfing trails.

3) By demonstrating the technique to attack secure searching service, we accentuate the importance of universal encryption to protect Internet communications and encourage searching service providers to crawl, index and show users more HTTPS webpages in search results.

The remainder of this paper is organized as follows. In Section 2, we review the related work in encrypted webpage identification and analysis. Section 3 introduces the attacking scenarios and presents our attacking method against secure search. Section 4 presents experimental evaluation of our attacking method and discusses several possible influencing factors. Section 5 concludes our work.

2. Related Work

Related work includes encrypted webpage identifying, website fingerprinting, user activity inferring, network traffic classifying and the resistant countermeasures.

PPI makes use of fingerprint derivation method for encrypted Web-browsing traffic, but it needs cross reference between the encrypted HTTPS webpages and the plaintext HTTP webpages [6]. TCPI is a framework using Calculation of Timing Characteristics (CTC) algorithm to extract the timing characteristics as identification signature of encrypted page from time feature of traffic [7]. Brad Miller *et al*. used Bag-of-Gaussians and Hidden Markov Model to identify HTTPS encrypted webpages of 10 websites [8].

Alfredo Pironti *et al*. presented an HTTPS attack for user activity inferring that reveals the precise identities of users by combining public social network profiles with TLS traffic analysis [9]. Computer forensic investigation sometimes needs to reconstruct user-browser interactions from network traces. ReSurf constructs the referrer graph of HTTP requests [10]. ClickMiner is basing on ReSurf's approach by matching attributions of webpages and the URL clicked, but it also focuses on HTTP requests [11]. Hviz is an interactive tool for aggregating and visualizing the timeline of HTTP web browsing activity. It also supports HTTPS traffic recorded by a SSL/TLS man-in-the-middle proxy server [12]. Mauro Conti *et al*. proposed a framework to infer the particular actions user executed on some Android applications via network traffic analysis [13]. Yaoqi Jia *et al*. presented a systematic study of browser cache poisoning attacks, wherein a network attacker performs a one-time Man-In-The-Middle attack on a user's HTTPS session, and substitutes cached resources with malicious ones [14].

Shuo Chen *et al*. inferred user activity on a website. They targeted sensitive websites such as online tax and online health [15]. Marc Juaez *et al*. claimed that certain variables, for example, user's browsing habits, differences in location and version of Tor Browser Bundle, had a significant impact on the efficacy of website fingerprinting attack against Tor [16]. Maciej Korzynski *et al*. used first-order homogeneous Markov chains to fingerprint parameters of chosen applications to detect abnormal SSL/TLS sessions [17].

A lot of work has been done on traffic classification and protocol identification using different techniques, especially machine learning methods on statistical features of traffic. To name a few: Katerina Goseva-Postojanova *et al*. used supervised machine-learning methods on 43 features to classify attacker activities to two classes: vulnerability scans and attacks [18]. By dividing the training set into clusters, forming sub-classifiers and integrating classifiers, Cluster-Min-Max (CMM) method effectively reduces the false positive rate of traffic classification, their experiments showed its effectiveness for large-scale network [19]. Set-Based Constrained K-Means (SBCK) algorithm is a constrained variant of K-Means. It makes decisions with consideration of some background knowledge in addition to traffic statistics [20]. Zhang Luoshi *et al*. evaluated different Machine Learning techniques for traffic classification under different network environments. They found that the identification accuracy was affected more by network scale and network environment, but less by machine learning techniques or statistical features [21]. Meanwhile some resistant methods have been proposed to protect privacy against traffic analysis. Herd is an anonymity network providing VoIP caller/callee anonymity [22]. Rook embeds the targeted data in the network traffic of an online game to defeat deep-packet inspection and traffic shape analysis [23]. Marionette is a programmable network traffic obfuscation system capable of emulating many existing obfuscation systems [24].

Most of the aforementioned works commonly verified their method with a limited set of encrypted web pages, or exploited protocol vulnerabilities for effective analysis on encrypted traffic. Our work focuses on correlated attack on secure search. We reverse query phrases submitted to secure search engine through HTTP webpages contents visited, and further infer user's click sequences and query intent through combing Domain Name Service-based (DNS-based) URLs comparison and statistic-based flow traffic analysis.

3. Attack Model

Secure search service of search engine is based on HTTPS protocol. Search engines service providers hope high on secure search service for protecting user privacy from being eavesdropped. We illustrate two scenarios that an attacker attacks secure search service through correlating analysis of encrypted search with unencrypted webpages visited.

3.1. Secure search via HTTPS

With SSL/TLS, HTTPS provides confidentiality, integrity and one-way or two-way non-repudiation authentication. SSL was originally put forward by Netscape Communications Corp. It lies between transport layer and application layer of TCP/IP protocol suite and provides secure transport services for application layer. The last version of SSL is 3.0. TLS is designed by IETF in RFC 5246 as an upgrade version of SSL and the latest version is TLS 1.2. TLS 1.3 is still a working draft. TLS implements communicating security through encryption, provides data integrity through MAC mechanism, and achieves identity authentication through digital certificates. Therefore, TLS introduces extensible cipher suites to support encryption, identity authentication, message authentication code, key exchange, key derivation, etc. TLS protocol comprises a series of sub-protocols. The most important includes the Record protocol, the Handshake protocol, the Alert protocol, the ChangeCipherSpec protocol and Application Data protocol. Figure 1 depicts the architecture of TCP/IP protocol suite with TLS protocol.

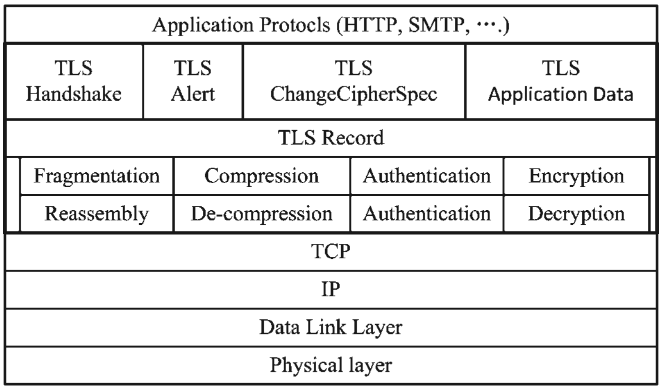

Figure 1. TCP/IP Protocol Suite with TLS

HTTPS-based web browsing is the most popular Internet application secured by SSL/TLS protocol. HTTPS is HTTP communication encrypted and authenticated by SSL/TLS. Secure search service consists of two phases. First, user client communicates with the search engine with HTTPS, submits a query phrase and receives the results pages, as depicted in Figure 2. Then, user client communicates with the objective website with HTTP or HTTPS, depending on whether the website supports SSL/TLS, and retrieves the webpage, as depicted in Figure 3(a) and 3(b) respectively.

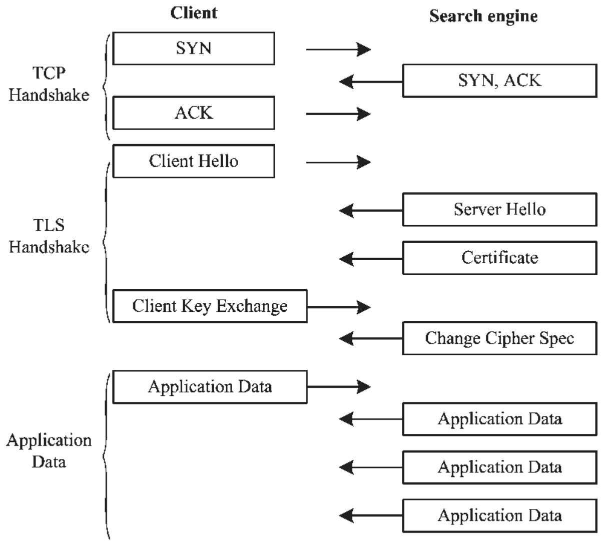

Figure 2. Interaction of an HTTPS-enabled Search

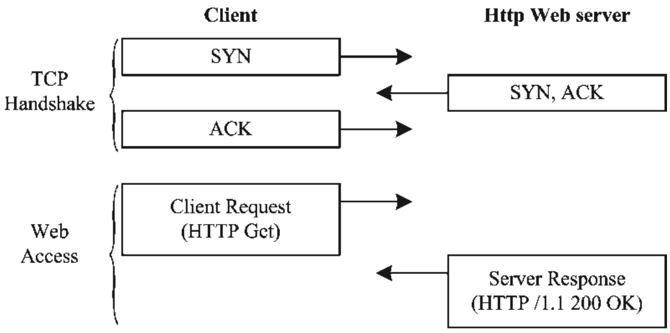

Figure 3(a). Interaction of HTTP Website Browsing

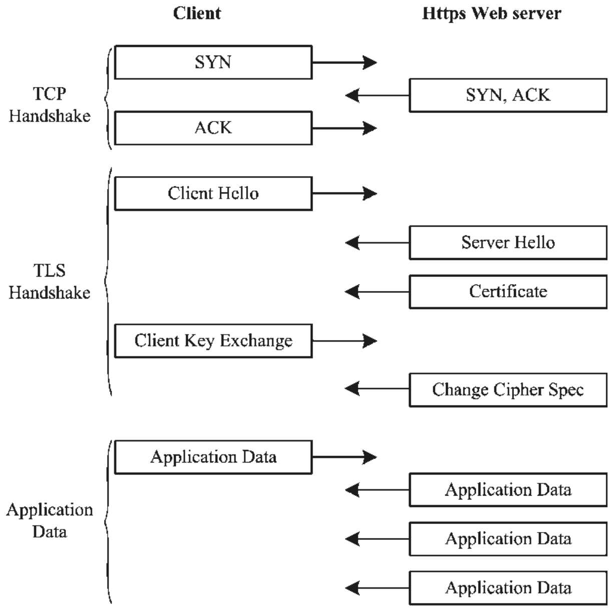

Figure 3(b). Interaction of HTTPS Website Browsing

3.2. Attack Scenarios

Figure4 illustrates two scenarios that an attacker wiretaps a user's communication. The first appears in a situation that the attacker, a disgruntled inside colleague or hidden outside intruder, is on the same subnet as the user where he can passively eavesdrop bi-directional network traffic from and to the user's computer by manipulating some configuration of Local Area Network (LAN) switch. The second situation appears where the attacker, an employee of the network infrastructure provider or an intruder, connects at some point at the gateway of the backbone network where he can collect bi-directional or uni-directional network traffic data. The possible positions in the network where the attacker can sit to monitor the user (u) are marked out as a in Figure 4. Due to asymmetric routing in backbone network, the attacker probably can only observe uplink (forward) or downlink (backward) direction of traffic data. Backward traffic from search engine to user client is sufficient for our attack model.

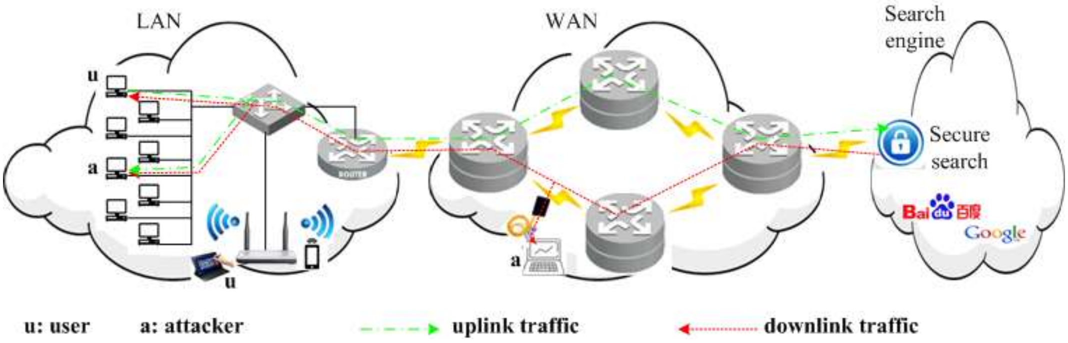

Figure 4. Attacking Scenarios

To be different with web surfing, users of search engine are always with definite purpose, i.e. submitting a query phrase, obtaining the searching results pages and retrieving target information through URL links indexed in results pages. No matter how search engines refine, index and rank webpages, they should return those webpages more relevant, more popular, or more authoritative to users. Prevailing web search engines like Google Search and Baidu Search have deployed HTTPS secure search to protect user privacy. Attackers cannot tap the search query submitted by users and the returned results pages from search engine. Unfortunately, most websites still have not had a secure implementation of HTTPS yet. If the search engine user accessed several webpages on HTTP-enabled websites, the URLs and/or contents of these webpages might be captured by the attacker and subsequent aggregated analysis on the contents enables the attacker to deduce the encrypted query phrases. Figure 5 depicts the process the attacker inferring the encrypted search phrases, where EP denotes encrypted phrase, RP encrypted results pages, EU encrypted URLs, EC encrypted webpage content, PU plain-text URL, and PC plain-text webpage content. From plain-text webpage contents, the attacker can guess the search phrases by a phrase feature selection method. With the guessed phrase, attackers can further reconstruct user's web-surfing trail through computing traffic similarity between user's flow feature vector and his own.

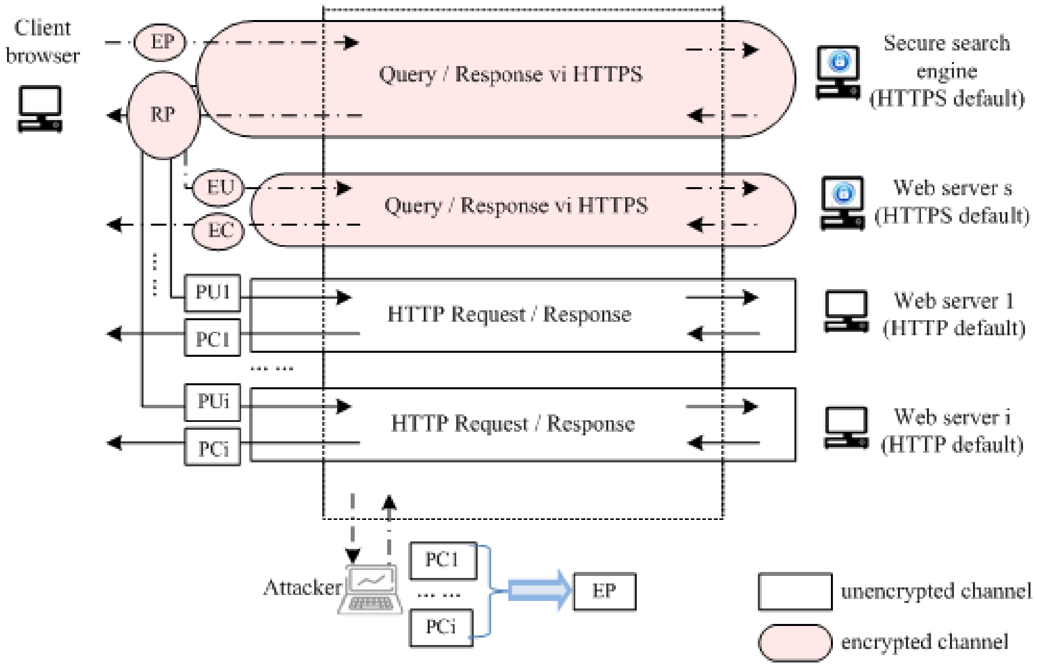
Client browser
EP
RP
Query / Response vi HTTPS
Secure search engine (HTTPS default)
EU
EC
Query / Response vi HTTPS
Web server s (HTTPS default)
PU1
PC1
HTTP Request / Response
Web server 1 (HTTP default)
PUi
PCi
HTTP Request / Response
Web server i (HTTP default)
Attacker
PC1
PCi
EP
unencrypted channel
encrypted channel

Figure 5. Attacking Models

3.3. Attack Method

Suppose k_u is the phrase queried by search engine user u, P_{all} is the set of URLs returned in the results pages for this query. User u selectively clicks a set of URLs, $P_u = \{p_{u1}, p_{u2}, \cdots, p_{ul}, \cdots\}$. P_u is an ordered set, that is, P_{ui} listed before P_{uj} if u visited u_i before u_j. Let $P_{eu} \subseteq P_u$, $P_{eu} = \{p_{eu1}, p_{eu2}, \cdots, p_{eui}, \cdots\}$ is the set of webpages accessed with HTTPS, and $P_{tu} \subseteq P_u$, $P_{tu} = \{p_{tu1}, p_{tu2}, \cdots, p_{tuj}, \cdots\}$ the set of webpages accessed with HTTP. Let d_i be the text content of p_{tui}, $D = \{d_1, d_2, \cdots, d_j, \cdots\}$, $|D| = n$. We imitate an attacker attacking secure search by guessing out the query phrases submitted by user u. Feature selection methods are employed to rank guessing phrase candidates. Features here are usually keywords or terms. Typical feature selection methods include frequency-based or probabilistic-based statistics, Chi-Square Test, information entropy or information gain-based, Principal Component Analysis and n-gram lexical co-occurrence. The most widely used and effective method is Term Frequency/Inverse Document Frequency (TF/IDF). There are a lot of variants of TF/IDF weighting methods. Let $tf_{t,d}$ be the frequency of term t in document d, n the total number of documents in the corpus, n_t the number of documents including term t, we choose the common TF/IDF method as in (2) to compute weight $WTI_{t,d}$ of term t in document d. Due to the strong correlation among the contents of multiple webpages accessed by search engine users, we proposed a weighted TF/DF method to compute the weight of term t in document d as in (4), where α and β are weight coefficients of TF and DF respectively, $\alpha + \beta = 1$.

$$idf_{t,d} = \log(n / n_t + 0.01) \qquad (1)$$

$$WTI_{t,d} = tf_{t,d} \times idf_{t,d} \Big/ \sqrt{\sum_{t_i \in d} \left[tf_{t_i,d} \times idf_{t_i,d} \right]^2} \tag{2}$$

$$df_{t,d} = n_t / n \tag{3}$$

$$WTD_{t,d} = (\alpha \times tf_{t,d} + \beta \times df_{t,d}) \Big/ \sqrt{\sum_{t_i \in d} \left[(\alpha \times tf_{t_i,d} + \beta \times df_{t_i,d}) \right]^2} \tag{4}$$

Phrases are ranked according to the value of $WTI_{t,d}$ and the Top m phrases are selected as candidate guessing phrases. Suppose $k_a \in \{k_{g1}, k_{g2}, \cdots, k_{gm}\}$ are all candidate guessing phrases of attacker a. For m, all URLs in the returned results pages are $P_{all}^{'} = \{p_{a1}, p_{a2}, \cdots, p_{al}, \cdots\}$. $P_{all}^{'}$ is an ordered set, that is, p_{ai} listed before p_{aj} if m visited a_i before a_j. Let $P_{ea} = \{p_{ea1}, p_{ea2}, \cdots, p_{eai}, \cdots\}$ is the set of webpages accessed with HTTPS, and $P_{ta} = \{p_{ta1}, p_{ta2}, \cdots, p_{taj}, \cdots\}$ the set of webpages accessed with HTTP. If $P_{tu} \subseteq P_t$, then we regard $k_a = k_u$.

For Chinese search phrase k_u, the overall idea of our attack is as follows:

1) Imitate search engine user querying search engine with k_u.

2) Pick out n most related links from the first page returned by the search engine.

3) Visit the n URLs one by one and save the content of each webpage as a text document.

4) Word segment the documents.

5) Compute TF/DF weighting of all phrases according to formula (4).

6) Sort these phrases by TF/DF weight in descending order.

7) Verify the guessing phrases.

8) Analysis network traffic for reconstructing web-surfing trail.

3.3. Reconstruct user's web-surfing activity

The reconstruction of web-surfing activity from network traffic trace is not only an appealing option for attacker, but also very useful for forensic analysis system to record browsing sessions and to reconstruct web security incidents. With the guessed phrases, the attacker can reconstruct the user's click trail by redoing the query and analyzing the network traffic. Considering the fact that users visit only part of URL links indexed in search results pages, we put forward two similarity-based options to hit the target, one is domain name similarity-based, and the other is flow feature similarity-based.

3.3.1.DNS-based URLs Comparison: Let $U_u = \{u_{u1}, u_{u2}, \cdots, u_{um}\}$ be the user's visiting URL sequence, $U_a = \{u_{a1}, u_{a2}, \cdots, u_{am}\}$ the attacker's guessing URLs. If the attacker can monitor bi-directional network traffic data, then for each URL $u_{ui} \in U_u$, the corresponding $u_{ai} \in U_a$ is determined as below:

1) If u_{ui} is an HTTP URL, then u_{ai} can be observed directly through plain-text network traffic monitoring.

2) If u_{ui} is an HTTPS URL, then u_{ai} cannot be parsed out from the encrypted network traffic, but the hostname, u_{hi}, of the web server dispensing webpage u_{ui} can be captured through DNS query process prior to accessing u_{ui} through HTTPS protocol. The attacker redoes the query to the secure search engine with the guessed keywords with an HTTPS client, parses the returned results pages and retrieves all URLs (U_{all}) related to u_{hi}. Let $U_{all} = \{u_{x1}, u_{x2}, \cdots, u_{xk}\}$, we define u_{ai} as (5).

$$u_{ai} = \begin{cases} u_{x1} & \text{if } |U_{all}| = 1 \\ u_{xi} & \text{if } |U_{all}| \neq 1 \text{and } L_u(Pu_{xi}) = L_u(Pu_{ui}) \text{ and } L_d(Pu_{xi}) = L_d(Pu_{ui}) \end{cases} \tag{5}$$

Where $L_u(Pu_y)$ is the returned webpage byte length of outgoing traffic and $L_d(Pu_y)$ the byte length of incoming traffic when accessing url u_y.

3.3.2.Flow Feature Statistics-based Network Traffic Analysis: We use a statistical feature-based approach for evaluating the similarity of two webpages. It applies to both plain-text HTTP and/or encrypted HTTPS URL identification with bi-directional or uni-directional network traffic. When we access a webpage, the single webpage surfing might generate multiple network flows. Each flow can be distinguished with a 5-tuple consisting of source ip-address, destination ip-address, source port, destination port and transport protocol. Suppose the webpage accessed is p , the flow set generated from downlink traffic (i.e. traffic data from search engine to user client) is $F_p = \{F_1, F_2, \ldots, F_s\}$. For each flow F_i, we consider the following statistical features: packet count (f_1), total size in bytes (f_2), number of concurrent sessions (f_3), packet rate in packets/sec (f_4), mean packets sizes in bytes (f_5), standard deviation of packets sizes in bytes (f_6), mean packet interval time in milliseconds (f_7), and standard deviation of packet interval time in milliseconds (f_8). Then a flow can be modeled as a feature vector, written as $F_i = \langle f_{i1}, f_{i2}, \ldots, f_{i8} \rangle$. We define the distance between two flows, F_j and F_j , as (6).

$$D(F_i, F_j) = \sum\nolimits_k \left(|f_{ik} - f_{jk}| \big/ |f_{ik} + f_{jk}| \right) \big/ 8 \tag{6}$$

The similarity between F_j and F_j is defined as (7).

$$Sim(F_i, F_j) = 1 - D(F_i, F_j) \tag{7}$$

Suppose user u visits webpage p_u which generates flow set $FS_{up} = \{F_{up1}, F_{up2}, \ldots, F_{ups}\}$, $F_{upi} = \langle f_{upi1}, f_{upi2}, \ldots, f_{upi8} \rangle$. We use 1-Nearest Neighbor Classifier to determine the most similar webpage as p_u. Attacker a queries search engine with the hit query phrase and visits webpage p_u and retrieves all URLs indexed in the returned results pages, denoted by U_{all} . For each $u_{ai} \in U_{all}$, attacker a visits its webpage p_{ai} , generates a corresponding flow set $FS_{ap} = \{F_{ap1}, F_{ap2}, \ldots, F_{apt}\}$, $F_{api} = \langle f_{api1}, f_{api2}, \ldots, f_{api8} \rangle$. Let γ is an empirical threshold value, In conditions of $s = t$, for each $F_{upi} \in FS_{up}$, $\exists F_{apj} \in FS_{ap}$, if

$$Sim(F_{upi}, F_{apj}) < \gamma \tag{8}$$

then we hold that p_u and p_{ai} are the same webpage.

4. Experiment and Analysis

A report from Statista pointed out that Google has dominated the global search engine market, maintaining an 89.44 percent market share as of the first quarter of 2016 (http://www.statista.com/statistics/216573/worldwide-market-share-of-search-engines/). But Google is not a big player in Chinese search engine market, according to China Internet Watch (http://www.chinainternetwatch.com/category/search-engine/), Incitezchina said that Baidu ranked the first with 92.1% penetration rate (http://www.slideshare.net/incitezchina/china-search-engine-market-overview-2015). Baidu enabled secure search at the end of 2014. We target Chinese search engine giant Baidu (https://www.baidu.com) to verify our attacking method.

4.1. Dataset

We choose 180 phrases including 140 Chinese phrases and 40 English phrases. These phrases consist of buzz words, product names, celebrity names and a wide variety of other phrases. In order to test discriminative power of our approach on similar concepts, we specifically choose seven categories, 4 nouns of each category, including car, finance, celebrity name, soccer, university name, home appliance and air pollution. English translations of the 28 nouns are also included in the chosen 40 English phrases.

Due to limitations of space, we only list the 28 categorized Chinese search phrases in Table 1.The Chinese phonetic form and corresponding English translation are also given in the table. Some phrases are in abbreviated form and some are noun compounds. Table 2 lists several examples of abbreviations and their root.

Table 1. Information about Search Phrases

id	Search phrases	Chinese phonetic form	English translation
1	新能源汽车	xīn-néng-yúan-qì-chē	New energy vehicle
2	宝马	bǎo-mǎ	BMW
3	奔驰	bēn-chí	Benz
4	奥迪	ào-dí	Audi
5	股市	gǔ-shì	stock market
6	理财	lǐ-cái	financial planning
7	国债	guó-zhài	treasury bond
8	信用卡	xìn-yòng-kǎ	credit card
9	周杰伦	zhōu-jié-lún	Jay CHOU
10	范冰冰	fàn-bīng-bīng	Bingbing FAN
11	谢霆锋	xiè-tíng-fēng	Nicholas TSE
12	汪峰	wāng-fēng	Feng WANG

13	国足	guó-zú	National soccer team
14	英超	yīng-chāo	English Premier League
15	意甲	yì-jiǎ	Italian Serie A
16	世界杯	shì-jiè-bēi	World Cup
17	北建大	běi-jiàn-dà	Beijing University of Civil Engineering and Architecture
18	清华	qīng-huá	Tsinghua
18	哈工大	hā-gōng-dà	Harbin Institute of Technology
19	北大	běi-dà	Peiking University
20	空气净化器	kōng-qì-jìng-huà-qì	air purifier
21	空调	kōng-tiáo	air conditioner
22	冰箱	bīng-xiāng	refrigerator
23	洗衣机	xǐ-yī-jī	washing machine
24	雾霾	wù-mái	fog and haze
25	口罩	kǒu-zhào	mask
26	气管炎	qì-guǎn-yán	tracheitis
27	环保	huán-bǎo	environmental protection
28	股市	xīn-néng-yúan-qì-chē	New energy vehicle

Table 2. Abbreviated Search Phrases

id	Search phrases	Abbreviated form	Root form
1	国足	guó-zú	guó-jiā-zú-qiú-duì
2	英超	yīng-chāo	yīng-gé-lán-chāo-jí-lián-sài
3	意甲	yì-jiǎ	yì-dà-lì-jiǎ-jí-lián-sài
4	北建大	běi-jiàn-dà	běi-jīng-jiàn-zhù-dà-xué
5	清华	qīng-huá	qīng-huá-dà-xué
6	哈工大	hā-gōng-dà	hā-ěr-bīn-gōng-yè-dà-xué
7	北大	běi-dà	běi-jīng- dà-xué

For each Chinese document, we word segmented it with an online segmentation tool (http: //life.chacuo.net/convertexportword), filtered all punctuations and some auxiliary words with a customized dictionary, and then saved it as a new document represented as word vector.

For English document, there is no need to do word segmentation since English text has explicit word boundary markers. Phrases can be located more accurately without the influence of word ambiguities as in Chinese word segmentation. A three-phase procedure is used to construct our English vocabulary: pre-processing, word frequency calculating and word co-occurrence frequency calculating. Firstly, stop words in data corpus are removed using Glasgov stop-words vocabulary. Then, term frequency and document frequency are calculated on all non-stopping words over the corpus using a linear list of link structure. Each node of the list is a structure containing two members: one is the count of words with the same hash value equaling to the node's index and the other a pointer to a linked list. Each node of the linked list is a structure corresponding to a word. It stores member variables as word string, term frequency and document frequency. We only consider words whose document frequency no less than 2 and denote the resulted words set as $\mathrm{TW}_{t,d}=\{t \mid t\in d \text{ and } df_t\geq 2\}$. Finally, term frequency and document frequency of words co-occurrences are calculated over $\mathrm{TW}_{t,d}$ using a two word collocational window to capture bigram word co-occurrence. Let tc be a word co-occurrence phrase starting with term t, $TWC_{tc,d} = \{tc \mid tc \in d \text{ and } df_{tc} \geq 2\}$, we build our English vocabulary $WTI_{t \cup tc,d} = \mathrm{TW}_{t,d} \cup TWC_{tc,d}$.

4.2. Result

For each phrase, we employ the aforementioned attacking process on document set with its size $n=5$ and $n=3$ respectively. We use formula (4) to calculate its TF/DF weights with α and β both taking value 0.5. We count the number of phrases guessed right on the first guess, on the second guess, on the third guess, as shown in Table.3. We can see from the table that our approach works both on Chinese and English languages in principle. Actually, the problem for English language becomes easier. First and foremost, the feasibility of our approach hasn't changed yet. That is, during the searching period, the majority of the webpages visited by user, no matter indexed by the search engine or hyper-linked in the indexed webpages, focus on the target subject and have similarity in content. Secondly, there is no need to do word segmentation since English text has explicit word boundary markers. Phrases can be located more accurately without the influence of word ambiguities as in Chinese word segmentation. In statistical Natural Language Processing (NLP) methods, word co-occurrence shows probabilistic word association. Since we only consider bi-gram English word co-occurrence in our experiment but use a professional Chinese word segmentation software with self-contained lexicon and optimized segmentation rules, the result of English language is slightly lower than Chinese language.

Table 3. Statistics of the Guessing Results

Language	Phrases	First guess(%)	Second guess(%)	Third Guess(%)
Chinese	140	96(68.57)	122(87.14)	135(96.63)
English	40	26(65)	34(85)	38(95)
Total	180	122(67.78)	156(86.67)	173(96.11)

We now make a detailed analysis on the results for the 28 Chinese categorized phrases as shown in Table 4. For these 28 phrases, our attacking method hits the search phrases at first guess with probability 67.86%, and achieves 96.43% hit rate in the first three guesses. The column “TD5 rank” and “TI5 rank” denote the ranking result of our TF/DF method and TF/IDF method respectively on document set size five, and “TD3 rank” and “TI3 rank” the ranking result of our TF/DF method and TF/IDF method respectively on document set size three. As we can see, there is no difference between the two different document set size. Out of the 28 search phrases, our method guessed right on the first guess 19 phrases, on the second guess 3 phrases, on the third guess 1 phrases. Four phrases are Chinese noun compounds and they are both the combinations of the first two or three guessed keywords. Three phrases belonging to the category of university are in abbreviated form and the attacking result presents the full name and the abbreviated name of these universities in order of precedence. Only keyword “tracheitis”, spelled as “qì-guǎn-yán” in Chinese, is mistaken for “bronchitis” which in Chinese spelled as “zhī-qì-guǎn-yán”. We reviewed the five documents and found that they were more related to bronchitis than tracheitis.

Table 4. Guessing Result of Search Phrases

id	Search phrases	TD5 rank	TI5 rank	TD3 rank	TI5 rank
1	xīn-néng-yúan-qì-chē	1, 2*	1147, 2021*	2, 1*	919, 466*
2	bǎo-mǎ	1	1009	1	839
3	bēn-chí	2	1	1	594
4	ào-dí	1	1209	1	1041
5	gǔ-shì	1	5791	1	3645
6	lǐ-cái	3	1	3	1
7	guó-zhài	2	2	2	1
8	xìn-yòng-kǎ	1	747	1	364
9	zhōu-jié-lún	1	1242	1	1206
10	fàn-bīng-bīng	1	33	1	1052
11	xiè-tíng-fēng	1	969	1	757
12	wāng-fēng	1	1341	1	706
13	guó-zú	2	3191	2	2791
14	yīng-chāo	1	1726	1	836
15	yì-jiǎ	1	3096	1	2068
16	shì-jiè-bēi	1	1562	1	786
17	běi-jiàn-dà	1,3,2*	1746, 1745, 1748*	1,3,2*	1609, 1607, 1610*
18	qīng-huá	1	1498	1	1019
19	hā-gōng-dà	1,2*	1586, 4007*	1,2*	1193, 1194*
20	běi-dà	1	1	1	4
21	kōng-qì-jìng-huà-qì	1,2*	586, 722*	1,2*	234, 386*
22	kōng-tiáo	1	701	1	857
23	bīng-xiāng	1	1048	1	747
24	xǐ-yī-jī	1	366	1	332
25	wù-mái	1	2302	1	1093
26	kǒu-zhào	1	618	1	327
27	qì-guǎn-yán	28	652	81	190
28	huán-bǎo	1	3909	1	3672

*-the search phrases are Chinese noun compounds of the listed phrases.

4.3. Reconstructing User's Web-surfing Activity

For each of the 173 hit phrases, we first imitated the user querying Baidu search engine, retrieving 5 URLs indexed within the first three results pages and surfing them. The network traffic traces were recorded as dataset DS_u . We then imitated the attacker querying Baidu search engine, retrieving all 30 URLs indexed within the first three results pages and surfing them. The network traffic traces were recorded as dataset DS_a . The flow feature statistics were calculated for each flow within DS_u and DS_a .

As there are no HTTPS webpages indexed by Baidu search engine, and if all URLs requested can be observed directly through bi-direction or uplink direction plain-text network traffic monitoring, DNS-based URLs comparison is sufficient for the identification task to achieve 100% accuracy.

Under the possible circumstance that only downlink traffic data can be monitored, we can apply flow feature statistics-based similarity computing approach to fulfill the identification task.. The approach goes for both bi-direction and uni-direction network traffic capture and for webpages visited through both plain-text HTTP and encrypted HTTPS. Let $\gamma = 0.05$, we use (8) to compute webpage session similarity between that of the user and that of the attacker. Experimental result shows that our Flow feature statistics-based identifying approach also exhibits a promising performance with 100% accuracy.

4.4. Discussion of Impact of Evaluation Conditions

Some conditions may have impacts on the accuracy of our experimental evaluation. They include value of α, performance of Chinese word segmentation, and ranking mechanisms of search engine.

4.4.1.Effects of α : Our TF/DF feature selection method takes both term frequency and document frequency into consideration. We use a pair of parameters, α and β, to adjust their weights. For simplicity, β was set to $1-\alpha$ in the experiment. We evaluated the hit rate of first guess and three guess under some typical values of α. The result is listed in Table 6. As we can see, the optimal hit rate appears at $\alpha = 0.4$ and $\alpha = 0.5$.

Table 5. Effects of α

α	β	First guess	Three guess	α	β	First guess	Three guess
0.0	1.0	66.67%	86.11%	0.5	0.5	67.78%	96.11%
0.1	0.9	67.22%	90.56%	0.6	0.4	67.78%	93.33%
0.3	0.7	67.78%	93.33%	0.7	0.3	67.78%	93.33%
0.4	0.6	67.78%	96.11%	1.0	0.0	67.78%	86.11%

4.4.2.Effects of Chinese Word Segmentation: Chinese word segmentation process may bias the statistics of word frequency to some extent. Since Chinese word boundaries are not marked by spaces, some searched phrases may be segmented either as the right phrases or as part of Chinese noun compounds depending on the specific segmentation algorithm. As we found in the experiment, it did not affect our result much.

4.4.3.Effects of Indexing and Ranking Mechanisms of Search Engine: The first algorithm used by Google Search to rank webpages was PageRank. PageRank measures a webpage's ranking weight by counting the number and quality of its incoming links. Google later updated the algorithm to pay more attention to the quality of content and link with Panda and Penguin. The Latest major update is Hummingbird. Hummingbird places greater emphasis on page content and aims to take the whole search query and its meaning into account, rather than a few particular words. Hyperlink-Induced Topic Search (HITS) is another link-based ranking algorithms for Web pages. Chinese search engine giant Baidu hasn't made public its ranking algorithm. Baidu offers Paid Search andPay for Placement (P4P), which enables customers creating text-based descriptions of their web pages and bid on keywords that trigger the display of their webpage information and links. These mechanisms to some extent bias the result for a search query. Indexing and ranking mechanisms of different search engines may lead to significant differences on search results. But it has little impact on attacking effect of our phrase counting-based approach since a user's intension is goal-oriented when he visits searching service, contents of the webpages he clicks are often homogenous. Some search engines such as Google provides personalized search results based on the user's activity history. This may lead to striking differences between search results pages of the user and that of the attacker and increase the difficulty in encrypted network traffic analyzing.

5. Conclusion

Web search engine has become the portal for Internet users to obtain information resources. The queries submitted to the search engine represent the users' motivation or behavior at that moment, thus are highly related to personal privacy. In order to protect the data communicating from eavesdropped by network attackers, some search engines enable secure search Relying on HTTPS to encrypt all data transferred between search engine and the users. This gives their users a delusive sensation of security and privacy. While since the majority of Internet websites haven't been HTTPS-enabled, webpages indexed in the results pages can only be visited through HTTP. Once the generated plain-text network traffic be monitored by an attacker, the query phrase might be reversed. We presented the attacking scenarios and demonstrated the feasibility of such attacking. The effectiveness of our TF-DF phrase selection method is based on the fact that during the searching period, the majority of the webpages visited by user, no matter indexed by the search engine and linked in the indexed webpages, focus on the target subject and have similarity in content.

By demonstrating the technique to attack secure search service, we accentuate the importance of universal encryption to protect Internet communications and encourage searching service providers to crawl, index and show users more HTTPS webpages in search results.

Besides, pervasive encryption has limited the ability of computer network incident response team and law enforcement to perform network forensics, our technique can be utilized to assist analyzing and tracking sensitive queries such as violence, terrorism and drugs to secure search service, that might follow potential offline criminal activities, and this is the focus of our next work.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61571144.

References

[1] P. Gill, M. Crete-Nishihata, J. Dalek, S. Goldberg, A. Senft, G. Wiseman, "Characterizing Web Censorship Worldwide: Another Look at the OpenNet Initiative Data", ACM Transactions on the Web (TWEB), vol. 9, no. 1, **(2015)**, Article No.4. DOI:10.1145/2700339.

[2] F. Mouton, L. Leenen, H.S. Venter, "Social engineering attack examples, templates and scenarios", Computers & Security, vol. 59, Issue C, **(2016)**, pp.186-209. DOI:10.1016/j.cose.2016.03.004

[3] Y. Li, Y. Li, Q. Yan, R. H. Deng, "Privacy leakage analysis in online social networks", Computers & Security, vol. 49, **(2015)**, pp.239-254. DOI:10.1016/j.cose.2014.10.012.

[4] S. Chakravarty, G. Portokalidis, M. Polychronakis, A. D.Keromytis, "Detection and Analysis of eavesdropping in anonymous communication networks", International Journal of Information Security. vol.14, no. 3, **(2015)**, pp.205-220. DOI:10.1007/s10207-014-0256-7.

[5] D. Naylor, A. Finamore, I. Leontiadis, Y. Grunenberger, M. Mellia, M. Munafo, K. Papagiannaki and P. Steenkiste, "The Cost of S in HTTPS", Proceedings of CoNext, **(2014)**, pp. 133-140. DOI: 10.1145/2674005.2674991.

[6] Z. Yuan, Y. Xue and W. Xia, "PPI: Towards Precise Page Identification for Encrypted Web-browsing Traffic", Proceedings of ANCS, **(2013)**, pp. 109-110. DOI: 10.1109/ANCS.2013.6665182.

[7] W. Xia, Y. Ren, Z. Yuan and Y. Xue, "TCPI: A Novel Method of Encrypted Page Identification", Proceedings of CCIS, **(2013)**, pp. 453-456. DOI: 10.2991/ccis-13.2013.105.

[8] B. Miller, L. Huang, A.D. Joseph and J.D. Tygar, "I Know Why You went to the Clinic: Risks and Realization of HTTPS Traffic Analysis", Proceedings of PETS, **(2014)**, pp. 143-163. DOI:10.1007/978-3-319-08506-7_8.

[9] A. Pironti, P.-Y. Strub and K. Bhargavan, "Indentifying Website Users by TLS Traffic Analysis: New Attacks and Effective Countermeasures", INRIA, Research Report RR-8067, **(2012)**.

[10] G. Xie, M. Iliofotou, T. Karagiannis, M. Faloutsos and Y. Jin, "Reconstructing Web-Surfing Activity from Network Traffic", Proceedings of IFIP Networking Conference, **(2013)**, pp. 1-9.

[11] C. Neasbitt, "Click Miner: Towards Forensic Reconstruction of User-Browser Interactions from Network Traces", Proceedings of ACM CCS, **(2014)**, pp. 1244-1255. DOI: 10.1145/2660267.2660268.

[12] D. Gugelmann, “Hviz: HTTP(S) traffic aggregation and visualization for network forensics”, Digital Investigation, vol. 12, no. Sup1, **(2015)**, pp. S1-S11. DOI: 10.1016/j.diin.2015.01.005.

[13] M. Conti, Luigi V.Mancini, Riccardo Spolaor and Nini V.Verde, “Can’t You Hear Me Knocking: Identification of User Actions on Android Apps via Traffic Analysis”, Proceedings of ACM SIGSAC CODASPY, **(2015)**. DOI: 10.1145/2699026.2699119.

[14] Y. Jia, Y. Chen, X. Dong, P. Saxena, J. Mao and Z. Liang, “Man-in-the-browser-cache: Persisting HTTPS attacks via browser cache poisoning”, Computers & Security, vol. 55, **(2015)**, pp. 62-80. DOI: 10.1016/j.cose.2015.07.004.

[15] S. Chen, R. Wang, X. F. Wang and K. Zhang, “Side-Channel Leaks in Web Applicatins: A Reality Today, a Challenge Tomorrow”, Proceedings of 2010 IEEE Symposium on Security and Privacy, **(2010)**, Oakland, CA, USA, IEEE, pp. 191-206. DOI: 10.1109/SP.2010.20.

[16] M. Juarez, S. Afroz, G. Acar, C. Diaz and R. Greenstadt, “A Critical Evaluation of Website Fingerprinting Attacks”, Proceedings of ACM SIGSAC Conference on Computer and Communications Security (CCS’14), **(2014)**, pp. 263-274. DOI: 10.1145/2660267.2660368.

[17] M. Korczynski, A. Duda, “Markov Chain Fingerprinting to Classify Encrypted Traffic”, Proceedings of the 2014 IEEE Conference on Computer Communications (IEEE INFOCOM 2014), **(2014)**, Toronto ON, IEEE. DOI: 10.1109/INFOCOM.2014.6848005.

[18] K. Goseva-Postojanova, G. Anastasovski, A. Dimitrijevik, R. Pantev and B. Miller, “Characterization and Classification of Malicious Web Traffic”, Computers & Security, vol. 42, **(2014)**, pp. 92-115. DOI: 10.1016/j.cose.2014.01.006.

[19] Z. Luoshi, X. Yibo and B. Yuanyuan, “A New Network Traffic Classification Method Based on Classifier Integration”, IJGDC, vol.8, no. 3, **(2015)**, pp. 309-322. DOI: 10.14257/ijgdc.2015.8.3.29.

[20] Y. Wang, Y. Xiang, J. Zhang, W. Zhou, G. Wei and L. T. Yang, “Internet Traffic Classification Using Constrained Clustering”, IEEE Transactions on Parallel and Distributed Systems, vol. 25, no.11, **(2014)**, pp. 2932-2943. DOI: 10.1109/TPDS.2013.307.

[21] Z. Luoshi, X. Yibo and W. Dawei, “The Effectiveness Study of ML-based Methods for Protocol Identification in Different Network Environments”, IJFGCN, vol.8, no.2, **(2015)**, pp. 213-224. DOI: 10.14257/ijfgcn.2015.8.2.16.

[22] S. L. Blond, D. Choffnes, “Herd: A Scalable, Traffic Analysis Resistant Anonymity Network for VoIP Systems”, Proceedings of SIGCOM, **(2015)**, pp. 639-652. DOI: 10.1145/2829988.2787491.

[23] P. Vines, T. Kohno, “Rook: Using video games as a low-bandwidth censorship resistant communication platform”, Proceedings of WPES, **(2015)**, pp. 75-84. DOI: 10.1145/2808138.2808141.

[24] K. P. Dyer, S. E. Coull and T. Shrimpton, “Marionette: A Programmable Network-Traffic Obfuscation System”, Proceedings of USENIX, **(2015)**, pp. 367-382.

Authors

Qian Liping, received the B.S. degree in mathematics from Department of Mathematics, Anhui Normal University, China, in 1993, the M.S. degree in computer application from Inner Mongolia University, China, in 2001, and the Ph.D. degree in computer application from Renmin University of China, Beijing, in 2015. From 1993 to 1997, she was an assistant in computer application with the Foundation Department, Anhui Finance & Trade College and now she is an associate professor of the College of Electrical and Information Engineering, Beijing University of Civil Engineering and Architecture. Her research interest includes network security, artificial information processing and natural language processing.

Wang Lidong, received the B.S. degree in mathematics from Anhui Normal University, China in 1993, the M.S. degree in computer application in 1998 and the Ph.D. degree in computer architecture in 2002 both from Harbin Institute of Technology, Harbin, China. From 1993 to 1996, he was an assistant in computer application with the Foundation Department, Anhui Finance & Trade College and now he is a professor of the Information Security Laboratory, National Computer network Emergency Response technical Team/Coordination Center of China (CNCERT/CC), Beijing, China. His research interest includes network security and big data analysis.