International Journal of Engineering Science Invention (IJESI)
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org || PP. 77-82

Implication Hit on Browsing History of Twitter Users using Back Propagation with Public Click Analytics and Twitter Metadata

R.Jothi.,[1] , K.Pushpalatha[2] , Dr.C.Bhuvaneswari[3]
[1](Assistant Proessor and head, Department of Computer Application,
Krishnasamy College of Science, Arts & Management for Women, Cuddalore-9, TamilNadu. India)
[2](Research Scholar, Department of Computer Science,
Krishnasamy College of Science, Arts & Management for Women, Cuddalore-9, TamilNadu, India)
[3](Assistant Professor and head , Department of computer science,
Thiruvalluvar university college of arts and science, Thiruvennainallur, TamilNadu, India)

Abstract: *The increased use of Internet, Online Social Networks (OSN) has become a part of life for millions of people nowadays. Every day, users of such networks including Face book, Whatsapp, Twitter, etc. execute millions of activities, such as sharing information, posting comments, uploading audio, video, photos, and updating status. The demand on a huge amount of information and application that users upload, install, and execute on the social networks makes the social networks an attractive target for attackers. In this study, a new model has been built based on Back Propagation Neural Network (BPNN) so as to identify the vulnerability level of the user with the practical attack techniques inferring who clicks which shortened URLs on Twitter using the combination of public information: Twitter metadata and public click analytics in a social network, nodes correspond to people or other social entities, and edges correspond to social links between them. In an effort to preserve privacy the use of Ad Hoc On-Demand Distance Vector (AODV) protocol is implemented, the practice of anonymization replaces names with meaningless unique identifiers. Unlike the conventional browser history stealing attacks, inference attacks only demand publicly available information provided by Twitter and URL shortening services. Evaluation results show that our attack can compromise Twitter user's privacy with high accuracy.*

Keywords: *Inference, Privacy leak, Twitter, URL Shortening Service.*

I. Introduction

Twitter is a popular online social network and microbloging service for exchanging messages also known as tweets among people, supported by a huge ecosystem. Twitter announces that it has over 140 million active users creating more than 340 million messages every day and over one million registered applications built by more than 750,000 developers. The third party applications include client applications for various platforms, such as Windows, Mac, iOS, and Android, and web-based applications such as URL shortening services, image sharing services, and news feeds. Among the third party services, URL shortening services which provide a short alias of a long URL is an essential service for Twitter users who want to share long URLs through tweets having length restriction. Twitter allows the users to post up to 140-character tweets containing only texts. Therefore, when users want to share complicated information e.g., news and multimedia, they should include a URL of a web page containing the information into a tweet. Since the length of the URL and associated texts may exceed 140 characters, Twitter users demand URL shortening services further reducing it. Some URL shortening services e.g., bit.ly and goo.gl, also provide shortened URLs public click analytics consisting of the number of users clicks, countries, browsers, and referrers of visitors. Although anyone can access the data to analyze visitor statistics, no one can extract specific information about individual visitors from the data because URL shortening services provide them as an aggregated form to protect the privacy of visitors from attackers.

However, we detect a simple inference attack that can estimate individual visitors from the aggregated, public click analytics using public metadata provided by Twitter. First, we examine the metadata of client application and location because they can be correlated with those of public click analytics. For instance, if a user Cheng, updates the messages using the official Twitter client application for iPhone, "Twitter for iPhone" will be included in the source field of the corresponding metadata. Moreover, Cheng may disclose on her profile page that she lives in the US or activate the location service of a Twitter client application will automatically fill the location field in the metadata. Using this information, we can determine that Cheng is an iPhone user who lives in the US. Next, we perform the simple inference attack on behalf of Cheng's boyfriend, Lee, as follows.

Lee first posts a tweet with a URL shortened by goo.gl, if Cheng clicks on Shortened URL, records "country":"US","platform":"iPho","referrer""twitter.com","browser":"Mobile" in the click analytics of the

shortened URL. Otherwise, goo.gl records no information. Later on, Lee retrieves the click analytics of the shortened URL to know whether Cheng clicks on his URL. If the click analytics is unchanged or if its changes do not include information about the US, iPhone, and twitter.com, he infers that Cheng does not click on his URL. Otherwise, he infers that Cheng click on his URL. The main advantage of the preceding inference attack over the conventional browser history stealing attacks is that it only demands public information. The conventional browser history stealing attacks rely on private information, such as Cascading Style Sheet (CSS), browser cache, visited styles, DNS cache, and latency. To collect such information, attackers should

(i) Prepare attack pages containing fraudulent scripts / malware and lure target users for extracting the information from their web browsers or (ii) monitor the DNS requests for measuring DNS lookup time of a target host. In other words, attackers should deceive or compromise target users or their networks to obtain the browsing history, which relies on strong assumption. In contrast, anyone can access the metadata of Twitter and the public click analytics of URL shortening services so that passive monitoring is enough for performing our attack. In this paper, we propose novel attack methods for inferring whether a specific user clicked on certain shortened URLs on Twitter. As shown in the preceding simple inference attack, our attacks rely on the combination of publicly available information: click analytics from URL shortening services and metadata from Twitter. The goal of the attacks is to know which URLs are clicked on by target users. We introduce two different attack methods: (i) an attack to know who click on the URLs updated by target users and (ii) an attack to know which URLs are clicked on by target users.

To perform the first attack, we find a number of Twitter users who frequently distribute shortened URLs, and investigate the click analytics of the distributed shortened URLs and the metadata as the followers of the Twitter users. To perform the second attack, we create monitoring accounts that monitor messages from all followings of target users to collect all shortened URLs that the target users may click on. We then monitor the click analytics of those shortened URLs and compare them with the metadata of the target user. Furthermore, we propose an advanced attack method to reduce attack overhead while increasing inference accuracy using the time model of target users, representing when the target users frequently use Twitter. Evaluation results show that our attacks can successfully infer the click information with high accuracy and low overhead.

II. Related Work

2.1 Browser History Stealing

There are several types of history stealing attacks. First, attackers exploit Cascading Style Sheet (CSS) visited styles. They use the fact that browsers display visited links differently from unvisited links. This attack is first introduced in 2000, and it has been discovered several times. In the academic community, propose a CSS based history stealing attack. They analyze behaviors of each browser related to CSS visited styles and build a system to know and browse the history of users efficiently. Second, attackers exploit browser and DNS cache to conduct history stealing attacks. Felten et al. describe attack methods using browser and DNS cache. They measure the time required to load web pages and to look up DNS. Jackson et al. enforce a same-origin policy to prevent history stealing attack. They implement a Firefox extension to enforce the policy on the browser cache and visited link. Jakobsson et al. personalize URL to prevent sniffing of browser cache and history. Attacker's snoofing DNS caches to infer visited sites of users. Grangeia proposes a technique of DNS cache snoofing Krishnan et al. describe how to leak user's privacy by using DNS prefetching .Third, some researchers propose attack methods to steal browsing history using user interactions and side-channels. Weinberg et al. exploit CAPTCHA to deceive users and to induce user's interaction. They also use a webcam to detect the light of the screen reflected at the user's face, which can be used to distinguish the colors of visited from those of unvisited links. The conventional history stealing attacks usually assume that victims visit a malicious web page or victims are infected by malware. However, our inference attacks do not need to make these assumptions. Our inference attacks only use the combinations of publicly available information, so anyone can be an attacker or a victim.

2.2 Privacy Leaks from Public Information

Previous studies have considered attack techniques that cause privacy leaks in social networks, such as inferring private attributes and de-anonymizing users. Most of them combine public information from several different data sets to infer hidden information. Some studies introduce de-anonymizing attacks in social networks. Backstrom etal. Try to identify edges existence in an anonymzed network, and Narayanan and Shmatikov identify Netflix records of known users using only a little bit of data about users. Furthermore, they combine their results with IMD b data and inferred user's political preferences or religious view. Narayanan and Shmatikov also prove that users who have accounts in both Twitter and Flicker can be recognized in the anonymous Twitter graph. Wondracek et al. propose a de-anonymized attack using group membership information obtained by browser history stealing attack. Other studies consider how to infer the private

attributes of users in social networks. He et al. and Lindamoodet al. build a Bayesian network to predict un disclosed personal attributes. Zheleva and Getoor show how an attacker can exploit a mixture of private and public data to predict private attributes of a target user.

Similarly, Mnislove et al. infer the attributes of a target user by using a combination of attributes of the user's friends and other users who are loosely (not directly) connected to the target user. Calandrino et al. propose algorithms inferring customer's transactions in the recommender systems, such as Amazon and Hunch. They combine public data of the recommender systems and some of the transactions of a target user in order to infer the target user's unknown transactions. Chaabane et al. propose an inference attack to predict undisclosed attributes by using only music interests. They derive semantics using Wikipedia ontology and measured the similarity between users.

III. URL Shortening Services

In this section, we briefly introduce URL shortening services. The first notable URL shortening service is Tiny URL, which was launched in 2002, and its success influences the development of many URL shortening services.URL shortening services reduce the length of URLs by providing short aliases of URLs to requesters and redirecting later visitors to the original URLs. The services are especially convenient for Twitter users, which imposes a limit on the length of a message. In the past, Twitter used Tiny URL and bit.ly as the default URL shortening services. As of from October 10, 2011, Twitter started using its own URL shortening service, t.co, to wrap all URLs in tweets in order to protect Twitter users from malicious URLs. Some URL shortening services also provide click analytics about each shortened URL. Whenever a user clicks on a shortened URL, information about the user is recorded in the corresponding click analytics. The click analytics is usually made public and anyone can access it. Among the services, we focus on bit.ly and goo.gl because they are broadly used and provide detailed information.

We summarize the main contributions of this paper as follows

- In this study, a new model has been built based on Back Propagation Neural Network (BPNN) so as to identify the vulnerability level of the user with the practical attack techniques inferring who clicks which shortened URLs on Twitter using the combination of public information: Twitter metadata and public click analytics. Unlike the conventional browser history stealing attacks, our attacks only demand publicly available information provided by Twitter and URL shortening services. Evaluation results show that our attack can compromise Twitter users' privacy with high accuracy.
- We propose novel attack techniques to determine whether a specific user clicks on certain shortened URLs on Twitter. To the best of our knowledge, this is the first study that infers URL visiting history on Twitter. We only use public information provided by URL shortening services and Twitter (i.e., click analytics and Twitter metadata). We determine whether a target user visits a shortened URL by correlating the publicly available information.
- Our approach does not need complicated techniques or assumptions such as script injection, phishing, malware intrusion, or DNS monitoring. All we need is publicly available information. We further decrease attack overhead while increasing accuracy by considering target users time models. It can increase the practicality of our attacks so that we demand immediate countermeasures.

3.1 goo.gl

In December 2009, Google launched the Google URL shortened at goo.gl. Its click analytics provides the following information about the visitors.

- Referrers
- Countries
- Browsers
- Platforms

For example, let us assume a user uses a Black Berry phone and lives in the USA. If he clicks on a shortened URL from goo.gl on Twitter, gool.gl records (i) t.co in the Referrers field, (ii) Mobile Safari in the Browsers field, (iii) US in the Countries field, and (iv) BlackBerry in the Platforms field of the corresponding click analytics. The reason why t.co is recorded in the Referrers field is that all links shared on Twitter are wrapped using t.co by Twitter from October 10, 2011.

3.2 bit.ly

Bitly company launched a URL shortening service bit.ly in 2008. [5] Its click analytics provides the following information about visitors.

- Referrers

- Countries

Although bit.ly does not provide information about browsers and platforms, its Referrers field contains detailed information which can be utilized for inference. When a user clicks on a shortened URL on Twitter, bit.ly records the entire URL of the referrer site in the Referrers field, as "http://t.co/*****". In contrast, goo.gl records only "t.co"in the Referrers field. If we use the information provided by bit.ly, we can determine the exact URL of the tweet containing the clicked shortened URL. This information makes our inference attack possible even without having information about browsers and platforms.

3.2.1 Referrers

We determine whether a new visitor comes from Twitter by using the changed referrer information of public we determine whether a new visitor comes from Twitter by using the changed referrer information of public click analytics. The click analytics of goo.gl only records the hostname of the referrer site. If a visitor comes from Twitter, "t.co" or "twitter.com" is recorded in the Referrers m field. In most cases, "t.co" is recorded because all links shared on Twitter are automatically shortened to t.co links. t.co handles redirections by context and user agents so that the Referrer information varies according to the source of a click.

In some cases, "twitter.com" is recorded because some Twitter applications directly use original links instead of t.co links. Consequently, if the Referrers information of the visitor is "t.co" or "twitter.com", we regard the visitor as coming from Twitter. In the case of bit.ly, we can analyze a shortened URL in detail because bit.ly records the entire URL of the referrer site in click analytics. When a target user clicks on a bit.ly URL converted into a t.co URL, bit.ly records the entire t.co URL in the Referrers field. Referrer matching is considered successful when a t.co URL recorded in the click analytics is the same as thet.co URL of the target shortened URL.

3.2.2 Country

We infer the country information of target users using the location field of their profile pages and compare it with the changed country information of public click analytics. In many cases, Twitter users fill in the location field with their city or place name. We can determine the user's country by searching Geo Names with the information in the location field of the user's Twitter profile. Geo Names returns the country code that corresponds to the search keywords.

The country information provided by the click analytics is also a country code, so we have a successful country match if both country codes are the same. However, the country matching has a limitation: it does not work when Twitter users leave the location field empty or fill in the location field with meaningless information (e.g., "earth" or "in your heart"). According to Hecht et al, approximately 34% of Twitter users do not provide real location information. In the later experiments, we avoid such problems by only selecting target users who filled in valid location names in the location field. However, even without location information, our attacks are still possible with other information because the country information is not a major feature for conducting inference. Additionally, we can utilize recent studies inferring the location of Twitter users based on their posts for our attacks.

3.3 Browsers and Platforms

When our target users click on a shortened URL provided by goo.gl, we use the information about the user's browser and platform to increase the inference accuracy. Although Twitter does not provide information of this nature about its users, it does record the name of the application that was used to post a tweet. For example, when someone posts a tweet using the official Twitter client application for the iPhone, the information "Twitter for iPhone" is recorded in the source field of the tweet, which enables us to use this information to infer the browser and platform that were used [7]. We should consider applications supporting multiple platforms, such as Tweet Deck, which is a multi-platform application that is supported by the iOS, Android, Windows, and Mac OS X operating systems. A target user, who uses a multi-platform application, should be regarded as using all the platforms that support the application.

IV. Basics of Inference Attack

In this section, we introduce the basics of our inference attack. The basic idea of our attack is capturing instant changes in the public click analytics of shortened URLs by periodically monitoring it and matching the instant changes with the information about target users to infer whether our target users make the changes.

4.1 Periodic Monitoring and Matching

We periodically monitor click analytics of shortened URLs to observe its instant changes made by a new visitor. Whenever we notice that there is a new visitor, we match his or her information with each of our target users to know whether the new visitor is one of our target users. We can estimate information about visitors by checking the differences between the new and the old click analytics, the process for obtaining the information about the visitor who clicks on a goo.gl URL. In this figure, we easily infer that the new visitor is an iPhone user lives in the US because the numbers of clicks by "US" and "iPhone" simultaneously increase. In the periodic monitoring, determining the optimal query interval is important, which depends on the variety of the characteristics of followers. When there are some characteristics to be observed at the same time and their whole values change rapidly, the query interval should be short enough to catch a small change. As we have many followers, the slope becomes stiff so that we should have a short interval. In general, an interval should be decided by considering the slope of change in overall characteristics. However, the periodic monitoring and matching have a limitation because Twitter does not officially provide personal information about users such as country, browsers, and platforms. We need to infer the information about target users by investigating their timeline and profile pages.

Thus we tweet history of Twitter users to build time models, and then evaluate the advanced attack by conducting experiments with the virtual users with the time models. On average, the Twitter users as analyzed do not post tweets more than hours a day and five percent of the Twitter post tweets in all day, in both accounts the inference system excludes the click logs in the time periods that the users do not click. The performance of the inference system can improve even when the system does not monitor Twitter all the time.

Table 1. An example of bit.ly clicks analytics. UID stands for a Twitter user ID and TID stands for a numerical ID of each tweet.

		#
	direct	2
Referrers	http://t.co/3slAb	8
	http://t.co/xInA4	4
	https://twitter.com/[UID]/status/[TID]	3
	US	9
Countries	KR	5
	ID	1
	CH	2

V. Conclusion

The proposed inference attacks to infer which shortened URLs clicked on by a target user. All the information that are needed in our attacks is public information: the click analytics of URL shortening services and Twitter metadata. They have described several variations in the practice of re-tweeting messages[4] on Twitter and the ways in which varying styles lead to ambiguity in and around authorship, attribution, and conversational fidelity, especially as the content of messages morph as they are passed along.

To evaluate our attacks, we monitored the click analytics of URL shortening services and Twitter data. The brevity of messages allows them to be produced, consumed, and shared without a significant amount of effort, allowing a fast placed conversational environment to emerge. In future we propose a new vulnerability identification method for the OSN user to create awareness on their security and privacy status. It finds out the relation between OSN user behaviors and attacker scenarios.

The future is to validate the correctness of our inference. To know the timeline and the favorites of the user by checking whether a tweet containing the shortened URL is exists. Twitter users include URLs in their tweets and favorites tweets only when they visit the previous URLs. It suspects that the contrast between a period model and the history is few since it is essential to concentrate on over online networking clients who regularly post or tweets all their working hours. It utilizes time models to arrange time based practices of virtual clients in a most situation.

References

Books:

[1] Margaret Levine Young, *Internet and WWW* (Tata McGraw Hill, 2002).

[2] Harley Hahn, *the Internet* (Tata McGraw Hill).

Proceedings Papers:

[3] L.Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou r3579x? Anonymized social networks, hidden patterns, and structural Steganography. *Proc. 16th Int'l World Wide Web Conf.* (WWW), 2007.

[4] D. Boyd, S. Golder, and G. Lotan. Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. *Proc. 43rd Hawaii International Conference on System Sciences* (HICSS), 2010.

[5] Bugzilla. Bug 57351: css on a: visited can load an image and/or reveal if visitor been to a site, 2000. https://bugzilla.mozilla.org/showbug.cgi?id=57351.
[6] J. A. Calandrino, A. Kilzer, A. Narayanan, E. W. Felten, and V. Shmatikov. "you might also like:" privacy risks of collaborative filtering. *Proc. IEEE Symp. Security and Privacy* (S&P), 2011.
[7] A. Chaabane, G. Acs, and M. A. Kaafar. You are what you like! Information leakage through users' interests. *Proc. 19th Network and Distributed System Security* (NDSS), 2012.
[8] Z. Cheng, J. Caverlee, and K. Lee. You are where you tweet: A content-based approach to geo-locating twitter users. *Proc.19th ACM International Conference on Information and Knowledge Management* (CIKM), 2010.
[9] A. Clover. Css visited pages disclosure, 2002.http://seclists.org/bugtraq/2002/Feb/271.
[10] C. Dwork. Differential privacy. *Proc. 33rd International Colloquium on Automata, Languages and Programming* (ICALP),2006.
[11] E. W. Felten and M. A. Schneider. Timing attacks on web privacy. *Proc. 7th ACM Conf. Computer and Comm. Security* (CCS), 2000.
[12] C. Jackson, A. Bortz, D. Boneh, and J. C. Mitchell. Protectingbrowser state from web privacy attacks. In Proc. 15th Int'l WorldWide Web Conf. (WWW), 2006.
[13] B.Hecht, L.Hong, B.Suh Tweetss from justinbiebers
[14] Heart: The dynamics of the location field in user profiles. In Proc. SIGCHI conference on human factors in computing systems, 2011.