FUZZY FINGERPRINT METHOD FOR DETECTION OF SENSITIVE DATA EXPOSURE

Staicy Ulahannanl[1] and Roshni Jose[2]

[1] Student, Department of Computer Science Engineering, MBITS Nellimattom
[2]Assistant Professor, Department of Computer Science, MBITS Nellimattom

ABSTRACT

Protecting confidential information is a major concern for organizations and individuals alike, who stand to suffer huge losses if private data falls into the wrong hands. Network-based information leaks pose a serious threat to confidentiality. This paper describes network-based data-leak detection (DLD) technique, the main feature of which is that the detection does not require the data owner to reveal the content of the sensitive data. Instead, only a small amount of specialized digests are needed. The technique referred to as the fuzzy fingerprint – can be used to detect accidental data leaks due to human errors or application flaws. The privacy-preserving feature of algorithms minimizes the exposure of sensitive data and enables the data owner to safely delegate the detection to others.

KEYWORDS

Network Security, Privacy, Data Leak, Detection, Collection Intersection

1. INTRODUCTION

Information leaks are a major problem of computer systems. The leak of confidential data either be it accidental or intentional, may cause huge losses to the data owner. Though there are number of systems designed for the data security by using different encryption algorithms, there is a big issue of the integrity of the users of those systems. It is very hard for any system administrator to trace out the data leaker among the system users. It creates a lot many ethical issues in the working environment.

Typical approaches to preventing data leak are under two categories – host-based solutions and network-based solutions. Host-based approaches may include encrypting data when not used and enforcing policies to restrict the transfer of sensitive data. Most of the host-based solutions require the use of virtualization or special hardware to ensure the system integrity of the detector. This paper present a novel network-based data-leak detection (DLD) solution that is both efficient and privacy-preserving In comparison to host-based approaches, network-based data-leak detection focuses on analyzing the (unencrypted) content of outbound network packets for sensitive information.

Another motivation for the privacy-preserving DLD work is cloud computing, which provides a natural platform for conducting data-leak detection by cloud providers as an add on service. In cloud computing environments, an organization (data owner) may have already outsourced its

DOI: 10.5121/ijci.2016.5207

services to a cloud provider, such as the email service for its own employees. The cloud provider may offer additional services such as inspecting email traffic for inadvertent data leak and serves as a DLD provider. This add-on DLD service requires minimal changes to the cloud provider's infrastructure and makes the cloud service more attractive. However, privacy is a major roadblock for realizing outsourced data-leak detection. Conventional solutions require the data owner to reveal its sensitive data to the DLD provider.

However, the DLD provider is always modeled as an honest-but-curious (aka semi-honest) adversary who is trusted to perform the inspection, but may attempt to learn about the data. Existing work on cryptography-based multiparty computation is not efficient enough for practical data leak inspection in this setting.

This paper design, implement, and evaluate a new privacy preserving data-leak detection system that enables the data owner to safely deploy locally, or to delegate the traffic inspection task to DLD providers without exposing the sensitive data. In this model, the data owner computes a special set of digests or fingerprints from the sensitive data, and then discloses only a small amount of digest information to the DLD provider [3]. These fingerprints have important properties, which prevent the provider from gaining knowledge of the sensitive data, while enable accurate comparison and detection. The DLD provider performs deep-packet inspection to identify whether these fingerprint patterns exist in the outbound traffic of the organization or not, according to a quantitative metric. To prevent the DLD provider from gathering exact knowledge about the sensitive data, the collection of potential leaks is composed of real leaks and noises. It is the data owner, who post-processes the potential leaks sent back by the DLD provider and determines whether there is any real data leak. Data leak is intentional or unintentional release of secure information to an untrusted environment.

These technical contributions are summarized as follows.

1. This paper describes a novel fuzzy fingerprint method for detecting inadvertent data leak in network traffic. Network security consists of the policies adopted to prevent and monitor unauthorized access, misuse, modification, or denial of computer network and computer accessible resource. Its main feature is that the detection can be performed based on special digests without the sensitive data in plaintext, which minimizes the exposure of sensitive data during the detection. This strong privacy guarantee yields a powerful application of fuzzy fingerprint method in the cloud computing environment, where the cloud provider can perform data-leak detection as an add-on service to its clients. This paper describes the quantitative privacy model, algorithms, and analysis in fuzzy fingerprint. The privacy model is useful beyond the specific fuzzy fingerprint problem studied. The detection is based on the fast set-intersection operation between the set of fingerprints generated from the payload of intercepted traffic (done by the DLD provider) and the set of fingerprints generated from the sensitive data (done by the data owner).
2. This paper implement detection system and perform extensive experimental evaluation on 2.6 GB Enron dataset, Internet surfing traffic of 20 users, and also 5 simulated real-world data-leak scenarios to measure the privacy guarantee, detection rate, and efficiency of proposed technique. The results indicate high accuracy performed by underlying scheme with very low false positive rate. It also shows that the detection

accuracy does not degrade when partial sensitive-data digests are used. In addition, these partial fingerprints fairly represent the fully set of data without any bias.

There are two technical challenges associated with network-based DLD detection. First, the DLD provider gains knowledge about the sensitive data when the traffic contains a leak. The challenge is how to restrict the degree of information that can be learned by the DLD provider in case of data leaks – the DLD provider has the access to the plaintext packet payload. The second challenge is how to make the detection noise-tolerant, for example, the intercepted packet payload may contain unrelated bytes or the sensitive data is truncated.

2. RELATED WORKS

Rabin fingerprint based on shingles was used previously for identifying similar spam messages in a collaborative setting, as well as collaborative worm containment, virus scan, Web template detection, and fragment detection.

This work fundamentally differs from the shingle based studies. Consider the new problem of data-leak detection in a unique outsourced setting where the DLD provider is not fully trusted. Such privacy requirement does not exist in the virus-scan paradigm, for the virus signatures are non-sensitive. In comparison, data-leak detection is more challenging because of the additional privacy requirement, which limits the amount of data that can be used during the detection and the amount of sensitive information gained by the DLD provider. In the meantime, the provider's detection accuracy cannot be compromised with partial digests based on the sensitive data. Fuzzy fingerprint method is new, and this work describes the first systematic solution to privacy preserving data-leak detection with convincing results.

Information leak through outbound web traffic was studied by Borders and Prakash [1]. Both works detect suspicious data flow on unencrypted network traffic. Their approach is based on the key observation that network traffic has high regularities and that information (e.g., header data) may be repeated. They proposed an elegant solution that detects any substantial increase in the amount of new information in the traffic.[10] Their anomaly-detection method detects deviations from normal data-flow scenarios, which are captured in rules. In comparison, this work inspects traffic for signatures of sensitive-data and does not require any assumption on the patterns of normal header fields or payload. Furthermore, solution provides privacy protection of the sensitive data against semi-honest DLD providers. This paper also gives performance evidences indicating the efficiency of the solution in practice.

The method of deep packet inspection is also widely used in network intrusion detection system. They focus on designing and implementing efficient string matching algorithms to handle short and flexible patterns in network traffic. However, NIDS is not designed for various kinds of sensitive data (e.g. long non-duplicated data), it may cause problems (e.g. large amount of states in an automata) in data leak detection scenarios. On the contrary, solution is not limited to very special types of sensitive data, and provides a unique privacy-preserving feature for service outsourcing. An alternative to this approach for privacy-preserving computation is to use cryptographic mechanisms.

Another category of approaches for data-leak detection is tracing and enforcing the sensitive data flows. The approaches include data flow and taint analysis [6], legal flow marking, and file-descriptor sharing enforcement [8]. These approaches are different from this paper because they

do not aim to provide an remote service. However, pure network-based solution cannot handle maliciously encrypted traffic [3], and these methods are complementary to our approach in detecting different forms (e.g., encrypted) of data leaks.

3. PROBLEM DEFINITION

According to current Statistics from various security organization research firms and government institutes suggest that there has been a rapid growth of data leak in past 8 years. There are various reasons for data leaks amongst which human errors is most endorsing of all. There are many ways in which this paper can have an audit trail verifying for data leak in networks, but still not all consider human errors as an important check factor, which makes them more prone to fail. Some of the common solutions include keeping a copy of sensitive data at the providers end and maintaining an audit trial for checking whether there is any data leakage in networks, which in turn notifies the organization about data leakage. All these methods in turn are prone to data attacks, as the keep copy of data while auditing. Also methods like deep packet analysis which searches for any relating data patterns. In this method, payloads of TCP/IP packets isanalyzed for any alter in data or any data pattern matching which may form sensitive data collection in network. This data is then compared with the threshold value, and if it exceeds the threshold value, the detection system alerts or notifies the organization.

From the detection perspective, a straightforward method is for the DLD provider to raise an alert if any sensitive fingerprint matches the fingerprints from the traffic. However, this approach has a privacy issue. If there is a data leak, there is a match between two fingerprints from sensitive data and network traffic. Then, the DLD provider learns the corresponding shingle, as it knows the content of the packet. Therefore, the central challenge is *to* prevent the DLD provider from learning the sensitive values even in data-leak scenarios, while allowing the provider to carry out the traffic inspection. This propose an efficient technique to address this problem. The main idea is to relax the comparison criteria by strategically introducing matching instances on the DLD provider's side without increasing false alarms for the data owner.

4. FUZZY FINGERPRINT METHOD

This paper describe the technical details of fuzzy fingerprint mechanism for privacy-preserving data-leak detection, by first introducing shingle and Rabin fingerprint, and then presenting randomization method for detection. The Rabin fingerprint scheme is a method for implementing fingerprints using polynomials over a finite field.

There are two players in this model: the organization (i.e. data owner) and the data-leak detection (DLD) provider.

- Organization owns the sensitive data and authorizes the DLD provider to inspect the network traffic from the organizational networks for anomalies, namely inadvertent data leak. However, the organization does not want to directly
  reveal the sensitive data to the provider.
- DLD provider inspects the network traffic for potential data leaks. The inspection can be performed offline without causing any real-time delay in routing the packets. However, the provider may attempt to gain knowledge about the sensitive data. This paper model

the DLD provider as a honest-but-curious adversary (aka semi-honest), who follows this protocols to carry out the operations, but may attempt to gain knowledge about the sensitive data.

The workflow in a network-based data-leak detection framework is as follows: DATA PRE-PROCESSING by the data owner, TRAFFIC PRE-PROCESSING AND DETECTION by the DLD provider, and ANALYSIS by the data owner. Data pre-processing is where the data owner takes the sensitive dataset and computes the corresponding set of digests. Traffic pre-processing and detection is where the DLD provider gathers network packets and inspects the content for data leaks. Analysis is where the data owner efficiently examines the alerts generated by the DLD provider, identifies and investigates the true leak instances and ignore false positives.

The privacy goal in our fuzzy fingerprint mechanism is to prevent the DLD provider from inferring the exact knowledge of the sensitive data; the DLD provider is given the fingerprints of sensitive data and the content of network traffic which may or may not contain data leak. In our model, this paper aim to hide the sensitive values among other nonsensitive values, so that the DLD provider is unable to pinpoint sensitive data among them even under data-leak scenarios. This paper define our privacy goal as follows, following the K-anonymity privacy definition in the relational databases

Our privacy goal is defined as follows. The DLD provider is given digests of sensitive data from the data owner and the content of network traffic to be examined. The DLD provider should not find out the exact value of a piece of sensitive data with more than 1 K probability, where K is an integer representing the number of all possible sensitive-data candidates that can be inferred by the DLD provider. This describe a novel fuzzy fingerprinting mechanism in the next section to improve the data protection against semi honest DLD provider, by utilizing simple and effective randomization technique in fingerprint generation. The privacy guarantee is much higher than 1 K when there is no leak in traffic, because the adversary's inference can only be done through brute-force guesses.This paper will propose the algorithmic steps with the help of below data flow diagram:

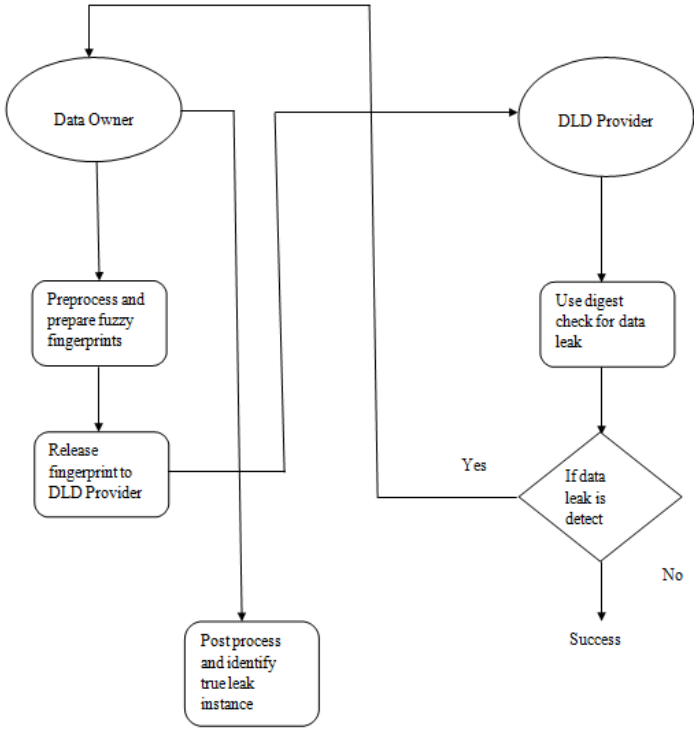

Figure 1. Data Flow Diagram

So, the above flow diagram can be explained as follows:

1. First the data owner will mark his set of sensitive data.
2. Secondly he will pre-compute all data and create a set of fuzzy fingerprint along with set of data digest.
3. He will add noise to the exposed digest, in order to assure that the semi honest provider does not gain complete knowledge about the sensitive data.
4. Then he will release the digest to the semi honest provider, to keep a track of any data leak detection in the network.
5. The DLD provider on receiving the digest, will start to check for any data leak using the digest.
6. If the provider finds any data leak in the current traffic network, he will notify it to the data owner.
7. The data owner on receiving the notification will post compute the data digest neglecting the noise he added, to check whether there was any data leakage in real time.

So, trying to give the data owner control rights i.e. he can decide what part of data to be revealed to the semi honest provider and which not to reveal. In this survey paper, propose a data leak detection framework which can be used as a semi honest provider in the network itself or can also be outsourced. In this system implementing a fuzzy finger print technique that is an additional security check parameter for data leakage method [2].

4.1 SHINGLES AND FINGERPRINTS

To achieve the privacy goal, the data owner generates a special type of digests, which call fuzzy fingerprints. Intuitively, the purpose of fuzzy fingerprints is to hide the true sensitive data in a crowd. It prevents the DLD provider from learning its exact value. The DLD provider obtains digests of sensitive data from the data owner. The data owner uses a sliding window and Rabin fingerprint algorithm to generate short and hard to-reverse (i.e., one-way) digests through the fast polynomial modulus operation. The sliding window generates small fragments of the processed data (sensitive data or network traffic), which preserves the local features of the data and provides the noise tolerance property. Rabin fingerprints [9] are computed as polynomial modulus operations, and can be implemented with fast XOR, shift, and table look-up operations. The Rabin fingerprint algorithm has a unique min-wise independence property, which supports fast random fingerprints selection (in uniform distribution) for partial fingerprints disclosure.

The shingle-and-fingerprint process is defined as follows. A sliding window is used to generate *q*-grams on an input binary string first. The fingerprints of *q*-grams are then computed.

A shingle (*q*-gram) is a fixed-size sequence of contiguous bytes. For example, the 3-gram shingle set of string abcdefgh consists of six elements {abc, bcd, cde, def, efg, fgh}. Local feature preservation is accomplished through the use of shingles. Therefore, this approach can tolerate sensitive data modification to some extent, e.g., inserted tags, small amount of character substitution, and lightly reformatted data. The use of shingles for finding duplicate web documents first appeared in [13] and [14].

This method proves to be faster than any another method and is based on one way computation of exposure of sensitive data. It gives the data owner rights of integrating data specific content securely to the DLD without actually exposing the sensitive data. So, this ensures that the semi honest provider has a very less amount of knowledge of the actual sensitive data and given provisions wherein individual can themselves mark their sensitive data and ask the admin of their local repository to check for any data leak. In the solution procedure, compute a method where the owner of the data contains a set of fingerprints or information digests of his own from the marked data, and can expose a small amount of part of the sensitive digest to the semi honest provider. The provider will then check for any data leak detection in that part of digest, where the digest is composed of real leaks and noise.

Using the min-wise independent property of Rabin fingerprint, the data owner can quickly disclose partial fuzzy fingerprints to the DLD provider. The purpose of partial disclosure is two-fold: *i)* to increase the scalability of the comparison in the DETECT operation, and *ii)* to reduce the exposure of data to the DLD provider for privacy. The method of partial release of sensitive data fingerprints is similar to the suppression technique in database anonymization.[11][12]

5. CONCLUSION

Preventing sensitive data from being compromised is an important and practical research problem. This paper proposed a fuzzy fingerprint framework and algorithms to realize privacy-preserving data-leak detection. Using special digests, the exposure of the sensitive data is kept to

a minimum during the detection. This paper described its application in the cloud computing environments, where the cloud provider naturally serves as the DLD provider. This paper defined privacy goal by quantifying and restricting the probability that the DLD provider identifies the exact value of the sensitive data. The extensive experiments validate the accuracy, privacy, and efficiency of the solutions.

REFERENCES

[1] X. Shu and D. Yao, "Data leak detection as a service," in Proc. 8th Int. Conf. Secur. Privacy Commun.Netw., 2012, pp. 222–240.

[2] Risk Based Security. (Feb. 2014). Data Breach Quick-View: An Executive's Guide to 2013 Data Breach Trends.[Online].Available:https://www.riskbasedsecurity.com/reports/2013-DataBreachQuickView.pdf, accessed Oct. 2014.

[3] Ponemon Institute. (May 2013). 2013 Cost of Data Breach Study: Global Analysis. [Online]. Available: https://www4.symantec.com/mktginfo/whitepaper/053013_GL_NA_WP_Ponemon-2013-Cost-of-a-Data-Breach-Report_daiNA_cta72382.pdf, accessed Oct. 2014.

[4] Identity Finder. Discover Sensitive Data Prevent Breaches DLP Data Loss Prevention. [Online]. Available: http://www.identityfinder.com/, accessed Oct. 2014.

[5] K. Borders and A. Prakash, "Quantifying information leaks in outbound web traffic," in Proc. 30th IEEE Symp. Secur. Privacy, May 2009, pp. 129–140.

[6] H. Yin, D. Song, M. Egele, C. Kruegel, and E. Kirda, "Panorama: Capturing system-wide information flow for malware detection and analysis," in Proc. 14th ACM Conf. Comput. Commun.Secur., 2007, pp. 116–127.

[7] K. Borders, E. V. Weele, B. Lau, and A. Prakash, "Protecting confidential data on personal computers with storage capsules," in Proc. 18th USENIX Secur. Symp., 2009, pp. 367–382.

[8] J. Kleinberg, C. H. Papadimitriou, and P. Raghavan, "On the value of private information," in Proc. 8th Conf. Theoretical Aspects Rationality Knowl., 2001, pp. 249–257.

[9] M. O. Rabin, "Fingerprinting by random polynomials," Dept. Math., Hebrew Univ. Jerusalem, Jerusalem, Israel, Tech. Rep. TR-15-81, 1981.

[10] S. Xu, "Collaborative attack vs. collaborative defense," in Collaborative Computing: Networking, Applications and Worksharing(Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering), vol. 10. Berlin, Germany: Springer- Verlag, 2009, pp. 217–228.

[11] G. Aggarwalet al., "Anonymizing tables," in Proc. 10th Int. Conf. Database Theory, 2005, pp. 246–258.

[12] R. Chen, B. C. M. Fung, N. Mohammed, B. C. Desai, and K. Wang,"Privacy-preserving trajectory data publishing by local suppression," Inf.Sci., vol. 231, pp. 83–97, May 2013.

[13] A. Z. Broder, "Some applications of Rabin's fingerprinting method," in Sequences II. New York, NY, USA: Springer-Verlag, 1993, pp. 143–152.

[14] A. Z. Broder, "Identifying and filtering near-duplicate documents," in Proc. 11th Annu. Symp.Combinat. Pattern Matching, 2000, pp. 1–10.

[15] GTB Technologies Inc. SaaS Content Control in the Cloud. [Online]. Available: http://www.gtbtechnologies.com/en/solutions/dlp_as a_service, accessed Oct. 2014.

[16] S. Geravand and M. Ahmadi, "Bloom filter applications in network security: A state-of-the-art survey," Comput. Netw., vol. 57, no. 18, pp. 4047–4064, Dec. 2013.

[17] B. Wang, S. Yu, W. Lou, and Y. T. Hou, "Privacy-preserving multikeyword fuzzy search over encrypted data in the cloud," in Proc. 33th IEEE Conf. Comput. Commun., Apr./May 2014, pp. 2112–2120.

AUTHORS

StaicyUlahannanis currently pursuing M.Techin Cyber Security in MBITS, Nellimattom. She completed her B. Tech. in Computer Science and engineering from MBITS, Nellimattom. Her areas of research are Network Security and Information Forensics

Roshni Jose is currently working as Assistant Professor in Department of Computer Science and Engineering in MBITS, Nellimattom. She received her B-Tech Degree in Computer Science and Engineering from College of Engineering, thodupuzha and M.Tech in Computer Science and engineering from Sathyabama Institute of Science & Technology. Her areas of research are Network Security and Computer Organisation.