A Review and Analysis of Privacy Preserving Detection Techniques of Sensitive Data Exposure

Vikram Shirol
Department of Computer Science & Engineering
SKSVMA College of Engineering & Technology, Laxmeshwar, Karnataka, India

Arunkumar Joshi
Department of Computer Science & Engineering
SKSVMA College of Engineering & Technology, Laxmeshwar, Karnataka, India

Abstract - Statistics from the different security firms, research institutions and government organizations show that there is a rapid growth in the numbers of data-leak instances in recent years. Among different data-leak cases, human mistakes are one of the main causes of data loss. There exist alternate solutions in detecting inadvertent sensitive data leaks caused by human mistakes and to provide alerts for organizations. One of the common approaches is to screen content in storage and transmission for exposed sensitive information. Such an approach usually requires secrecy in conducting the detection operation. However, this secrecy requirement is challenging to satisfy in practice, as detection servers may be compromised or outsourced. In this proposed research work, we present a privacy preserving data-leak detection (DLD) solution to solve the issue where a special set of sensitive data digests is used in detection. The advantage of this method is that, it enables the data owner to safely delegate the detection operation to a semi honest provider without revealing the sensitive data to the provider. It is also possible, how Internet service providers can offer their customers DLD as an add-on service with strong privacy guarantees. This proposal will help to support accurate detection with very less number of false alarms under different data-leak scenarios.

Keywords: Data Leak, DLD, Network Security, Privacy, Fuzzy

I. INTRODUCTION

As per the report analyzed by the Risk Based Security (RBS) [2], there is a dramatically increase in the number of leaked sensitive data records during the last few years, i.e., from 412 million in 2012 to 822 million in 2013. Deliberately planned attacks, inadvertent leaks (e.g., forwarding confidential emails to unclassified email accounts), and human mistakes (e.g., assigning the wrong privilege) lead to most of the data-leak incidents [3]. Data leak detection & preventing requires a set of complementary solutions, which may include data-leak detection [4], [5], data confinement [6]–[8], stealthy malware detection [9], [10], and policy enforcement [11]. Network data-leak detection (DLD) typically performs deep packet inspection (DPI) and searches for any occurrences of sensitive data patterns. DPI is a technique to analyze payloads of IP/TCP packets for inspecting application layer data, e.g., HTTP header/content. Alerts are triggered when the amount of sensitive data found in traffic passes a threshold. The detection system can be deployed on a router or integrated into existing network intrusion detection systems (NIDS). Straightforward realizations of data-leak detection require the plaintext sensitive data. However, this requirement is undesirable, as it may threaten the confidentiality of the sensitive information. If a detection system is compromised, then it may expose the plaintext sensitive data (in memory).
In addition, the data owner may need to outsource the data-leak detection to providers, but may be unwilling to reveal the plaintext sensitive data to them. Therefore, one needs new data-leak detection solutions that allow the providers to scan content for leaks without learning the sensitive information.

There have been several advances in understanding the privacy needs [12] or the privacy requirement of security applications [13]. In the proposed research work will try to identify the privacy needs in an outsourced data-leak detection service and provide a systematic solution to enable privacy-preserving DLD services.

A privacy preserving technique called Data-leak detection (DLD) is used to solve the issue where a special set of sensitive data digests are used in detection. Proposed method enables the data owner to safely delegate operation to a semi honest provider without providing the sensitive data to the provider. Here how Internet service providers can offer their customers DLD to improve the privacy guarantees is made.

Literature Survey: There have been several advances in understanding the privacy needs [12] or the privacy requirement of security applications [13]. In the proposed research work will help in identifying the privacy needs in an outsourced data-leak detection service and provide a systematic solution to enable privacy-preserving DLD services.

Shingle with Rabin fingerprint [14] was used previously for identifying similar spam messages in a collaborative setting [15], as well as collaborative worm containment [16], virus scan [17], and fragment detection [18].

In comparison, to tackle the unique data-leak detection problem in an outsourced setting where the DLD provider is not fully trusted. Such privacy requirement does not exist in above models, e.g., the virus signatures are non-sensitive in the virus-scan paradigm [18].

Most data-leak detection products offered by the industry, e.g., Symantec DLP [19], Global Velocity [20], Identity Finder [4], do not have the privacy-preserving feature and cannot be outsourced. GoCloudDLP [21] is a little different, which allows its customers to outsource the detection to a fully honest DLD provider.

Bloom filter [22] is a space-saving data structure for set membership test, and it is used in network security from network layer [23] to application layer [24]. The fuzzy Bloom filter invented in [25] constructs a special Bloom filter that probabilistically sets the corresponding filter bits to 1's. Although it is designed to support a resource-sufficient routing scheme, it is a potential privacy-preserving technique.

Besides fingerprint-based detection, other approaches can be applied to data-leak detection. If the sensitive data size is small and the patterns of all sensitive data are enumerable, string matching [26], [27] in network intrusion detection system can be used to detect data leaks. Privacy-preserving keyword search [28] or fuzzy keyword search [29] provide string matching approaches in semi-honest environments, but keywords usually do not cover enough sensitive data segments for data-leak detection.

Anomaly detection in network traffic can be used to detect data leaks [5] which will detects any substantial increase in the amount of new information in the traffic, and entropy analysis is used in [30]. We present a signature-based model to detect data leaks and focus on the design that can be outsourced, thus the two approaches are different.

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 ours because they do not aim to provide an remote service. However, pure network-based solution cannot handle maliciously encrypted traffic [31], and these methods are complementary to our approach in detecting different forms (e.g., encrypted) of data leaks.

Besides this proposed fuzzy fingerprint solution for data-leak detection, there are other privacy-preserving techniques invented for specific processes, e.g., DNA matching [32], or for general purpose use, e.g., secure multi-party computation (SMC). Similar to string matching methods discussed above, [32] uses anonymous automata to perform comparison. SMC [33] is a cryptographic mechanism, which supports a wide range of fundamental arithmetic, set, and string operations as well as complex functions such as knapsack computation, automated trouble-shooting, network event statistics [34], private information retrieval , genomic computation, private database query [35], private join operations, and distributed data mining [36]. The provable privacy guarantees offered by SMC comes at a cost in terms of computational complexity and realization difficulty. The advantage of our approach is its concision and efficiency.

II. PROPOSED METHODOLOGY

We propose the fuzzy fingerprint approach to meet the special privacy requirement and present the first systematic solution to privacy-preserving data-leak detection with convincing results.
A DLD (Data leak detection) technique is described in proposed solution. First what are Fuzzy fingerprints?
Fuzzy fingerprint is a technique to protect the data from DLD provider yet they do not cause additional false alarms for data owner as data owner can quickly distinguish between true and false leaks instance.
The fuzzy fingerprint includes:

Fuzzy length A fuzzy fingerprint f is given, fuzzy length d is the number of the least significant bits in fingerprint f that may be perturbed by the data owner, and d which is used to generate fingerprints is less than the degree of the polynomial.
Fuzzy set Given, a collection of fingerprints f and a fuzzy length d, the fuzzy set S (f,d) is the number of distinct collection of fingerprints whose values differ from f by at most $2^d - 1$.

We abstract the privacy-preserving data-leak detection problem with a threat model, a security goal and a privacy goal. First we describe the two most important players in our abstract 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 DLD provider may attempt to gain knowledge about the sensitive data.

A. Security Goal and Threat Model
We categorize three causes for sensitive data to appear on the outbound traffic of an organization, including the legitimate data use by the employees.
• Case I *Inadvertent data leak*: The sensitive data is accidentally leaked in the outbound traffic by a legitimate user. This paper focuses on detecting this type of accidental data leaks over supervised network channels. Inadvertent data leak may be due to human errors such as forgetting to use encryption, carelessly forwarding an internal email and attachments to outsiders, or due to application flaws (such as described in [37]).
• Case II *Malicious data leak*: A rogue insider or a piece of stealthy software may steal sensitive personal or organizational data from a host. Because the malicious adversary can use strong private encryption, steganography or covert channels to disable content-based traffic inspection, this type of leaks is out of the scope of our network-based solution.
Case III *Legitimate and intended data transfer*: The sensitive data is sent by a legitimate user intended for legitimate purposes. Assuming that the data owner is aware of legitimate data transfers and permits such transfers. So the data owner can tell whether a piece of sensitive data in the network traffic is a leak using legitimate data transfer policies.

B. Privacy Goal and Threat Model
To prevent the DLD provider from gaining knowledge of sensitive data during the detection process, we need to set up a privacy goal that is complementary to the security goal above. We model the DLD provider as a semi-honest adversary, who follows our protocol to carry out the operations, but may attempt to gain knowledge about the sensitive data of the data owner. 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 a probability greater than $1K$, where K is an integer representing the number of all possible sensitive-data candidates that can be inferred by the DLD provider. We present a privacy-preserving DLD model with a new fuzzy fingerprint mechanism to improve the data protection against semi-honest DLD provider. We generate digests of sensitive data through a one-way function, and then hide the sensitive values among other non-sensitive values via fuzzification. The privacy guarantee of such an approach

Overview of Privacy-Enhancing DLD
Our privacy-preserving data-leak detection method supports practical data-leak detection as a service and minimizes the knowledge that a DLD provider may gain during the process. The below Figure 1 lists the six operations executed by the data owner and the DLD provider in our protocol.

They include PREPROCESS run by the data owner to prepare the digests of sensitive data, RELEASE for the data owner to send the digests to the DLD provider, MONITOR and DETECT for the DLD provider to collect outgoing traffic of the organization, compute digests of traffic content, and identify potential leaks, REPORT for the DLD provider to return data-leak alerts to the data owner where there may be false positives (i.e., false alarms), and POSTPROCESS for the data owner to pinpoint true data-leak instances. Details are presented in the next section.

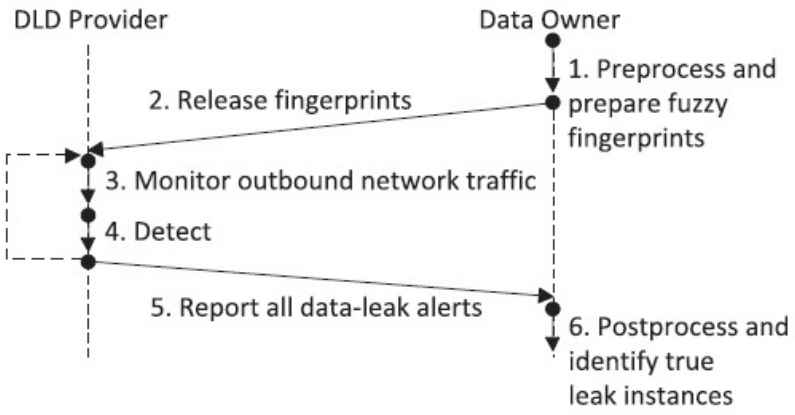

Figure 1. Privacy-preserving Data-Leak Detection Model

III. DESIGN CONSIDERATION

- Generate fingerprint for each sensitive data.
- Generate the token id for sensitive data.
- Release the fingerprint and reveal the small amount of data to the provider.
- DLD provider monitors the network traffic.
- Detect the data leaks.
- Report all data leak alerts to the data owner, it enables to identify the guilt agents.
- Data owner decide whether or not it is a true leak.

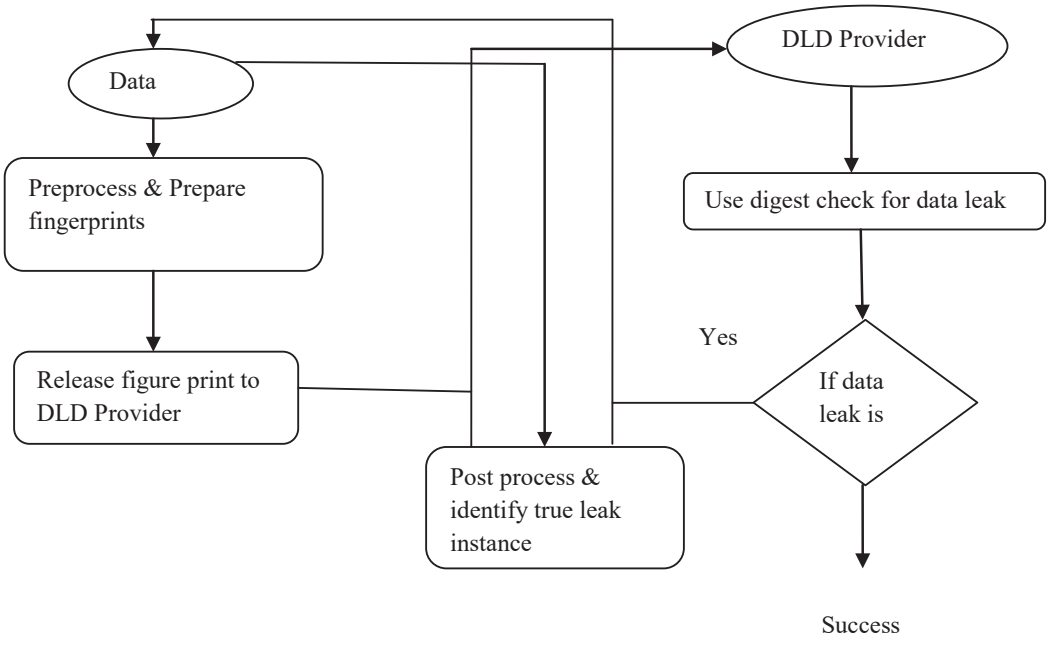

Figure 2. Data Flow Diagram

The data flow diagram shown above 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.

IV. CONCLUSION

The fuzzy fingerprint method differs from the other solutions and enables its adopter to provide data leak detection as a service. The customer or data owner does not need to fully trust the DLD provider using this approach.

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: DataBreachQuickView.pdf, accessed Oct. 2014.

[3] Ponemon Institute. (May 2013). *2013 Cost of Data Breach Study: Global Analysis*. [Online]. Available: ttps://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] A. Nadkarni and W. Enck, "Preventing accidental data disclosure in modern operating systems," in *Proc. 20th ACM Conf. Comput. Commun. Secur.*, 2013, pp. 1029–1042.

[9] A. Kapravelos, Y. Shoshitaishvili, M. Cova, C. Kruegel, and G. Vigna, "Revolver: An automated approach to the detection of evasiveweb-based malware," in *Proc. 22nd USENIX Secur. Symp.*, 2013, pp. 637–652.

[10] X. Jiang, X. Wang, and D. Xu, "Stealthy malware detection and monitoring through VMM-based 'out-of-the-box' semantic view reconstruction," *ACM Trans. Inf. Syst. Secur.*, vol. 13, no. 2, 2010, p. 12.

[11] G. Karjoth and M. Schunter, "A privacy policy model for enterprises," in *Proc. 15th IEEE Comput. Secur. Found. Workshop*, Jun. 2002, pp. 271–281.

[12] 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.

[13] 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.

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

[15] K. Li, Z. Zhong, and L. Ramaswamy, "Privacy-aware collaborative spam filtering," *IEEE Trans. Parallel Distrib. Syst.*, vol. 20, no. 5, pp. 725–739, May 2009.

[16] M. Cai, K. Hwang, Y.-K. Kwok, S. Song, and Y. Chen, "Collaborative Internet worm containment," *IEEE Security Privacy*, vol. 3, no. 3, pp. 25–33, May 2005.

[17] F. Hao, M. Kodialam, T. V. Lakshman, and H. Zhang, "Fast payloadbased flow estimation for traffic monitoring and network security," in *Proc. ACM Symp. Archit. Netw. Commun. Syst.*, Oct. 2005, pp. 211–220.

[18] L. Ramaswamy, A. Iyengar, L. Liu, and F. Douglis, "Automatic detection of fragments in dynamically generated web pages," in *Proc. 13th Int. Conf. World Wide Web*, 2004, pp. 443–454.

[19] Symantec. *Data Loss Prevention (DLP) Software*. [Online]. Available: http://www.symantec.com/data-loss-prevention/, accessed Oct. 2014.

[20] Global Velocity Inc. *Cloud Data Security From the Inside Out*. [Online]. Available: http://www.globalvelocity.com/, accessed Oct. 2014.

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

[22] A. Broder and M. Mitzenmacher, "Network applications of bloom filters: A survey," *Internet Math.*, vol. 1, no. 4, pp. 485–509, 2004.

[23] 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.

[24] 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.

[25] C. P. Mayer, "Bloom filters and overlays for routing in pocket switched networks," in *Proc. 5th Int. Student Workshop Emerg. Netw. Experim. Technol.*, 2009, pp. 43–44.
[26] A. V. Aho and M. J. Corasick, "Efficient string matching: An aid to bibliographic search," *Commun. ACM*, vol. 18, no. 6, pp. 333–340, 1975.
[27] P.-C. Lin, Y.-D. Lin, Y.-C. Lai, and T.-H. Lee, "Using string matching for deep packet inspection," *IEEE Comput.*, vol. 41, no. 4, pp. 23–28, Apr. 2008.
[28] S. Ananthi, M. Sadish Sendil, and S. Karthik, "Privacy preserving keyword search over encrypted cloud data," in *Advances in Computing and Communications* (Communications in Computer and Information Science), vol. 190. Berlin, Germany: Springer-Verlag, 2011, pp. 480–487.
[29] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "Fuzzy keyword search over encrypted data in cloud computing," in Proc. 29th IEEE Conf. Comput. Commun., Mar. 2010, pp. 1–5.
[30] T. W. Fawcett, "ExFILD: A tool for the detection of data exfiltration using entropy and encryption characteristics of network traffic," M.S. thesis, Dept. Elect. Comput. Eng., Univ. Delaware, Newark, DE, USA, 2010.
[31] V. Varadharajan, "Internet filtering—Issues and challenges," *IEEE Security Privacy*, vol. 8, no. 4, pp. 62–65, Jul./Aug. 2010.
[32] J. R. Troncoso-Pastoriza, S. Katzenbeisser, and M. Celik, "Privacy preserving error resilient dna searching through oblivious automata," in *Proc. 14th ACM Conf. Comput. Commun. Secur.*, 2007, pp. 519–528.
[33] A. C.-C. Yao, "How to generate and exchange secrets," in *Proc. 27th Annu. Symp. Found. Comput. Sci.*, 1986, pp. 162–167.
[34] M. Burkhart, M. Strasser, D. Many, and X. Dimitropoulos, "SEPIA: Privacy-preserving aggregation of multi-domain network events and statistics," in *Proc. 19th USENIX Conf. Secur. Symp.*, 2010, p. 15.
[35] X. Yi, M. G. Kaosar, R. Paulet, and E. Bertino, "Single-database private information retrieval from fully homomorphic encryption," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 5, pp. 1125–1134, May 2013.
[36] G. Jagannathan and R. N. Wright, "Privacy-preserving distributed *k*-means clustering over arbitrarily partitioned data," in *Proc. 11th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2005, pp. 593–599.
[37] J. Jung, A. Sheth, B. Greenstein, D. Wetherall, G. Maganis, and T. Kohno, "Privacy oracle: A system for finding application leaks with black box differential testing," in *Proc. 15th ACM Conf. Comput. Commun. Secur.*, 2008, pp. 279–288.