IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 8, Issue 8, August 2020

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.488

9940 572 462 | 6381 907 438 | ijircce@gmail.com | www.ijircce.com

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | Impact Factor: 7.488 |

|| Volume 8, Issue 8, August 2020 ||

Enabling Secure and Effective Near Duplicate Detection Over Encrypted in Network Storage

Menka Yadav[1], Yatin Agarwal[2]

M. Tech Student, Department of CSE, Greater Noida Institute of Technology, Greater Noida, Dr. APJ Abdul Kalam Technical University, Lucknow, India[1]

Asst. Professor, Department of CSE Greater Noida Institute of Technology, Greater Noida, Dr. APJ Abdul Kalam Technical University, Lucknow, India[2]

ABSTRACT: In many evolving network architectures, near-duplicate detection (NDD) plays an important role in efficient resource utilization and potential traffic alleviation, leveraging in-network capacity for various content-centered services. When data protection is growing in network storage, it has become one big concern. While encryption is feasible for in-network data security, current techniques are still lacking to effectively locate near-duplicate encrypted data, thus essentially invalidating the benefits of NDD. Additionally, implementing encryption in network storage further complicates user authorization when finding near-duplicate data under separate keys from multiple service providers. In this paper we propose a stable and efficient NDD framework that supports multiple service providers over encrypted in-network storage. Our architecture bridges locality sensitive hashing (LSH) with a newly built primitive, multi-key searchable encryption that enables the user to submit only one encrypted question to access near-duplicate data encrypted under various keys. This relieves the users either from several rounds of interactions or sends several separate queries. Since simply applying LSH does not ensure the detection accuracy, we then exploit Yao's garbled circuits to create a safe protocol without user-side post-processing to obtain highly accurate results. We formally evaluate the strength of the defense. Experiments show our program achieves realistic efficiency with comparable plaintext accuracy.

KEYWORDS: near duplicate detection, network storage

I.INTRODUCTION

As indicated by EMC, the volume of digital data we are creating will grow exponentially in the coming years to reach 44 zettabytes by 2020. To cope with such an immense amount of data, evolving network architectures, such as NDN and CDN, are proposed for content-centered applications where the data is stored on global distributed in-network server. Thus, detecting duplicate data, an important technique in modern networked and storage systems, is inevitably transferred from original data service providers to in-network servers in order to further reduce network traffic and delay in transmission. At the other hand, many forms of data today, such as images , videos, and web pages, are commonly seen to contain "essentially the same" content, but vary in formats, encodings, or edits. As conventional deduplication via exact matching of bit-streams can hardly identify them, techniques for effectively detecting such near-duplicate data may be absolutely necessary for high-quality content-centric services, e.g. delivery and sharing of multimedia content.

As more data is cached on in-network servers, they also become a high-value option for internal as well as external attacks. Content owners want good protection of copyrighted data against unauthorized access to or misuse of content, although consumers do not want eavesdropping of the accessed data. Although encrypting the data before installing it in the network is a feasible solution for resolving security issues, it will specifically invalidate all the advantages of near-duplicate plaintext detection, prohibiting users from accessing near-duplicates from near-by in-network servers. Furthermore, another important condition when using encrypted in-network storage is to keep the user experience essentially the same as in the plaintext domain. That is, the network will react autonomously with the resulting near-duplicate data when a user sends a query, preferably without multiple rounds of user interactions or multiple user queries subsequently sent. However, this does not seem simple under a many-to-many encrypted scenario, under which each in-network server will host encrypted data from multiple service providers under different keys when processing queries from different users.

II.LITERATURE SURVEY

The NDD has been widely reviewed in the literature in recent years. In general, NDD uses data-dependent features to classify the data objects, doing this by matching features. International feature-based solutions simply use a compact fingerprint for one data element, e.g. color histogram, while various local features, e.g., SIFT, are implemented in the other. A recent research by Hua et al. introduced the idea of in-network de duplication for SDN, which removes duplicate data inside the network by verifying the SDN controller data fingerprints. In, they introduced a near-duplicate detection scheme in-network to enable the recovery of photos in the sense of disaster relief. The above work does operate in the plaintext domain despite being very useful. There is also a line of similar designs to research the safe deduplication over encrypted data (to name only a few).
In Bellare et al. the DupLESS proposed allowing for safe deduplication over encrypted documents with resistance to off-line brute-force attacks. Later, Zheng et al . developed a safe layer-level deduplication framework for stable and efficient video transmission over encrypted scalable video encoding images. When considering privacy security, these designs are all based on removing duplicates by exact matching of fingerprints. In comparison, our research is aimed at a more general scenario, that is, stable NDD. Our work also includes multi-user searchable encryption, which allows multi-user applications to search over encrypted data
To expand symmetric key-based searchable encryption in multi-user contexts, Curtmola et al. suggested using broadcast encryption to share a randomness for the authentication of search keys for all legal users. Jarecki and. Al.proposed to allow legal users to get the data owner's search tokens if they wish to search. While these systems allow multiple user searches, they allow only one data owner. Namely, when there are multiple data owners, users need multiple search tokens for a given query. At the other hand, the multi-user environment is also being explored by asymmetric key based approaches. Boneh et al scheme supporting multiple proprietors of info. We will scan and decrypt the data as long as the user has the private key. Similarly, if she tries to scan several databases, it also needs multiple tokens.

III.RESEARCH METHODOLOGY

We propose a safe and efficient framework allowing in-network servers to locate near-duplicates encrypted for approved users through multiple content providers. We build and extend an approved NDD algorithm over encrypted data running on a single server, by proposing a protocol across distributed servers in the network. To obtain accurate detection results, we also design and implement a secure two-party computation protocol based on Yao's garbled circuits. In a cloud-based test framework, we systematically evaluate the security intensity and incorporate all components. Extensive tests on real-world data set show that our program is achieving.

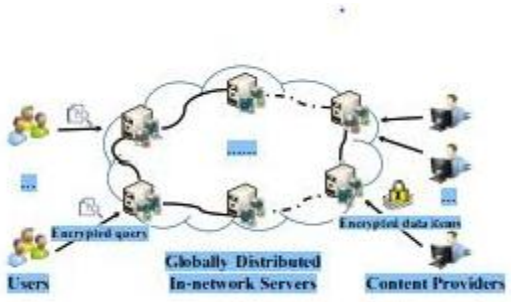

Fig. 1. Architecture diagram

The infrastructure is composed of three major entities: service providers (abbr. CPs), in-network servers (abbr. ISs), and users, as shown in Fig. 1. In a realistic situation, CPs aim to outsource the data for high quality web hosting and distribution services to internationally dispersed ISs. Because ISs are usually implemented by third-party service providers in untrusted network environments, such as CDN and Cloud, CPs must encrypt the data against data leakage and unauthorized access. We also demand ISs for bandwidth efficiency to conduct stable and accurate NDD over encrypted in-network storage.
A certain IS, on the other hand, can host data items from different CPs that are encrypted with their own keys. Consequently, the implementation of existing powerful searchable encryption schemes[12],[19] for secure NDD would require the user to generate multiple encrypted queries if separate CPs approve it. Furthermore, if the near-duplicate data is not located "nearby," all the questions will be propagated to the next IS, or resent. The above inefficiency can also be discussed from the user 's

viewpoint. Our architecture aims to enable the user to only submit one encrypted query, comparable to the situation in the plaintext domain, whereas protected NDD may still be performed over encrypted data items from various CPs. Our system's operation flow consists of the three phases:

1. Preparation phase
2. Detection phase
3. Evaluation phase

1) Preparation phase: In this phase, two procedures are executed. (1) The CPs encrypt the data items and prepare the cipher text metadata fcg, where c is derived from each data item, and used for later secure NDD. We use "fg" to represent a collection. The network service provider assigns the encrypted data items with fcg to ISs that are close to the users.
(2) The CPs authorize their users so that fcg is accessible to legal users. Explicitly, each user generates her own key, and each CP generates an authorization digest for every user from the user key. As long as an IS hosts the encrypted data from a certain CP, the corresponding's for users of that CP are transmitted to the IS. Here, we emphasize that the above authorization is fully distributed without relying on any trusted authority for key distribution.

2) Detection phase: The user generates an encrypted query q from the data of interest via her own key, and sends it to "nearby" IS. The IS will first check whether the user has access privileges, i.e., authorization digests _s, to access the data from different CPs. If "a" exists, tq will be transformed into the form that can be tested with the cipher text metadata fcg from the corresponding CP. When a match is found, the encrypted data item will be considered as a near-duplicate candidate.Otherwise, tq will be forwarded to the next IS until locating sufficient near-duplicate candidates. In this phase, the user just needs to send one tq only, which will be forwarded and transformed by ISs without any interaction with the user.

3) Evaluation phase: To eliminate the false positives from the initial detection results, a secure two-party computation protocol is initiated between the corresponding IS and a garbled-circuit generator. Here, the generator can be a server from another service provider, which does not collude with the IS. Specifically, the generator prepares a garbled circuit for the IS, which securely evaluates whether the distances between the encrypted fingerprints of candidates and the query data item are within a pre-defined threshold. The candidates that satisfy the evaluation will be directly sent back to the user as the final near-duplicate results.

IV.CONCLUSION

In this paper we propose a stable and efficient NDD framework that supports multiple service providers over encrypted in-network storage. Our architecture bridges LSH with a newly built cryptographic primitive MKSE, which enables the user to send only one encrypted query to access near-duplicate data encrypted under different keys, to relieve users from multiple rounds of interactions or to send several different queries.

In addition, we exploit a stable two-party computation protocol based on Yao's garbled circuits to ensure the accuracy of stable NDD to obtain highly accurate results without user-side post-processing. We formally evaluate the strength of the defense. Extensive tests on real-world data set show that our program achieves realistic efficiency with comparable plaintext precision. We will carry out detailed security review of the system as future research. We will explore the potential extension in other practical scenarios, such as querying a targeted in-network server, and even speed-up performance through batch processing.

REFERENCES

[1] IDC, "Executive Summary: Data Growth, Business Opportunities, and IT Imperatives," Online at http://www.emc.com/leadership/digital- universe/2014iview/executive-summary.htm.
[2] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, P. Crowley, C. Papadopoulos,L. Wang, B. Zhang et al., "Named data networking," ACM SIGCOMM Computer Comm. Review, vol. 44, no. 3, pp. 66–73, 2014.
[3] Akamai, "Akamai," Onlineathttps://www.akamai.com/, 2015.
[4] Y. Hua, X. Liu, and D. Feng, "Smart in-network deduplication for storage-aware sdn," in Proc. of ACM SIGCOMM, 2013.
[5] Y. Hua, W. He, X. Liu, and D. Feng, "Smarteye: Real-time and efficient cloud image sharing for disaster environments," in Proc. of IEEE INFOCOM, 2015.

[6] Y. Ke, R. Sukthankar, and L. Huston, "Efficient near-duplicate detection and sub-image retrieval," in Proc. of ACM MM, 2004.
[7] B. Thomee, M. J. Huiskes, E. Bakker, and M. S. Lew, "Large scale image
copy detection evaluation," in Proc. of ACM international conference on Multimedia information retrieval, 2008.
[8] X. Wu, A. G. Hauptmann, and C.-W. Ngo, "Practical elimination of near- duplicates from web video search," in Proc. of ACM MM, 2007.
[9] X. Yuan, X. Wang, C. Wang, A. Squicciarini, and K. Ren, "Enabling privacy-preserving image-centric social discovery," in Proc. of IEEE ICDCS, 2014.
[10] W. Sun, S. Yu, W. Lou, Y. T. Hou, and H. Li, "Protecting your right: Attribute-based keyword search with fine-grained owner-enforced search authorization in the cloud," in Proc. of IEEE INFOCOM, 2014.
[11] A. Gionis, P. Indyk, R. Motwani et al., "Similarity search in high dimensions via hashing," in Proc. of VLDB, 1999.
[12] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: improved definitions and efficient constructions," in Proc. of ACM CCS, 2006.
[13] M. Kuzu, M. Islam and M. Kantarcioglu, "Efficient similarity search over encrypted data," in Proc. of IEEE ICDE, 2012.
[14] H. Cui, X. Yuan, and C. Wang, "Harnessing encrypted data in cloud for secure and efficient image sharing from mobile devices," in Proc. Of IEEE INFOCOM, 2015.
[15] M. Li, S. Yu, N. Cao, and W. Lou, "Authorized private keyword search over encrypted data in cloud computing," in Proc. of IEEE ICDCS, 2011.
[16] R. A. Popa and N. Zeldovich, "Multi-key searchable encryption," IACR Cryptology ePrint Archive, 2013.
[17] C. Liu, X. S. Wang, K. Nayak, Y. Huang, and E. Shi, "Oblivm: A programming framework for secure computation," in Proc. of IEEE Security and Privacy (S&P), 2015.
[18] V. Nikolaenko, U. Weinsberg, S. Ioannidis, M. Joye, D. Boneh, and N. Taft, "Privacy-preserving ridge regression on hundreds of millions of records," in Proc. of IEEE Security and Privacy (S&P), 2013.
[19] S. Jarecki, C. Jutla, H. Krawczyk, M. Rosu, and M. Steiner, "Outsourced symmetric private nformation retrieval," in Proc. of ACM CCS, 2013.
[20] R. A. Popa, E. Stark, J. Helfer, S. Valdez, N. Zeldovich, M. F. Kaashoek, and H. Balakrishnan, "Building web applications on top of encrypted data using mylar," in Proc. of USENIX NSDI, 2014.
[21] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in Proc. of EUROCRYPT, 1999.
[22] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Dupless: Server-aided encryption for deduplicated storage," in Proc. of USENIX Security, 2013.
[23] Y. Zheng, X. Yuan, X. Wang, J. Jiang, C. Wang, and X. Gui, "Enabling encrypted cloud media center with secure deduplication," in Proc. Of ASIACCS, 2015.
[24] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in Proc. of EUROCRYPT, 2004.
[25] F. Bao, R. H. Deng, X. Ding, and Y. Yang, "Private query on encrypted data in multi-user settings," in Proc. of ISPEC, 2008.
[26] C. Dong, G. Russello, and N. Dulay, "Shared and searchable encrypted data for untrusted servers," Journal of Computer Security, vol. 19, no. 3, pp. 367–397, 2011.

INNO SPACE

SJIF Scientific Journal Impact Factor

Impact Factor:
7.488

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details