Privacy-preserving for Offloading Services on Cloud

Chuan Pham, and Choong Seon Hong*

Department of Computer Science and Engineering, Kyung Hee University

Abstract: Offload computation has been a promising research area for recent years and provided the capability to extend mobile resource limitations in terms of CPU, GPU, memory, storage and battery. Without the burden of local resources, offloading services open appealing advantages to mobile users and cloud service providers. Even though offloading data on Cloud makes more challenges, it also faces to many challenges in securing user's data, such as separately controlling in cloud service providers, the mobility of mobile users, the limitation of mobile resources, transparent requirements in security implementations. In this paper, we study about the lightweight privacy-preserving for offloading services on Cloud that we apply the third party auditor model to authenticate and audit the offloading services. The simulation results show the efficiency of our model in comparison with the existing SSL method.

1. Introduction

Cloud computing has enabled to provide offloading services at both personal and business level with many advantages. Employing computational offloading, Mobile Cloud computing opens intensive mobile applications on smart mobile devices. From users' perspectives, offloading computation brings appealing benefits, such as lack of local physical resources, avoidance of implementation cost on hardware, software.

Even though Cloud computing owns these advantages more appealing than ever, it faces to many challenges, especially in security of users' outsourced data. In particular, data offloading is actually relinquishing user's ultimate control over the fate of their data [1]. Moreover, cloud service providers might reclaim resources for monetary reasons by discarding data that has not been or is rarely accessed. As users no longer possess their resources on cloud, traditional cryptographic primitives cannot be adopted to protect their data. In particular, with a large data needed to offload on cloud, the tasks of auditing the data correctness can be really expensive and unacceptable for users.

Moreover, the overhead of performing security for offloading tasks of users should be carefully considered when the cloud service providers must perform too

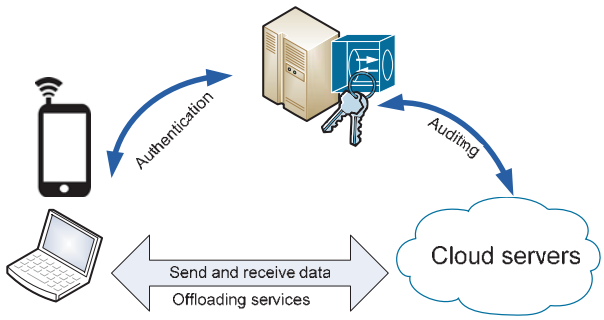

Fig.1. System model of the third party auditor in privacy-preserving for offloading services.

many operations for those security services. Especially in the environment of mobile cloud, requests are diverse and complex, we need an efficient mechanism to secure offloading data of users.

Considering existing approaches in offloading computation that are integrated in Cloud providers, such as using asymmetric encryption to protect data [2], user-group-based data encryption [3], we still need an efficient mechanism to implement on Cloud providers because of limitations as follows: i) the limited resources on mobile devices (e.g., CPU, memory,

Dr. CS Hong is the corresponding author.

battery) ii) the mobility of mobile devise makes the diverse connections that impact directly to the authentication phase before and during offloading tasks iii) When using different applications, users have to install many authentication services that require various methods in implementation.

In this paper, we study a mechanism that uses for security services in offloading computation for privacy of users' data via an independent third party auditor, who owns expertise and capabilities to provide security services for cloud providers and users. Based on third party auditor, users can dynamically interact with the Cloud services to offload their tasks for various application purposes as free worry. While Cloud services can resort to such a third party auditor for ensuring the integrity of their services and keeping users' data privately.

The rest of the paper is organized as follows. Section 2 discusses about the problem statement and the solution used in our work. Section 3 presents about the simulation results. Finally, we present the conclusion in Section 4.

2. Problem Statement

2.1. System Model

In this paper, we consider a cloud service provider scenario as shown in Fig. 1, including the *cloud users*, the *cloud providers* and the *auditor*. The cloud providers provide offloading services to users, where offloading tasks can be executed on the resource pool of cloud. The auditor is the third party auditor who users subscribe to secure their offloading data. On behalf of the auditor, cloud users are trusted to offload their tasks without any worry about their data. Our protocol design for privacy-preserving in offloading services is as follows:

- *Setup authentication*: The user has to sign a contract to the auditor, who will initialize the secret key for mobile users by executing a key generation procedure. The secret key will be added into the offloading data.
- Audit: This procedure is executed to ensure the cloud server has retained the data of the offloading tasks properly at the time of the audit. Since the mobility of mobile users, the network connection of offloading tasks may be lost in some senses. By managing the key of users, the auditor can verify and resume the offloading services of users when the connection of users are reacted.

2.2. HLA-based Solution

In this work, we apply the homomorphic linear authenticators (HLA) approach [4] in our model, a well-known approach in security issues. For ease understanding, we explain a toy example for our model as in Fig. 2.

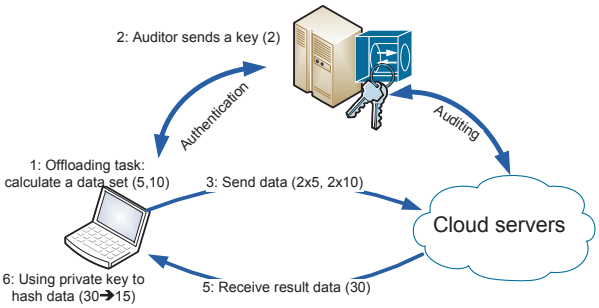

Fig. 2: Example of privacy-preserving for offloading task on Cloud based on HLA solution.

User A has a big data set, which he wants to offload on cloud servers for calculation. His data includes two blocks data 5 and 10. Based on the auditor, who sends to him a key for encryption, before offloading data, he encrypts all his data by multiplying his data with his key, as shown in step 3. On cloud servers, without the knowledge about the real data of user A, they calculate then replies the result to user A as step 5. At the client side, user A decrypts the result by his private key and gets the result.

Based on the HLA approach, we denote $F = \{f_1, f_2, \dots, f_n\}$ is a set of block data of the offloading task. From the generation key of the auditor, a random tag $V = \{v_1, v_2, \dots, v_n\}$ is sent to users corresponding block i. Each block f_i can be added a key to secure the data, then offloaded to Cloud servers. The client can issue some indices of the blocks, and the server responds with the corresponding block-tag pairs ($\mu = \sum_i v_i f_i, \sigma = \sum_i v_i \sigma_i$). The client then verify the validity of those block-tag pairs with his secret key, and accept the blocks if those pairs pass verification. The valid blocks can be fed to an extraction algorithm, which is a

decoder of erasure codes, to extract the whole original data F. Given the prover's response (μ', σ'), the verifier (client) is able to use the authentication key to check whether μ holds or not with overwhelming probability. Using some simple hash functions [3] that can encrypt and decrypt data without heavy computation requirement, user A can offload the meta data to cloud. When the response result are received, the client invokes the decryption to obtain the real outcome from the cloud server. Using HLA approach, the privacy-preserving procedure can send the metadata on the legacy network with a reliability and lightweight mechanism.

Design goal. To implement the privacy-preserving for data offloading in Cloud, our design targets to achieve the following security and performance guarantees as follows:

- To allow users to offload their tasks without installing many security services from the cloud providers that require various security methods depending on offloading services.
- To allow the cloud provider executing and handling their services without any worry about updating the current security services.
- To ensure that there exists no cheating cloud server that can explore the user data that are located and executed on the physical devices of cloud providers.
- To enable a lightweight and reliable mechanism to users.

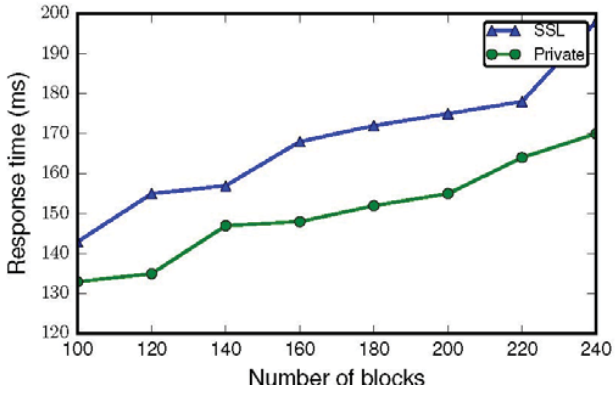

Fig. 4: Evaluation of response time in offloading data.

3. Simulation results

We evaluate our proposed model by analyzing the execution time that we want to show the lightweight mechanism. The experiment is conducted using Python on Ubuntu system with Intel Core i5 processor running 2.67GHz, 4GB of RAM and 7200 RPM Western Digital 500GB Serial ATA drive as a Cloud server. To simulate for the Audit, we setup a LAN network (network bandwidth 100Mbps) and another Virtual machine that creates random keys. For the client, we also create another virtual machine that offloads to Cloud server with 100 to 240 block text files containing 1000 random numbers. For calculating the offloading task (e.g., summation), our proposed model is faster than using SSL [5], as shown in Fig.3.

4. Conclusion

In this paper, we have studied about privacy-preserving in offloading services on Cloud. We have proposed a third party authentication to secure offloading data of users in terms of reliable and lightweight. The simulation result with a testbed showed the efficiency of our method in response time comparing with private security SSL method.

ACKNOWLEDGMENT

"This research was supported by the MSIP, Korea, under the G-ITRC support program (IITP-2016-R6812-15-0001) supervised by the IITP." *Dr. CS Hong is the corresponding author.

REFERENCES

[1] Jiao, Wenzhe, et al. "Dynamic data possession checking for secure cloud storage service." *Journal of Networks* 8.12 (2013): 2713-2720.

[2] E. Goh, H. Shacham, N. Modadugu, and D. Boneh. Sirius: Securing remote untrusted storage. In Proceedings of the Internet Society (ISOC) Network and Distributed Systems Security (NDSS) Symposium, pages 131–145, 2003

[3] S. D. C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati. Over-encryption: management of access control evolution on outsourced data. In Proceedings of the international conference on Very large data bases, pages 123–134, 2007

[4] Liu, Shengli, and Kefei Chen. "Homomorphic Linear Authentication Schemes for Proofs of Retrievability." *Intelligent Networking and Collaborative Systems (INCoS), 2011 Third International Conference on*. IEEE, 2011.

[5] SSL [Online] https://en.wikipedia.org/wiki/SSL.