International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 04 Issue 03
March 2017

A Novel Authenticated key exchange Parallel session's using Forward secrecy

GUFRANUL KARIM[1], SHAHEENA GHAZALA[2], MD AFROJ[3]& SHAIK MAHEBOOB[4]

[1]B-Tech, Lords Institute Of Engineering And Technology, Hyderabad, TS.
[2]B-Tech, Lords Institute Of Engineering And Technology, Hyderabad, TS.
[3]B-Tech, Lords Institute Of Engineering And Technology, Hyderabad, TS.
[4] Assistant professor, Lords Institute of Engineering and Technology, , Hyderabad, TS.

ABSTRACT

The problem is inspired by the proliferation of large-scale distributed file systems supporting parallel access to multiple storage devices. Our work focuses on the current Internet standard for such file systems, i.e., parallel Network File System which makes use of Kerberos to establish parallel session keys between clients and storage devices. Our review of the existing Kerberos-based protocol shows that it has a number of limitations. In this paper, we propose a variety of authenticated key exchange protocols that are designed to address the above issues. We show that our protocols are capable of reducing up to approximately of the workload of the metadata server and concurrently supporting forward secrecy and escrow-freeness. All this requires only a small fraction of increased computation overhead at the client.

Keywords: - Parallel sessions, Authenticated key exchange, Forward secrecy.

1. INTRODUCTION

In a parallel file system, file data is distributed across multiple storage devices or nodes to allow concurrent access by multiple tasks of a parallel application. This is typically used in large-scale cluster computing that focuses on high performance and reliable access to large datasets. That is, higher I/O bandwidth is achieved through concurrent access to multiple storage devices within large compute clusters; while data loss is protected through data mirroring using fault-tolerant striping algorithms. Some examples of highperformance parallel file systems that are in production use are the IBM General Parallel File System

(GPFS) , Google File System (GoogleFS) , Lustre , Parallel Virtual File System (PVFS) , and Panasas File System ; while there also exist research projects on distributed object storage systems such as Usra Minor , Ceph , and Gfarm . These are usually required for advanced scientific or data-intensive applications such as, seismic data processing, digital animation studios, computational fluid dynamics, and semiconductor manufacturing. In these environments, hundreds or thousands of file system clients share data and generate very high aggregate I/O load on the file system supporting petabyte- or terabyte-scale storage capacitiesvices. In this work, we investigate the problem of secure manyto-many communications in large-scale network file systems that support parallel access to multiple storage devices. That is, we consider a communication model where there are a large number of clients (potentially hundreds or thousands) accessing multiple remote and distributed storage devices (which also may scale up to hundreds or thousands) in parallel. Particularly, we focus on how to exchange key materials and establish parallel secure sessions between the clients and the storage devices in the parallel Network File System (pNFS) —the current Internet standard—in an efficient and scalable manner. The development of pNFS is driven by Panasas and thus it shares many common features and is compatible with many existing commercial/proprietary network file systems.

2. RELEGATED WORK

Existing system

Study the problem of key establishment for secure many-to-many communications. The problem is inspired by the proliferation of large-scale distributed file systems supporting parallel access to multiple storage devices. Our work focuses on the current Internet standard for such file systems, i.e., parallel Network File System (pNFS), which makes use of Kerberos to establish parallel session keys between clients and storage devices. Our review of the existing Kerberos-based protocol shows that it has a number of limitations: (i) a metadata server facilitating key exchange between the clients and the storage devices has heavy workload that restricts the scalability of the protocol; (ii) the protocol does not provide forward secrecy; (iii) the metadata server generates itself all the session keys that are used between the

clients and storage devices, and this inherently leads to key escrow.

Proposed system

We propose a variety of authenticated key exchange protocols that are designed to address the above issues. We show that our protocols are capable of reducing up to approximately of the workload of the metadata server and concurrently supporting forward secrecy and escrow-freeness. All this requires only a small fraction of increased computation overhead at the client.

Advantages

Finally, in the last augmented game, we can claim that the adversary has no advantage in winning the game since a random key is returned to the adversary. Our protocols offer three appealing advantages over the existing Kerberos-based pNFS protocol.

3. IMPLEMENTATION

Parallel sessions

Parallel secure sessions between the clients and the storage devices in the parallel Network File System (pNFS) The current Internet standard—in an efficient and scalable manner. This is similar to the situation that once the adversary compromises the long-term secret key, it can learn all the subsequence sessions. If an honest client and an honest storage device complete matching sessions, they compute the same session key. Second, two our protocols provide forward secrecy: one is partially forward secure with respect to multiple sessions within a time period.

Authenticated key exchange

Our primary goal in this work is to design efficient and secure authenticated key exchange protocols that meet specific requirements of pNFS. The main results of this paper are three new provably secure authenticated key exchange protocols. We describe our design goals and give some intuition of a variety of pNFS authenticated key exchange6 (pNFS-AKE) protocols that we consider in this work

Forward secrecy

The protocol should guarantee the security of past session keys when the long-term secret key of a client or a storage device is compromised. However, the protocol does not provide any forward secrecy. To address key escrow while achieving forward secrecy simultaneously, we incorporate a Diffie-Hellman key agreement technique into Kerberos-like pNFS-AKE-I. However, note that we achieve only partial forward secrecy with respect to v), by trading efficiency over security.

International Journal of Research
Available at https://edupediapublications.org/journals
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 04 Issue 03
March 2017

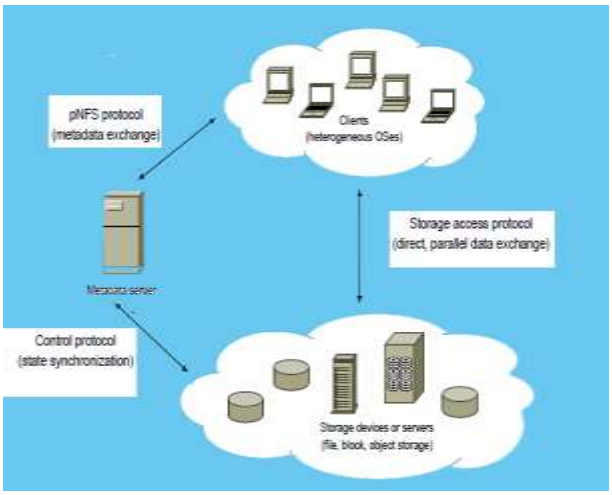

Fig:-1 System Architecture

Client

Share Data

The user can share their data into another user in same group the data will translate by path setting data.

Upload Data

The user can upload the file to cloud. And the Admin can allow the data to store the cloud.

Download File

The user also downloads the cloud file by the conditions.

Server Authentication

Accept user

The admin can accept the new user request and also black the users.

Allow user file

The users can upload the file to cloud. And the admin can allow the files to cloud then only the file can store the cloud.

4. EXPERIMENTAL RESULTS

Fig:-2 Home Screen

Fig:-3 Authentication and Authorization

Fig:-4 File Upload

Fig:-5 File Download

5. CONCLUTION

We proposed three authenticated key exchange protocols for parallel network file system (pNFS). Our protocols offer three appealing advantages over the existing Kerberos-based pNFS protocol. First, the metadata server executing our protocols has much lower workload than that of the Kerberos-based approach. Second, two our protocols provide forward secrecy: one is partially forward secure (with respect to multiple sessions within a time period), while the other is fully forward secure (with respect to a session). Third, we have designed a protocol which not only provides forward secrecy, but is also escrow-free.

6. FUTURE SCOPE

In Future we enhance pNFSAKE- II with a key update technique based on any efficient one-way function, such as a keyed hash function. In Phase I, we require C and each Si to share some initial key material in the form of a Diffie-Hellman key. First, the metadata server executing our protocols has much lower workload than that of the Kerberos-based approach. Second, two our protocols provide forward secrecy: one is partially forward secure (with respect to multiple sessions within a time period), while the other is fully forward secure (with respect to a session).Third, we have designed a protocol which not only provides forward secrecy, but is also escrow-free.

7. REFERENCES

[1] S. Kamara, and K. Lauter,
"Cryptographic cloud storage,"

[2] S. Grzonkowski, and P. M. Corcoran, "Sharing cloud services: user authentication for social enhancement of home networking,"

[3] P.A. Cabarcos, F.A. Mendoza, R.S. Guerrero, A.M. Lopez, and D. Diaz-Sanchez, "SuSSo: seamless and ubiquitous single sign-on for cloud service continuity across devices,"

[4] D. Diaz-Sanchez, F. Almenarez, A. Marin, D. Proserpio, and P.A. Cabarcos, "Media cloud: an open cloud computing middleware for content management,"

[5] J. Li, Q. Wang, C. Wang, N. Cao, K. Ren, and W. Lou, "Fuzzy keyword search over encrypted data in cloud computing,"

[6] C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, "Secure ranked keyword search over encrypted cloud data,"

[7] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data,"

[8] Q. Chai, and G. Gong, "Verifiable symmetric searchable encryption for semi-honest-but-curious cloud servers,"

Web Sites Referred

www.cloudxl.com

www.cloud-computing.com

www.talkincloud.com

www.cloudcomputing.sys-con.com

www.virtualizationreview.com/Home.aspx

www.thecloudtutorial.com