Data Exchange in Network File System Using Authenticated Key Exchange Protocol

M.NAGA MALLIKAARJUN REDDY[1]& S.SRAVANI[2]
[1]ASSISTANT PROFESSOR Dept. of CSE SVR ENGINEERING COLLEGE, AYYALURU, NANDYAL Email:- malli51arjun@gmail.com
[2]PG-Scholar Dept. of CSE SVR ENGINEERING COLLEGE, AYYALURU, NANDYAL
Email:- sravanisharon@gmail.com

Abstract

Already we studied the issues of key establishment for secure many-to-many communications. The main problem is inspiredby the proliferation of large-scale distributed file systems supporting parallel access to multiple storage devices. The system workfocuses on the current Internet standard for such file systems, i.e., parallel Network File System (pNFS), which makes use of Kerberosto establish parallel session keys between clients and storage devices. Our review of the existing Kerberos-based protocol shows that ithas a number of limitations: (i) a metadata server facilitating key exchange between the clients and the storage devices has heavyworkload that restricts the scalability of the protocol; (ii) the protocol does not provide forward secrecy; (iii) the metadata servergenerates itself all the session keys that are used between the clients and storage devices, and this inherently leads to key escrow. . Inthis paper, we propose a variety of authenticated key exchange protocols that are designed to address the above issues. We show thatour protocols are capable of reducing up to approximately 90% of the workload of the metadata server and concurrently supportingforward secrecy and escrow-freeness. All this requires only a small fraction of increased computation overhead at the client.

Keywords: Parallel sessions; authenticated key exchange; network file systems; forward secrecy; key escrow.

1. Introduction

In parallel file system, the file data is spread across the multiple storage devices or nodes to allow the concurrent accessby many different tasks of a parallel application. [7]This is frequently used in large-scale cluster computing that focuseson high performance and reliable access to large datasets. That is, higher I/O bandwidth is achieved through concurrentaccess to multiple storage devices within large compute clusters; while data loss is protected through data mirroring usingfault-tolerant striping algorithms. [1]In this work, we examine the problems of secure many-to-many communications inlarge-scale network file systems which support the parallel access to multiple storage devices.That is, we examine a communication model where there are a huge number of clients (potentially hundreds orthousands) accessing multiple remote and distributed storage devices (which also may scale up to hundreds or thousands)in parallel. Especially, we focus on how to exchange key materials and build parallel secure sessions between the clientsand the storage devices in the parallel Network File System (pNFS).[6] The development of pNFS is driven by Netapp,Panasas, Sun, IBM, EMC and UMich/CITI, and thus it shares many common features and is compatible with manyexisting commercial/proprietary network file systems.The primary goal here is to design an efficient and secure authenticated key exchange protocol that meets thespecific requirements of pNFS. The main aim is to achieve the properties like scalability, forward secrecy, Escrow-free.The main aim of this paper is to propose a variety of authenticated key exchange protocol which is very efficient tohandle the reducing up to approximately of the workload of the metadata server and concurrently supporting forwardsecrecy and escrow-freeness.[3] All this requires only a small fraction of increased computation overhead at the client.We define an appropriate security model and prove that our protocols are secure in the model.2. 2. **Related Work**

2.1 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.

2.2 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

3.1 Coordinatemeet

Parallel secure sessions between the clients and the storage devices in the parallel Network File System (pNFS) the current Internet standardin 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 securing with respect to multiple sessions within a time period.

3.2 Validate key Swap:

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

3.3 Onward Seclusion:

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.

3.4 Escrow-free:

The metadata server should not learn any information about any session key used by the client and the storage device, provided there is no collusion among them.

3.5 Scalability:

The metadata server facilitating access requests from a client to multiple storage devices should bear as little workload as possible such that the server will not become a performance bottleneck, but is capable of supporting a very large number of clients.

4. Experimental Work

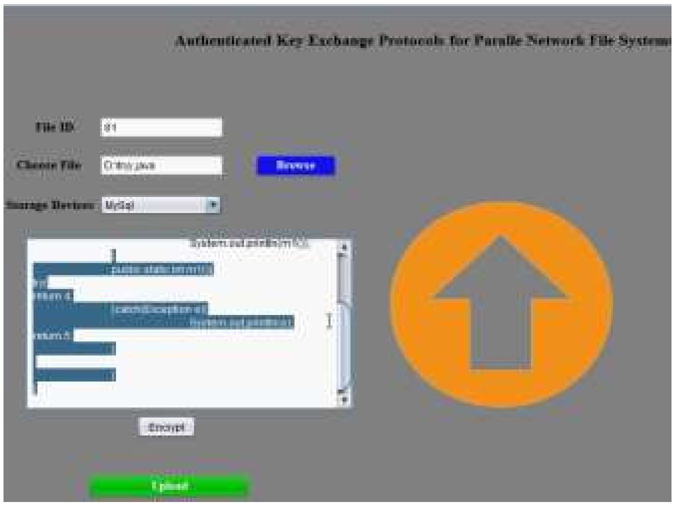
Authenticated Key Exchange Protocols for Paralle Network File Systems
File ID
01
Choose File
Browse
Storage Devices
MySql
Encrypt
Upload

Fig 1: Secure File Uploading Process.

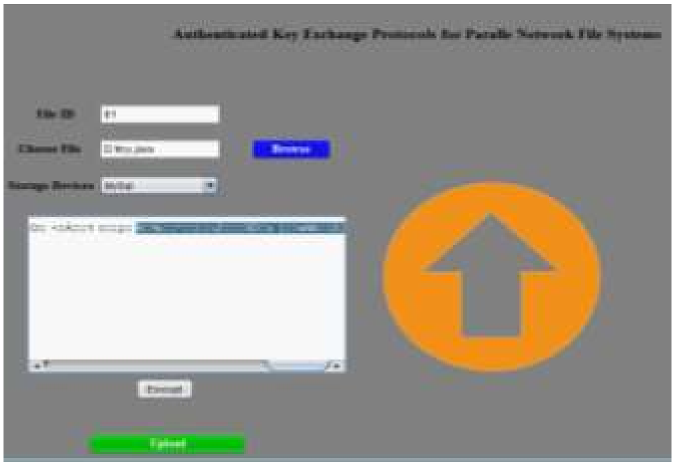
Authenticated Key Exchange Protocols for Paralle Network File Systems
File ID
01
Choose File
Browse
Storage Devices
MySql
Encrypt
Upload

Fig 2: File with Secure Key encrypted format.

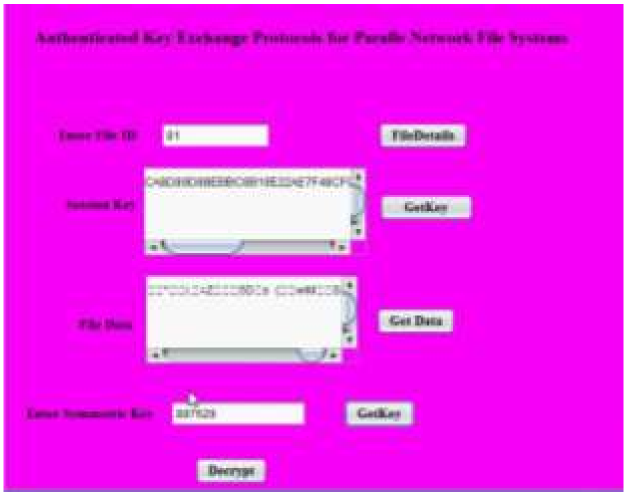

Fig 3: File decrypt from storage folder

5. Conclusion

We proposed the three authenticated key exchange protocols for the parallel network file system (pNFS). The threeappealing advantages are offered by our protocols over the existing Kerberos-based pNFS protocol. Firstly the metadataserver which is executing our protocols has much lower workload as compared to that of the Kerberos-based approach.Secondly, two of our protocols provide the forward secrecy: one which is partially forward secure, while other is the fullyforward secure. Thirdly we also have designed a protocol which provides forward secrecy as well as is escrow-free.

6. References

[1] M. Abd-El-Malek, W.V. Courtright II, C. Cranor, G.R.Ganger, J. Hendricks, A.J. Klosterman, M.P. Mesnier,M. Prasad, B. Salmon, R.R. Sambasivan, S.Sinnamohideen,

J.D. Strunk, E. Thereska, M. Wachs,and J.J. Wylie. Ursa Minor: Versatile cluster-basedstorage. In Proceedings of the 4th USENIX Conferenceon File and Storage Technologies (FAST), pages 59–72.USENIX Association, Dec 2005.

[2] C. Adams. The simple public-key GSS-API mechanism(SPKM). The Internet Engineering Task Force (IETF),RFC 2025, Oct 1996.

[3] A. Adya, W.J. Bolosky, M. Castro, G. Cermak, R.Chaiken, J.R. Douceur, J. Howell, J.R. Lorch, M.

Theimer, and R. Wattenhofer. FARSITE: Federated,available, and reliable storage for an incompletely\trusted environment. In Proceedings of the 5thSymposium on Operating System Design andImplementation (OSDI). USENIX Association, Dec2002.

[4] M.K. Aguilera, M. Ji, M. Lillibridge, J. MacCormick, E.Oertli, D.G. Andersen, M. Burrows, T. Mann, and C.A.Thekkath. Blocklevel security for network-attacheddisks.In Proceedings of the 2nd International Conferenceon File and Storage Technologies (FAST). USENIXAssociation, Mar 2003.

[5] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H.Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin,I. Stoica, and M. Zaharia. A view of cloud computing.Communications of the ACM, 53(4):50–58. ACM Press,Apr 2010.

[6] Amazon simple storage service (Amazon S3).http://aws.amazon.com/ s3/.

[7] M. Bellare, D. Pointcheval, and P. Rogaway.Authenticated key exchange secure against dictionaryattacks. In Advances in Cryptology – Proceedings ofEUROCRYPT, pages 139–155. Springer LNCS 1807,May 2000.

[8] D. Boneh, C. Gentry, and B. Waters. Collusion resistantbroadcast encryption with short cipher texts and privatekeys. In Advances in Cryptology – Proceedings ofCRYPTO, pages 258–275. Springer LNCS 3621, Aug2005.

[9] B. Callaghan, B. Pawlowski, and P. Staubach.NFSversion 3 protocol specification. The InternetEngineering Task Force (IETF), RFC 1813, Jun 1995.