INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

SPECIAL ISSUE FOR
INTERNATIONAL CONFERENCE ON
“INNOVATIONS IN SCIENCE & TECHNOLOGY: OPPORTUNITIES & CHALLENGES"

VALIDATION KEY SWAPPING PROTOCOLS FOR PARALLEL NETWORK FILE TECHNIQUE

SATISH RASAWAR, PROF. V. SRINATH

Assistant Professor and Research Scholar, Department of computer science. govterment college of enginnering chandrapur , India,

Accepted Date: 07/09/2016; Published Date: 24/09/2016

Abstract: *We go with the problem in regarding secure many-to-many communication of key establishment. The consideration of problem when 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 Technique (pNFT), which makes use to establish parallel session keys between clients and storage devices. Our review of the existing based protocol shows that it has a number of limitations [1]. A metadata server facilitating key exchange between the clients and the storage devices has heavy workload that restricts the scalability of the protocol [2]. The protocol does not provide forward secrecy [3].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. In this paper, we propose a variety of validation key swapping protocols that are designed to address the above issues[6]. We show that our protocols are capable of reducing 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[10-11].*

Keywords: Parallel sessions, validation key swapping, network file technique, forward secrecy, key escrow.

Corresponding Author: MR. SATISH RASAWAR

Co Author: PROF. V. SRINATH

Access Online On:

www.ijpret.com

How to Cite This Article:

Satish Rasawar, IJPRET, 2016; Volume 5 (2): 765-771

PAPER-QR CODE

INTRODUCTION

In a parallel file technique, file data is distributed across multiple storage devices or nodes to allow concurrent access by multiple tasks of a swapping application. This is typically used in large-scale cluster computing that focuses on high performance and reliable access to large datasets [6-7]. That is, higher I/O bandwidth is achieved through concurrent access to multiple storage devices within large compute clusters; while data loss is preserve through data mirroring using fault-tolerant striping algorithms. Some examples of high performance parallel file technique that are in production use are the IBM General Parallel File System (GPFS), Google File System, Lustre, Parallel Virtual File System (PVFS) , and Panamas File System; while there also exist research projects on distributed object storage systems such as Usra Minor , Ceph , Extremes ,and Farm[8]. 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 technique supporting pet byte- or terabyte-scale storage capacities.

In this work, we explore the problem of secure many-to-many communications in large-scale network file systems that assist parallel access to multiple storage devices. That is, we consider a communication version where there are a large number of consumer (potentially hundreds or thousands) accessing multiple remote and spread storage devices (which also may scale up to hundreds or thousands) in parallel [12-13]. Particularly, we focus on how to swap key materials and establish parallel secure sessions between the clients and the storage devices in the parallel Network File Technique (pNFT) —the current Internet standard—in an efficient and scalable manner. The development of pNFT is driven by Panamas, Netapp, Sun, EMC, IBM, and UMich/CITI, and thus it shares many common features and is compatible with many existing commercial/proprietary network file systems [15]. Our primary goal in this work is to design efficient and secure validation key swapping protocols that meet specific requirements of pNFT. Particularly, we attempt to meet the following desirable properties, which either have not been satisfactorily achieved or are not achievable by the current Kerberos-based solution[4].

Related Work:

Study the problem of key building for secure many-to-many communications. The problem is inspired by the proliferation of large-scale spread file systems supporting parallel access to multiple storage devices [11]. Our work focuses on the current Internet standard for such file systems, i.e., parallel Network File Technique (pNFT), 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 [1-2]. A metadata server facilitating key exchange between the clients and the storage devices has heavy workload that restricts the scalability of the protocol [3-4]. 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 conduct to key escrow [7].

Proposed Work:

We propose a variety of validations key swapping protocols that are designed to address the above issues[10]. We show that our protocols are capable exactly reducing of the workload of the metadata server and concurrently assist forward secrecy and escrow-freeness. All this requires only a small fraction of increased computation overhead at the client.

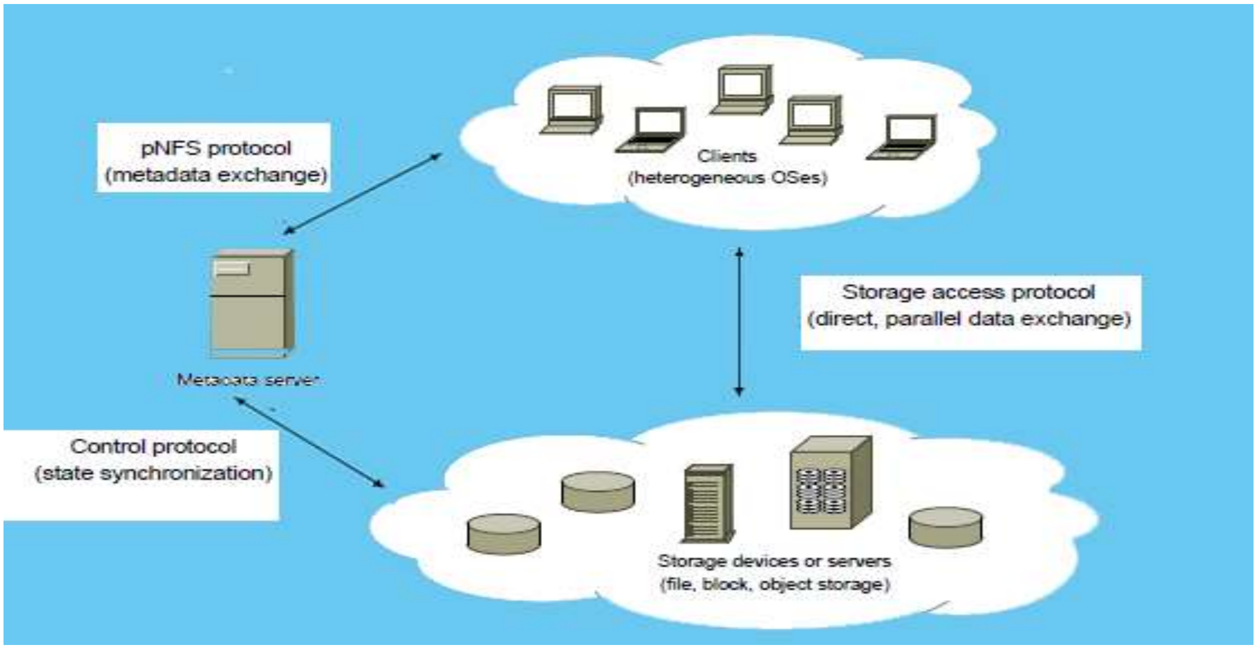


Figure 1: Key Swapping Protocols for Parallel Network File Technique

Validation Key Swapping:

Our primary goal in this work is to design efficient and secure validation key swapping protocols that meet specific requirements of pNFT. The main results of this paper are three new provably secure validation key swapping protocols. We describe our design goals and give some intuition of a variety of pNFT validation key swapping (pNFT-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 Daffier- 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 [6].

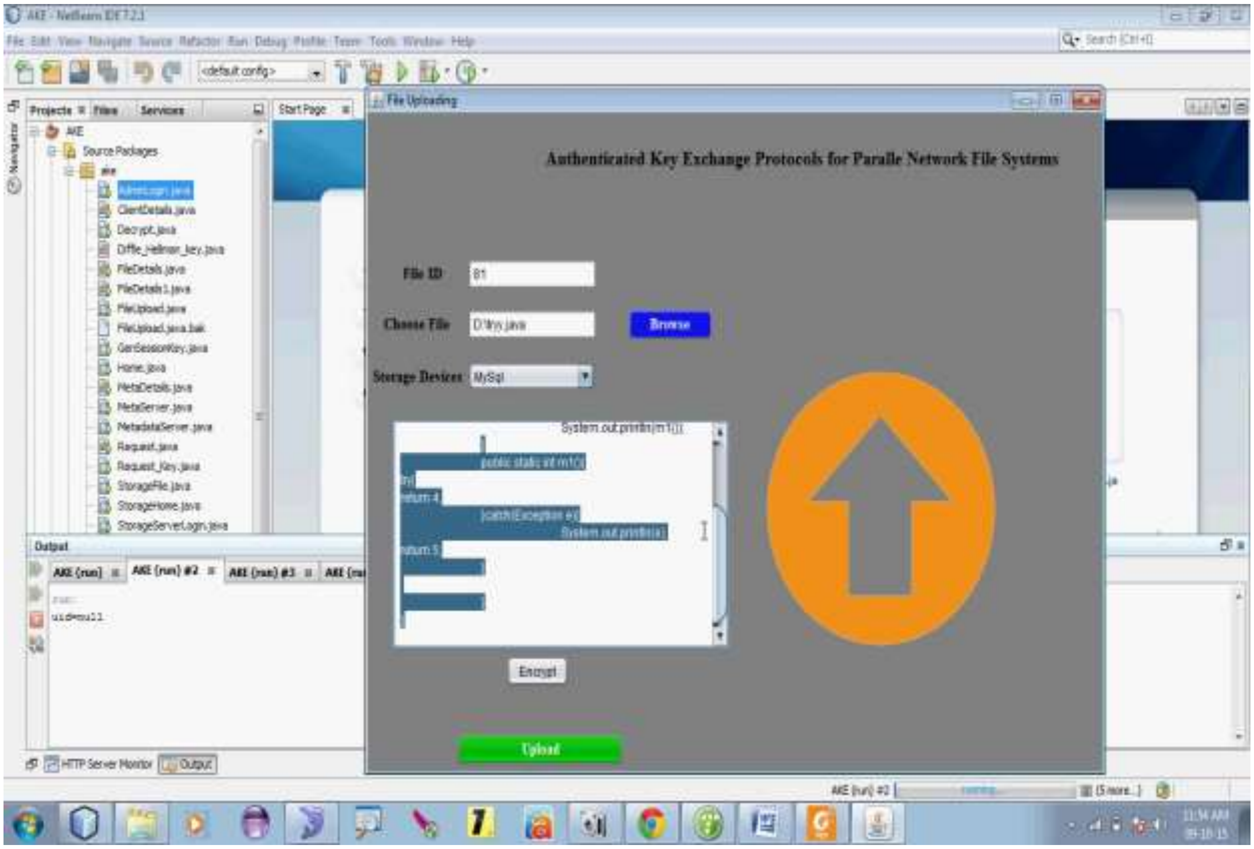
Escrow-free*:*

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

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 [9-11].

Results:



File Uploading

Authenticated Key Exchange Protocols for Paralle Network File Systems

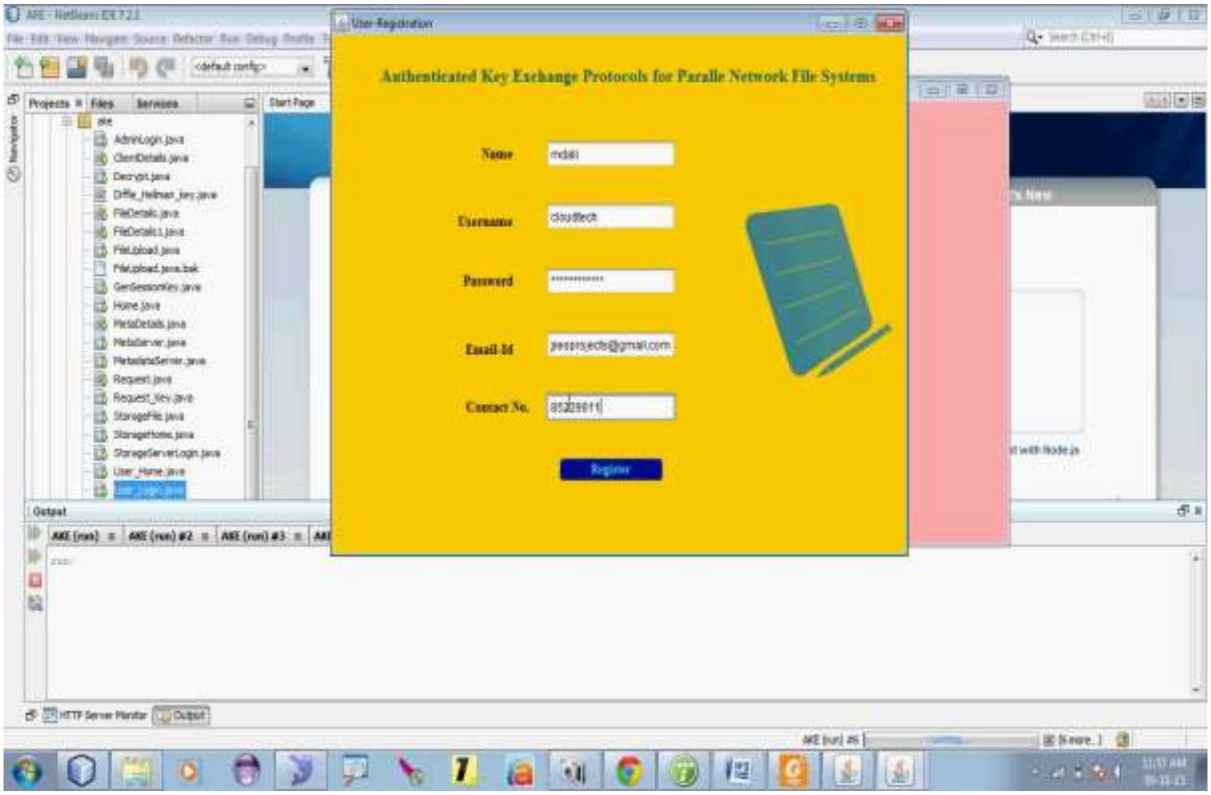
File ID: 81

Choose File: D:\hy.java

Storage Devices: MySql

```
System.out.println(m1());
}
public static int m1()
try{
return 4;
}catch(Exception e){
System.out.println(e);
return 5;
}
}
}
```

Figure 2: File Uploading in Mysql server



User Registration

Authenticated Key Exchange Protocols for Paralle Network File Systems

Name: mdali

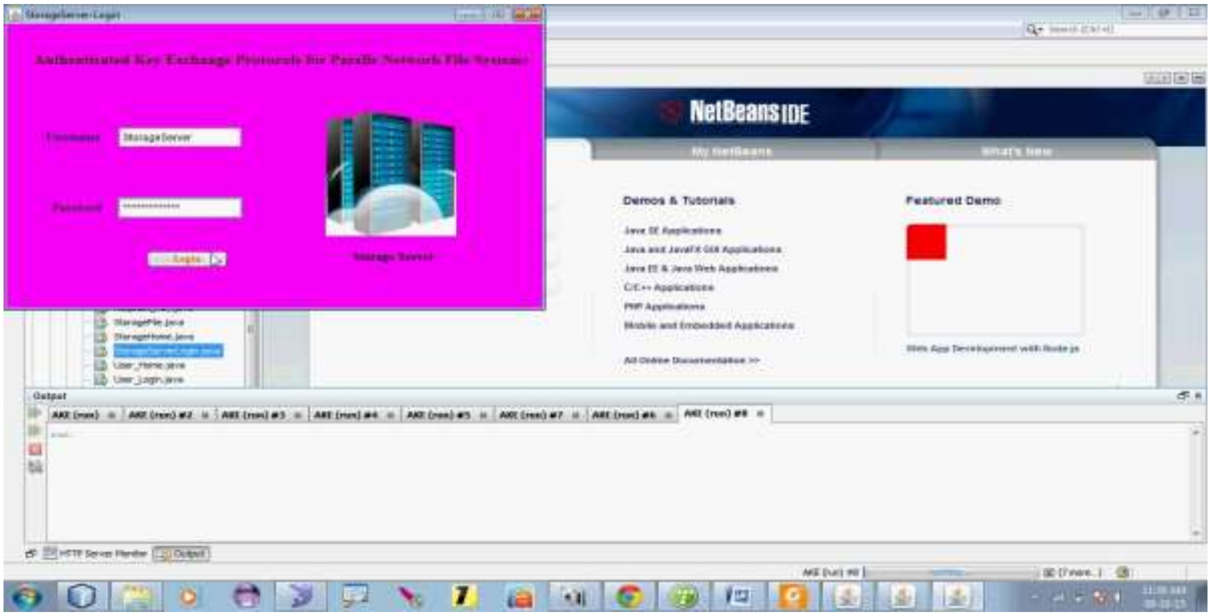
Username: cloudtech

Password: ************

Email-Id: jpesprojects@gmail.com

Contact No. 85229011

Figure 3: User Registration Details



StorageServer-Login

Authenticated Key Exchange Protocols for Paralle Network File Systems

Username: StorageServer

Password: ************

Figure 4: Storage Server Login

CONCLUSION:

We proposed three validated key swapping protocols for parallel network file technique (pNFT). 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 securing (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.

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-based storage. In Proceedings of the 4th USENIX Conference on 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 5th Symposium on Operating System Design and Implementation (OSDI). USENIX Association, Dec 2002.

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-attached disks. In Proceedings of the 2nd International Conference on File and Storage Technologies (FAST). USENIX Association, 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 dictionary attacks. In Advances in Cryptology – Proceedings of EUROCRYPT, pages 139–155. Springer LNCS 1807, May 2000.

8. D. Boneh, C. Gentry, and B. Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In Advances in Cryptology – Proceedings of CRYPTO, pages 258–275. Springer LNCS 3621, Aug 2005.

9. B. Callaghan, B. Pawlowski, and P. Staubach. NFS version 3 protocol specification. The Internet Engineering Task Force (IETF), RFC 1813, Jun 1995.

10. R. Canetti and H. Krawczyk. Analysis of key-exchange protocols and their use for building secure channels. In Advances in Cryptology – Proceedings of EUROCRYPT, pages 453–474. Springer LNCS 2045, May 2001.

11. CloudStore. http://gcloud.civilservice.gov.uk/cloudstore/. [12] Crypto++ 5.6.0 Benchmarks. http://www.cryptopp.com/benchmarks. html.

12. J. Dean and S. Ghemawat. MapReduce: Simplified data processing on large clusters. In Proceedings of the 6th Symposium on Operating System Design and Implementation (OSDI), pages 137–150. USENIX Association, Dec 2004.

13. M. Eisler. LIPKEY - A Low Infrastructure Public Key mechanism using SPKM. The Internet Engineering Task Force (IETF), RFC 2847, Jun 2000.

14. M. Eisler. XDR: External data representation standard. The Internet Engineering Task Force (IETF), STD 67, RFC 4506, May 2006.