Implementation of highly efficient Authentication and Transaction Security

Garima Verma
Asstt. Professor, MCA Department
Dehradun Institute of Technology
Dehradun, India

Prof R.P. Arora
Head, Department of CSE
Dehradun Institute of Technology
Dehradun, India

ABSTRACT
Kerberos is a network authentication protocol & is designed to provide strong authentication for client/server applications by using secret-key cryptography. Our research was aimed at enhancing the security of transactions over a network. In this paper, we used Kerberos Encryption Technique for authentication and transaction security in the network. Further, we created an Authentication Server that used to derive a 64 bit key from user's password. This password was of arbitrary length. The generated key then was used by authentication server, to encrypt ticket granting ticket + session key. The key generated by authentication server was then used by the client at the time of transaction through the transaction server to validate an authentic transaction. However, there was an issue of cross-validation of the ticket by the transaction server for which we included a database and encryption of all the text sent by any client to the transaction server.

Keywords
Secret key, cryptography, authentication, ticket, session key etc.

1. INTRODUCTION
With the advent of computer the need for automated tools for protecting files and other information stored on the computer became evident [15]. One thing to keep in mind is that network security costs money: It costs money to hire, train, and retain personnel; to buy hardware and software to secure an organization's networks; and to pay for the increased overhead and degraded network and system performance that result from firewalls, filters, and intrusion detection systems (IDSs). As a result, network security is not cheap.

The three legs of the "security trinity," prevention, detection, and response, comprise the basis for network security. The security trinity should be the foundation for all security policies and measures that an organization develops and deploys.

Figure 1- The security trinity

Prevention

The foundation of the security trinity is prevention. To provide some level of security, it is necessary to implement measures to prevent the exploitation of vulnerabilities.

Detection

Once preventative measures are implemented, procedures need to be put in place to detect potential problems or security breaches; in the event preventative measures fail. It is very important that problems be detected immediately. The sooner a problem is detected the easier it is to correct and cleanup.

Response

Organizations need to develop a plan that identifies the appropriate response to a security breach. The plan should be in writing and should identify who is responsible for what actions and the varying responses and levels of escalation

1.1 KERBEROS

Kerberos is a network authentication protocol. It is designed to provide strong authentication for client/server applications by using secret-key cryptography.

The Internet is an insecure place. Many of the protocols used in the Internet do not provide any security. Tools to "sniff" passwords off of the network are in common use by malicious hackers. Thus, applications which send an unencrypted password over the network are extremely vulnerable. Worse yet, other client/server applications rely on the client program to be "honest" about the identity of the user who is using it. Other applications rely on the client to restrict its activities to those which it is allowed to do, with no other enforcement by the server.

1.1.1 Basic Concepts

The Kerberos protocol relies heavily on an authentication technique involving shared secrets. The basic concept is quite simple: If a secret is known by only two people, then either person can verify the identity of the other by confirming that the other person knows the secret.

For example, let's suppose that Alice often sends messages to Bob and that Bob needs to be sure that a message from Alice really has come from Alice before he acts on its information. They decide to solve their problem by selecting a password, and

they agree not to share this secret with anyone else. If Alice's messages can somehow demonstrate that the sender knows the password, Bob will know that the sender is Alice.

The only question left for Alice and Bob to resolve is *how* Alice will show that she knows the password. She could simply include it somewhere in her messages, perhaps in a signature block at the end—*Alice, Our$ecret*. This would be simple and efficient and might even work if Alice and Bob can be sure that no one else is reading their mail. Unfortunately, that is not the case. Their messages pass over a network used by people like Carol, who has a network analyzer and a hobby of scanning traffic in hope that one day she might spot a password. So it is out of the question for Alice to prove that she knows the secret simply by saying it. To keep the password secret, she must show that she knows it without revealing it.

The Kerberos protocol solves this problem with secret key cryptography. Rather than sharing a password, communication partners share a cryptographic key, and they use knowledge of this key to verify one another's identity. For the technique to work, the shared key must be *symmetric*—a single key must be capable of both encryption and decryption. One party proves knowledge of the key by encrypting a piece of information, the other by decrypting it. Figure 2.

2. LITERATURE REVIEW

Dr. S. Santhosh Baboo, K. Gokulraj (2010), Authentication is one of the essential security features in network communication. Authentication process ascertains the legitimacy of the communicating partners in communication. The authors introduced a new authentication scheme based on dynamicity which is relatively a different approach to ensure and enhance the smart card based remote authentication and security. This method discusses about the authentication for smart card based network systems. This method introduces a dynamic authentication scheme which includes number of factors, among them the password, password index, and date of modification are important factors which decides the dynamicity.

K. Aruna et. al (2010), The aim of this paper is to establish a collaborative trust enhanced security model for distributed system in which a node either local or remote is trustworthy. They have also proposed a solution with trust policies as authorization semantics. Kerberos, a network authentication protocol is also used to ensure the security aspect when a client requests for certain services. In the proposed solution, they have also considered the issue of performance bottlenecks.

Steve Mallard(2010), He has defined various authentication method In order to protect the assets on your network. Like username and password, Biometric systems, Kerberos etc.

Dr.Mohammad N. Abdullah & May T. Abdul-Hadi (2009) they try to establish a secure communication between the clients and mobile-bank application server in which they can use their mobile phone to securely access their bank accounts, make and receive payments, and check their balances.

Hongjun liu et. al(2008), This paper has discussed potential server bottleneck problem when the Kerberos model is applied in large-scale networks because the model uses centralized management. They have proposed an authentication model based on Kerberos. Which tries to overcomes the potential server bottleneck problem and can balance the load automatically

2.1 Proposed Research Work

In this research work, we propose to extend our previous research work [1] to fill the gap left in that research work. Our previous work was based on Kerberos which is a network authentication protocol & is designed to provide strong authentication for client/server applications by using secret-key cryptography. We used Kerberos Encryption Technique for authentication and transaction security in Local Area Network. Further, we created an Authentication Server that used to derive a 64 bit key from user's password. This password was of arbitrary length. The generated key then was used by authentication server, to encrypt ticket granting ticket + session key. The key generated by authentication server was then used by the client at the time of transaction through the transaction server to validate an authentic transaction.

With this research, we were able to enhance the security of transactions marginally. However, there was an issue of cross-validation of the ticket by the transaction server which was not addressed in our previous work. Therefore, we carried out the same research further by providing solution for the problem by including database and encryption of all the text sent by any client to the transaction server.

The scope of present research would be increased on following lines:

- Take separate databases for Data server, client and authentication server so that databases of exist in their respective sites only.
- Encrypt the whole message + ticket instead of ticket only so that it is sent to the server using either public key encryption or private key encryption.

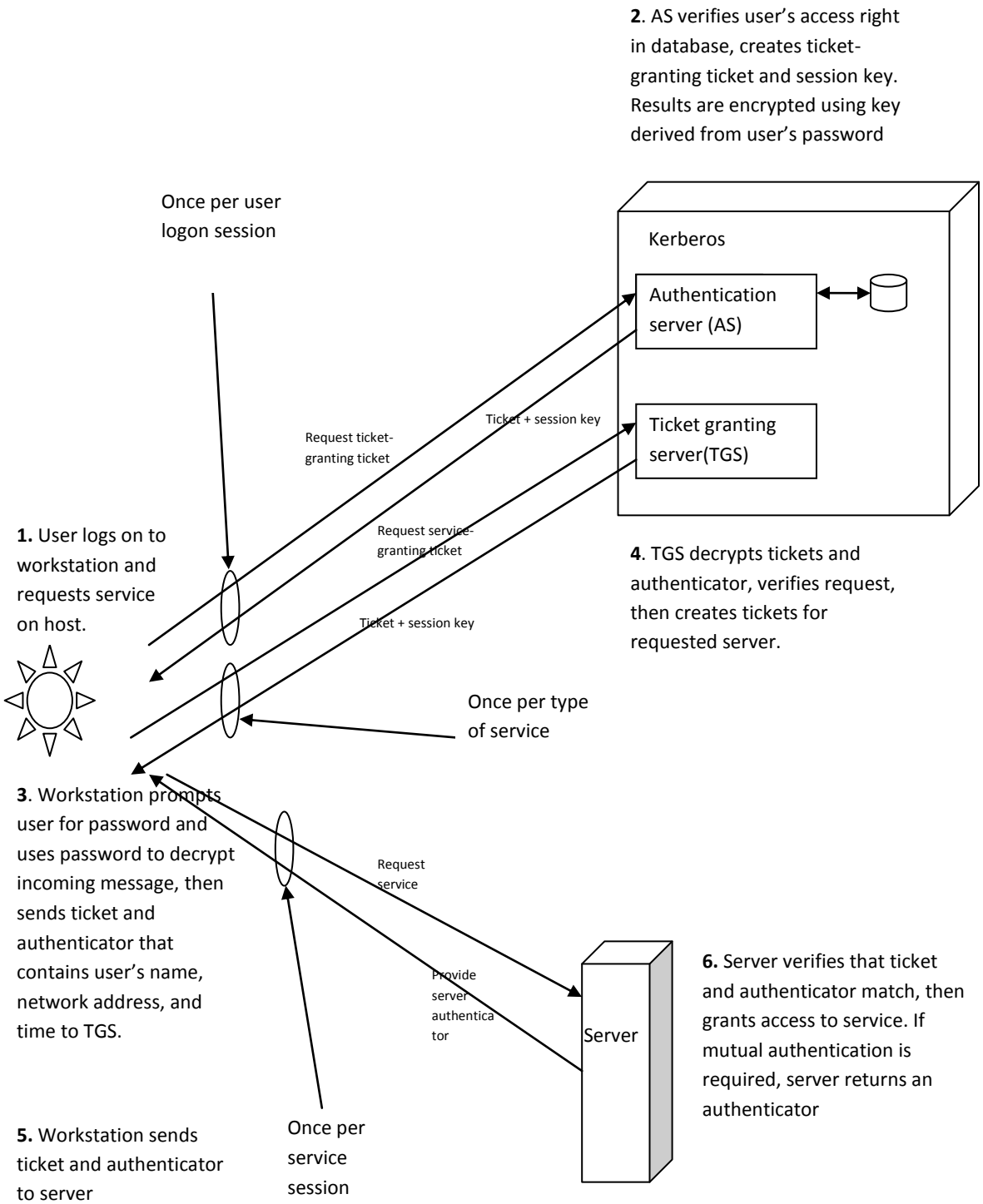


Figure 2- Model of the system

3. RESEARCH METHODOLOGY

To implement this project we have used Java and NetBeans 5.5 because we found Java as most suitable language to do the network programming. We have used concept of socket programming to implement client, authentication server and transaction server with the use of JDBC. In which the interface will be designed for client side only. To maintain the databases at all sites, I shall use mysql 5.2.27.

4. IMPLEMENTATION

In the previous work we have already created three modules- Client site, authentication server and data server. Here is a brief description about that.

- In the Client module we have given the facility of logon to its own terminal by using user name and password. These user and passwords and predefined and assigned to every client on the network. Every client have unique user name with two passwords, one password is used to logon to the client terminal and another is called as transaction password which he will submit to the authentication server. After the successful login client will submit its details with transaction password to the authentications server. Details include – username, transaction password and name of the data server. Again entered transaction password will checked into the client database then finally sent to the authentication server. Figure 3

- Authentication server is central authority that knows the passwords of all clients and stores these in a centralized database. In addition, the AS shares a unique secret key with each server. These keys have been distributed physically or in some other secure manner. For example – the user logs on to a workstation and requests access to server V: the client module C in the user's workstation requests the user's password and then sends a message to AS that includes the user's ID, server's ID and user's password. The AS checks its database to see if the user has supplied the proper password for this user ID and whether this user is permitted access to server V. If both tests are passed, the AS accepts the user as authentic and creates a Ticket. This ticket is then send back to Client. For the encryption we have used DES algorithm. After receiving ticket from the authentication server the client will send message + ticket to the data server.

- In Data server module after receiving ticket from authentication server the client can now apply to Server for service. Client sends a message to server containing its ID and ticket. Server decrypts the ticket and match it with ticket stored in the database. If these two match, the server considers the user authenticated and requested service. Figure 4.

In the present research work we have extended the previous work by introducing following things –

1. Taken separate databases

In the previous work we had taken the centralized database approach for maintaining the details of client on to the authentication server. However, we found that this approach was not suitable when we work in distributed environment, especially when there was more than one client that wanted to access more than one data server. Therefore we changed the centralized database implementation into distributed. We created database for Client, Authentication server and Data server separately. Now, whenever any client will take the ticket from the authentication server it will send the details of client including ticket to the data server. The data server will store the information into the database and will keep the copy with it also. After receiving ticket from the authentication server, the client will send the message with ticket to the server. First, the server will validate the details of client with database & if details are correct as sent by the authentication server then client will be authenticated by the Data server and communication will be proceed.

2. Encrypted the whole message + ticket instead of ticket only

Whenever client wants to send any message to the data server it will send ticket with that. In the previous work we had sent message as plain text. Now to provide more security and authentication we have done encryption of the message with ticket using RSA algorithm. It is a Public key cryptography algorithm in which every user has two keys, one public key and one private key. The client will encrypt the text with the help of public key of the Server and the Server will decrypt the message by using its private key.

5. CONCLUSION

Security is always an important issue whenever a transaction is carried in network. We usually need to share the data with authorised client only. For ensuring authentication of the clients there are several protocols available and Kerberos is one of them.

In this research paper we have made an attempt to enhance the security issue by using the concept of Kerberos. We have implemented the highly authenticated transaction scheme in which Authentication server creates a ticket which is further encrypted using the secret key shared by the server and authentication Server. This ticket is then sent back to client. Since the ticket is encrypted, it cannot be altered by client or by an opponent. Further we have encrypted the whole message of client with ticket before sending it to the server. For this we have used RSA algorithm. Also, we have implemented database on the authentication server and data server so that whenever client sends message with ticket for communication to the data server, it can be validated by the data server from its database.

6. REFERENCES

[1] Prof R.P. Arora, Garima Verma, "Implementation of Authentication and Transaction Security based on Kerberos", IJITCE, Feb 2011

[2] Dr. S. Santhosh Baboo, K. Gokulraj, "A Secure Dynamic Authentication Scheme for Smart Card based Networks", International Journal of Computer Applications, Number 8-Article 2, pp. 1605-2157, 2010

[3] K. Aruna et. al (2010), "A new collaborative trust enhanced security model for distributed systems". International Journal of Computer Application, No-26

[4] Steve Mallard(2010), "Methods of authentication", Bright Hub

[5] Hongjun liu et. al(2008), "A distributed expansible authentication model based on Kerberos" Journal of Network and Computer Application, Vol.31, Issue 4

[6] Dr.Mohammad N. Abdullah & May T. Abdul-Hadi, "A Secure Mobile Banking Using Kerberos Protocol", Engg & Technology Journal, Vol 27, No 6, 2009.

[7] "How Kerberos Authentication Works", Network on line magazine, Jan 2008

[8] "How Kerberos Authentication Works",Learn Networking on line magazine, Jan'2008

[9] Frederick Butler, Iliano Cervesato, Aaron D. Jaggard, Andre Scedrov and Christopher Walstad, "Formal Analysis of Kerberos 5", Sep 2006

[10] Rong Chen, Yadong Gui and Ji Gao, "Modification on Kerberos Authentication Protocol in Grid Computing Environment", vol 3032, 2004.

[11] I. Cervesato,A. D. Jaggard,A. Scedrov,C. Walstad, "Specifying Kerberos 5 cross-realm authentication**",**vol 3032, 2004.

[12] "Security of Network Identity: Kerberos or PKI", System News (2002), Vol.56, Issue-II

[13] Ian Downnard, "Public-key cryptography extensions into Kerberos". IEEE Potentials 2002.

[14] B. Clifford Neuman and Theodore Ts'o, Kerberos: An Authentication Service for Computer Networks, IEEE Communications 32 (1994), no. 9, 33--38.

[15] MIT Kerberos Website, "http://web.mit.edu/kerberos/www".

[16] William Stallings, "Cryptography and Network Security", Third Edition.

[17] Ravi Ganesan, "Yaksha' : Augmenting Kerberos with Public Key cryptography".

[18] John E. Canavan, "Fundamentals of Network Security".

[19] Chris Brenton with Cameron hunt , " ACTIVE DEFENCE A comprehensive guide to network security"

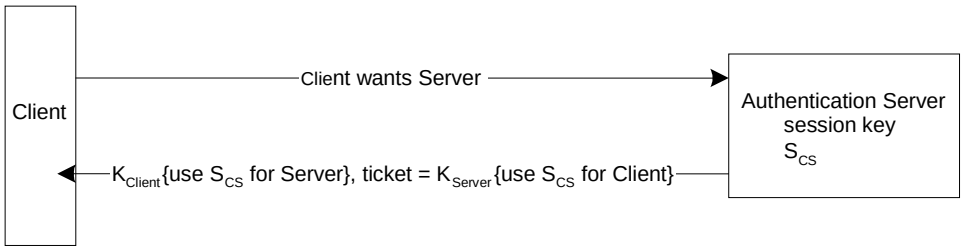


Figure -3 Generation of Ticket

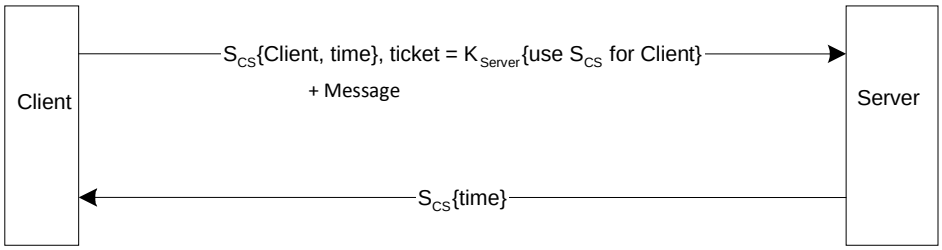
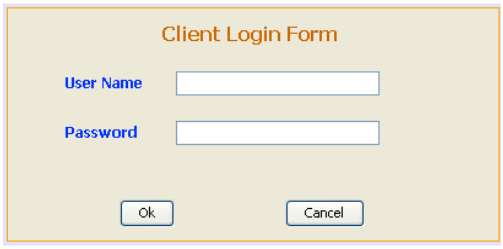


Figure -4 Client sending message and Ticket to Data server

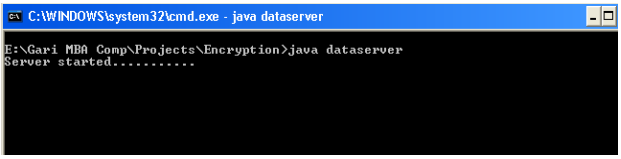
Appendix- Snapshots

Client Interface

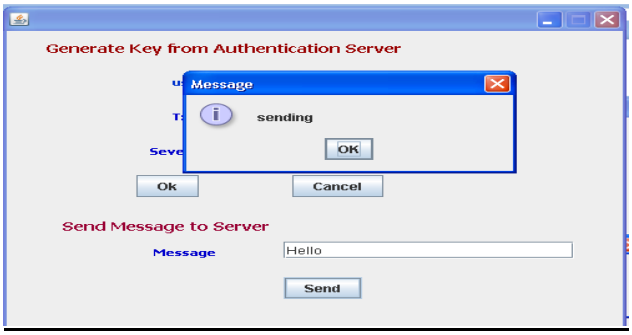


Authentication Server

Data Server



Client sending Message and ticket to the Dataserver



Client is authenticated by the Dataserever