IMPLEMENTATION ON PUBLIC KEY CRYPTOGRAPHY AND LOCATION BASED AUTHENTICATION FOR KERBEROS AUTHENTICATION PROTOCOL

Amit Thorat ¹ | Khadir Sayyad ¹ | Pranay Share ¹ | Darshan Thorat ¹
¹ Department of Computer Engineering, Z.C.O.R.E, Pune, Maharashtra.

ABSTRACT
Kerberos is one of the advanced security system that helps prevent people from stealing information that gets sent across the network from one computer to another. Usually, these people are after your password. The Kerberos security system, guards electronic transmissions that get sent across the Internet. It does this by scrambling the information -- encrypting it -- so that only the computer that is supposed to receive the information can unscramble it, or say decrypt it. In addition, it makes sure that your password itself never gets sent across the wire: only a scrambled "key" to your password. To make a certain addition to the Kerberos Protocol we add the public key Cryptography for the phases of the Kerberos Authentication System. This would help in a more secure and a password attack free system. There would also be addition of the current location of the user requesting the service to the Data provided by the user which would add to more authentication and security.

KEYWORD: Authentication, Kerberos, public-key cryptography, PKINIT, PKCROSS, PKTAPP

1. INTRODUCTION
Authentication is a fundamental building block for a secure network environment. In modern computer systems authentication provides service to multiple users and provides the ability to accurately identify the user making a request. Today more common in computer network architecture is a distributed architecture consisting of dedicated user workstations and distributed or centralised servers in this environment. Thus we need to protect user information and resources housed at the server. This environment must provide means to ensure that the workstation can identify its users properly. A communication procedure runs between two or more cooperative principles in this environment is called a protocol. Authentication protocol is a series of steps and message exchanges between multiple entities in order to achieve authentication objective. These protocols are usually a combination of cryptographic techniques and other means. Some of the authentication protocols use symmetric cryptography techniques. In the authentication protocols that use asymmetric cryptography techniques both the encryption and the decryption process need the same key, so the communication entities need to share a secret key. Authentication based on public key cryptography has an advantage over many other authentication schemes because no secret information has to be shared by the entities involved in the exchange. A user attempting to authenticate oneself must use a private key to digitally sign a random number challenge issued by the verifying entity. If the verifier can successfully verify the signed response using the user’s public key, then the user has been successfully authenticated.

II. NEED FOR KERBEROS PROTOCOL
The World Wide Web provides a way that the users access the services over the internet remotely and in such cases the data that the user provides is confidential enough to be known by any of the third parties. There has to be secure authentication mechanism to only let the authorized users access the data remotely. The simplest way how authentication works is to verify whether the user proves who he claims to be. Consider one of the following scenarios.

- A user gaining access to a particular workstation and pretending to be another user operating from that workstation.
- A user altering the network address of a workstation so that the requests sent from the altered workstation appear to come from the impersonated workstation.
- A user can eavesdrop on exchanges and use a reply attack to gain entrance to a server or to disrupt operations.

In any of these cases, an unauthorized user may be able to gain access to services and data that he or she is not authorized to access. Rather than building in elaborate authentication protocols at each server, Kerberos provides a centralized authentication server whose function is to authenticate users to servers and servers to users. Unlike most other authentication schemes, Kerberos relies exclusively on symmetric encryption, making no use of public-key encryption. But the secret key encryption provides the benefit at the cost of the passwords, hence the security ends once the password is cracked. Hence we plan the replacement of secret key with the public key cryptography in the first phase.

The entire Kerberos process has 8 steps.
1. The authentication service, or AS, receives the request by the client and verifies that the client is indeed the computer it claims to be. This is usually just a simple database lookup of the user’s ID.
2. Upon verification, a timestamp is created. This puts the current time in a user session, along with an expiration date. The default expiration date of a timestamp is 8 hours. The encryption key is then created. The timestamp ensures that when 8 hours is up, the encryption key is useless. (This is used to make sure a hacker doesn’t intercept the data, and try to crack the key. Almost all keys are able to be cracked, but it will take a lot longer than 8 hours to do so.)
3. The key is sent back to the client in the form of a ticket-granting ticket, or TGT. This is a simple ticket that is issued by the authentication service. It is used for authenticating the client for future reference.
4. The client submits the ticket-granting ticket to the ticket-granting server, or TGS, to get authenticated.
The TGT and the message encrypted under \( k \) are as intra- 

cage "C, TGT, {AK, n1, tK, T}k" is very similar to reply in basic Kerberos, the dif- 

The Public key extension proposed in PKCROSS take place only between pairs 

d of this procedure is for the Kerberos server to transmit credentials(TGT and Session Key) to the client user who requested it. In PKINIT the, 

PKINIT modifies the AS exchange but 

Not other part of basic Kerberos 

Protocol.

B. PKCROSS Prototype 

In traditional Kerberos system, the KDC issues all TGS, remote KDC and server 

tickets in its realm. Thus, most authentication transaction should transit the 

KDC. Therefore it can become a performance bottleneck. Although secondary 

KDCs can be included in the system, their typical use is just as backups in the 

event of primary KDC failure. The aim of PKTAPP specification is to eliminate 

this potential bottleneck and reduce this communication traffic by enabling the 

authentication exchange to be directly performed between the client and the 

application server.

IV. KERBEROS WITH PKINIT 

The Public key addition in the first phase of the Kerberos authentication process 

would definitely add up to the security level of the protocol. Already the some of 

the implementers of Kerberos have added this security level.

PKINIT describes how Public Key Cryptography can be added to Kerberos in the 

Initial Authentication stages. Already Microsoft, Cyber safe and Heimdal have 

adopted it in their implementations of Kerberos. PKINIT requires the verifica- 

tion of the initial request message sent by the client to the local KDC (Authenti- 

cation Server). Any authentication requests made by intruders masquerading as 

legitimate users will be denied. PKINIT also requires that the Ticket granting 

Ticket and the session key be encrypted so that the user's credentials will also 

remain confidential.

This diagram shows the basic message for first phase of the Kerberos process 

using public key. The Client's name, the name of TGS and the nonce part of the 

message is same as in Kerberos version 5. The Certificate is added by the 

PKINIT. This is the client's certificate and there is also a timestamp and nonce 

encrypted over a secret key.

The second phase shows the formalization of response. The last part of the mes- 

sage "C, TGT, {AK, n1, tK, T}k" is very similar to reply in basic Kerberos, the dif- 

ference is that the symmetric key \( k \) protecting AK is now generated by AS and not 

a long-term shared key. The TGT and the message encrypted under \( k \) are as in tra- 

ditional Kerberos. Because \( k \) is freshly generated for the reply, it must be 

informed to C before C can learn AK. PKINIT does this by adding the boxed mes- 

sage \( \{\text{CertK}, [k, n2]_{skK}\}_{pkC} \). This contains K's certificates and its signature, 

using its secret key \( skK \), over \( k \) and the nonce \( n2 \) from C's request; all of this is 

encrypted under C's public key \( pkC \).

V. LOCATION BASED 

We shall also add the location based authentication to our Kerberos authentica- 
tion system for its advantages. Consider the example of a multinational company 

that gives a separate email id and database access to the employees. Now every 

employee would get a employee email id specific to the company. This id he can 

access in the company premises as well as any other area outside the company. 

But when you take the case of the database access you certainly don't want it to 

be accessed outside the company, it should only be dealt with in the premises. So 

location based authentication would help here. Consider an employee of that 

company wanting to access the company database at his home, the location based 

authentication would deny him the permissions of any such access. It would work 

on the longitudes and latitudes and anything outside its limit would result in 

denial of that service. Now of course that location based addition should not 

affect the email access of the employee anywhere in the world. We shall also add 

the administrator rights so that the location based service can be enabled and dis- 

abled appropriately.

VI. FUTURE SCOPE AND CONCLUSION 

The Kerberos Authentication protocol has been used by many companies as a 

trusted authentication protocol. There are a lot of advantages of the protocol like 

a mutual authentication between the clients and the server. The passwords are 

never sent across the network. The addition of public key and location based 

would certainly add up the authentication and security level. Unlike many alter- 
native authentication mechanisms, Kerberos is entirely based on open Internet 

standards. A number of well-tested and widely-understood reference implemen-
tations are available free of charge to the Internet community. Commercial 

implementations based on the accepted standards are also available.

VII. REFERENCES 

1. B. Clifford Neuman and Theodore Ts'o, “Kerberos: An Authentication Service for 

Computer Networks”, IEEE communication magazine, (Sept 1994) 

2. Qin Li Fan Yang, Huibiao Zhu and Longfei Zhu, “Formal Modeling and Analyzing 

Kerberos Protocol”, (2009) 

3. Farhana S. Munnee, AnirudhJonnavitula, “Kerberos using public Key Cryptography”, 

GMU-ECE-546 Fall 2007 


1510. September 1993. 

5. L. Zhu B. Tung, “Public Key Cryptography for Initial Authentication in Kerberos 

(PKINIT),” RFC 4556. 


www.mit.edu/kerberos/