Diyala Journal
of Engineering
Sciences

ISSN 1999-8716

Vol. 01, No. 01, pp. 150-161, December 2008

Kerberos Security Evaluation

Mohammad Najm Abdullah, May T. Abdul-Hadi, and Hussain F. Mahdi

College of Engineering, University of Diyala

(Received:18/3/2008; Accepted:15/11/2008)

ABSTRACT - Authentication of a person is an important task in many areas of day-to-day life including electronic commerce, system security and access control. We present Kerberos a client\server authentication protocol which can perform a secure communication over unsecured environments (internet). For example, an e-bank application the client can log on in domain environment using password (single factor authentication) or a smart card running java card application in combine with PIN and the server is the banking hosted system at the bank. Smart card can enhance the security by storing the cryptographic key to perform dual factor authentication, it also can manage the encryption and decryption of the Kerberos keys on it rather then on the client workstation memory. A common methodology depends on the national standardizations is used to evaluate security of that authentication scenarios of Kerberos protocol.

1. INTRODUCTION

Identification of a person is a basic task in day-to-day life. We identify our friends, family members and business associates effortlessly. In a small village or in a small community, in olden days business was run on personal identification methods without computers. For example, a village banker approved a loan application based on the personal knowledge of the background of the applicant. There were no credit checks or credit rating bureaus either. With the growth in transportation and communication, now the concept of a village or community where every one knew one another is becoming extinct. Often, business needs demand that a person moves to a new location for a brief period.

To cater to such needs in a cashless society, automatic methods of identification are required moving beyond the old methods of family and personal trust. In a complex and fast-moving world, one has to prove several times a day who they claim to be.(2)

This paper presents a security evaluation of three scenarios for Kerberos authentication. The first based on password or PIN, second on dual factor authentication (smart card in combine with PIN), third the encryption/decryption mechanism on the smart card rather than on the client workstation memory. We propose methodology for security evaluation of the three scenarios. The rest of the paper distributes as fallows: a brief discusses to the authentication, Kerberos protocol, possible attacks, analysis and testing.

1.1. Security Services

A security service is a collection of mechanisms, procedures and other controls that are implemented to help to reduce the risk associated with a specific threat to a system. The most important security services are:

Confidentiality: To ensure that data, software and messages are not disclosed to unauthorized parties. The uses of encryption mechanism reduce the risk of data disclosure.

Integrity: To ensure that unauthorized parties do not modify data, software and messages. The services help to ensure that a message is not altered, deleted or added during transmission.

Authentication: To prove identity and allow access to assets. Many other security services are based on this mechanism.

Non-repudiation: Conclusively tracing an action to an individual. It helps to ensure that entities in a communication cannot deny having participated in all or part of the communication.

Access Control: To ensures that network resources are being used in an authorized manner.

Availability: To ensure that a service is available at all times.(6,11)

1.2- Smart Cards

Smart cards are a tamper-resistant and portable way to provide security solutions for tasks such as client authentication (by protecting private keys and other forms of personal information). For example, if a malicious person obtains a user's password, that person can assume the user's identity on the network simply through use of the password. In the case of smart cards, that same malicious person would have to obtain both the user's smart card and the PIN to impersonate the user. This combination is obviously more difficult to attack because an additional layer of information is needed to impersonate a user. An additional benefit is that, after a small number of unsuccessful personal identification number (PIN) inputs occur consecutively; a smart card is locked, making a dictionary attack against a smart card extremely difficult.(3.7)

2. AUTHENTICATION

Authentication is an important network security function. Message authentication allows the communicating parties to verify that the received messages are authentic. The two important aspects are to verify that the contents of the message have not been altered and that the source is authentic. Authentication is the process for verifying that an entity or object is who or what it claims to be. Authentication is a fundamental aspect of system security. It confirms the identity of any user trying to log on to a domain or access network resources.(1,7,8)

2.1. Authentication Concepts

One of the most common ways for authentication on computer-based systems is through the use of a user name and password. You can provide different characteristics to authenticate a user:(1,6,13)

- Knowledge: Something you know (e.g., a password)
- Possession: Something you have (e.g., a smart card)
- Characteristic: Something you are (biometrics e.g., a fingerprint, retina, or voice)

Individually, any one of the three concepts has problems. “Something you have” can be stolen. “Something you know” can be guessed, shared or lost to other methods. “Something you are” is the strongest, but generally the costliest and still vulnerable to attack. Based on these single-factor authentication problems, the next step is two-factor authentication. Combining two methods defines two-factor authentication. For example, “something you know” and “something you have” can be hardware token and a PIN number. This method has two advantages. First, it is resistant to all the reusable and most one-time password attacks. Second, it can be a method of non-repudiation. There are many types of strong user authentication in use today. These include smart cards, challenge-response, hardware tokens and biometric authentication; all combined with pins PINs or passwords. These solutions can give a great deal of comfort, but the costs must be considered.(8,11)

2.2. Network Authentication

Network authentication confirms the user's identification to any network service that the user is attempting to access. To provide this type of authentication, the security system includes these authentication mechanisms: Kerberos V5 , Public key certificates , Secure Sockets Layer/Transport Layer Security (SSL/TLS) Digest , NTLM (for compatibility with Windows NT® 4.0-based systems).[7]

2.3- Single Sign-On (SSO)

SSO is considered to be the ability to sign on only once to the access resources over the network and then accesses all participating services, without re-entering a user ID or password. If an SSO solution is compromised, all systems and environments are exposed. [1,7]

2.3- *Passwords*

Passwords are the most common type of computer system authentication. Most multi-user systems in the past relied on password authentication to control access to processor time and to segregate users for charge-back.

The common way to raise the level of security is to use rules that force increased complexity of the password and user name. The downside of this is that users have a tendency to write down passwords, use the same password on other systems. There are two types of passwords, Reusable password (a string of letters and numbers used many times for system access) and One-time password (a string of letters and numbers used for system access and always changing) that can be generated from a token device that either generates passwords on a given interval or generates passwords from a PIN (Note that a PIN does not have to be a series of numbers, it can also use other alphanumeric characters) or code that is provided for each sign-on attempt.

Each type of password has unique problems to address. Reusable passwords have reached the end of their life cycle for critical business uses and one-time passwords need additional controls to remain effective. Reusable passwords are vulnerable to many attacks, including keystroke monitoring, social engineering, brute force attacks and network monitoring.[1,11]

3. CRYPTOGRAPHY

Cryptography is the science of protecting data or messages. Many cryptographic algorithms mathematically combine input plaintext data and an encryption key to generate encrypted data referred to as *ciphertext*. With a good cryptographic algorithm, it is computationally infeasible to reverse the encryption process and derive the plaintext data from the ciphertext. In order to decrypt the ciphertext some additional data, a decryption key, is needed to perform the transformation.[3]

3.1- Encryption Methods

Cryptographic algorithms are used to encode and decode messages. There are two kinds of cryptographic algorithms: symmetric secret key algorithms and asymmetric secret key algorithms. In the symmetric algorithms the same secret key is used in encryption and decryption. Hence, it is required that the secret key be known to both sender and receiver. In the asymmetric method, the message is encoded using public key and decrypted using a private key. The public key is different from the private key. As the name indicates, the public key can be known to many parties. The private key is only known to the decryption module which can decrypt the message.[2]

4. KERBEROS THE PROTOCOL

Kerberos is a network authentication protocol developed by the Massachusetts Institute of Technology (MIT). Kerberos is designed to provide strong authentication for client/server applications by using secret-key cryptography. From the user point of view, it does not differ much from a normal sign-on process. Kerberos still relies on the user providing some form of credentials to verify their identity. The exchange of credentials is encrypted throughout the entire authentication process, enabling a secure authentication mechanism. The major difference is that after an identity is proven, a temporary *ticket* is issued to the client. This ticket allows the user to access other systems and applications that exist within the circle of trust, or more correctly, the *Kerberos realm*[1]. The Kerberos protocol uses a unique ticketing system (as defined in RFC 1510[5]) that provides faster authentication also provides the following security services: Mutual authentication, delegated access control, privacy and data integrity[3,4,6].

Kerberos contains of three basic entities the client and the server whom want to authenticate and the Key Distribution Center (KDC), the KDC is a single process comprised of two services: The Authentication Service (AS) and the Ticket-Granting Service (TGS). The AS issues Ticket Granting Tickets (TGTs) to authenticated principals (that is, users, machines, services) for admission to the TGS. The TGS issues tickets for admission to other services in the domain or to a TGS in another trusted domain. Any domain controller can accept authentication requests and ticket-granting requests addressed to the domain's KDC(3). The Kerberos has advantages over other authentication protocols first the password never travels throw the network even in encrypted form (but used for encryption/decryption on the client workstation memory), second Kerberos does not depend on the firewalls because it does not consider that attack could came from the intruders.

The protocol consists of the following steps, which are illustrated in the figure bellow:

1. A client sends a message to the KDC requesting the issuance of a TGT. The request includes the username in plaintext form of the client, but does not include his password.
2. The KDC issues a TGT to the client. The TGT contains a session key in encrypted form. To encrypt the session key, the KDC uses a key derived from the client's password. This means that only the client can decrypt the TGT and fetch the session key.
3. The client decrypts the TGT and extracts the session key from it. The client then authors a request for a service ticket. A service ticket is valid only for communication between two parties that is, between the client and the a server .
4. The KDC authors a service ticket for the server. This ticket contains the client's authentication data and a new cryptographic key, called a sub-session key. The KDC encrypts the service ticket with the secret key of the server (the secret key is a shared secret between the KDC and the server). This means that only the server can decrypt the service ticket.
5. The KDC authors a message and wraps the service ticket inside of it. The KDC also copies the sub-session key inside the message. Notice that the sub-session key is now contained in the message twice: once directly in the message and again inside the service ticket.
6. The KDC encrypts the complete message with the session key from steps 2 and 3. Thus, only the client can decrypt the message and extract the sub-session key as well as the service ticket. But the client cannot decrypt the service ticket only the server can.

Therefore, no one else can use the service ticket for any purpose. The KDC then sends the message to the client.

7. The client decrypts the message received from the KDC and fetches the sub-session key inside the message as well as the service ticket. It sends the service ticket to the server.
8. The server receives the service ticket and decrypts it to fetch the authentication data of the requesting client as well as the sub-session key. The server then acknowledges the client's request, and a new secure session is established between the client and the server. Both client and server now possess the same sub-session key, which they can use for secure communication with each other.

The client can repeat steps 3 through 8 for another server application. This means that our Kerberos service can be used to share authentication data and that the same client (which represents a single user) can authenticate with different applications. This effectively enables SSO.(15)

5. CLASSIFICATION OF ATTACKS

While there are many variations of specific attacks and attack techniques, their goals can be reduced to the following list:

1. Key stroke monitoring can be done by run a program to monitor keys pressed on a keyboard and storing the results in a file for later observation. A number of popular Trojan horse programs, such as Back Office and Net Bus offer this functionality. This attack is used when physical and logical access to the computer is not possible.
2. Social engineering is manipulating people for information. This includes the attacker posing as a member of a firm’s help desk, calling an executive’s assistant, and asking for their (or the executive’s) password to fix a computer problem. Also, this type includes ‘shoulder surfing’ which is just as it sounds a person will casually watch another person’s fingers as they enter their password to steal the letters and numbers.
3. Brute force attacks, sometimes called dictionary attacks, a user accesses a system in an authorized or unauthorized fashion. Once the user gains access to a command prompt, they can copy the encrypted passwords and run a ‘crack’ program to guess the passwords. The program compares the encrypted words from the dictionary to the ones copied from the system and when they match, you know the password.
4. Network monitoring (also known as “sniffing”) is the most critical concern with reusable passwords. Most networks today are Ethernet based. On Ethernet networks, all

messages sent from one machine to another are read by all systems on the network, but only processed by the intended recipient. However, the network cards of any of the computers on the network can be put into ‘promiscuous mode’ where they read and log all messages that reach the computer. Utilities to perform this include the Sniffer from Network Associated and the Network Monitor released by Microsoft. Using these tools, any user on the network can record all the traffic to automatically collect the network passwords. Once collected, they can be used for unauthorized access.

5. A man-in-the-middle attack is just as it sounds. An attacker places a computer between the user and the system using a one-time password. In some way, the user must capture the packets as they pass over the wire, resending them as their own. The user needs control over the network and a high degree of skill to perform this attack.

In response to this problem, security vendors have taken measures to compensate by using encryption or by putting logic into their products to address and defend against these types of attacks.(8,11)

6. KERBEROS’ ANALYSIS

1- Kerberos authentication protocol without a smartcard: that scenario is vulnerable to dictionary attack and other attacks mansion in section 4, a user key Ku is shared between a user and a KDC. Ku is derived from a password, a workstation reads the password from a user, converts it to Ku, and uses it to decrypt a TGT. When a user attempts to login to a workstation, the workstation sends a request to the KDC. KDC generates a TGT, encrypts it with Ku, and sends it back to the workstation. The workstation asks the user for a password, hashes it into key Ku, and uses the key to decrypt the TGT. If the TGT decrypts properly, the user is authenticated and is allowed to login. In this protocol, Ku is exposed to two parties, a user and a workstation. A key memorized by a user can be vulnerable because she can tell it to another person, or an adversary might \shoulder surf" it when she types it. A key in a workstation can be vulnerable if the workstation is not securely protected or cannot be trusted for other reasons. For example, if an adversary can scan the entire physical memory of the workstation, he can obtain the key. Along the same lines, if someone has administrative access rights to the workstation, it is straightforward to install a rogue login program in the workstation that stores a user's password in the adversary's directory. (This is called a Trojan horse attack.)To solve these problems, it is desirable to decrypt the TGT outside a workstation. Therefore, an external encryption device is required.Kerberos stores some keys in computers, e.g., session keys in a workstation and user keys in KDC. However, typical computers cannot store information securely. Information in a computer system is stored either in memory or in a hard disk, but neither is suciently secure. Therefore, secure storage outside a workstation and KDC is an important goal.As long as Kerberos uses passwords for secure information, dictionary attack cannot be solved completely. Therefore, it is desirable to replace passwords with randomly generated bits stored in tamper-resistant hardware(10).

2- Integration smart card into Kerberos

A user possesses 4 types of secret information –listed in order of criticalness-: The master key (replacing the user password), the user's PIN number, session key shared with the TGS, and session keys shared with different application servers. The criticalness of these keys depends on their lifetime. For instance, the master key has usually a long lifetime, whereas session keys have limited lifetime –i.e. few hours or minutes. Ideally all keys should not be exposed to the workstation, and hence all cryptographic functions making use of these keys have to be done by the smart card. Unfortunately, smart cards have limited processing power and one cannot migrate all of the cryptographic functions from the workstation to the smart card. In fact, there are a number of possible approaches to using smart cards with Kerberos authentication. The distinguishing feature is the amount of processing performed by the smart card, and hence the level of exposure of the keys.

Here are different approaches to integrating smart cards:

i. Key store: The simple memory card is used to store the master key. All keys are exposed to the workstation.

ii. TGT decryption: the user's master key is never exposed to the workstation, when the latter receives a TGT, it passes it to the smart where it is decrypted and the TGS session key is extracted and returned to the workstation. In this scenario, a corrupt workstation can still issue requests for a service to any application server the user has access rights.

iii. Application server ticket (AST) decryption: The TGS session key is never revealed to the workstation –as well as the master key-. The smart card would decrypt the Application server ticket and return the application server session key to the workstation. Thus, a corrupted workstation cannot request for new services–could only request services for those that were made explicitly by the user-.

iv. All tickets decrypted: This is an extreme solution, where all cryptographic processing is done by the smart card and no key is exposed to the workstation.

The two last solutions seem to introduce a lot of overhead and the first solution does not protect against a corrupt workstation –or a login facility-. The second approach is the most appropriate and is a reasonable compromise between complexity and functionality.

One of Kerberos weak point is that it does not authenticate the user when that user applies for an initial ticket (AS_REQ). A user's identity is only checked by the TGS when the user requests for an application server ticket. Solving this problem by pre-authentication it send user name and timpstamp encrypted with Ku for KDC to ensure that the client know the Ku. The present integration approach provides an early authentication of the user to the AS.[8]

7. MODEL TESTING

The testing based on the use of formal methods for automatic validation and conformance testing of security protocols in a Secure Java Card Application (for the E-Bank system). We perform validation and conformance testing because a large percentage of software is implemented with errors and, security applications cannot provide a guarantee to the confidentiality and integrity of information if the implementations are not correct.

1- Protocol Validation: The implementation of Kerberos in the system using high level protocol specification language and then perform validation using AVISPA tool[9] through HLPSL model of Kerberos.

2- Conformance Testing: Develop formal specifications using either LOTOS[14] or PROMELA[12] .

8.CONCLUSION

- All information exchange is encrypted by a secret key installed on the customer's smart card. The secret key would comprise of the user's pin code which is chosen at the time of installation and an E-banking key which is known only to the bank. The Java Card technology has to ensure that the E-bank's key is never exposed to any client application accessing the Java Card application.
- A smart card is used for storing the user's password as well as for decrypting the TGT sent back by the AS. One of the smart card functionality has to include a DES encryption/decryption algorithm, since DES is normally used by Kerberos. The overall objective is to enhance the security of Kerberos-based authentication. Any changes to

the original scheme should preferably introduce as little changes as possible to the way the Kerberos entities interact.

REGULATIONS AND STANDARDS

1- Common Criteria (CC): In 1998, the United States and other national governments adopted a new security evaluation scheme called the Common Criteria.Common Criteria was adopted by the International Organization for Standards (ISO) and International Electrotechnical Commission (IEC) as an international standard, ISO/IEC 15408, in 1999.(1)

2- The Payment Card Industry (PCI) Data Security Standard offers a single approach to safeguard sensitive data for all card brands worldwide. This standard is a result of a collaboration between Visa and MasterCard. It is designed to create common industry security requirements. Other card companies have endorsed the PCI Data Security Standard(16). For more information about the PCI Data Security Standard, refer to(17).

3- The Gramm-Leach Bliley Act (GLBA) applies to US Banking Security Regulations. This act is also known as the *Financial Modernization Act of 1999*. GLBA includes provisions to protect consumer personal financial information held by financial institutions.

REFERENCES

1. Debble Landon et al. "IBM System i Security Guide", IBM Redbooks, Oct 2006
2. Nalini K. Ratha and Ruud Bolle, "Smart Card based Authentication"
3. Microsoft Windows 2000 Server, Smart Card Logon, White Paper, Microsoft Co., 1999
4. Jan De Clercq ," Windows Server 2003 security infrastructures", October 2004
5. J. Kohl and C. Neuman, "The Kerberos Network Authentication Service (V5)", RFC 1510 ,September 1993
6. Stefan Stadlober Bakk, "An Evaluation of Security Threats and Countermeasures in Distributed RFID Infrastructures ", July 2005, TUG
7. Microsoft Windows 2003 Server, Technical Overview of Security for Windows Server, Microsoft Corporation, July 2002
8. Tariq Assaf, "Smart Card Technology and Integration", April 2001

9. The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications. 17th International Conference on Computer Aided Verification, 2005
10. Naomaru Itoi and Peter Honeyman, "Smartcard Integration with Kerberos V5", CITI, December 1998
11. Mark Lobel, "Case for Strong User Authentication ", RSA Security Inc.
12. The PROMELA Language http://www.dai-arc.polito.it/daiarc/manual/tools/jcat/main/node168.html.
13. Gary Ian Gaskell, "Integrating Smart Cards into Kerberos", Master Thesis, Feb 2000
14. T. Bolognesi and E. Brinksma, "Introduction to the iso specification language lotos",COMP. NETWORKS ISDN SYST., 14(1):25–59, 1987.
15. Faheem Khan ," Simplify enterprise Java authentication with single sign-on", IBM, Sep 2005
16. Visa Payment Card Industry Data Security Standard, http://usa.visa.com/download/business/ accepting _visa/ops_risk_management/cisp_PCI_Data_Security_Standard.pdf
17. Merchant e-solutions Payment Card Industry (PCI) Data Security Standard http://www.merchante-solutions.net/infosecurity/mandates.htm