SANS Security Essentials
GSEC Practical Assignment
Version 1.2d

Introduction to Security of LDAP Directory Services

Wenling Bao

May 16, 2001

Introduction

The Lightweight Directory Access Protocol (LDAP) is currently a widely used protocol for providing networking directory services on the TCP/IP model. It is based on the X.500, a standard protocol for directory services on the OSI model. The LDAP can be used to store and locate all kinds of information, including information about entities, such as organizations, individuals, and other network resources like file systems, applications and configuration information. For example, you want to search for an individual's information in the network. If you know its location like the domain name, the Domain Name System (DNS) can be used to map this domain name to a specific network address for you to find it out. What if you do not know the domain name? LDAP would help you search for the individual without knowing where it is located.

As a directory service provider, security is an essential issue for LDAP to protect its information and resources from confidentiality, integrity and availability attacks. As stated by Tim Howes, the co-author of LDAP and LDAP API, in his article "LDAP: Use as Directed" from NetworkMagazine.com on 02/01/99:

> *"LDAP also has an important role to play in tighter security, with the directory acting as gatekeeper and deciding who has access to what. In this capacity, LDAP performs two critical jobs. First, it serves as an authentication database. Second, once the identity of a user has been established, it controls access to resources, applications, and services".*

Background

Currently, the LDAP has two versions, version 2 and version 3. The LDAP v2 is the original version when LDAP was developed. It is defined in the RFC1777 specification. The LDAP v3, defined in RFC2251, is designed to provide more security and other functions that the LDAP v2 lacks.

Before we discuss the security features of LDAP, it is helpful to understand the structure of an LDAP directory. An LDAP directory is a special type of database to store information. It is designed to provide better performance for reading from than written to. It is organized as a simple tree-like hierarchy, which is called Directory Information Tree (DIT). The topmost level is the root or the source directory, which is generally the domain name component (dc) of a company, organization or a country. This level branches out to organizational units like departments, branches, divisions, etc. Below that are entries for individuals with common name (cn), such as users, specific network resources. Each entry stored in the structure has a distinguished name (dn) and its own attributes followed by specific values. The distinguished name is unique throughout the LDAP directory.

Here is an example of the dn of an entry (an individual) stored in a LDAP directory:

cn=Mike Robinson, ou=Computer Science Department, o=University of Michigan, c=United States

The meaning of the dn is explained as:

Country: United States

Organization: University of Michigan

Organizational unit: Computer Science Department

Common name: Mike Robinson

An LDAP directory can be stored on many distributed directory servers by replication. A distributed LDAP server is called a Directory System Agent (DSA). Each distributed DSA is synchronized regularly to keep data accuracy and integrity. When a client sends a request to a DSA, the LDAP server responds accordingly. If it is not able to process the request, the server forwards it to other LDAP servers to get solution.

Security Threats

Since the LDAP is a network protocol for directory services, like DNS, NIS and any other network protocols, it is subject to attacks and tampering from network. In addition, the directory servers can also be comprised by physical and remote attacks. So the security threats to the LDAP can be basically categorized into two groups, directory service oriented threats and non-directory service threats.

<u>Directory service oriented threats:</u>

- Unauthorized access to data by monitoring or spoofing authorized users' operations
- Unauthorized access to resources by physically retrieve others' authenticated connections and sessions
- Unauthorized modification or deletion of data, security settings and other configurations

- Spoofing of directory services: To gain access information of others, deceiving valid users to a faked directory, by interjecting misleading information into the normal communications between the client and the real server.
- Excessive use of resources

Non-directory service oriented threats:

- Common network-based attacks against the LDAP servers, including the operating system, opening ports, processes and services running on the hosts, to comprise the availability of resources. Accomplished by viruses, worms, Trojan horses, etc.
- Attacks against the hosts by physically access the resources: operating system, files and directories, peripheral equipments, and so forth. This could affect the availability, integrity and confidentiality of resources.
- Attacks against the back-end databases providing directory services

Authentication

The LDAP is based on client-server model. To access the LDAP directory service, the LDAP client must tell the LDAP server who it is, i.e., authenticate itself to the server. Once the identity of the client has been established, the server will decide what resources, applications, and services the client are permitted to access according to its identity. This is called authorization, or access control. So to implement the security features of LDAP, the authentication and authorization are the key aspects. In addition, another aspect of the LDAP security is the way in which the client and the server are communicated.

In LDAP, authentication information is provided in the 'Bind' operation. Defined by Yeong, W. etc. in the "Lightweight Directory Access Protocol", RFC 1777, in 1995, "The function of the Bind Operation is to initiate a protocol session between a client and a server, and to allow the authentication of the client to the server."

Authentication mechanisms in LDAP v2

In LDAP v2, the bind request must be the first communication initiated by a client to a LDAP server, containing the authentication information. The authentication methods supported by LDAP include: simple, anonymous, and Kerberos version 4.

The anonymous authentication occurs when no specific authentication method has been chosen. Under such circumstances, the client will connect to the server as anonymous identity, provided that the server allows anonymous connections and allow certain data access for anonymous users.

The simple authentication method is to send the LDAP server the authentication field with only of the client's clear-text password. Certainly, this mechanism has security problems because the password is sent in plain text and is readable if tampered from the network. To avoid the password being exposed when the simple authentication is used, the communications between client and server should be going through a secure channel, such as SSL. Without underlying a secure transferring way, use of the simple authentication is highly vulnerable and should be disabled.

The Kerberos 4 authentication mechanism can be implemented either by the LDAP servers, or by the DSAs respectively. For both of implementations, the kerberos ticket for the service and the service name running on the provider are required to be presented from the client as authentication information. The client has to respond to two challenges provided by the server for this authentication before the data communication could start. In the beginning, the server provides a 32-bit random number in network byte order as the first challenge. The client has to respond the server with its kerberos ticket and the authenticator, where the server's service name and hostname are included, as well as the original random number in network byte order from the server, encrypted in the checksum field. When the server gets the response from the client, it checks its kerberos ticket and authenticator. If it is successful, it will then verify that the checksum consisting of the 32-bit random number is the same as the original number provided by the server itself. After this verification is completed, the server generates the second challenge, by adding a piece of 8-octet of data encrypted in the DES ECB mode, and sends it out. The 8-octet of data contains the checksum issued by the server and the checksum information issued in the previous client's response. So when the client receives the challenge from the server, it first checks the checksum in the 8-octet data field to verify the identity of the server. The client then must respond with the encrypted data by the DCE PCBC mode, which includes the checksum previously issued by the server and the authorization identity information. In this response, the server will verify both its original checksum and the authorization identity using the kerberos ticket provided by the client. If they are verified, the authentication process is successful. The data connection starts.

Authentication mechanisms in LDAP v3

The authentication mechanisms supported in LDAP version 3 are anonymous, simple, and the Simple Authentication and Security Layer (SASL) authentication. The Simple Authentication is still supported in version 3, same as version 2. The SASL is a scheme used for launching a security layer between connection-based protocols, to protect all communications between the LDAP server and the LDAP client for the purpose of authentication. This is set up upon the type of authentication mechanism coordinated by both sides. So this method provides more flexibility for optimizing the authentication method, depending on the real situation and agreement of both participants.

Several SASL mechanisms are currently defined: CRAM-MD5, DIGEST-MD5, External, Kerberos v4 and v5, Anonymous, S/Key, Generic Security Service Application Program Interface (GSSAPI), Keyd-Hashing, and so forth. The LDAP v3 server and the LDAP v3 client can use any SASL mechanisms, as long as both the server and the client support. There is not a standard SASL for current LDAP versions. The LDAP servers from various vendors support different mechanisms. For instance, the iPlanet^{TM} Directory Server from Netscape® supports the CRAM-MD5 and the External mechanisms. In the following section, various SASL mechanisms will be briefly described.

The Challenge-Response Authentication Mechanism (CRAM) used for authentication is CRAM-MD5. The Keyed MD5 Message-Digest algorithm is used in this mechanism for encrypting a series of characters represented in hexadecimal format, as challenge-response authentication. The key for this computation is a shared secret only recognized by the client and server.

The Digrest-MD5 applies the HTTP Digest Access Authentication, specified in the RFC2617, to the SASL method for challenge-response. It is designed to make improvements over the CRAM-MD5, by adding the integrity protection on the application layer, authentication identity information by third party servers and other security enhancements.

The External authentication mechanism, as the name indicates, means that the server applies the external authentication information to SASL to verify the client's identity. The external resource may be provided by IP sec, Transport Layer Security (TLS), Secure Socket Layer (SSL) and other security mechanisms. For example, SSL can accomplish authentication and other security verifications prior to LDAP authentication, because LDAP is established at the application layer, above the SSL. Therefore, the verified authentication information by SSL may be used to LDAP authentication, as external security information.

In the S/Key mechanism, the MD4 message-digest algorithm is used. The client issues a response with the authorization identity information. The challenge from the server consists of a decimal number and information for that authorization identity. The client must respond with an encrypted one-time password, which will be verified by the server for authentication.

One issue needs to be pointed out is, that, any connection before authentication is in clear-text and could be modified by an attacker. So the important communications should be established after the authentication is completed. If important information exchanges have to be done prior to authentication, they should be revalidated after.

Authorization

Authorization is access control to the authenticated clients. Access control is determined by the access control list (ACL), which is configured and implemented in various ways by different LDAP directory server vendors. As an example, let us look at the authorization implementation of iPlanet^{TM} Directory Server.

In the iPlanet^{TM} Directory Server structure, the ACL is maintained in the DIT. The ACL is stored by a set of access control instructions (ACIs), as a specific attribute attached to an object in the DIT. The permissions defined by ACIs are Read, Write, Search, Compare, Selfwrite, Add and Delete. ACIs take effects in hierarchy. The ACIs rule on objects in lower level takes precedence over ACIs on objects in higher levels. We still use the previous example of the individual 'Mike Robinson' in a DIT. The ACIs on 'Mike Robinson' as common name has priority over the ACIs on 'Computer Science Department' as organizational unit, which in turn, takes precedence over 'University of Michigan' as organization.

Other Security Considerations

So far, authentication and authorization in LDAP have been discussed. To protect confidentiality, integrity of data, and non-repudiation, establishing secure requests and responses between the client and the server is also important. To achieve this, the communications in between need be carried out through secure channels or sockets. Currently, most LDAP servers allow their services to be accessed by secure sockets and channels, to provide higher levels of security for connections with clients, such as SSL. To establish SSL connections, a port number should be specified to run the service on the LDAP server. Generally, the LDAP server uses port 636, as a standard SSL socket number of LDAP for TCP and UDP. The directory server can also support custom sockets. But the client has to identify the appropriate socket to access the directory services on the server through SSL.

References

Bialaski Tom, "Directory Server Security", December 2000
http://www.sun.com/blueprints/1200/ldap-security.pdf

Haller N., "The S/KEY One-Time Password System", RFC 1760, February 1995

Hodges Jeff, "LDAP Directory Services: Security", Aug 12, 1999
http://www.stanford.edu/~hodges/talks/WebSec99/DirectoryServiceSecurity-1999-08-11/

Howes Tim., "LDAP: Use as Directed", Feb 1, 1999
http://www.networkmagazine.com/article/DCM20000502S0039

http://192.9.48.9/products/jndi/tutorial/ldap/TOC.html

"Introduction to LDAP"
http://fr2.php.net/manual/pt_BR/ref.ldap.php

Klensin J., Catoe R., and Krumviede P., "IMAP/POP Authorize Extension for Simple Challenge/Response", RFC2195, September 1997

Leach P. and Newman C., "Using Digest Authentication as a SASL Mechanism", RFC2831, May 2000

Marshall Brad, "Introduction to LDAP".
http://staff.pisoftware.com/bmarshal/publications/ldap_tut.html

Meyers, J., "Simple Authentication and Security Layer", RFC2222, October 1997

Wahl M., Howes T., and Kille S., "Lightweight Directory Access Protocol (v3)", RFC2251, December 1997

Yeong, W., Howes, T., and Kille S., "Lightweight Directory Access Protocol", RFC1777, March 1995