CAT Working Group
Internet Draft
Document: draft-swift-win2k-krb-user2user-00.txt
Category: Informational

Michael M. Swift
Microsoft

October 1999

User to User Kerberos Authentication using GSS-API

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026 [1].

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts. Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

1. Abstract

This draft documents a simple extension to the Kerberos GSS-API mechanism to support user to user authentication both in the case where the client application explicitly requests user to user authentication and when it does not know whether the server supports user to user authentication. This is used in Microsoft's Windows 2000 Implementation of Kerberos.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [2].

3. Introduction

The Kerberos user to user authentication mechanism allows for a client application to connect to a service that is not in possession of a long term secret key. Instead, the authentication request (AP request) is encrypted using the session key from the serviceÆs ticket granting ticket. According to RFC 1510 [3]:

> If the ENC-TKT-IN-SKEY option has been specified and an additional ticket has been included in the request, the KDC will decrypt the additional ticket using the key for the server

to which the additional ticket was issued and verify that it is a ticket-granting ticket. If the request succeeds, the session key from the additional ticket will be used to encrypt the new ticket that is issued instead of using the key of the server for which the new ticket will be used (This allows easy implementation of user-to-user authentication, which uses ticket-granting ticket session keys in lieu of secret server keys in situations where such secret keys could be easily compromised).

The current Kerberos GSS-API mechanism does not support this flavor of authentication, and new messages and flags are defined to add this support. For the case that the client knows that the service requires user-to-user authentication, a new message (KERB-TGT-REQUEST) is defined. In the case that a client sends a normal AP request but the service only supports user-to-user authentication, a new Kerberos error as well as error data type is defined.

4. User to User as a New Mechanism

In the case that the client application knows that the server only supports user-to-user authentication, then it is easiest to add this functionality as a new mechanism. The new protocol extends the existing Kerberos GSS-API protocol by adding an additional round trip to request the TGT from the service. As with all Kerberos GSS-API messages, the following tokens are encapsulated in the GSS-API framing. The first token of the exchange is as follows:

```
KERB-TGT-REQUEST ::= SEQUENCE {
        pvno[0]                         INTEGER,
        msg-type[1]                     INTEGER,
        server-name[2]                  PrincipalName OPTIONAL,
        realm[3]                        Realm OPTIONAL
}
```

The TGT request consists of four fields:

pvno and msg-type are as defined in RFC1510 section 5.4.1. msg-type is KRB_TGT_REQ (16).

server-name û this field optionally contains the name of the server. If the client application doesnÆt know the server name this can be left blank and the server application will pick the appropriate server credentials.

realm û this field optionally contains the realm of the server. If the client application doesnÆt know the server realm this field can be left blank and the server application will pick the appropriate server credentials.

The server name and realm are included to allow a server application to act for multiple principles in different realms and to choose

which credentials to use. Depending on the implementation of the Kerberos mechanism, the application may call gss_accept_sec_context() multiple times until the token is accepted.

The response to the KERB-TGT-REQUEST message is as follows:

```
KERB-TGT-REPLY ::= SEQUENCE {
        pvno[0]                         INTEGER,
        msg-type[1]                     INTEGER,
        ticket[2]                       Ticket
        }
```

The TGT reply contains the following fields:

pvno and msg-type are as defined in RFC1510 section 5.4.1. msg-type is KRB_TGT_REP (17)

ticket û contains the TGT for the service specified by the server name and realm passed by the client or the default service.

If the service does not possess a ticket granting ticket, it should return the error KRB_AP_ERR_NO_TGT (0x43).

If the server name and realm in the KERB-TGT-REQUEST message do not match the name of the service, then the service should return the error KRB_AP_ERR_NOT_US.

The mechanism ID for user to user GSS-API Kerberos, in accordance with the mechanism proposed by SPNEGO for negotiating protocol variations, is:

{iso(1) member-body(2) United States(840) mit(113554) infosys(1) gssapi(2) krb5(2) usertouser(3)}

Following the exchange of the TGT request messages, the rest of the authentication is identical to the Kerberos GSS-API mechanism defined in RFC 1964 [4].

As with the Kerberos GSS-API mechanism, the innerContextToken field of the context establishment tokens contain context message (KERB-TGT-REQUEST, KERB-TGT-REPLY) preceded by a 2-byte TOK_ID field containing 04 00 for the KERB-TGT-REQUEST message and 04 01 for the KERB-TGT-REPLY message.

5. User to User With The Existing Mechanism

In the case that the client application doesnÆt know that a service requires user-to-user authentication and sends a normal AP request, it may be useful to recover and have the server return the TGT in the error message. In this case, the server returns a KRB-ERROR message with the KRB_AP_ERR_USER_TO_USER_REQUIRED (0x43). The error

data for contains a KERB-TGT-REPLY. The Kerberos mechanism then continues as in [4] but with a user-to-user ticket instead of a normal session ticket.

6. Security Considerations

There is some risk in a server handing out its ticket-granting-ticket to any client that requests it, in that it gives an attacker a piece of encrypted material to decrypt. However, the same material may be obtained from listening to any legitimate client connect.

7. References

1 Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

2 Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997

3 J. Kohl, C. Neuman, "The Kerberos Network Authentication Service(V5)", RFC 1510.

4 J. Linn, "The Kerberos Version 5 GSS-API Mechanism", RFC 1964

8. Author's Addresses

Michael Swift
Microsoft
One Microsoft Way
Redmond, Washington
Email: mikesw@microsoft.com

Full Copyright Statement