Kerberos Working Group

Internet Draft
Document: draft-ietf-krb-wg-kerberos-referrals-03.txt
Category: Standards Track

Karthik Jaganathan
Larry Zhu
John Brezak
Microsoft
Mike Swift
University of Washington
Jonathan Trostle
Cisco Systems

Expires: August 2004

Generating KDC Referrals to locate Kerberos realms

Status of this Memo

1. Abstract

The draft documents a new method for a Kerberos Key Distribution Center (KDC) to respond to client requests for kerberos tickets when the client does not have detailed configuration information on the realms of users or services. The KDC will handle requests for principals in other realms by returning either a referral error or a cross-realm TGT to another realm on the referral path. The clients will use this referral information to reach the realm of the target principal and then receive the ticket.

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

Current implementations of the Kerberos AS and TGS protocols, as defined in [3], use principal names constructed from a known user or service name and realm. A service name is typically constructed from a name of the service and the DNS host name of the computer that is providing the service. Many existing deployments of Kerberos use a single Kerberos realm where all users and services would be using the same realm. However in an environment where there are multiple trusted Kerberos realms, the client needs to be able to determine what realm a particular user or service is in before making an AS or TGS request. Traditionally this requires client configuration to make this possible.

When having to deal with multiple trusted realms, users are forced to know what realm they are in before they can obtain a ticket granting ticket (TGT) with an AS request. However, in many cases the user would like to use a more familiar name that is not directly related to the realm of their Kerberos principal name. A good example of this is an RFC-822 style email name. This document describes a mechanism that would allow a user to specify a user principal name that is an alias for the user's Kerberos principal name. In practice this would be the name that the user specifies to obtain a TGT from a Kerberos KDC. The user principal name no longer has a direct relationship with the Kerberos principal or realm. Thus the administrator is able to move the user's principal to other realms without the user having to know that it happened.

Once a user has a TGT, they would like to be able to access services in any trusted Kerberos realm. To do this requires that the client be able to determine what realm the target service's host is in before making the TGS request. Current implementations of Kerberos typically have a table that maps DNS host names to corresponding Kerberos realms. In order for this to work on the client, each application canonicalizes the host name of the service by doing a DNS lookup followed by a reverse lookup using the returned IP address. The returned primary host name is then used in the construction of the principal name for the target service. In order for the correct realm to be added for the target host, the mapping table [domain_to_realm] is consulted for the realm corresponding to the DNS host name. The corresponding realm is then used to complete the target service principal name.

This traditional mechanism requires that each client have very detailed configuration information about the hosts that are providing services and their corresponding realms. Having client side configuration information can be very costly from an administration point of view - especially if there are many realms and computers in the environment.

There are also cases where specific DNS aliases (local names) have been setup in an organization to refer to a server in another organization (remote server). The server has different DNS names in

each organization and each organization has a Kerberos realm that is configured to service DNS names within that organization. Ideally users are able to authenticate to the server in the other organization using the local server name. This would mean that the local realm be able to produce a ticket to the remote server under its name. You could give that remote server an identity in the local realm and then have that remote server maintain a separate secret for each alias it is known as. Alternatively you could arrange to have the local realm issue a referral to the remote realm and notify the requesting client of the server's remote name that should be used in order to request a ticket.

This draft proposes a solution for these problems and simplifies administration by minimizing the configuration information needed on each computer using Kerberos. Specifically it describes a mechanism to allow the KDC to handle Canonicalization of names, provide for principal aliases for users and services and provide a mechanism for the KDC to determine the trusted realm authentication path by being able to generate referrals to other realms in order to locate principals.

To rectify these problems, this draft introduces three new kinds of KDC referrals:

1. AS ticket referrals, in which the client doesn't know which realm contains a user account.
2. TGS ticket referrals, in which the client doesn't know which realm contains a server account.
3. Cross realm shortcut referrals, in which the KDC chooses the next path on a referral chain

4. Realm Organization Model

This draft assumes that the world of principals is arranged on multiple levels: the realm, the enterprise, and the world. A KDC may issue tickets for any principal in its realm or cross-realm tickets for realms with which it has a direct trust relationship. The KDC also has access to a trusted name service that can resolve any name from within its enterprise into a realm. This trusted name service removes the need to use an untrusted DNS lookup for name resolution.

For example, consider the following configuration, where lines indicate trust relationships:

```
               MS.COM
             /        \
            /          \
     OFFICE.MS.COM    NT.MS.COM
```

In this configuration, all users in the MS.COM enterprise could have a principal name such as alice@MS.COM, with the same realm portion. In addition, servers at MS.COM should be able to have DNS host names

from any DNS domain independent of what Kerberos realm their principal resides in.

5. Client Name Canonicalization

A client account may have multiple principal names. More useful, though, is a globally unique name that allows unification of email and security principal names. For example, all users at MS may have a client principal name of the form "joe@MS.COM" even though the principals are contained in multiple realms. This global name is again an alias for the true client principal name, which indicates what realm contains the principal. Thus, accounts "alice" in the realm NT.MS.COM and "bob" in OFFICE.MS.COM may logon as "alice@MS.COM" and "bob@MS.COM".

This utilizes a new client principal name type, as the AS-REQ message only contains a single realm field, and the realm portion of this name doesn't correspond to any Kerberos realm. Thus, the entire name "alice@MS.COM" is transmitted in the client name field of the AS-REQ message, with a name type of KRB-NT-ENTERPRISE-PRINCIPAL.

```
KRB-NT-ENTERPRISE-PRINCIPAL     10
```

The KDC will recognize this name type and then transform the requested name into the true principal name. The true principal name can be using a name type different from the requested name type. Typically the returned principal name will be a KRB-NT-PRINCIPAL. The returned name will be the same in the AS response and in the ticket. The KDC will always return a different name type than KRB-NT-ENTERPRISE-PRINCIPAL. This is regardless of the presence of the "canonicalize" KDC option.

If the "canonicalize" KDC option is set, then the KDC MAY change the client principal name and type in the AS response and ticket regardless of the name type of the client name in the request. For example the AS request may specify a client name of "fred@MS.COM" as an KRB-NT-PRINCIPAL with the "canonicalize" KDC option set and the KDC will return with a client name of "104567" as a KRB-NT-UID.

6. Requesting a referral

In order to request referrals, the Kerberos client must explicitly request the canonicalize KDC option (bit 15) in the KDC options for the TGS-REQ. This flag indicates to the KDC that the client is prepared to receive a reply that contains a principal name other than the one requested. Thus, the KDCOptions types is redefined as:

```
     KDCOptions ::=   BIT STRING {
                       reserved(0),
                       forwardable(1),
                       forwarded(2),
                       proxiable(3),
                       proxy(4),
```

```
                       allow-postdate(5),
                       postdated(6),
                       unused7(7),
                       renewable(8),
                       unused9(9),
                       unused10(10),
                       unused11(11),
                       canonicalize(15),
                       renewable-ok(27),
                       enc-tkt-in-skey(28),
                       renew(30),
                       validate(31)
      }
```

The client should expect, when sending names with the "canonicalize" KDC option, that names in the KDC's reply will be different than the name in the request.

6.1 Client Referrals

The simplest form of ticket referral is for a user requesting a ticket using an AS-REQ. In this case, the client machine will send the AS request to a convenient trusted realm, either the realm of the client machine or the realm of the client name. In the case of the name Alice@MS.COM, the client may optimistically choose to send the request to MS.COM. The realm in the AS request is always the name of the realm that the request is for as specified in [3].

The client will send the string "alice@MS.COM" in the client principal name field using the KRB-NT-ENTERPRISE-PRINCIPAL name type with the crealm set to MS.COM. The KDC will try to lookup the name in its local account database. If the account is present in the realm of the request, it MUST return a KDC reply structure with the appropriate ticket.

If the account is not present in the realm specified in the request and the "canonicalize" KDC option is set, the KDC will try to lookup the entire name, Alice@MS.COM, using a name service. If this lookup is unsuccessful, it MUST return the error KDC_ERR_C_PRINCIPAL_UNKNOWN. If the lookup is successful, it MUST return an error KDC_ERR_WRONG_REALM (0x44) and in the error message the crealm field will contain the the true realm of the client or another realm that has better information about the client's true realm. The client MUST NOT use a cname returned from a referral.

If the KDC contains the account locally and "canonicalize" KDC option is not set, it MUST return a normal ticket. The client name and realm portions of the ticket and KDC reply message MUST be the client's true name in the realm, not the globally unique name.

If the client receives a KDC_ERR_WRONG_REALM error, it will issue a new AS request with the same client principal name used to generate the first referral to the realm specified by the realm field of the

kerberos error message from the first request. This request MUST produce a valid AS response with a ticket for the canonical user name.

An implementation should limit the number of referrals that it processes to avoid infinite referral loops. A suggested limit is 5 referrals before giving up. In MicrosoftÆs implementation the default limit is 3 since through the use of the global catalog any domain in one forest is reachable from any other domain in another trusting forest with 3 or less referrals.

6.2 Service Referrals

The primary problem is that the KDC must return a referral ticket rather than an error message as is done in AS request referrals. There needs to be a place to include in the TGS response information about what realm contains the service. This is done by returning information about the service name in the pre-auth data field of the KDC reply.

If the KDC resolves the service principal name into a principal in the realm specified by the service realm name, it will return a normal ticket. When using canonicalization, the client can omit the service realm name. If it is supplied, it is used as a hint by the KDC, but the service principal lookup is not constrained to locating the service principal name in that specified realm. If the "canonicalize" flag in the KDC options is not set, then the KDC MUST only look up the name as a normal principal name in the specified service realm.

If the "canonicalize" flag in the KDC options is set and the KDC doesn't find the principal locally, the KDC can return a cross-realm ticket granting ticket to the next hop on the trust path towards a realm that may be able to resolve the principal name.

If the KDC can determine the service principal's realm, it SHOULD return the service realm as KDC supplied pre-authentication data element. The preauth data MUST be encrypted using the sub-session key from the authenticator if present or the session key from the ticket.

The data itself is an ASN.1 encoded structure containing the server's realm, and if known, the real principal name.

```
PA-SERVER-REFERRAL-INFO        25

PA-SERVER-REFERRAL :: = KERB-ENCRYPTED-DATA
                           -- PA-SERVER-REFERRAL-DATA

PA-SERVER-REFERRAL-DATA ::= SEQUENCE {
        referred-server-realm[0]  KERB-REALM
        referred-name[1]         PrincipalName OPTIONAL
        ...
```

```
}
```

If applicable to the encryption type, the key derivation value will for the PA-SERVER-REFERRAL is 22.

If the referred-name field is present, the client MUST use that name in a subsequent TGS request to the service realm when following the referral.

The client will use this information to request a chain of cross-realm ticket granting tickets until it reaches the realm of the service, and can then expect to receive a valid service ticket.

However an implementation should limit the number of referrals that it processes to avoid infinite referral loops. A suggested limit is 5 referrals before giving up.

This is an example of a client requesting a service ticket for a service in realm NT.MS.COM where the client is in OFFICE.MS.COM.

```
+NC = Canonicalize KDCOption set
+PA-REFERRAL = returned PA-SERVER-REFERRAL-INFO

C: TGS-REQ sname=server/foo.nt.ms.com srealm=NULL +NC to
OFFICE.MS.COM
S: TGS-REP sname=krbtgt/MS.COM@OFFICE.MS.COM +PA-REFERRAL
containing NT.MS.COM
C: TGS-REQ sname=krbtgt/NT.MS.COM@MS.COM +NC to MS.COM
S: TGS-REP sname=krbtgt/NT.MS.COM@MS.COM
C: TGS-REQ sname=server/foo.nt.ms.com srealm=NT.MS.COM +NC to
NT.MS.COM
S: TGS-REP sname=server/foo.nt.ms.com@NT.MS.COM
```

Notice that the client only specifies the service name in the initial and final TGS request.

7. Cross Realm Routing

The current Kerberos protocol requires the client to explicitly request a cross-realm TGT for each pair of realms on a referral chain. As a result, the client need to be aware of the trust hierarchy and of any short-cut trusts (those that aren't parent-child trusts). Instead, the client should be able to request a TGT to the target realm from each realm on the route. The KDC will determine the best path for the client and return a cross-realm TGT. The client has to be aware that a request for a cross-realm TGT may return a TGT for a realm different from the one requested.

For compatibility, the client MUST use the "canonicalize" KDC option if it is able to use cross-realm routing from the KDC.

8. Compatibility with earlier implementations of name canonicalization

The Microsoft Windows 2000 release included an earlier form of name-canonicalization [4]. It has these differences:

1) The TGS referral data was returned inside of the KDC message as "encrypted pre auth data".

```
          KERB-ENCRYPTED-KDC-REPLY ::=  SEQUENCE {
                  session-key[0]   KERB-ENCRYPTION-KEY,
                  last-request[1]  PKERB-LAST-REQUEST,
                  nonce[2]         INTEGER,
                  key-expiration[3] KERB-TIME OPTIONAL,
                  flags[4]         KERB-TICKET-FLAGS,
                  authtime[5]      KERB-TIME,
                  starttime[6]     KERB-TIME OPTIONAL,
                  endtime[7]       KERB-TIME,
                  renew-until[8]   KERB-TIME OPTIONAL,
                  server-realm[9]  KERB-REALM,
                  server-name[10]  KERB-PRINCIPAL-NAME,
                  client-addresses[11] PKERB-HOST-ADDRESSES
          OPTIONAL,
                  encrypted-pa-data[12] SEQUENCE OF KERB-PA-DATA
          OPTIONAL
          }
```

2) The preauth data type definition in the encrypted preauth data is as follows:

```
          PA-SVR-REFERRAL-INFO        20

          PA-SVR-REFERRAL-DATA ::= SEQUENCE {
                  referred-server-name[1]  PrincipalName OPTIONAL
                  referred-server-realm[0] KERB-REALM
          }
```

9. Security Considerations

In the case of TGS requests the client may be vulnerable to a denial of service attack by an attacker that replays replies from previous requests. The client can verify that the request was one of its own by checking the client-address field or authtime field, though, so the damage is limited and detectable. Clients MUST NOT process cross realm referral TGTs if the KDC reply does not include the encrypted PA-SERVER-REFERRAL-INFO.

For the AS exchange case, it is important that the logon mechanism not trust a name that has not been used to authenticate the user. For example, the name that the user enters as part of a logon exchange may not be the name that the user authenticates as, given that the KDC_ERR_WRONG_REALM error may have been returned. The relevant Kerberos naming information for logon (if any), is the

client name and client realm in the service ticket targeted at the workstation that was obtained using the user's initial TGT.

How the client name and client realm is mapped into a local account for logon is a local matter, but the client logon mechanism MUST use additional information such as the client realm and/or authorization attributes from the service ticket presented to the workstation by the user, when mapping the logon credentials to a local account on the workstation.

10. Acknowledgements

The authors wish to thank Ken Raeburn for his comments and suggestions.

11.1 Normative 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 Neuman, C., Kohl, J., Ts'o, T., Yu, T., Hartman, S., and K. Raeburn, "The Kerberos Network Authentication Service (V5)", draft-ietf-krb-wg-kerberos-clarifications-00.txt, February 22, 2002. Work in progress.

11.2 Informative References

4 J. Trostle, I. Kosinovsky, and M. Swift,"Implementation of Crossrealm Referral Handling in the MIT Kerberos Client", In Network and Distributed System Security Symposium, February 2001.

12. Author's Addresses

Karthik Jaganathan
Microsoft
One Microsoft Way
Redmond, Washington
Email: karthikj@Microsoft.com

Larry Zhu
Microsoft
One Microsoft Way
Redmond, Washington
Email: lzhu@Microsoft.com

Michael Swift
University of Washington

Seattle, Washington
Email: mikesw@cs.washington.edu

John Brezak
Microsoft
One Microsoft Way
Redmond, Washington
Email: jbrezak@Microsoft.com

Jonathan Trostle
Cisco Systems
170 W. Tasman Dr.
San Jose, CA 95134
Email: jtrostle@cisco.com

Full Copyright Statement