NETWORK WORKING GROUP
Internet-Draft
Updates: 4120 (if approved)
Intended status: Standards Track
Expires: November 16, 2008

L. Zhu
P. Leach
Microsoft Corporation

May 15, 2008

Anonymity Support for Kerberos
draft-ietf-krb-wg-anon-06

Status of this Memo

Abstract

This document defines extensions to the Kerberos protocol for the Kerberos client to authenticate the Kerberos Key Distribution Center and the Kerberos server, without revealing the client's identity. It updates RFC 4120. These extensions can be used to secure communication between the anonymous client and the server.

Table of Contents

1. Introduction

In certain situations, the Kerberos [RFC4120] client may wish to authenticate a server and/or protect communications without revealing its own identity. For example, consider an application which provides read access to a research database, and which permits queries by arbitrary requestors. A client of such a service might wish to authenticate the service, to establish trust in the information received from it, but might not wish to disclose its identity to the service for privacy reasons.

Extensions to Kerberos are specified in this document by which a client can authenticate the Key Distribution Center (KDC) and request an anonymous ticket. The client can use the anonymous ticket to authenticate the server and protect subsequent client-server communications.

By using the extensions defined in this specification, the client may reveal its identity in its initial request to its own KDC, but it can remain anonymous thereafter to KDCs on the cross-realm authentication path, and to the server with which it communicates.

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 [RFC2119].

3. Definitions

The anonymous Kerberos realm name is defined as a well-known realm name based on [KRBNAM]. The value is the literal "WELLKNOWN: ANONYMOUS". An anonymous Kerberos realm name MUST NOT be present in the transited field of a ticket. However, as specified in Section 4, the true name of the realm that issued the anonymous ticket MAY be present in the transited field of a ticket.

The anonymous Kerberos principal name is defined as a well-known Kerberos principal name based on [KRBNAM]. The value of the name-type field is KRB_NT_WELLKNOWN [KRBNAM], and the value of the name-string field is a sequence of two KerberosString components: "WELLKNOWN", "ANONYMOUS".

Note that in this specification, the anonymous principal name and realm are only applicable to the client in Kerberos messages, the server MUST NOT be anonymous in any Kerberos message.

The anonymous ticket flag is defined as bit 14 (with the first bit being bit 0) in the TicketFlags:

```
TicketFlags     ::= KerberosFlags
  -- anonymous(14)
  -- TicketFlags and KerberosFlags are defined in [RFC4120]
```

An anonymous ticket is a ticket that has all of the following properties:

- The cname field contains the anonymous Kerberos principal name.
- The crealm field contains the client's realm name, or the name of the realm that issued the initial ticket for the client principal (when the anonymous ticket is obtained using anonymous PKINIT from a Kerberos realm other than that of the client as described in Section 4), or the anonymous realm name.
- The anonymous ticket contains no information that can reveal the client's identity. However the ticket may contain the client realm, intermediate realms on the client's authentication path, and authorization data that may provide information related to the client's identity. For example, an anonymous principal that is identifiable only within a particular group of users can be implemented using authorization data and such authorization data, if included in the anonymous ticket, shall disclose the client's membership of that group.
- The anonymous ticket flag is set.

The anonymous KDC option is defined as bit 14 (with the first bit being bit 0) in the KDCOptions:

```
KDCOptions      ::= KerberosFlags
  -- anonymous(14)
  -- KDCOptions and KerberosFlags are defined in [RFC4120]
```

As described in Section 4, the anonymous KDC option is set to request an anonymous ticket.

4. Protocol Description

In order to request an anonymous ticket, the client sets the anonymous KDC option in an Authentication Exchange (AS) or Ticket Granting Service (TGS) request. The client can request an anonymous Ticket Granting Ticket (TGT) based on a normal TGT. Unless otherwise specified, the client can obtain an anonymous ticket with the

anonymous realm name only by requesting an anonymous ticket in an AS exchange with the client realm set as anonymous in the request.

If the client wishes to authenticate to the KDC anonymously, it sets the client name as anonymous in the AS exchange and provides a PA_PK_AS_REQ pre-authentication data [RFC4556] where both the signerInfos field and the certificates field of the SignedData [RFC3852] of the PA_PK_AS_REQ are empty. Because the anonymous client does not have an associated asymmetric key pair, the client MUST choose the Diffie-Hellman key agreement method by filling in the Diffie-Hellman domain parameters in the clientPublicValue [RFC4556].

If the ticket in the PA-TGS-REQ of the TGS request is anonymous, or if the client in the AS request is anonymous, the anonymous KDC option MUST be set in the request. Otherwise, the KDC MUST return a KRB-ERROR message with the code KDC_ERR_BADOPTION.

Upon receiving the AS request with a PA_PK_AS_REQ [RFC4556] from the anonymous client, the KDC processes the request according to Section 3.1.2 of [RFC4120]. The KDC skips the checks for the client's signature and the client's public key (such as the verification of the binding between the client's public key and the client name), but performs otherwise-applicable checks, and proceeds as normal according to [RFC4556]. For example, the AS MUST check if the client's Diffie-Hellman domain parameters are acceptable. The Diffie-Hellman key agreement method MUST be used and the reply key is derived according to Section 3.2.3.1 of [RFC4556]. If the clientPublicValue is not present in the request, the KDC MUST return a KRB-ERROR with the code KDC_ERR_PUBLIC_KEY_ENCRYPTION_NOT_SUPPORTED [RFC4556]. If all goes well, an anonymous ticket is generated according to Section 3.1.3 of [RFC4120] and a PA_PK_AS_REP [RFC4556] pre-authentication data is included in the KDC reply according to [RFC4556]. If the KDC does not have an asymmetric key pair, it MAY reply anonymously or reject the authentication attempt. If the KDC replies anonymously, both the signerInfos field and the certificates field of the SignedData [RFC3852] of PA_PK_AS_REP in the reply are empty. The server name in the anonymous KDC reply contains the name of the TGS.

The KDC conforming to this specification MUST indicate the support of anonymous PKINIT as described above in this section according to Section 3.4 of [RFC4556].

Upon receipt of the KDC reply that contains an anonymous ticket and a PA_PK_AS_REP [RFC4556] pre-authentication data, the client can then authenticate the KDC based on the KDC's signature in the PA_PK_AS_REP. If the KDC's signature is missing in the KDC reply (the reply is anonymous), the client MUST reject the returned ticket

if it cannot authenticate the KDC otherwise.

The client can use the client keys to mutually authenticate with the KDC, request an anonymous TGT in the AS request. And in that case, the reply key is selected as normal according to Section 3.1.3 of [RFC4120].

For the TGS exchange, the reply key is selected as normal according to Section 3.3.3 of [RFC4120].

When policy allows, the KDC issues an anonymous ticket. Based on local policy, the client realm in the anonymous ticket can be the anonymous realm name or the realm of the KDC. However, in all cases, the client name and the client realm in the EncTicketPart of the reply MUST match with the corresponding client name and the client realm of the anonymous ticket in the reply. The client MUST use the client name and the client realm returned in the KDC-REP in subsequent message exchanges when using the obtained anonymous ticket.

When propagating authorization data in the ticket or in the enc-authorization-data field of the request, the TGS MUST ensure that the client confidentiality is not violated in the returned anonymous ticket. The TGS MUST process the authorization data recursively according to Section 5.2.6 of [RFC4120] beyond the container levels such that all embedded authorization elements are interpreted. Identity-based authorization data SHOULD NOT be present in an anonymous ticket in that it typically reveals the client's identity. The specification of a new authorization data type MUST specify the processing rules of the authorization data when an anonymous ticket is returned. If there is no processing rule defined for an authorization data element or the authorization data element is unknown, the TGS MUST process it when an anonymous ticket is returned as follows:

- If the authorization data element may reveal the client's identity, it MUST be removed unless otherwise specified.

- If the authorization data element is intended to restrict the use of the ticket or limit the rights otherwise conveyed in the ticket, it cannot be removed in order to hide the client's identity. In this case, the authentication attempt MUST be rejected, and the KDC MUST return an error message with the code KDC_ERR_POLICY. Note this is applicable to both critical and optional authorization data.

- o If the authorization data element is unknown, the TGS MAY remove it, or transfer it into the returned anonymous ticket, or reject the authentication attempt, based on local policy for that authorization data type unless otherwise specified. If there is no policy defined for a given unknown authorization data type, the authentication MUST be rejected. The error code is KDC_ERR_POLICY when the authentication is rejected.

The AD-INITIAL-VERIFIED-CAS authorization data as defined in [RFC4556] contains the issuer name of the client certificate. If it is undesirable to disclose such information about the client's identity, the AD-INITIAL-VERIFIED-CAS authorization data SHOULD be removed from an anonymous ticket based on local policy of the TGS.

The TGS encodes the name of the previous realm into the transited field according to Section 3.3.3.2 of [RFC4120]. Based on local policy, the TGS MAY omit the previous realm if the cross realm TGT is an anonymous one to hide the authentication path of the client. The unordered set of realms in the transited field, if present, can reveal which realm may potentially be the realm of the client or the realm that issued the anonymous TGT.

If the client is anonymous and the KDC does not have a key to encrypt the reply (this can happen when, for example, the KDC does not support PKINIT [RFC4556]), the KDC MUST return an error message with the code KDC_ERR_NULL_KEY [RFC4120].

If a client requires anonymous communication then the client MUST check to make sure that the ticket in the reply is actually anonymous by checking the presence of the anonymous ticket flag in the flags field of the EncKDCRepPart. This is because KDCs ignore unknown KDC options. A KDC that does not understand the anonymous KDC option will not return an error, but will instead return a normal ticket.

The subsequent client and server communications then proceed as described in [RFC4120].

A server accepting an anonymous service ticket may assume that subsequent requests using the same ticket originate from the same client. Requests with different tickets are likely to originate from different clients.

5. GSS-API Implementation Notes

At the GSS-API [RFC2743] level, the use of an anonymous principal by the initiator/client requires the initiator/client to assert the "anonymous" flag when calling GSS_Init_Sec_Context().

GSS-API does not know or define "anonymous credentials", so the (printable) name of the anonymous principal will rarely be used by or relevant for the initiator/client. The printable name is relevant for the acceptor/server when performing an authorization decision based on the initiator name that is returned from the acceptor side upon the successful security context establishment.

A GSS-API initiator MUST carefully check the resulting context attributes from the initial call to GSS_Init_Sec_Context() when requesting anonymity, because (as in the GSS-API tradition and for backwards compatibility) anonymity is just another optional context attribute. It could be that the mechanism doesn't recognize the attribute at all or that anonymity is not available for some other reasons -- and in that case the initiator must NOT send the initial security context token to the acceptor, because it will likely reveal the initiators identity to the acceptor, something that can rarely be "un-done".

GSS-API defines the name_type GSS_C_NT_ANONYMOUS [RFC2743] to represent the anonymous identity. In addition, Section 2.1.1 of [RFC1964] defines the single string representation of a Kerberos principal name with the name_type GSS_KRB5_NT_PRINCIPAL_NAME. For the anonymous principals, the name component within the exportable name as defined in Section 2.1.3 of [RFC1964] MUST signify the realm name according to Section 2.1.1 of [RFC1964]. Note that in this specification only the client/initiator can be anonymous.

Portable initiators are RECOMMENDED to use default credentials whenever possible, and request anonymity only through the input anon_req_flag [RFC2743] to GSS_Init_Sec_Context().

6. Security Considerations

Since KDCs ignore unknown options, a client requiring anonymous communication needs to make sure that the ticket is actually anonymous. This is because a KDC that that does not understand the anonymous option would not return an anonymous ticket.

By using the mechanism defined in this specification, the client does not reveal its identity to the server but its identity may be revealed to the KDC of the server principal (when the server principal is in a different realm than that of the client), and any KDC on the cross-realm authentication path. The Kerberos client MUST verify the ticket being used is indeed anonymous before communicating with the server, otherwise the client's identity may be revealed unintentionally.

In cases where specific server principals must not have access to the client's identity (for example, an anonymous poll service), the KDC can define server principal specific policy that insure any normal service ticket can NEVER be issued to any of these server principals.

If the KDC that issued an anonymous ticket were to maintain records of the association of identities to an anonymous ticket, then someone obtaining such records could breach the anonymity. Additionally, the implementations of most (for now all) KDC's respond to requests at the time that they are received. Traffic analysis on the connection to the KDC will allow an attacker to match client identities to anonymous tickets issued. Because there are plaintext parts of the tickets that are exposed on the wire, such matching by a third party observer is relatively straightforward.

The client's real identity is not revealed when the client is authenticated as the anonymous principal. Application servers MAY reject the authentication in order to, for example, prevent information disclosure or as part of Denial of Service (DOS) prevention. Application servers MUST avoid accepting anonymous credentials in situations where they must record the client's identity; for example, when there must be an audit trail.

7. Acknowledgements

JK Jaganathan helped editing early revisions of this document.

Clifford Neuman contributed the core notions of this document.

Ken Raeburn reviewed the document and provided suggestions for improvements.

Martin Rex wrote the text for GSS-API considerations.

Nicolas Williams reviewed the GSS-API considerations section and suggested ideas for improvements.

Sam Hartman and Nicolas Williams were great champions of this work.

In addition, the following individuals made significant contributions: Jeffrey Altman, Tom Yu, Chaskiel M Grundman, Love Hornquist Astrand, Jeffrey Hutzelman, and Olga Kornievskaia.

8. IANA Considerations

Section 3 defines the anonymous Kerberos name and the anonymous

Kerberos realm based on [KRBNAM]. The IANA registry for [KRBNAM] need to be updated to add references to this document.

9. Normative References

[KRBNAM] Zhu, L., "Additonal Kerberos Naming Contraints", draft-ietf-krb-wg-naming, work in progress.

[RFC1964] Linn, J., "The Kerberos Version 5 GSS-API Mechanism", RFC 1964, June 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2743] Linn, J., "Generic Security Service Application Program Interface Version 2, Update 1", RFC 2743, January 2000.

[RFC3852] Housley, R., "Cryptographic Message Syntax (CMS)", RFC 3852, July 2004.

[RFC4120] Neuman, C., Yu, T., Hartman, S., and K. Raeburn, "The Kerberos Network Authentication Service (V5)", RFC 4120, July 2005.

[RFC4556] Zhu, L. and B. Tung, "Public Key Cryptography for Initial Authentication in Kerberos (PKINIT)", RFC 4556, June 2006.

Authors' Addresses

Larry Zhu
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
US

Email: lzhu@microsoft.com

Paul Leach
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
US

Email: paulle@microsoft.com

Full Copyright Statement

Intellectual Property