

KERBEROS WORKING GROUP
Internet-Draft
Intended status: Standards Track
Expires: June 12, 2008

Johansson
Stockholm university
December 10, 2007

**An information model for Kerberos version 5
draft-ietf-krb-wg-kdc-model-00**

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

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>.

This Internet-Draft will expire on June 12, 2008.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

This document describes an information model for Kerberos version 5 from the point of view of an administrative service. There is no standard for administrating a kerberos 5 KDC. This document describes the services exposed by an administrative interface to a KDC.

Table of Contents

1.	Requirements notation	3
2.	Introduction	4
3.	How to interpret RFC2119 terms	5
4.	Acknowledgments	6
5.	Information model demarcation	7
6.	Information model specification	8
6.1.	Principal	8
6.1.1.	Principal: Attributes	8
6.1.2.	Principal: Associations	9
6.1.3.	Principal: Remarks	9
6.2.	KeySet	9
6.2.1.	KeySet: Attributes	9
6.2.2.	KeySet: Associations	9
6.2.3.	KeySet: Remarks	10
6.3.	Key	10
6.3.1.	Key: Attributes	10
6.3.2.	Key: Associations	11
6.3.3.	Key: Remarks	11
6.4.	Policy	11
6.4.1.	Policy: Attributes	11
6.4.2.	Password Quality Policy	12
6.4.3.	Password Management Policy	12
6.4.4.	Keying Policy	12
7.	Implementation Scenarios	13
7.1.	LDAP backend to KDC	13
7.2.	LDAP frontend to KDC	13
7.3.	SOAP	13
8.	Security Considerations	14
9.	IANA Considerations	15
10.	Remarks	16
11.	References	17
11.1.	Normative References	17
11.2.	Informative References	17
	Author's Address	18
	Intellectual Property and Copyright Statements	19

1. Requirements notation

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](#)].

2. Introduction

The Kerberos version 5 authentication service described in [[RFC4120](#)] describes how a Key Distribution Service (KDC) provides authentication to clients. The standard does not stipulate how a KDC is managed and several "kadmin" servers have evolved. This document describes the services required to administrate a KDC and the underlying information model assumed by a kadmin-type service.

The information model is written in terms of "attributes" and "services" or "interfaces" but the use of these particular words MUST NOT be taken to imply any particular modeling paradigm so that neither an object oriented model or an LDAP schema is intended. The author has attempted to describe in natural language the intended semantics and syntax of the components of the model. An LDAP schema (for instance) based on this model will be more precise in the expression of the syntax while preserving the semantics of this model.

Implementations of this document MAY decide to change the names used (eg principalName). If so an implementation MUST provide a name to name mapping to this document.

3. How to interpret [RFC2119](#) terms

This document describes an information model for kerberos 5 but does not directly describe any mapping onto a particular schema- or modelling language. Hence an implementation of this model consists of a mapping to such a language - eg an LDAP or SQL schema. The precise interpretation of terms from [[RFC2119](#)] therefore require some extra explanation. The terms MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT mean that an implementation MUST provide a feature but does not mean that this feature MUST be REQUIRED by the implementation - eg an attribute is available in an LDAP schema but marked as OPTIONAL. If a feature must be implemented and REQUIRED this is made explicit in this model. The term MAY, OPTIONAL and RECOMMENDED means that an implementation MAY need to REQUIRE the feature due to the particular nature of the schema/modelling language. In some cases this is expressly forbidden by this model (feature X MUST NOT be REQUIRED by an implementation).

Note that any implementation of this model SHOULD be published as an RFC.

4. Acknowledgments

Love Hoernquist-Aestrand <lha@it.su.se> for important contributions.

5. Information model demarcation

The information model specified in the next chapter describes objects, properties of those objects and relations between those objects. These elements comprise an abstract view of the data represented in a KDC. It is important to understand that the information model is not a schema. In particular the way objects are compared for equality beyond that which is implied by the specification of a syntax is not part of this specification. Nor is ordering specified between elements of a particular syntax.

Further work on Kerberos will undoubtedly prompt updates to this information model to reflect changes in the functions performed by the KDC. Such extensions to the information model **MUST** always use a normative reference to the relevant RFCs detailing the change in KDC function.

6. Information model specification

6.1. Principal

The fundamental entity stored in a KDC is the principal. The principal is associated to keys and generalizes the "user" concept. The principal **MUST** be implemented in full and **MUST NOT** be optional in an implementation

6.1.1. Principal: Attributes

6.1.1.1. principalName

The principalName **MUST** uniquely identify the principal within the administrative context of the KDC. The type of the principalName is not described in this document. It is a unique identifier and can be viewed as an opaque byte string which can be compared for equality. The attribute **SHOULD** be single valued. If an implementation supports multiple values it **MUST** treat one of the values as special and allow it to be fetched as if it was a single value.

6.1.1.2. principalNotUsedBefore

The principal may not be used before this date. The syntax of the attribute **MUST** be semantically equivalent with the standard ISO date format. The attribute **MUST** be single valued.

6.1.1.3. principalNotUsedAfter

The principal may not be used after this date. The syntax of the attribute **MUST** be semantically equivalent with the standard ISO date format. The attribute **MUST** be single valued.

6.1.1.4. principalIsDisabled

A boolean attribute used to (temporarily) disable a principal. The attribute **MUST** default to false.

6.1.1.5. principalAliases

This multivalued attribute contains an unordered set of aliases for the principal. Each alias **SHOULD** be unique within the administrative domain represented by the KDC. The syntax of an alias is an opaque identifier which can be compared for equality.

6.1.2. Principal: Associations

Each principal MAY be associated with 1 or more KeySet and MAY be associated with 1 or more Policies. The KeySet is represented as an object in this model since it has attributes associated with it (the key version number). In typical situations the principal is associated with exactly 1 KeySet but implementations MUST NOT assume this case, i.e an implementation of this standard (e.g an LDAP schema) MUST be able to handle the general case of multiple KeySet associated with each principal.

6.1.3. Principal: Remarks

Traditionally a principal consists of a local-part and a realm denoted in string form by local-part@REALM. The realm concept is used to provide administrative boundaries and together with cross-realm authentication provides scalability to Kerberos 5. However the realm is not central to an administrative information model. For instance the initialization or creation of a realm is equivalent to creating a specific set of principals (krbtgt@REALM, etc) which is covered by the model and services described in this document. A realm is typically associated with policy covering (for instance) keying and password management. The management of such policy and their association to realms is beyond the scope of this document.

6.2. KeySet

A KeySet is a set of keys associated with exactly one principal. This object and its associations MUST NOT be REQUIRED by an implementation. It is expected that most implementations of this standard will use the set/change password protocol for all aspects of key management [[I-D.ietf-krb-wg-kerberos-set-passwd](#)]. This information model only includes these objects for the sake of completeness.

6.2.1. KeySet: Attributes

6.2.1.1. keySetVersionNumber

This is traditionally called the key version number (kvno). This is a single valued attribute containing a positive integer.

6.2.2. KeySet: Associations

To each KeySet MUST be associated a set of 1 or more Keys.

6.2.3. KeySet: Remarks

The reason for separating the KeySet from the Principal is security. The security of Kerberos 5 depends absolutely on the security of the keys stored in the KDC. The KeySet type is provided to make this clear and to make separation of keys from other parts of the model clear.

Implementations of this standard (eg an LDAP schema) MUST make a clear separation between the representation of KeySet from other information objects.

6.3. Key

Implementations of this model MUST NOT REQUIRE keys to be represented.

6.3.1. Key: Attributes

6.3.1.1. keyEncryptionType

The enctype SHOULD be represented as an enumeration of the encetypes supported by the KDC.

6.3.1.2. keyValue

The binary representation of the key data. This MUST be a single valued octet string.

6.3.1.3. keySaltValue

The binary representation of the key salt. This MUST be a single valued octet string.

6.3.1.4. keyStringToKeyParameter

This MUST be a single valued octet string representing an opaque parameter associated with the enctype.

6.3.1.5. keyNotUsedAfter

This key MUST NOT be used after this date. The syntax of the attribute MUST be semantically equivalent with the standard ISO date format. This MUST be a single-valued attribute.

6.3.1.6. keyNotUsedBefore

This key MUST NOT be used before this date. The syntax of the attribute MUST be semantically equivalent with the standard ISO date format. This MUST be a single-valued attribute.

6.3.1.7. keyIsDisabled

This is a boolean attribute which must be set to false by default. If this attribute is true the key MUST NOT be used. This is used to temporarily disable a key.

6.3.2. Key: Associations

None

6.3.3. Key: Remarks

The security of the keys is an absolute requirement for the operation of Kerberos 5. If keys are implemented adequate protection from unauthorized modification and disclosure MUST be available and REQUIRED by the implementation.

6.4. Policy

Implementations SHOULD implement policy but MAY allow them to be OPTIONAL.

6.4.1. Policy: Attributes

6.4.1.1. policyIdentifier

The policyIdentifier MUST be unique within the local administrative context and MUST be globally unique. Possible types of identifiers include:

An Object Identifier (OID)

A URN

A UUID

6.4.1.2. policyIsMandatory

This boolean attribute indicates that the KDC MUST be able to correctly interpret and apply this policy for the key to be used.

6.4.1.3. policyContent

This is an optional opaque binary value used to store a representation of the policy. In general a policy cannot be fully expressed using attribute-value pairs. The policyContent is OPTIONAL in the sense that an implementation MAY use it to store an opaque value for those policy-types which are not directly representable in that implementation.

6.4.2. Password Quality Policy

6.4.2.1. Password Quality Policy: Attributes

Password quality policy controls the requirements placed by the KDC on new passwords. TODO: update with information from Nico

6.4.3. Password Management Policy

6.4.3.1. Password Management Policy: Attributes

Password management policy controls how passwords are changed. TODO: update with information from Nico and Ludovic

6.4.4. Keying Policy

6.4.4.1. Keying Policy: Attributes

A keying policy specifies the association of enttypes with new principals, i.e when a principal is created one of the possibly many applicable keying policies determine the set of keys to associate with the principal. In general the expression of a keying policy may require a Turing-complete language.

7. Implementation Scenarios

There are several ways to implement an administrative service for Kerberos 5 based on this information model. In this section we list a few of them.

7.1. LDAP backend to KDC

Given an LDAP schema implementation of this information model it would be possible to build an administrative service by backending the KDC to a directory server where principals and keys are stored. Using the security mechanisms available on the directory server keys are protected from access by anyone apart from the KDC. Administration of the principals, policy and other non-key data is done through the directory server while the keys are modified using the set/change password protocol [[I-D.ietf-krb-wg-kerberos-set-passwd](#)].

7.2. LDAP frontend to KDC

An alternative way to provide a directory interface to the KDC is to implement an LDAP-frontend to the KDC which exposes all non-key objects as entries and attributes. As in the example above all keys are modified using the set/change password protocol [[I-D.ietf-krb-wg-kerberos-set-passwd](#)]. In this scenario the implementation would typically not use a traditional LDAP implementation but treat LDAP as an access-protocol to data in the native KDC database.

7.3. SOAP

Given an XML schema implementation of this information model it would be possible to build a SOAP-interface to the KDC. This demonstrates the value of creating an abstract information model which is mappable to multiple schema representations.

8. Security Considerations

This document describes an abstract information model for Kerberos 5. The Kerberos 5 protocol depends on the security of the keys stored in the KDC. The model described here assumes that keys MUST NOT be transported in the clear over the network and furthermore that keys are treated as write-only attributes that SHALL only be modified (using the administrative interface) by the change-password protocol [[I-D.ietf-krb-wg-kerberos-set-passwd](#)].

Exposing the object model of a KDC typically implies that objects can be modified and/or deleted. In a KDC not all principals are created equal, so that for instance deleting `krbtgt/EXAMPLE.COM@EXAMPLE.COM` effectively disables the `EXAMPLE.COM` realm. Hence access control is paramount to the security of any implementation. This document does not (at the time of writing - leifj) mandate access control. This only implies that access control is beyond the scope of the standard information model, i.e that access control may not be accessible via any protocol based on this model. If access control objects is exposed via an extension to this model the presence of access control may in itself provide points of attack by giving away information about principals with elevated rights etc. etc.

9. IANA Considerations

None

10. Remarks

A few notes and TODOs:

Do we want to model access control? I have received a few notes on that from Love. It will affect both the model and the security considerations but It may be relevant. The catch is that most implementations (SOAP, LDAP, etc) will have acl mechanisms separate from the data which makes modeling acls difficult. Perhaps there are certain aspects of access control which can be modeled with relative ease - for instance the ability to make an object immutable.

Explanatory text on a few of the basic attributes that doesn't just repeat the section title.

Expand on the password policy types. Is the subdivision into quality and management policies valid?

11. References

11.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC4120] Neuman, C., Yu, T., Hartman, S., and K. Raeburn, "The Kerberos Network Authentication Service (V5)", [RFC 4120](#), July 2005.

11.2. Informative References

- [I-D.ietf-krb-wg-kerberos-set-passwd]
Williams, N., "Kerberos Set/Change Key/Password Protocol Version 2", [draft-ietf-krb-wg-kerberos-set-passwd-07](#) (work in progress), September 2007.

Author's Address

Leif Johansson
Stockholm university
Sektionen foer IT och Media
Stockholm SE-106 91

Email: leifj@it.su.se

URI: <http://people.su.se/~leifj/>

Full Copyright Statement

Copyright (C) The IETF Trust (2007).

This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).

