Kerberos Working Group
Internet Draft

Matt Crawford
Fermilab

21 October 2006

Passwordless Initial Authentication to Kerberos by Hardware Preauthentication

<draft-ietf-krb-wg-hw-auth-04.txt>

Status of this Memo

Abstract

This document specifies an extension to the Kerberos protocol for performing initial authentication of a user without using that user's long-lived password. Any "hardware preauthentication" method may be employed instead of the password, and the key of another principal must be nominated to encrypt the returned credential.

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 [KWORD].

1. Motivation

Many sites using Kerberos for authentication have users who are often, or even always, away from the site. Sometimes these users may need to connect to their site while they have no immediate access to a trustworthy computer with Kerberos software or any other trusted secure remote-access mechanism. Requiring hardware preauthentication in addition to a password for all such users is an incomplete solution because an eavesdropper with access to both the remote users' path to the host in the site and that host's path to the KDC can still steal the user's credential.

This document specifies a method by which a Kerberos application server can request that a KDC authenticate a user using a hardware preauthentication method and use a key held by the server in the decryption of the KDC's reply, in place of the user's password.

2. Definitions

The following terms used here are defined in [KRB5] and [KRB5bis]:

> KDC_ERR_PREAUTH_FAILED, KDC_ERR_PREAUTH_REQUIRED, KRB_AS_REQ, KRB_ERROR, PrincipalName, e-data, enc-part, error-code, kdc-options, padata-type, padata-value.

These terms are defined in [KRB5bis]:

> PA-SAM-CHALLENGE, PA-SAM-CHALLENGE2, PA-SAM-RESPONSE, PA-SAM-RESPONSE2.

The term "service" denotes some Kerberos service which normally requires a client/server authentication exchange [KRB5] for access and which is capable of both communicating with the KDC's Authentication Service and interacting with the user to the extent required to carry out a single-use authentication mechanism (SAM). It must have access to some principal's long-lived key. Telnet and FTP services are examples.

The Kerberos Authentication Service will be denoted by "AS" to avoid confusion with the service.

3. Method

This mechanism is intended to be employed when a user connects to a service which normally allows only Kerberos-authenticated access. When the service determines that the user will not authenticate (for example, it receives a telnet "WONT AUTHENTICATION" command [TELAUTH], or an FTP "USER" command without a preceding "AUTH" command [FTPSEC]), it may accept a user principal name and attempt to perform passwordless hardware authentication in the following manner.

3.1. Initial AS Request and reply

The service, on behalf of the user, prepares a KRB_AS_REQ [KRB5] message with the flag OPT-HARDWARE-AUTH set in the kdc-options field, in addition to any other desired options and lifetimes. The service sends this message to a KDC. If the KDC's policy permits this form of authentication for the user named in the request, and the request is acceptable in all other respects, the KDC determines what hardware preauthentication methods are available for the user principal and constructs a KRB_ERROR message with the error-code set to KDC_ERR_PREAUTH_REQUIRED. The e-data field of this KRB_ERROR message contains a sequence of PA-DATA which includes an element with padata-type equal to PA-ALT-PRINC and an empty padata-value. In addition to that are any elements needed for hardware preauthentication of the user. Typically this will include an element with padata-type PA-SAM-CHALLENGE or PA-SAM-CHALLENGE2 and padata-value appropriate to the authentication method.

3.2. Second AS Request

The service, upon receiving the KRB_ERROR message from the KDC, must process the PA-ALT-PRINC element by selecting a principal whose long-lived key it has access to, and which is in the same realm as the client. This principal will be referred to as the alternate principal. It processes the PA-SAM-CHALLENGE normally, except that whenever the user's long-lived (password-derived) encryption key is called for, it uses the alternate principal's key instead.

The service constructs a second KRB_AS_REQ, again with the OPT-HARDWARE-AUTH flag set in the kdc-options field, and this time with a padata field which includes at least these two PA-DATA items, in this order:

> One with padata-type equal to PA-ALT-PRINC and as padata-value the encoded PrincipalName of the alternate principal,

One with padata-type appropriate for hardware token-based preauthentication, such as PA-SAM-RESPONSE or PA-SAM-RESPONSE2, and padata-value constructed as it would be for normal hardware preauthentication, but with the alternate principal's key used in place of the user's key.

Other PA-DATA may be present before, between or after these items.

The service sends this second KRB_AS_REQ to a KDC.

3.3. Final AS Reply

The KDC begins processing the AS request normally. When the PA-ALT-PRINC field is encountered, the KDC does the following:

First, if this use of the alternate principal named in the request is against local policy, or if the alternate principal does not exist in the database, a KRB_ERROR message with error-code KDC_ERR_PREAUTH_FAILED is returned and processing ends.

Then, the alternate principal's key is fetched from the database and held for use in subsequent processing. It will be needed to process the PA-SAM-RESPONSE, PA-SAM-RESPONSE2, or similar preauthentication data, and to encrypt the enc-part of the KRB_AS_REP if authentication is successful.

The remainder of the AS request processing is normal, with the noted substitution of the alternate principal's key for the user's.

The service, upon receiving a KRB_AS_REP, uses the alternate principal's key to decrypt the enc-part, saves the user's credential and takes appropriate measures to ensure that the KRB_AS_REP came from a legitimate KDC and not an imposter.

4. IANA Considerations

No new naming or numbering spaces are created by this specification. Two values from existing spaces are defined in [KRB5bis] for the mechanism of this document:

The flag OPT-HARDWARE-AUTH is bit 11 in the kdc-options field of a KDC-REQ-BODY.

The preauthentication type PA-ALT-PRINC is denoted by padata-type 24.

5. Security Considerations

There are no means provided here for protecting the traffic between the user and the service, so it may be susceptible to eavesdropping, hijacking and alteration. This authentication mechanism is not intended to be used as an alternative to the Kerberos client/server authentication exchange, but as an improvement over making an unprotected connection with a Kerberos password alone, or a password plus a single-use authenticator.

The alternate principal's key MUST be involved in construction of the PA-SAM-RESPONSE (or PA-SAM-RESPONSE2) padata-value, to prevent an adversary constructing a KRB_AS_REQ using that data but a different alternate principal. In practice, this means that the response data alone must not determine the encryption key for the padata-value.

A service impersonator can obtain a presumably-valid SAM response from the user which may (or may not) be usable for impersonating the user at a later time. And of course in the case of successful authentication the service obtains access to the user's credentials. As always, if the service host is compromised, so are the credentials; but, with this mechanism, at least the service host never has access to the user's password.

A service host which accepts a Kerberos password for access typically protects itself against an impostor KDC by using the received ticket-granting credential to get a ticket for a service for which it has the key. This step may be unnecessary when the service host has already successfully used such a key to decrypt the ticket-granting credential itself.

Use of this authentication method employs the service's long-term key, providing more ciphertext in that key to an eavesdropper. This key is generally of better quality than a password-derived key and any remaining concerns about the strength of the KRB_AS_REP are better addressed by a general mechanism applicable to all AS exchanges.

6. Acknowledgments

The first implementation of this extension grew from a beginning by Ken Hornstein, which in turn was built on code released by the MIT Kerberos Team.

7. References

[FTPSEC] Horowitz, M. and S. Lunt, "FTP Security Extensions", RFC 2228.

[KRB5]   Kohl, J., and C. Neuman, "The Kerberos Network Authentication Service (V5)", RFC 1510.

[KRB5bis] Neuman, C., T. Yu, S. Hartman, and K. Raeburn, "The Kerberos Network Authentication Service (V5)", RFC 4120.

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

[TELAUTH] Ts'o, T. and J. Altman, "Telnet Authentication Option", RFC 2941.

8. Author's Address

Matt Crawford
Fermilab MS 368
PO Box 500
Batavia, IL 60510
USA

Phone: +1 630 840-3461
EMail: crawdad@fnal.gov

Disclaimer of Validity

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.