Network Working Group
Internet-Draft: draft-altman-rfc2942bis-03
Obsoletes: 2942

T. Ts'o
J. Altman
Columbia University
December 2000

Telnet Authentication: Kerberos Version 5

Status of this Memo

Abstract

This document describes how Kerberos Version 5 [1] is used with the telnet protocol. It describes an telnet authentication suboption to be used with the telnet authentication option [2]. This mechanism can also used to provide keying material to provide data confidentiality services in conjunction with the telnet encryption option [3].

This document updates a previous specification of the Telnet Authentication Kerberos 5 method, RFC 2942 [4], to allow Kerberos 5 Telnet authentication to be used in conjunction with the START_TLS option [5].

0. Changes since RFC 2942

. Consolidates and expands Security Considerations section

. Describes integration with START_TLS option

1. Command Names and Codes

Authentication Types

KERBEROS_V5	2

Sub-option Commands

AUTH	0
REJECT	1
ACCEPT	2
RESPONSE	3
FORWARD	4
FORWARD_ACCEPT	5
FORWARD_REJECT	6
TLS_VERIFIER	7

2. Command Meanings

IAC SB AUTHENTICATION IS <authentication-type-pair> AUTH
<KRB_AP_REQ message> IAC SE

This is used to pass the Kerberos V5 [1] KRB_AP_REQ message to the remote side of the connection. The first octet of the <authentication-type-pair> value is KERBEROS_V5, to indicate that Version 5 of Kerberos is being used.

The Kerberos V5 authenticator in the KRB_AP_REQ message must contain a Kerberos V5 checksum of the two-byte authentication-type-pair. If the ENCRYPT_START_TLS bit mask is in use, the checksum must concatenate the TLS Client Finished Message and TLS Server Finished Message to the authentication-type-pair. This checksum must be verified by the server to assure that the authentication type pair was correctly negotiated.

The Kerberos V5 authenticator must also include the optional subkey field, which shall be filled in with a randomly chosen key. This key shall be used for encryption purposes if encryption is negotiated, and shall be used as the negotiated session key (i.e., used as keyid 0) for the purposes of the telnet encryption option; if the subkey is not filled in, then the ticket session key will be used instead.

If data confidentiality services is desired the ENCRYPT_US-ING_TELOPT flag must be set in the authentication-type-pair as specified in [2].

IAC SB AUTHENTICATION REPLY <authentication-type-pair> ACCEPT IAC SE

This command indicates that the authentication was successful.

If the AUTH_HOW_MUTUAL bit is set in the second octet of the

authentication-type-pair, the RESPONSE command must be sent before the ACCEPT command is sent.

IAC SB AUTHENTICATION REPLY <authentication-type-pair> REJECT <optional reason for rejection> IAC SE

This command indicates that the authentication was not successful, and if there is any more data in the sub-option, it is an ASCII text message of the reason for the rejection.

IAC SB AUTHENTICATION REPLY <authentication-type-pair> RESPONSE <KRB_AP_REP message> IAC SE

This command is used to perform mutual authentication. It is only used when the AUTH_HOW_MUTUAL bit is set in the second octet of the authentication-type-pair. After an AUTH command is verified, a RESPONSE command is sent which contains a Kerberos V5 KRB_AP_REP message to perform the mutual authentication.

IAC SB AUTHENTICATION REPLY <authentication-type-pair> TLS_VERIFY MK_SAFE(<server TLS finished><client TLS finished>) IAC SE

This command is used to verify that there is no man in the middle attack when a Kerberos 5 authentication is performed over a TLS protected session. It is only used with AUTH_ENCRYPT_START_TLS is set. If the client and server's finished messages cannot be verified, the connection MUST be terminated. If this message is not received prior to the receipt of the ACCEPT message, the connection must be terminated.

IAC SB AUTHENTICATION <authentication-type-pair> FORWARD <KRB_CRED message> IAC SE

This command is used to forward kerberos credentials for use by the remote session. The credentials are passed as a Kerberos V5 KRB_CRED message which includes, among other things, the forwarded Kerberos ticket and a session key associated with the ticket. Part of the KRB_CRED message is encrypted in the key previously exchanged for the telnet session by the AUTH suboption.

IAC SB AUTHENTICATION <authentication-type-pair> FORWARD_ACCEPT IAC SE

This command indicates that the credential forwarding was successful.

IAC SB AUTHENTICATION <authentication-type-pair> FORWARD_REJECT <optional reason for rejection> IAC SE

This command indicates that the credential forwarding was not successful, and if there is any more data in the suboption, it is an ASCII text message of the reason for the rejection.

3. Implementation Rules

If the second octet of the authentication-type-pair has the AUTH_WHO bit set to AUTH_CLIENT_TO_SERVER, then the client sends the initial AUTH command, and the server responds with either ACCEPT or REJECT. In addition, if the AUTH_HOW bit is set to AUTH_HOW_MUTUAL, the server will send a RESPONSE before it sends the ACCEPT.

If the second octet of the authentication-type-pair has the AUTH_WHO bit set to AUTH_SERVER_TO_CLIENT, then the server sends the initial AUTH command, and the client responds with either ACCEPT or REJECT. In addition, if the AUTH_HOW bit is set to AUTH_HOW_MUTUAL, the client will send a RESPONSE before it sends the ACCEPT.

The Kerberos principal used by the server will generally be of the form "host/<hostname>@realm". That is, the first component of the Kerberos principal is "host"; the second component is the fully qualified lower-case hostname of the server; and the realm is the Kerberos realm to which the server belongs.

Any Telnet IAC characters that occur in the KRB_AP_REQ or KRB_AP_REP messages, the KRB_CRED structure, or the optional rejection text string must be doubled as specified in [4]. Otherwise the following byte might be mis-interpreted as a Telnet command.

4. Examples

User "joe" may wish to log in as user "pete" on machine "foo". If "pete" has set things up on "foo" to allow "joe" access to his account, then the client would send IAC SB AUTHENTICATION NAME "pete" IAC SE IAC SB AUTHENTICATION IS KERBEROS_V5 AUTH <KRB_AP_REQ_MESSAGE> IAC SE

The server would then authenticate the user as "joe" from the KRB_AP_REQ_MESSAGE, and if the KRB_AP_REQ_MESSAGE was accepted by Kerberos, and if "pete" has allowed "joe" to use his account, the server would then continue the authentication sequence by sending a RESPONSE (to do mutual authentication, if it was requested) followed by the ACCEPT.

If forwarding has been requested, the client then sends IAC SB AUTHENTICATION IS KERBEROS_V5 CLIENT|MUTUAL FORWARD <KRB_CRED structure with credentials to be forwarded> IAC SE. If the server succeeds in reading the forwarded credentials, the server sends FORWARD_ACCEPT else, a FORWARD_REJECT is sent back.

```
    Client                          Server
                                    IAC DO AUTHENTICATION
    IAC WILL AUTHENTICATION

    [ The server is now free to request authentication information.  ]
```

```
                                IAC SB AUTHENTICATION SEND
                                KERBEROS_V5 CLIENT|MUTUAL
                                KERBEROS_V5 CLIENT|ONE_WAY IAC
                                SE
```

[The server has requested mutual Version 5 Kerberos authentication. If mutual authentication is not supported, then the server is willing to do one-way authentication.

The client will now respond with the name of the user that it wants to log in as, and the Kerberos ticket.]

```
IAC SB AUTHENTICATION NAME
"pete" IAC SE
IAC SB AUTHENTICATION IS
KERBEROS_V5 CLIENT|MUTUAL AUTH
<KRB_AP_REQ message> IAC SE
```

[Since mutual authentication is desired, the server sends across a RESPONSE to prove that it really is the right server.]

```
                                IAC SB AUTHENTICATION REPLY
                                KERBEROS_V5 CLIENT|MUTUAL
                                RESPONSE <KRB_AP_REP message>
                                IAC SE
```

[The server responds with an ACCEPT command to state that the authentication was successful.]

```
                                IAC SB AUTHENTICATION REPLY
                                KERBEROS_V5 CLIENT|MUTUAL ACCEPT
                                IAC SE
```

[If so requested, the client now sends the FORWARD command to forward credentials to the remote site.]

```
IAC SB AUTHENTICATION IS KER-
BEROS_V5 CLIENT|MUTUAL
FORWARD <KRB_CRED message> IAC
SE
```

[The server responds with a FORWARD_ACCEPT command to state that the credential forwarding was successful.]

```
                                IAC SB AUTHENTICATION REPLY
                                KERBEROS_V5 CLIENT|MUTUAL
                                FORWARD_ACCEPT IAC SE
```

5. Security Considerations

As an implementation of the TELNET AUTH option [2] all of the

Security Considerations from that RFC MUST be considered applicable to this sub-option.

The selection of the random session key in the Kerberos V5 authenticator is critical, since this key will be used for encrypting the telnet data stream if encryption is enabled. It is strongly advised that the random key selection be done using cryptographic techniques that involve the Kerberos ticket's session key. For example, using the current time, encrypting it with the ticket session key, and then correcting for key parity is a strong way to generate a subsession key, since the ticket session key is assumed to be never disclosed to an attacker.

Care should be taken before forwarding a user's Kerberos credentials to the remote server. If the remote server is not trustworthy, this could result in the user's credentials being compromised. Hence, the user interface should not forward credentials by default; it would be far safer to either require the user to explicitly request credentials forwarding for each connection, or to have a trusted list of hosts for which credentials forwarding is enabled, but to not enable credentials forwarding by default for all machines.

This mechanism does not include all of the telnet authentication negotiation exchanges in the integrity checksum as recommended in [2]. This means that the selection of this option is vulnerable to downgrade attacks when multiple authentication type pairs are offered by the server.

The IAC SB AUTHENTICATION NAME name IAC SE message is unprotected in the AUTH option and it is not verified by the AUTH KRB5 suboption. The name MUST be verified by a secure method after authentication completes before it is used to access authorization information or perform a login. One method of verification is for the server to request the client to transmit the desired name using the TELNET NEW-ENVIRONMENT option [6]. This can be accomplished by sending the message after authentication and encryption activation completes:

```
  IAC SB NEW-ENVIRONMENT SEND VAR USER IAC SE
```

The client's response should include the same name that was transmitted as part of the IAC SB AUTHENTICATION NAME name IAC SE message.

6. IANA Considerations

The authentication type KERBEROS_V5 and its associated suboption values are registered with IANA. Any suboption values used to extend the protocol as described in this document must be registered with IANA before use. IANA is instructed not to issue new suboption values without submission of documentation of their use.

7. Acknowledgments

This document was originally written by Dave Borman of Cray Research, Inc. Theodore Ts'o of MIT revised it to reflect the latest implementation experience. Cliff Neuman and Prasad Upasani of USC's Information Sciences Institute developed the credential forwarding support.

In addition, the contributions of the Telnet Working Group are also gratefully acknowledged.

8. References

[1] Kohl, J. and B. Neuman, "The Kerberos Network Authentication System (V5)", RFC 1510, September 1993.

[2] Ts'o, T. and J. Altman, "Telnet Authentication Option", draft-altman-rfc2941bis-??.txt.

[3] Ts'o, T., "Telnet Data Encryption Option", RFC 2946, September 2000.

[4] Postel, J. and J. Reynolds, "Telnet Option Specifications", STD 8, RFC 855, May 1983.

[5] Altman, J. and Boe, M., "TLS-based Telnet Security", draft-ietf-tn3270e-telnet-tls-??.txt.

[6] "Telnet (New) Environment Option", RFC 1572

9. Editor's Address

Jeffrey Altman
Columbia University
Watson Hall Room 716
612 West 115th Street
New York NY 10025

Phone: +1 (212) 854-1344
EMail: jaltman@columbia.edu

Theodore Ts'o
43 Pleasant St.
Medford, MA 02155

Phone: (781) 391-3464
EMail: tytso@mit.edu

Mailing List: telnet-wg@BSDI.COM

10. Full Copyright Statement

Acknowledgement

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