Network Working Group
Internet-Draft: draft-altman-telnet-rfc2944bis-02.txt
Obsoletes: 2944
Expires: 16 July 2007
December 15, 2006

Jeffrey Altman

Telnet Authentication: SRP

Status of this memo

This Internet-Draft will expire on 16 July 2007.

Copyright Notice

Abstract

This document specifies an authentication scheme for the Telnet protocol [RFC 854] under the framework described in [RFC YYYY], using the Secure Remote Password Protocol (SRP) authentication mechanism. The specific mechanism, SRP-SHA1, is described in [RFC2945].

This document updates a previous specification of the Telnet Authentication SRP method, RFC 2944, to allow SRP Telnet authentication to be used in conjunction with the Telnet START_TLS option [RFC YYYY].

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.

1. Introduction

This document specifies an authentication scheme for the Telnet protocol [RFC 854] under the framework described in [RFC YYYY], using the Secure Remote Password Protocol (SRP) authentication mechanism. The specific mechanism, SRP-SHA1, is described in [RFC2945].

This document updates a previous specification of the Telnet Authentication SRP method, RFC 2944, to allow SRP Telnet authentication to be used in conjunction with the Telnet START_TLS option [RFC YYYY].

2. Command Names and Codes

Authentication Types

SRP	5

Suboption Commands

AUTH	0
REJECT	1
ACCEPT	2
CHALLENGE	3
RESPONSE	4
EXP	8
PARAMS	9

3. Command Meanings

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

This command indicates that the client has supplied the username and is ready to receive that user's field parameters. There is no authentication information to be sent to the remote side of the connection yet. This should only be sent after the IAC SB AUTHENTICATION NAME command has been issued. If the modifier byte (second byte of the authentication-type-pair) has any bits other than AUTH_WHO_MASK or AUTH_HOW_MASK set, both bytes are included in the session key hash described later. This ensures that the authentication type pair was correctly negotiated, while maintaining backward-compatibility with existing software.

If the ENCRYPT_START_TLS bit mask is in use, the checksum must concatenate the TLS Client Finished Message and TLS Server Finished Message [RFC 4346] to the authentication-type-pair in the session key hash. This ensure that the TLS negotiation was not susceptible to a man in the middle attack if the TLS credentials are not verified.

IAC SB AUTHENTICATION REPLY <authentication-type-pair> PARAMS <values of modulus, generator, and salt> IAC SE

This command is used to pass the three parameter values used in the exponentiation to the client. These values are often called n, g, and s.

IAC SB AUTHENTICATION IS <authentication-type-pair> EXP <client's exponential residue> IAC SE

This command is used to pass the client's exponential residue, otherwise known as A, computed against the parameters exchanged earlier.

IAC SB AUTHENTICATION REPLY <authentication-type-pair> CHALLENGE <server's exponential residue> IAC SE

This command is used to pass the server's exponential residue, computed against the same parameters. This quantity is actually the sum of two residues, i.e. g^x + g^b. For details see [SRP] and [RFC2945].

IAC SB AUTHENTICATION IS <authentication-type-pair> RESPONSE <response from client> IAC SE

This command gives the server proof of the client's authenticity with a 160-bit (20 byte) response.

IAC SB AUTHENTICATION REPLY <authentication-type-pair> ACCEPT <server's response> IAC SE

This command indicates that the authentication was successful. The server will construct its own proof of authenticity and include it as sub-option data.

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.

For the PARAMS command, since three pieces of data are being transmitted, each parameter is preceded by a 16-bit (two byte) length specifier in network byte order. The EXP commands do not have a count in front of the data because there is only one piece of data in that suboption. The CHALLENGE, RESPONSE, and ACCEPT data also do not have a count because they are all fixed in size.

4. Implementation Rules

Currently, only AUTH_CLIENT_TO_SERVER mode is supported. Although the SRP protocol effectively performs implicit mutual authentication as a result of the two-way proofs, only the AUTH_HOW_ONE_WAY authentication mode is currently defined. The AUTH_HOW_MUTUAL setting is being reserved for an explicit mutual-authentication variant of the SRP protocol to be defined in future specifications.

All large number data sent in the arguments of the PARAMS and EXP commands must be in network byte order, i.e. most significant byte first. No padding is used.

The SRP-SHA1 mechanism, as described in [RFC2945] generates a 40-byte session key, which allows implementations to use different keys for incoming and outgoing traffic, increasing the security of the encrypted session. It is recommended that the Telnet ENCRYPT method, if it is used, be able to take advantage of the longer session keys.

5. Examples

User "tjw" may wish to log in on machine "foo". The client would send IAC SB AUTHENTICATION NAME "tjw" IAC SE IAC SB AUTHENTICATION IS SRP AUTH IAC SE. The server would look up the field and salt parameters for "tjw" from its password file and send them back to the client. Client and server would then exchange exponential residues and calculate their session keys (after the client prompted "tjw" for his password). Then, the client would send the server its proof that it knows the session key. The server would either send back an ACCEPT or a REJECT. If the server accepts authentication, it also sends its own proof that it knows the session key to the client.

```
Client                           Server
                                 IAC DO AUTHENTICATION
IAC WILL AUTHENTICATION

[ The server is now free to request authentication information.]
                                 IAC SB AUTHENTICATION SEND
                                 SRP CLIENT|ONE_WAY|
                                 ENCRYPT_USING_TELOPT
                                 SRP CLIENT|ONE_WAY
                                 IAC SE

[ The server has requested SRP authentication.  It has indicated
  a preference for ENCRYPT_USING_TELOPT, which requires the
  Telnet ENCRYPT option to be negotiated once authentication
  succeeds.  If the client does not support this, the server
  is willing to fall back to an encryption-optional mode.
  The client will now respond with the name of the
  user that it wants to log in as. ]

IAC SB AUTHENTICATION NAME
"tjw" IAC SE
IAC SB AUTHENTICATION IS
SRP CLIENT|ONE_WAY|ENCRYPT_USING_TELOPT AUTH
IAC SE

[ The server looks up the appropriate information for "tjw" and
  sends back the parameters in a PARAMS command.  The parameters
  consist of the values N, g, and s, each preceded with a two-
  byte size parameter. ]

                                 IAC SB AUTHENTICATION REPLY
                                 SRP CLIENT|ONE_WAY|
                                 ENCRYPT_USING_TELOPT PARAMS
                                 ss ss nn nn nn nn ...
                                 ss ss gg gg gg gg ...
                                 ss ss tt tt tt tt ...
                                 IAC SE

[ Both sides send their exponential residues.  The client
  sends its value A and the server sends its value B.  In SRP,
  the CHALLENGE message may be computed but not sent before
  the EXP command.  ]

IAC SB AUTHENTICATION IS
SRP CLIENT|ONE_WAY|ENCRYPT_USING_TELOPT EXP
aa aa aa aa aa aa aa aa ...
IAC SE
                                 IAC SB AUTHENTICATION REPLY
                                 SRP CLIENT|ONE_WAY|
                                 ENCRYPT_USING_TELOPT CHALLENGE
                                 bb bb bb bb bb bb bb bb ...
                                 IAC SE

[ The client sends its response to the server.  This is the
  message M in the SRP protocol, which proves possession of
  the session key by the client.

  Since ENCRYPT_USING_TELOPT is specified, the two octets
  of the authentication-type-pair are appended to the
  session key K before the hash for M is computed.  If
  the client and server had agreed upon a mode without
  the encryption flag set, nothing would be appended to K.

  Both this message and the server's response are as long as
  the output of the hash; the length is 20 bytes for SHA-1. ]

IAC SB AUTHENTICATION IS
```

```
        SRP CLIENT|ONE_WAY|ENCRYPT_USING_TELOPT RESPONSE
        xx xx xx xx xx xx xx xx ...
        IAC SE

        [ The server accepts the response and sends its own proof. ]

                                         IAC SB AUTHENTICATION REPLY
                                         SRP CLIENT|ONE_WAY|
                                         ENCRYPT_USING_TELOPT ACCEPT
                                         yy yy yy yy yy yy yy yy ...
                                         IAC SE
```

6. Security Considerations

The ability to negotiate a common authentication mechanism between client and server is a feature of the authentication option that should be used with caution. When the negotiation is performed, no authentication has yet occurred. Therefore, each system has no way of knowing whether or not it is talking to the system it intends. An intruder could attempt to negotiate the use of an authentication system which is either weak, or already compromised by the intruder.

Since SRP relies on the security of the underlying public-key cryptosystem, the modulus "n" should be large enough to resist brute-force attack. A length of at least 1024 bits is recommended, and implementations should reject attempts to use moduli that are shorter than 512 bits, or attempts to use invalid moduli and generator parameters (non-safe-prime "n" or non-primitive "g").

As an implementation of the TELNET AUTH option [RFC YYYY] all of the Security Considerations from that RFC MUST be considered applicable to this sub-option.

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

7. IANA Considerations

The authentication type SRP 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.

8. Normative References

[RFC 854] Postel, J. and J. Reynolds, "Telnet Protocol Specification", STD 8, RFC 854, May 1983.

[RFC YYYY] J. Altman, "Telnet Authentication Option", draft-altman-telnet-rfc2941bis-??.txt.

[RFC XXXX] Altman, J., "Telnet START_TLS Option", draft-altman-telnet-starttls-??.txt.

[RFC 4346] Dierks, T. and E. Rescorla. "The Transport Layer Security (TLS) Protocol Version 1.1.", RFC4346, April 2006.

[RFC 2945] Wu, T., "The SRP Authentication and Key Exchange System", RFC 2945, September 2000.

[RFC 4346] Dierks, T. and E. Rescorla. "The Transport Layer Security (TLS) Protocol Version 1.1.", RFC4346, April 2006.

9. Informational References

[SRP] T. Wu, "The Secure Remote Password Protocol", In Proceedings of the 1998 ISOC Network and Distributed System Security Symposium, San Diego, CA, pp. 97-111.

10. Editor's Address

Jeffrey Altman
Secure Endpoints Inc
255 W 94th Street
New York NY 10025

EMail: jaltman@secure-endpoints.com

Full Copyright Statement

Intellectual Property

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).