Internet Draft
Category: Informational
Expires: March 2, 2004

K. Murchison
M. Crispin

28 August 2003

The LOGIN SASL Mechanism

<draft-murchison-sasl-login-00.txt>

Status of this Memo

Copyright Notice

Abstract

This document documents the obsolete clear-text user/password Simple Authentication and Security Layer (SASL) mechanism called the LOGIN mechanism. The LOGIN mechanism was intended to be used, in combination with data confidentiality services provided by a lower layer, in protocols which lack a simple password authentication command.

Conventions Used in the 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 [KEYWORDS].

1. Background and Intended Usage

This document documents the obsolete LOGIN Simple Authentication and Security Layer ([SASL]) mechanism which was in use in protocols with no clear-text login command (e.g., [SMTP-AUTH]).

Note: The LOGIN SASL mechanism is obsoleted in favor of the PLAIN SASL mechanism ([PLAIN]). The LOGIN mechanism is documented here only for the purpose of backwards compatibility with legacy software. Clients SHOULD implement the PLAIN SASL mechanism and use it whenever offered by a server. The LOGIN SASL mechanism SHOULD NOT be used by a client when other plaintext mechanisms are offered by a server.

The name associated with this mechanism is "LOGIN".

The LOGIN SASL mechanism does not provide a security layer. This mechanism MUST NOT be used without adequate security protection as the mechanism affords no integrity nor confidentiality protection itself. The LOGIN SASL mechanism MUST NOT be advertised or used in any configuration that prohibits the PLAIN mechanism or plaintext LOGIN (or USER/PASS) command that sends passwords in the clear.

2. LOGIN SASL Mechanism

The authorization identity is the same string as the "username" in the traditional (non-SASL) LOGIN or USER commands; the authorization authenticator is the same string as the traditional "password". The authentication identity is the same as the authorization identity in this mechanism.

Only US-ASCII printable characters SHOULD be used in the username and password to permit maximal interoperability. If non-US-ASCII characters are used in a username, they MUST use UTF-8. Passwords MAY contain arbitrary binary data excluding NUL, CR and LF characters. However, if a password is supplied to the client as a sequence of characters (e.g., a password dialog box), those characters MUST be encoded as UTF-8.

The username MUST be less than 64 characters in length.

2.1. Client side of authentication protocol exchange

The client expects the server to issue a challenge. The client then responds with the authorization identity. The client then expects the server to issue a second challenge. The client then responds with the authorization authenticator. The contents of both challenges SHOULD be ignored.

2.2. Server side of authentication protocol exchange

The server issues the string "User Name" in challenge, and receives a client response. This response is recorded as the authorization identity. The server then issues the string "Password" in challenge, and receives a client response. This response is recorded as the authorization authenticator. The server must verify that the authorization authenticator permits login as the authorization identity.

Note: There is at least one widely deployed client which requires that the challenge strings transmitted by the server be "Username:" and "Password:" respectively. For this reason, server implementations MAY send these challenge strings instead of those listed above.

2.3. Example

This example shows the use of the LOGIN mechanism with the SMTP AUTH command [SMTP-AUTH] under the protection of SMTP STARTTLS [SMTP-TLS]. The user name is "tim" and the password is "tanstaaftanstaaf". The base64 encoding of the challenges and responses is part of the SMTP AUTH command, not part of the LOGIN specification itself. "C:" and "S:" indicate lines sent by the client and server respectively.

```
S: 220 smtp.example.com ESMTP server ready
C: EHLO test.example.com
S: 250-smtp.example.com
S: 250-STARTTLS
S: 250 AUTH CRAM-MD5
C: STARTTLS
S: 220 Ready to start TLS
<TLS negotiation, further commands are under TLS layer>
C: EHLO test.example.com
S: 250-smtp.example.com
S: 250 AUTH LOGIN CRAM-MD5
C: AUTH LOGIN
S: 334 VXNlciBOYW1lAA==
```

```
C: dGlt
S: 334 UGFzc3dvcmQA
C: dGFuc3RhYWZ0YW5zdGFhZg==
S: 235 Authentication successful.
```

3. Security Considerations

The LOGIN mechanism relies upon an underlying encryption layer or other secure channel for security. When used without an encryption layer or secure channel, it is vulnerable to a common network eavesdropping attack. Therefore the LOGIN mechanism MUST NOT be advertised or used in any configuration that prohibits the PLAIN mechanism or a plaintext LOGIN (or USER/PASS) command that sends passwords in the clear.

4. IANA Considerations

The registration for the LOGIN SASL mechanism follows:

SASL mechanism name: LOGIN
Security Considerations: See section 3 of this memo
Published specification: this memo
Person & email address to contact for futher information:
    See section 7 of this memo
Intended usage: OBSOLETE
Owner/Change controller: See section 7 of this memo

5. References

5.1. Normative References

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

[SASL] Melnikov, A., Ed., "Simple Authentication and Security Layer (SASL)", Isode, draft-ietf-sasl-rfc2222bis-xx.txt, Work In Progress.

5.2. Informative References

[PLAIN] Zeilenga, Kurt D., Ed., "The Plain SASL Mechanism", OpenLDAP Foundation, draft-ietf-sasl-plain-xx.txt, Work In Progress.

[SMTP-AUTH] Myers, J., "SMTP Service Extension for Authentication", Netscape Communications, RFC 2554, March 1999.

[SMTP-TLS] Hoffman, P., "SMTP Service Extension for Secure SMTP over Transport Layer Security", Internet Mail Consortium, RFC 3207, February 2002.

6. Acknowledgments

Thanks to Rob Siemborski for his input and feedback on this document.

7.

Author's Address

Kenneth Murchison
Oceana Matrix Ltd.
21 Princeton Place
Orchard Park, NY 14127

Phone: (716) 662-8973

EMail: ken@oceana.com

Mark R. Crispin
Networks and Distributed Computing
University of Washington
4545 15th Avenue NE
Seattle, WA 98105-4527

Phone: (206) 543-5762

EMail: MRC@CAC.Washington.EDU

8. Intellectual Property Considerations

The IETF takes no position regarding the validity or scope of any intellectual property 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; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication 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 implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

9. Full Copyright Statement