Network Working Group	D. Cridland	
Internet-Draft	Isode Limited	
Intended status: Informational	November 09, 2007	
Expires: May 12, 2008		

TOC

On the use of TLS Session resumption and SASL EXTERNAL
draft-cridland-sasl-tls-sessions-00

Status of this Memo

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

Abstract

Some SASL mechanisms provide a fast reauthentication option. TLS also provides this, and this memo outlines a proposal to use the TLS session resumption as a method for mechanism-independent fast reauthentication using SASL EXTERNAL.

Table of Contents

1. Conventions used in this document

TOC

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] (Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels," March 1997.).

2. Introduction

TOC

The [DIGEST-MD5] (Leach, P. and C. Newman, "Using Digest Authentication as a SASL Mechanism," May 2000.) [SASL] (Melnikov, A. and K. Zeilenga, "Simple Authentication and Security Layer (SASL)," June 2006.) mechanism provides a method of performing subsequent authentications using many fewer round-trips, known as "fast reauthentication". Most SASL mechanisms do not, which can cause a relatively high number of round-trips at application protocol startup.
In addition, the cost of cryptographic computation in clients can be quite high, leading to a slower application session setup.
This memo proposes a use of the EXTERNAL mechanism, defined in [SASL] (Melnikov, A. and K. Zeilenga, "Simple Authentication and Security Layer (SASL)," June 2006.), in conjunction with TLS session resumption as specified in [TLS] (Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.1," April 2006.), which effectively provides fast reauthentication in a generic manner for any SASL mechanism which supports channel binding.

3. Implementation

TOC

3.1. Initial Authentication

TOC

To use this method, the client first negotiates TLS as normal, then uses any SASL mechanism which supports channel binding to authenticate as normal whilst TLS is in effect. On successful authentication, the server then records the authorization identifier used against the SessionID used in TLS.
If the mechanism used in this initial authentication does not support, or use, channel binding, then the server MUST NOT record the authorization identifier against the SessionID.
The client SHOULD NOT provide a certificate during this initial TLS negotiation, as this would cause there to be multiple potential identities.

The TLS session so created SHOULD NOT be resumed except to reauthenticate to obtain the same authorization identifier.

3.2. Fast Reauthentication

TOC

First, a client resumes a TLS session, using the SessionID previously recorded as per Section 3.1 (Initial Authentication). The server, on resumption of the TLS session, then determines whether any authorization identifiers have been cached as per Section 3.1 (Initial Authentication), and advertises EXTERNAL if so. If EXTERNAL is advertised, then the client then authenticates using the EXTERNAL mechanism.
The server verifies this by checking that the authorization identifier was previously used with the SessionID recorded as per Section 3.1 (Initial Authentication).
If this check fails, the application protocol will reject the authentication. Clients SHOULD retry using a traditional SASL mechanism.

4. Open Issues

TOC

A significant problem with this method is that there is no negotiation to indicate the source of an externally asserted authorization identifier, in particular, there is no protocol by which a client can determine whether using EXTERNAL without specifying an authorization identifier will grant it the authorization identifier it was expecting. In particular, this suggests that use of the mechanism with a TLS client certificate may be particularly difficult.
Clients also cannot easily specify authorization identifiers to resolve this - not only do explicitly specified authorization identifiers tend to be treated as proxy-authentication requests, but the client cannot formally know what authorization identifier it was granted by default in the initial authentication.
This could be addressed by definition of a new SASL mechanism which would explicitly use the authorization identifier previously associated with the TLS session.
The problems outlined with EXTERNAL may be sufficient to consider replacing EXTERNAL itself with a family of mechanisms whose name indicates the source of the implicit authorization identifier.

5. IANA Considerations

TOC

This document has no actions for IANA.

TOC

6. Security Considerations

This method is only suitable in the case where the SASL mechanism used in initial authentication is actively using channel bindings, and the SessionID is secure.
In the case where the SessionID, and related session information, could be compromised on the wire, then the server cannot rely on this to provide an authorization identifier. This is the case where the encryption algorithm used is NULL, for example.
In the case where channel binding was not used by the SASL mechanism, then the server cannot know whether the client has been subjected to an MITM. The client can know by suitable verification of the server certificate, if one is provided, however, the server cannot know if this was carried out, and for anonymous cipher suites, there is no certificate to verify.
If an MITM is in effect without channel binding, use of this mechanism could otherwise allow the MITM to later reauthenticate.
Caching credentials sufficient to reauthenticate non-interactively, whether using this method, that of [DIGEST-MD5] (Leach, P. and C. Newman, “Using Digest Authentication as a SASL Mechanism,” May 2000.), or merely storing the plaintext password, may allow a third-party unauthorized program to obtain the credentials and access the protected service. Use of this method entirely removes the ability to recover a password from this cached data, however, since neither the password nor any data derived from it is needed to reauthenticate.
Other security considerations applicable to this method are to be found in both [SASL] (Melnikov, A. and K. Zeilenga, “Simple Authentication and Security Layer (SASL),” June 2006.) and [TLS] (Dierks, T. and E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.1,” April 2006.).
[The author notes that he is neither an expert on TLS nor on cryptography in general, hence there is probably more to consider than this.]

---

7. Acknowledgements

TOC

Comments were received on the idea, and/or this draft, from Sam Hartman, Kurt Zeilenga, Tony Finch, Alexey Melnikov, and others. Whether in agreement or dissent, the comments have refined and otherwise influenced the document.

---

8. References

TOC

8.1. Normative References

TOC

[KEYWORDS]	Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels,” BCP 14, RFC 2119, March 1997 (TXT, HTML, XML).

[SASL]	Melnikov, A. and K. Zeilenga, “Simple Authentication and Security Layer (SASL),” RFC 4422, June 2006 (TXT).
[TLS]	Dierks, T. and E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.1,” RFC 4346, April 2006 (TXT).

8.2. Informative References

TOC

[DIGEST-MD5]	Leach, P. and C. Newman, “Using Digest Authentication as a SASL Mechanism,” RFC 2831, May 2000 (TXT).

Author's Address

TOC

	Dave Cridland
	Isode Limited
	5 Castle Business Village
	36, Station Road
	Hampton, Middlesex TW12 2BX
	GB
Email:	dave.cridland@isode.com

Full Copyright Statement

TOC

Intellectual Property

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.