SIP WG V. Gurbani
Internet-Draft Bell Laboratories, Alcatel-Lucent
Updates: RFC3261 S. Lawrence
(if approved) Nortel Networks, Inc.
Intended status: Standards Track A. Jeffrey
Expires: November 19, 2009 Bell Laboratories, Alcatel-Lucent
May 18, 2009

Domain Certificates in the Session Initiation Protocol (SIP)
draft-ietf-sip-domain-certs-04

Status of this Memo

Copyright Notice

Abstract

This document describes how to construct and interpret certain information in a X.509 PKIX-compliant certificate for use in a Session Initiation Protocol (SIP) over Transport Layer Security (TLS) connection. More specifically, this document describes how to encode and extract the identity of a SIP domain in a certificate and how to use that identity for SIP domain authentication. As such, this document is relevant both to implementors of SIP and to issuers of cetificates.

Table of Contents

1. Terminology

1.1. Key Words

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].

Additional definition(s):

SIP domain identity: An identity (e.g., "sip:example.com") contained in an X.509 certificate bound to a subject that identifies the subject as an authoritative SIP server for a domain.

2. Introduction

RFC 5246 [3] Transport Layer Security (TLS) has started to appear in an increasing number of Session Initiation Protocol (SIP) RFC 3261 [2] implementations. In order to use the authentication capabilities of TLS, certificates as defined by the Internet X.509 Public Key Infrastructure RFC 5280 [4] are required.

Existing SIP specifications do not sufficiently specify how to use certificates for domain (as opposed to host) authentication. This document provides guidance to ensure interoperability and uniform conventions for the construction and interpretation of certificates used to identify their holders as being authoritative for the domain.

The discussion in this document is pertinent to an X.509 PKIX-compliant certificate used for a TLS connection; this document does not define use of such certificates for any other purpose (such as S/MIME).

3. Problem statement

TLS uses RFC 5280 [4] X.509 Public Key Infrastructure to bind an identity or a set of identities, to the subject of a X.509 certificate. While RFC3261 provides adequate guidance on the use of X.509 certificates used for S/MIME, it is relatively silent on the use of such certificates for TLS. With respect to certificates for TLS, RFC3261 (Section 26.3.1) says:

> "Proxy servers, redirect servers and registrars SHOULD possess a site certificate whose subject corresponds to their canonical hostname."

The security properties of TLS and S/MIME as used in SIP are different: X.509 certificates for S/MIME are generally used for end-to-end authentication and encryption, thus they serve to bind the identity of a user to the certificate and RFC3261 is sufficiently clear that in certificates used for S/MIME, the subjectAltName field will contain the appropriate identity. On the other hand, X.509 certificates used for TLS serve to bind the identities of the per-hop domain sending or receiving the SIP messages. However, the lack of guidelines in RFC3261 on exactly where to put identities -- in the subjectAltName field or carried as a Common Name (CN) in the Subject field -- of a X.509 certificates created ambiguities. Following the accepted practice of the time, legacy X.509 certificates were allowed to store the identity in the CN field of the certificate instead of the currently specified subjectAltName extension. Lack of further guidelines on how to interpret the identities, which identity to choose if more than one identity is present in the certificate, the behavior when multiple identities with different schemes were present in the certificate, etc. lead to ambiguities when attempting to interpret the certificate in a uniform manner for TLS use.

This document shows how the certificates are to be used for mutual authentication when both the client and server possess appropriate certificates, and normative behavior for matching the DNS query string with an identity stored in the X.509 certificate. Furthermore, a certificate can contain multiple identities for the subject in the subjectAltName extension (the "subject" of a certificate identifies the entity associated with the public key stored in the public key field.) As such, this document specifies appropriate matching rules to encompass various subject identity representation options. And finally, this document also provides guidelines to service providers for assigning certificates to SIP servers.

The rest of this document is organized as follows: the next section provides an overview of the most primitive case of a client using DNS to access a SIP server and the resulting authentication steps. Section 5 looks at the reason why mutual inter-domain authentication is desired in SIP, and the lack of normative text and behavior in RFC3261 for doing so. Section 6 outlines normative guidelines for a service provider assigning certificates to SIP servers. Section 7 provides normative behavior on the SIP entities (user agent clients, user agent servers, registrars, redirect servers, and proxies) that need perform authentication based on X.509 certificates. Section 8 includes the security considerations.

4. SIP domain to host resolution

Routing in SIP is performed by having the client execute RFC 3263 [6] procedures on a URI, called the "Application Unique String (AUS) (c.f. Section 8 of RFC 3263 [6]). These procedures take as input a SIP AUS (the SIP domain) and return an ordered set containing one or more IP addresses, and a port number and transport corresponding to each IP address in the set (the "Expected Output") by querying an Domain Name Service (DNS). If the transport indicates the use of TLS, then a TLS connection is opened to the server on a specific IP address and port. The server presents an X.509 certificate to the client for verification as part of the initial TLS handshake.

The client extracts identifiers from the Subject and any subjectAltName extension in the certificate (see Section 7.1) and compares these values to the AUS. If any identifier match is found, the server is considered to be authenticated and subsequent signaling can now proceed over the TLS connection. Matching rules for X.509 certificates and the normative behavior for clients is specified in Section 7.3.

As an example, consider a request that is to be routed to the SIP address "sips:alice@example.com". This address requires a secure connection to the SIP domain "example.com", which becomes the SIP AUS value. Through a series of DNS manipulations, the AUS is mapped to a set of host addresses and transports. The entity attempting to create the connection selects an address appropriate for use with TLS from this set. When the connection is established to that server, the server presents a certificate asserting the identity "sip:example.com". Since the domain part of the SIP AUS matches the subject of the certificate, the server is authenticated (see Section 7.2 for the normative rules that govern this comparison)

> SIPS borrows this pattern of server certificate matching from HTTPS. However, RFC 2818 [7] prefers that the identity be conveyed as a subjectAltName extension of type dNSName rather than the common practice of conveying the identity in the CN field of the Subject field. Similarly, this document recommends that the SIP domain identity be conveyed as a subjectAltName extension of type uniformResourceIdentifier (c.f. Section 6, Section 7.1).

> A domain name in an X.509 certificates is properly interpreted only as a sequence of octets to be compared to the URI used to reach the host. No inference can be made based on the DNS name hierarchy. For example, a valid certificate for "example.com" does not imply that the owner of that certificate has any relationship at all to "subname.example.com".

5. The need for mutual interdomain authentication

Consider the SIP trapezoid shown in Figure 1.

```
  Proxy-A.example.com           Proxy-B.example.net
     +--------+                    +--------+
     | Proxy  |--------------------| Proxy  |
```

```
        +----+--+                    +---+---+
             |                           |
             |                           |
             |                           |
             |                         +---+
           0---0                       |   |
            /-\                        |___|
           +----+                     /    /
                                     +-----+

      alice@example.com          bob@example.net
```

Figure 1: SIP Trapezoid

An user, alice@example.com, invites bob@example.net for a multimedia communication session. Alice's outbound proxy, Proxy-A.example.com, uses normal RFC 3263 [6] resolution rules to find a proxy -- Proxy-B.example.net -- in the example.net domain that uses TLS. Proxy-A actively establishes an interdomain TLS connection with Proxy-B and each presents a certificate to authenticate that connection.

RFC 3261 [2] section 26.3.2.2 "Interdomain Requests" states that when a TLS connection is created between two proxies:

> "Each side of the connection SHOULD verify and inspect the certificate of the other, noting the domain name that appears in the certificate for comparison with the header fields of SIP messages."

However, RFC3261 is silent on whether to use the subjectAltName or CN of the certificate to obtain the domain name, and which takes precedence when there are multiple names identifying the holder of the certificate.

The authentication problem for Proxy-A is straightforward: assuming a secure DNS infrastructure and no routing attacks, Proxy-A already knows that Proxy-B is a valid proxy for the example.net domain. Thus, in the certificate Proxy-A receives from Proxy-B, Proxy-A looks for the host name ("Proxy-B.example.net") or an identity consisting of a SIP URI ("sip:example.net") that asserts Proxy-B's authority

over the example.net domain. Normative behavior for a TLS client like Proxy-A is specified in Section 7.3.

The problem for Proxy-B is slightly more complex since it accepted the TLS request passively. Thus, Proxy-B does not possess an equivalent AUS that it can use as an anchor in matching identities from Proxy-A's certificate.

> RFC 3261 [2] section 26.3.2.2 only tells Proxy-B to "compare the domain asserted by the certificate with the 'domainname' portion of the From header field in the INVITE request." The difficulty with that instruction is that the domainname in the From header field is not always that of the domain from which the request is received.

The normative behavior for a TLS server like Proxy-B that passively accepts a TLS connection and requires authentication of the sending peer domain is provided in Section 7.4.

6. Certificate usage by a SIP service provider

Service providers MAY continue the practice of using existing certificates for SIP usage with the identity conveyed in the Subject field; however, such usage is NOT RECOMMENDED for new certificates, which MUST contain the identity or identities in the subjectAltName extension field.

When assigning certificates to authoritative servers, a SIP service provider MUST ensure that the SIP AUS used to reach the server appears as an identity in the subjectAltName field, or for compatibility with existing certificates, the Subject field of the certificate. In practice, this means that a service provider distributes to its users SIP URIs whose domain portion corresponds to an identity for which the service provider has been issued a certificate.

7. Behavior of SIP entities

This section normatively specifies the behavior of SIP entities when using X.509 certificates to determine an authenticated SIP domain identity.

The first two subsections apply to all SIP implementations that use TLS to authenticate the peer: Section 7.1 describes how to extract a set of SIP identities from the certificate obtained from a TLS peer, and Section 7.2 specifies how to compare SIP identities. The remaining subsections provide context for how and when these rules are to be applied by entitites in different SIP roles.

7.1. Finding SIP Identities in a Certificate

Implementations (both clients and server) MUST determine the validity of a certificate by following the procedures for constructing a certificate path and checking revocation status described in RFC 5280 [4]. This document adds additional rules for interpreting an X.509 certificate for use in SIP.

As specified by RFC 5280 [4] section 4.2.1.12, implementations MUST check for restrictions on certificate usage declared by any extendedKeyUsage extentions in the certificate. The SIP Extended Key Usage (EKU) document [11] defines an extendedKeyUsage for SIP.

Given an X.509 certificate that the above checks have found to be acceptable, the following describes how to determine what SIP domain identity or identities the certificate contains. A single certificate can serve more than one purpose - that is, the certificate might contain identities not acceptable as SIP, domain identities and/or might contain one or more identities that are acceptable for use as SIP domain identities.

1. Examine each value in the subjectAltName field. The subjectAltName field and the constraints on its values are defined in Section 4.2.1.6 of RFC 5280 [4]. The subjectAltName field can be absent or can contain one or more values. Each value in the subjectAltName has a type; the only types acceptable for encoding a SIP domain identity SHALL be:

   URI If the scheme of the URI is not "sip", then the implementation MUST NOT accept the value as a SIP domain identity.

      If the scheme of the URI value is "sip", and the URI value that contains a userpart (there is an '@'), the implementation MUST NOT accept the value as a SIP domain identity (a value with a userpart identifies an individual user, not a domain).

      If the scheme of the URI value is "sip", and there is no userinfo component in the URI (there is no '@'), then the implementation MUST accept the hostpart as a SIP domain identity.

Note: URI scheme tokens are always case insensitive

DNS An implementation MUST accept a domain name system identifier as a SIP domain identity if and only if no other identity is found that matches the "sip" URI type described above.

2. If and only if the subjectAltName does not appear in the certificate, the implementation MAY examine the CN field of the certificate. If a valid DNS name is found there, the implementation MAY accept this value as a SIP domain identity.

3. Accepting a DNS name in the CN value is allowed for backward compatibility, but a Certificate Authority SHOULD NOT issue new certificates with the identity as a DNS name in the CN value; see Section 6.

The above procedure yields a set containing zero or more identities from the certificate. A client uses these identities to authenticate a server (see Section 7.3) and a server uses them to authenticate a client (see Section 7.4).

7.2. Comparing SIP Identities

When an implementation (either client or server) compares two values as SIP domain identities:

Implementations MUST compare only the DNS name component of each SIP domain identifier; an implementation MUST NOT use any scheme or parameters in the comparison.

Implementations MUST compare the values as DNS names, which means that the comparison is case insensitive as specified by RFC 4343 [5]. Implementations MUST handle Internationalized Domain Names (IDNs) in accordance with Section 7.2 of RFC 5280 [4] .

Implemenations MUST match the values in their entirety:

Implementations MUST NOT match suffixes. For example, "foo.example.com" does not match "example.com".

Implemenations MUST NOT match any form of wildcard, such as a leading "." or "*." with any other DNS label or sequence of labels. For example, "*.example.com" matches only "*.example.com" but not "foo.example.com". Similarly, ".example.com" matches only ".example.com", and does not match "foo.example.com."

RFC 2818 [7] (HTTP over TLS) allows the dNSName component to contain a wildcard; e.g., "DNS:*.example.com". RFC 5280 [4], while not disallowing this explicitly, leaves the interpretation of wildcards to the individual specification. RFC 3261 [2] does not provide any guidelines on the presence of wildcards in certificates. Through the rule above, this document prohibits such wildcards in certificates for SIP domains.

7.3. Client behavior

A client uses the domain portion of the SIP AUS to query a (possibly untrusted) DNS to obtain a result set, which is one or more SRV and A records identifying the server for the domain (see Section 4 for an overview.)

The SIP server, when establishing a TLS connection, presents its certificate to the client for authentication. The client MUST determine the SIP domain identities in the server certificate using the procedure in Section 7.1. Then, the client MUST compare the original domain portion of the SIP AUS used as input to the RFC 3263 [6] server location procedures to the SIP domain identities obtained from the certificate.

- o If there were no identities found in the server certificate, the server is not authenticated.

- o If the AUS matches any SIP domain identity obtained from the certificate when compared as described in section Section 7.2, the server is authenticated for the domain.

If the server is not authenticated, the client MUST close the connection immediately.

7.4. Server behavior

When a server accepts a TLS connection, the server presents its own X.509 certificate to the client. To authenticate the client, the server asks the client for a certificate. If the client possesses a certificate, that certificate is presented to the server. If the client does not present a certificate, the client MUST NOT be considered authenticated.

> Whether or not to close a connection if the client does not present a certificate is a matter of local policy, and depends on the authentication needs of the server for the connection. Some currently deployed servers use Digest authentication to authenticate individual requests on the connection, and choose to

> treat the connection as authenticated by those requests for some purposes (but see Section 8.1).
>
> If the local server policy requires client authentication for some local purpose, then one element of such a local policy might be to allow the connection only if the client is authenticated. For example, if the server is an inbound proxy that has peering relationships with the outbound proxies of other specific domains, the server might allow only connections authenticated as coming from those domains.

To authenticate the client, the server MUST obtain the set of SIP domain identities from the client certificate as described in Section 7.1. Because the server accepted the TLS connection passively, unlike a client, the server does not possess an AUS for comparison. Nonetheless, server policies can use the set of SIP domain identities gathered from the certificate in Section 7.1 to make authorization decisions.

For example, a very open policy could be to accept a X.509 certificate and validate the certificate using the procedures in RFC 5280 [4] . If the certificate is valid, the identity set is logged.

Alternatively, the server could have a list of all SIP domains the server is allowed to accept connections from; when a client presents its certificate, for each identity in the client certificate, the server searches for the identity in the list of acceptable domains to decide whether or not to accept the connection. Other policies that make finer distinctions are possible.

The decision of whether or not the authenticated connection to the client is appropriate for use to route new requests to the client domain is independent of whether or not the connection is authenticated; the connect-reuse [9] draft discusses this aspect in more detail.

7.5. Proxy behavior

A proxy MUST use the procedures defined for a User Agent Server (UAS) in Section 7.4 when authenticating a connection from a client.

A proxy MUST use the procedures defined for a User Agent Client (UAC) in Section 7.3 when requesting an authenticated connection to a UAS.

If a proxy adds a Record-Route when forwarding a request with the expectation that the route is to use secure connections, the proxy MUST insert into the Record-Route header a URI that corresponds to an identity for which the proxy has a certificate; if the proxy does not

insert such a URI, then creation of a secure connection using the value from the Record-Route as the AUS will be impossible.

7.6. Registrar behavior

A SIP registrar, acting as a server, follows the normative behavior of Section 7.4. When the SIP registrar accepts a TLS connection from the client, the SIP registrar presents its certificate. Depending on the registrar policies, the SIP registrar can challenge the client with HTTP Digest.

7.7. Redirect server behavior

A SIP redirect server follows the normative behavior of a UAS as specified in Section 7.4.

7.8. Virtual SIP Servers and Certificate Content

In the "virtual hosting" cases where multiple domains are managed by a single application, a certificate can contain multiple subjects by having distinct identities in the subjectAltName field as specified in RFC 4474 [8]. Clients seeking to authenticate a server on such a virtual host can still follow the directions in Section 7.3 to find the identity matching the SIP AUS used to query DNS.

Alternatively, if the TLS client hello "server_name" extension as defined in RFC 5246 [3] is supported, the client SHOULD use that extension to request a certificate corresponding to the specific domain (the SIP AUS) that the client is seeking to establish a connection with.

8. Security Considerations

The goals of TLS (when used with X.509 certificates) include the following security guarantees at the transport layer:

Confidentiality: packets tunneled through TLS can be read only by the sender and receiver.

Integrity: packets tunneled through TLS cannot be undetectably modified on the connection between the sender and receiver.

Authentication: each principal is authenticated to the other as possessing a private key for which a certificate has been issued. Moreover, this certificate has not been revoked, and is verifiable by a certificate chain leading to a (locally configured) trust anchor.

We expect appropriate processing of domain certificates to provide the following security guarantees at the application level:

Confidentiality: SIPS messages from alice@example.com to bob@example.net can be read only by alice@example.com, bob@example.net, and SIP proxies issued with domain certificates for example.com or example.net.

Integrity: SIPS messages from alice@example.com to bob@example.net cannot be undetectably modified on the links between alice@example.com, bob@example.net, and SIP proxies issued with domain certificates for example.com or example.net.

Authentication: alice@example.com and proxy.example.com are mutually authenticated; moreover proxy.example.com is authenticated to alice@example.com as an authoritative proxy for domain example.com. Similar mutual authentication guarantees are given between proxy.example.com and proxy.example.net and between proxy.example.net and bob@example.net. As a result, alice@example.com is transitively mutually authenticated to bob@example.net (assuming trust in the authoritative proxies for example.com and example.net).

8.1. Connection authentication using Digest

Digest authentication in SIP provides for authentication of the message sender to the challenging UAS. As commonly deployed, digest authentication provides only very limited integrity protection of the authenticated message, and has no provision for binding the authentication to any attribute of the transport. Many existing SIP deployments have chosen to use the Digest authentication of one or more messages on a particular transport connection as a way to authenticate the connection itself - by implication, authenticating other (unauthenticated) messages on that connection. Some even choose to similarly authenticate a UDP source address and port based on the digest authentication of another message received from that address and port. This use of digest goes beyond the assurances that the Digest Authentication mechanism was designed to provide. A SIP imlementation SHOULD NOT use the Digest Authentication of one message on a TCP connection or from a UDP peer to infer any authentication of any other messages on that connection or from that peer. Authentication of the domain at the other end of a connection SHOULD be accomplished using TLS and the certificate validation rules described by this specification instead.

9. IANA Considerations

This memo does not contain any considerations for IANA.

10. Acknowledgments

The following IETF contributors provided substantive input to this document: Jeroen van Bemmel, Michael Hammer, Cullen Jennings, Paul Kyzivat, Derek MacDonald, Dave Oran, Jon Peterson, Eric Rescorla, Jonathan Rosenberg, Russ Housley. Special acknowledgement goes to Stephen Kent for extensively reviewing draft versions and suggesting invaluable feedback, edits, and comments.

Paul Hoffman, Eric Rescorla and Robert Sparks provided much valuable WGLC comments.

11. References

11.1. Normative References

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

[2] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[3] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[4] Cooper, D., Santesson, S., Farrell, S., Boyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.

[5] Eastlake, D., "Domain Name System (DNS) Case Insensitivity Clarification", RFC 4343, January 2006.

11.2. Informative References

[6] Rosenberg, J. and H. Schulzrinne, "Session Initiation Protocol (SIP): Location SIP Servers", RFC 3263, June 2002.

[7] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000.

[8] Peterson, J. and C. Jennings, "Enhancements for Authenticated

Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, August 2006.

[9] Mahy, R., Gurbani, V., and B. Tate, "Connection Reuse in the Session Initiation Protocol", draft-ietf-sip-connect-reuse-13.txt (work in progress), March 2009.

[10] Drage, K., "A Process for Handling Essential Corrections to the Session Initiation Protocol (SIP)", draft-drage-sip-essential-correction-03.txt (work in progress), July 2008.

[11] Lawrence, S. and V. Gurbani, "Using Extended Key Usage (EKU) for Session Initiation Protocol (SIP) X.509 Certificates", draft-ietf-sip-eku-05.txt (work in progress), May 2009.

Appendix A. Editorial guidance (non-normative)

This document is intended to update RFC 3261 in accordance with the SIP Working Group procedures described in [10] or its successor.

This appendix provides guidance to the editor of the next comprehensive update to RFC 3261 [2] on how to incorporate the changes provided by this document.

A.1. Additions

The content of sections Section 4 through Section 7 inclusive can be incorporated as subsections within a section that describes SIP domain authentication.

The contents of Section 8.1 can be incorporated into the Security Considerations section of the new document.

All normative references from this document can be carried forward to the successor document.

A.2. Changes

The following subsections describe changes in specific sections of RFC 3261 [2] that need to be modified in the successor document to align them with the content of this document. In each of the following, the token <domain-authentication> is a reference to the section added as described in Appendix A.1.

A.2.1. 26.3.1

The current text says:

> Proxy servers, redirect servers and registrars SHOULD possess a site certificate whose subject corresponds to their canonical hostname.

The suggested replacement for the above is:

> Proxy servers, redirect servers, registrars, and any other server that is authoritative for some SIP purpose in a given domain SHOULD possess a certificate whose subject is a SIP domain as described in <domain-authentication>.

Authors' Addresses

Vijay K. Gurbani
Bell Laboratories, Alcatel-Lucent
1960 Lucent Lane
Room 9C-533
Naperville, IL 60566
USA

Phone: +1 630 224-0216
Email: vkg@alcatel-lucent.com

Scott Lawrence
Nortel Networks, Inc.
600 Technology Park
Billerica, MA 01821
USA

Phone: +1 978 288 5508
Email: scott.lawrence@nortel.com

Alan S.A. Jeffrey

Bell Laboratories, Alcatel-Lucent
1960 Lucent Lane
Room 9C-533
Naperville, IL 60566
USA

Email: ajeffrey@alcatel-lucent.com