Network Working Group
Internet-Draft
Expires: December 30, 2002

J. Peterson
NeuStar

July 1, 2002

Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)
draft-peterson-sip-identity-01

Status of this Memo

This Internet-Draft will expire on December 30, 2002.

Copyright Notice

Abstract

The existing mechanisms for expressing identity in the Session Initiation Protocol oftentimes do not permit an administrative domain to verify securely the identity of the originator of a request. This document recommends practices and conventions for authenticating end users, and proposes a way to distribute secure authenticated identities within SIP messages.

Table of Contents

1. Introduction

This document provides enhancements to the existing mechanisms for authenticated identity management in the Session Initiation Protocol (SIP [1]).

The baseline SIP protocol allows a user agent to express the identity of its user in a number of headers. The primary place for identity information asserted by the sender of a request is the From header. The From header field contains a URI (like 'sip:alice@atlanta.com') and an optional display-name (like "Alice") that identifies the originator of the request. A user may have many identities that are used in different contexts.

Typically, this URI is an address-of-record that can be dereferenced in order to contact the originator of the request; specifically, it is usually the same address-of-record under which a user registers their devices in order to receive incoming requests. This address-of-record is assigned and maintained by the administrator of the SIP service in the domain identified by the host portion of the address-of-record (which may have any of a number of relationships with the end user). However, the From field of a request can usually be set arbitrarily by the user of a SIP user agent; the From header of a message provides no internal assurance that the originating user can legitimately claim this identity. Nevertheless many SIP user agents will obligingly display the contents of the From field as the identity of the originator of a received request (as a sort of 'Caller-ID' function).

To satisfy the requirement for a more reliable way of identifying parties in a SIP session, a number of cryptographic authentication systems are described in the SIP standard, including mechanisms based on HTTP Digest, S/MIME and transport or network layer security. Among other things, these mechanisms allow a server to verify that a user agent can legitimately assert a specific identity. Whether or not the recipient of these credentials can verify them is based on whether the credentials are asymmetric, and publicly verifiable by third parties, or symmetric, and verifiable only by parties that have a pre-existing relationship with the user.

Symmetric: Authentication with symmetric keys usually entails the transmission of some sort of secret credentials (typically a username and password) from the client to the server. Secrets-based authentication assumes a pre-existing relationship (an agreement on a secret) between the client that originates a request and the server that responds with a challenge. Useful secrets-based authentication schemes use cryptography to conceal the credentials so they can not be observed and reused by

eavesdroppers on the network. Secrets are usually memorized by end users, and thus do not necessitate any special configuration of user agents.

Asymmetric: Asymmetric credentials require a Public Key Infrastructure (PKI) that manages public and private keys. PKI-based authentication usually relies on a certificate authority that issues a custom certificate to each entity that would like to prove its identity, and a common root certificate to each entity that would like to verify the identity of others. In this system there is no need for any pre-existing relationship between the clients and servers (unless in the absence of a certificate authority self-signed certificates are used). However, PKI has to date only been effective in asserting the identity of a hostname - there is a widespread belief that implementing a PKI for certificates that assert the identity of individuals is currently impractical. Each host MUST be configured with any certificate that asserts its identity.

Most user agents authenticate themselves with shared secrets. In baseline SIP, the Digest authentication method (which is required for all user agents and servers) allows users to provide a username and password to authenticate themselves in the context of a particular realm (for example, the identity 'alice' within the realm 'atlanta.com' might have the password 'x63Mdo+').

Digest therefore works well for functions like SIP registration, in which the target of a request is a server within the realm in which a user can prove an identity. However, the credentials with which a user proves that they are 'sip:alice@atlanta.com' cannot be verified by a server in another realm, like biloxi.com - Alice shares the secret with atlanta.com, not biloxi.com. Thus, if Alice were to send a request to a proxy server in the biloxi.com realm, biloxi.com has no way of determining whether or not Alice can legitimately claim the identity 'sip:alice@atlanta.com'. biloxi.com is then left only with the unreliable From field for ascertaining the identity of the originator of interdomain requests.

However, were Alice to proxy her request through an atlanta.com proxy server, atlanta.com might be able to verify her identity before passing the request to its destination, biloxi.com. Therefore, this document proposes a new logical role for network intermediaries called an authentication service. The authentication service role would most likely be instantiated by the local outbound proxy server within an administrative domain. Once an authentication service has verified the identity of the originator of a request, it can add the result of the authentication process to the request for the benefit of downstream recipients.

User agents can be configured, statically or on a per-call basis, to send requests through an authentication service. After a request has passed through an authentication service in a given domain (e.g. atlanta.com) downstream recipients (e.g. in biloxi.com) will be able to determine that atlanta.com asserts a specific authenticated identity for the originator of this message, like 'sip:alice@atlanta.com'.

In some cases, this authenticated identity can be distributed by the authentication service to any potential recipients of the request without restriction. In other cases, for reasons of network policy, or user privacy constraints, the distribution of the authenticated identity will be restricted.

In summary, the identity that appears in the From field of a SIP request provides a way that the originator can be canonically reached (and therefore provides some accountability for that user). The best way for a SIP user to prove that they can legitimately claim an identity is to provide the same credentials they would need to provide in order to register to receive requests for that identity. For that purpose, this document defines an authentication service that verifies the credentials of an end user in their local administrative domain before sending requests to their destinations. This authentication service can then sign the identity that results from this authentication and make this identity available to recipients of the request, thereby proving that the administrative domain responsible for the originating user registers has verified that user's identity. Effectively, this allows a user's authentication with a single server to be bootstrapped into a publicly-verifiable authentication.

By way of example, the manner in which a user authenticates themselves to an authentication service is in this document restricted to the mechanisms that are available in [1], specifically the Digest authentication scheme, as it is common to all SIP-compliant endpoints. However, these mechanisms have no dependency on any particular authentication scheme.

2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC2119 [2] and indicate requirement levels for compliant SIP implementations.

3. Sending Requests through an Authentication Service

An authentication service is a logical role played by a network intermediary, such as a SIP proxy server. Commonly, the authentication service function will be instantiated by a local outbound proxy server. Authentication services are capable of verifying the identity of users through some means. Authentication services are also capable of sharing this verified identity in a secure manner.

Requests from a user agent are sent through an authentication service because the user agent is configured (with a pre-loaded Route header, perhaps) to send all requests through the service.

A user has some incentive to send calls through an authentication service, in that:

- Authentication services help to prevent identity theft, and the many potential annoyances that could result from being impersonated, and,

- Administrative domains could implement policies that reject requests from users that have not gone through an authentication service appropriate for the administrative domain listed in the From header of their messages.

Optimally, a user should be able to send SIP messages to their authentication service directly, without going through any SIP proxy servers, since many authentication systems (including Digest) are not optimally secure when handled by intermediaries. An authentication service will frequently be co-located with a user agent's first-hop local outbound proxy.

4. Authentication Practices

When a user first forms a connection to a SIP entity that implements the authentication service role, the user SHOULD make use of network or transport layer security, preferably contacting the authentication service without going through any intermediaries. This document recommends the use of Transport Layer Security (TLS) to connect to the authentication service. This in turn allows the authentication service to potentially offer a certificate directly to the end user, as well as ensuring confidentiality, integrity, and replay protection during the challenge phase. If the user agent is more than one hop away from the authentication service, it may make sense to use the SIPS URI scheme to improve the security of requests routed through the authentication service.

Any entity that implements the authentication service role MUST possess a certificate that has been issued by a certificate authority. The SubjectAltName of the certificate SHOULD be the fully-qualified domain name of the device on which the authentication service is running. Only one logical authentication service SHOULD operate in a given administrative domain. The manner in which an authentication service can be recognized to be the canonical authority for an administrative domain is currently an open issue.

4.1 Issuing Challenges with Realms

Obviously, all authentication services have their own sets of users and corresponding credentials; when a user is challenged by an authentication service, that user selects credentials that are appropriate for the service in question. For the purposes of this document, following the terminology in Digest authentication, an authentication service has a 'realm' which provides a context in which a user is asked to authenticate. For example, when an authentication service challenges a user for Digest authentication with the 407 status code, the challenge MUST be sent for a realm corresponding to the hostname of the authentication service. Note that a user agent SHOULD consider it a cause for concern (though not necessarily an error condition) if the realm of a challenge does not correspond both with the hostname of the authentication service and any certificate presented by the authentication service on connection - users SHOULD be notified of this occurrence.

Once again, note that Digest authentication is used merely by way of example. Other mechanisms within SIP, or out-of-band mechanisms, could be used to authenticate the user. If these authentication systems do not explicitly support the concept of realms, the realm in which a challenge occurs should be understood by the user agent to be the hostname of the authentication service. Any method of

authentication used by an authentication service, MUST therefore have a means of communicating, at the very least, its hostname to a user. If these authentication systems do not support credentials that express a specific username, then a username SHOULD be taken by the authentication service from the username portion of the URI in the From header field of the SIP request.

4.2 Determining Identity with Credentials

If a user agent is challenged (in SIP Digest, for example, with a 401 or 407 response code) and it has access to credentials for the realm in question, these credentials will be provided in a resubmission of the request to the authentication service. In Digest, the authentication service MUST extract and verify these credentials from the resubmission of the request.

The username associated with these credentials SHOULD be combined with the name of the administrative domain of the authentication service in order to form the authenticated identity of the user. For example, if the credentials were valid for the username 'alice', for an authentication service within the atlanta.com administrative domain, the authenticated identity would be 'sip:alice@atlanta.com'. An authentication service MAY also have a particular display-name which it associates with particular users that will be included in the authenticated identity.

Note that some user agents MAY provide 'anonymous' credentials with no password in a resubmission of a request after a challenge. Whether or not an authentication service considers this to be a successful authentication is a matter of local policy, but the authentication service SHOULD NOT assert this 'anonymous' identity to others in the manner described in Section 5.

The credentials that a user provides to an authentication service SHOULD be the same credentials that are provided when the user registers in this administrative domain.

4.3 Analyzing Requests

Some authentication services MAY wish to inspect the contents of the From header of an outbound request. Depending on the policy of the authentication service, it might not be appropriate for the From header to differ from the authenticated identity that the service has verified. Some authentication services MAY reject requests (with a 403 Forbidden) that assert an inappropriate identity in the From.

Authentication services MAY also place restrictions on the display-name as well as the URI associated with the From header.

Note that some users MAY supply an anonymous From header (see [3]) for some requests. Authentication services generally SHOULD NOT consider this to conflict with any identity information learned in the authentication process. For more information on the obligations of authentication services with respect to privacy, see Section 8.1.

It may not be necessary for an authentication service to prevent the arbitrary assignment of the From field if the authentication service has another way of sharing authenticated identity information (see Section 5). Steps for reconciling the user-asserted From header with authenticated identity data are given in Section 6.2.

4.4 Accounting for Authentication

Authentication services MAY record the successful authentication of a request, including its dialog identifiers and Request-URI, in order to provide some accountability when administrative requests are made for information about the parties participating in a particular session. Some mechanisms by which a user is authenticated and authorized may also persist accounting data about the request, although such mechanisms are outside the scope of this document.

Authentication services MAY also wish to log failed authentication attempts, especially those that reflect repeated attempts to try different credentials for the same username.

4.5 Forwarding the Request

Once an authentication service has authenticated the originator of a request, if it does not wish to provide any further identification services, it MUST subsequently forward the request in accordance with the conventional request routing logic in the SIP specification.

If the authentication services also wishes to share the authenticated identity it has verified, it can use the mechanisms described in Section 5.

5. Sharing Verified Identities

Authenticated identities SHOULD be shared unless the authentication service has a reason to do otherwise. By authenticating themselves, originating users must understand that they are giving the authentication service the right to share the provided identity with others. If they wish to prevent this, users MUST request privacy for their authentication information (see Section 8.1).

Note that the practices described in this section can also be leveraged by a logical authentication service that is instantiated by a user agent, provided the user agent holds a certificate that is publicly verifiable.

5.1 Authenticated Identity within a Body

As a way of sharing authenticated identity among parties in the network, a special type of MIME body, which will subsequently be referred to as an 'authenticated identity body', is defined in this section. An authenticated identity body allows an authentication service to cryptographically sign the identity of the originator of the message in question.

An authenticated identity body is a MIME body of type 'message/sip' or 'message/sipfrag' (see [4]). This body MUST have a Content-Disposition disposition-type of 'auth-id', a new value defined in this document specifically for authenticated identity bodies. The Content-Disposition header SHOULD also contain a 'handling' parameter indicating that this MIME body is optional.

Authenticated identity bodies of the 'message/sipfrag' MIME type MUST contain the following headers: From, Date and Call-ID; they SHOULD also contain the To, Contact and Cseq header. The From header field MUST be populated by the authentication service with the authenticated identity itself, as discussed above in Section 4.2; if the authentication service verifies the display-name of the From header field, it MUST be included in the authenticated identity body, and if it does not verify the display-name of the From header field it MUST NOT be included. Authenticated identity bodies MAY contain any other headers that help to uniquely identify the transaction or provide related reference integrity. An example of an authenticated identity body is:

```
Content-Type: message/sipfrag
Content-Disposition: auth-id; handling=optional

From: Alice <sip:alice@atlanta.com>
To: Bob <sip:bob@biloxi.com>
Contact: <sip:alice@pc33.atlanta.com>
Date: Thu, 21 Feb 2002 13:02:03 GMT
Call-ID: a84b4c76e66710
Cseq: 314159 INVITE
```

Once the authenticated identity body has been fully populated, it MUST be signed by the authentication service that has created it. An unsigned authenticated identity body MUST NOT be honored by any recipients. An authenticated identity body MUST be signed with the public key corresponding to the same certificate that the authentication service uses to authenticate itself for the purposes of transport of network layer security such as TLS (see Section 4). A full example of a message with a signed authenticated identity MIME body is given in Section 5.4.

After the authenticated body has been signed, some entity SHOULD added it to any existing MIME bodies in the request, if necessary by transitioning the outermost MIME body to a 'multipart/mixed' format. But which participant in the dialog should add the authenticated identity body, the authentication service or the originating user agent? Both options are considered in the following sections. Authentication services MUST support the mechanism in Section 5.3 and MAY support the mechanism in Section 5.2.

5.2 Body Added by Authentication Service

The first possibility is that the authentication service could add the body to the request itself before forwarding the request. However, the authentication service role is usually played by entities that act as proxy servers for most requests, and proxy servers cannot modify message bodies. In order to add an authenticated identity body, the authentication service needs to act as a transparent back-to-back user agent, effectively terminating the request and re-originating it with a new body appended to any existing MIME bodies.

This mechanism has some potential advantages over sending the authenticated identity body back to the originating user agent. For one, it saves on additional round-trip times for signaling that result from passing the body back to the user agent. It also requires no new SIP mechanisms, whereas any method of asking a user agent to include a body in a resubmission to the current request would introduce new protocol requirements.

However, there are proposed SIP integrity mechanisms that place a signature over the entire message body in a SIP message header. Were a server to add to the body of a message that was protected by such signature, that could be perceived as an integrity violation by downstream recipients of the message.

5.3 Body Added by Client

Alternatively, the authentication service could in some fashion return the authenticated identity MIME body to the originating user agent, prompting the user agent to retry the request with the authenticated identity MIME body attached. No existing SIP mechanism performs this function. Therefore, this document defines a 428 "Use Authenticated Identity" response code.

An authentication service sends a 428 with a MIME body in order to request that a user agent add the enclosed MIME body to their request and retry the request. A 428 MUST have at most a single MIME body. This MIME body MUST be signed by the authentication service.

The use of 428 without any MIME body is also defined in this document. It can be sent by any server to reject a request because the request does not contain an authenticated identity body. A user agent receiving this rejection SHOULD retry their request through an authentication service.

In order to signal to the authentication services that the originating user agent supports the receipt of the 428 response code, a new option-tag has been defined, the 'auth-id' option-tag. User agents SHOULD supply the 'auth-id' option-tag in a Supported header whenever they provide credentials to a server (for example, in Digest authentication, whenever a Proxy-Authorization header is added to a request).

Using the 428 response code may introduce extra round-trip times for messages, delaying the setup of requests (one RTT for the 407, another for the 428). However, there are some circumstances under which extra RTTs may not impede performance. If the originating user agent possesses a non-stale nonce (assuming Digest authentication) from the authentication service, it can pre-emptively include a Proxy-Authorization header, eliminating one RTT. With regard to the second RTT, note that the request needn't necessarily go through the authentication service again once the authenticated identity body has been added - it could go directly to its destination, which reduce the impact of the second RTT.

There are two reasons why the originating user agent should be the party responsible for adding the authenticated identity body to the

request. Firstly, because this gives the client the opportunity to inspect the body itself (perhaps only to see whether or not it is encrypted; see Section 8.2) in order to verify that the authenticated identity corresponds with the provided credentials and the user's preferences. Secondly, the client can provide a signature over the entire body of the message (either with S/MIME or some header-based mechanism) so that the final recipient of messages can be assured that all information in the body is there at the originator's behest.

5.4 Example of a Request with an Authenticated Identity Body

The following shows a full SIP INVITE request with an authenticated identity body (one that has been added by the originating user agent):

```
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Max-Forwards: 70
Date: Thu, 21 Feb 2002 13:02:03 GMT
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: multipart/mixed; boundary=unique-boundary-1

--unique-boundary-1

Content-Type: application/sdp
Content-Length: 147

v=0
o=UserA 2890844526 2890844526 IN IP4 here.com
s=Session SDP
c=IN IP4 pc33.atlanta.com
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

--unique-boundary-1
Content-Type: multipart/signed;
  protocol="application/pkcs7-signature";
  micalg=sha1; boundary=boundary42
Content-Length: 608

--boundary42
Content-Type: message/sipfrag
Content-Disposition: auth-id; handling=optional
```

```
From: Alice <sip:alice@atlanta.com>
To: Bob <sip:bob@biloxi.com>
Contact: <sip:alice@pc33.atlanta.com>
Date: Thu, 21 Feb 2002 13:02:03 GMT
Call-ID: a84b4c76e66710
Cseq: 314159 INVITE

--boundary42
Content-Type: application/pkcs7-signature; name=smime.p7s
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename=smime.p7s;
   handling=required

ghyHhHUujhJhjH77n8HHGTrfvbnj756tbB9HG4VQpfyF467GhIGfHfYT6
4VQpfyF467GhIGfHfYT6jH77n8HHGghyHhHUujhJh756tbB9HGTrfvbnj
n8HHGTrfvhJhjH776tbB9HG4VQbnj7567GhIGfHfYT6ghyHhHUujpfyF4
7GhIGfHfYT64VQbnj756

--boundary42--

--unique-boundary-1--
```

6. Receiving an Authenticated Identity

There are a number of ways that identity information can be presented in a SIP request, and all of them must be reconcilable - that is, there must be a way to arrive at the identity that should be displayed to the user as the caller's identity, and so forth.

It is therefore RECOMMENDED in this document that these forms of identity be reduced into two broad categories: suspect and valid identities. All of the following SHOULD be considered suspect by recipients:

- Recipients may receive a normal From header field in the SIP message - unsigned and unverified. All SIP requests must contain such a header, but in some cases it may not purport to contain a usable value (it may assert an anonymous identity), and no other identity might be asserted by the request.

- Recipients may receive a From header in an authenticated identity body that was signed by a self-signed certificate that is unrecognized and/or untrusted by the user, or signed by an authentication service using a certificate authority that the user cannot verify.

- Recipients may receive a From header in an authenticated identity body that has been signed by an authentication service with a valid certificate, but which has internal consistency problems: the hostname asserted by the certificate may not correspond to the domain in the From header, the Date may be obviously stale, the Call-ID may be a repeat of a recently received value, or mandatory headers may be missing from the authenticated identity body.

Recipients may also unknowingly receive From headers in encrypted bodies which they cannot decrypt (see Section 8.2) that, of course, cannot be usable identities.

The following are identities that SHOULD be considered valid by a recipient:

- Recipients may receive a From header in an authenticated identity body that has been signed by a self-signed certificate that is recognized and trusted by the recipient.

- Recipients may receive a From header in an authenticated identity body that has been signed by an end-user certificate issued by a certificate authority that is recognized and trusted by the recipient.

- Recipients may receive a From header in an authenticated identity body that has been signed by an authentication service that properly follows the practices described in Section 4.

The exact behavior that is followed on receipt of a suspect or valid identity varies with the role of the recipient.

6.1 Proxy Server Handling

Proxy servers generally should not attempt to inspect MIME bodies. However, intermediaries that implement the authentication service logical role MAY inspect MIME bodies in order to find one with a Content-Disposition of 'auth-id'.

For the most part, the actual value of an authenticated identity is not likely to be of interest to a proxy server, though it MAY refuse to process a request that does not contain a valid authenticated identity body (using the 428 request, as described in Section 5.3). However, proxy servers MAY additionally maintain lists of known problem users that are banned from making requests, for example, and subsequently reject some requests after comparing their authenticated identities to this list.

6.2 User Agent Handling

A user agent needs to determine which identity for the originator of a request should be displayed to the user, perhaps as a 'Caller-ID' function. The following is a RECOMMENDED set of precedence rules for arriving at a single identity that should be displayed.

If one valid form of identity is present, the user agent displays that identity. If both valid forms of identity are present, the authenticated identity (rather than a recognized self-signed S/MIME signature) is preferred, but both potentially are viewable by a user.

If neither of the valid forms of identity are available, the user agent displays the normal From field in the SIP message, but other identities are viewable by a user. However, if that From field would display an anonymous identity, the user agent SHOULD display another value instead (probably an identity in a signed S/MIME body).

When it displays an identity to its user, a user agent SHOULD also have some way of designating between a valid and suspect identity that is easy for the user to distinguish.

Note that user agents also need to determine the identity of the originator of a request for the purposes of per-user blocking or screening before the user is alerted and any identity is displayed.

Generally, if any of the asserted identities in a request match an identity that is blocked, the user should not be alerted and the request SHOULD be rejected.

7. Identity in Responses

Many of the practices described in the preceding sections can be applied to responses as well as requests, with some important differences. Primarily, the distinction is that a response cannot be challenged or resubmitted in the same manner as a request. However, when a user agent registers under a particular identity, and thereby becomes eligible to receive requests and send responses associated with that identity, it provides credentials that prove its identity, and thus the registrar is in a reasonable position to act as an authentication service for responses.

An authentication service that acts as a registrar can add to a response an authenticated identity body that corresponds to the identity of the originator of that response in roughly the same manner described in Section 5.2 - the authentication service adds the authenticated identity body to a response before it forwards the response towards the originator of the request. There is no way for an authentication service to perform a function for responses comparable to the mechanism described in Section 5.3.

The same rules for the creation of the authentication identity body for requests given in Section 5.1 apply to responses, including the mandatory and optional inclusion of various headers in 'message/sipfrag' bodies, with the following exception - when the authentication service creates the authenticated identity body, it should substitute the actual identity of the user (derived, as described in Section 4.2, from the username and realm for which the user has registered) for any conflicting value in the To header field of the response before signing the response.

When the originating user agent of a request receives a response containing an authenticated identity body, it SHOULD compare the identity in the To header field of the authenticated identity body of the response with the original value of the To header field in the request. If these represent different identities, the user agent SHOULD render the identity in the authenticated identity body of the response to its user. Note that a discrepancy in these identity fields is not necessary an indication of a security breach; normal retargeting may simply have directed the request to a different final destination. User agents might furthermore indicate that this identity is suspect or valid in accordance with the guidelines given in Section 6.

8. Selective Sharing of Identity

Most of the time, there is no need to restrict the propagation of verified identities in the network. User agents and intermediaries benefit from receiving verified identities. However, in some cases intermediaries wish to restrict the distribution of identity information, for example if

- the authenticated identity body contains an identity that is only meaningful as an internal identifier within a particular service provider's network, or,

- the originating user agent has requested privacy, and the unregulated distribution of the authenticated identity body would violate that request.

If it is not appropriate to share an authenticated identity, an authenticated identity body SHOULD NOT be created and distributed. However, in some cases there may be other entities in the administrative domain of the authentication service that are consumers of the authenticated identity. If, for example, each of these servers needed to challenge the user individually for identity, it might significantly delay the processing of the request. For that reason, it may be appropriate to circulate authenticated identity bodies among a controlled set of entities. For that purpose, an encryption mechanism for authenticated identities is provided.

8.1 Requesting Privacy

When users provide credentials to an authentication service, they MAY explicitly notify the service that they do not wish their authenticated identity to be circulated. Usually, the user in question would also be taking other steps to preserve their privacy (perhaps by including an anonymous From header).

Therefore, authentication services MUST support the privacy mechanisms described in [3]. Users requesting privacy should also support the mechanisms described in that document.

In particular, this document uses an identity-specific priv-value that can appear in the Privacy header, a value of 'id'. This Privacy value requests that the results of authentication should not be shared by the authenticating server. An authentication service SHOULD NOT create an authenticated identity body for a request when 'id' privacy has been requested. If such a body is created, it MUST be encrypted.

8.2 Encryption of Identity

Many SIP entities that support the use of S/MIME for signatures will also support S/MIME encryption. Encryption of a body prevents any parties other those that hold the decryption key from inspecting the body. Note that the key used for encryption SHOULD be unrelated to the public key in a certificate that is used by an authentication service to prove its identity.

While encryption of an authenticated identity body entails that only the holder of a specific key can decrypt the body, that single key could be distributed throughout a network of hosts that exist under common policies. The security of the body is therefore predicated on the secure distribution of the key. However, for some networks (in which there are federations of trusted hosts under a common policy), the widespread distribution of a decryption key could be appropriate. Some telephone networks, for example, might require this model.

When an authenticated identity is encrypted, the authenticated identity body SHOULD always be encrypted before it is signed. Note that this means that the recipients of the request, even if they are unable to inspect the authenticated identity body, will still be able to see which authentication service signed that body (although it will not necessarily be obvious that the body contains an authenticated identity). An example of a signed and encrypted authenticated identity MIME body follows:

8.3 Example of Encryption

The following is an example of an encrypted and signed authenticated identity body (without any of the preceding SIP headers). In a rendition of this body sent over the wire, the text wrapped in asterisks would be encrypted.

```
   Content-Type: multipart/signed;
     protocol="application/pkcs7-signature";
     micalg=sha1; boundary=boundary42
   Content-Length: 568

   --boundary42

   Content-Type: application/pkcs7-mime; smime-type=enveloped-data;
     name=smime.p7m
   Content-Transfer-Encoding: base64
   Content-Disposition: attachment; filename=smime.p7m
     handling=required
   Content-Length: 231

   ************************************************************
   * Content-Type: message/sipfrag                            *
   * Content-Disposition: auth-id; handling=optional          *
   *                                                          *
   * From: sip:alice@atlanta.com                              *
   * Call-ID: a84b4c76e66710                                  *
   * Date: Thu, 21 Feb 2002 13:02:03 GMT                     *
   ************************************************************

   --boundary42

   Content-Type: application/pkcs7-signature; name=smime.p7s
   Content-Transfer-Encoding: base64
   Content-Disposition: attachment; filename=smime.p7s;
      handling=required

   ghyHhHUujhJhjH77n8HHGTrfvbnj756tbB9HG4VQpfyF467GhIGfHfYT6
   4VQpfyF467GhIGfHfYT6jH77n8HHGghyHhHUujhJh756tbB9HGTrfvbnj
   n8HHGTrfvhJhjH776tbB9HG4VQbnj7567GhIGfHfYT6ghyHhHUujpfyF4
   7GhIGfHfYT64VQbnj756

   --boundary42--
```

9. Security Considerations

Users SHOULD NOT provide credentials to an authentication service to which they cannot initiate a direct connection, preferably one secured by a network or transport layer security protocol such as TLS. If a user does not receive a certificate from the authentication service over this lower-layer protocol that corresponds to the realm in a challenge, then it is possible that a rogue server is attempting to pose as a authentication service for a realm that it does not control, possibly in an attempt to collect valid user passwords for that realm.

If a user cannot connect directly to the desired authentication service, the user SHOULD at least use a SIPS URI to ensure that mutual TLS will be used to reach the remote server.

The certificates that are required to operate an authentication service need to assert only the hostname of the authentication service, and for that reason, existing certificate authorities could provide adequate certificates for this mechanism. However, not all proxy servers and user agents will be able support the root certificates of all certificate authorities, and moreover there are some significant differences in the policies by which certificate authorities issue their certificates. This document makes no recommendations for the usage of particular certificate authorities, nor does it describe any particular policies that certificate authorities should follow, but it is anticipated that operational experience will create de facto standards for the purposes of authentication services. Some federations of service providers, for example, might only trust certificates that have been provided by a certificate authority operated by the federation.

10. IANA Considerations

This document defines a new MIME Content-Disposition disposition-type value of 'auth-id'. This value is reserved for MIME bodies that contain an authenticated identity, as described in section Section 5.1.

This document also defines a new SIP status code, 428 Use Authenticated Identity. The use of this status code is further described below in Section 5.3.

Finally, this document also uses a new priv-value for the Privacy header specified in [3], the token 'id'. This is further described in Section 8.1.

References

[1] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and E. Schooler, "SIP: Session Initiation Protocol", draft-ietf-sip-rfc2543bis-09 (work in progress), February 2002.

[2] Bradner, S., "Key words for use in RFCs to indicate requirement levels", RFC 2119, March 1997.

[3] Peterson, J., "A Privacy Mechanism for the Session Initiation Protocol (SIP)", draft-peterson-sip-privacy-00 (work in progress), April 2002.

[4] Sparks, R., "Internet Media Types message/sip and message/sipfrag", draft-sparks-sip-mimetypes-01 (work in progress), March 2002.

Author's Address

Jon Peterson
NeuStar, Inc.
1800 Sutter St
Suite 570
Concord, CA 94520
US

Phone: +1 925/363-8720
EMail: jon.peterson@neustar.biz
URI: http://www.neustar.biz/

Appendix A. Acknowledgments

The authors would like to thank Eric Rescorla, Rohan Mahy, Robert Sparks, Jonathan Rosenberg, Mark Watson and Patrik Faltstrom for their comments.

Appendix B. To Do

S/MIME authentication: If the authentication between the client and server is performed with S/MIME, possibly using a shared secret, a number of optimizations could be realized for this mechanism (essentially, the client could provide a version of the token that it asks the server to sign and/or encrypt).

Priority of encryption/signing: When privacy is requested, should an auth service encrypt then sign, or sign then encrypt? In the former case, you may lose some integrity protection. In the latter case, the certificate of the authentication service is associated with the message. Need more analysis - sign then encrypt may be preferable.

Identifying a canonical auth service: A lot of resistance was offered to the concept of a hostname convention for authentication services. However, there must be some way for a recipient of an authenticated identity body to know that it was generated by a (the?) canonical authentication service of a particular administrative domain. How can this be communicated? Perhaps with a certificate attribute?

Assertion roles: Do we need a way to tie an authenticated identity body to a particular form of identity (called, calling, forwarding, referring, etc)? Currently, an authenticated identity body represents the identity of the originator of the message.

Appendix C. Changelog

Changes from draft-peterson-sip-identity-00:

- Added a section on authenticated identities in responses

- Removed hostname convention for authentication services

- Added text about using 'message/sip' or 'message/sipfrag' in authenticated identity bodies, also RECOMMENDED a few more headers in sipfrags to increase reference integrity

- Various other editorial corrections

Full Copyright Statement

Acknowledgement

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