Network Working Group
Request for Comments: 5361
Category: Standards Track
October 2008

G. Camarillo
Ericsson

A Document Format for Requesting Consent

Status of This Memo

Abstract

This document defines an Extensible Markup Language (XML) format for a permission document used to request consent. A permission document written in this format is used by a relay to request a specific recipient permission to perform a particular routing translation.

Table of Contents

1. Introduction

The framework for consent-based communications in the Session Initiation Protocol (SIP) [RFC5360] identifies the need for a format to create permission documents. Such permission documents are used by SIP [RFC3261] relays to request permission to perform translations. A relay is defined as any SIP server, be it a proxy, B2BUA (Back-to-Back User Agent), or some hybrid, which receives a request and translates the Request-URI into one or more next-hop URIs to which it then delivers a request.

The format for permission documents specified in this document is based on Common Policy [RFC4745], an XML document format for expressing privacy preferences.

2. Definitions and Terminology

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 [RFC2119].

This document uses the terms defined in [RFC5360]. For completeness, these terms are repeated here. Figure 1 of [RFC5360] shows the relationship between target and recipient URIs in a translation operation.

Recipient URI:

The Request-URI of an outgoing request sent by an entity (e.g., a user agent or a proxy). The sending of such request can have been the result of a translation operation.

Relay:

Any SIP server, be it a proxy, B2BUA (Back-to-Back User Agent), or some hybrid, that receives a request, translates its Request-URI into one or more next-hop URIs (i.e., recipient URIs), and delivers the request to those URIs.

Target URI:

The Request-URI of an incoming request that arrives to a relay that will perform a translation operation.

Translation logic:

The logic that defines a translation operation at a relay. This logic includes the translation’s target and recipient URIs.

Translation operation:

Operation by which a relay translates the Request-URI of an incoming request (i.e., the target URI) into one or more URIs (i.e., recipient URIs) that are used as the Request-URIs of one or more outgoing requests.

3. Permission Document Structure

A permission document is an XML document, formatted according to the schema defined in [RFC4745]. Permission documents inherit the MIME type of common policy documents, 'application/auth-policy+xml'. As described in [RFC4745], this type of document is composed of three parts: conditions, actions, and transformations.

This section defines the new conditions and actions defined by this specification. This specification does not define any new transformation.

3.1. Conditions

The conditions in a permission document are a set of expressions, each of which evaluates to either TRUE or FALSE. Note that, as discussed in [RFC4745], a permission document applies to a translation if all the expressions in its conditions part evaluate to TRUE.

3.1.1. Recipient Condition

The recipient condition is matched against the recipient URI of a translation. Recipient conditions can contain the same elements and attributes as identity conditions.

When performing a translation, a relay matches the recipient condition of the permission document that was used to request permission for that translation against the destination URI of the outgoing request. When receiving a request granting or denying permissions (e.g., a SIP PUBLISH request as described in [RFC5360]), the relay matches the recipient condition of the permission document that was used to request permission against the identity of the entity granting or denying permissions (i.e., the sender of the PUBLISH request). If there is a match, the recipient condition evaluates to TRUE. Otherwise, the recipient condition evaluates to FALSE.

Since only authenticated identities can be matched, this section defines acceptable means of authentication, which are in line with those described in Section 5.6.1 of [RFC5360].

The ’id’ attribute in the elements <one> and <except> MUST contain a scheme when these elements appear in a permission document.

When used with SIP, a recipient granting or denying a relay permissions is considered authenticated if one of the following techniques is used:

SIP Identity [RFC4474], as described in Section 5.6.1.1 of [RFC5360]. For PUBLISH requests that are authenticated using the SIP Identity mechanism, the identity of the sender of the PUBLISH request is equal to the SIP URI in the From header field of the request, assuming that the signature in the Identity header field has been validated.

P-Asserted-Identity [RFC3325] (which can only be used in closed network environments) as described in Section 5.6.1.2 of [RFC5360]. For PUBLISH requests that are authenticated using the P-Asserted-Identity mechanism, the identity of the sender of the PUBLISH request is equal to the P-Asserted-Identity header field of the request.

Return Routability Test, as described in Section 5.6.1.3 of [RFC5360]. It can be used for SIP PUBLISH and HTTP GET requests. No authentication is expected to be used with return routability tests and, therefore, no identity matching procedures are defined.

SIP digest, as described in Section 5.6.1.4 of [RFC5360]. The identity of the sender is set equal to the SIP Address of Record (AOR) for the user that has authenticated themselves.

3.1.2. Identity Condition

The identity condition, which is defined in [RFC4745], is matched against the URI of the sender of the request that is used as input for a translation.

When performing a translation, a relay matches the identity condition against the identity of the sender of the incoming request. If they match, the identity condition evaluates to TRUE. Otherwise, the identity condition evaluates to FALSE.

Since only authenticated identities can be matched, the following subsections define acceptable means of authentication, the procedure for representing the identity of the sender as a URI, and the procedure for converting an identifier of the form user@domain, present in the ’id’ attribute of the <one> and <except> elements, into a URI.

3.1.2.1. Acceptable Means of Authentication

When used with SIP, a request sent by a sender is considered authenticated if one of the following techniques is used:

SIP Digest: the relay authenticates the sender using SIP digest authentication [RFC2617]. However, if the anonymous authentication described on page 194 of [RFC3261] is used, the sender is not considered authenticated.

Asserted Identity: if a request contains a P-Asserted-ID header field [RFC3325] and the request is coming from a trusted element, the sender is considered authenticated.

Cryptographically Verified Identity: if a request contains an Identity header field as defined in [RFC4474], and it validates the From header field of the request, the request is considered to be authenticated. Note that this is true even if the request contained a From header field of the form sip:anonymous@example.com. As long as the signature verifies that the request legitimately came from this identity, it is considered authenticated.

3.1.2.2. Computing a URI for the Sender

For requests that are authenticated using SIP Digest, the identity of the sender is set equal to the SIP Address of Record (AOR) for the user that has authenticated themselves. For example, consider the following "user record" in a database:

```
SIP AOR: sip:alice@example.com
digest username: ali
digest password: f779ajvvh8a6s6
digest realm: example.com
```

If the relay receives a request and challenges it with the realm set to "example.com", and the subsequent request contains an Authorization header field with a username of "ali" and a digest response generated with the password "f779ajvvh8a6s6", the identity used in matching operations is "sip:alice@example.com".

For requests that are authenticated using [RFC3325], the identity of the sender is equal to the SIP URI in the P-Asserted-ID header field. If there are multiple values for the P-Asserted-ID header field (there can be one sip URI and one tel URI [RFC3966]), then each of them is used for the comparisons outlined in [RFC4745]; if either of them match a <one> or <except> element, it is considered a match.

For requests that are authenticated using the SIP Identity mechanism [RFC4474], identity of the sender is equal to the SIP URI in the From header field of the request, assuming that the signature in the Identity header field has been validated.

SIP also allows for anonymous requests. If a request is anonymous because the digest challenge/response used the "anonymous" username, the request is considered unauthenticated and will not match the <identity> condition. If a request is anonymous because it contains a Privacy header field [RFC3323], but still contains a P-Asserted-ID header field, the identity in the P-Asserted-ID header field is still used in the authorization computations; the fact that the request was anonymous has no impact on the identity processing. However, if the request had traversed a trust boundary and the P-Asserted-ID header field and the Privacy header field had been removed, the request will be considered unauthenticated when it arrives at the relay, and thus not match the <sender> condition. Finally, if a request contained an Identity header field that was validated, and the From header field contained a URI of the form sip:anonymous@example.com, then the sender is considered authenticated, and it will have an identity equal to sip:anonymous@example.com. Had such an identity been placed into a <one> or <except> element, there will be a match.

3.1.2.3. Computing a SIP URI from the id Attribute

If the <one> or <except> condition does not contain a scheme, conversion of the value in the 'id' attribute to a SIP URI is done trivially. If the characters in the 'id' attribute are valid characters for the user and hostpart components of the SIP URI, a 'sip:' is appended to the contents of the 'id' attribute, and the result is the SIP URI. If the characters in the 'id' attribute are not valid for the user and hostpart components of the SIP URI, conversion is not possible and, thus, the identity condition evaluates to FALSE. This happens, for example, when the user portion of the 'id' attribute contains UTF-8 characters.

3.1.3. Target Condition

The target condition is matched against the target URI of a translation. The target condition can contain the same elements and attributes as identity conditions.

When performing a translation, a relay matches the target condition against the destination of the incoming request, which is typically contained in the Request-URI. If they match, the target condition evaluates to TRUE. Otherwise, the target condition evaluates to FALSE.

3.1.4. Validity Condition

The <validity> element is not applicable to this document. Each <permission> element has an infinite lifetime and can be revoked using an independent mechanism, as described in Section 5.8 of [RFC5360]. In any case, as discussed in Section 4.1 of [RFC5360], permissions are only valid as long as the context where they were granted is valid. If present, <validity> elements MUST be ignored.

3.1.5. Sphere Condition

The <sphere> element is not applicable to this document and therefore is not used. If present, <sphere> elements MUST be ignored.

3.2. Actions

The actions in a permission document provide URIs to grant or deny permission to perform the translation described in the document.

> Note that the <trans-handling> element is not an action, as defined in Common Policy [RFC4745], but rather an informational element. Therefore, the conflict resolution mechanism does not apply to it.

Each policy rule contains at least two <trans-handling> elements; one element with a URI to grant and another with a URI to deny permission.

3.2.1. Translation Handling

The <trans-handling> provides URIs for a recipient to grant or deny the relay permission to perform a translation. The defined values are:

deny: this action tells the relay not to perform the translation.

grant: this action tells the server to perform the translation.

The ’perm-uri’ attribute in the <trans-handling> element provides a URI to grant or deny permission to perform a translation.

4. Example Document

In the following example, a client adds ’sip:bob@example.org’ to the translation whose target URI is ’sip:alices-friends@example.com’. The relay handling the translation generates the following permission document in order to ask for permission to relay requests sent to ’sip:alices-friends@example.com’ to ’sip:bob@example.org’. The target URI is 'sip:alices-friends@example.com', and the recipient URI is 'sip:bob@example.org'. The sender's identity does not play a role in this example. Therefore, the permission document does not put any restriction on potential senders.

```
  +--------+        +--------------------------------+  Permission
  |        |        |                                |   Request
  | Client |        |             Relay              |    with
  |        |        | sip:alices-friends@example.com |  Permission
  +--------+        |                                |   Document
      |             |+-------+                       |-------------+
      |             ||Transl.|                       |             |
      |Manipulation ||Logic  |                       |             |
      +------------>|+-------+                       |             |
           Add      +--------------------------------+             |
     sip:bob@example.org                                           V
                                                 +---------------------+
                                                 |                     |
                                                 |      Recipient      |
                                                 | sip:bob@example.org |
                                                 |                     |
                                                 +---------------------+
  <?xml version="1.0" encoding="UTF-8"?>
         <cp:ruleset
             xmlns="urn:ietf:params:xml:ns:consent-rules"
             xmlns:cp="urn:ietf:params:xml:ns:common-policy">
             <cp:rule id="f1">
          <cp:conditions>
              <cp:identity>
                  <cp:many/>
              </cp:identity>
              <recipient>
                  <cp:one id="sip:bob@example.org"/>
              </recipient>
              <target>
                  <cp:one id="sip:alices-friends@example.com"/>
              </target>
          </cp:conditions>
          <cp:actions>
              <trans-handling
                  perm-uri="sips:grant-1awdch5Fasddfce34@example.com"
                  >grant</trans-handling>
              <trans-handling
                  perm-uri="https://example.com/grant-1awdch5Fasddfce34"
                  >grant</trans-handling>
              <trans-handling
                  perm-uri="sips:deny-23rCsdfgvdT5sdfgye@example.com"
                  >deny</trans-handling>
              <trans-handling
                  perm-uri="https://example.com/deny-23rCsdfgvdT5sdfgye"
                  >deny</trans-handling>
          </cp:actions>
          <cp:transformations/>
      </cp:rule>
      </cp:ruleset>
```

5. XML Schema

```
<?xml version="1.0" encoding="UTF-8"?>
   <xs:schema
     targetNamespace="urn:ietf:params:xml:ns:consent-rules"
     xmlns:cr="urn:ietf:params:xml:ns:consent-rules"
     xmlns:cp="urn:ietf:params:xml:ns:common-policy"
     xmlns:xs="http://www.w3.org/2001/XMLSchema"
     elementFormDefault="qualified"
     attributeFormDefault="unqualified">

     <!-- Conditions -->
     <xs:element name="recipient" type="cp:identityType"/>
     <xs:element name="target" type="cp:identityType"/>

    <!-- Actions -->
    <xs:simpleType name="trans-values">
       <xs:restriction base="xs:string">
         <xs:enumeration value="deny"/>
         <xs:enumeration value="grant"/>
       </xs:restriction>
     </xs:simpleType>

     <xs:element name="trans-handling">
       <xs:complexType>
         <xs:simpleContent>
           <xs:extension base="trans-values">
             <xs:attribute name="perm-uri" type="xs:anyURI"
                           use="required"/>
           </xs:extension>
         </xs:simpleContent>
       </xs:complexType>
     </xs:element>

   </xs:schema>
```

6. Extensibility

This specification defines elements that do not have extension points in the "urn:ietf:params:xml:ns:consent-rules" namespace. Instance documents that utilize these element definitions SHOULD be schema valid. Applications processing instance documents with content that is not understood by the application MUST ignore that content. IETF extension documents of this specification MAY reuse the "urn:ietf:params:xml:ns:consent-rules" namespace to define new elements.

7. IANA Considerations

This section registers a new XML namespace and a new XML schema per the procedures in [RFC3688].

7.1. XML Namespace Registration

URI: urn:ietf:params:xml:ns:consent-rules

Registrant Contact: IETF SIPPING working group <sipping@ietf.org>,
Gonzalo Camarillo <Gonzalo.Camarillo@ericsson.com>

XML:

```
BEGIN
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
  "http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="content-type"
        content="text/html;charset=iso-8859-1"/>
  <title>Consent Rules Namespace</title>
</head>
<body>
  <h1>Namespace for Permission Documents</h1>
  <h2>urn:ietf:params:xml:ns:consent-rules</h2>
<p>See <a href="http://www.rfc-editor.org/rfc/rfc5361.txt">RFC 5361
  </a>.</p>
</body>
</html>
END
```

7.2. XML Schema Registration

URI: urn:ietf:params:xml:schema:consent-rules

Registrant Contact: IETF SIPPING working group <sipping@ietf.org>,
Gonzalo Camarillo <Gonzalo.Camarillo@ericsson.com>

XML: The XML schema to be registered is contained in Section 5.

8. Security Considerations

RFC 5360 [RFC5360] discusses security-related issues, such as how to authenticate SIP and HTTP requests granting permissions and how to transport permission documents between relays and recipients, that are directly related to this specification.

9. Acknowledgements

Jonathan Rosenberg provided useful ideas on this document. Hannes Tschofenig helped align this document with common policy. Ben Campbell and Mary Barnes performed a thorough review of this document. Lakshminath Dondeti provided useful comments.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2617] Franks, J., Hallam-Baker, P., Hostetler, J., Lawrence, S., Leach, P., Luotonen, A., and L. Stewart, "HTTP Authentication: Basic and Digest Access Authentication", RFC 2617, June 1999.

[RFC3261] 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.

[RFC3323] Peterson, J., "A Privacy Mechanism for the Session Initiation Protocol (SIP)", RFC 3323, November 2002.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.

[RFC4474] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, August 2006.

[RFC4745] Schulzrinne, H., Tschofenig, H., Morris, J., Cuellar, J., Polk, J., and J. Rosenberg, "Common Policy: A Document Format for Expressing Privacy Preferences", RFC 4745, February 2007.

[RFC5360] Rosenberg, J., Camarillo, G., and D. Willis, "A Framework for Consent-Based Communications in the Session Initiation Protocol (SIP)", RFC 5360, October 2008.

10.2. Informative References

[RFC3966] Schulzrinne, H., "The tel URI for Telephone Numbers", RFC 3966, December 2004.

[RFC3325] Jennings, C., Peterson, J., and M. Watson, "Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks", RFC 3325, November 2002.

Author’s Address

Gonzalo Camarillo
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

EMail: Gonzalo.Camarillo@ericsson.com

Full Copyright Statement

Intellectual Property