RAP Working Group
Internet Draft

Document: draft-ietf-rap-rsvp-authsession-03.txt

L-N. Hamer
B. Gage
M. Broda
Nortel Networks
B. Kosinski
University of Alberta
Hugh Shieh
AT&T Wireless

June 2002

Session Authorization for RSVP

Status of this Memo

Copyright Notice

Abstract

This document describes the representation of session authorization information in the POLICY_DATA object (RFC 2750) for supporting policy-based per-session authorization and admission control in RSVP. The goal of session authorization is to allow the exchange of information between network elements in order to authorize the use of resources for a service and to co-ordinate actions between the signaling and transport planes. This document describes how a process on a system authorizes the reservation of resources by a host and then provides that host with a session authorization policy element which can be inserted into the RSVP PATH message to facilitate proper and secure reservation of those resources within the network. We describe the encoding of media authorization

information as RSVP policy elements and provide details relating to operations, processing rules and error scenarios.

Contents

1. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC-2119].

2. Introduction

RSVP [RFC-2205] is a resource reservation setup protocol designed for an integrated services [RFC-1633] or Integrated Services over Diffserv networks [RFC-2998]. The RSVP protocol is used by a host to request specific services from the network for particular application data streams or flows. RSVP is also used to deliver quality-of-service (QoS) requests to all routers along the path(s) of the flows and to establish and maintain state to provide the requested quality of service. RSVP requests will generally result in resources being reserved in each router along the data path. RSVP allows users to obtain preferential access to network resources, under the control of an admission control mechanism. Such admission control is often based on user or application identity [RFC-3182], however, it is also valuable to provide the ability for per-session admission control.

In order to allow for per-session admission control, it is necessary to provide a mechanism for ensuring use of resources by a host has been properly authorized before allowing the reservation of those resources. In order to meet this requirement, there must be information in the RSVP message which may be used to verify the validity of the RSVP request. This can be done by providing the host with a token upon authorization which is inserted into the RSVP PATH message and verified by the network.

This document describes the session authorization element (AUTH_SESSION) contained in the POLICY_DATA object. The user process must obtain an AUTH_SESSION object from an authorizing entity, which it then passes to the RSVP process (service) on the originating host. The RSVP service then inserts the AUTH_SESSION object into the RSVP PATH message to allow verification of the network resource request. Network elements verify the request and then process the RSVP message based on admission policy.

[S-AUTH] describes a framework in which a session authorization policy element may be utilized to contain information relevant to the network's decision to grant a reservation request.

3. Policy Element for Session Authorization Data

3.1 Policy Data Object Format

POLICY_DATA objects contain policy information and are carried by RSVP messages. A detailed description of the format of POLICY_DATA object can be found in "RSVP Extensions for Policy Control" [RFC-2750].

3.2 Session Authorization Data Policy Element

In this section we describe a policy element (PE) called session authorization data (AUTH_SESSION). The AUTH_SESSION policy element contains a list of fields which describe the session, along with other attributes.

```
+-------------+-------------+-------------+-------------+
| Length                    | P-Type = AUTH_SESSION     |
+-------------+-------------+-------------+-------------+
// Session Authorization Attribute List               //
+-------------------------------------------------------+
```

Length: 16 bits
The length of the policy element (including the Length and P-Type) is in number of octets (MUST be in multiples of 4) and indicates the end of the session authorization information block.

P-Type: 16 bits (Session Authorization Type)
AUTH_SESSION = TBD-by-IANA
The Policy element type (P-type) of this element. The Internet Assigned Numbers Authority (IANA) acts as a registry for policy element types for identity as described in [RFC-2750].

Session Authorization Attribute List: variable length
The session authorization attribute list is a collection of objects which describes the session and provides other information necessary to verify the RSVP request. An initial set of valid objects is described in Section 3.

3.3 Session Authorization Attributes

A session authorization attribute may contain a variety of information and has both an attribute type and subtype. The attribute itself MUST be a multiple of 4 octets in length, and any attributes that are not a multiple of 4 octets long MUST be padded to a 4-octet boundary. All padding bytes MUST have a value of zero.


```
+--------+--------+--------+--------+
| Length          | S-Type |SubType |
+--------+--------+--------+--------+
| Value ...
+--------+--------+--------+--------+
```

Length: 16 bits

The length field is two octets and indicates the actual length of the attribute (including Length, S-Type and SubType fields) in number of octets. The length does NOT include any bytes padding to the value field to make the attribute a multiple of 4 octets long.

S-Type: 8 bits

Session authorization attribute type (S-Type) field is one octet. IANA acts as a registry for S-Types as described in section 7, IANA Considerations. Initially, the registry contains the following S-Types:

```
1   AUTH_ENT_ID          The unique identifier of the entity
                         which authorized the session.

2   SESSION_ID           Unique identifier for this session.

3   SOURCE_ADDR          Address specification for the
                         session originator.

4   DEST_ADDR            Address specification for the
                         session end-point.

5   START_TIME           The starting time for the session.

6   END_TIME             The end time for the session.
```

```
   7  RESOURCES            The resources which the user is
                           authorized to request.

   8  AUTHENTICATION_DATA  Authentication data of the session
                           authorization policy element.
```

SubType: 8 bits
Session authorization attribute sub-type is one octet in length. The value of the SubType depends on the S-Type.

Value: variable length
The attribute specific information.

3.3.1 Authorizing Entity Identifier

AUTH_ENT_ID is used to identify the entity which authorized the initial service request and generated the session authorization policy element. The AUTH_ENT_ID may be represented in various formats, and the SubType is used to define the format for the ID. The format for AUTH_ENT_ID is as follows:

```
+-------+-------+-------+-------+
| Length        |S-Type |SubType|
+-------+-------+-------+-------+
| OctetString ...
+-------+-------+-------+-------+
```

Length
Length of the attribute, which MUST be > 4.

S-Type
AUTH_ENT_ID

SubType
The following sub-types for AUTH_ENT_ID are defined. IANA acts as a registry for AUTH_ENT_ID sub-types as described in section 7, IANA Considerations. Initially, the registry contains the following sub-types of AUTH_ENT_ID:

```
1  IPV4_ADDRESS        IPv4 address represented in 32 bits

2  IPV6_ADDRESS        IPv6 address represented in 128 bits
```

```
   3  FQDN                    Fully Qualified Domain Name as defined
                              in RFC-1034 as an ASCII string.

   4  ASCII_DN                X.500 Distinguished name as defined
                              in RFC-2253 as an ASCII string.

   5  UNICODE_DN              X.500 Distinguished name as defined
                              in RFC-2253 as a UNICODE string.

   6  URI                     Universal Resource Identifier, as
                              defined in RFC-2396.

   7  KRB_PRINCIPAL          Fully Qualified Kerberos Principal name
                             represented by the ASCII string of a
                             principal followed by the @ realm name as
                             defined in RFC-1510 (e.g.
                             principalX@realmY).

   8  X509_V3_CERT            A chain of authorizing entity's X.509 V3
                              digital certificates.
```



```
   9  PGP_CERT                The PGP digital certificate of the
                              authorizing entity.
```

OctetString
   Contains the authorizing entity identifier.

3.3.2 Session Identifier

SESSION_ID is a unique identifier used by the authorizing entity to identify the request. It may be used for a number of purposes, including replay detection, or to correlate this request to a policy decision entry made by the authorizing entity. For example, the SESSION_ID can be based on simple sequence number or on a standard NTP timestamp.

```
+-------+-------+-------+-------+
| Length        |S-Type |SubType|
+-------+-------+-------+-------+
| OctetString ...
+-------+-------+-------+-------+
```

Length
   Length of the attribute, which MUST be > 4.

S-Type
   SESSION_ID

SubType
   No subtypes for SESSION ID are currently defined; this field MUST be set to zero. The authorizing entity is the only network entity that needs to interpret the contents of the SESSION ID therefore the contents and format are implementation dependent.

OctetString
   Contains the session identifier.

3.3.3 Source Address

SOURCE_ADDR is used to identify the source address specification of the authorized session. This S-Type may be useful in some scenarios to make sure the resource request has been authorized for that particular source address and/or port.

```
+-------+-------+-------+-------+
| Length        |S-Type |SubType|
+-------+-------+-------+-------+
| OctetString ...
+-------+-------+-------+-------+
```


Length
   Length of the attribute, which MUST be > 4.

S-Type
   SOURCE_ADDR

SubType
   The following sub types for SOURCE_ADDR are defined. IANA acts as a registry for SOURCE_ADDR sub-types as described in section 7, IANA Considerations. Initially, the registry contains the following sub types for SOURCE_ADDR:

1	IPV4_ADDRESS	IPv4 address represented in 32 bits
2	IPV6_ADDRESS	IPv6 address represented in 128 bits
3	FQDN	Fully Qualified Domain Name as defined in RFC-1034 as an ASCII string.
4	ASCII_DN	X.500 Distinguished name as defined

in RFC-2253 as an ASCII string.

```
   5  UNICODE_DN          X.500 Distinguished name as defined
                          in RFC-2253 as a UNICODE string.

   6  UDP_PORT LIST       list of UDP port specifications,
                          represented as 16 bits per list entry.

   7  TCP_PORT LIST       list of TCP port specifications,
                          represented as 16 bits per list entry.
```

OctetString
The OctetString contains the source address information.

In scenarios where a source address is required (see Section 5), at least one of the subtypes 1 through 5 (inclusive) MUST be included in every Session Authorization Data Policy Element. Multiple SOURCE ADDR attributes MAY be included if multiple addresses have been authorized. The source address field of the RSVP datagram MUST match one of the SOURCE ADDR attributes contained in this Session Authorization Data Policy Element when resolved to an IP address.

At most, one instance of subtype 6 MAY be included in every Session Authorization Data Policy Element. At most, one instance of subtype 7 MAY be included in every Session Authorization Data Policy Element. Inclusion of a subtype 6 attribute does not prevent inclusion of a subtype 7 attribute (i.e. both UDP and TCP ports may be authorized).

If no PORT attributes are specified, then all ports are considered valid; otherwise, only the specified ports are authorized for use.

Every source address and port list must be included in a separate SOURCE_ADDR attribute.

3.3.4 Destination Address

DEST_ADDR is used to identify the destination address of the authorized session. This S-Type may be useful in some scenarios to make sure the resource request has been authorized for that particular destination address and/or port.

```
   +-------+-------+-------+-------+
   | Length        |S-Type |SubType|
   +-------+-------+-------+-------+
```

```
   | OctetString ...
   +-------+-------+-------+-------+
```

Length
   Length of the attribute, which MUST be > 4.

S-Type
   DEST_ADDR

SubType
   The following sub types for DEST_ADDR are defined. IANA acts as a registry for DEST_ADDR sub-types as described in section 7, IANA Considerations. Initially, the registry contains the following sub types for DEST_ADDR:

```
   1  IPV4_ADDRESS        IPv4 address represented in 32 bits

   2  IPV6_ADDRESS        IPv6 address represented in 128 bits

   3  FQDN                Fully Qualified Domain Name as defined
                          in RFC-1034 as an ASCII string.

   4  ASCII_DN            X.500 Distinguished name as defined
                          in RFC-2253 as an ASCII string.

   5  UNICODE_DN          X.500 Distinguished name as defined
                          in RFC-2253 as a UNICODE string.

   6  UDP_PORT LIST       list of UDP port specifications,
                          represented as 16 bits per list entry.

   7  TCP_PORT LIST       list of TCP port specifications,
                          represented as 16 bits per list entry.
```

OctetString
   The OctetString contains the destination address specification.

In scenarios where a destination address is required (see Section 5), at least one of the subtypes 1 through 5 (inclusive) MUST be included in every Session Authorization Data Policy Element. Multiple DEST ADDR attributes MAY be included if multiple addresses have been authorized. The destination address field of the RSVP datagram MUST match one of the DEST ADDR attributes contained in this Session Authorization Data Policy Element when resolved to an IP address.

At most, one instance of subtype 6 MAY be included in every Session Authorization Data Policy Element. At most, one instance of subtype 7 MAY be included in every Session Authorization Data Policy Element. Inclusion of a subtype 6 attribute does not prevent inclusion of a subtype 7 attribute (i.e. both UDP and TCP ports may be authorized).

If no PORT attributes are specified, then all ports are considered valid; otherwise, only the specified ports are authorized for use.

Every destination address and port list must be included in a separate DEST_ADDR attribute.

3.3.5 Start time

START_TIME is used to identify the start time of the authorized Session and can be used to prevent replay attacks. If the AUTH_SESSION policy element is presented in a resource request, the network SHOULD reject the request if it is not received within a few seconds of the start time specified.

```
+-------+-------+-------+-------+
| Length        |S-Type |SubType|
+-------+-------+-------+-------+
| OctetString ...
+-------+-------+-------+-------+
```

Length
   Length of the attribute, which MUST be > 4.

S-Type
   START_TIME

SubType
   The following sub types for START_TIME are defined. IANA acts as a registry for START_TIME sub-types as described in section 7, IANA Considerations. Initially, the registry contains the following sub types for START_TIME:

```
1  NTP_TIMESTAMP        NTP Timestamp Format as defined in
                        RFC-1305.
```


OctetString
   The OctetString contains the start time.

3.3.6 End time

END_TIME is used to identify the end time of the authorized session and can be used to limit the amount of time that resources are authorized for use (e.g. in prepaid session scenarios).

```
+-------+-------+-------+-------+
| Length        |S-Type |SubType|
+-------+-------+-------+-------+
| OctetString ...
+-------+-------+-------+-------+
```

Length
Length of the attribute, which MUST be > 4.

S-Type
END_TIME

SubType
The following sub types for END_TIME are defined. IANA acts as a registry for END_TIME sub-types as described in section 7, IANA Considerations. Initially, the registry contains the following sub types for END_TIME:

1 NTP_TIMESTAMP NTP Timestamp Format as defined in RFC-1305.

OctetString
The OctetString contains the end time.

3.3.7 Resources Authorized

RESOURCES is used to define the characteristics of the authorized session. This S-Type may be useful in some scenarios to specify the specific resources authorized to ensure the request fits the authorized specifications.

```
+-------+-------+-------+-------+
| Length        |S-Type |SubType|
+-------+-------+-------+-------+
| OctetString ...
+-------+-------+-------+-------+
```

Length
Length of the attribute, which MUST be > 4.

S-Type

RESOURCES

SubType
The following sub-types for RESOURCES are defined. IANA acts as a registry for RESOURCES sub-types as described in section 7, IANA Considerations. Initially, the registry contains the following sub types for RESOURCES:

```
1  BANDWIDTH     Maximum bandwidth (kbps) authorized.

2  FLOW_SPEC     Flow spec specification as defined in
                 RFC-2205.

3  SDP           SDP Media Descriptor as defined in
                 RFC-2327.

4  DSCP          Differentiated services codepoint as
                 defined in RFC-2474.
```

OctetString
The OctetString contains the resources specification.

In scenarios where a resource specification is required (see Section 5), at least one of the subtypes 1 through 4 (inclusive) MUST be included in every Session Authorization Data Policy Element. Multiple RESOURCE attributes MAY be included if multiple types of resources have been authorized (e.g. DSCP and BANDWIDTH).

3.3.8 Authentication data

The AUTHENTICATION_DATA attribute contains the authentication data of the AUTH_SESSION policy element and signs all the data in the policy element up to the AUTHENTICATION_DATA. If the AUTHENTICATION_DATA attribute has been included in the AUTH_SESSION policy element, it MUST be the last attribute in the list. The algorithm used to compute the authentication data depends on the AUTH_ENT_ID SubType field. See Section 4 entitled Integrity of the AUTH_SESSION policy element.

A summary of AUTHENTICATION_DATA attribute format is described below.

```
+-------+-------+-------+-------+
| Length        |S-Type |SubType|
+-------+-------+-------+-------+
| OctetString ...
+-------+-------+-------+-------+
```

Length

Length of the attribute, which MUST be > 4.

S-Type
AUTHENTICATION_DATA

SubType
No sub types for AUTHENTICATION_DATA are currently defined. This field MUST be set to 0.

OctetString
OctetString contains the authentication data of the AUTH_SESSION.

4. Integrity of the AUTH_SESSION policy element

This section describes how to ensure the integrity of the policy element is preserved.

4.1 Shared private keys

In shared private key environments, the AUTH_ENT_ID MUST be of subtypes: IPV4_ADDR, IPV6_ADDR, FQDN, ASCII_DN, UNICODE_DN or URI. An example AUTH_SESSION policy element is shown below.

```
+--------------+--------------+--------------+--------------+
| Length                      | P-type = AUTH_SESSION       |
+--------------+--------------+--------------+--------------+
| Length                      |SESSION_ID    |     zero     |
+--------------+--------------+--------------+--------------+
| OctetString (The session identifier) ...
+--------------+--------------+--------------+--------------+
| Length                      |AUTH DATA.    |     zero     |
+--------------+--------------+--------------+--------------+
| OctetString (Authentication data) ...
+--------------+--------------+--------------+--------------+
```

4.1.1 Operational Setting using shared private keys

This assumes both the Authorizing Entity and the Network router/PDP are provisioned with shared private keys and with policies detailing which algorithm to be used for computing the authentication data.

Key maintenance is outside the scope of this document, but AUTH_SESSION implementations MUST at least provide the ability to manually configure keys and their parameters locally. The key used to produce the authentication data is identified by the AUTH_ENT_ID field. Each key must also be configured with lifetime parameters for the time period within which it is valid as well as an associated cryptographic algorithm parameter specifying the algorithm to be used with the key. At a minimum, all AUTH_SESSION implementations

MUST support the HMAC-MD5-96 [RFC-2104][FRC-1321] cryptographic algorithm for computing the authentication data.

It is good practice to regularly change keys. Keys MUST be configurable such that their lifetimes overlap allowing smooth transitions between keys. At the midpoint of the lifetime overlap between two keys, senders should transition from using the current key to the next/longer-lived key. Meanwhile, receivers simply accept any identified key received within its configured lifetime and reject those that are not.

4.2 Kerberos

In a Kerberos environment, the AUTH_ENT_ID MUST be of the subtype KRB_PRINCIPAL. Kerberos [RFC 1510] authentication uses a trusted third party (the Kerberos Distribution Center - KDC) to provide for authentication of the AUTH_SESSION to a network server. It is assumed that a KDC is present and both host and verifier of authentication information (authorizing entity and router/PDP) implement Kerberos authentication.

An example of the Kerberos AUTH_DATA policy element is shown below.

```
+--------------+--------------+--------------+--------------+
| Length                      | P-type = AUTH_SESSION       |
+--------------+--------------+--------------+--------------+
| Length                      |SESSION_ID    |     zero     |
+--------------+--------------+--------------+--------------+
| OctetString (The session identifier) ...
+--------------+--------------+--------------+--------------+
| Length                      | AUTH_ENT_ID  | KERB_P.      |
+--------------+--------------+--------------+--------------+
| OctetString (The principal@realm name) ...
+--------------+--------------+--------------+--------------+
```

4.2.1. Operational Setting using Kerberos

An authorizing entity is configured to construct the AUTH_SESSION policy element that designates use of the Kerberos authentication method (KRB_PRINCIPAL). Upon reception of the RSVP request, the router/PDP contacts the local KDC to request a ticket for the authorizing entity (principal@realm). The router/PDP uses the ticket to access the authorizing entity and obtain authentication data for the message.

For cases where the authorizing entity is in a different realm (i.e. administrative domain, organizational boundary), the router/PDP needs to fetch a cross-realm Ticket Granting Ticket (TGT) from its local KDC. This TGT can be used to fetch authorizing entity tickets

from the KDC in the remote realm. Note that for performance considerations, tickets are typically cached for extended periods.

4.3 Public Key

In a public key environment, the AUTH_ENT_ID MUST be of the subtypes: X509_V3_CERT or PGP_CERT. The authentication data is used for authenticating the authorizing entity. An example of the public key AUTH_SESSION policy element is shown below.

```
+--------------+--------------+--------------+--------------+
| Length                      | P-type = AUTH_SESSION       |
+--------------+--------------+--------------+--------------+
| Length                      |SESSION_ID    |     zero     |
+--------------+--------------+--------------+--------------+
| OctetString (The session identifier) ...
+--------------+--------------+--------------+--------------+
| Length                      | AUTH_ENT_ID  |   PGP_CERT   |
+--------------+--------------+--------------+--------------+
| OctetString (Authorizing entity Digital Certificate) ...
+--------------+--------------+--------------+--------------+
| Length                      |AUTH DATA.    |     zero     |
+--------------+--------------+--------------+--------------+
| OctetString (Authentication data) ...
+--------------+--------------+--------------+--------------+
```

4.3.1. Operational Setting for public key based authentication

Public key based authentication assumes following:

- Authorizing entities have a pair of keys (private key and public key).

- Private key is secured with the authorizing entity.

- Public keys are stored in digital certificates and a trusted party, certificate authority (CA) issues these digital certificates.

- The verifier (PDP or router) has the ability to verify the digital certificate.

Authorizing entity uses its private key to generate AUTHENTICATION_DATA. Authenticators (router, PDP) use the authorizing entity's public key (stored in the digital certificate) to verify and authenticate the policy element.

5. Framework

[S-AUTH] describes a framework in which the AUTH_SESSION policy element may be utilized to transport information required for authorizing resource reservation for media flows. [S-AUTH] introduces 4 different models:
1- the coupled model
2- the associated model with one policy server
3- the associated model with two policy servers
4- the non-associated model.

The fields that are required in an AUTH SESSION policy element is dependent on which of the models is used.

5.1 The coupled model

In the Coupled Model, the only information that MUST be included in the policy element is the SESSION ID; it is used by the Authorizing Entity to correlate the resource reservation request with the media authorized during session set up. Since the End Host is assumed to be untrusted, the Policy Server SHOULD take measures to ensure that the integrity of the SESSION ID is preserved in transit; the exact mechanisms to be used and the format of the SESSION ID are implementation dependent.

5.2 The associated model with one policy server

In this model, the contents of the AUTH_SESSION policy element MUST include:

- A session identifier - SESSION_ID. This is information that the authorizing entity can use to correlate the resource reservation request with the media authorized during session set up.

- The identity of the authorizing entity _ AUTH_ENT_ID. This information is used by the Edge Router to determine which authorizing entity (Policy Server) should be used to solicit resource policy decisions.

In some environments, an Edge Router may have no means for determining if the identity refers to a legitimate Policy Server within its domain. In order to protect against redirection of authorization requests to a bogus authorizing entity, the AUTH_SESSION MUST also include:

- AUTHENTICATION_DATA. This authentication data is calculated over all other fields of the AUTH_SESSION policy element.

5.3 The associated model with two policy servers

The content of the AUTH_SESSION Policy Element is identical to the associated model with one policy server.

5.4 The non-associated model

In this model, the AUTH_SESSION MUST contain sufficient information to allow the Policy Server to make resource policy decisions autonomously from the authorizing entity. The policy element is created using information about the session by the authorizing entity. The information in the AUTH_SESSION policy element MUST include:

- Calling party IP address or Identity (e.g. FQDN) - SOURCE_ADDR S-TYPE
- Called party IP address or Identity (e.g. FQDN) - DEST_ADDR S-TYPE
- The characteristics of (each of) the media stream(s) authorized for this session - RESOURCES S-TYPE
- The authorization lifetime - START_TIME S-TYPE
- The identity of the authorizing entity to allow for validation of the token in shared private key and Kerberos schemes - AUTH_ENT_ID S-TYPE
- The credentials of the authorizing entity in a public-key scheme - AUTH_ENT_ID S-TYPE
- Authentication data used to prevent tampering with the

AUTH_SESSION policy element - AUTHENTICATION_DATA

Furthermore, the AUTH_SESSION policy element MAY contain:

- The lifetime of (each of) the media stream(s) - END_TIME S-TYPE
- Calling party port number - SOURCE_ADDR S-TYPE
- Called party port number - DEST_ADDR S-TYPE

All AUTH_SESSION fields MUST match with the resource request. If a field does not match, the request SHOULD be denied.

6. Message Processing Rules

6.1 Message Generation (RSVP Host)

An RSVP message is created as specified in [RFC-2205] with following modifications.

1. RSVP message MUST contain at most one AUTH_SESSION policy element.

2. A Session Authorization policy element (AUTH_SESSION) is created and the IdentityType field is set to indicate the identity type

in the policy element. Only the required Session Authorization attributes are added.

3. POLICY_DATA object (containing the AUTH_SESSION policy element) is inserted in the RSVP message in the appropriate place.

6.2 Message Reception (Router)

RSVP message is processed as specified in [RFC-2205] with following modifications.

1. If router is policy aware then it SHOULD send the RSVP message to the PDP and wait for response. If the router is policy unaware then it ignores the policy data objects and continues processing the RSVP message.

2. Reject the message if the response from the PDP is negative.

3. Continue processing the RSVP message.

6.3 Authorization (Router/PDP)

1. Retrieve the AUTH_SESSION policy element. Check the PE type

field and return an error if the identity type is not supported.

2. Verify the message integrity.

- Shared private key authentication: Get authorizing entity ID, identify appropriate algorithm and shared private key for the authorizing entity, and validate signature.

- Public Key: Validate the certificate chain against trusted Certificate Authority (CA) and validate the message signature using the public key.

- Kerberos Ticket: If the AUTH_ENT_ID is of subtype KRB_PRINCIPAL, Request a ticket for the authorizing entity (principal@realm) from the local KDC. Use the ticket to access the authorizing entity and obtain authentication data for the message.

3. Verify the requested resources do not exceed the authorized QoS.

7. Error Signaling

If a PDP fails to verify the AUTH_SESSION policy element then it MUST return a policy control failure (Error Code = 02) to the PEP. The error values are described in [RFC-2205] and [RFC-2750]. Also the PDP SHOULD supply a policy data object containing an AUTH_DATA Policy Element with A-Type=POLICY_ERROR_CODE containing more details on the Policy Control failure [RFC-3182]. The PEP

MUST include this Policy Data object in the outgoing RSVP Error message.

8. IANA Considerations

Following the policies outlined in [IANA-CONSIDERATIONS], Standard RSVP Policy Elements (P-type values) are assigned by IETF Consensus action as described in [RFC-2750].

P-Type AUTH_SESSION is assigned the value TBD-by-IANA.

Following the policies outlined in [IANA-CONSIDERATIONS], session authorization attribute types (S-Type)in the range 0-127 are allocated through an IETF Consensus action; S-Type values between 128-255 are reserved for Private Use and are not assigned by IANA.

S-Type AUTH_ENT_ID is assigned the value 1.
S-Type SESSION_ID is assigned the value 2.

S-Type SOURCE_ADDR is assigned the value 3.
S-Type DEST_ADDR is assigned the value 4.
S-Type START_TIME is assigned the value 5.
S-Type END_TIME is assigned the value 6.
S-Type RESOURCES is assigned the value 7.
S-Type AUTHENTICATION_DATA is assigned the value 8.

Following the policies outlined in [IANA-CONSIDERATIONS], AUTH_ENT_ID SubType values in the range 0-127 are allocated through an IETF Consensus action, SubType values between 128-255 are reserved for Private Use and are not assigned by IANA.

AUTH_ENT_ID SubType IPV4_ADDRESS is assigned the value 1.
SubType IPV6_ADDRESS is assigned the value 2.
SubType FQDN is assigned the value 3.
SubType ASCII_DN is assigned the value 4.
SubType UNICODE_DN is assigned the value 5.
SubType URI is assigned the value 6.
SubType KRB_PRINCIPAL is assigned the value 7.
SubType X509_V3_CERT is assigned the value 8.
SubType PGP_CERT is assigned the value 9.

Following the policies outlined in [IANA-CONSIDERATIONS], SOURCE_ADDR SubType values in the range 0-127 are allocated through an IETF Consensus action, SubType values between 128-255 are reserved for Private Use and are not assigned by IANA.

SOURCE_ADDR SubType IPV4_ADDRESS is assigned the value 1.
SubType IPV6_ADDRESS is assigned the value 2.
SubType FQDN is assigned the value 3.
SubType ASCII_DN is assigned the value 4.
SubType UNICODE_DN is assigned the value 5.
SubType UDP_PORT_LIST is assigned the value 6.

SubType TCP_PORT_LIST is assigned the value 7.

Following the policies outlined in [IANA-CONSIDERATIONS], DEST_ADDR SubType values in the range 0-127 are allocated through an IETF Consensus action, SubType values between 128-255 are reserved for Private Use and are not assigned by IANA.

DEST_ADDR SubType IPV4_ADDRESS is assigned the value 1.
SubType IPV6_ADDRESS is assigned the value 2.
SubType FQDN is assigned the value 3.
SubType ASCII_DN is assigned the value 4.
SubType UNICODE_DN is assigned the value 5.
SubType UDP_PORT_LIST is assigned the value 6.

SubType TCP_PORT_LIST is assigned the value 7.

Following the policies outlined in [IANA-CONSIDERATIONS], START_TIME SubType values in the range 0-127 are allocated through an IETF Consensus action, SubType values between 128-255 are reserved for Private Use and are not assigned by IANA.

START_TIME SubType NTP_TIMESTAMP is assigned the value 1.

Following the policies outlined in [IANA-CONSIDERATIONS], END TIME SubType values in the range 0-127 are allocated through an IETF Consensus action, SubType values between 128-255 are reserved for Private Use and are not assigned by IANA.

END TIME SubType NTP_TIMESTAMP is assigned the value 1.

Following the policies outlined in [IANA-CONSIDERATIONS], RESOURCES SubType values in the range 0-127 are allocated through an IETF Consensus action, SubType values between 128-255 are reserved for Private Use and are not assigned by IANA.

RESOURCES SubType BANDWIDTH is assigned the value 1.
SubType FLOW_SPEC is assigned the value 2.
SubType SDP is assigned the value 3.
SubType DSCP is assigned the value 4.

9. Security Considerations

The purpose of this draft is to describe a mechanism for session authorization to prevent theft of service.

Replay attacks MUST be prevented. In the non-associated model, the AUTH_SESSION policy element MUST include a START_TIME field. The start time is used to verify that the request is not being replayed at a later time. In all other models, the SESSION_ID is used by the

Policy Server to ensure that the resource request successfully correlates with records of an authorized session. If a AUTH_SESSION is replayed, it MUST be detected by the policy server (using internal algorithms) and the request MUST be rejected.

To ensure that the integrity of the policy element is preserved in untrusted environments, the AUTHENTICATION_DATA attribute MUST be included.

In order to keep the AUTH_SESSION policy element size to a strict minimum, in environments where shared private keys are possible, they should be used. This is especially true in wireless environments where the AUTH_SESSION policy element is sent over-the-air. The shared private keys authentication option MUST be supported by all AUTH_SESSION implementations.

If shared private keys are not a valid option, the Kerberos authentication mechanism is reasonably well secured and efficient in terms of AUTH_SESSION size. The AUTH_SESSION only needs to contain the principal@realm name of the authorizing entity. This is much more efficient than the PKI authentication option.

PKI authentication option provides a high level of security and good scalability, however it requires the presence of credentials in the AUTH_SESSION policy element which impacts its size.

10. Acknowledgments

We would like to thank Louis LeVay, Francois Audet, Don Wade, Hamid Syed, Kwok Ho Chan and many others for their valuable comments.

In addition, we would like to thank S. Yadav, et al, for their efforts on RFC 3182, as this document borrows from their work.

11. Normative References

[S-AUTH] Hamer, L.-N., Gage, B., Shieh, H., "Framework for session setup with media authorization", Internet-Draft, draft-ietf-rap-session-auth-04.txt, June 2002.

[ASCII] Coded Character Set -- 7-Bit American Standard Code for Information Interchange, ANSI X3.4-1986.

[RFC-2750] Herzog, S., "RSVP Extensions for Policy Control", RFC 2750, January 2000.

[RFC-2753] Yavatkar, R., Pendarakis, D. and R. Guerin, "A Framework for Policy-based Admission Control RSVP", RFC 2753, January 2000.

[RFC-1034] Mockapetris, P.V., "Domain names - concepts and facilities", RFC 1034, November 1987.

[RFC-1305] Mills, David L., "Network Time Protocol (Version 3) Specification, Implementation, and Analysis", RFC 1305, March 1992.

[RFC-1321] Rivest, R., "The MD5 Message-Digest Algorithm",RFC 1321, April 1992.

[RFC-1510] Kohl, J. and C. Neuman, "The Kerberos Network Authentication Service (V5)", RFC 1510, September 1993.

[RFC-2104] Krawczyk, H., Bellare, M. and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997.

[RFC-2253] Wahl, M. et al., "UTF-8 String Representation of Distinguished Names", RFC 2253, December 1997.

[RFC-2205] Braden, R., Zhang, L., Berson, S., Herzog, S. and S. Jamin, "Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification", RFC 2205, September 1997.

[RFC-2209] Braden, R. and L. Zhang, "Resource ReSerVation Protocol (RSVP) - Version 1 Message Processing Rules", RFC 2209, September 1997.

[RFC-2327] Handley, M., Jacobson, V., "SDP: Session Description Protocol", RFC 2327, October 1998.

[RFC-2396] Berners-Lee, T., Fielding, R., Irvine, U.C., Masinter, L., "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998.

[RFC-2474] Nichols, K., Blake, S., Baker, F., Black, D., "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998.

[UNICODE] The Unicode Consortium, "The Unicode Standard,Version 2.0", Addison-Wesley, Reading, MA, 1996.

[X.509] Housley, R., Ford, W., Polk, W. and D. Solo, "Internet X.509 Public Key Infrastructure Certificate and CRL Profile", RFC 2459, January 1999.

[X.509-ITU] ITU-T (formerly CCITT) Information technology Open Systems Interconnection - The Directory: Authentication Framework Recommendation X.509 ISO/IEC 9594-8

12. Informative References

[RFC-3182] S. Yadav et al, "Identity Representation for RSVP", RFC 3182, October 2001

[RFC-2998] Bernet, Y., Ford, P., Yavatkar, R., Baker, F.,Zhang, L., Speer, M., Braden, R., Davie, B., Wroclawski, J., Felstaine, E., "A Framework for Integrated Services Operation over Diffserv Networks", RFC 2998, November 2000.

[RFC-1633] Braden, R., Clark, D., Shenker, S., "Integrated Services in the Internet Architecture: An Overview", RFC 1633, June 1994.

[IANA-CONSIDERATIONS] Alvestrand, H. and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

13. Author Information

Louis-Nicolas Hamer
Nortel Networks
PO Box 3511 Station C
Ottawa, Ontario
Canada K1Y 4H7
Phone: +1 613.768.3409
EMail: nhamer@nortelnetworks.com

Brett Kosinski
University of Alberta
Edmonton, Alberta
Canada T6G 2M7
EMail: kosinski@cs.ualberta.ca

Bill Gage
Nortel Networks
PO Box 3511 Station C
Ottawa, Ontario
Canada K1Y 4H7
Phone: +1 613.763.4400
EMail: gageb@nortelnetworks.com

Matt Broda
Nortel Networks
PO Box 3511 Station C
Ottawa, Ontario
Canada K1Y 4H7
Phone: +1 613.763.7399
EMail: mbroda@nortelnetworks.com

Hugh Shieh
AT&T Wireless
7277 164th Avenue NE
Redmond, WA
USA 98073-9761
Phone: +1 425.580.6898
Email: hugh.shieh@attws.com

14. Full Copyright Statement

15. Notices

16. RFC Editor Considerations

This document references an IETF Internet-Draft that is in the IESG last call stage. Please use the corresponding RFC number prior to publishing of this document as a RFC. The referenced IETF I-D is [S-AUTH].