Network Working Group
Request for Comments: 3520
Category: Standards Track

L-N. Hamer
B. Gage
Nortel Networks
B. Kosinski
Invidi Technologies
H. Shieh
AT&T Wireless

April 2003

Session Authorization Policy Element

Status of this Memo

Copyright Notice

Abstract

This document describes the representation of a session authorization policy element for supporting policy-based per-session authorization and admission control. 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 a resource reservation protocol (e.g., the Resource ReSerVation Protocol (RSVP) PATH message) to facilitate proper and secure reservation of those resources within the network. We describe the encoding of session authorization information as a policy element conforming to the format of a Policy Data object (RFC 2750) and provide details relating to operations, processing rules and error scenarios.

Table of 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 BCP 14, RFC 2119 [RFC-2119].

2. Introduction

RSVP [RFC-2205] is one example of a resource reservation protocol that is used by a host to request specific services from the network for particular application data streams or flows. 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 resource reservation message which may be used to verify the validity of the reservation request. This can be done by providing the host with a session authorization policy element which is inserted into the resource reservation message and verified by the network.

This document describes the session authorization policy element (AUTH_SESSION) used to convey information about the resources authorized for use by a session. The host must obtain an AUTH_SESSION element from an authorizing entity via a session signaling protocol such as SIP [RFC-3261]. The host then inserts the AUTH_SESSION element into the resource reservation message to allow verification of the network resource request; in the case of RSVP, this element MUST be encapsulated in the Policy Data object [RFC-2750] of an RSVP PATH message. Network elements verify the request and then process the resource reservation message based on admission policy.

[RFC-3521] 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.

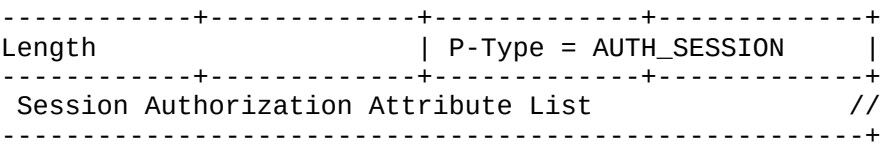
3. Policy Element for Session Authorization

3.1 Policy Data Object Format

The Session Authorization policy element conforms to the format of a POLICY_DATA object which contains policy information and is carried by policy based admission protocols such as RSVP. A detailed description of the POLICY_DATA object can be found in "RSVP Extensions for Policy Control" [RFC-2750].

3.2 Session Authorization Policy Element

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



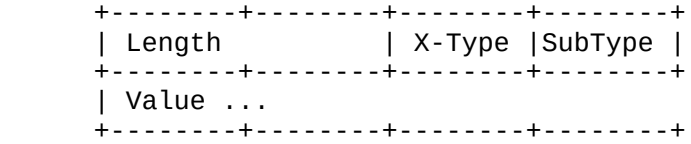
Length: 16 bits
The length of the policy element (including the Length and P-Type) is in number of octets (MUST be in multiples of 4) and indicates the end of the session authorization information block.

P-Type: 16 bits (Session Authorization Type)
AUTH_SESSION = 0x04
The Policy element type (P-type) of this element. The Internet Assigned Numbers Authority (IANA) acts as a registry for policy element types 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 resource reservation request. An initial set of valid objects is described in Section 3.3.

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: 16 bits
The length field is two octets and indicates the actual length of the attribute (including Length, X-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.

X-Type: 8 bits
Session authorization attribute type (X-Type) field is one octet. IANA acts as a registry for X-Types as described in section 7, IANA Considerations. Initially, the registry contains the following X-Types:

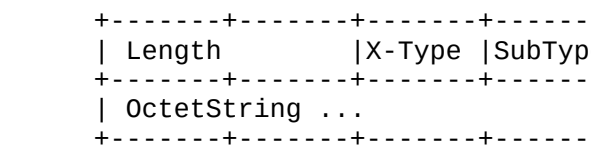
1	AUTH_ENT_ID	The unique identifier of the entity which authorized the session.
2	SESSION_ID	Unique identifier for this session.
3	SOURCE_ADDR	Address specification for the session originator.
4	DEST_ADDR	Address specification for the session end-point.
5	START_TIME	The starting time for the session.
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 X-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
Length of the attribute, which MUST be > 4.

X-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:

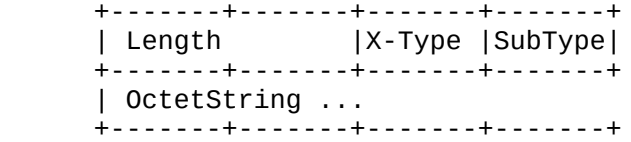
1	IPV4_ADDRESS	IPv4 address represented in 32 bits
2	IPV6_ADDRESS	IPv6 address represented in 128 bits
3	FQDN	Fully Qualified Domain Name as defined in RFC 1034 as an ASCII string.
4	ASCII_DN	X.500 Distinguished name as defined in RFC 2253 as an ASCII string.
5	UNICODE_DN	X.500 Distinguished name as defined in RFC 2253 as a UTF-8 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	The Distinguished Name of the subject of the certificate as defined in RFC 2253 as a UTF-8 string.
9	PGP_CERT	The PGP digital certificate of the authorizing entity as defined in RFC 2440.

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 numbers or on a standard NTP timestamp.



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

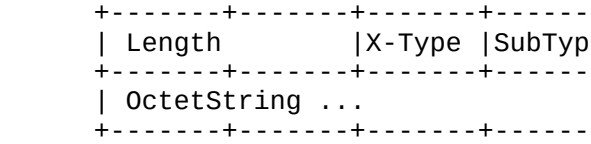
X-Type
SESSION_ID

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

X-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	UDP_PORT_LIST	list of UDP port specifications, represented as 16 bits per list entry.
4	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 2 (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 resource reservation datagram (e.g., RSVP PATH) MUST match one of the SOURCE_ADDR attributes contained in this Session Authorization Data Policy Element.

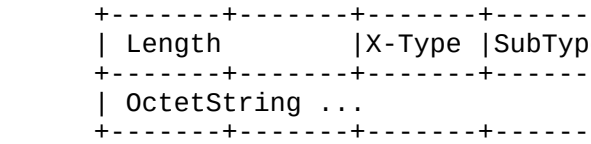
At most, one instance of subtype 3 MAY be included in every Session Authorization Data Policy Element. At most, one instance of subtype 4 MAY be included in every Session Authorization Data Policy Element. Inclusion of a subtype 3 attribute does not prevent inclusion of a subtype 4 attribute (i.e., both 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 X-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
Length of the attribute, which MUST be > 4.

X-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	UDP_PORT_LIST	list of UDP port specifications, represented as 16 bits per list entry.
4	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 2 (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 resource

reservation datagram (e.g., RSVP PATH) MUST match one of the DEST_ADDR attributes contained in this Session Authorization Data Policy Element.

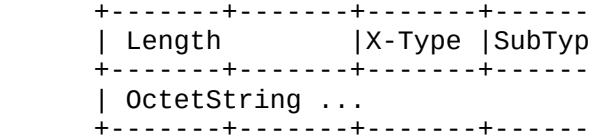
At most, one instance of subtype 3 MAY be included in every Session Authorization Data Policy Element. At most, one instance of subtype 4 MAY be included in every Session Authorization Data Policy Element. Inclusion of a subtype 3 attribute does not prevent inclusion of a subtype 4 attribute (i.e., both 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
Length of the attribute, which MUST be > 4.

X-Type
START_TIME

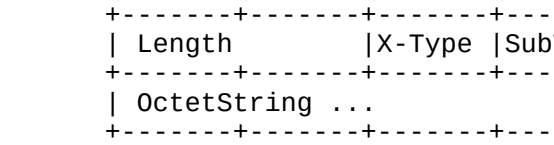
SubType
The following sub types for START_TIME are defined. IANA acts as a registry for START_TIME sub-types as described in section 7, IANA Considerations. Initially, the registry contains the following sub types for START_TIME:

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
Length of the attribute, which MUST be > 4.

X-Type
END_TIME

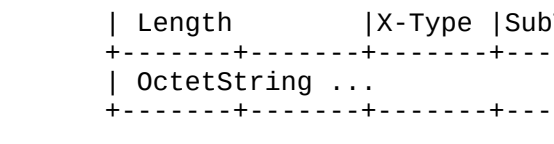
SubType
The following sub types for END_TIME are defined. IANA acts as a registry for END_TIME sub-types as described in section 7, IANA Considerations. Initially, the registry contains the following sub types for END_TIME:

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 X-Type may be useful in some scenarios to specify the specific resources authorized to ensure the request fits the authorized specifications.



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

X-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.

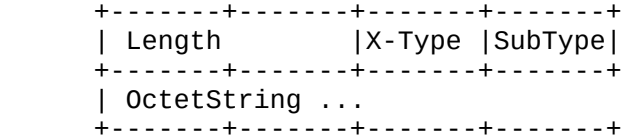
OctetString
The OctetString contains the resources specification.

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
Length of the attribute, which MUST be > 4.

X-Type
AUTHENTICATION_DATA

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

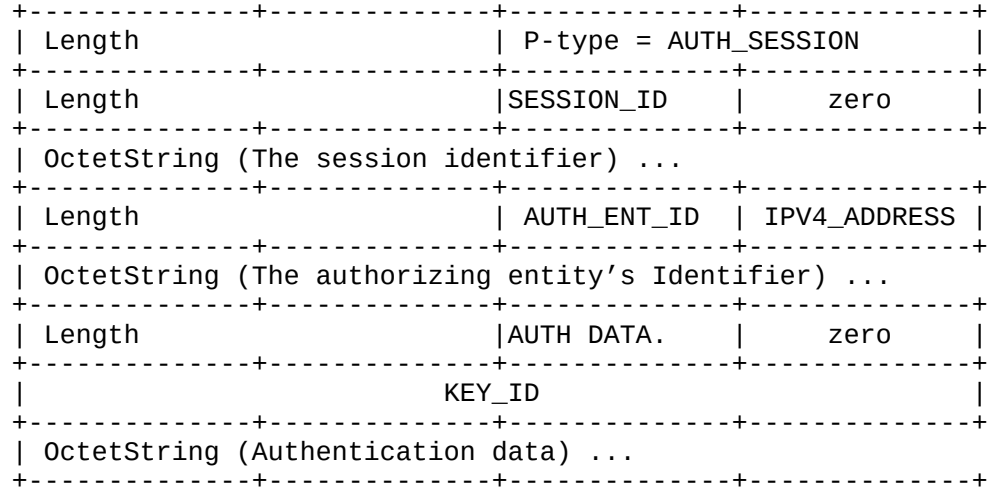
OctetString
The OctetString contains the authentication data of the AUTH_SESSION.

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 symmetric keys

In shared symmetric key environments, the AUTH_ENT_ID MUST be of subtypes: IPV4_ADDRESS, IPV6_ADDRESS, FQDN, ASCII_DN, UNICODE_DN or URI. An example AUTH_SESSION policy element is shown below.



4.1.1 Operational Setting using shared symmetric keys

This assumes both the Authorizing Entity and the Network router/PDP are provisioned with shared symmetric keys and with policies detailing which algorithm to be used for computing the authentication data along with the expected length of the authentication data for that particular algorithm.

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

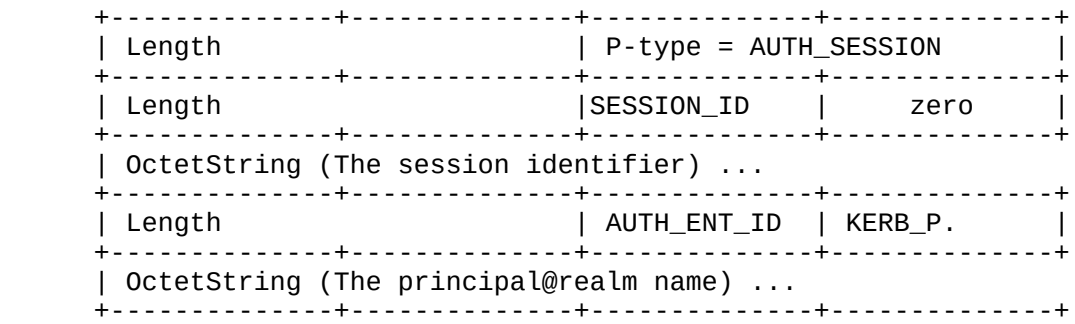
to produce the authentication data is identified by the AUTH_ENT_ID field. Since multiple keys may be configured for a particular AUTH_ENT_ID value, the first 32 bits of the AUTH_DATA field MUST be a key ID to be used to identify the appropriate key. Each key must also be configured with lifetime parameters 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-MD5-128 [RFC-2104], [RFC-1321] cryptographic algorithm for computing the authentication data. New algorithms may be added by the IETF standards process.

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. The KRB_PRINCIPAL field is defined as the Fully Qualified Kerberos Principal name of the authorizing entity. 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.



4.2.1. Operational Setting using Kerberos

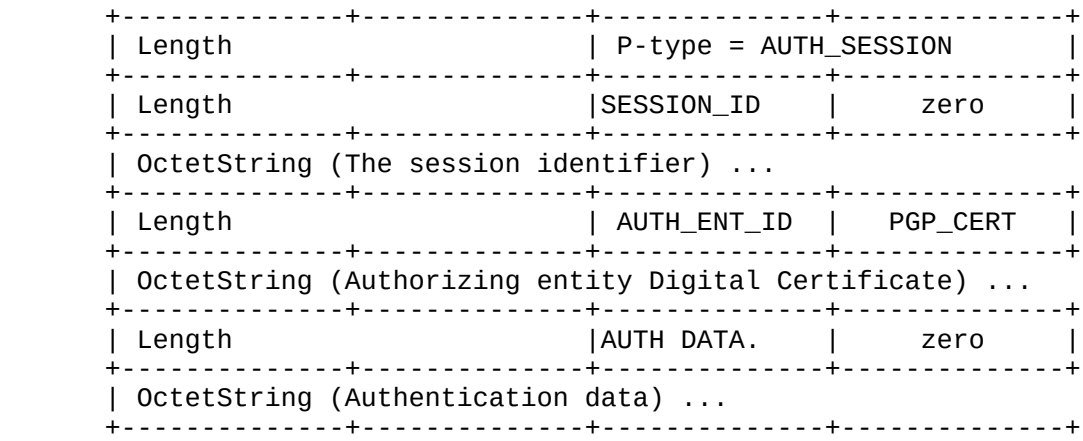
An authorizing entity is configured to construct the AUTH_SESSION policy element that designates use of the Kerberos authentication method (KRB_PRINCIPAL) as defined in RFC 1510. Upon reception of the resource reservation request, the router/PDP contacts the local KDC, with a KRB_AS_REQ message, to request credentials for the authorizing entity (principal@realm). In this request, the client (router/PDP) sends (in cleartext) its own identity and the identity of the server (the authorizing entity taken from the AUTH_ENT_ID field) for which it is requesting credentials. The local KDC responds with these credentials in a KRB_AS_REP message, encrypted in the client's key. The credentials consist of 1) a "ticket" for the server and 2) a temporary encryption key (often called a "session key"). The router/PDP uses the ticket to access the authorizing entity with a KRB_AP_REQ message. The session key (now shared by the router/PDP and the authorizing entity) is used to authenticate the router/PDP, and is used to authenticate the authorizing entity. The session key is an encryption key and is also used to encrypt further communication between the two parties. The authorizing entity responds by sending a concatenated message of a KRB_AP_REP and a KRB_SAFE. The KRB_AP_REP is used to authenticate the authorizing entity. The KRB_SAFE message contains the authentication data in the safe-body field. The authentication data must be either a 16 byte MD5 hash or 20 byte SHA-1 hash of all data in the AUTH_SESSION policy element up to the AUTHENTICATION_DATA (note that when using Kerberos the AUTH_SESSION PE should not include AUTHENTICATION_DATA as this is sent in the KRB_SAFE message). The router/PDP independently computes the hash, and compares it with the received hash in the user-data field of the KRB-SAFE-BODY [RFC-1510].

At a minimum, all AUTH_SESSION implementations using Kerberos MUST support the Kerberos des-cbc-md5 encryption type [RFC-1510] (for encrypted data in tickets and Kerberos messages) and the Kerberos rsa-md5-des checksum type [RFC-1510] (for the KRB_SAFE checksum) checksum. New algorithms may be added by the IETF standards process. Triple-DES encryption is supported in many Kerberos implementations (although not specified in [RFC-1510]), and SHOULD be used over single DES.

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.



4.3.1. Operational Setting for public key based authentication

Public key based authentication assumes the 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.

4.3.1.1 X.509 V3 digital certificates

When the AUTH_ENT_ID is of type X509_V3_CERT, AUTHENTICATION_DATA MUST be generated following these steps:

- A Signed-data is constructed as defined in section 5 of CMS [RFC-3369]. A digest is computed on the content (as specified in section 6.1) with a signer-specific message-digest algorithm. The certificates field contains the chain of authorizing entity's X.509 V3 digital certificates. The certificate revocation list is defined in the crls field. The digest output is digitally signed following section 8 of RFC 3447, using the signer's private key.

When the AUTH_ENT_ID is of type X509_V3_CERT, verification MUST be done following these steps:

- Parse the X.509 V3 certificate to extract the distinguished name of the issuer of the certificate.
- Certification Path Validation is performed as defined in section 6 of RFC 3280.
- Parse through the Certificate Revocation list to verify that the received certificate is not listed.
- Once the X.509 V3 certificate is validated, the public key of the authorizing entity can be extracted from the certificate.
- Extract the digest algorithm and the length of the digested data by parsing the CMS signed-data.
- The recipient independently computes the message digest. This message digest and the signer's public key are used to verify the signature value.

This verification ensures integrity, non-repudiation and data origin.

4.3.1.2 PGP digital certificates

When the AUTH_ENT_ID is of type PGP_CERT, AUTHENTICATION_DATA MUST be generated following these steps:

- AUTHENTICATION_DATA contains a Signature Packet as defined in section 5.2.3 of RFC 2440. In summary:
  - Compute the hash of all data in the AUTH_SESSION policy element up to the AUTHENTICATION_DATA.
  - The hash output is digitally signed following section 8 of RFC 3447, using the signer's private key.

When the AUTH_ENT_ID is of type PGP_CERT, verification MUST be done following these steps:

- Validate the certificate.
- Once the PGP certificate is validated, the public key of the authorizing entity can be extracted from the certificate.
- Extract the hash algorithm and the length of the hashed data by parsing the PGP signature packet.
- The recipient independently computes the message digest. This message digest and the signer's public key are used to verify the signature value.

This verification ensures integrity, non-repudiation and data origin.

5. Framework

[RFC-3521] describes a framework in which the AUTH_SESSION policy element may be utilized to transport information required for authorizing resource reservation for media flows. [RFC-3521] 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 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 X-TYPE
- Called party IP address or Identity (e.g., FQDN) - DEST_ADDR X-TYPE
- The characteristics of (each of) the media stream(s) authorized for this session - RESOURCES X-TYPE
- The authorization lifetime - START_TIME X-TYPE
- The identity of the authorizing entity to allow for validation of the token in shared symmetric key and Kerberos schemes - AUTH_ENT_ID X-TYPE
- The credentials of the authorizing entity in a public-key scheme - AUTH_ENT_ID X-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 X-TYPE
- Calling party port number - SOURCE_ADDR X-TYPE
- Called party port number - DEST_ADDR X-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 Generation of the AUTH_SESSION by the authorizing entity

1. Generate the AUTH_SESSION policy element with the appropriate contents as specified in section 5.

2. If authentication is needed, the entire AUTH_SESSION policy element is constructed, excluding the length, type and subtype fields of the AUTH_SESSION field. Note that the message MUST include either a START_TIME or a SESSION_ID (See Section 9), to prevent replay attacks. The output of the authentication algorithm, plus appropriate header information, is appended to the AUTH_SESSION policy element.

6.2 Message Generation (RSVP Host)

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

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

2. The AUTH SESSION policy element received from the authorizing entity (Section 3.2) MUST be copied without modification into the POLICY DATA object.

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

6.3 Message Reception (RSVP-aware 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.

6.4 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 symmetric key authentication: The Network router/PDP uses the AUTH_ENT_ID field to consult a table keyed by that field. The table should identify the cryptographic authentication algorithm to be used along with the expected length of the authentication data and the shared symmetric key for the authorizing entity. Verify that the indicated length of the authentication data is consistent with the configured table entry and validate the authentication data.

   - Public Key: Validate the certificate chain against the 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. Once the identity of the authorizing entity and the validity of the service request has been established, the authorizing router/PDP MUST then consult its local policy tables (the contents of which are a local matter) in order to determine whether or not the specific request is authorized. To the extent to which these access control decisions require supplementary information, routers/PDPs MUST ensure that supplementary information is obtained securely. An example of insecure access control decisions would be if the authorizing party relies upon an insecure database (such as DNS or a public LDAP directory) and authorizes with a certificate or an FQDN.

4. 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]. If RSVP is being used, 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 0x04.

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

X-Type AUTH_ENT_ID is assigned the value 1.
X-Type SESSION_ID is assigned the value 2.
X-Type SOURCE_ADDR is assigned the value 3.
X-Type DEST_ADDR is assigned the value 4.
X-Type START_TIME is assigned the value 5.
X-Type END_TIME is assigned the value 6.
X-Type RESOURCES is assigned the value 7.
X-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 UDP_PORT_LIST is assigned the value 3.
SubType TCP_PORT_LIST is assigned the value 4.

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 UDP_PORT_LIST is assigned the value 3.
SubType TCP_PORT_LIST is assigned the value 4.

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 document 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 and the Policy Servers MUST support NTP to ensure proper clock synchronization. Failure to ensure proper clock synchronization will allow replay attacks since the clocks of the different network entities may not be in-synch. 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 environments where shared symmetric keys are possible, they should be used in order to keep the AUTH_SESSION policy element size to a strict minimum. This is especially true in wireless environments where the AUTH_SESSION policy element is sent over-the-air. The shared symmetric keys authentication option MUST be supported by all AUTH_SESSION implementations.

If shared symmetric 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 Francois Audet, Don Wade, Hamid Syed, Kwok Ho Chan and many others for their valuable comments. Special thanks to Eric Rescorla who provided numerous comments and suggestions that improved this document.

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

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

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

[RFC-1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, November 1987.

[RFC-1305] Mills, D., "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-2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC-2205] Braden, R., Ed., 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-2253] Wahl, M., Kille, S. and T. Howes , "UTF-8 String Representation of Distinguished Names", RFC 2253, December 1997.

[RFC-2279] Yergeau, F., "UTF-8, a transformation format of ISO 10646", RFC 2279, January 1998.

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

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

[RFC-2440] Callas, J., Donnerhacke, L., Finney, H. and R. Thayer, "OpenPGP Message Format", RFC 2440, November 1998.

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

[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-3182] Yadav, S., Yavatkar, R., Pabbati, R., Ford, P., Moore, T., Herzog, S. and R. Hess, "Identity Representation for RSVP", RFC 3182, October 2001

[RFC-3280] Housley, R., Polk, W., Ford, W. and D. Solo, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3280, April 2002.

[RFC-3369] Housley, R., "Cryptographic Message Syntax", RFC 3369, August 2002.

[RFC-3447] Jonsson, J. and B. Kaliski, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1", RFC 3447, February 2003.

[RFC-3521] Hamer, L.-N., Gage, B. and H. Shieh, "Framework for Session Setup with Media Authorization", RFC 3521, April 2003.

12. Informative References

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

[RFC-3261] 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.

13. Intellectual Property Statement

14. Contributors

Matt Broda
Nortel Networks

EMail: mbroda@nortelnetworks.com

Louis LeVay
Nortel Networks

EMail: levay@nortelnetworks.com

Dennis Beard
Nortel Networks

EMail: beardd@nortelnetworks.com

Lawrence Dobranski
Nortel Networks

EMail: ldobran@nortelnetworks.com

15. Authors' Addresses

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
Invidi Technologies
Edmonton, Alberta
Canada T5J 3S4

EMail: brettk@invidi.com

Bill Gage
Nortel Networks
PO Box 3511 Station C
Ottawa, Ontario
Canada K1Y 4H7

Phone: +1 613.763.4400
EMail: gageb@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

16. Full Copyright Statement

Acknowledgement

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