Internet-Draft
IETF Common Authentication Technology WG
<draft-ietf-cat-snego-02.txt>

Eric Baize, Denis Pinkas
Bull

17 October 1996

Simple GSS-API Negotiation Mechanism

STATUS OF THIS MEMO

2. ABSTRACT

This draft document specifies a Security Negotiation Mechanism for the Generic Security Service Application Program Interface (GSS-API) which is described in [1].

The GSS-API provides a generic interface which can be layered atop different security mechanisms such that if communicating peers acquire GSS-API credentials for the same security mechanism, then a security context may be established between them (subject to policy). However, GSS-API doesn't prescribe the method by which GSS-API peers can establish whether they have a common security mechanism.

The Simple GSS-API Negotiation Mechanism defined here is a pseudo-security mechanism, represented by the object identifier iso.org.dod.internet.security.mechanism.snego (1.3.6.1.5.5.2) which enables GSS-API peers to determine in-band whether their credential

share common GSS-API security mechanism(s), and if so, to invoke normal security context establishment for a selected common security mechanism. This is most useful for applications that are based on GSS-API implementations which support multiple security mechanisms.

As most existing GSS-API security mechanisms can support different options (such as differing cryptographic algorithms due to policy or legislative constraints), the Simple GSS-API Negotiation Mechanism allows to negotiate security mechanisms including their options (i.e. variants). Mechanism options can be considered as providing a type of "quality of protection" for security contexts.

To facilitate mechanism negotiation, the OID which currently defines a security mechanism is "extended" to be able to specify options within a security mechanism rather than simply the basic mechanism. When the OID specifies the mechanism only and no explicit option, then this means that the default option is used. The default option and the specific options for a given mechanism are as defined in the IETF GSS-API specification(s) for the mechanism.

This allows to negotiate basic security mechanisms, different options within a given security mechanism or different options from several basic security mechanisms.

In addition, a given security mechanism may still negotiate mechanism-specific options during the context establishment for that mechanism, i.e. after the mechanism has been selected by the negotiation process.

The simple GSS-API mechanism negotiation is based on a two-ways negotiation model : The initiator proposes one or several security mechanisms, the target either accepts the proposed security mechanism, or chooses one from an offered set, or rejects the proposed value(s). The target informs the initiator of its choice and may also return mechanism specific information related to the chosen mechanism.
The simple GSS-API mechanism negotiation does not provide any security feature to protect the initially exchanged values for security context parameters (i.e. during the negotiation process).

The Simple GSS-API Negotiation Mechanism uses the concepts developed in GSS-API specification [1], and requires the use of a new GSS-API context-level token : the negotiation token. Callers of the GSS-API do not need to be aware of the negotiation token but only of the new

pseudo-security mechanism. A failure in the negotiation phase causes a major status code to be returned: GSS_S_BAD_MECH.

3. NEGOTIATION MODEL

3.1. Negotiation description

The model for security mechanism negotiation reuses a subset of the concepts specified in [2].

Each security mechanism represents one basic security mechanism along with one option for this security mechanism (when no option is present the default option is assumed).

- When one security mechanism is proposed by the initiator, it represents the only security mechanism option supported or selected (when the additional APIs defined in the Annex A are used) by the initiator.

- When several security mechanisms are proposed by the initiator, they represent a set of security mechanisms supported or selected (when the additional APIs defined in the Annex A are used) by the initiator.

The target negotiation reply contains the result of the negotiation (accept or reject) and, in case of accept, the agreed security mechanism along with optional mechanism specific information.

In case of a successful negotiation, the security mechanism represents the value suitable for the target, and picked up from the list offered by the initiator. The target selects the value according to a simple

selection criteria: it checks if the first entry from its own list is present in the set offered by the initiator. If the entry is present, then it is the agreed mechanism, if not then the second entry from its own ordered list is checked and the process continues until all entries have been checked. Thus, the target's mechanism preferences have precedence when more than one common mechanism is available between the target and initiator.

3.2. Negotiation procedure

The negotiation procedure is summarised as follows:

(a) the GSS-API initiator invokes GSS_Init_sec_context as normal, but requests (either explicitly, with the negotiation mechanism, or through accepting a default, when the default is the negotiation mechanism) that the Simple GSS-API Negotiation Mechanism be used;

(b) the initiator GSS-API implementation emits a negotiation token containing the set of supported security mechanisms for the credentials used for this context establishment, and indicates GSS_CONTINUE_NEEDED status;

(c) The GSS-API initiator sends the token to the target application;

(d) The GSS-API target deposits the token through invoking GSS_Accept_sec_context. The target GSS-API implementation emits a negotiation token containing which if any of the proposed mechanisms it supports (or has selected).

If the proposed mechanism(s) are accepted, GSS_Accept_sec_context() indicates GSS_CONTINUE_NEEDED status.

If the proposed mechanism(s) are rejected, GSS_Accept_sec_context() indicates GSS_S_BAD_MECH status. The security context initialisation has failed.

(e) The GSS-API target returns the token to the initiator application;

(f) The GSS-API initiator deposits the token through invoking GSS_Init_sec_context.

If the negotiation token carries an accept result, GSS_Init_sec_context() returns an initial context token as output_token, and indicates GSS_CONTINUE_NEEDED or GSS_COMPLETE status. The initiator sends the output_token to the target. The security context initialisation is then performed according to the standard GSS-API conventions for the selected mechanism. When GSS_COMPLETE is returned, the mech_type output parameter indicates the selected mechanism. Since the negotiation exchanges are not cryptographically protected, the initiator GSS-API implementation must check the returned/selected mechanism options with its originally submitted list of mechanism options. When GSS_CONTINUE_NEEDED is returned, the mech_type output parameter is not yet valid.

Note that the *_req_flag input parameters for context establishment are relative to the selected mechanism, as are the *_state output parameters. i.e., these parameters are not applicable to the negotiation process per se.

If the negotiation token carries a reject result, the context establishment is impossible, and GSS_Init_sec_context() indicates GSS_S_BAD_MECH status. For example, a rejection will occur if the target doesn't support the initiator's proposed mechanism type(s) and/or mechanism option(s). Upon failure of the mechanism negotiation procedure, the mech_type output parameter value is the negotiation mechanism type. However, upon failure of the selected mechanism context establishment, the mech_type output parameter value is the selected mechanism type.

On receipt of a negotiation token on the target side, a GSS-API implementation that does not support negotiation would indicate the GSS_FAILURE status as if a particular basic security mechanism had been requested but was not supported.

When GSS_Acquire_cred is invoked with the negotiation mechanism as desired_mechs, an implementation-specific default credential is used to carry on the negotiation. A set of mechanisms as specified locally by the system administrator is then available for negotiation. If there is a desire for the caller to make its own choice, then an additional API has to be used (see Appendix A).

4. DATA ELEMENTS

4.1. Mechanism Type

MechType::= OBJECT IDENTIFIER

mechType
: The concept of mechType is extended to specify a basic security mechanism including its options. Each basic security mechanism

is as defined in [1], and must provide a single default option which fully specifies the mechanism. The default option is represented by the OID of the mechanism itself (i.e. without any extension).

The options are specified by extending the OID. This extension is defined in the same IETF GSS-API specification as the security mechanism context token specification.

4.2. Negotiation Token

The negotiation token syntax follows InitialContextToken syntax defined in [1]. The negotiation token is identified by an Object Identifier (tbd). This section specifies the syntax of the corresponding "innerContextToken" field.

```
NegotiationToken ::= CHOICE {
                              negTokenReq  [0]  NegTokenReq,
                              negTokenRep  [1]  NegTokenRep }

NegTokenReq ::= SEQUENCE of MechType
```

negTokenReq
: Negotiation token sent by the initiator to the target, which contains one or more security mechanisms supported by the initiator.

negTokenRep
: Negotiation token returned by the target to the initiator which contains a global negotiation result, the security mechanism selected (if any) and optional information specific to the security mechanism selected by the target.

```
NegTokenRep ::= SEQUENCE {
          negResult          [0] ENUMERATED { accept (0), reject (1) }
          supportedMech      [1] MechType OPTIONAL
          MechSpecInfo       [2] OCTET STRING   OPTIONAL}
```

negResult
: Result of the negotiation exchange, specified by the target. This can be either :

  accept
  : The target accepts one of the proposed security mechanisms, or,

  reject
  : The target rejects all the proposed security mechanisms.

supportedMech
: This field has to be present when negResult is "accept". It is a choice from the mechanisms offered by the initiator.

MechSpecInfo
This field may be used to transmit mechanism specific information relative to the security mechanism selected by the target.

5. EXAMPLES : SECURITY MECHANISM NEGOTIATION

Follow some examples of security mechanism options negotiation between an initiator (I) and a target (T).

5.1. Initial steps

(I) supports two security mechanism types (GSS-MECH1 and GSS-MECH2), and two options for GSS-MECH2 : OPTION1, identified by GSS-MECH2-OPTION1 and OPTION2, identified by GSS-MECH2-OPTION2.

(I) invokes GSS_Init_sec_context() with :

Input
mech_type = OID for negotiation mechanism or NULL, if the negotiation mechanism is the default mechanism.

Output
major_status = GSS_CONTINUE_NEEDED
output_token = negTokenReq

The negotiation token (negTokenReq) contains three security mechanisms with :
mechType = GSS-MECH1 or
mechType = GSS-MECH2-OPTION1 or
mechType = GSS-MECH2-OPTION2

(I) sends to (T) the negotiation token.

5.2 Successful negotiation steps

(T) supports GSS-MECH2-OPTION1.
(T) receives the negotiation token (negTokenReq) from (I)
(T) invokes GSS_Accept_sec_context() with :

Input
input_token = negTokenReq

Output
major_status = GSS_CONTINUE_NEEDED
output_token = negTokenRep

```
The negotiation token (negTokenRep) contains :
     negResult = accept (the negotiation result)
     supportedMech : mechType = GSS-MECH2-OPTION1

(T) returns the negotiation token (negTokenRep) to (I)
(I) invokes GSS_Init_sec_context() with :
```



```
Input
     input_token = negTokenRep

Output
     major_status = GSS_COMPLETE
     output_token = initialContextToken (initial context token
                                         for GSS-MECH2-OPTION1)
     mech_type = GSS-MECH2-OPTION1

The subsequent steps are security mechanism specific, and works as
specified in [1].
```

5.3. Failed negotiation steps

```
(T) supports GSS-MECH3.
(T) receives the negotiation token (negTokenReq) from (I)
(T) invokes GSS_Accept_sec_context() with :

Input
     input_token = negTokenReq

Output
     major_status = GSS_S_BAD_MECH
     output_token = negTokenRep

The negotiation token (negTokenRep) contains :

     negResult = reject (the negotiation result)

(T) returns the negotiation token (negTokenRep) to (I)
(I) invokes GSS_Init_sec_context() with :

Input
     input_token = negTokenRep

Output
     major_status = GSS_S_BAD_MECH
```

The security context establishment has failed.

6. ACKNOWLEDGEMENTS

Acknowledgement are due to Piers McMahon and Tom Parker of ICL, Stephen Farrell of SSE, Doug Rosenthal of EINet and John Linn of Openvision for reviewing earlier versions of this document and for providing useful inputs.

7. SECURITY CONSIDERATIONS

The purpose of the generic simple GSS-API mechanism negotiation mechanism is to enable peers to agree on the value for a security

mechanism and security related options required for initialising security services.

As this mechanism is called prior to any initialisation of a security service, it cannot make use of any security feature. Therefore it is exposed to all threats a non secured service is exposed. Thus communicating peers may be exposed to the denial of service threat, or can be forced by an active attacker to use a security mechanism which is not their common preferred one (when multiple security mechanisms are shared between peers) but which is acceptable anyway to the target.

APPENDIX A

GSS-API NEGOTIATION SUPPORT API

In order to provide to a GSS-API caller (either the initiator or the target or both) the ability to choose among the set of supported mechanisms a reduced set of mechanisms for negotiation, two additionnal APIs are defined:

GSS_Get_neg_mechs() indicates the set of security mechanisms available on the local system to the caller for negotiation.

GSS_Set_neg_mechs() specifies the set of security mechanisms to be used on the local system by the caller for negotiation.

A.1. GSS_Get_neg_mechs call

Input:
     cred_handle   OCTET STRING - NULL specifies default credentials

Outputs:
     major_status INTEGER,
     minor_status INTEGER,
     mech_option_set SET OF OBJECT IDENTIFIER

Return major_status codes :
     GSS_COMPLETE indicates that the set of security mechanism
     options available for negotiation has been returned in
     mech_option_set.
     GSS_FAILURE indicates that the requested operation could not
     be performed for reasons unspecified at the GSS-API level.

Allows callers to determine the set of security mechanism options available for negotiation. This call is intended for support of specialised callers who need to reduce the set of negotiable security mechanism options from the set of supported security mechanisms available to the caller (based on available credentials).

Note: The GSS_Indicate_mechs() function indicates the full set of mechanism types available on the local system. Since this call does not use a credential handle as an input parameter, the returned set is not necessarily available for all credentials.

A.2. GSS_Set_neg_mechs call

Input:
     cred_handle   OCTET STRING - NULL specifies default credentials
     mech_option_set SET OF OBJECT IDENTIFIER

Outputs:

```
     major_status INTEGER,
     minor_status INTEGER,

Return major_status codes :
     GSS_COMPLETE indicates that the set of security mechanisms
     available for negotiation has been set to mech_option_set.
     GSS_FAILURE indicates that the requested operation could not be
     performed for reasons unspecified at the GSS-API level.

Allows callers to specify the set of security mechanism options that
may be negotiated: A NULL mech_option_set specifies that only the
default mech_type with the default option is available for the GSS-API
implementation. This call is intended for support of specialised
callers who need to restrict the set of negotiable security mechanism
options from the set of all security mechanism options available to
the caller (based on available credentials). Note that if more than
one mechanism is specified in mech_option_set, the order in which those
mechanisms are specified implies a relative mechanism preference for
the target.
```

REFERENCES

[1] Linn, J., "Generic Security Service Application Program Interface", RFC 1508, OpenVision, September 1993.

[2] Standard ECMA-206, "Association Context Management including Security Context Management", December 1993. Available on http://www.ecma.ch

AUTHORS'S ADDRESSES

Eric Baize — Internet email: E.Baize@ma02.bull.com
Bull HN - MA02/211S — Phone: +1 508 294 61 37
Technology Park — Fax: +1 508 294 61 09
Billerica, MA 01821 - USA

Denis Pinkas — Internet email: D.Pinkas@frcl.bull.fr
Bull — Phone: +33 1 30 80 34 87
Rue Jean-Jaures — Fax: +33 1 30 80 34 70
BP 68
78340 Les Clayes-sous-Bois - FRANCE