The ATM Forum

Technical Committee

Control Plane Security

AF-SEC-0172.000

November 2001

The ATM Forum
Worldwide Headquarters
P.O. Box 29920
572 B Ruger Street
San Francisco, CA 94129-0920
Tel: +1.415.561.6110
Fax: +1.415.561.6120

Acknowledgments

The production of this specification would not be possible without the enormous amount of effort provided by many individuals. Special acknowledgements for their hard work and dedication go to Richard Graveman and Wolfgang Klasen, the current chair and vice-chair.

In addition, the following individuals (listed alphabetically), among others, contributed their time and expertise to the development of this specification:

Robert Dianda
Jim Harford
Kim Hebda
Chris Kubic
Brian Rosen
John Rutemiller
Jeffery See

Gary Buda
Editor, ATM Forum Security Working Group

TABLE OF CONTENTS

1. Introduction

This specification defines mechanisms and procedures for providing security services for control plane information. Whereas the *ATM Security Specification* [3] contains procedures for providing control plane integrity using preplaced keys, this specification provides new security functionality for the control plane. This specification also provides an underlying mechanism for implementing PNNI routing security [6].

1.1 Goals

The goals of this specification are to provide:

1. Peer entity authentication for control plane communications.

2. Security services for AAL and SAAL level SDUs:

   ?? Confidentiality for control plane messages.

   ?? Data origin authentication and data integrity for control plane messages.

   ?? Replay detection for control plane messages.

3. Negotiation of the above security services for the control plane.
   Unlike the *ATM Security Specification* [3], this specification provides a mechanism that allows negotiation of confidentiality and integrity services to protect control plane messages.

4. Automated key exchange.
   The *ATM Security Specification* [3] defines control plane authentication and integrity using only preplaced keys. This specification provides initial key exchange and key update for control plane security.

5. Security policy enforcement.
   This specification provides a method to specify how security is applied to outgoing messages based upon properties of the message and recipient. It also provides a method to specify how incoming messages are handled based upon the properties of the message and the sender.

6. An underlying mechanism for implementing PNNI Routing security.
   This specification provides the underlying security services and mechanisms to support PNNI routing security [6].

1.2 References

[1] "Advanced Encryption Standard," NIST, http://csrc.nist.gov/encryption/aes/rijndael/.

[2] ATM Forum Technical Committee, "ATM Forum Addressing: Reference Guide," AF-RA-0106.000, February 1999.

[3] ATM Forum Technical Committee, "ATM Security Specification," Version 1.1, AF-SEC-0100.002, October 2000.

[4] ATM Forum Technical Committee, "Integrated Local Management Interface," Version 4.0, AF-ILMI-0065.000, September 1996.

[5] ATM Forum Technical Committee, "Private Network-Network Interface Specification," Version 1.0, AF-PNNI-0055.000, March 1996.

[6] ATM Forum Technical Committee, "Addendum to PNNI v1.0—Secure Routing," AF-RA-PNNI-RSEC-0171.000, August 2001.

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

[8] IETF, "Diffie-Hellman Key Agreement Method," RFC 2631, E. Rescorla, June 1999.

[9] IETF, "The ESP CBC-Mode Cipher Algorithms," RFC 2451, R. Pereira, R. Adams, November 1998.

[10] IETF, "The ESP DES-CBC Cipher Algorithm With Explicit IV," RFC 2405, C. Madson, N. Doraswamy, November 1998.

[11] IETF, "The ESP Triple DES Transform," RFC 1851, P. Karn, P. Metzger, W. Simpson, September 1995.

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

[13] IETF, "The Internet Key Exchange (IKE)," D. Harkins, D. Carrel, RFC 2409, November 1998.

[14] IETF, "The Internet IP Security Domain of Interpretation for ISAKMP," RFC 2407, D. Piper, November 1998.

[15] IETF, "Internet Security Association and Key Management Protocol (ISAKMP)," RFC 2408, D. Maughan, M. Schertler, M. Schneider, J. Turner, November 1998.

[16] IETF, “IP Encapsulating Security Payload (ESP),” RFC 2406, S. Kent, R. Atkinson, November 1998.

[17] IETF, “IP Security Document Roadmap,” RFC 2411, R. Thayer, N. Doraswamy, R. Glenn, November 1998.

[18] IETF, “The NULL Encryption Algorithm and Its Use With IPsec,” RFC 2410, R. Glenn, S. Kent, November 1998.

[19] IETF, “The OAKLEY Key Determination Protocol,” RFC 2412, H. Orman, November 1998.

[20] IETF, “Security Architecture for the Internet Protocol,” RFC 2401, S. Kent, R. Atkinson, November 1998.

[21] IETF, “Test Cases for HMAC-MD5 and HMAC-SHA-1,” RFC 2202, P. Cheng, R. Glenn, September 1997.

[22] IETF, “The Use of HMAC-MD5-96 within ESP and AH,” RFC 2403, C. Madson, R. Glenn, November 1998.

[23] IETF, “The Use of HMAC-SHA-1-96 within ESP and AH,” RFC 2404, C. Madson, R. Glenn, November 1998.

[24] ITU-T, “B-ISDN Signaling ATM Adaptation Layer – Service Specific Coordination Function for Support of Signaling at the User Network Interface (SSFC at UNI),” Recommendation Q.2130, July 1994.

1.3 Definition

CPS - a protocol that provides strong integrity, authentication, and confidentiality for control plane SDUs.

1.4 Acronyms

AAL	ATM Adaptation Layer
AES	Advanced Encryption Standard
AESA	ATM End System Address
AH	Authentication Header
AINI	ATM Inter-Network Interface
BLLI	Broadband Lower Layer Information
CPS	Control Plane Security
DOI	Domain of Interpretation
ESP	Encapsulating Security Payload
IKE	Internet Key Exchange
ILMI	Integrated Local Management Interface
IPSec	IP Security

IV	Initialization Vector
MAC	Message Authentication Code
MIB	Management Information Base
MTU	Maximum Transmission Unit
OUI	Organizationally Unique Identifier
PDU	Protocol Data Unit
PID	Protocol Identifier
PKI	Public Key Infrastructure
PNNI	Private Network-Network Interface
PTSE	PNNI Topology State Element
RCC	Routing Control Channel
SAAL	Signalling ATM Adaptation Layer
SAP	Service Access Point
SDU	Service Data Unit
SME	Security Message Exchange
SPD	Security Policy Database
SPI	Security Parameter Index
SVCC	Switched Virtual Channel Connection
UNI	User-Network Interface

1.5 Scope

The scope of this specification is indicated in Figure 1. The checked cells of the matrix indicate what is within the scope of this specification.

	User Plane	Control Plane	Management Plane
Authentication		?	
Confidentiality		?	
Data Integrity		?	
Access Control			

Figure 1: Security Services Supported in this Specification.

The following protocols are covered by the security services and mechanisms defined in this document: UNI signaling, PNNI signaling, PNNI routing, and AINI signaling. The security services defined in this document protect the SDUs transferred by these protocols and authenticate the entities sending and receiving these SDUs.

The mechanisms defined in this document neither interoperate with nor preclude the implementation and use of the control plane security mechanisms defined in [3].

This specification defines three separate protocols: Control Plane Security (CPS), which provides strong integrity, authentication, and confidentiality for control plane SDUs; IKE; and SME, which provide two alternative security negotiation and key management services for CPS.

1.5.1 CPS Protection for Control Plane SDUs

CPS is an ATM protocol that provides three security services for signaling or routing messages:

1. Strong cryptographic data integrity implemented with a cryptographic checksum. The message recipient can verify that a party holding a shared key created the checksum and the message has not been modified since the checksum was applied. This service protects against spoofing and malicious modification threats.
2. Optionally, replay and reordering detection, which detects duplicated or out-of-sequence messages.
3. Optionally, confidentiality, which conceals the contents of a SDU from eavesdroppers.

1.5.2 IKE Security Negotiation

IKE is one of two alternatives defined in this specification to perform negotiation, key exchange, and initial authentication for CPS. It is an authentication and key exchange protocol defined by the Internet Engineering Task Force (IETF) [13] and [14]. The following IKE exchanges, as well as preplaced keys, are supported:

?? Main Mode,
?? New Group Mode,
?? Aggressive Mode,
?? Quick Mode,
?? Informational Exchange.

This document contains the changes needed to implement IKE for the protection of ATM SDUs with CPS.

1.5.3 SME Security Negotiation

SME is the other alternative defined in this specification to perform key exchange and initial authentication for CPS. SME is an authentication and key exchange protocol defined by the ATM Forum in [3]. Previously, SME was used only to negotiate and establish security services for ATM user plane traffic, authenticate nodes, and perform initial key exchange. This specification further defines the use of SME in the control plane to accomplish those same tasks. One of the changes involves enhancements to SME for negotiation of the AAL confidentiality service for control plane or PNNI routing messages.

1.5.4 Preplaced Keys

Preplaced keys, that is a shared master key and initial session keys, are installed and verified by mechanisms outside the scope of this specification. When using preplaced keys, optionally, a SKC operation (as defined in [3]) may be carried out after AAL-ESTABLISH is received.

1.6 Compliance

Table 1 specifies the algorithms and modes (where applicable) required for each ATM control plane security algorithm profile defined. Compliance requirements for the algorithms are specified in [3]. Some algorithms discussed in [3] do not appear in Table 1.

To claim compliance with this specification, an implementation must support at least one of the following mechanisms for key management: (1) IKE with or without manual key distribution, (2) SME with OAM key update, or (3) preplaced keys and OAM key update. Additionally, a compliant implementation must support the CPS frame format and a MAC algorithm. Table 1 lists reasonable choices as of the time of publication. When SME is used, key update must be performed with OAM cells even when preplaced keys are used.

Table 1: Control Plane Security Algorithm Profiles.

Profile	MAC	Confidentiality	Key Update	Security Exchange Protocol
CPS-SME-1	H-MD5	AES/CBC	OAM	SME
CPS-SME-2	H-MD5	Triple DES/CBC	OAM	SME
CPS-SME-3	H-SHA-1	AES/CBC	OAM	SME
CPS-SME-4	H-SHA-1	Triple DES/CBC	OAM	SME
CPS-IKE-1	H-MD5	AES/CBC	Quick Mode AES/CBC, MD5	IKE
CPS-IKE-2	H-MD5	Triple DES/CBC	Quick Mode Triple DES/CBC, MD5	IKE
CPS-IKE-3	H-SHA-1	AES/CBC	Quick Mode AES/CBC, SHA-1	IKE
CPS-IKE-4	H-SHA-1	Triple DES/CBC	Quick Mode Triple DES/CBC, SHA-1	IKE
CPS-PPK-1	H-MD5	AES/CBC	OAM	Preplaced
CPS-PPK-2	H-MD5	Triple DES/CBC	OAM	Preplaced
CPS-PPK-3	H-SHA-1	AES/CBC	OAM	Preplaced
CPS-PPK-4	H-SHA-1	Triple DES/CBC	OAM	Preplaced

2. Security Services for the Control Plane

This specification defines the Control Plane Security (CPS) protocol that provides data origin authentication, data integrity, optionally replay and reordering detection, and optionally confidentiality. Security is provided in a hop-by-hop manner between physically or logically adjacent signaling and routing elements.

2.1 Reference Models

2.1.1 Reference Model (Signaling)

Figure 2 shows the reference model for securing signaling messages. Note that the figure illustrates physically adjacent nodes, but this can also be applied to logically adjacent nodes (nodes that terminate a VP connection). Security OAM cell processing is not shown in this figure.

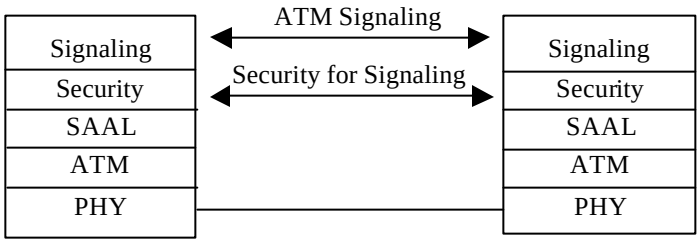

Figure 2: Reference Model (Signaling).

Between the Signaling and SAAL protocols, the function labeled “Security” in the figure above provides CPS and optionally IKE or SME.

2.1.2 Reference Model (Routing)

Figure 3 shows the reference model for securing routing messages. Note that the figure illustrates physically adjacent nodes, but this can also be applied to logically adjacent nodes (nodes that terminate a SVCC RCC). Security OAM cell processing is not shown in this figure.

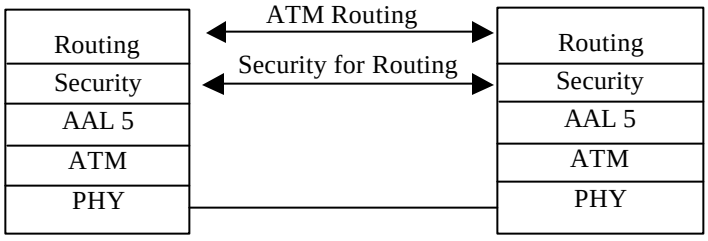

Figure 3: Reference Model (Routing).

Between the Routing and AAL protocols, the function labeled “Security” in the figure above provides CPS and optionally IKE or SME.

2.1.3 Keys

There are several different options for how key material may be established for use with this specification. First, preplaced keys may be used. Preplaced keys are handled identically to the master keys and initial session keys as used by SME in [3]. OAM cells are used to update the session keys.

Second, SME may be used to establish master keys and session keys. OAM cells are used to update the session keys.

Third, IKE may be used in two different ways. Using the first method, an IKE Main Mode or Aggressive Mode exchange is used to establish a Phase 1 ISAKMP security association, optionally, New Group exchanges may take place, and then a Quick Mode exchange is used to establish the traffic protection keys. Using the second method, IKE is used in manual key distribution mode. With this option, the manually distributed key is used to replace a Phase 1 exchange, and a quick mode exchange is still used to establish traffic protection keys. When manual key distribution is used with IKE, there is no automated mechanism to update the Phase 1 exchange.

2.1.4 Layer Interaction

The security layer intercepts incoming messages. The security layer processes these messages according to the security association for the link and then passes the validated messages up the protocol stack. On generation of an outgoing message, the message is passed down to the security layer for processing prior to transmission. Based on its local policy, the security layer selectively may or may not discard each incoming or outgoing unprotected signaling messages (e.g., to support a fall-back mechanism in the event that one of the two nodes does not support security). In the event that the security layer is not present, any security negotiation messages or protected messages will be ignored by the higher layers in the protocol stack.

2.2 Procedures for Link Bring Up

Upon link bring-up, both ends of the link must determine whether or not the link should be secured with CPS, and whether IKE or SME or neither shall be used. If both peers consider that the link need not be secured, then no additional procedures are necessary. If either peer considers that the link need be secured, then both peers must agree on which protocols shall be used. One of the following methods shall be used.

2.2.1 Pre-Established Configuration

If this method is used, the choice of protocols is manually configured at both ends of the link. Pre-established configuration is appropriate where other methods are unavailable or undesirable due to security policy. This method imposes an administrative burden and introduces the potential for human error.

2.2.2 ILMI

If this method is used, the choice of protocols is determined by ILMI exchanges across the link. The advantages of this approach are that it is automated and builds upon existing ILMI mechanisms. Note that for PNNI, separate MIB objects are needed to cover PNNI Signaling and PNNI Routing. This choice will not work for PNNI Routing Control Channels established by SVCCs.

The following MIB objects are added to [4].

```
atmfAtmLayerSignalingSecurity  OBJECT-TYPE
SYNTAX INTEGER {
        unsecured(1),          -- no security supported on this channel
        Cps(2),                -- CPS protocol; use preplaced key
        CpsSme(3),             -- CPS protocol; use SME for negotiation
        CpsIke(4),             -- CPS protocol; use IKE for negotiation
        CpsSmeIke(5)           -- CPS protocol; use SME or IKE for negotiation
}
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
        "The protocols used to secure the UNI or NNI signaling channel."
::= { atmfAtmLayerEntry 16 }


atmfAtmLayerPnniRccSecurity  OBJECT-TYPE
SYNTAX INTEGER {
        unsecured(1),          -- no security supported on this channel
        Cps(2),                -- CPS protocol; use preplaced key
        CpsSme(3),             -- CPS protocol; use SME for negotiation
        CpsIke(4),             -- CPS protocol; use IKE for negotiation
        CpsSmeIke(5)           -- CPS protocol; use SME or IKE for negotiation
}
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
        "The protocols used to secure the PNNI Routing Control Channel.  Note
        that this applies only to PNNI RCCs established over a physical link and
        not those PNNI RCCs established via SVCCs."
::= { atmfAtmLayerEntry 17 }
```

2.2.3 Signaling

If this method is used, the choice of protocols is determined by ATM signaling messages. Note that this method is only applicable to the case of establishing a PNNI Routing Control Channel via a SVCC. The advantages of this approach are that it is automated

and builds upon existing signaling mechanisms. This shall be done via Broadband Lower Layer Information (BLLI) negotiation during call setup.

Prior to this specification, the sole BLLI codepoint used to establish a RCC across a SVCC was 0x00A03E000A, where the first three octets identify the ATM Forum's Organizationally Unique Identifier (OUI) and the last two octets denote a Protocol Identifier (PID) of "PNNI Routing Control Channel." When BLLI negotiation is used, the initiator of the call sends up to three BLLI codepoints representing the desired security selection in order of preference. The responder sends back the one BLLI codepoint that it has selected.

This specification adds the following new and redefined BLLI codepoints:

PID	Security Protocols	Semantic Meaning (with respect to security)
0x000A	none	PNNI without security
0x0015	CPS	PNNI with security; use preplaced key
0x0016	CPS, SME	PNNI with security; use SME for negotiation
0x0017	CPS, IKE	PNNI with security; use IKE for negotiation

2.3 Frame Formats

2.3.1 CPS Frame Format

The frame format of the Control Plane Security protocol data unit (CPS PDU) for the transport of secured messages is defined below. Note that a common CPS frame format is used, regardless of whether an IKE, SME, or preplaced keys are used.

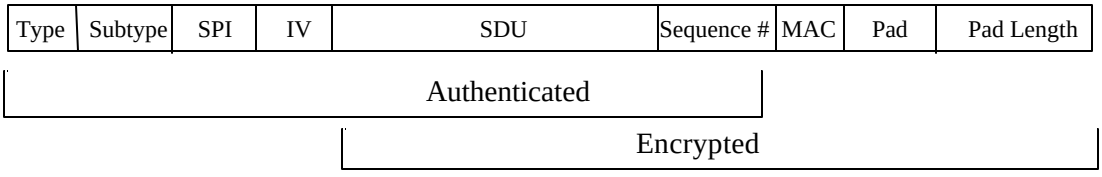

Figure 4: Frame Format for Control Plane Security.

The following definitions for the various frame fields apply:

- **Type** – This one octet field identifies the contents of the message as a security message. The value of this field is purposely chosen to avoid conflicts with ITU protocol discriminators and PNNI routing message types. This field is coded as the value 0xF0.
- **Subtype** – This one octet field identifies the contents of the remainder of the message. The Subtype field is coded as 0x00 to denote a CPS message.

- ?? **SPI** or "Security Parameters Index" – identifies a security association if IKE was used. Otherwise, this two octet field shall be encoded as 0x0000 and ignored upon reception.
- ?? **IV** – the initialization vector for certain encryption algorithms. This field exists only when the confidentiality service is provided and the particular encryption algorithm and mode require an initialization vector. The length of this field, when it exists, is mechanism dependent.
- ?? **SDU** or "Service Data Unit" – carries the secured payload message.
- ?? **Sequence Number** – contains a value that is incremented with each successive PDU in a given security association. This four octet field exists only when the message integrity service with replay protection is used.
- ?? **MAC** – contains the output of a computation based on the message and the integrity key that authenticates the message. This field is mandatory. The length of this field is mechanism dependent.
- ?? **Pad** – padding present with certain encryption algorithms. This field exists when the confidentiality service is provided and the particular encryption algorithm and mode require or allow for padding. Padding may be used, for example, because of the block size or desire to hide the actual message length. The length of this field, when it exists, is specified in the Pad Length field.
- ?? **Pad Length** – the length of the Pad field. This two octet field exists when the Pad field exists.

When encoding a control plane message for subsequent transmission across a secured control channel, the following order of operations is used. First, the authentication algorithm is applied over the following fields within the frame: Type, Subtype, SPI, IV (if it exists), SDU, and Sequence Number (if it exists). If confidentiality is provided, then the encryption algorithm is applied over the following fields within the frame: SDU, Sequence Number (if it exists), MAC, Pad (if it exists), and Pad Length (if it exists).

Implementation Note: The order of applying the authentication and confidentiality algorithms is the opposite of the order used in IPSec ESP.

2.3.2 Encapsulated IKE Frame Format

The frame format of encapsulated IKE messages for the negotiation of security services is defined below.

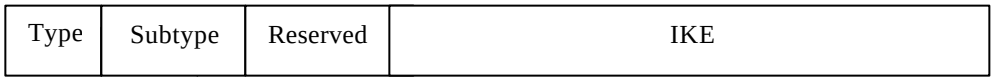

Figure 5: Frame Format for Encapsulated IKE Messages.

The following definitions for the various frame fields apply:

- ?? **Type** – This one octet field identifies the contents of the message as a security message. The value of this field is purposely chosen to avoid conflicts with

ITU protocol discriminators and PNNI routing message types. This field is coded as the value 0xF0.

- **Subtype** – This one octet field identifies the contents of the remainder of the message. The Subtype field is coded as 0x01 to denote an encapsulated IKE message.
- **Reserved** – This two octet field is used to align the IKE message on a 32-bit boundary. This field shall be coded as all zeros.
- **IKE** – This variable length field contains the actual IKE message that is used to negotiate security services.

2.3.3 Encapsulated SME Frame Format

- The frame format of encapsulated SME messages for the negotiation of security services is defined below.

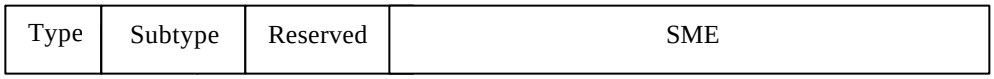

Figure 6: Frame Format for Encapsulated SME Messages.

The following definitions for the various frame fields apply:

- **Type** – This one octet field identifies the contents of the message as a security message. The value of this field is purposely chosen to avoid conflicts with ITU protocol discriminators and PNNI routing message types. This field is coded as the value 0xF0.
- **Subtype** – This one octet field identifies the contents of the remainder of the message. The Subtype field is coded as 0x02 to denote an encapsulated SME message.
- **Reserved** – This two octet field is used to align the SME message on a 32-bit boundary. This field shall be coded as all zeroes.
- **SME** – This variable length field contains the actual SME message that is used to negotiate security services as defined in Section 5.1.5.3.2 in [3].

2.4 Security Interoperability

If one node is security capable (and uses security negotiation) and the other node is not, the security-capable node begins by sending security negotiation messages. If a node not capable of security receives a security negotiation message, the Type field is unidentifiable, and the message is discarded. The security-capable node retransmits the security negotiation message a predetermined number of times. If a response to the security message is not received, it then concludes that the peer node is not capable of security. If the security policy for the secure node allows unprotected connections, the control plane message can then be sent unprotected. If the security policy for the security-capable node does not allow unprotected connections, control plane messages between these two nodes are not permitted.

Here are several example cases that may occur involving interoperability:

1) Security agent receives an unprotected message. This scenario deals with a secure node receiving an unprotected message. The secure node either discards the unprotected message or passes it up the stack according to local security policy.

2) A node that is not security aware receives a security message. This scenario occurs when a secure node and a node that is not security aware are adjacent. The node that is not security aware discards the security messages it receives because the value in the Type field is not a recognized protocol discriminator.

3) Security agent mismatch between IKE and SME. This scenario occurs when a node that implements only one of the two methods of negotiating security receives a message with the other's Subtype field. The receiving node is able to understand that the message is a security message. However, the Subtype field, which identifies the message as an IKE or SME message, specifies an incompatible security option. Therefore, the connection cannot be secured using this method to negotiate the security association.

4) Receiving security agent supports neither IKE nor SME. This scenario covers a node that only implements preplaced keys for control plane security receiving a security message from a node that implements IKE-based or SME-based security negotiation. The receiving node is able to understand that the message is a security message. However, the Subtype field, which identifies the message as an IKE or SME message, specifies an incompatible security option. The connection cannot be secured using IKE or SME to negotiate the security association. If the initiating node implements SME and is configured with the appropriate preplaced key, the connection can be secured using preplaced keys. If the receiving node implements only IKE, the connection cannot be secured using preplaced keys.

5) Initiating security agent does not use IKE or SME. This scenario covers a node that implements either IKE or SME receiving a protected message from a node that only implements preplaced keys for control plane security. If the receiving node implements SME and is configured with the appropriate preplaced keys, the connection can be secured. If the receiving node implements only IKE the connection cannot be secured using preplaced keys.

6) Mismatch between security options supported on both sides. It is possible for both nodes to support the same security negotiation mechanisms but not to have any security algorithms or modes in common. If this is the case, the connection cannot be secured. The only option is to allow an unprotected connection if the security policies for both nodes permit.

Note: Whether a security negotiation message has priority over the use of preplaced keys is a matter of policy.

3. Signaling Security

Implementation Note: The use of CPS reduces the MTU size available to signaling. Implementations should take this into account.

3.1 Interaction With SAAL

A design goal of security services for Signaling is to appear transparent both to the protocol layer immediately above (Signaling) and to the layer immediately below (SAAL). This section describes the behavior of the security services intended to realize that goal. Additional information can be found in Appendix A of this specification. Figure 7 illustrates the SAPs that exist above and below security.

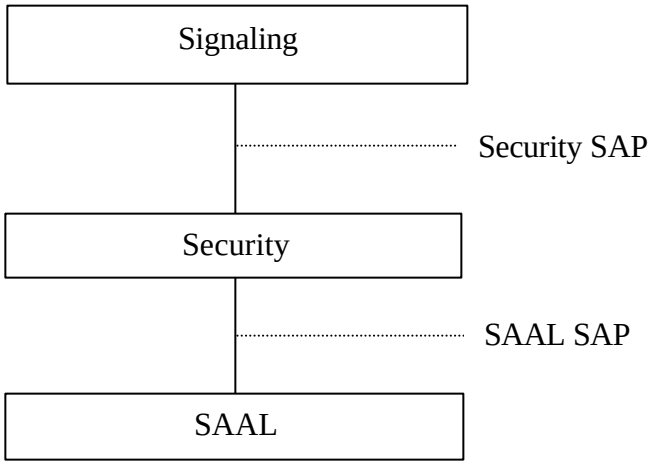

Figure 7: Security SAP and SAAL SAP.

When SAAL releases the connection, Security should preserve the security association for a period of TS1. If the connection is re-established within the period TS1, the same security association shall be used. The recommended value of TS1 is 2 minutes, but it may be configured to other values.

3.1.1 Primitives

Prior to the introduction of signaling security services, the service access point (SAP) between the SAAL layer and the Signaling layer consists of these primitives:

?? AAL-ESTABLISH request/indication/confirm
?? AAL-RELEASE request/indication/confirm
?? AAL-DATA request/indication.

For further details on these primitives, see [24]. Note that the AAL-UNITDATA primitive is not used.

When signaling security services are employed, the Security layer is present between the SAAL layer and the Signaling layer (see Figure 7). In this case, the SAP described in the previous paragraph specifies the interface between the SAAL layer and the Security layer. A separate, analogous SAP is present between the Security layer and the Signaling layer. This new SAP shall have the following primitives:

?? SECURE-AAL-ESTABLISH request/indication/confirm
?? SECURE-AAL-RELEASE request/indication/confirm
?? SECURE-AAL-DATA request/indication.

The semantics and parameters of this new SAP shall be the same as the SAP directly above the SAAL layer, except the primitive names have been changed to denote that these primitives are invoked between the Security layer and the Signaling layer.

3.1.2 States

The following states are used to describe the behavior of the Security layer with respect to SAAL:

?? SAAL-UNAVAILABLE
?? SAAL-UNSECURE
?? SAAL-SECURE.

The following variables are used to describe the behavior of the Security layer with respect to SAAL:

?? LOCAL-INITIATION
?? DATA-QUEUE
?? BLOCK-RELEASE-CONFIRM.

Upon initialization, the Security layer shall be in state SAAL-UNAVAILABLE. Variable LOCAL-INITIATION shall be set to FALSE. Variable DATA-QUEUE shall be initialized to an empty queue. Variable BLOCK-RELEASE-CONFIRM shall be set to FALSE.

Upon establishment of the SAAL link, the state changes to SAAL-UNSECURE. During this state, the Security layer will attempt to establish a security association with its peer, depending on local policy.

Upon establishment of the security association, the state changes to SAAL-SECURE. During this state, the Signaling layer is allowed to exchange messages with its peer.

If at any time, a connectivity loss is detected across the SAAL link, or the security association is deemed corrupted, then the state immediately returns to SAAL-UNAVAILABLE.

3.1.3 Finite State Machine

3.1.3.1 Initialization

Upon initialization, the Security layer remains in state SAAL-UNAVAILABLE until changed by one of the events specified below.

3.1.3.2 AAL-ESTABLISH

The following actions are taken upon receipt of an AAL-ESTABLISH primitive (from SAAL):

Type	State	Action(s)
indication	SAAL-UNAVAILABLE	?? State changes to SAAL-UNSECURE. ?? IKE or SME state machines attempt to establish a security association with peer if none currently exists.
indication	SAAL-UNSECURE	?? IKE or SME state machines attempt to establish a security association with peer if none currently exists.
indication	SAAL-SECURE	?? State changes to SAAL-UNSECURE. ?? IKE or SME state machines attempt to establish a security association with peer if none currently exists.
confirm	SAAL-UNAVAILABLE	?? State changes to SAAL-UNSECURE. ?? IKE or SME state machines attempt to establish a security association with peer if none currently exists.
confirm	SAAL-UNSECURE	?? IKE or SME state machines attempt to establish a security association with peer if none currently exists.
confirm	SAAL-SECURE	?? State changes to SAAL-UNSECURE. ?? IKE or SME state machines attempt to establish a security association with peer if none currently exists.

3.1.3.3 AAL-RELEASE

The following actions are taken upon receipt of an AAL-RELEASE primitive (from SAAL):

Type	State	Action(s)
indication	SAAL-UNAVAILABLE	?? Set variable LOCAL-INITIATED to FALSE and set variable DATA-QUEUE to empty. ?? Issue a SECURE-AAL-RELEASE indication primitive to Signaling.
indication	SAAL-UNSECURE	?? State changes to SAAL-UNAVAILABLE. ?? Set variable LOCAL-INITIATED to FALSE and set variable DATA-QUEUE to empty. ?? Issue a SECURE-AAL-RELEASE indication primitive to Signaling.
indication	SAAL-SECURE	?? State changes to SAAL-UNAVAILABLE. ?? Issue a SECURE-AAL-RELEASE indication primitive to Signaling.
confirm	SAAL-UNAVAILABLE	?? If variable BLOCK-RELEASE-CONFIRM is FALSE, then issue a SECURE-AAL-RELEASE confirm primitive to Signaling.
confirm	SAAL-UNSECURE	?? State changes to SAAL-UNAVAILABLE. ?? If variable BLOCK-RELEASE-CONFIRM is FALSE, then issue a SECURE-AAL-RELEASE confirm primitive to Signaling.
confirm	SAAL-SECURE	?? State changes to SAAL-UNAVAILABLE. ?? Issue a SECURE-AAL-RELEASE confirm primitive to Signaling.

3.1.3.4 AAL-DATA

The following actions are taken upon receipt of an AAL-DATA primitive (from SAAL):

Type	State	Action(s)
indication	SAAL-UNAVAILABLE	?? No action is needed; this is a protocol error condition that should never occur.
indication	SAAL-UNSECURE	?? If the data parameter is an IKE or SME message, then process the message using the appropriate IKE or SME procedures.
indication	SAAL-SECURE	?? If the data parameter is an IKE or SME message, then process the message using the appropriate IKE or SME procedures. ?? If the data parameter is a CPS message, and the message can be decoded without error, then issue a SECURE-AAL-DATA indication primitive to Signaling (along with the decoded data parameter). ?? If the data parameter is a CPS message, and decoding the message yields an error, then discard the data and take no further action.

3.1.3.5 SECURE-AAL-ESTABLISH

The following actions are taken upon receipt of a SECURE-AAL-ESTABLISH primitive (from Signaling):

Type	State	Action(s)
request	SAAL-UNAVAILABLE	?? Set variable LOCAL-INITIATION to TRUE. ?? If there is a data parameter, then queue the data in DATA-QUEUE for later transmission. ?? Issue an AAL-ESTABLISH request primitive to SAAL.
request	SAAL-UNSECURE	?? Set variable LOCAL-INITIATION to TRUE. ?? If there is a data parameter, then queue the data in DATA-QUEUE for later transmission.
request	SAAL-SECURE	?? If there is a data parameter, then encode it via the CPS protocol and issue an AAL-DATA request primitive to SAAL. ?? Issue a SECURE-AAL-ESTABLISH confirm primitive to Signaling.

3.1.3.6 SECURE-AAL-RELEASE

The following actions are taken upon receipt of a SECURE-AAL-RELEASE primitive (from Signaling):

Type	State	Action(s)
request	SAAL-UNAVAILABLE	?? Set the following variables: LOCAL-INITIATION = FALSE DATA-QUEUE = empty BLOCK-RELEASE-CONFIRM = FALSE.
request	SAAL-UNSECURE	?? State changes to SAAL-UNAVAILABLE. ?? Set the following variables: LOCAL-INITIATION = FALSE DATA-QUEUE = empty BLOCK-RELEASE-CONFIRM = FALSE. ?? Issue an AAL-RELEASE request primitive to SAAL.
request	SAAL-SECURE	?? If there is a data parameter, then encode it via the CPS protocol and issue an AAL-DATA request primitive to SAAL. ?? State changes to SAAL-UNAVAILABLE. ?? Set variable BLOCK-RELEASE-CONFIRM to FALSE. ?? Issue an AAL-RELEASE request primitive to SAAL.

3.1.3.7 SECURE-AAL-DATA

The following actions are taken upon receipt of a SECURE-AAL-DATA primitive (from Signaling):

Type	State	Action(s)
request	SAAL-UNAVAILABLE	?? Return an error indicating AAL layer not available for service.
request	SAAL-UNSECURE	?? Return an error indicating AAL layer not available for service.
request	SAAL-SECURE	?? Encode the data parameter via the CPS protocol. ?? Issue an AAL-DATA request primitive to SAAL.

3.1.3.8 Security Association Established

The following actions are taken when a security association becomes established. This could result from (1) the IKE or SME state machines determining that a valid security association exists, or (2) when preplaced keys are used and other events have caused the local security layer's state to reach SAAL-UNSECURE.

?? State changes to SAAL-SECURE.
?? If variable LOCAL-INITIATION is TRUE, then reset the variable to FALSE and issue a SECURE-AAL-ESTABLISH confirm primitive to Signaling. Otherwise, issue a SECURE-AAL-ESTABLISH indication primitive to Signaling.
?? If variable DATA-QUEUE is not empty, then the data parameter (from a previous SECURE-AAL-ESTABLISH primitive) that is currently queued shall be dequeued, encoded via the CPS protocol, and transmitted to the peer via an AAL-DATA request primitive issued to SAAL.

3.1.3.9 Security Association Invalidated

The following actions are taken when a security association becomes invalidated:

?? State changes to SAAL-UNAVAILABLE.
?? Set variable BLOCK-RELEASE-CONFIRM to TRUE.
?? Issue an AAL-RELEASE request primitive to SAAL.
?? Issue a SECURE-AAL-RELEASE indication primitive to Signaling.

This specification does not define the conditions under which a security association is deemed invalid; the detection of an invalid security association is a matter of implementation and policy.

One potential (but not mandatory) algorithm for such detection follows. The Security layer shall count within a time period, *T*, these events:

?? The number of received CPS messages that could be decoded without error (*G* for "Good").
?? The number of received CPS messages that had errors upon decoding (*B* for "Bad"). The security association is deemed invalid if within the duration of *T*, *B* is greater than some threshold, *X*, and *G* equals zero. Values for parameters *T* and *X* are implementation dependent.

3.2 Naming (Node Authentication)

For SME implementations, ATM network elements shall be identified by the initiator, responder, and security agent distinguished name octet groups defined in [3].

For IKE implementations, the following fields are used for the Identification Type field found in the Identification Payload:

ID Type	Value
Reserved	0
ID_IPV4_ADDR	1
ID_FQDN	2
ID_IPV6_ADDR	5
ID_DER_ASN1_DN	9
ID_DER_ASN1_GN	10
ID_KEY_ID	11
ID_AESA	249
ID_E164	250

When using certificate-based authentication, local policy decisions can be made with SME or IKE by including all relevant IDs within the certificates.

4. Routing Security

The fundamental strategy of PNNI routing security is to provide strong authentication before PNNI peer discovery, an SDU level integrity mechanism for PNNI peer entity communication, and confidentiality of routing information. The CPS protocol provides strong authentication, data integrity, replay detection, and confidentiality for all PNNI routing information. This mechanism directly counters the threats of unauthorized introduction, modification, and disclosure of routing information while in transit.

Secure PNNI routing is designed to offer protection with minimal configuration requirements for the communicating parties. Security for the complete PNNI routing infrastructure relies on an explicit chain of trust, which requires each node to take responsibility for the data that it summarizes and transmits.

The approach specifies the use of either shared secret key or public key cryptographic techniques for peer entity authentication prior to any Hello protocol exchanges.

Mechanisms required to implement routing security as described below are defined in [6]. In cases of discrepancies between this document and [6], [6] takes precedence.

Implementation Note: The use of CPS reduces the MTU size available to routing. Implementations should take this into account.

4.1 Secure Tags

In a scenario of mixed PNNI security capable and incapable nodes, even though the mechanisms to allow secure mesh operation are provided, a peer may need to interact with insecure PNNI sources. In this case, PTSEs from these insecure sources will be received and will have to be propagated as not trusted. Another possible scenario is the case in which secure PTSEs have to be sent to neighbors that do not support secure operation. In both of these cases, PNNI information must be identifiable as secure or insecure in order to maintain the chain of trust on which the secure operation of PNNI is based.

To understand when PNNI information becomes insecure, different concepts have to be introduced indicating when:

?? a PTSE has been originated by an insecure node,

?? a PTSE has been constructed based on insecure or a mixture of secure and insecure information, or

?? a PTSE was transferred through an untrusted link, therefore losing its security.

A distinction is made between information that has traversed only secure links and information that has traversed insecure links. PTSEs that have been transmitted only over secure links are called transmit-secure.

Additionally, as indicated above, the scenario is likely where a PTSE will be derived using information provided from a mixture of secure and insecure PTSEs and therefore the PTSE itself must be marked as not secure. As an example, there is the potential for PTSEs to be formed that contain aggregated prefix information that is based on longer prefixes contained in both secure and insecure PTSEs. To allow PNNI to differentiate between secure and insecure PTSEs in this case, a secure tag is used. This tag is set if and only if all information contained in the PTSE is derived from PTSEs:

?? having secure tags set and

?? being received through trusted links and

?? being transmit-secure.

The PTSE with this tag set will be called tagged-secure, which indicates that the information contained within has been derived from secure sources. Figure 8 shows examples of the relationship between secure and insecure status based on the tagged-secure and transmit-secure tags.

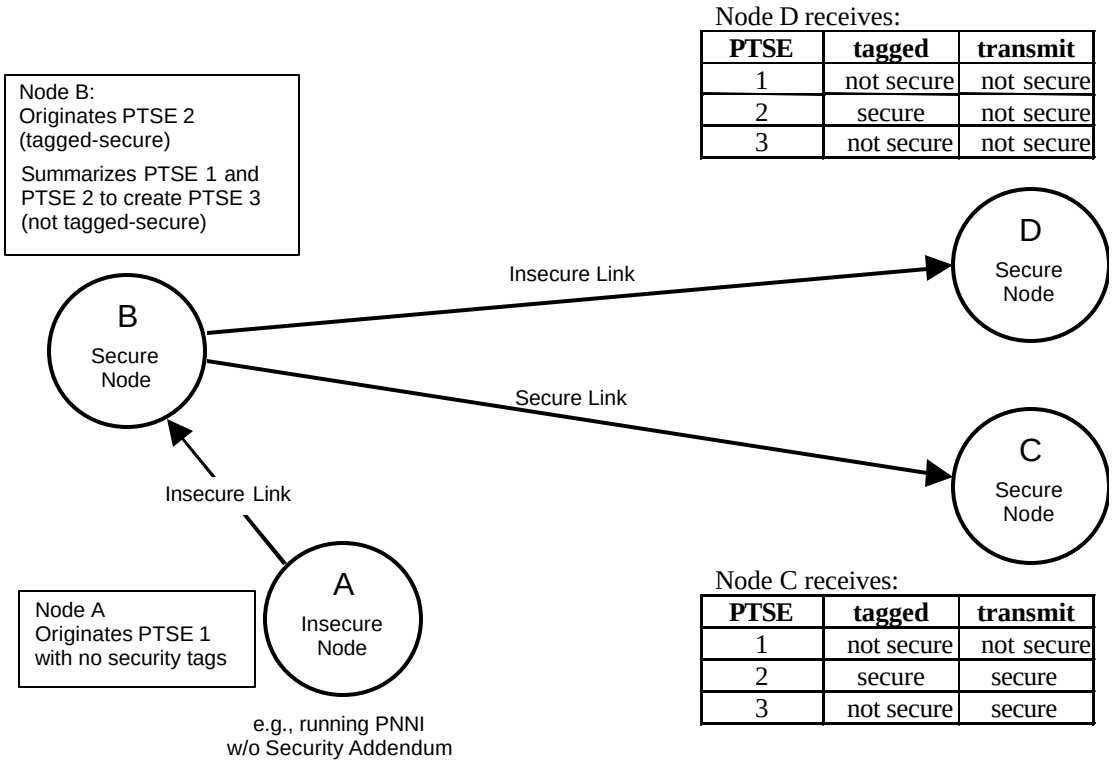

Figure 8: Tagged-Secure and Transmit-Security Example.

The example starts with an insecure node A originating a PTSE with identifier 1. Since it is not security capable, it cannot tag the PTSE. Upon reception, the secure node B marks this PTSE as not tagged-secure, since it lacks an explicit tag, and as transmit-insecure, since it was received over an insecure link. Node B, having itself originated a tagged-

secure PTSE 2, summarizes from both PTSE 1 and PTSE 2 and creates a new tagged-insecure PTSE 3. All of the above are flooded to node C connected through a secure link and to node D connected through an insecure link. At node C, only PTSE 2 is considered both tagged-secure and transmit-secure. At node D, all of the PTSEs are considered not transmit-secure, since they were received over an insecure link.

In order to maintain the integrity of the security mechanisms, the tagged-secure indication must not be changed when a PTSE is flooded through the network.

The secure tag plays a rather important role when database summary packets are being sent out. Given the fact that a PTSE, although tagged-secure, can be not transmit-secure, it should not be summarized in the database description as tagged-secure. Therefore, the originating node sets the secure tag in the database summary only when it is both tagged-secure and transmit-secure. This allows the receiver of the summary to decide whether it wants to request a secure transmission of this PTSE when already holding an insecure instance of the PTSE.

4.2 Node Authentication

The purpose of peer entity authentication is to validate node identity. Two mechanisms are specified, one based on shared secret key cryptography, and the other based on public key cryptography. Although shared secret key techniques are easier to implement, they do not possess good scalability properties.

4.2.1 Shared Secret Key

In the case of two nodes using shared secret key, a test pattern is protected by both nodes and transmitted to the neighboring node after the security association has been established. This test pattern will prevent unpredictable results which may arise when two nodes with different keys attempt to exchange protected information. The test pattern shall be all “0”s.

Due to concerns arising from multiple nodes sharing the same secret, it is recommended that shared secret key be used on a link by link basis.

4.2.2 Public Key

A public/private key pair may be assigned to each node within the network. In this case, authentication down to the node level is possible. This implementation requires a public key infrastructure (PKI) which must handle the issues of key revocation and key expiration. These PKI mechanisms are outside the scope of this specification.

4.2.3 Node Keys

Each node may have its own public/private node key pair associated with it to authenticate itself and to exchange session keys with its neighbors. Node keys are not required, except when authentication of each node is desired.

When shared secret key techniques are used to authenticate peer group entities, each peer within a peer group will need to share the same authentication key. Care should be taken in assuring that this key remains private. It is recommended that shared secret keys be used on a link by link basis. The mechanism for establishing and distributing the peer group shared secret key is beyond the scope of this document.

4.3 Access Control

In order to control which nodes are permitted to become members of different peer groups, an access control list shall be created. The access control list shall be used for nodes that support certificate based authentication. The access control list shall be an ordered list that is searched the same way each time. The access control list shall be consulted when certificates are exchanged with SME or IKE.

For the nodes that use shared secret keys, possession of the proper key shall constitute permission to join the peer group.

4.4 Secure Routing Protocol Procedures

The security agent shall stop all outgoing PNNI packets from being transmitted until a security association is established with the neighboring node. After the security association is established, the PNNI FSM shall proceed unaltered as defined in [5]. All outgoing PNNI packets shall be protected according to the security association that has been negotiated.

Upon receipt of a packet from a neighboring node, the security agent examines the protocol discriminator of the packet. If the protocol discriminator identifies the packet as containing a CPS message, the security agent processes the packet according to the established security association for the link. If the packet can be properly processed by the security agent, it is then sent to the PNNI routing stack. If the protocol discriminator identifies the packet as an unprotected PNNI packet, the security agent discards the packet if unprotected PNNI messages are not permitted by the security policy. If unprotected PNNI messages are permitted by the security policy, the message is passed up to PNNI unaltered.

Once a security association has been established, the security agent shall apply the negotiated security services to all outgoing PNNI packets before any AAL 5 processing. The existing protocol discriminator and payload will be encapsulated with a new protocol discriminator that indicates that it is a protected message.

The Routing Control Channel is initialized through the exchange of HELLO protocol messages and does not require any specific indication from the security layer.

5. Using Mechanisms Derived from IKE

IKE is used to authenticate nodes (i.e., security agents), to negotiate and to establish security associations between nodes, and to establish keying material and session keys. A security policy database (SPD), as defined in [20] and modified in this specification, specifies how selectors are used to protect outgoing messages and to process incoming messages.

This mechanism works by first establishing a single IKE Phase 1 security association between peer security agents. The security agents may then initiate a Phase 2 exchange to establish a separate security association for each control protocol. For example, the IKE Phase 1 security association formed between two nodes, because they need to secure PNNI routing exchanges, may subsequently be used between the same two nodes to establish another IKE Phase 2 security association to protect PNNI signaling messages. IKE Phase 2 exchanges are also used for key updates. The initiator determines whether to reuse a security association for multiple control plane protocols. If not, new Phase 1 and Phase 2 security associations may be established for each control plane protocol. The responding security agent must support both schemes.

All IKE exchanges will take place in-band and shall be in accordance with [20], [13], and [14], except for the following details, which define the ATM Domain of Interpretation (DOI) and other differences necessary to use IKE over an ATM VC rather than with IP. See also [7], [11], [12], [21], [18], [9], and [8].

For RFC 2401:

1. The traffic security protocol for the ATM DOI is CPS rather than AH or ESP. CPS is defined, above, in this specification. It more closely resembles ESP than AH, so it is described in terms of its differences from ESP.
2. In the CPS message format specified in Section 2.3.1 of this specification, there are no provisions for nesting the CPS mechanism or using SA bundles, no IP header, and no next protocol. Therefore, there are no distinctions between transport and tunnel modes and no IP Path MTU considerations. Also, the address component of the name of a security association in the Security Association Database is an ATM AESA or E.164 address, and the length of the SPI is two bytes.
3. If manual key distribution is used, authentication shall be performed using IKE quick mode.
4. For selectors in the Security Policy Database, addresses are ATM AESA or E.164 addresses, names are as described for the ISAKMP Identification Payload below, labels and port numbers are not supported, and the ITU-T Protocol Discriminator replaces the Transport Layer Protocol. Also, the PNNI Routing selectors listed in Section 5.2 of this specification are used when the Protocol Discriminator specifies PNNI Routing.
5. If replay detection is used, the ReplayWindowSize in Appendix C of RFC 2401 is 0.
6. In RFC 2401, Section 4.6.3 on locating a security gateway, Section 4.7 on IP multicast, Sections 5.1.2 and 5.2.2 on tunnel mode headers, Section 6 on ICMP error

processing, Section 8 on security labels, Appendix B on fragmentation, and Appendix D on ICMP do not apply.

For RFC 2409:

1. In RFC 2409, Section 2, client negotiation is not supported.
2. In RFC 2409, Section 3.2, the identification payload is as modified for the ATM DOI (see below).

For RFC 2407 (all section numbers below apply to RFC 2407):

1. All references to the Internet DOI shall be translated to references to the ATM DOI.
2. In Section 2, the ATM DOI is coded as zero or a value to be assigned by IANA. In all other cases, it is ignored.
3. In Section 4.2, the Situation for the ATM DOI is SIT_IDENTITY_ONLY.
4. In Section 4.4.1, the ATM DOI Security Protocol Identifier shall be
   PROTO_ISAKMP 1 or
   PROTO_CPS 249.
5. With respect to Section 4.4.1, payload compression shall not be used.
6. Section 4.4.3 on AH shall not apply.
7. Section 4.4.4 shall define the service CPS rather than ESP, and all references to ESP shall be changed to CPS. The CPS transforms correspond to the ESP transforms except as follows:
   - The following transforms have no CPS counterpart:
     ESP_DES_IV64
     ESP_DES_IV32
     3IDEA.
   - The following new transform is defined:
     AES_CBC 7 [1]
   - When used with IKE, the reference for 3DES is [11] and the new transform number is 5.
8. Section 4.4.5, IPCOMP, shall not apply.
9. In Section 4.5, the security association attributes shall be modified as follows:
   - The following attributes shall not be used:
     Encapsulation Mode,
     Compress Dictionary Size, and
     Compress Private Algorithm.
   - The Authentication Algorithm must be used.
   - The Authentication Algorithm shall not be KPDK.
10. In Section 4.6.1, the Security Association payload shall include only the first 12 octets.
11. In Section 4.6.2, for the Identification Payload:
    - Protocol ID and Port shall be coded as 0.
    - The following Identification Types shall not be used:
      ID_USER_FQDN,
      ID_IPV4_ADDR_SUBNET,
      ID_IPV6_ADDR_SUBNET,

ID_IPV4_ADDR_RANGE, and
ID_IPV6_ADDR_RANGE.

?? The following additional Identification Types shall be coded as in [2]:
ID_AESA 249.
ID_E164 250.

12. In Section 4.6.3, the SPI size is either 16 bytes for ISAKMP or 2 bytes for CPS.
13. In Section 4.6.3.2, the REPLAY_STATUS message should be used with Phase 2 Quick Mode exchanges. If the REPLAY_STATUS message is not used, then the replay and reordering protection option of CPS shall be used.

The following IKE exchanges as well as preplaced keys are supported:

?? Main Mode,
?? New Group Mode,
?? Aggressive Mode,
?? Quick Mode, and
?? Informational Exchange.

The Phase 1 key may be shared across multiple ATM protocols for control information exchanged between a pair of security agents. The Phase 2 keys that are generated during IKE quick mode correspond to the initial session keys generated by SME or session key updates. Phase 2 keys are unique to each control plane application.

5.1 Security Association

A Phase 2 security association is a one way association between two peers. A security association is uniquely identified by a combination of the SPI, the local physical port ID, and the VPI/VCI. These three parameters are used to identify security associations for incoming SDUs in the Security Association Database (SAD).

5.2 Security Policy Database

To secure the PNNI routing protocol, the following selectors shall be added to the security policy database:

?? Peer Group ID: This may be any value that is a valid peer group ID, a range of values, or a wildcard value.

?? PNNI packet Type: This may be any value from 1 to 7 or a wildcard value.

?? Destination Node Identifier: ATM End System Address: This may a single Node ID or a range of Node IDs. (Note that this selector is the 20 octet AESA as defined in [5]).

?? Source Node Identifier: ATM End System Address: This may be a single Node ID or a range of Node IDs.

?? Source and Destination Peer Group ID: This may be any single value expressed as a 14 octet value as defined in [5].

?? Level Indicator: This consists of two values defining the highest and lowest level in

the PNNI hierarchy to which the selector applies. (Note that this is the first octet of the Node Identifier as defined in [5]).

6. Using Mechanisms Derived from SME

This section specifies the changes and enhancements to SME, compared to the *ATM Security Specification* [3]. With the exception of those items listed here, SME follows the same approach used in [3]. The procedures for SME within the control plane are the same as securing a point-to-point PVC for user plane data. The SME used is the 3-Way in-band SME as defined in Section 5.1.5.3.2 of [3].

The following codepoints are added to the SME "Data Confidentiality Service Options" to support confidentiality services at the AAL SDU level:

?? 0x02 = Supported at AAL SDU Level (applies only when used by the initiator in FLOW1-3WE)
?? 0x82 = Required at AAL SDU Level.

Section 2.3 of this specification, shows the CPS frame format for control plane security. The SPI field shall be coded as all "0" when using SME, because the security association is specified by the VPI/VCI of the ATM cells comprising the CPS PDU.

Key updates continue to be performed with OAM cells, consistent with [3].

Appendix A: SAAL/Signaling Event-Scenario Diagrams (Informative)

The diagrams in this appendix illustrate flows among the various layers of two nodes exchanging secured control plane messages. In each diagram, the three leftmost vertical bars represent the Signaling, Security, and SAAL protocols on one side of the link. The three rightmost vertical bars represent the corresponding peers in reverse order on the other side of the link. Horizontal arrows are the messages between layers, progressing forward in time from the top to the bottom of each diagram.

Please note the following shorthand notations have been employed in the primitive names:

- In all primitives between the Signaling and Security layers, the prefix "SECURE-AAL-" is omitted.
- In all primitives between the Security and SAAL layers, the prefix "AAL-" is omitted.
- All request primitives are suffixed with "(R)."
- All indication primitives are suffixed with "(I)."
- All confirm primitives are suffixed with "(C)."

For example, the "SECURE-AAL-ESTABLISH request" primitive is simply denoted "ESTABLISH (R)."

A.1 Connection Establishment

A.1.1 No Data Transfer

In this scenario, illustrated in Figure A-1, the Signaling layer at one end initiates establishment of the signaling channel. After a SAAL link is established, the Security layer peers negotiate a security association. Using IKE or SME to negotiate the security association requires that messages be exchanged across the SAAL link. Finally, after the security association becomes established, both Signaling layer peers are notified that the signaling channel is available.

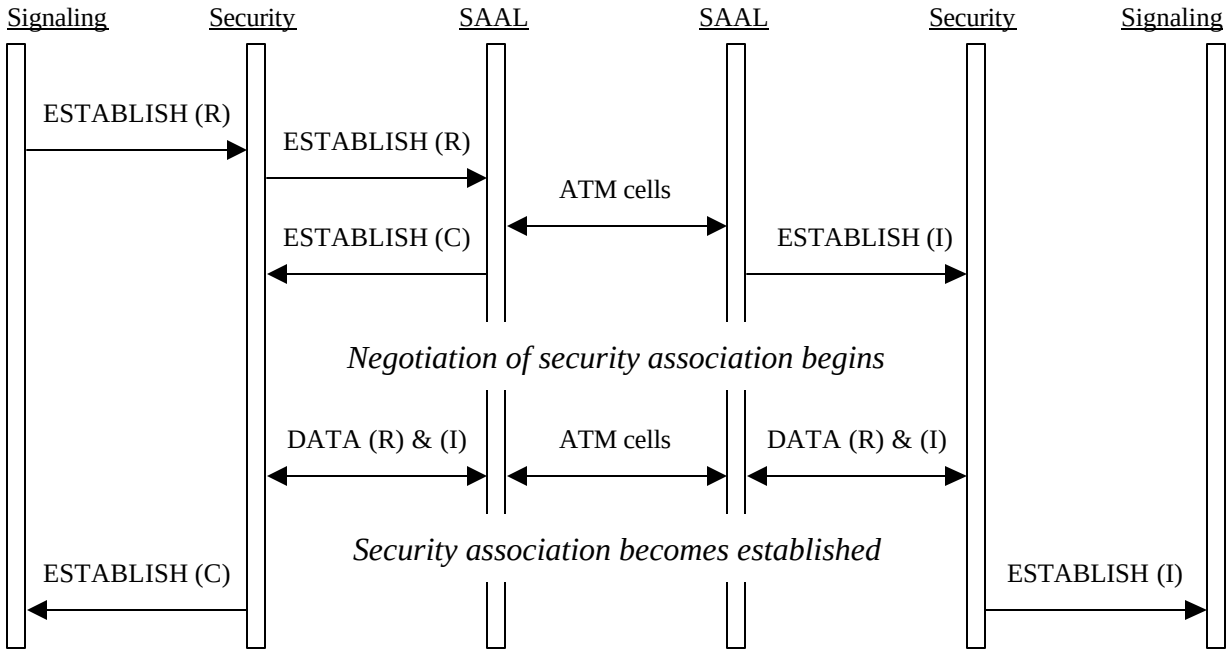

Figure A-1: Connection Establishment With No Data Transfer.

A.1.2 With Data Transfer

This scenario, illustrated in Figure A-2, is similar to the previous one, except that the Signaling layer initiating establishment of the signaling channel desires to send an optional data parameter to the peer Signaling layer. This optional data parameter must be queued while the security association is being negotiated. After the security association becomes established, the optional data parameter is sent to the Signaling layer peer.

The Signaling layer initiating the connection sees the establishment of the signaling channel and the transfer of the optional data parameter as an atomic action. The Signaling layer peer, however, sees a simple establishment of the signaling channel, immediately followed by a data transfer indication.

Figure A-2: Connection Establishment With Data Transfer.

A.2 Data Transfer

In this scenario, illustrated in Figure A-3, the Signaling layer at one end sends a data message to the Signaling layer peer via the signaling channel.

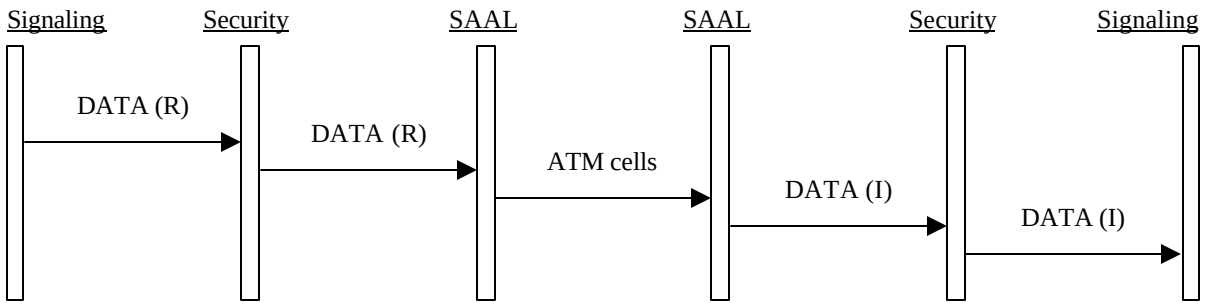

Figure A-3: Signaling Message Exchange.

A.3 Connection Termination

A.3.1 No Data Transfer

In this scenario, illustrated in Figure A-4, the Signaling layer at one end initiates release of the signaling channel. This action does not cause the security association to be broken between the peer Security layers. The goal is to reuse the same security association upon reestablishment of the signaling channel.

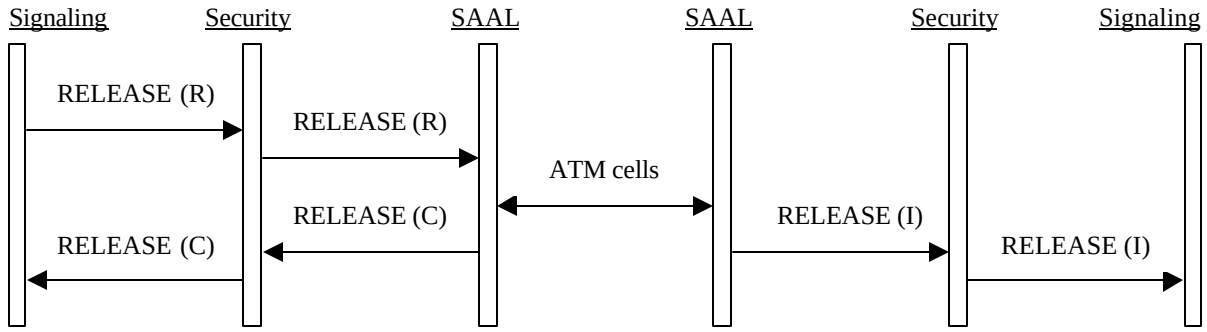

Figure A-4: Connection Termination With No Data Transfer.

A.3.2 With Data Transfer

This scenario, illustrated in Figure A-5, is similar to the previous one, except that the Signaling layer initiating release of the signaling channel desires to send an optional data parameter to the peer Signaling layer. This optional data parameter is sent before the SAAL link is terminated.

The Signaling layer releasing the connection sees the release of the signaling channel and the transfer of the optional data parameter as an atomic action. The Signaling layer peer, however, sees a data transfer indication, immediately followed by a simple release of the signaling channel.

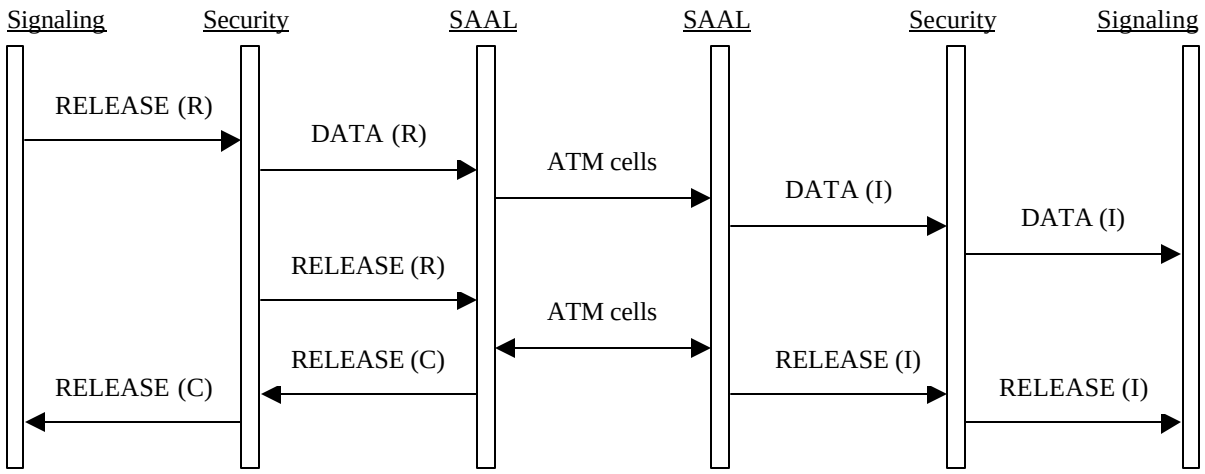

Figure A-5: Connection Termination With Data Transfer.

A.3.3 Link Breakage

In this scenario, illustrated in Figure A-6, some problem causes the SAAL layer peers to detect a link failure. Both Signaling layer peers are notified that the signaling channel is no longer available. This action does not cause the security association to be broken between the peer Security layers. The goal is to reuse the same security association upon reestablishment of the signaling channel.

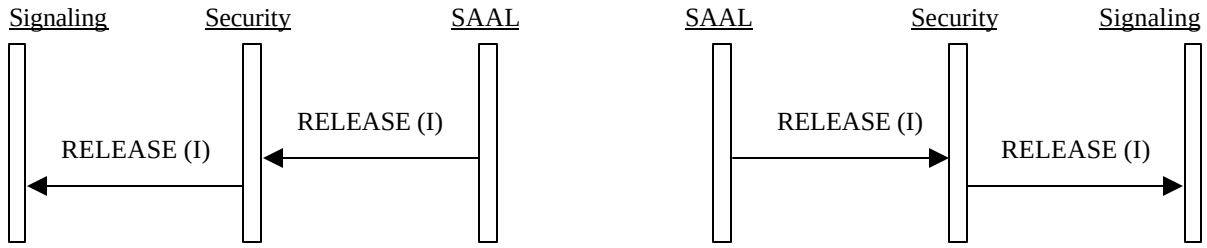

Figure A-6: Link Breakage.

A.3.4 Security Association Invalidation

In this scenario, illustrated in Figure A-7, some problem causes at least one Security layer peer to detect a security anomaly. Use of IKE or SME to manage the security association may require that messages be exchanged across the SAAL link to communicate the anomaly. Eventually, at least one Security layer peer takes the actions specified in Section 3.1.3.9.

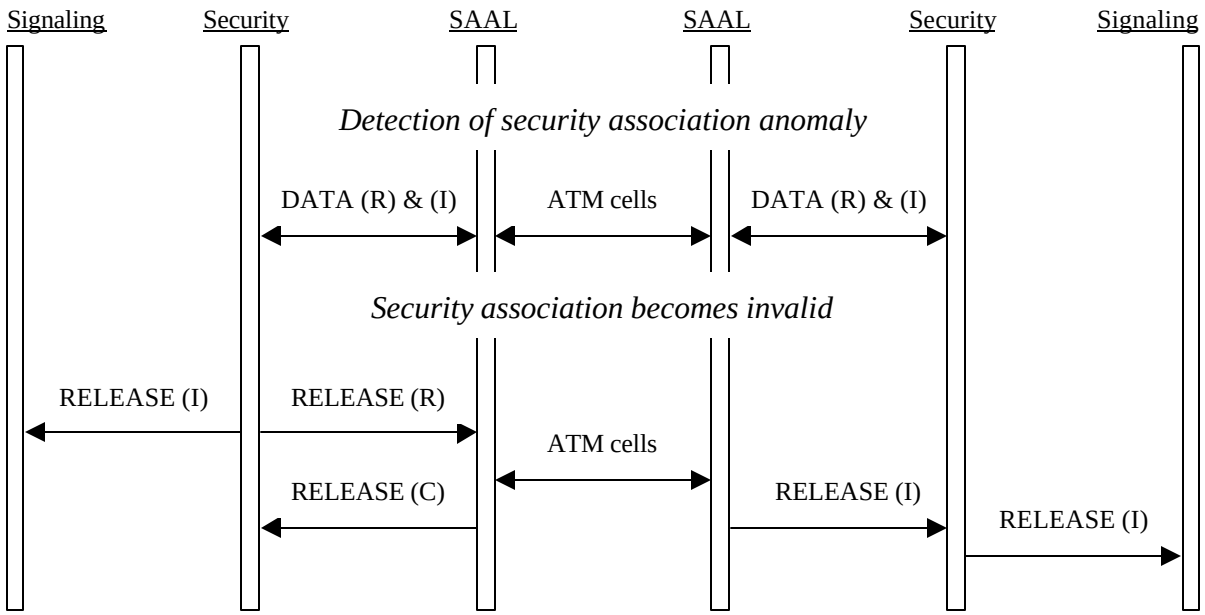

Figure A-7: Security Association Invalidation.