

A SECURITY PROTOCOL PROVIDING QoS IN ATM NETWORKS

Fari Schlake¹, Christoph Ruland²

¹ University of Siegen, Siegen, Germany, 2498 Birchwood Way, Oceanside, CA 92054, USA, f.schlake@cox.net

² University of Siegen, Hoelderlinstrasse 3, D-57068 Siegen, Germany, ruland@nue.et-inf.uni-siegen.de

Abstract - ATM networks are not more or less secure than other networks, but the demand for secure communications is increasing. The ATM forum has defined the Security Specifications using the Three-Way and Two-Way Security Message Exchange Protocols (SME) to support the negotiation and establishment of security services, mechanisms and other parameters needed for a secure connection. Different security mechanisms may degrade the guaranteed QoS through additional delays, error propagations and throughput limitations. The purpose of this paper is to overcome these negative influences of security operations on QoS. A new Three-Way SME_Q Protocol is introduced. It provides the simultaneous capability of implementing required security services, while still offering the user requested QoS parameters during an ATM connection. The Three-Way SME_Q protocol calculates and considers the impact of the security operations on the QoS parameters. This way, only security mechanisms are chosen, which fit into the QoS range requested.

Keywords - High-Speed Networks, Data Communications, ATM, Security Protocols, Quality of Service (QoS), Cryptography

I. INTRODUCTION

The ATM Security Specification Version 1.1 [1], developed by the ATM Forum, defines the security procedures and protocols required to establish security associations in the User and Control Planes of the layered ATM network architecture.

The security protocols define the elements and procedures required for both the *Two-Way* and *Three-Way* Security Message Exchange. These operations are executed and supported by the Security Agents, which are components of the ATM network along the path of a virtual ATM connection. The Three-Way SME at the connection establishment phase is the focus of this work. The case of Two-Way SME is a simplified version of the Three-Way SME and has been neglected here because of the length limitations of this paper and for redundancy reasons. The UNI Signaling Specification 4.0 [2] and ITU-T Recommendations Q.2931 [3] define the required elements and procedures for the negotiation and definition of requested Quality of Service for the establishment phase of a particular ATM connection. Cell Delay Variation (CDV), Cell Transit Delay (CTD) and Cell Loss Ratio (CLR) are defined to be negotiable QoS parameters and specifically selected by the end user at the connection establishment phase of a virtual ATM connection. The QoS has an end-to-end characteristic for the period of the

connection. This means, the end user will not experience any degradation of performance below the requested parameters for the life of the established connection.

The security services and operations during a connection, however, introduce degradations to these requested parameters in form of additional delays and cell losses according to the selected mode of operation ([8]) for the particular security mechanism. The Confidentiality and Data Integrity security mechanisms are of importance here because of their on-line performance during the data transfer phase.

The existing security protocols to date, however, do not take these degradations of requested and agreed upon Quality of Service into account.

It is the goal of this research work to bring the separately defined and standardized QoS signaling procedures and the ATM Security protocols together. A solution is developed, which allows the execution of the security services and mechanisms throughout an ATM connection without exceeding the established QoS during the life of the connection. The user will not then experience any performance decrease (loss of QoS) due to the governing security protocols and policy.

The proposed solution extends the existing ATM Three-Way Security Message Exchange protocol. The new Three-Way SME_Q protocol ensures the execution of the security services while providing the requested Quality of Service for the particular connection.

II. DESIGN REQUIREMENTS AND CONSTRAINTS

The new QoS provisioning Three-Way (In-Band) Security Message Exchange protocol (Three-Way SME_Q) is designed to comply with the following constraints and requirements:

- The Three-Way SME_Q should be a proposed extension and enhancement to the existing Three-Way SME defined in the ATM Security Specification 1.1 [1].
- The Three-Way SME_Q should be compliant with the signaling protocols defined in the Signaling Specification 4.0 [2] and the ITU-T Recommendation Q.2931 [3] for the negotiation of the QoS parameters.
- The QoS should still keep its end-to-end characteristic. This means, the user should not experience a decrease of network performance (loss of QoS) based on the security operations during the data transfer phase.
- The QoS parameter degradation caused by performing security operations should be considered and compensated at

the connection establishment phase.

- QoS degradation values, which are specific to each network element containing a security agent and selected security service and algorithm, are pre-calculated and available to the new Three-Way SME_Q protocol in the SAC_Q Table (Table 1). This table is generated and maintained by the security management system.

III. CURRENT APPROACH

Quality of Service and Security aspects of ATM networks have been paid through attention by researchers in recent years but rather independently. Different standardization bodies have also contributed to these efforts with recent specifications for each distinguished field. Among which, the ATM Security Specification Version 1.1 [1], the UNI Signaling Specification 4.0 [2] and the ITU-T Recommendations Q.2931[3] build the basis of this research work.

Other published scientific works deal with various mechanisms for improving the overall QoS of the ATM networks and do not concurrently deal with the security aspects and their impact. Some have developed new mechanisms for scheduling and data movement at the end system level [4], some at the application level [5] and the others take a "neural fuzzy" approach at the traffic management level [6].

III. THE THREE-WAY SME_Q

A. Three-Way SME_Q Protocol Basics

This section defines the proposed changes and additions to the SSIE structure and Three-Way SME procedures used in the current approved ATM Security Specification. These additions and proposed changes grant the QoS provisioning capability to the current Three-Way SME and as a whole make the Three-Way SME_Q protocol.

1) *SA Characteristics with regard to QoS (SAC_Q)*: The Three-Way SME_Q requires the availability of the QoS degradation values of the negotiable parameters caused by the security operations in the security agent of a network element. These values build the Security Agent Characteristics with regard to QoS (SAC_Q) and are available in the SAC_Q Table. They provide the rates of performance decrease caused by different combinations of security algorithms and modes of operation for each required security service. Table 1 illustrates an example of a SAC_Q Table presenting a Security Agents' Characteristics with regard to QoS. This information is prepared and maintained by security management.

These SAC_Q_(I,R) parameters are: CDV_(I,R), CTD_(I,R) and CLR_(I,R) for the initiating (I) or responding (R) SAs respectively. For a required security service in the case of Three-Way

SME_Q, a SA_(I,R) negotiates the security algorithms and options according to its SAC_Q_(I,R) Table values with the partner SA in order to still meet or exceed the requested QoS.

2) *Security Association Section (SAS_Q) of the SSIE_Q*: The only proposed change to the SAS of current SME protocol is that the octet 5.9 (*Security Service Data* field) of SAS_Q is not optional and is a required field for the negotiation of the security services according to the SAC_Q values. The changes to this field are defined below. The other fields of the SAS_Q will remain identical to the current SAS.

3) *Security Service Data Section of the SSIE_Q*: The first three octets of the current Security Service Data Section need modifications to allow the operations of the In-Band SME_Q.

3.1) *Security Message Exchange Format of the SSIE_Q*: In octet 5.9 an additional code should be considered for the three-way SME_Q as optional SME types.

3.2) *Security Entity Identifier of the SSIE_Q*: If the Three-Way SME_Q option is selected, this octet should ONLY contain the Security Entity Identifier of the initiating SA which is selecting or is providing negotiation options to a peer SA. This means, in the FLOW1_3WE of the SME_Q protocol, this octet contains the identity of the initiating SA. This way, the peer SA can recognize the owner of the SAC_Q values, which would be provided along with the SAS_Q in the next field.

4) *Security Service Specification Section of the SSIE_Q*: The *Security Service Algorithm Section* of this section is a required field for the Three-Way SME_Q, which allows the negotiation of the algorithms and modes of operation according to their SAC_Q values.

4.1) *Data Confidentiality Algorithm of the SSIE_Q*: Fig.1 illustrates the additional octets required as an extension to the current *Data Confidentiality Algorithm* primitive for the In-Band SME_Q.

The octets x.9.x and x.10.x are proposed additions to the current structure of this primitive. Octets x.9.x are assigned to

Table1
Example of a SA_(I,R)'s SAC_Q_(I,R) Table

SAC_Q _(I,R) Table						
Security Service	Encryption	Mode	MAC	CDV _(I,R) (ms)	CTD _(I,R) (ms)	CLR _(I,R) (10 ⁻⁶)
Confidentiality	DES	CBC	—	0.3	3	9
	DES	Counter	—	0.1	1	12
	Triple-DES	CBC	—	0.7	7	7
	Triple-DES	Counter	—	0.4	4	8
	FEAL	CBC	—	0.9	9	7
	FEAL	Counter	—	0.8	8	8
Data Integrity	—	—	DES/CBC	0.2	3	9
	—	—	H-MD5	0.4	7	10
	—	—	H-SHA-1	0.5	4	7
	—	—	FEAL/CBC	0.6	9	8

CDV_(I,R). First octet states a predefined identifier for CDV_(I,R). Its value is coded according to the defined format of the UNI Signaling Specification 4.0 using three bytes. Octets x.10.x are assigned to CTD_(I,R). First octet states a predefined identifier for CTD_(I,R). Its value coded according to the defined format of the ITU-T Recommendation Q.2931 using two bytes. These values indicate the QoS degradation characteristic of in the preceding fields selected combination of algorithm and mode of operation for the SA identified in the *Security Entity Identifier* of the Security Service Data Section.

4.2) *Data Integrity Algorithm of the SSIE_Q*: The octets x.6.x and x.7.x are proposed additions to the current structure of this primitive and carry the same information as the above case illustrated in Fig. 1. Octets x.6.x are assigned to CDV_(I,R). First octet states a predefined identifier for CDV_(I,R). Its value is coded according to the defined format of the UNI Signaling Specification 4.0 using three bytes. Octet x.7.x are assigned to CTD_(I,R). First octet states a predefined identifier for CTD_(I,R). Its value is coded according to the defined format of the ITU-T Recommendation Q.2931[3] using two bytes. These values indicate the QoS degradation characteristic of in the preceding fields selected combination of algorithm and replay mode for the SA identified in the *Security Entity Identifier* of the Security Service Data Section.

B. The Three-Way SME_Q Protocol

Fig. 2 illustrates the time diagram of the proposed In-Band (Three-Way) SME_Q. The proposed protocol procedures of each SA are described in the following sections.

1) *Initiating Security Agent Procedures*: The following procedures are proposed in addition to the already existing procedures in the ATM Security Specification Version 1.1[1].

Upon receipt of a SETUP message on the initiating side (Fig 2, 1), the SA_(I) notes the requested acceptable QoS parameter val-

Bits								Octets
8	7	6	5	4	3	2	1	
x	x	x	x	x	x	x	x	x.9
Cell Delay Variation (CDV _(I,R)) Identifier								
Cell Delay Variation (CDV _(I,R))								x.9.1
CDV _(I,R) (continued)								x.9.2
CDV _(I,R) (continued)								x.9.3
x	x	x	x	x	x	x	x	x.10
Cell Transit Delay (CTD _(I,R)) Identifier								
Cell Transit Delay (CTD _(I,R))								x.10.1
CTD _(I,R) (continued)								x.10.2

Fig. 1. Additional Octets of the SME_Q Data Confidentiality Algorithm Primitive

ues indicated in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element* for that connection. It compares the SAC_{Q(I)} values for the security services, which are to be provided with the indicated acceptable parameters.

If none of the security options in the table can exceed or in the case of CLR meet the acceptable values, the SA_(I) rejects the connection request with the cause *SME_Q failure for the requested QoS*. Otherwise, the SA_(I) forwards the SETUP message toward the responding endpoint.

Upon receipt of a CONNECT message on the initiating side (Fig 2, 5), the SA_(I) prevents the CONNECT message to proceed to the initiating endpoint. It analyzes the cumulative QoS parameters indicated in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element*. It compares the SAC_{Q(I)} values for the security services which are to be provided with the, at the downstream saved, acceptable parameters and the received cumulative values from the responding endpoint.

The SA_(I) decides on a corresponding alternative list of the security algorithms and modes of operation (in case of Confiden-

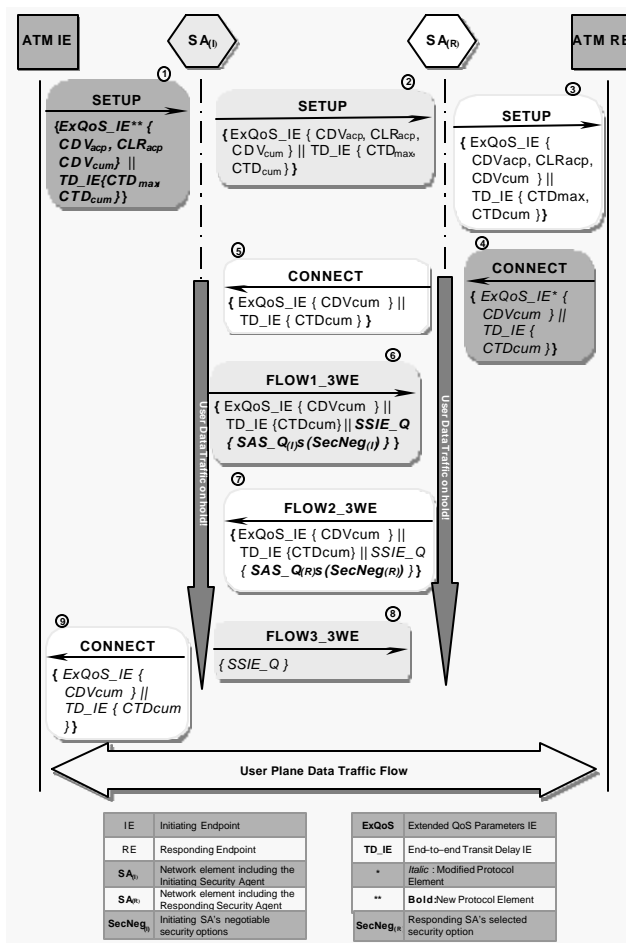


Fig. 2. The Three-Way (In-Band) SME_Q

tiality security service) to include in the FLOW1_3WE for negotiation with the SA_(R) (Fig 2, 6). The suggested options are to be selected, so that, the worst option, which is the one with the highest values of degradation according to the SAC_Q_(I) table, if added to the received cumulative values, would still result to lower rates than the indicated acceptable parameters by the initiating endpoint. The CLR value of each suggested option, which is not a cumulative value, should either be equal to or lower than the saved acceptable CLR indicated originally by the initiating endpoint.

If more than one service are requested to be supported between SA_(I) and SA_(R), only the parameters for confidentiality and data integrity, if applicable, should be considered in the calculations (S1: first security service and S2: second security service). As described before, only these two services impact the network performance during the user data transfer phase. Table 2 defines the abbreviations used in the following equations.

$$CDV_{(I)} = CDV_{(I),S1} + CDV_{(I),S2} \quad (1)$$

$$CTD_{(I)} = CTD_{(I),S1} + CTD_{(I),S2} \quad (2)$$

$$CLR_{(I),S1} \leq CLR_{acp} \quad (3)$$

$$CLR_{(I),S2} \leq CLR_{acp} \quad (4)$$

$$CDV_{cum} + CDV_{(I)} < CDV_{acp} \quad (5)$$

$$CTD_{cum} + CTD_{(I)} < CTD_{max} \quad (6)$$

In this case, the SA_(I) should first prioritize these services. It first selects its preferred options for the service with the higher pri-

Table2
The QoS Parameter Definition for SA_(I,R)

QoS Parameter	Definition
CDV _{acp}	The user requested (acceptable) Cell Delay Variation for the connection.
CDV _{cum}	The cumulative value of the Cell Delay Variation along the connection.
CDV _(I,R)	The SA _(I,R) introduced Cell Delay Variation value according to its SAC_Q table.
CDV _{(I,R),S1}	The SA _(I,R) introduced Cell Delay Variation value according to its SAC_Q table for the first service.
CDV _{(I,R),S2}	The SA _(I,R) introduced Cell Delay Variation value according to its SAC_Q table for the second service.
CTD _{max}	The user requested (maximum allowable) Cell Transfer Delay for the connection.
CTD _{cum}	The cumulative value of the Cell Transfer Delay along the connection.
CTD _(I,R)	The SA _(I,R) introduced Cell Transfer Delay value according to its SAC_Q table.
CTD _{(I,R),S1}	The SA _(I,R) introduced Cell Transfer Delay value according to its SAC_Q table for the first service.
CTD _{(I,R),S2}	The SA _(I,R) introduced Cell Transfer Delay value according to its SAC_Q table for the second service.
CLR _{acp}	The user requested (acceptable) Cell Loss Ratio for the connection.
CLR _{(I,R),S1}	The SA _(I,R) introduced Cell Loss Ratio according to its SAC_Q table for the first service.
CLR _{(I,R),S2}	The SA _(I,R) introduced Cell Loss Ratio according to its SAC_Q table for the second service.

ority (S1) and then optimizes the selection of the options for the second service (S2) according to the chosen selections for the first option. In addition to (4), the following should apply:

$$CDV_{(I),S2} < CDV_{acp} - CDV_{cum} - CDV_{(I),S1} \quad (7)$$

$$CTD_{(I),S2} < CTD_{max} - CTD_{cum} - CTD_{(I),S1} \quad (8)$$

Naturally, in addition to (3), the same equations apply for the first service:

$$CDV_{(I),S1} < CDV_{acp} - CDV_{cum} - CDV_{(I),S2} \quad (9)$$

$$CTD_{(I),S1} < CTD_{max} - CTD_{cum} - CTD_{(I),S2} \quad (10)$$

The above set of equations should be true for all suggested options. If any option does not satisfy the above, it should not be considered and provided for the negotiation.

In case only one of the services (confidentiality or data integrity) is supported, the S2 parameters are equal to zero. This would be a special case of the above equations.

After the selection of the security options, the SA_(I) generates the appropriate SAS_Q_{(I)S}.

The SA_(I) starts the Three-Way SME_Q with the FLOW1-3WE including the SecNeg_(I) (Fig 2, 6). The suggested security options are contained in the generated SAS_Q_{(I)S} with the SAC_Q_(I) values of each option. The *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element* received in the CONNECT message from the responding endpoint are also appended.

Upon receipt of the FLOW2_3WE from the SA_(R) (Fig 2, 7), the SA_(I) parses the SAC_Q_(R) values of each negotiated security option from the received SAS_Q_{(R)S} in the SecNeg_(R). It then adds these and the local SAC_Q_(I) values to the cumulative parameters of the *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element*.

$$CDV_{cum} [:= CDV_{cum (received)} + CDV_{(I)} + CDV_{(R)}] \leq CDV_{acp} \quad (11)$$

$$CTD_{cum} [:= CTD_{cum (received)} + CTD_{(I)} + CTD_{(R)}] \leq CTD_{max} \quad (12)$$

if the above equations are satisfied, the SA_(I) accepts the chosen options, it completes the Three-Way SME_Q by sending the FLOW3_WE to the SA_(R).

The SA_(I) now forwards the updated CONNECT message with the new cumulative values to the initiating endpoint (Fig 2, 9). It then unblocks the user data transfer (Fig 2, 8).

If the above equations were not satisfied, the SA_(I) clears the connection with the cause *SME_Q failure for the requested QoS*.

2) *Responding Security Agent Procedures*: The following procedures are proposed in addition to the already existing procedures in the ATM Security Specification Version 1.1[1].

Upon receipt of a SETUP message on the responding side (Fig 2, 2), the SA_(R) notes the requested acceptable QoS parameter

values indicated in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element* for that connection. It compares the $SAC_{Q(R)}$ values for the security services, which are to be provided with the indicated acceptable parameters.

If none of the security options with the $SAC_{Q(R)}$ values could exceed or in the case of CLR meet the user requested acceptable values, the $SA_{(R)}$ rejects the connection request with the cause *SME_Q failure for the requested QoS*. Otherwise, the $SA_{(R)}$ forwards the SETUP message to the responding endpoint. Upon receipt of a CONNECT message on the responding side (Fig 2, 4), the $SA_{(R)}$ blocks the user data transfer and forwards the CONNECT message toward the initiating endpoint. It then awaits the invocation of the SME_Q from the initiating side over the user connection.

Upon receipt of the FLOW1-3WE from $SA_{(I)}$ (Fig 2, 6), the $SA_{(R)}$ decides which security options to accept.

FLOW1-3WE includes the $SA_{(I)}$ suggested negotiable security options in the $SecNeg_{(I)}$ containing the corresponding $SAC_{Q(I)}$ QoS degradation values of the options. It also holds the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element*.

The $SA_{(R)}$ compares its local $SAC_{Q(R)}$ values for the security services, which are to be provided, in addition to the sum of the received cumulative QoS parameters and $SAC_{Q(I)}$ degradation values of suggested options with the previously noted acceptable QoS parameters values of this connection.

The $SA_{(R)}$ selects the one option for each security service so that if added to the sum of the received cumulative values and the degradation values of the $SA_{(I)}$ it would still meet or result to lower rates than the saved acceptable parameters on the downstream for this connection. The CLR value of the option for this security service, which is not a cumulative value should either meet or be lower than the desired acceptable CLR.

In addition to (1) and (2) the following should also be satisfied according to the variable definitions in (11) and (12). Table 2 defines the abbreviations used in the following equations.

$$CDV_{(R)} = CDV_{(R),S1} + CDV_{(R),S2} \quad (13)$$

$$CTD_{(R)} = CTD_{(R),S1} + CTD_{(R),S2} \quad (14)$$

In this case, the $SA_{(R)}$ should process the received $SAS_{Q(I),S}$ in descending order, which is the order of preference of $SA_{(I)}$ for the requested services. It first examines the preferred options for the service with the higher priority (S1) and then optimizes the selection of the options for the second service (S2) according to the selections for the first option. The parameters of the chosen security service S2 are calculated according to the following equations with respect to (11) and (12).

$$CDV_{(R),S2} < CDV_{acp} - CDV_{cum} - CDV_{(I)} - CDV_{(R),S1} \quad (15)$$

$$CTD_{(R),S2} < CTD_{max} - CTD_{cum} - CTD_{(I)} - CTD_{(R),S1} \quad (16)$$

$$CDV_{(R),S1} < CDV_{acp} - CDV_{cum} - CDV_{(I)} - CDV_{(R),S2} \quad (17)$$

$$CTD_{(R),S1} < CTD_{max} - CTD_{cum} - CTD_{(I)} - CTD_{(R),S2} \quad (18)$$

The $SA_{(R)}$ processes the algorithm options also in descending order, which is the order of preference of $SA_{(I)}$. In case only one service is supported, the S2 parameters are equal to zero. This would be a special case of the above equations.

The $SA_{(R)}$ then generates the $SecNeg_{(R)}$ with the new $SAS_{Q(R),S}$ and indicates its $SAC_{Q(R)}$ values for the security options chosen. The selected option for each security service is communicated to the $SA_{(I)}$ in the $SecNeg_{(R)}$ of FLOW2_3WE (Fig 2, 7). Upon completion of the Three-Way SME_Q the $SA_{(R)}$ unblocks the user data transfer. If the $SA_{(R)}$ could not support the options according to the QoS requirements the $SA_{(R)}$ should clear the connection with the cause *SME_Q failure for the requested QoS*.

IV. SUMMARY AND CONCLUSION

With the developed SME_Q Protocol the desired Quality of Service of an ATM connection will no longer have to be sacrificed in favor of the implemented security measures. The increasing necessity of security can be satisfied in the Quality of Service and bandwidth demanding applications such as Video Conferencing, Telemedicine and streaming media.

REFERENCES

- [1] ATM Forum, "ATM Security Specification", Ver. 1.1, af-sec-0100.002, March 2001
- [2] ATM Forum, "ATM User-Network Interface (UNI) Signaling Specification", Ver. 4.0, af-sig-0061.000, July 1996
- [3] ITU-T, "B-ISDN Application Protocols for Access Signaling", ITU-T Recommendation Q.2931, Feb 1995
- [4] R. Gopalakrishnan, et al, "Efficient User-Space Protocol Implementations with QoS Guarantees Using Real-Time Upcalls", IEEE/ACM Transactions on Networking, Vol. 6, No. 4, August 1998
- [5] David K. Y. Yau, Simon S. Lam, "Migrating Sockets - End System Support for Networking with Quality of Service Guarantees", IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998
- [6] Ray-Guang Cheng, et al, "A QoS-Provisioning Neural Fuzzy Connection Admission Controller for Multimedia High-Speed Networks", IEEE/ACM Transactions on Networking, Vol. 7, No. 1, February 1999
- [7] ATM Forum, "Traffic Management Specification", Version 4.0, af-tm-0121.000, March 1999
- [8] ISO/IEC, "Modes of Operation for an n-bit Block Cipher", ISO/IEC 10116, Second Edition.