Optimal Consumer-Centric Delay-Efficient Security Management in Multi-Agent Networks - A Game & Mechanism Design Theoretic Approach

by

Farimehr Schlake

Dissertation to be submitted to the faculty of
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Electrical Engineering

Lamine M. Mili, Chair
Ing-Ray Chen
C. Robert Clauer
Virgilio Centeno
Mohamed Eltoweissy

April 10, 2012
Falls Church, VA 22043

Key Words:

Game Theory, Mechanism Design Theory, Incentive Compatibility, Dominant Strategy, Bayesian Games, Optimization, Performance, Security Protocols, Delay, Quality of Service (QoS), ATM, IPSEC

Optimal Consumer-Centric Delay-Efficient Security Management in Multi-Agent Networks - A Game & Mechanism Design Theoretic Approach

Farimehr Schlake

Abstract

The main aspiration behind the contributions of this research work is the achievement of simultaneuos delay-efficiency, autonomy, and security through innovative protocol design to address complex real-life problems. To achieve this, we take a holistic approach. We apply theoretical mathematical modeling implementing implications of social-economic behavioral characteristics to propose a cross-layer network security protocol. We further complement this approach by a layer-specific focus with implementations at two lower OSI layers.

For the cross-layer design, we suggest the use of game and mechanism design theories. We design a network-wide consumer-centric and delay-efficient security protocol, DSIC-S. It induces a Dominant Strategy Incentive Compatible equilibrium among all rational and selfish nodes. We prove it is network-wide socially desirable and Pareto optimal. We address resource management and delay-efficiency through synergy of several design aspects. We propose a scenario-based security model with different levels. Furthermore, we design a valuation system to integrate the caused delay in selection of security algorithms at each node without consumer's knowledge of the actual delays. We achieve this by incorporating the consumer's valuation system, in the calculation of the credit transfers through the Vickrey-Clarke-Groves (VCG) payments with Clarke's pivotal rule. As the utmost significant contribution of this work, we solve the revelation theorem's problem of misrepresentation of agents' private information in mechanism design theory through the proposed design. We design an incentive model and incorporate the valuations in the incentives. The simulations validate the theoretical results. They prove the significance of this model and among

others show the correlation of the credit transfers to actual delays and security valuations.

In the layer-specific approach for the network-layer, we implement the DSIC-S protocol to extend current IPsec and IKEv2 protocols. IPsec-O and IKEv2-O inherit the strong properties of DSIC-S through the proposed extensions.

Furthermore, we propose yet another layer-specific protocol, the SME_Q, for the datalink layer based on ATM. We develop an extensive simulation software, SMEQSIM, to simulate ATM security negotiations. We simulate the proposed protocol in a comprehensive real-life ATM network and prove the significance of this research work.

Dedication

I would like to dedicate this work first and foremost to my loving family, whose teachings, devotion, love and belief in me is that what has made me the person, who I am …

Secondly, I dedicate this accomplishment to all people of values! In specific to those, who had to go long ways in effort and time to stay true to their principles! It especially goes to all the women in underrepresented walks of life, who have to tame even harder waves, while having to take on several turbulent seas in parallel, to be able to get to the same destination! To those, who aim high, whom no road is long enough, no wave is hard enough and no storm is powerful enough to throw them out of their genuine pathway...

In my family, it goes most heartedly to my loving dad, Mr. Dariush Amirsehi, a multi-talented genius, an astonishing charismatic mentor, and the most amazing humanitarian caring and supportive soul to ever have been around. He never ceased to pass on the strong international educational tradition and pride of both sides of my family, he so strongly believed in. “Never let anything come between you and your education. There is simply no life challenge you cannot master to make it happen!” was among many voices of his invaluable wisdom, which he so generously and passionately always extended to us. Little that he knew, how this particular one turned to become my life’s motto, since I last heard it from him at the young age of 12! Dad you are the main pillar of my constitution. Thanks for all I got from you and carry in me of you …

I’d like to also share this joy with my precious mom, Mrs. Farideh Sheybani-Amirsehi, whose enormous belief in me and my strengths gave me the wings to fly freely and have the opportunity to make pieces of her own dreams come true through me. I started my journey at Virginia Tech at her loving side and with her incredible emotional support. Unfortunately once again, one of those life challenges prevented her being at my side for the harvest. Mom you are always in my heart ...

I especially dedicate this work to the love of my life, my husband and best friend, Dr.-Ing. Oliver Schlake. Thanks for so highly believing in me. I am deeply grateful for your embracing the true me

and taking pride in my accomplishments. It is our shared love for exploration of higher knowledge and wisdom, and its special significance for both of us, which makes this work even a sweeter achievement for me. This is for you and our little family... This is just a beginning; I look forward to sharing the rest of my life with you and “boldly go where no two have gone before ...”!

In view of the same dedication, this dearly goes to the two new inseparable souls in my life, our two precious daughters, Faye Shahdouneh and Fawn Dordouneh. Thanks for your innocent hugs, insurmountable love and loving support notes, “Mom you can do it”! May my experience become a leading light in your future paths …

I also sincerely dedicate this work to my loving grandparents, Mrs. Ezat and Mr. Rahmatollah Sheybani, for their influence on me as a person through their invaluable role model teachings of selflessness, love, generosity and higher principles in life. You both continue to live in me ...

Last but not least, my dearest siblings with the passions they share with their little sister, next to the love and care we embrace! My brother, Cyrus Amir, Ph.D., with whom I share my love of logic, math, liberating sense of humor and inner peace; my sister, Mrs. Farinaz Amirsehi, with whom I share the curiosity about the mysteries of life and human psyche!

Acknowledgement

I would like to thank everyone, who supported me and my Ph.D. efforts in any special way throughout this adventurous and precious endeavor of mine.

Special thanks go to Dr. Lamine Mili, who took on my special case and grew to believe in me and the depth of my potentials through this process. He bravely supported me with my daring ideas of research and let me fly to the distance! I'd like to thank my doctoral committee, Dr. Ing-Ray Chen, Dr. Robert Clauer, Dr. Virgilio Centeno and Dr. Mohamed Eltoweissy for their kind support throughout my process.

Every journey has its beginnings, which deserve to be remembered at the completion! I'd also like to thank all the people, who in some way were involved in my path of admissions to Virginia Tech's ECE Doctrol program. Special thanks go to Dr. Zaghloul, Dr. Rahman, Dr. Da Silva, Dr. Tranter and Cynthia Hopkins.

I would like to also extend my deep appreciations to my external mentors outside of Virginia Tech. Dr. Bharat Bhargava, Professor at Purdue University, CS, for his tremendous belief in me and enormous emotional support throughout my endeavor! I'd like to give special thanks to Dr. Theodore Groves, Professor Emiritus at UCSD, Dept. of Economics, for giving me the extraordinary opportunity to discuss my results of research work with him as the father of one of the main theories I used, the Vickrey-Clarke-Groves, VCG, mechanism. It was an amazing delight to be able to have scientific exchange with the source, as he himself calls it, with the "G"!

Also my thanks go to VT's NCR and Blacksburg graduate school administration for handling the tremendous application work involved throughout such an endeavor.

Leaving the best for last, I would like to round up with the special acknowledgement to my heart for the unconditional emotional support of many precious friends and relatives from around the world throughout this amazing path, whom I share this sweet glory of reach with!

Contents

Table of Content

Figures

List of Figures

Chapter 3

Chapter 5

Chapter 6

Chapter 8

Chapter 9

Chapter 10

Chapter 11

Chapter 12

Chapter 13

Tables

List of Tables

Chapter 5

Chapter 9

Chapter 11

Chapter 13

Abbreviations

List of Abbreviations

Term	Definition
AAL	ATM Adaption Layer
ABR	Available Bit Rate
AES	Advanced Encryption Standard
AH	Authentication Header
ANSI	American National Standards Organization
AOF	Aggregate Objective Function
ATM	Asynchronous Transfer Mode. A high-speed connection-oriented transmission method utilizing 53-byte fixed cells for transport.
$ATM_{(IE)}$	ATM Initiating Endpoint. It is the ATM Device, which starts the connection establishment.
$ATM_{(RE)}$	ATM Responding Endpoint. It is the ATM Device at the destination, which responds to the connection establishment request.
B-ISDN	Broadband Integrated Services Digital Network

CBC Cipher Block Chaining

CBR Constant Bit Rate

CDV Cell Delay Variation

$CDV_{(I)}$ The $SA_{(I)}$ introduced Cell Delay Variation value according to its SAC_Q table.

$CDV_{(I),S1}$ The $SA_{(I)}$ introduced Cell Delay Variation value according to its SAC_Q table for the first service.

$CDV_{(I),S2}$ The $SA_{(I)}$ introduced Cell Delay Variation value according to its SAC_Q table for the second service.

$CDV_{(I)n}$ The $SA_{(I)n}$ introduced Cell Delay Variation value according to its SAC_Q table.

$CDV_{(I)n,S1}$ The $SA_{(I)n}$ introduced Cell Delay Variation value according to its SAC_Q table for the first service.

$CDV_{(I)n,S2}$ The $SA_{(I)n}$ introduced Cell Delay Variation value according to its SAC_Q table for the second service.

$CDV_{(IE)}$ The $ATM_{(IE)}$ introduced Cell Delay Variation value.

$CDV_{(NTWK)}$ The intervening network introduced Cell Delay Variation value.

$CDV_{(R)}$ The $SA_{(R)}$ introduced Cell Delay Variation value according to its SAC_Q table.

$CDV_{(R),S1}$ The $SA_{(R)}$ introduced Cell Delay Variation value according to its SAC_Q table for the first service.

$CDV_{(R),S2}$ The $SA_{(R)}$ introduced Cell Delay Variation value according to its SAC_Q table for the second service.

$CDV_{(R)n}$ The $SA_{(R)n}$ introduced Cell Delay Variation value according to its SAC_Q table.

$CDV_{(R)n,S1}$ The $SA_{(R)n}$ introduced Cell Delay Variation value according to its SAC_Q table for the first service.

$CDV_{(R)n,S2}$ The $SA_{(R)n}$ introduced Cell Delay Variation value according to its SAC_Q table for the second service.

$CDV_{(RE)}$ The $ATM_{(RE)}$ introduced Cell Delay Variation value.

CDV_{acp} The user requested maximum (acceptable) Cell Delay Variation for the connection.

CDV_{cum} The cumulative value of the Cell Delay Variation across the connection.

CDV_{QC} In the QoS Class predefined (acceptable) Cell Delay Variation for the connection.

CFB Cipher Feedback

CLR Cell Loss Ratio

$CLR_{(I)}$ The $SA_{(I)}$ introduced Cell Loss Ratio according to its SAC_Q table.

$CLR_{(I),S1}$ The $SA_{(I)}$ introduced Cell Loss Ratio according to its SAC_Q table for the first service.

$CLR_{(I),S2}$ The $SA_{(I)}$ introduced Cell Loss Ratio according to its SAC_Q table for the second service.

$CLR_{(I)n,S1}$ The $SA_{(I)n}$ introduced Cell Loss Ratio according to its SAC_Q table for the first service.

$CLR_{(I)n,S2}$ The $SA_{(I)n}$ introduced Cell Loss Ratio according to its SAC_Q table for the second service.

$CLR_{(IE)}$ The $ATM_{(IE)}$ introduced Cell Loss Ratio value.

$CLR_{(NTWK)}$ The intervening network introduced Cell Loss Ratio value.

$CLR_{(R)}$ The $SA_{(R)}$ introduced Cell Loss Ratio according to its SAC_Q table.

$CLR_{(R),S1}$ The $SA_{(R)}$ introduced Cell Loss Ratio according to its SAC_Q table for the first service.

$CLR_{(R),S2}$ The $SA_{(R)}$ introduced Cell Loss Ratio according to its SAC_Q table for the second service.

$CLR_{(R)n,S1}$ The $SA_{(R)n}$ introduced Cell Loss Ratio according to its SAC_Q table for the first service.

$CLR_{(R)n,S2}$ The $SA_{(R)n}$ introduced Cell Loss Ratio according to its SAC_Q table for the second service.

$CLR_{(RE)}$ The $ATM_{(RE)}$ introduced Cell Loss Ratio value.

CLR_{acp} The user requested maximum (acceptable) Cell Loss Ratio for the connection.

CLR_{QC} In the QoS Class predefined (acceptable) Cell Loss Ratio for the connection.

CoS Classes of Service

CPS Common Part Sublayer

CS Convergence Sublayer

CTD Cell Transfer Delay

$CTD_{(I)}$ The $SA_{(I)}$ introduced Cell Transfer Delay value according to its SAC_Q table.

$CTD_{(I),S1}$ The $SA_{(I)}$ introduced Cell Transfer Delay value according to its SAC_Q table for the first service.

$CTD_{(I),S2}$ The SA(I) introduced Cell Transfer Delay value according to its SAC_Q table for the second service.

$CTD_{(I)n}$ The $SA_{(I)n}$ introduced Cell Transfer Delay value according to its SAC_Q table.

$CTD_{(I)n,S1}$ The $SA_{(I)n}$ introduced Cell Transfer Delay value according to its SAC_Q table for the first service.

$CTD_{(I)n,S2}$ The $SA_{(I)n}$ introduced Cell Transfer Delay value according to its SAC_Q table for the second service.

$CTD_{(IE)}$ The $ATM_{(IE)}$ introduced Cell Transfer Delay value.

$CTD_{(NTWK)}$ The intervening network introduced Cell Transfer Delay value.

$CTD_{(R)}$ The $SA_{(R)}$ introduced Cell Transfer Delay value according to its SAC_Q table.

$CTD_{(R),S1}$ The $SA_{(R)}$ introduced Cell Transfer Delay value according to its SAC_Q table for the first service.

$CTD_{(R),S2}$ The $SA_{(R)}$ introduced Cell Transfer Delay value according to its SAC_Q table for the second service.

$CTD_{(R)n}$	The $SA_{(R)n}$ introduced Cell Transfer Delay value according to its SAC_Q table.
$CTD_{(R)n,S1}$	The $SA_{(R)n}$ introduced Cell Transfer Delay value according to its SAC_Q table for the first service.
$CTD_{(R)n,S2}$	The $SA_{(R)n}$ introduced Cell Transfer Delay value according to its SAC_Q table for the second service.
$CTD_{(RE)}$	The $ATM_{(RE)}$ introduced Cell Transfer Delay value.
CTD_{cum}	The cumulative value of the Cell Transfer Delay across the connection.
CTD_{max}	The user requested (maximum allowable) Cell Transfer Delay for the connection.
CTD_{QC}	In the QoS Class Predefined (maximum allowable) Cell Transfer Delay for the connection.
Ctot	Cumulative total value of the error term C
DAH	The partial value of the error term D caused by AH
DES	Data Encryption Standard
DESP	The partial value of the error term D caused by ESP
Dgen	The partial value of the error term D caused by general IPSEC processing tasks.
diffServ	Differentiated Services
DSA	The partial value of the error term D caused by SA
DSCP	Differentiated Services Code Point
Dsec	The IPSEC contribution to the value of the error term D.
Dtot	Cumulative total value of the error term D
ECB	Electronic Code Book
ESP	Encapsulating Security Payload
ExQoS_IE	Extended Quality of Service Information Element.
FEAL	Fast Encryption Algorithm
FLOW1_2WE	FLOW 1 of the Two-Way SME
FLOW1_3WE	FLOW 1 of the Three-Way SME
FLOW2_2WE	FLOW 2 of the Two-Way SME
FLOW2_3WE	FLOW 2 of the Three-Way SME
FLOW3_3WE	FLOW 3 of the Three-Way SME
GUI	Graphical User Interface
HMAC	Hash function based Message Authentication Code
ICV	Integrity Check Value
IDU	Interface Data Unit
IE	Information Element
IE	Information Element

IETF	Internet Engineering Task Force
IntServ	Integrated Services
IP	Internet Protocol
IPSEC	IP Security Protocol
IPSEC_Q	The developed protocol guaranteeing QoS based on IPSEC and IP signaling protocols
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
ISDN	Integrated Services Digital Network
ISP	Internet Service Provider
ITU	International Telecommunications Union
MAC	Message Authentication Code
MD5	Message Digest 5
nrt-VBR	Non-real-time Variable Bit Rate
NTWK	Intervening network underlying between the network elements across the ATM connection.
$NTWK_n$	Intervening network number n underlying between the network elements across the ATM connection.
MOP	Multi-objective Optimization Problem
OFB	Output Feedback
OSI	Open Systems Interconnection
PDU	Protocol Data Unit
PHB	Per Hub Behavior
PNNI	Private Network-to-Network Interface
PVC	Permanent Virtual Connection
QoS	Quality of Service
RFC	Request For Comments
RSVP	Resource reSerVation Protocol
Rt-VBR	Real-time Variable Bit Rate
SA (in ATM)	Security Agent
SA (in IP)	Security Association
$SA_{(I)}$	Initiating Security Agent
$SA_{(I)n}$	The Initiating Security Agent number n.
$SA_{(R)}$	Responding Security Agent
$SA_{(R)n}$	The Responding Security Agent number n.

SAC_Q	Security Agent's Characteristics with regard to QoS. The SAC_Q Table is specific to each SA
$SAC\_Q_{(I)}$	The Initiating Security Agent's Characteristics with regard to QoS. The SAC_Q Table, which is specific to the $SA_{(I)}$.
$SAC\_Q_{(I)n}$	The Initiating Security Agent number n's Characteristics with regard to QoS. The SAC_Q Table, which is specific to the $SA_{(I)n}$.
$SAC\_Q_{(R)}$	The Responding Security Agent's Characteristics with regard to QoS. The SAC_Q Table, which is specific to the $SA_{(R)}$.
$SAC\_Q_{(R)n}$	The Responding Security Agent number n's Characteristics with regard to QoS. The SAC_Q Table, which is specific to the $SA_{(R)n}$.
SAD	Security Association Database
SAP	Service Access Point
SAR	Segmentation and Reassembly
SAS	Security Association Section
SAS_Q	Security Association Section of SME_Q
SAS_QC	Security Association Section of SME_QC
SDH	Synchronous Digital Hierarchy
SDU	Service Data Unit
$SecNeg_{(I)}$	Initiating SA's negotiable security options.
$SecNeg_{(R)}$	Responding SA's selected security option.
$SecOpt_{(I)}$	Initiating SA's selected security option.
SHA-1	Secure Hash Algorithm 1
SLA	Service Level Agreement
SME	Security Message Exchange
SME_Q	Security Message Exchange protocol with regard to QoS. A new protocol including proposed enhancement to the existing SME to provide Quality of Service.
SME_QC	Security Message Exchange protocol with regard to QoS Classes. A new protocol including proposed enhancement to the existing SME to provide Quality of Service based on QoS Classes.
SMEQSIM	The developed simulator for the SME_Q protocols
SONET	Synchronous Optical Network
SPD	Security Policy Database
SPI	Security Parameters Index
SS7	Signaling System Number 7
SSCS	Service-Specific Convergence Sublayer
SSIE	Security Services Information Element

SSIE_Q	Security Service Information Element of the SME_Q
STM	Synchronous Transfer Mode
SV	State Vector
SVC	Switched Virtual Connection
TCA	Traffic Conditioning Agreement
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TD_IE	Transit Delay Information Element
TTL	Time To Live
UBR	Unspecified Bit Rate
UDP	User Datagram Protocol
UNI	User-to-Network Interface
VBR	Variable Bit Rate
VC	Virtual Channel
VCC	Virtual Channel Connection
VCI	Virtual Channel Identifier
VP	Virtual Path
VPI	Virtual Path Identifier
WFQ	Weighted Fair Queuing
WS	Weighted Sum Pareto Frontier Generation Method
Θ_i	$:= [\underline{V}_A , \bar{V}_A] \times [\underline{D} , \bar{D}]$. It is the set of all possible types of SA_i.
S_i	Set of possible actions or pure strategies of SA_i, with $i \in N= \{1, 2, ..., n\}$.
θ	$:= (\theta_1, \theta_2, ... , \theta_n)$. It is a profile of types of all SAs.
$s = (s_i , s_{\_ i})$	$:= (s_1 , s_2 , \dots , s_n)$. It is a profile of announced types or strategies of all SAs.
Θ	$:= \Theta_1 \times \Theta_2 \times \dots \times \Theta_n$. It is the set of type profiles of all SAs.
$\theta_{\_ i}$	$:= (\theta_1, \dots , \theta_{i-1}, \theta_{i+1}, ... , \theta_n)$. It is a profile of types of all SAs without SA_i.
$s_{\_ i}$	$:= (s_1 , s_{i-1} , s_{i+1}, \dots , s_n)$. It is a profile of announced types or strategies of all SAs without SA_i.
$\Theta_{\_ i}$	$:= \Theta_1 \times \Theta_{i-1} \times \Theta_{i+1} \times \dots \times \Theta_n$. It is the set of all profiles of all types of all SAs without SA_i.
$S_{\_ i}$	$:= S_1 \times S_{i-1} \times S_{i+1} \times \cdots \times S_n$. It is set of all possible actions or pure strategies of all SAs without SA_i.
X	Set of all outcomes
$f(\theta)$	$:=(k(\theta), t(\theta))$. It is the outcome of the SCF for type profile θ.

$k(\theta)$	Allocation vector for type profile θ.
$t(\theta)$	Payment vector for type profile θ.
$t_i(\theta)$	Payment to SA_i for type profile θ.
u_i	$:= \Theta \times X \rightarrow \mathbb{R}$. It is the payoff and utility function of SA_i.
$v_i(k(\theta), \theta i)$	The valuation of SA_i for allocation function $k(\theta)$ & type θ_i.

Chapter

1

Introduction

1.1 Overview and Motivation

Never before have the global health, economic, technological, educational and political operations been this dependent on the integrated digital voice, data and video communications. Multimedia applications have found their commonplace in both private and public business sectors. Applications from power and smart grid communications, telemedicine, electronic commerce, video conferencing and distant learning to video-on-demand, movie and interactive gaming have become steady parts of daily life.

This vital reliance introduces even more critical requirements on the security, privacy and protection of the communication networks and simultaneously on the level of Traffic Quality of Service (QoS) they can offer. The world's political events of the recent years are self-explanatory confirmation for the greater need for privacy protection and security in communications and the realization of the need to address the social component in addition to the technological aspects.

The technology cannot be observed in isolation. Social economic aspects constitute an important part of this communications network. Human behavioral characteristics of selfishness and rationality affect decision making process in management and parameter selection at the technical level. By the same token, economics and market driven forces impact these implementations profoundly.

The hybrid and heterogeneous nature of the world's communications network comprising of various technologies, underline this necessity to ensure the attention to every single technology in this landscape and not only to a few selected ones. In some network security research today this diversity is used as an added benefit for stronger protection against attacks. Altyough historically, now and then, one or the other technology gets the most attention for a period of time!

At the same time, the intolerance to delay and loss at different levels demands special attention to implementations for interactive and real-time applications. In order to keep these daily critical operations in tact, the quality of service provided by the underlying communication network infrastructure should be managed and controlled. Network security and QoS need to be paid attention to hand-in-hand and in a concert. New shifts of realms and ever expanding cutting-edge networking technologies such as cloud computing make this necessity even more critical by introducing new services as utilities.

All of these impose the need for a holistic approach. Mathematical modeling and design of delay-efficient security protocols are needed, which enable this network-wide control and management naturally and truthfully, given the social behavioral and economics characteristics. The consumers should be rest assured that their desired security services for each connection are delivered within the promised levels. These innovative protocols should establish a natural equilibrium among the nodes, so they find it to their best advantage to follow a socially desired and Pareto optimal strategy for their actions in terms of providing network security and delay efficiency. Both cross layer attention to security protocols and proprietary implementation at each specific layer is needed. Existing security protocols need to be updated in this respect and new ones should be innovated.

These requirements have been the motivation and driving factors behind this research work, the design of delay-efficient security protocols both with a cross-layer, across network approach

complemented by a layer-specific focus. For the first objective, we use game and mechanism design theoretical concepts to develop a cross-layer optimal consumer-centric and delay-efficient security protocol. For the latter goal, we take the most important and in use security protocols of two lower layers, the datalink and network layers, namely *IPsec* [RFC_4301], and *ATM Security Specification Version 1.1* [SEC_11] to propose extensions to their existing designs.

1.2 Statement of the Problem: Security and QoS Tradeoff

Today's more than ever critical network protection is offered by security services such as authentication, confidentiality, data integrity and access control. These services are implemented by selecting desired and appropriate security algorithms. These, however, introduce degradations to the requested level of Traffic Quality of Service in the form of additional delays and cell losses according to the selected mode of operation for a particular security mechanism and the chosen service during a connection, to name but a few. Confidentiality and data integrity security services in particular introduce significant degradations that are repeatedly implemented during the data transmission phase at a cell, datagram or packet level and are the two services used to illustrate the proposed solutions of this work.

Confidentiality security service is supported by encryption of the user data at the cell or packet level. Depending on the used security protocol and the specific layer, the payload or parts of the header are encrypted and routed with the header information toward the network. This mechanism is negotiated at the connection establishment phase and performed repeatedly in the data transfer phase of the connection for every cell of the data flow. At this time, the security algorithm and the mode of operation for this security association are determined for the life of the connection. Therefore, the cryptographic algorithm chosen to provide confidentiality and the mode the encryption device is operating in, are of significance to the influence on the requested Traffic QoS objectives.

The Data Integrity security service is supported by appending Message Authentication Code (MAC) or in some cases also a sequence number. This provides a means for detection of modification to data values or a sequence of data values. This mechanism is also performed repeatedly in the data transfer phase of the connection for every cell of the data flow according to the, at the connection

establishment phase, negotiated parameters. At this time, the cryptographic algorithm for this security association is determined for the life time of the connection. Therefore, what cryptographic algorithm is chosen to provide Data Integrity is also of significance to the influence on the requested Traffic QoS objectives.

On the other hand, the strength of a chosen security algorithm is correlated with this degradation and the caused delay. The strength is mainly evaluated by the security algorithm's invulnerability to various attacks of cryptanalysis. This, by itself, is influenced by many aspects like the design of the algorithm, the method of initial key setup, the number of key changes, the number of rounds in the design of the algorithm to produce final encryption, the used block-size, the key size used for the encryption and the size of data it is used for [NaJm_05][EAH_10][BiEl_99][SbWd_00]. The performance, in addition to the above, is affected by the combination of different aspects like the computing environment, what CPU speed, software or hardware implementation, software language choice, the optimization of code, mode of operation and the data transfer size and scenario.

In recent years this need to address both security and QoS have been paid attention through different more general approaches. The current literature and existing standardized security protocols at different layers do not yet take security mechanisms as an additional specific source for the QoS degradations into account. In addition none has paid a cross-disciplinary social-economic-technological approach to the design of optimal delay-efficient secruity protocol. The standardized protocols defining security operations and QoS signaling procedures today coexist rather separately without regards to the above problem.

It is the goal of this research work to bring the separately defined and standardized Traffic QoS signaling procedures and security protocols together and optimize the decision making and negotiation for this tradeoff with regards to social-economic-technological impacts. We approach this, first by taking a generalized cross-layer, across network approach and develop an optimal consumer-centric delay-efficient security protocol. Then, we focus on two existing standardized security protocols for two different layers, ATM Security Specifications Version 1.1 [SEC_11] and IPsec [RFC_4301] and their related specifications.

1.3 Brief Literature Review

The current security protocol literature does not take security mechanisms as an additional specific source for the QoS degradations into account. As a holistic approach, addressing social-economic-technological impact, there has been no attempt in use of mathematical modeling tools such as game and mechanism design theories to develop cross layer, across network optimal delay-efficient security protocols. As a layer specific approach, the standardized protocols defining security operations and QoS signaling procedures coexist rather separately without regards to above problem.

To address this holistic view, game theoretic approaches have been proposed for modeling and optimization in different areas of the field of communications networks for quite some time. Implementation of mechanism design theory in this field is also becoming increasingly popular.

These approaches in the areas of network security and its impact on performance and resource management focus on variety of different problems. Theodorakopoulos and Baras [TgBj_08] enforce upper bounds on damage from malicious users. Chen and Leneutre [ClLj_09] derive the expected behaviors of rational attackers, the minimum monitor resource requirement, and the optimal strategy of the defenders and then provide guidelines for IDS design and deployment. Otrok et al. [OMWDB_08] address the tradeoff between security and the resource consumption of IDSs for prolonging the lifetime of nodes and increasing their security in MANETs. Liu et al. [LZY_05] present a general incentive-based method to model and infer attacker intent, objectives and strategy. Bohacek et al. [BHLLO_07], introduce the Game-Theoretic Stochastic Routing (GTSR) framework, to make connection eavesdropping attacks maximally difficult and address the tradeoff among security, fault-tolerance, delay and throughput. Chen and Wu [CsWm_10] improve routing security risk and delivery ratio according to a tradeoff coefficient.

Non-holistic approaches – only at the technology level – of addressing the need of security and QoS have been getting increased attentions in the recent years as well. Specifically E-Health applications constitute one of the recent forerunners with applications such as Wireless Body Area Network (WBAN), Barua, et.al [BAX_11] propose packet scheduling offering a data integrity scheme. Mao et al. [MSL_10] propose a trust-based QoS management scheme to assure security.

Zhu and Zhang [ZhZh_08] integrate QoS measures into a proposed trust model. As part of this research work, we take reputation and trust as a driving force impacting the decisions of the agents. In fact, as interesting further research, we recommend the input of our solution into a reputation and trust-based model.

As pointed out above, we can see that the use of game theory and mechanism design theories is quite popular in addressing the security and overall performance of communications from diverse angles and with different focuses. To the best of our knowledge, there has been no mechanism design theoretic approach to design of optimal delay-efficient security protocols. The research work presented here introduces yet another promising perspective and implementation use of these theories.

At the Network Layer, in the IP paradigm, the *Integrated Services (IntServ)* Architecture [RFC_1633] is designed to support an end-to-end capability providing "*Guaranteed QoS*". The intolerant applications can take advantage of the *Guaranteed Service* model [RFC_2212]. On the security side, *IPsec* [RFC_4301] offers security services at the IP layer. Through IKEv2 protocol [RFC_5996], it provides the capability to select required security protocols, determine algorithms to use for the services and use cryptographic keys required to provide these services. The corresponding *Security Associations (SA)* or SA bundles are implemented accordingly to a particular packet and the security service(s) are provided. The IPsec [RFC_4301] and IKEv2 [RFC_5996] specifications do not provide a solution to the problem of causing additional degradations to the QoS of the network. As we can see, even at this layer the existing standards coexist separately. As part of this research work, we propose IPsec-O and IKEv2-O as a delay-efficient IPsec.

At the Datalink Layer, the *ATM Security Specification Version 1.1* [SEC_11] defines and standardizes functions and procedures to support security objectives in ATM networks. Security services are defined to support negotiation of multiple standardized and non–standardized security algorithms and modes of operation to allow interoperability in wide area networks. Two security protocols are defined for this purpose, The *Out-band Security Message Exchange (SME)* and the *In-band SME*. The *Out-band SME* supports security negotiation during the establishment phase of an

ATM connection using specific signaling Virtual Channels. *In-band SME*, however, communicates these objectives through the already established user channel after the connection establishment phase. This specification does not offer a solution to the problem of causing additional degradations to the QoS of the network.

The *ATM* Standards, *UNI Signaling Specification 3.1* [USIG_31] and *UNI Signaling Specification 4.1* [USIG_41], in addition to the *ITU–T Recommendation Q.2931* [ITUQ_2931] have defined signaling protocols for handling the QoS negotiations in ATM SVCs in user to network interfaces. These parameters are determined in the connection establishment phase and are to be met or exceeded by the intervening networks during the course of the ATM connection. After the agreement upon the QoS parameters the ATM connection is established and the ATM endpoint observes a consistent level of QoS for its connection. The end-to-end objective of QoS is then met.

UNI Signaling Specification 4.1 [USIG_41] defines the information elements and procedures required for specifying individual QoS parameter values for the ATM connections. This standard defines *Cell Transfer Delay (CTD)*, *Cell Delay Variation (CDV)* and *Cell Loss Ratio (CLR)* QoS parameters to be negotiable during the ATM connection establishment phase. The negotiation of these parameters is either done individually or on the basis of predefined QoS Classes.

As described above, there is a need of an holistic approach developing cross layer delay-efficient security protocols. It is also imperative to enhance the existing standardized layer-specific security protocols in terms of delay efficiency. The existing standardized security protocols to date do not take the degradations of requested and agreed upon Traffic Quality of Service caused by security implementations into account. The standardized security and QoS protocols coexist and research each area of interest rather separately without regard to above problem. It is the goal of this research work to bring the separately defined and standardized Traffic QoS signaling procedures and security protocols together and optimize the decision making and negotiation for this tradeoff in both a cross-layer and a layer-specific approach.

1.4 Main Contributions of this Research Work

The degradation of the QoS parameters induced by the transmission characteristics of the intervening networks and the processing procedures in the endpoints is a well-known fact. The various causes such as queuing behavior, buffer capacity and resource allocation procedures, to name but a few, have been paid a thorough attention.

Security operations are another important source of Traffic QoS degradation along secured networks. The different security services can be supported by a set of alternative security mechanisms. Each of which introduces a different value of degradation to various negotiable Traffic QoS parameters. This fact builds the core focus of this work. There is always a tradeoff between the strength of the security through the selection of different mechanisms and algorithms and the provided QoS. Without any attempt to consider these additional degradations, the already established negotiable Traffic QoS requested by the end user will decrease during the security operations or in worst case, would fail the connection.

On the other hand, the technology cannot be observed in isolation. Social economic aspects constitute an important part of this communications network. Human behavioral characteristics of selfishness and rationality affect decision making process in management and parameter choice at the technical level. By the same token, economics and market driven forces impact these implementations profoundly.

The main contribution of this research work is to address these issues as a whole and propose delay-efficient autonomous security protocols from two different perspectives. First, through a holistic cross layer and across network approach using mathematical modeling tools such as game and mechanism design theories to design and develop consumer-centric delay efficient security protocols. Second, a layer specific approach by proposing extensions to already existing standardized protocols at different layers, the datalink and network layers.

As the cross-layer and across network approach, we develop DSIC-S, a scenario-based model for three different levels of security. In this model, the security algorithms for each level are ranked based on the valuation of the level of security they offer and their occurring delay. We propose the

use of incentives to counteract the selfishness of the players to participate voluntarily in a manner requested by the consumer or social planner of the mechanism -individual rationality. We incorporate appropriate allocation rules and VCG payment scheme with Clarke's pivotal rule in the security protocol to implement the Social Choice Function truthfully in dominant strategy. To enforce the security and delay trade-off in the decision making of the nodes, we propose that the VCG payment calculation be a function of the valuation and rank of the announced security algorithm based on consumer's valuations and desired security or delay, depending on the governing security scenario. We further propose different weighted formulas according to the governing scenario for the calculation of these ranks.

We solve the important problem of misrepresentation of agents' private information in mechanism design and revelation theories for a delay-efficient security protocol through the proposed DSIC-S design. We incorporate a valuation system to integrate the caused delay at each node in selection of security algorithms without consumer's, the planner's, knowledge of the actual delays. The incentive model uniquely uses our proposed consumer's preference valuation system based on different security levels as an input for the VCG payment scheme with Clarke's pivotal rule and for the credit transfers.

DSIC-S achieves network-wide and individual Pareto optimality. This is an extremely desirable feature, which enforces natural adherence of the agents, people in the middle, to the mechanism rules. DSIC-S is consumer-centric. The consumer is the social planner of the mechanism to request the desired security and delay level for his data transmission. This is one of the main goals of this research work and a very sought-after proposition. This enforces a natural and automatic control of the behavior of the underlying participants. The consumer can rest assured that his services are provided as expected and promised to him.

DSIC-S is cheat-proof and strategy-proof. This is a strong property for real life applications, which enables the nodes, i.e. network providers, to naturally act responsibly and truthfully. Dominant strategy makes each node's best response independent of other nodes' choices of decisions and their belief functions.

DSIC-S is scenario based. It considers different levels of security and the resulting tradeoff manifestation. This focal design enforces Pareto optimality, individual rationality and allocative efficiency.

As a layer specific approach, at the network layer, we propose extensions to the current IPsec [RFC_4301] and IKEv2 [RFC_5996], by implementing the designed cross-layer DSIC-S protocol. IPsec-O and IKEv2-O inherit the strong properties of DSIC-S protocol.

At the datalink layer, we take ATM security protocols into consideration. We propose SME_Q as an extension to the current SME protocols [SEC_11] to achieve a desired end-to-end Traffic QoS and provide a consistent level of Traffic QoS for the user. In order to meet the user demands of Traffic QoS, the proposed SME_Q allows negotiation of different cryptographic algorithms and modes of operation according to the Security Agent Characteristics in regard to Traffic QoS (SAC_Q) at the connection establishment phase.

The significant contributions of this research work can be summarized as follows:

In general

- Manifestation of the security operation impact on network traffic QoS degradations in network security protocols.
- Innovative design of delay-efficient, autonomous, cheat-proof and optimal network security protocols integrating social behavioral and economics characteristics.
- Application of mathematical modeling theories integrating implications of social behavioral characteristics to develop a holistic framework able to better address critical real-life problems.

Cross Layer Approach

Dominant Strategy Incentive Compatible Security Protocol - DSIC-S

- Implementation of game and mechanism design theories for the development of a

new consumer-centric network security protocol providing network-wide security-delay tradeoff, DSIC-S.

- Solving the important problem of misrepresentation of agents' private information in mechanism design and revelation theories for a delay-efficient security protocol through the proposed infrastructure and design.
- Development of a scenario-based socially desirable, Pareto optimal protocol, which is cheat and strategy-proof implying dominant strategy.
- Design of a valuation system based on security strength and delay of algorithms to solve the revelation theorem's problem of private information misrepresentation in mechanism design theory.
- Design of an incentive compatible protocol, to counteract the selfishness of the players. The valuation system is used along with VCG and Clarke pivotal rule for the credit transfer calculations.
- Design of a consumer-centric security protocol to enforce a natural equilibrium and automatic control of the behavior of participants. The consumer can rest assured that his services are provided as expected and promised to him.

Layer Specific Approach

Network Layer: IP Delay Efficient Security Protocol, IPsec-O and IKEv2-O

- Research and analysis of the existing standardized security and QoS signaling protocols for the determination of the existence of a solution to the above-described problem.
- Implementation of DSIC-S protocol and proposal of extensions to the existing IPsec [RFC_4301] and IKEv2 [RFC_5996] protocols.

Datalink Layer: ATM Delay Efficient Security Protocols, SME_Q

- Research and analysis of the existing standardized security and QoS signaling protocols for the determination of the existence of a solution to the above-described problem.

- Development of new security protocols, SME_Q, based on and as extensions to the existing standardized security and QoS signaling protocols. These new protocols allow the implementation of the security services and mechanisms throughout a connection without exceeding the established Traffic QoS during the life of that connection. The user should not then experience any further performance decrease (loss of Traffic QoS) due to the governing security protocols and policy.
- Development of proprietary simulation software, which simulates the proposed new protocols as prototypes.

1.4.1 Publications of the Proposed Protocols

All of the proposed protocols in this research work have been documented in technical papers for peer-reviewed journal and conference publications. The following is the list of these technical papers:

SfMi_111 F. Schlake, L. Mili, "Efficient Network Security as a Strategic Game", peer-reviewed and accepted,to be published in the International Journal of Critical Infrastructures, 2012.

SfMi_112 F. Schlake, L. Mili, "IPsec-O, Optimal, Delay-Efficient and Cheat-Proof; A Mechanism Design Theoretic Approach", to be submitted for publication, 2012.

SfRc_02 F. Schlake, C. Ruland, "A Security Protocol Providing QoS in ATM Networks", The 8th International Conference on Communication Systems, ICCS2002, IEEE conference, IEEE Catalog: 02EX585, Vol. 2, p. 933-937, November 2002.

1.5 Organization of the Dissertation

Chapter 1 provides an overview and the motivation behind this research work. It discusses the problem statement and gives a brief overview of literature. It then describes the main

contributions of this work. At the end the organizational approach to the objectives is mapped.

Chapter 2 gives an overview of the main concepts of game and mechanism design theories, which are important to this work.

Chapter 3 gives an overview of network security. It briefly describes the components of a secure connection. It introduces some security algorithms and modes of operations, which are used in the context of this research work.

Chapter 4 portraits the literature review and state of the art in the field of security and QoS Tradeoff.

Chapter 5 presents the DSIC-S protocol. It describes the design assumptions and constraints. It describes the design of the scenario and valuation models. After explaining the security association model, the details of DSIC-S procedures are illustrated.

Chapter 6 Models DSIC-S strategic game in a game and mechanism design theoretic approach. It details the process of the agents' strategic decisions and reviews the design of the proposed incentive model.

Chapter 7 lists the theoretical results of the DSIC-S design in numerous Theorems alongside their proofs.

Chapter 8 reviews the simulation results. It confirms the validation of the theoretical results through several simulation cases.

Chapter 9 brings the attention to the network layer. It gives a review of the existing IPsec and IKEv2 protocols. It then illustrates the implementation of the DSIC-S protocol and the proposed IPsec-O and IKEv2-O extensions to the current protocols.

Chapter 10 brings the attention to the datalink layer and gives a brief summary of the ATM security area. It also describes the design assumptions and constraints for the development of SME_Q.

Chapter 11 presents the SME_Q protocol. It describes the proposed extensions and introduced parameters and procedures in detail.

Chapter 12 introduces the developed simulation software SMEQSIM. It illustrates the screens throughout the simulation process.

Chapter 13 Through a comprehensive quantitative simulation, this chapter illustrates the effect of the security operation influences on the traffic QoS based on a real world example of an ATM network.

Chapter 14 concludes the work with a summary of contributions and results. It also discusses some directions for future research.

Chapter

2

Game & Mechanism Design Theoretic Concepts

2.1 Strategic Form Game

Game theory can be described as mathematical modelling of interactions between rational and intelligent decision makers [MyRb_97]. It analyzes the decisions made by participants, the so called agents or players, in a game setting. There are numerous form games, in this research work, we, however, focus on the *strategic form games* or *normal form games*, the most suitable for network economics[NGNP_09]. Webb [WeJn_07] defines a strategic game as a game, which uses pure strategies. In the same manner, Narahari et al. [NGNP_09] define it as

> ***Definition 2.1:*** *" A strategic form game Γ is defined as a tuple $\langle N, (S_i)_{i\in N}, (u_i)_{i\in N}\rangle$, where $N = \{1,2, \ldots,n\}$ is a finite set of players; $S_1, S_2, \ldots, S_n$ are the strategy sets of the players $1, \ldots, n$, respectively; and $u_i : S_1 \times S_2 \times \cdots \times S_n \rightarrow R$ for $i = 1,2, \ldots,n$ are mappings called the utility functions or payoff functions".*

In this chapter and throughout this work, we use the notations and definition style used in [NGNP_09].

The pure strategies are also called the actions of the agents. Each rational player is in pursuit of maximizing its expected utility u_i or payoff.

In strategic games with *complete information,* the set of players, the set of strategies and the set of utilities are common knowledge. Each player knows them and every player knows that every player knows them.

This research work's environment requires the use of another set of strategic games, namely the ones with *incomplete information*, which build an important focus area of mechanism design theory. In these games, the agents have knowledge of some private information, which is beneficial to them in choosing their strategies, but is unknown to other players in the game. This builds one of the major problems in the mechanism design theory, which has given birth to key areas of *revelation theory* and *incentive compatibility* in order to solve it. The idea is to somehow get access to this private information truthfully before the game starts, so that the social planner can design a socially desirable mechanism for achieving desired outcome. Bayesian and Selten games deal with this set of strategic form games, namely games with incomplete information [NGNP_09].

2.1.1 Bayesian and Selten Games

In 1968, *John Charles Harsanyi*, who was one of the three joint Nobel Prize winners in Economic Sciences in 1994 along with *John Nash* and *Reinhard Selten*, proposed the *Bayesian form games* to address games with incomplete information. We can define a Bayesian game [NGNP_09] as

> ***Definition 2.2:*** *A Bayesian game Γ is defined as a tuple $\Gamma = \langle N, (\Theta_i), (S_i), (p_i), (u_i) \rangle$ where $N = \{1, 2, \ldots, n\}$ is a set of players. Θ_i is the set of types of player i where $i = 1, 2, \ldots, n$. S_i is the set of actions or pure strategies of player i where $i = 1, 2, \ldots, n$. The probability function p_i is a function from Θ_i into the set of probability distributions over Θ_{-i}. That is, for any possible type $\theta_i \in \Theta_i$, p_i specifies a probability distribution $p_i(.|\theta_i)$ over the set Θ_{-i} representing what player i would believe about the types of the other players if his own type were θ_i; the payoff function $u_i : \Theta \times S \rightarrow R$ is such that, for any profile of actions and any profile of types $(\theta , s) \in \Theta \times S$, $u_i(\theta , s)$ specifies the payoff that player i would get, if the players' actual types were all θ and the players all chose their actions according to s.*

In the Bayesian games, the players know the structure of the game and their own private types θ_i. All other agents have a probabilistic idea of what this private type of other players would be through their belief function p_i, which are common knowledge among all players. This implies that the deterministic value of θ_i is only known to agent i and is considered his own private information in these games.

Reinhard Selten, one of the joint Nobel Prize winners in Economic Sciences in 1994 along with *John Nash* and *John Charles Harsanyi*, introduces the notion of *type agents* to Harsanyi's Bayesian form games [HaJC_67]. His idea is to transform any game into a strategic form game. Each player in the Bayesian game is replaced with a number of type agents, the number of types θ_i of that player in former game. The utilities and strategies in Selten games are all defined based on θ_i. For the rest of this research work we use this representation.

2.1.2 Dominant Strategy Equilibria of Bayesian Games

Using the Selten game representation two different dominant strategy equilibria for Bayesian games can now be defined [NGNP_09]:

Strongly Dominant Strategy Equilibrium

Definition 2.3: *Given a Bayesian game* $\Gamma = \langle N, (\Theta_i), (S_i), (p_i), (u_i) \rangle$, *a profile* $s^*(.) = (s^*_1(.), s^*_2(.), \ldots, s^*_n(.))$ *is said to be a strongly dominant strategy equilibrium if*

$$u_{\theta i}(s^*_i(\theta_i), s_{-i}(\theta_{-i})) > u_{\theta i}(s_i, s_{-i}(\theta_{-i})) \quad (2.1)$$
$$\forall s_i \in S_i \setminus \{s^*_{-i}(\theta_i)\}, \forall s_{-i}(\theta_{-i}) \in S_{-i}, \forall i \in N, \forall \theta_i \in \Theta_i, \forall \theta_{-i} \in \Theta_{-i}$$

Weakly Dominant Strategy Equilibrium

Definition 2.4: *Given a Bayesian game* $\Gamma = \langle N, (\Theta_i), (S_i), (p_i), (u_i) \rangle$, *a profile* $s^*(.) = (s^*_1(.), s^*_2(.), \ldots, s^*_n(.))$ *is said to be a weakly dominant strategy equilibrium if*

$$u_{\theta i}(s^*_i(\theta_i), s_{-i}(\theta_{-i})) \geq u_{\theta i}(s_i, s_{-i}(\theta_{-i})) \quad (2.2)$$

$$\forall s_i \in S_i, \forall s_{-i}(\theta_{-i}) \in S_{-i}, \forall i \in N, \forall \theta_i \in \Theta_i, \forall \theta_{-i} \in \Theta_{-i}$$

and strict inequality is satisfied for at least one $s_i \in S_i$.

It is to be noted that the dominant strategy equilibrium is independent of the belief function p_i. The weakly dominant strategy equilibrium is used in mechanism design theory to define dominant strategy implementation of mechanism. It is also noteworthy that, as we can see above, a strongly dominant strategy equilibrium is automatically a weakly dominant strategy equilibrium. Also there could be several weakly dominant strategy equilibria but only one strongly dominant strategy equilibrium, if it exists at all.

2.2 Mechanism Design Theory Environment

Mechanism design theory prescribes system-wide solutions in form of mechanisms or protocols to achieve a desired outcome among self-interested agents withholding private information. Without it, the agents would follow their own payoff and profit interests in the way of a socially desirable solution. Since the agents hold private information not known to all, mechanism design theory is said to be developing the solution to an incompletely specified optimization problem.

In a mechanism design setting the are n rational and intelligent agents with $n \in N = \{1, 2, \ldots, n\}$. X is the set of outcomes from which each agent chooses one. Each agent has some private information only known to him and is called the agent's type θ_i. The set of all different types of an agent is called Θ_i. Then, the set of all types of all players is defined as $\Theta = \prod_{i \in N} \Theta_i$. A type profile $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ includes a collection of a particular selection of all agent's types. As described above, the belief function p_i, the belief of an agent about the type profiles of all other agents, should be able to be derived from a common prior distribution over Θ. Utility function $u_i : X \times \Theta_i \rightarrow R$ is the motivating factor behind each agent's action and selection of its strategy. In order to formulate a socially desirable mapping to the desired outcome a Social Choice Function (SCF) is defined as [NGNP_09],

Definition 2.5: *A Social Choice Function is a mapping* $f : \prod_{i \in N} \Theta_i \rightarrow X$ *, that assigns a collective choice* $x \in X$ *to each profile* $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ *from the set of outcomes X.*

This is a general definition for the SCF. As we will see later in the quasi-linear setting, it takes more detailed definition according to the mechanism design environment it is used at.

2.3 Direct Mechanisms and their Properties

As mentioned above, the missing piece in the puzzle is the private information of each agent. This private information needs to be elicited in truthful manner. One way of doing that, is to simply ask all agents directly before the game to announce their true types. These mechanisms are called *direct mechanisms* or *direct revelation mechanism*. Adhering to the Bayesian games definition, we can define [NGNP_09] a direct mechanism as

Definition 2.6: *Given a SCF* $f : \prod_{i \in N} \Theta_i \rightarrow X$*, a direct revelation mechanism is defined as* $D = ((\Theta_i)_{i \in N}, f(.))$.

The truthful elicitation is one of the main problems in the mechanism design theory that has gotten much attention in different proposed solutions. Revelation Theorem and incentive compatibility are among solutions offered by scientists and will be discussed in the next sections.

2.3.1 Revelation Theorem

In an environment where the agents are rational and intelligent, they always aim at maximizing their payoff. If revealing their true types counteracts this maximization of profit, they do any attempt to misrepresent this private knowledge. Revelation Theorem challenges to solve this important problem in mechanism design theory. Preference revelation is the art of eliciting truthful information that is held by each agent unknown to others. This can be done in several ways. It may be achieved through novel mechanism design, that enforces a natural environment that the agents find it to their best interest to reveal their true types. It can also be achieved by offering incentives to the agents to be truthful. Incentive Compatibility deals with this category and is described in the next section.

2.3.2 Incentive Compatibility

Major area of mechanism design theory deals with designing appropriate incentives for the agents to enforce the truthful revelation of their private information. The misrepresentation of their true types is specifically of importance and harder to solve for the real-world scenarios. Design of appropriate and high enough incentives makes a SCF *Incentive Compatible*. We can define IC [NGNP_09] as

> ***Definition 2.7:*** *A SCF is said to be incentive compatible or truthfully implementable if the Bayesian game induced by the direct mechanism* $D = ((\Theta_i)_{i\in N}, f(.))$ *has a pure strategy equilibrium* $s^*(.) = (s^*_1(.), s^*_2(.), \ldots, s^*_n(.))$, *where* $s^*_i(\theta_i) = \theta_i$, *for* $\forall\ \theta_i \in \Theta_i$ *and* $\forall i \in N$.

The SCF can be truthfully implementable or incentive compatible in different equilibria. Dominant Strategy is of interest for this research work and is defined next.

2.3.3 Dominant Strategy Incentive Compatible

This property implies that truth revelation by each agent builds a dominant strategy equilibrium of the game induced by $\mathcal{D}$ and is defined [NGNP_09] as

> ***Definition 2.8:*** *A Social Choice Function* $f : \prod_{i\in N} \Theta_i \rightarrow X$ *is said to be Dominant Strategy Incentive Compatible, DSIC, or truthfully implementable in dominant strategy, if the direct mechanism* $D = ((\Theta_i)_{i\in N}, f(.))$ *has a weakly dominant strategy equilibrium* $s^*(.) = (s^*_1(.), s^*_2(.), \ldots, s^*_n(.))$, *where* $s^*_i(\theta_i) = \theta_i$, *for* $\forall\ \theta_i \in \Theta_i$ *and* $\forall i \in N$.

This property is also called cheat- or strategy-proof. We shall recall that strongly dominant strategy equilibrium is automatically a weakly dominant strategy equilibrium.

2.3.4 Individual Rationality

Individual rationality is referred to as voluntary participation property [NGNP_09]. According to mechanism design theory concepts an agent will voluntarily participate in the game if she would not be worse off after playing. This implies that the agent gains a positive payoff by participating

in the game. There are three stages that an agent can decide not to participate, depending on the time they get to know their types and all have announced their information and the outcome has been chosen. In the case of interim individual rationality the agent i is allowed to withdraw from the mechanism only at an interim stage after knowing her type and before announcing it. This is the case for our security strategic game. In order for the agent not to withdraw from the game, her interim utility u_i should be greater than or at least equal to the utility $\bar{u}_i(\theta_i)$ she gets by withdrawing from the mechanism when her type is θ_i.

2.4 Quasi-Linear Environment Transfers

This environment is one of the most preferred environments to be assumed in mechanism design theory. In quasi-linear settings, money/credit/incentive transfer takes place to or from the agents, which makes this environment specially attractive for incentive design.

Given the type profile $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ and $k(\theta)$, the allocation function or project choice, the social choice function here takes the form

$$f(\theta) = (k(\theta), t_1(\theta), ..., t_n(\theta)) \qquad \forall\ \theta \in \Theta \qquad (2.3)$$

where the vector

$$t(\theta) = (t_1(\theta), t_2(\theta), ..., t_n(\theta)) \qquad \forall\ \theta \in \Theta \qquad (2.4)$$

constitutes the incentive vector transferred to or received from the agents.

For a direct revelation mechanism $\mathcal{D} = ((\Theta_i)_{i\in N}, f(.))$, the utility function takes the quasi-linear form of

$$u_i(\theta) = v_i(k(\theta), \theta_i) + t_i(\theta) \qquad \forall\ \theta \in \Theta \qquad (2.5)$$

where $v_i(k(\theta), \theta_i)$ is the valuation of agent i of the choice, given his type θ_i and allocation function $k(\theta)$. In many situations and mechanisms, as it is the case in this research work, this valuation function represents the type of the agent. In these cases, announcing a type is like reporting the agent's valuation function.

2.4.1 Allocative Efficiency

For the design of optimal mechanisms, it is an utmost desired property that the chosen allocation function maximizes the valuation of the whole system. This means, if K is the set of all possible allocation functions, then the chosen allocation function should be $k(\theta)_{max} \in K$. This can be formulated as

$$\max \sum_{i \in N} v_i (k(\theta), \theta_i) := \sum_{i \in N} v_i (k(\theta)_{max}, \theta_i) \qquad \forall \theta \in \Theta \qquad (2.6)$$

This means that the chosen allocation function will maximize the sum of the values of the players for every $\theta \in \Theta$ [NGNP_09]. We define the term *Allocative Efficiency* as

Definition 2.9: *A Social Choice Function $f(\theta) = (k(\theta), t_1(\theta), ..., t_n(\theta))$ is said to be allocatively efficient if for each $\theta \in \Theta$, $k(\theta)$ maximizes the value of all agents and satisfy (2.6).*

Thus every selected allocation in the mechanism will be a value maximizing allocation.

2.4.2 Vickrey-Clarke-Groves Mechanism

The VCG mechanism is named after *William Vickrey*, *Edward Clarke*, and *Theodore Groves* for their invaluable contribution to the field of mechanism design theory. William Vickrey, famous for his *Vickrey Auction* [ViWi_61], the second price sealed bid auction, won the Nobel Prize in economic sciences jointly with James A. Mirrlees in 1966 for the fundamental contributions to the economic theory of incentives under asymmetric information. Clarke [ClEd_71] and Groves [GrTh_73] have contributed to the generalization of the Vickrey mechanisms by defining a set of dominant strategy incentive compatible mechanisms in the quasi-linear environments [NGNP_09].

Groves Theorem provides a sufficient condition for an allocatively efficient social function in quasi-linear environment to be dominant strategy incentive compatible [NGNP_09].

Theorem 2.1: *Let the SCF $f(\theta) = (k(\theta), t_1(\theta), ..., t_n(\theta))$ be allocatively efficient. Then $f(\theta)$ is dominant strategy incentive compatible if it satisfies the following payment structure, the Groves payment rule:*

$$t_i(\theta) = \left[\sum_{j \neq i} v_j (k(\theta), \theta_j) \right] + h_i (\theta_{-i}) \qquad \forall i, j \in N \qquad (2.7)$$

where $h_i: \Theta_{-i} \rightarrow \mathbb{R}$ is any arbitrary function that satisfies the condition of weak budget balance, namely $\sum_{i \in N} t_i(\theta) \leq 0$ for every $\theta \in \Theta$. Any mechanism satisfying the above Groves Theorem is called *Groves Mechanism*.

Definition 2.10: *A direct mechanism $D = ((\Theta_i)_{i \in N}, f(\theta))$ in which $f(\theta) = (k(\theta), t_1(\theta), ..., t_n(\theta))$ is allocatively efficient and satisfies the Groves payment rule is called Groves Mechanism.*

In mechanism design realm, the Groves Mechanisms are referred to as Vickrey-Clarke-Groves (VCG) mechanisms. This is because the Clarke mechanism is a special case of the Groves mechanism and Vickrey mechanism is in turn a special case of the Clarke mechanism, as described in the next section. It is noteworthy that the Groves theorem provides a sufficient condition for an allocatively efficient SCF to be Dominant Strategy Incentive Compatible – DSIC. This is one of main reasons of popularity of use of the VCG rule for design of mechanisms.

2.4.3 Clarke Pivotal Mechanism

Clarke Pivotal Mechanism [ClEd_71] is a special case of the Groves mechanism. In 1971, Clarke independently developed a natural special form for the function $h_i(.)$ of Groves mechanism. He defined it as follows:

$$h_i(\theta_{-i}) = -\sum_{j \neq i} v_j(k_{-i}(\theta_{-i}), \theta_j) \qquad \forall \theta_{-i} \in \Theta_{-i}, \forall i \in N \qquad (2.8)$$

where $k_{-i}(\theta_{-i}) \in K_{-i}$ is the choice of a project that is allocatively efficient in the absence of agent i, namely when only n–1 agents were in the system. Also, the project choice $k_{-i}(\theta_{-i})$ must satisfy the following:

$$\sum_{j \neq i} v_j(k_{-i}(\theta_{-i}), \theta_j) \geq \sum_{j \neq i} v_j(k, \theta_j) \qquad \forall k \in K_{-i} \qquad (2.9)$$

This implies that the transfer in the Clarke mechanism is as follows:

$$t_i(\theta) = \left[\sum_{j \neq i} v_j(k(\theta), \theta_j)\right] - \left[\sum_{j \neq i} v_j(k_{-i}(\theta_{-i}), \theta_j)\right] \qquad \forall i,j \in N \quad (2.10)$$

It is noteworthy that Clarke's mechanism achieves weak budget balance in normal conditions.

Chapter

3

Network Security

3.1 Security Infrastructure

In this chapter, general component and mechanisms of network security are briefly described that are used in the context of this research work. At this point, although there might be some overlaps, layer specific terminologies are not addressed.

3.1.1 Security Agent

A Security Agent (SA) negotiates, determines and applies security services, algorithms and modes of operation to secure a connection. This communication always takes place between two SAs, the initiating SA and the responding SA. The SAs can reside at any node along the connection path or at end devices. Figure 3.1 illustrates the simplest case of a secure connection containing SAs.

3.1.2 Security Associations

For each security service to be provided for a connection, one security association is established with another SA. For example, if both confidentiality and data integrity are to be applied to a

connection, two security associations need to be established between the initiating SA and the responding SA. The security association contains the information (cryptographic algorithm, modes of operation, etc.) required to provide desired security on a given connection. Figure 3.1 shows a connection with a security association.

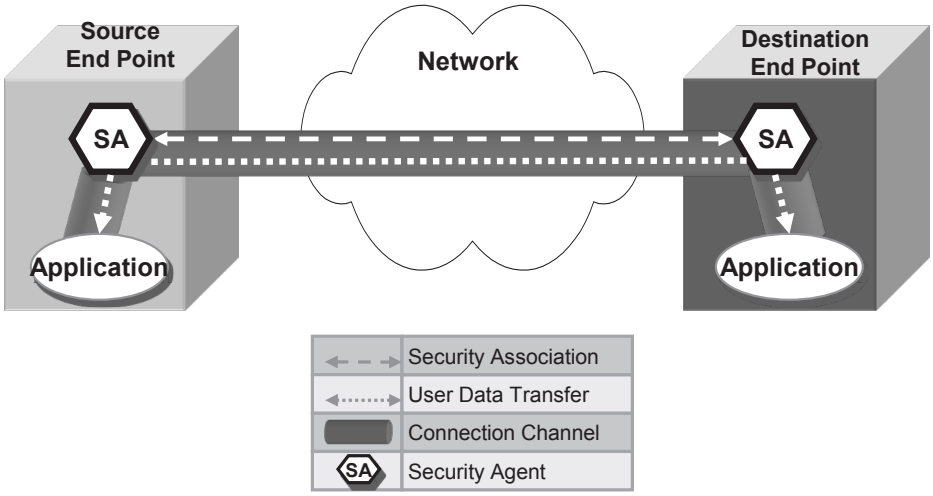


Figure 3.1 An Example of a Simple Secure Connection

3.1.3 Nesting and Multiple Security Associations

In a network, more than one pair of Security Agents and more than one security association can be involved providing security services. Multiple security associations provide security services specially for the cases that no prior knowledge of the structure of the network and existence of other SAs exists. They also allow usage of different security policies for different parts of the network. Figure 3.2 illustrates an example of multiple security associations with two SA pairs.

Each layer specific security protocol may define its own terminology to address this complexity. Terms such as *nesting*, *tunneling*, *bundling* and *sequential* with or without *overlap* are some examples. Each protocol may also set different restrictions and constraints in terms of level of nesting or tunneling.

In the context of DSIC-S protocol, the term *nesting* is used to address the existence of several SAs and security associations along the path. IP uses the terms of *SA bundles* and *iterated tunneling* for

multiple security associations along the path. ATM paradigm defines *nesting*, *sequential* with and without *overlap* as terms for this.

In general, in the context of this work, *nesting* is used for multiple security associations, where multiple SAs are involved, unless we are addressing layer specific protocols with their own proprietary terminologies and definitions. Nesting happens when two SAs along the path of another two also establish security associations. Figure 3.2 illustrates an example of nesting security associations. In this example, it would happen if SA_2 and SA_3 established an association in addition to an association between $SA_1$and SA_4. In this case, the segment $[SA_2, SA_3]$ is contained in the segment $[SA_1, SA_4]$, this means $[SA_2, SA_3] \subset [SA_1, SA_4]$. Other security associations would happen if the association existed between $[SA_1, SA_2]$ and $[SA_3, SA_4]$. In the context of this work, we

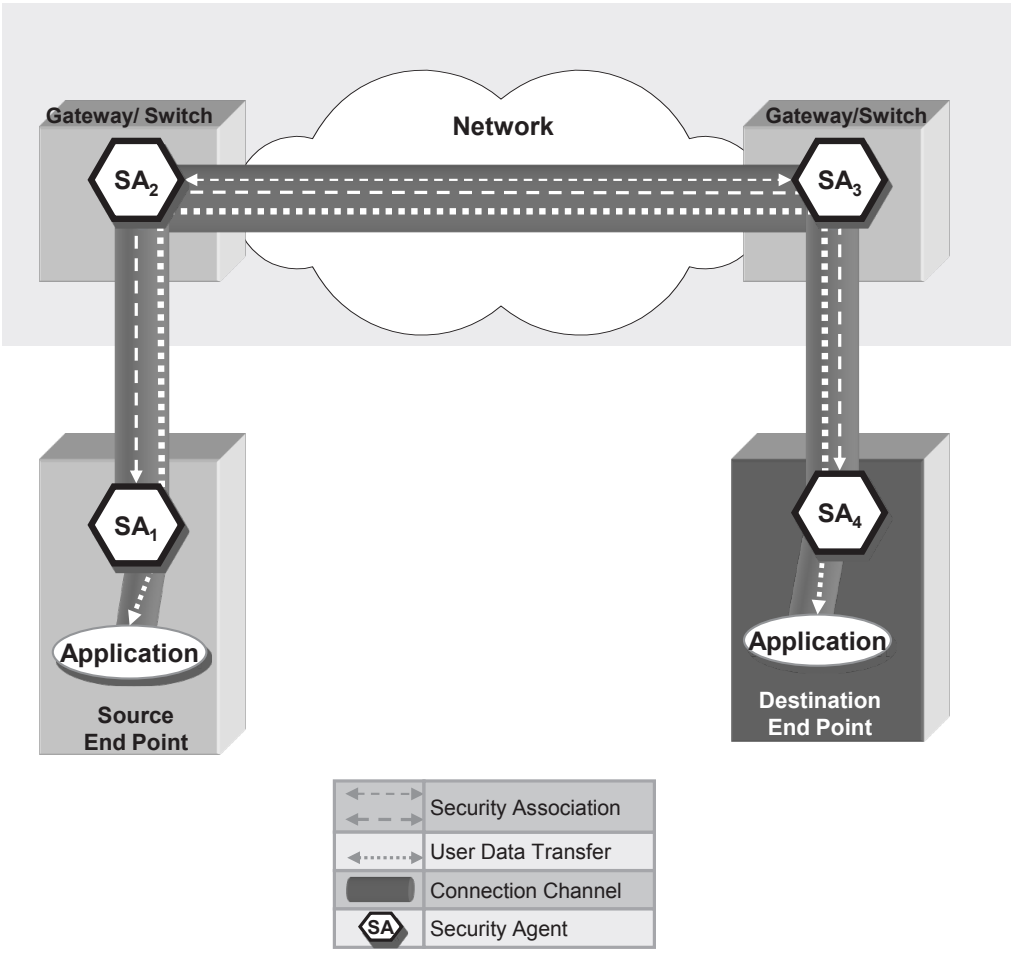


Figure 3.2 Example of Nesting Security Associations

do not assume cross associations for example between SA_1, SA_3 and SA_2, SA_4. Another case may legitimately be presented when one SA is in the path of two others for example if the association existed between [SA_1, SA_3] with SA_2 in the middle. In this case an association between SA_2, SA_4 would not be allowed.

3.2 Security Services

Security services are defined to categorize specific tasks and procedures for security implementations using appropriate security algorithms and modes of operation. Authentication, confidentiality, data integrity and access control are also referred to as security services. The confidentiality and data integrity services are elaborated further below, because of their on-line repetition during the data transmission phase and their significance for this research work in terms of accumulative delay they impose to the network.

3.2.1 Confidentiality Service

Confidentiality is provided by encryption. It is used to protect the original data to be available to unlawful or malicious access or use. Depending on the layer specific security protocol, either the payload of each packet or cell is only encrypted, allowing the routing and switching to be done easily using the header in original form or it is done to part of the header as well. In Section 3.3 some security algorithms used for encryption are briefly exploited.

3.2.2 Data Integrity Service

Data Integrity service provides protection against data modifications. It performs by building and adding Message Authentication Codes to the data units. There are two types of data integrity services provided: One with replay and reordering protection and the other without this protection. Data integrity service, which provides reordering and replay, first appends a sequence number at the end of each data unit, then calculates the MAC for the data unit including the sequence number. This way the repeated data units can be detected and discarded. In the case the reordering option

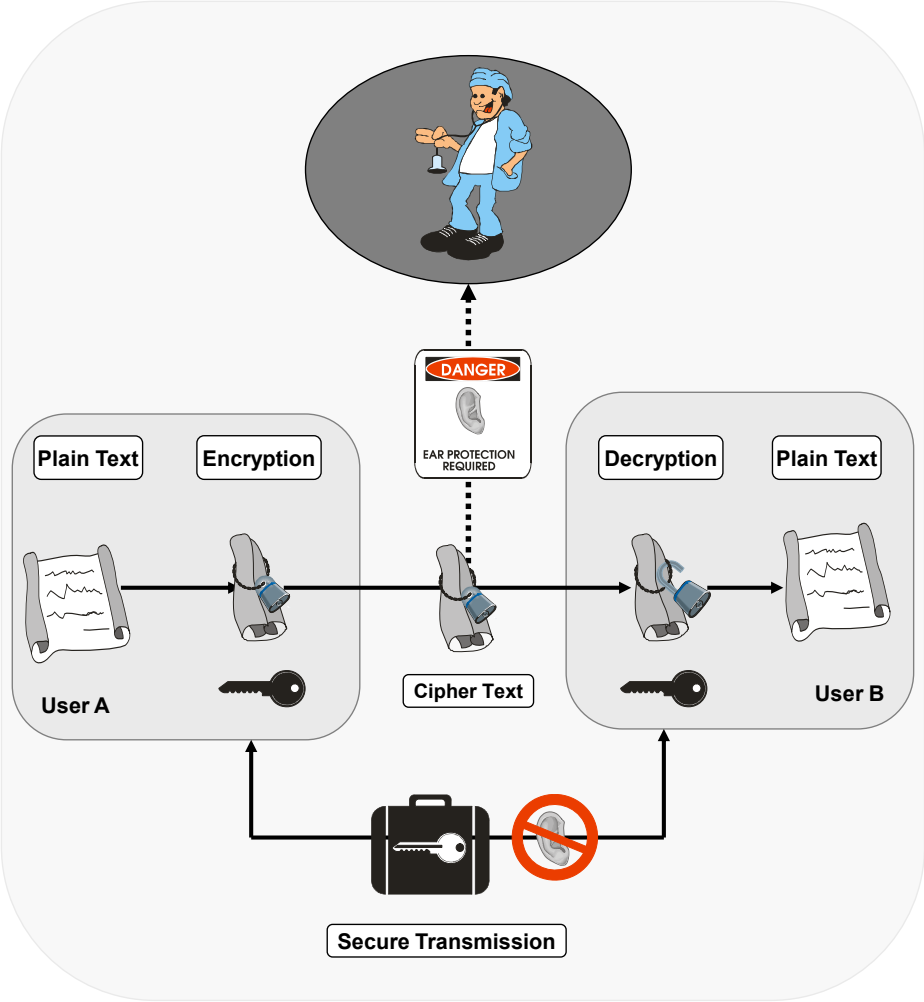


Figure 3.3 Confidentiality with Symmetric Algorithms

is not chosen, MAC is only calculated for the data unit itself and no sequence number is added. In Section 3.3 some security algorithms used for data integrity service are briefly described.

3.3 Security Algorithms

In this section, some most common security algorithms are briefly introduced, which are used and referred to in this research work.

AES- Rijndael-256, 192 and 128

Rijndael is a block cipher, developed by two Belgian cryptologists, Vincent Rijmen and Joan Daemen, has been chosen as the AES Advanced Encryption Standard algorithm and adopted by the

U.S. government [FIPS_197]. It is used with three different key sizes. AES is a substitution-linear transformation network with 10, 12 or 14 rounds, depending on the key size. Rijndael has no known security attacks [NBDFR_00].

Blowfish

Blowfish was designed in 1993 by Bruce Schneier as a fast, free alternative to existing encryption algorithms [Sb_11]. Blowfish is a symmetric encryption algorithm has a 64-bit block size and a variable key length - from 32 bits to 448 bits. It is a 16-round Feistel cipher and uses large key-dependent S-boxes.

DES

Data Encryption Standard (DES) was publicly introduced in the year 1974 [FIPS_463]. In 1977, it was adopted in the United States as a federal standard. It is a block algorithm based on the Feistel cipher in various rounds. It implies substitution and transposition for both encryption and decryption. DES takes 64 bits as input and produces cipher text of 64 bits. The standard DES algorithm uses 16 rounds and a 56–bit master key. It generates a different 48–bit key for each round. For many years, DES-enciphered data were safe because few organizations possessed the computing power to crack it. “In July 1998, the supercomputer DES Cracker designed by Electronic Frontier Foundation (EFF) was able to crack RSA’s DES Challenge II-2 in 56 hours. The same computer, assisted by 100,000 distributed.net PCs on the Internet, was able to crack DES Challenge III in only 22 hours” [RSA_11]. AES has replaced DES as US federal standard.

Twofish

Twofish is a block cipher by Counterpane Labs, published in 1998[Sb_11]. Twofish has a 128-bit block size, a key size ranging from 128 to 256 bits, and is optimized for 32-bit CPUs [Sb_11]. It accepts keys of any length up to 256 bits. Twofish has key dependent S-boxes like Blowfish. The Twofish algorithm can be operated by any key size up to 256 bits. Twofish is originally defined for the AES key sizes. It designed to be padded with zeroes to reach the next AES defined key size

[NBDFR_00].

RC5 and RC6

RC6 is a symmetric key block cipher derived from RC5. It was designed by Ron Rivest et al. [RSA_11]. The security of both algorithms relies on variable rotations as the principal source of non-linearity; there are no S-boxes. The variable rotation operation in RC6, unlike RC5, is regulated by a quadratic function of the data. The key schedules of RC5 and RC6 are identical [NBDFR_00].

Triple DES

This variation of DES works with a double DES key of 128 bit, where 112 bits are relevant for the operation to make the standard DES algorithm stronger [FIPS_463]. It performs three DES operations, two DES encryptions and one DES decryption.

SHA-1

The Secure Hash Algorithm–1 [FIPS_1801] was introduced from NIST in 1995 as a revision to SHA [FIPS_1801]. It takes a message of the length smaller than 2^{64} bits as input and computes a 160–bit message digest. Like MD5, it is used in the cases where large data should be first securely compressed before a signature transaction. It is based on the fact that it should be computationally infeasible to find two massages with the same message digest or to find any message for a known finger print. SHA–1 is slightly slower than MD5, thus more secure against brute–force collision and inversion attacks.

SHA-2: SHA-256 and SHA-512

The SHA-2 family of hash functions consists of SHA-224, SHA-256, SHA-384 and SHA-512) and is used by Federal agencies for all applications using secure hash algorithms [FIPS_1803]. SHA-256 uses six logical functions, where each function operates on 32-bit words. SHA-512 functions, however, operate on 64-bit words.

RIPEMD-160

RIPEMD-160 is a 160-bit cryptographic hash function, designed by Hans Dobbertin [DBP_96]. It was designed to be a secure replacement for MD-4 and MD-5. RIPEMD-160 is tuned to work with 32-bit processors.

MD-5

Message Digest 5 was introduced by R. Rivest in 1992 in a RFC of IETF [RFC_1321]. It takes any arbitrary long data and compresses it to 128–bit message digest or finger print. It was built for 32–bit machines. It is used in the cases, where large data should be first securely compressed before a signature transaction. It is based on the fact that it should be computationally infeasible to find two massages with the same message digest or to find any message for a known finger print. There have been attacks on MD-5, which makes the use of it less secure if collision-resistant has function is required.

3.4 Modes of Operation

Data can be encrypted using predefined mechanisms standardized in [ISO_10116]. The following four modes of operation are defined in this specification: ECB (Electronic Code Book), CBC (Cipher Block Chaining), CFB (Cipher FeedBack), and OFB (Output FeedBack). In addition, Counter mode and statistical self-synchronization modes are in extensive use today.These modes of operation are described in detail below.

3.4.1 Electronic Code Book (ECB) Mode

ECB, defined in [ISO_10116], processes each block of data independently according to the selected cryptographic algorithm. Therefore the encryption and decryption of data can be performed independently. Reordering of cipher text will result in a corresponding reordering of the plain text. The same block will result to the same cipher text when the same key is used. Use of ECB is therefore not suitable in cases where the same block repeatedly appears. Only n-bit block sizes can be processed by ECB. Therefore it might be a need for other lengths to be padded. Figure

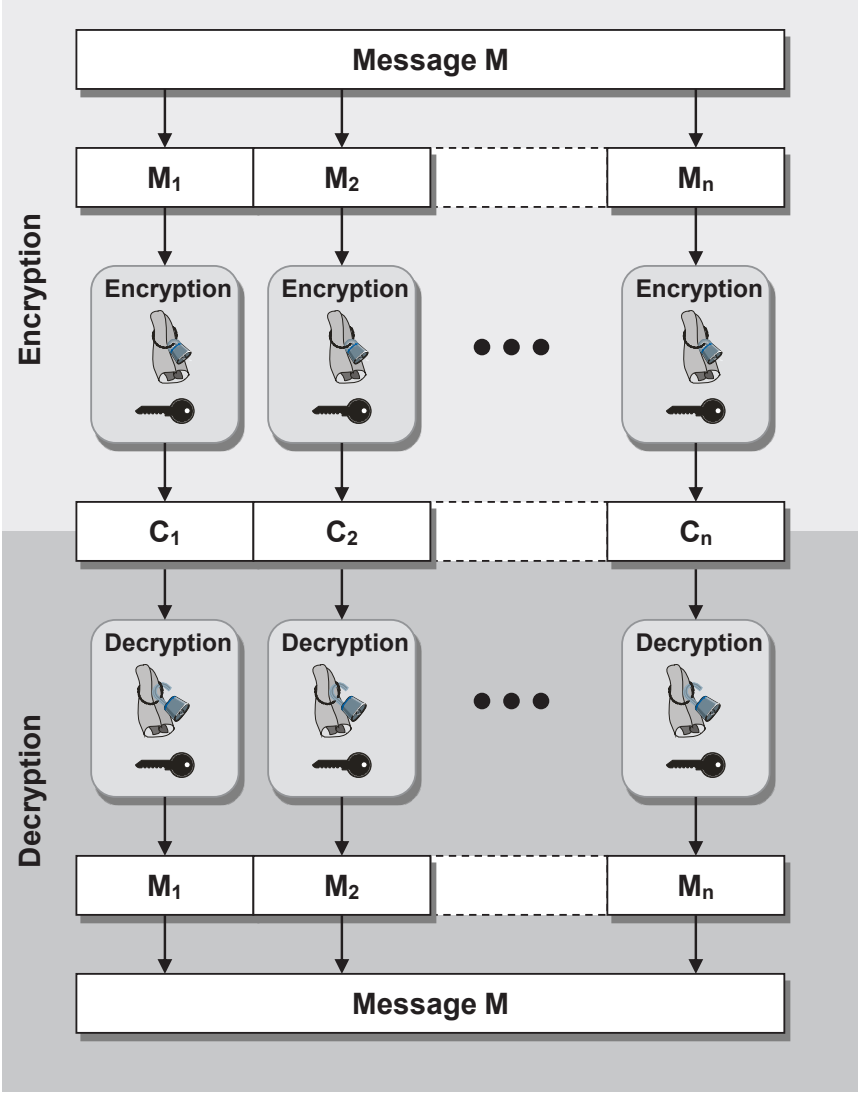


Figure 3.4 ECB Mode of Operation [ISO_10116**]**

3.4 illustrates the structure of the ECB Mode. One or more bit errors would result to errors in the same cipher block. The one or more bit errors in a cipher text block would result in a fifty percent error probability of each plain text bit of the corresponding block. If block boundaries are lost, synchronization between encryption and decryption is lost until the next correct block boundary arrives. The decryption results are incorrect till the synchronization is again established.

3.4.2 Cipher Block Chaining (CBC) Mode

CBC, defined in [ISO_10116], processes each block of data in XOR combination with the cipher text of the previous block. The cipher text is then dependent on the current and all other preceding blocks through the chaining process. Therefore the encryption and decryption of data cannot be

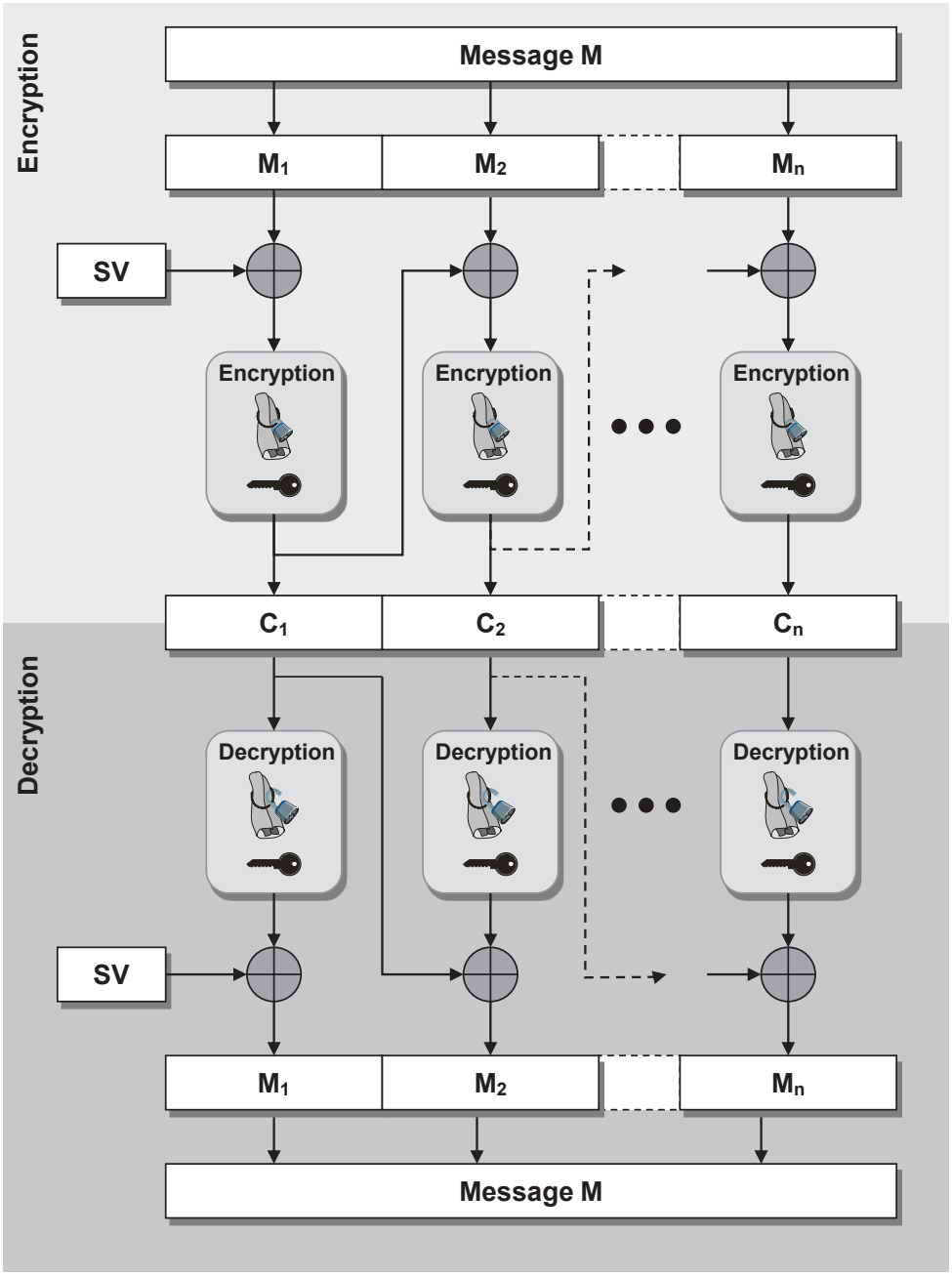


Figure 3.5 CBC Mode of Operation [ISO_10116]

performed independently. For the input of the first block a State Vector SV is used. This way, by altering the SV for each turn, it is prevented that the same plain texts result to same cipher texts. The same block will result to the same cipher text when the same key is used. Use of CBC is therefore not suitable in cases where the same block is repeated. Only n-bit block sizes can be processed by CBC. Therefore it might be a need for other lengths to be padded. Figure 3.5 illustrates the structure of the CBC Mode. One or more bit errors would result to errors in the same cipher block. The one or more bit errors in a cipher text block would result in a fifty percent error probability of each plain text bit of the corresponding block. If block boundaries are lost, synchronization between encryption and decryption is lost until the next correct block boundary arrives. The decryption results are incorrect till the synchronization is again established.

3.4.3 Cipher Feedback (CFB) Mode

CFB, defined in [ISO_10116], processes each block of data in XOR combination with some bits of the previous cipher text block. The cipher text is then dependent on the current and a number of immediately preceding bits of the plain text through the chaining process. Therefore the encryption and decryption of data cannot be performed independently. The same block will result to the same cipher text when the same key is used. Therefore, a State Vector SV is used for the input of the first block. This way, by altering the SV for each turn, it is prevented that the same plain texts result to same cipher texts. Only j-bit block sizes can be processed by CFB. Therefore it might be a need for other lengths to be padded. However, padding is preferably prevented by choosing a suitable j. The strength of CFB is dependent upon the selection of k. Maximum is if k = j. One or more bit errors would result to errors in the cipher text, as long as the errors are still in the CFB feedback buffer. One or more bit errors would result to errors in the same cipher block. The one or more bit errors in a cipher text block would result in a fifty percent error probability of each plain text bit until the errors are out of the feedback buffer. CFB is self–synchronizing. If j-bit block boundaries are lost, synchronization between encryption and decryption is lost until r-bits after the next correct j-bit block boundary arrives. The decryption results are incorrect till the synchronization is again established.

3.4.4 Output Feedback (OFB) Mode

OFB, defined in [ISO_10116], does not provide chaining of the cipher text blocks. The cipher text is then only dependent on the current plain text block. The same block will result to the same cipher text when the same key is produced. Therefore, a State Vector SV is used in addition to the key. This way, by altering the SV for each turn, it is prevented that the same plain texts result to same cipher texts. OFB is more vulnerable to attacks because of the absence of the chaining mechanism. One or more bit errors would result to errors in the same cipher block. If j-bit block boundaries are lost, synchronization between encryption and decryption is lost until the synchronization is reestablished.

3.4.5 Counter Mode

The counter mode with its independent structure can produce higher data rates [NISTCM_07]. Segment numbers allow the encryption and decryption be processed independently. Figure 3.6 illustrates the structure of the counter mode. The encryptor and the decryptor work synchronously and produce the same key stream. For each new encryption process a new State Vector (SV) is used. The infrastructure is composed of several counters and a Galois Linear Feedback Shift Register (LFSR). In counter mode, one or more bit errors would result to errors in the same position of the cipher block. If synchronization between encryption and decryption is lost, the synchronization must be reestablished. The decryptor must be updated with the same SV that the encryptor is using. The encryptor sends the current SV in a resync message. The decryptor starts using this new SV on the next cell received. The resynchronization is one of the negative aspects of this mode, which is critical for high speed transmissions.

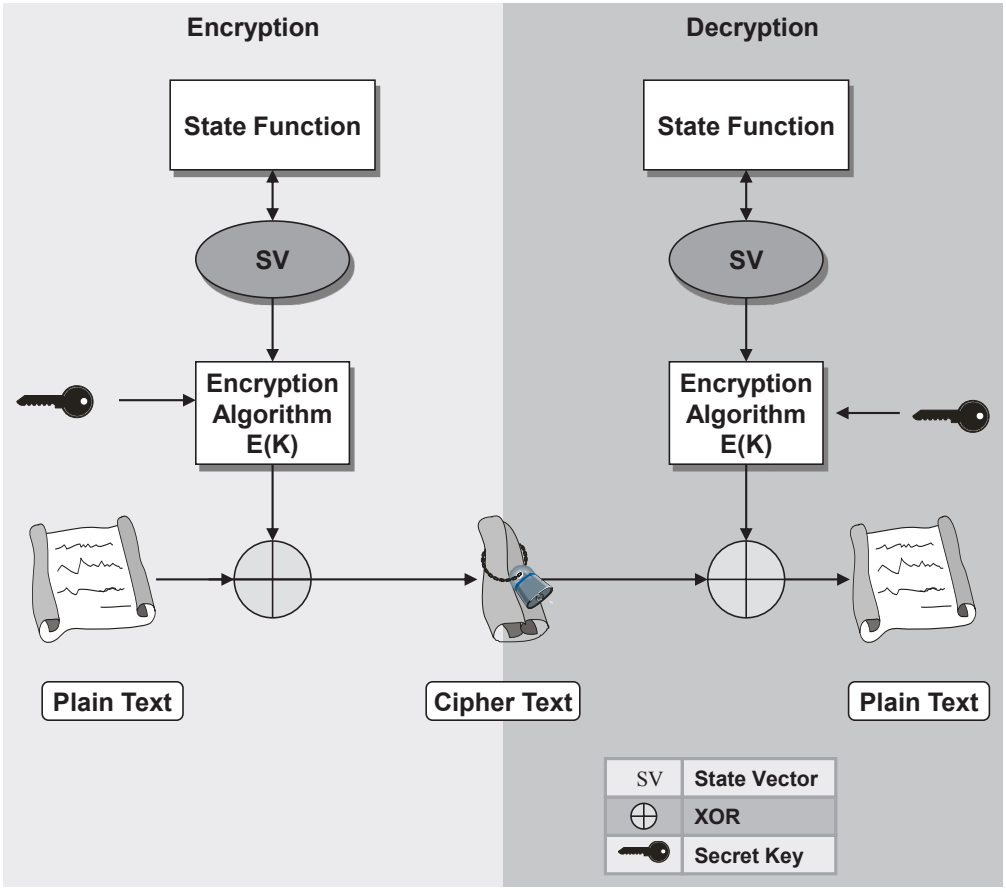


Figure 3.6 Counter Mode of Operation

Chapter

4

Security and QoS Tradeoff

4.1 Security Strength and Performance Research

This research work is mainly concerned with design of efficient security protocols in terms of QoS with special emphasis on the delay component. The proposed protocols, not only guarantee secure communications, but also do it in a delay efficient way. In this context, the first question that comes into mind is, what would the correlation between a security algorithm's strength in terms of security and its projecting operational delay be. There are two terms, whose correlation is in question.

In the first step, we need to know how we can assess and rank the *strength* of different security algorithms. In the literature and in practice, the strength of a security algorithm is mainly evaluated by its invulnerability to various attacks of cryptanalysis. This immunity to attacks, by itself, is influenced by many aspects such as the original design of the algorithm, the method of initial key setup, the number of key changes, the number of rounds in the design of the algorithm to produce final encryption, the used block-size, the key size used for the encryption and the size of data it

is used for [NaJm_05][EAH_10][BiEl_99][SbWd_00]. The strength is easier to evaluate if one is comparing two algorithms of the same design using different key sizes. Of course, one can easily intuitively conclude that the bigger the key size is, the more complex the operation gets, and the stronger the security of the same algorithm will be; i.e. AES 256-bit key is stronger than AES 128-bit key. When comparing different algorithms with diverse design and optimization architectures against each other, this judgment is not trivial. The strength comparison problem has always been one of the most debated issues in the crypto community. At this point, however, the level of confidence created through results of cryptanalycal attacks is the best tool of judgment. Hereby, the personal bias of each designer, having a preference for his own algorithm, need to be cautiously factored in. The consumers and users, however, in addition to above, put additional weight on the scenario of implementation, i.e. used for small sized files or very large ones, the importance of security for the specific data transfer and especially the performance and operational delay.

In the second step, the *performance* and *efficiency* of a security algorithm are of concern for evaluation of usability for a special case. Again here, in addition to the above, it is the combination of different aspects such as the computing environment, i.e. CPU speed, software or hardware implementation, software language choice, the optimization of code, mode of operation and the data transfer size and scenario. Comparing the performance of one algorithm with different implementations, i.e. different sizes of keys or different used modes of operations is less complicated. There have been some published studies comparing different algorithms in terms of different packet sizes, delay and different key sizes. In [NaJm_05], the authors compare the performance of DES [FIPS_463], TripleDES [FIPS_463], AES (Rijndael) [FIPS_197] [NBDFR_00] and Blowfish [Sb_11]. They list the execution time (in our terminology delay) of all of these algorithms in ECB and CFB modes on two different machines: Pentium-II 266 MHZ and Pentium-4, 2.4 GHZ. The implementation is in Java (JDK 1.4). In [EAH_10], the authors compare throughput and execution time (delay) for different file sizes and types and key lengths. None of these studies, however, make a unified conclusion as to the performance and security strength.

In the literature, there have been attempts to propose unified and normalized evaluations of these two aspects of security algorithms. Biham [BiEl_99] suggests the reduction of the number of rounds of each algorithm to the degree that it would still be confidently secure to the known attacks given each particular design. This is with the understanding that some designs are particularly conservative and include additional rounds to make the code *stronger*. This, however, affects the performance and operational delay reversely. The author attempts to find the least common denominator among all security algorithms in terms of security strength and operational performance and efficiency.

Others suggest to stay true to the results of cryptoanalytic attacks. In [SbWd_00], Bruce Schneier and Doug Whiting propose this unified front by incorporating the maximal number of rounds of each algorithm for which the best cryptoanalytic attack is less complex than 256-bit brute-force search and call it *Maximal Insecure Variants*.

The standardization of security algorithms by specific standardization organizations or governmental standards, helps their widespread implementation for particular scenarios of strength and performance. For example, National Institute of Standards and Technology, NIST, has selected *AES-Rijaendal* as the winner of a national competition and recognized it to be the most secure algorithm showing the best performance [FIPS_197]. AES is now the standard algorithm for the US Federal Government.

Whichever way one chooses to assess the performance and strength, the crypto community agrees upon the one fact inarguably that security operations introduce delay to the network. This is the important motivating factor for this research work. For our work, we assume that each node and security agent knows its own operational delays for each security algorithm and has them listed in a table.

4.2 Game & Mechanism Design Theoretic Research

The game theoretic approaches have been proposed for modeling and optimization in the different areas of the field of communications networks for quite some time. Implementation of mechanism design theory in this field is also becoming increasingly popular.

These approaches in the areas of network security and its impact on performance and resource management focus on variety of different problems. Theodorakopoulos and Baras [TgBj_08] enforce upper bounds on damage from malicious users. Chen, and Leneutre [ClLj_09] derive the expected behaviors of rational attackers, the minimum monitor resource requirement, and the optimal strategy of the defenders and then provide guidelines for IDS design and deployment. Otrok et al. [OMWDB_08] address the tradeoff between security and the resource consumption of IDSs for prolonging the lifetime of nodes and increasing their security in MANETs. Liu et al. [LZY_05] present a general incentive-based method to model and infer attacker intent, objectives and strategy. Bohacek et al. [BHLLO_07], introduce the Game-Theoretic Stochastic Routing (GTSR) framework, to make connection eavesdropping attacks maximally difficult and address the tradeoff among security, fault-tolerance, delay and throughput. Chen and Wu [CsWm_10] improve routing security risk and delivery ratio according to a tradeoff coefficient.

As pointed out above, we can see that the use of game theory and mechanism design theories is quite popular in addressing the security and overall performance of communications from diverse angles and with different focuses. To the best of our knowledge, there has been no mechanism design theoretic approach to design of delay-efficient security protocols. The research work presented here introduces yet another promising perspective and implementation use of these theories.

4.3 Network Layer Specific Research - IPsec

At the Network Layer, in the IP paradigm, the *Integrated Services (IntServ)* architecture [RFC_1633] is designed to support an end-to-end capability providing "*Guaranteed QoS*". The intolerant applications can take advantage of the *Guaranteed Service* model, which commits to an absolute reliable upper bound on delay and is the focus of this work [RFC_2212]. On the security side, *IPsec* [RFC_4301] offers security services at the IP layer. Through IKEv2 protocol [RFC_5996], it provides the capability to select required security protocols, determine algorithms to use for the services and use cryptographic keys required to provide these services. The corresponding *Security Associations (SA)* or SA bundles are implemented accordingly to a particular packet and the security service(s) are

provided. This IPsec [RFC_4301] and IKEv2 [RFC_5996] specifications do not provide a solution to the problem of causing additional degradations to the QoS of the network. As we can see even at this layer the existing standards coexist separately. As part of this research work, we propose IPsec-O and IKEv2-O as a delay-efficient IPsec.

4.4 Datalink Layer Specific Research - ATM Security

At the Datalink Layer, the *ATM Security Specification Version 1.1* [SEC_11] defines and standardizes functions and procedures to support security objectives in ATM networks. Security services are defined to support negotiation of multiple standardized and non–standardized security algorithms and modes of operation to allow interoperability in wide area networks. Two security protocols are defined for this purpose, The *Out-band Security Message Exchange (SME)* and the *In-band SME*. The *Out-band SME* supports security negotiation during the establishment phase of an ATM connection using specific signaling Virtual Channels. *In-band SME*, however, communicates these objectives through the already established user channel after the connection establishment phase. This specification does not offer a solution to the problem of causing additional degradations to the QoS of the network. The *ATM* Standards, *UNI Signaling Specification 3.1* [USIG_31] and *UNI Signaling Specification 4.1*[USIG_41], in addition to the *ITU–T Recommendation Q.2931* [ITUQ_2931] have defined signaling protocols for handling the QoS negotiations in ATM SVCs in user to network interfaces. These parameters are determined in the connection establishment phase and are to be met or exceeded by the intervening networks during the course of the ATM connection. After the agreement upon the QoS parameters the ATM connection is established and the ATM endpoint observes a consistent level of QoS for its connection. The end-to-end objective of QoS is then met. *UNI Signaling Specification 4.1* [USIG_41] defines the information elements and procedures required for specifying individual QoS parameter values for the ATM connections. This standard defines *Cell Transfer Delay (CTD)*, *Cell Delay Variation (CDV)* and *Cell Loss Ratio (CLR)* QoS parameters to be negotiable during the ATM connection establishment phase. The negotiation of these parameters is either done individually or on the basis of predefined QoS Classes.

The existing standardized ATM security protocols to date do not take the degradations of requested and agreed upon Traffic Quality of Service caused by security implementations into account. These protocols coexist and research each area of interest rather separately without regard to above problem. It is the goal of this research work to bring the separately defined and standardized Traffic QoS signaling procedures and security protocols together and enhance the decision making and negotiation for this tradeoff.

Chapter

5

The DSIC-S Protocol

The proposed DSIC-S Protocol has been peer-reviewed, accepted and will be published in the International Journal of Critical Infrastructures in 2012[SfMi_111].

5.1 Introduction

In this chapter, the use of game and mechanism design theory principles is proposed to model and develop a cross layer network security protocols.

In this framework, the building blocks of the game theory map in a natural and elegant way to those of the network security arena. In a strategic form game, we can summarize three structural components: The players or agents, their strategies and their utilities [MyRb_97]. In the network security realm, the agents become the security agents at each node, the strategies are the security policies governing each user and domain, and the utilities are the achieved gain from the combination of the level of transmission security, reduction of resource consumption and in this special case, the optimization of network-wide delay and improvement of reputation and ranking. Mechanism design theory induces a desired system-wide outcome designed by a social planner. This is a perfect

tool to use and make the user or consumer act as the social planner. She can then define and induce her desired security and delay outcome for her connection, in a way, that is preferred by all selected nodes and agents along the path. Mechanism design theory properties and theorems can be used to ensure the truthful and cheat-proof implementation of this outcome.

The underlying nodes are owned by autonomous users, domains, and network service providers, who are rational and selfish. Their main goal is to maximize their own individual and company-wide interests of security, reputation, economics and Quality of Service (QoS). In an end-to-end perspective the question is how to get around the selfishness of the traversing domains to achieve a network-wide desired outcome, preferably a Pareto optimal solution for all. Selfishness can be detrimental to the network as a whole. Each node or domain along the path of transmission needs to be willing to pass the packets within the end-to-end desired attributes - here the consumer requested values of security and delay.

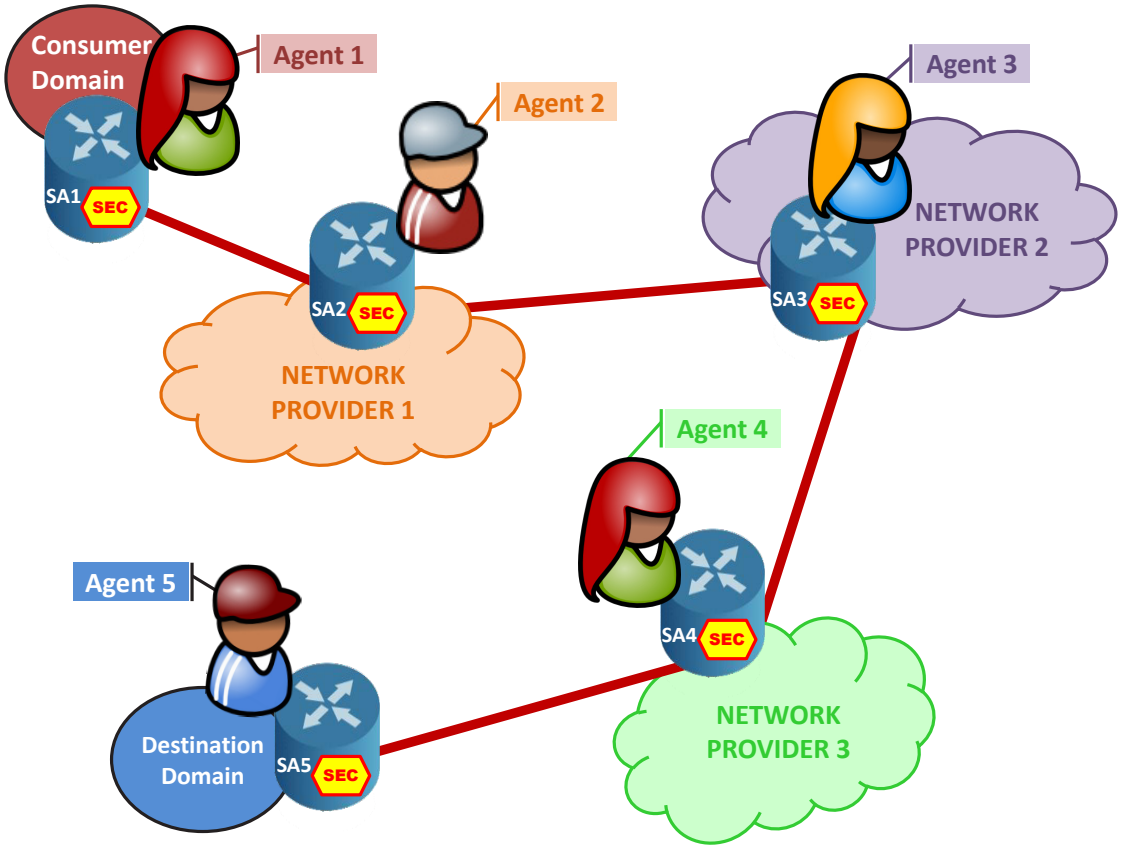


Figure 5.1 **Example of a Security Strategic Game's Components**

Each domain, however, has its own governing security and QoS policies in place, which maximizes its own utility or benefit. To model this in the most simplest way, one network device per domain is considered, as depicted in an example in Figure 5.1. Each node includes a Security Agent (SA). Each SA knows its own delay costs per security algorithm operations, which are only known to itself and are its own private values or types. In a network-wide game as depicted in Figure 1, Agents 2 through 5, governed by their domain security policies, might only think of their own benefits and follow a strategy, which is not in the interest of others in the network. One may choose not to care about the security of traversing packets through its domain and decide to save resource consumption by providing lower security measures. Another agent, however, again motivated by selfishness and desire to improve consumer satisfaction and increasing her reputation may think to make her domain most secure by implementing only the highest secure algorithms, not caring about the end-to-end network delay. Such rational behavior may lead to a sub-optimal state for the network as a whole. In the worst case and depending on the number of agents and traversed domains, this can lead to an undesirable situation, where the delays add up, yielding to dropped packets or cancelled communication.

Although the overall security of the network should not be compromised, at the same time, the end-to-end delay is of utmost importance for delay sensitive applications and operations.

In this part of our research, we explore and propose a protocol to make a tradeoff between security and the delay component of QoS. These delays are proposed to be listed in a Security Agent Delay, SAD, table at each SA, as depicted in Table 5.1. We propose a scenario-based model for three different levels of security. In this model, the security algorithms for each level are ranked based on the valuation of the level of security they offer and their occurring delay. We propose the use of incentives to counteract the selfishness of the players to participate voluntarily in a manner requested by the consumer or social planner of the mechanism implementing *individual rationality*. We incorporate appropriate allocation rules and VCG payment scheme [ViWi_61] with Clarke's pivotal rule [ClEd_71] in the security protocol to implement the Social Choice Function (SCF) truthfully in dominant strategy. To enforce the security and delay trade-off in the decision making

SAD Table			
Security Service	**Security Scenario**	**Security Algorithm A_i**	**D_i ns**
Confidentiality	**High-Security**	**AES- Rijndael-256**	**126**
		AES- Rijndael-192	**108**
		AES- Rijndael-128	**90**
	Medium-Security	**Twofish**	**80**
		Blowfish	**100**
		Triple-DES	**345**
	Low-Security	**RC6**	**80**
		RC5	**125**
		DES	**215**
Data Integrity	**High-Security**	**SHA-2-516**	**90**
		SHA-2-256	**80**
		SHA-1	**65**
	Medium-Security	**RIPEMD-160**	**80**
		Triple-DES	**345**
		AES-Rijndael-128	**590**
	Low-Security	**MD-5**	**45**
		DES	**160**
		RC6	**215**

Table 5.1 Example of a Security Agent Delay, SAD, Table

of the nodes, we propose that the VCG payment calculation be a function of the valuation and rank of the announced security algorithm based on consumer's valuations and desired security or delay, depending on the governing security scenario. We propose specific weighted formulas according to the governing scenario for the calculation of these ranks. These valuations are proposed to be listed in the consumer's Security Agent Ranking, SAR, table shown in Table 5.2. Each Agent will also have her own ranking and valuations of the security algorithms according to her own preferences.

In summary, the contributions of DSIC-S research and development are as follows: We introduce the implementation of mechanism design theory for the design of a consumer-centric network security protocol providing security-delay tradeoff, DSIC-S.

SAR Table				
Security Service	**Security Scenario**	**Security Algorithm A_i**	**V_{Ai}**	**V_{Di}**
Confidentiality	**High-Security**	**AES- Rijndael-256**	3	1
		AES- Rijndael-192	2	2
		AES- Rijndael-128	1	3
	Medium-Security	**Twofish**	2	3
		Blowfish	3	2
		Triple-DES	1	1
	Low-Security	**RC6**	3	3
		RC5	2	2
		DES	1	1
Data Integrity	**High-Security**	**SHA-2-516**	3	2
		SHA-2-256	2	3
		SHA-1	1	1
	Medium-Security	**RIPEMD-160**	3	3
		Triple-DES	2	2
		AES-Rijndael-128	1	1
	Low-Security	**MD-5**	3	3
		DES	2	2
		RC6	1	1

Table 5.2 Example of a Security Agent Ranking, SAR, Table

We solve the important problem of misrepresentation of agents' private information in mechanism design and revelation theories for a delay-efficient security protocol through the proposed DSIC-S design. We incorporate a valuation system to integrate the caused delay at each node in selection of security algorithms without consumer's knowledge of the actual delays. The incentive model uniquely uses our proposed consumer's preference valuation system based on different security levels as an input for the VCG payment scheme [ViWi_61] with Clarke's pivotal rule [ClEd_71] and for the credit transfers.

DSIC-S achieves network-wide and individual Pareto optimality, as proved in Section 7.7, Theorem 10. This is an extremely desirable feature, which enforces natural adherence of the agents

to the mechanism rules. DSIC-S is consumer-centric. The consumer is the social planner of the mechanism to request the desired security and delay level for her data transmission. This is one of the main goals of this research work and a very sought-after proposition. This enforces a natural and automatic control of the behavior of the underlying participants. The consumer can rest assured that her services are provided as expected and promised to him.

DSIC-S is cheat-proof and strategy-proof, as shown in Section 7.8, Theorem 9. This is a strong property for real life applications, which enables the nodes (i.e. network providers) to naturally act responsibly and truthfully. Dominant strategy makes each node's best response independent of other nodes' choices of decisions and their belief functions.

DSIC-S is scenario based. It considers different levels of security and the resulting tradeoff manifestation. This focal design enforces Pareto optimality, individual rationality and allocative efficiency, as shown in Sections 6 and 7, Theorems 7, 8 and 10.

5.2 Design Assumptions & Constraints

DSIC-S is designed to comply with a collection of constraints and assumptions, as listed next. The players of this induced security strategic game are the Security Agents (SA) residing in the network devices of different domains, as depicted in Figure 1. Since the devices in one domain would follow the same security policy and the same utility goal and selfishness, We assume one device per domain as representative and agent for that domain. We further assume that the values of delay caused by security operations are available and consider them to be the agents' private information. In this paper, the term "delay" is used as "security delay" or delay caused by security operations, which is specific to each SA. Furthermore, each delay is specific and correlated to a security algorithm at each device and node affected by different factors described in Section 3. We do not assume any specific proportionality of correlation between delay and security strength. We further assume that the average delay value for the encryption and decryption operations are listed in their SAD table, as in Table 1.

The security protocol is an extension to existing security protocols of the underlying network. We therefore assume the existence of basic security negotiations along with the possibility of nesting

security associations, QoS management and routing mechanisms. We assume the availability of communication between these systems for resource reservation and routing calculations. Since delay efficiency is one of the main goals, there will be a synergetic effect to use DSIC-S in cooperation with a least cost routing mechanism.

The protocol does not require any special agreements among the network providers along the path in terms of security or QoS. To address this, We propose a nesting security model, which requires associations to be established each time between the consumer and a participating node along the path, as illustrated in Figure 2. Since the players are selected at the beginning of the game based on their achieved reputation from previous games and the allocation rule, we assume that for each connection at least one end-to-end security association between the consumer and the destination is established. In addition to above criteria, the number of selected players and established associations is constraint by the social planners desires of general security and delay conditions. The protocol is designed for 3 or more players, so we assume there will always be at least one provider along the path. For the focus on the main objective, we illustrate our solution in this paper based on the scenario that for every communication two security services, i.e. confidentiality and data integrity, are to be implemented by the nodes.

In the game theoretic sense, the SAs, through their domains' individual policies, are considered "rational' and consistently in the pursuit of maximizing their own utility and reaching their own objectives. To achieve incentive compatibility and strong dominance in our Bayesian game, we use the VCG payment model with Clarke's pivotal rule, therefore assume a quasi-linear environment. We consider the consumer to be the social planner for the induced network-wide security protocol. Our consumer-centric mechanism, induces a consumer requested cheat-proof outcome for her data communication, which ensures all network providers and domains along the path to deliver the services promised to her truthfully.

5.3 Security Scenario Model

One of the main properties of the DSIC-S protocol is its scenario-based design pertaining to the requested level of security. We propose three distinct levels of security, namely High Security,

Medium Security, and Low Security. These levels are determined and selected by the consumer. It is consumer's request for a security condition for a given connection, which sets the security strategic game.

In a "High Security" scenario, the strength of the network security takes precedence over the delay. The network might be a little slower, but it should be strongly secured. Each node chooses its strongest preferred security algorithms for this scenario. In this case, DSIC-S ranking calculations are done with a stronger weight on the strength of security as given by (3). The credit payment calculations are also done based on and influenced by these achieved rankings according to the consumer's SAR table.

In a "Medium Security" scenario, although there is still demand for moderate security, but the delay is of a great importance to the consumer. In this case, each node chooses moderate strength security algorithms according to its preference. DSIC-S ranks these algorithms with emphasis on their impact on delay in a weighted form as given by (4). The credit payment calculations to each player are also done based on these achieved rankings according to the consumer's SAR table.

In a "Low Security" scenario, we still would like to have some security in place but not as critical as the last two scenarios. This case basically enforces some security control as opposed to not having any in place at all. In this case, each node chooses its most relaxed security algorithms according to its preferences. DSIC-S ranks these algorithms also like the medium security case with emphasis on their impact on delay as shown in (4). The credit payment calculations are done based on these achieved rankings according to the consumer's SAR table.

This scenario-based structure enforces delay-efficiency and resource consumption management and reduction particularly for cases, where no strong security is needed. This is one of the most impending concerns of the underlying nodes attributed to their selfishness. This way and along with a proper design of an allocation function and incentive model, as we will see in the following sections, DSIC-S enforces a natural equilibrium among the players. The agents find it to their best interest to voluntarily follow the consumer's directions and demands in terms of level of security and QoS.

5.4 Valuation & Ranking Model

In DSIC-S, our proposed cross layer protocol, We build upon the individual agent's preference and valuation of a security algorithm influenced by selfishness, personal bias, resource consumption concerns, domain policy, and delay. All nodes have a valuation or ranking table, SAR Table. Three security levels are proposed for this ranking: 1: High-Security, 2: Medium Security and 3: Low-Security scenarios. These are described in more detail in the next section. Table 2 depicts an example of a node's SAR table. The rankings are done within each scenario for the preferred available security algorithms pertaining to that particular scenario. The ranking of the valuation of a security algorithm in SA_i is

$$V_{Ai} \in [\underline{V}_A , \ \overline{V}_A] \subset [1, 3] , \tag{5.1}$$

with value of 3 for the most preferred security algorithm per scenario, which they have the highest desire to select and perform first. The SAR table also has a valuation ranking regarding the delay (listed in the SAD table) caused by that particular security algorithm A_i. The ranking of the valuation of the delay D_i in SA_i is

$$V_{Di} \in [\underline{V}_D , \ \overline{V}_D] \subset [1,3] , \tag{5.2}$$

with value of 3 for the lowest delay, most desirable, caused by a security algorithm per scenario. Table 1 shows this combination.

5.4.1 DSIC-S Consumer's Valuations & Rankings

m and security protocol has its own valuation of different security algorithms for different scenarios in its SAR table. The consumer's SAR table is the main valuation table for the security strategic game and is known to all players and Security Agents along the path. The incentives for the truthful participation are designed and calculated using this main table's valuations. Each player and Security Agent makes its strategic decision based on its own valuations and that of the consumer's for each specific scenario calculating its own utility.

Two different valuations, V_A , security algorithm weighted, and V_D, delay weighted, are calculated for each algorithm. In order to evaluate these valuations, first two total weighted rankings need to be calculated, one with emphasis on the security level, R_{AT} , Equation 5.3, and another with attention to the delay, R_{DT}, Equation 5.4.

The following describes the components of these calculations by using the example SAR in Table 2 in a high-security scenario. First, the three algorithms are ranked according to the achieved strength of the security. In this example, it is intuitively easy to see that the first one, AES-256, gets the highest ranking of $R^A_{AES\text{-}256} = 3$, because of its bigger key size and additional security compare to the next two algorithms in this scenario. With the same reasoning, the middle ranking goes to AES-192, $R^A_{AES\text{-}192} = 2$, and AES-128 gets ranking $R^A_{AES\text{-}128} = 1$.

On the other hand, the ranking of these algorithms based on their impact on delay can be directed from the SAD table in Table 1. The delay ranking for these three algorithms in this case is reciprocated to the one of the security as follows: $R^D_{AES\text{-}256} = 1$, then AES-192, $R^D_{AES\text{-}192} = 2$, and AES-128 gets $R^D_{AES\text{-}128} = 3$, with the latest being the fastest and the best in terms of delay performance. The proposed total weighted formula for the calculation of this scenario for each algorithm is as follows:

R_{AT} is weighted based on the preference and ranking of the security algorithm and its strength, hence, R^A square, multiplied by the delay ranking, R^D, according to the SAD table. That is

$$R_{AT} = ((R^A)^2 * R^D)^{1/3} \tag{5.3}$$

R_{AT} is calculated for each algorithm and each security association. Figure 5.2 illustrates theses calculations as an example of confidentiality. The total ranking in case of more than one security association will be the arithmetic average of the values for all associations implemented by a node. V_A is then determined by comparing the other resulted R_{AT} values in the scenario giving valuation ranks according to Equation 5.1, as listed in Table 2.

R_{DT} is weighted based on the occurring delay, defined as

$$R_{DT} = ((R^D)^2 * R^A)^{1/3} \tag{5.4}$$

R_{DT} is calculated for each algorithm and each security association. Figure 5.2 illustrates theses calculations as an example of confidentiality. The total ranking in case of more than one security association will be the arithmetic mean of the values for all associations implemented by a node. V_D is then determined by comparing the other resulted R_{DT} values in the scenario giving valuation ranks according to Equation 5.2, as listed in Table 5.2.

Depending on the security scenario, one ranking is preferred over the other for VCG calculations. For example, in a High-Security scenario, where the utmost goal is to secure the network, the mechanism takes V_A into consideration for selecting the most fitted and desired security algorithm. This selection is highlighted for the first three algorithms in Table 5.2. As also depicted there, in other two scenarios of medium and low security, V_D accentuates the selection process.

Confidentiality

	Customer Node SA1									
	SAR Table	R^A	R^D	$R^A^{2*}R^D$	$R^D^{2*}R^A$	R_{DT}	R_{AT}	V_D	V_A	Customer D1
1	AES- Rijndael-256	3	1	9	3	1.4	2.1	1	**3**	126
2	AES- Rijndael-192	2	2	8	8	2.0	2.0	2	**2**	108
3	AES- Rijndael-128	1	3	3	9	2.1	1.4	3	**1**	90
4	Blowfish	3	2	18	12	2.3	**2.6**	**2**	3	100
5	DES	1	1	1	1	1.0	1.0	**1**	1	215
6	Twofish	2	3	12	18	**2.6**	2.3	**3**	2	80
7	RC6	3	3	27	27	**3.0**	**3.0**	**3**	3	80
8	RC5	2	2	8	8	2.0	2.0	**2**	2	125
9	Triple DES	1	1	1	1	1.0	1.0	**1**	1	345

Figure 5.2 Consumer's Ranking and Valuation Calculations

5.4.2 DSIC-S agents' Valuations & Rankings

Each player and Security Agent makes its strategic decision based on its own valuations and that of the consumer's for each specific scenario calculating its own utility. They rank their preferred security algorithms for the three security scenarios based on their economics, availability of algorithms, and resource management, to name but a few.

As a consequence each agent's SAR table may be different than others. One agent might not support all the algorithms listed in the consumer's table. In some cases, it might even happen that one agent might rank one security algorithm for a different security scenario.

DSIC-S is designed to be implementable in any of these cases, enforcing the selection of the highest valued security algorithm by the consumer. As we will see in the next chapter, the agents will inherently have a higher utility choosing these most desired algorithms.

5.5 Security Association Model

In order to enforce the independence of the model of the individual security contracts among the agents, DSIC-S protocol integrates a nesting security association model along the connection path. These associations are always between the consumer's node and each of the agents' nodes. This ensures that security and all the other desired attributes are not neglected because of the consequences of rationality, selfishness and gain in utility by the agents. Figure 5.3 illustrates DSIC-S security association model in an example. As depicted here, DSIC-S proposes a nesting security model. As described in the assumptions section, it is assumed that the underlying network security infrastructure allows nesting security associations.

5.6 DSIC-S Protocol Procedures

The protocol is designed for a multi-agent system, with 3 or more agents. There are N_p = {3, 4,..., n} players or SAs along the path of a connection. The numbering of the nodes, $i \in N = \{1, 2, 3, ..., n\}$, starts with the consumer's node (i=1) as depicted in Figure 5.1. This node is the social planner and induces its desired outcome of security level and delay through the whole communication path by establishing nested and iterated security associations with all nodes along the path as depicted in Figure 5.2. This way the mechanism and protocol make sure that proper network security between the domains and within them is implied, without requesting any special agreements among the network providers. In her design, the consumer also chooses the nodes along the path of her communication according to a reputation model, which is based on the calculated payments from previous games. It is then imperative for the network providers to earn excellent incentive credits and reputation. This allows them to attract more customers. More consumers want to become their permanent clients. This raises their revenues and brings them their sought after economic gain.

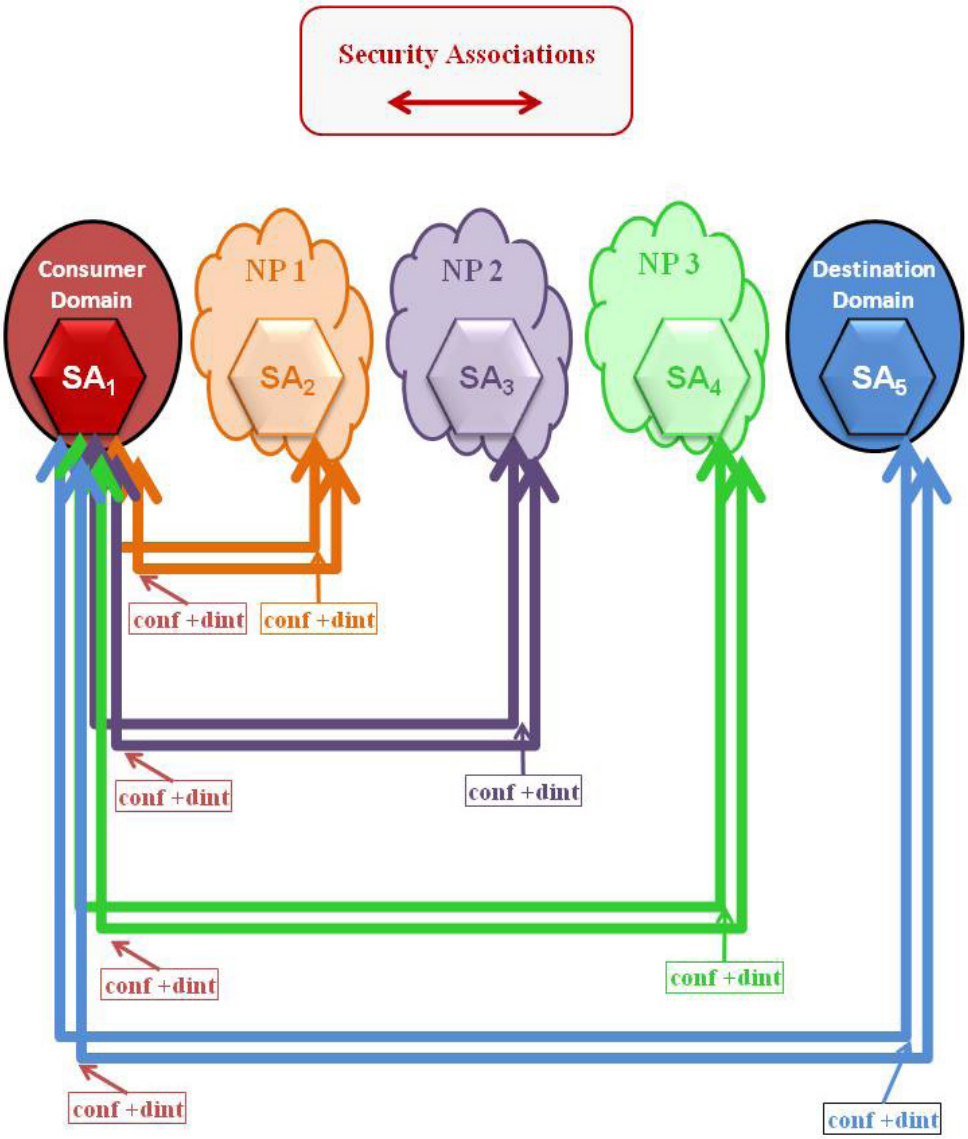


Figure 5.3 DSIC-S Protocol's Security Association Model in an Example

At the initiating phase of a connection, SA_1 first identifies the selected nodes along the path according to their reputation resulted from past games. Then it starts the mechanism by requesting a certain level of security from the three proposed high, medium or low security scenarios in the Flow1 of communication as depicted in Figure 5.4, Step 1. It identifies the security algorithms for both confidentiality and data integrity and their valuations within the desired level from its SAR

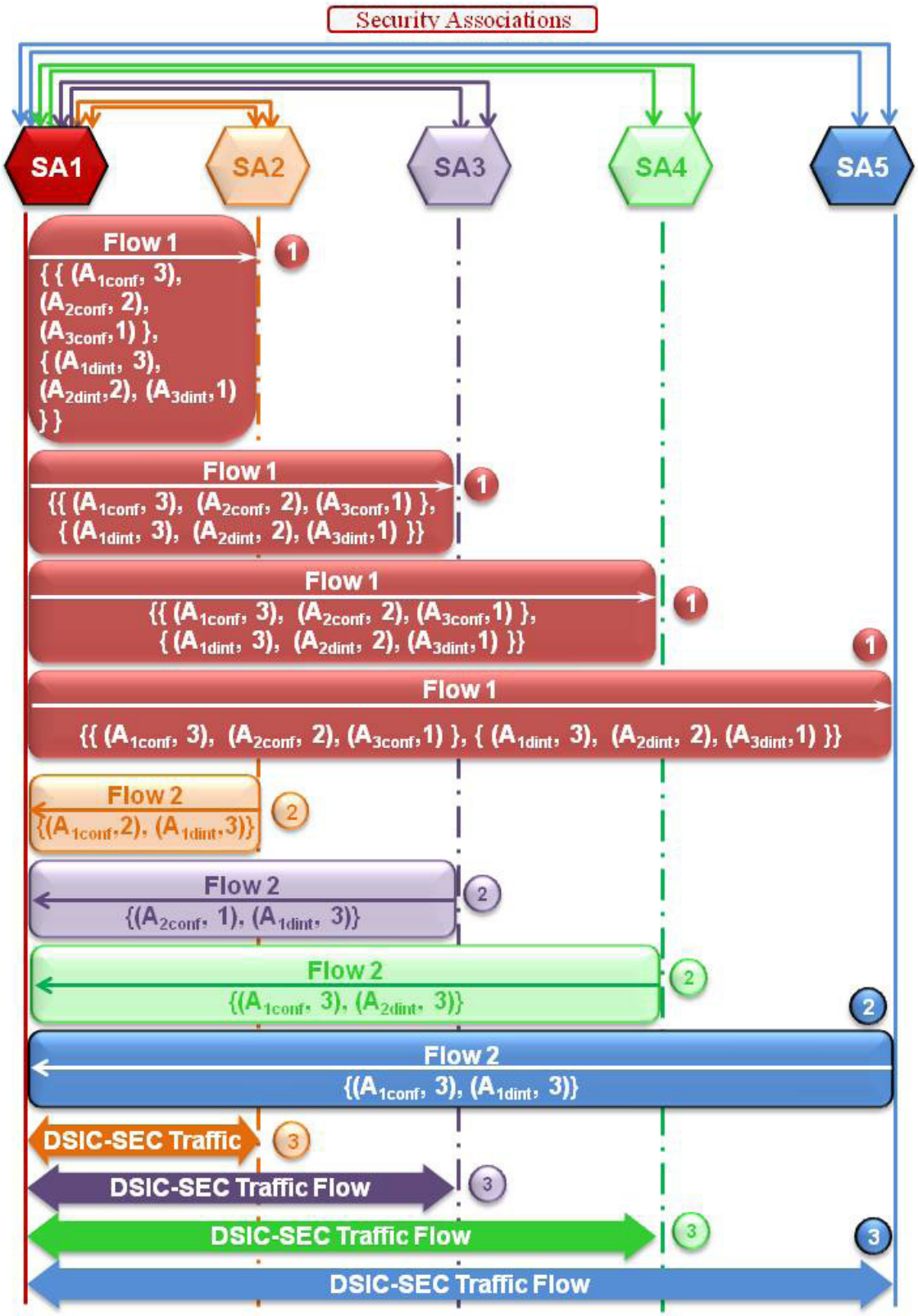


Figure 5.4 **DSIC-S Security Association Negotiation in an Example**

table and sends these to all SAs along the path. For this process, V_A is calculated as described above and used for the high-security scenario algorithms. V_D calculated as described above and is used for the algorithms in medium and low security scenarios. The SAs then according to their own preferences in their SAR table and implication of their selfishness, utility and gained reputation select one security algorithm for each security service and communicate this in the Flow2 to SA_1 as depicted in Figure 5.4, Step 2. If they can not accommodate the conditions asked they will be dropped from the game according to the allocation rule of Equation 6.7. Once all Flow 2 replies have received the consumer can decide if enough players can qualify for the game. If not the consumer cancels this game and repeats its selection for another game. There will be a negative credit note according to the game and the transfers, that those players, not willing to participate or not qualifying to participate, would have earned if they had qualified. We propose this to be integrated into the reputation model.

If all or an adequate number of the intended players qualify, SA_1 can establish its DSIC-S secure connection with each of the SAs along the path, as seen in Figure 5.4, Step 3. It can also now calculate all the incentives through the VCG mechanism and Clarke's pivotal rule described in the next Chapter for each SA for this particular connection's strategic game using its valuation table for the selected security algorithms.

Following the proposition that network providers in real-life need stronger incentives to comply with the theoretical mechanism design models,we recommend the integration of these calculated payments into a reputation based model.

Chapter

6

Modeling DSIC-S Strategic Game

6.1 DSIC-S Strategic Game

To better convey the design, the strategic game for DSIC-S is modeled in the assumed case that two security associations, i.e. confidentiality and data integrity, are to be established for each connection between a node and the consumer's node. Figures 5.2 and 5.3 depict the establishment of these security associations along the path.

In DSIC-S protocol, a connection path is comprised of $N_p = \{3, ..., n\}$ nodes or players including a Security Agent (SA). Let each node have a SAR table, Table 5.2, listing its preference valuations, V_A and V_D, for different security algorithms for the two security services, confidentiality and data integrity, in different scenarios of High-Security, Medium-Security and Low-Security. Let $j \in \sigma = \{conf, dint, ...\}$ the set of security associations for each node, then each type of an agent can be defined to be

$$\theta_i := \cup_{j\in\sigma} \theta_{i,j} = \theta_{i,conf} \cup \theta_{i,dint} \quad (6.1)$$

yielding to

$$\theta_i := \begin{cases} \cup_{j\in\sigma}(A_{i,j}, V_{Ai}) = \{(A_{i,conf}, V_{Ai}), (A_{i,dint}, V_{Ai})\} & \text{If High-Security} \\ \cup_{j\in\sigma}(A_{i,j}, V_{Di}) = \{(A_{i,conf}, V_{Di}), (A_{i,dint}, V_{Di})\} & \text{Otherwise} \end{cases} \quad (6.2)$$

will be the type for SA_i indicating its valuation of a security algorithm for each security service depending on the governing security scenario. It is assumed that each agent's type, θ_i, is independent of other agents' values and is its private information except for that of SA_1, which belongs to the consumer and is common knowledge to all. Since the proposed protocol is a dominant strategy mechanism as it is proved in Chapter 7, Theorem 8, the strategies of the agents will be their best response to any response of other agents, hence independent of their believes of the types of other agents. Let Θ_i be the set of all types of each node and $\Theta = \prod_{i\in N} \Theta_i$ be the set of all type profiles of all nodes with $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ being one specific type profile of all nodes. Further, let $e \in \varepsilon = \{1, 2, \ldots, n\}$ be the set of number of security algorithms designated to each scenario. Let X be the set of different possible and allowable outcomes and the union of three sets of outcomes of the protocol and mechanism pertaining to the three different security level scenarios, given by

$$X := H_1 \cup M_1 \cup L_1 \quad (6.3)$$

H_1 is the set of all security algorithms for the high-security case of SA_1 expressed as

$$\begin{aligned} H_1 &= \cup_{j\in\sigma,\, e\in\varepsilon}(A_{iH,j,e}, V_{AiH,e}) \\ &= \{(A_{1H,conf,1}, V_{A1H,1}), (A_{1H,conf,2}, V_{A1H,2}), (A_{1H,conf,3}, V_{A1H,3}), \\ &\quad (A_{1H,dint,1}, V_{A1H,1}), (A_{1H,dint,2}, V_{A2H,2}), (A_{1H,dint,3}, V_{A1H,3})\} \end{aligned} \quad (6.4)$$

i.e. in our example: $(A_{1H,conf,1}, V_{A1H,1}) = (AES\text{-}128, 1)$. In the same manner, the following can be defined,

$$M_1 = \cup_{j\in\sigma,\, e\in\varepsilon}(A_{iM,j,e}, V_{DiM,e}) \tag{6.5}$$

and

$$L_1 = \cup_{j\in\sigma,\, e\in\varepsilon}(A_{iL,j,e}, V_{DiL,e}) \tag{6.6}$$

As we can see, the set X of all possible outcomes is determined by consumer's valuations as a social planner. For a particular connection, the consumer's node chooses the participating nodes and players according to their reputation based on previously earned credits in the DSIC-S security strategic games. The selected nodes then use $k(\theta)$ as their allocation rule, which selects one particular security algorithm from set X based on θ and pertaining to the selected security scenario, therefore

(6.7)

$$k(\theta) = \begin{cases} 1 & (\forall\ \theta_i \in H_1 \ \&\ \text{High-Security}) \text{ or} \\ & (\forall\ \theta_i \in M_1 \ \&\ \text{Medium-Security}) \text{ or} \\ & (\forall\ \theta_i \in H_1 \ \&\ \text{Low-Security}) \\ 0 & \text{Otherwise} \end{cases}$$

Implementing Mechanism Design concepts, the Social Choice Function (SCF) in our quasi-linear environment will be

$$f(\theta) = (k(\theta), t_1(\theta), ..., t_n(\theta)) \qquad \forall\ \theta \in \Theta \tag{6.8}$$

The vector

$$t(\theta) = (t_1(\theta), t_2(\theta), ..., t_n(\theta)) \qquad \forall\ \theta \in \Theta \tag{6.9}$$

constitutes the incentive vector transferred to the nodes, given the type profile $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ according to Equation 6.13.

The utility function u_i will be the motivating factor for each node in selection of its strategy and final decision of truthful participation. According to its general structure in quasi-linear environments, where incentives are calculated as payments, this utility is comprised of the node's valuation, v_i, of the security algorithm given the allocation function. In our case, this will translate to its preferences in its SAR table, its type, given by

$$v_i(k(\theta), \theta_i) = \theta_i \tag{6.10}$$

of course, in addition to its gained transfer $t_i(\theta)$

$$u_i(\theta) = v_i(k(\theta), \theta_i) + t_i(\theta) \tag{6.11}$$

In our security strategic game, each participating node adhering to the allocation function for the given scenario will be reimbursed by a positive $t_i(\theta)$, Equation 6.13. In order to enforce the incentive compatibility through these payments and bring the theoretical model closer to the real-life scenario, we suggest the use of a reputation system. These payments should be then applied as an input into the system. This reputation will be a vital factor for the participating network providers to increase the consumer confidence and to attract more customers resulting to an increase in revenues and sought after economic gain.

6.2 Security Agent's Strategic Decision

Each node along the path will make a strategic decision if it would like to participate in this game at all or not. According to the theories of mechanism design and governed by selfishness, each node will participate voluntarily if it is not worse off after the participation. That means, it calculates its utility function to see if it is worth for her to play. As described above, she uses Equation 6.11 to calculate this utility. Each node knows that $t_i(\theta)$, Equation 6.13, is a positive reimbursement, given its truthful participation, but it is dependent on its selected θ_i and that of the consumer's θ_1.

This means, the factors governing her utility function to make it a positive one for him, are its own preference table and $v_i(k(\theta), \theta_i)$ of Equation 6.10. As long as she can choose one of the algorithms listed and requested by the consumer's scenario, it always can achieve Individual Rationality and score a positive utility function. The Individual Rationality property of DSIC-S is proved in Theorem 8 in the next chapter. Given a security scenario for a given connection, the consumer sends its set of choices and their valuations, H_1, M_1 or L_1, illustrated in Equations 8-10, to each node. This way, each node knows which security algorithm has the most consumer valuation and will get the most payment $t_i(\theta)$ resulting to best incentive and reputation for that particular game. This is an attractive incentive for the node to choose the highest valued security algorithm from the SA_1's selection list and according to its own preference table.

The truthful participation is induced by the unique proposed design of DSIC-S protocol and the allocation rule, resulting from having different security scenarios and a set of allowable security algorithms for each. In this process, we also propose the calculation of $t_i(\theta)$ based on the consumer's valuations, which makes it possible to induce truthfulness and cheat-proofness as a solution to one of mechanism design's problem of misrepresentation of private values, θ_i, especially in a real-world scenario. This property of DSIC-S is proved in Chapter 7, Theorem 9.

Governed by this induced environment, the node will find it to its best response to participate truthfully and select a security algorithm for each service of the consumer's list, which maximizes its utility. It is naturally enforced to take the highest valued one from the intersection of her own preference table and the consumer's preferences for a particular scenario to gain the most utility. If she can not find this intersection, as described in Equation 6.16, her utility, allocation function and the payment will be zero, which means it would not be beneficial for her to participate. She will automatically decline to participate in the game.

6.3 The DSIC-S Incentive Model

After the SAs have made their selection of strategies and security algorithms, now the mechanism designer, SA_1, can calculate the payments for each node. For the proposed protocol, we have chosen

the VCG scheme for these calculations. This way, we can also benefit from the known properties of VCG payment rule, which in a quasi-linear environment provides a sufficient condition for an allocatively efficient Social Choice Function, SCF, to be dominant strategy incentive compatible [NGNP_09]. Chapter 7, Theorem 7, proves the allocative efficiency property of DSIC-S. Theorem 9 proves that DSIC-S protocol is dominant strategy incentive compatible and cheat-proof.

As explained above, for the payment to each node as an attractive incentive for its truthful participation, the valuations of the consumer's SAR table are used for the selected strategy $s_i = \theta_i$. These valuations are used in the calculations of the node's externality through the Clarke's pivotal payment rule [ClEd_71]. For DSIC-S this implies that

$$P_i(\theta) = (V_1|\ \theta_i) + \sum_{m\in N,\, m\neq i} (V_1|\ \theta_{-i} \,\&\, \theta_m) - \sum_{m\in N,\, m\neq i} (V_1|\ \theta \,\&\, \theta_m), \tag{6.12}$$

Note that, when one SA_i is not present in the path, consumer's requested total delay and resulting valuation can be divided among (n-1) nodes. This relaxes the selection process for others in favor of selfishness and its resulting resource consumption decrease, one of the main goals of each agent. In DSIC-S, it is considered that each agent could select the next lesser valued security algorithm for each service, given the additional available bandwidth, as long as it is still within the given scenario and the consumer requested total maximum valuation for the path is not violated. This means, for the above calculations and in the absence of one SA, all other's valuations can be relaxed and decreased by one. For the agents, which would select the lowest valuation of the scenario, with the valuation of 1, their valuation for the second expression would be zero. This relaxed environment is explained in an example in the following section.

As we know, Clarke's pivotal rule calculates the tax that a player should pay because of her influence on the whole game and other agents' decisions. Therefore, these payments will be negative and as a confirmation of Weak Budget Balance, one of the properties of Clarke mechanism, where $\sum_{i\in N}\ t_i(\theta)\ \leq 0$.

DSIC-S considers the reimbursement of this amount, P_i, calculated for each node as payment and credit to it. As proved in Theorem 1, these payments are independent of the valuations. To enforce the agents to always aim to choose the highest valued algorithm, however, the transfers should be dependent on consumer's valuations. This is achieved in our design by integrating our calculated weighted rankings, R_{AT} and R_{DT}, in Equations 5.3 and 5.4. This way, additional payments are guaranteed for the selection of the most desirable algorithm. This ensures that the actual payment can be formulated as follows,

(6.13)

$$t_i(\theta) = \begin{cases} R_{AT} - P_i(\theta) & \forall\ \theta_i \in H_1 \ \&\ \text{High-Security} \\ R_{DT} - P_i(\theta) & (\forall\ \theta_i \in M_1 \ \&\ \text{Medium-Security}) \text{ or } (\forall\ \theta_i \in L_1 \ \&\ \text{Low-Security}) \\ 0 & \text{Otherwise} \end{cases}$$

Through these payments made at each game, and as means of performance evaluation in the social strategic game, the network provider increases its trustworthiness and attractiveness to draw more customers and increase her revenue.

DSIC-S Incentive Payment – An Example

As an example, DSIC-S security association establishment is taken, as depicted in Figure 6.1, for two security services and a path comprising of 5 nodes. To simplify the calculations and better illustrate the main point, it is assumed that all nodes would maximize their utilities and select the security algorithms with the highest consumer valuation of 3 for each security service. Let's further assume that the consumer is inducing a high-security scenario and uses the SAR table as depicted in Table 5.2.

In this case, SA_1 will be establishing 8 security associations with other 4 nodes, one for each security service. This brings the total security operation cost for SA_1 to a total of 8.

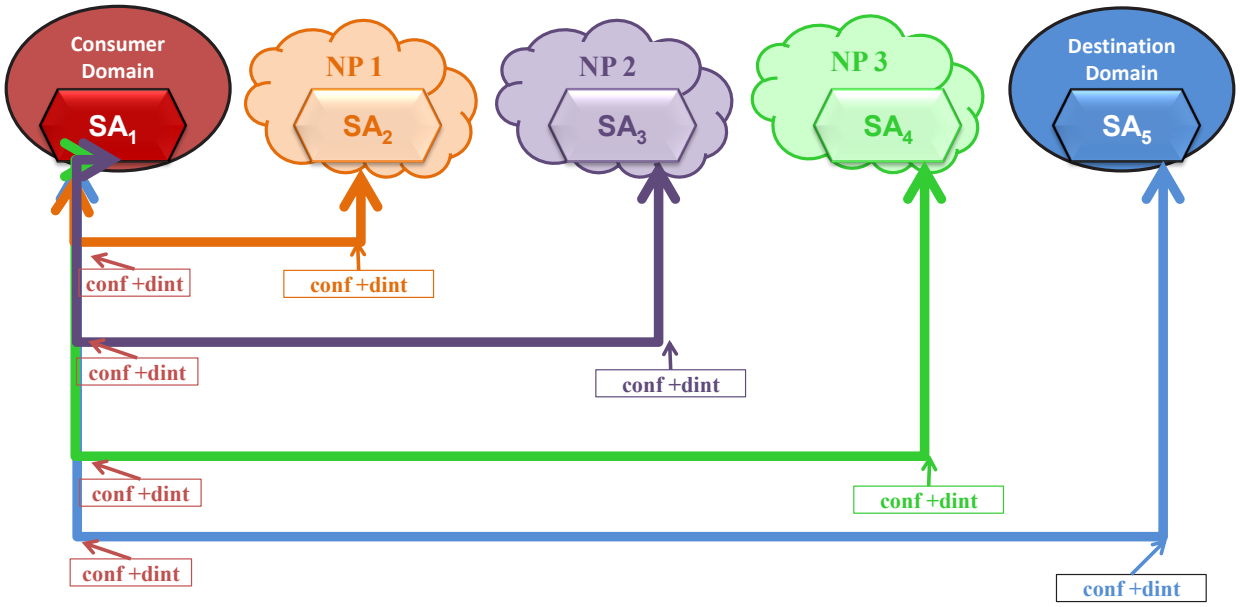


Figure 6.1 DSIC-S Incentive Payments in an Example

Taking our assumption of $V_{Ai} = 3$ into consideration, the total operational cost for SA_1 will be 8 * 3 = 24. Each of the other nodes, however, according to our association map in Figure 6.1, will have only 2 * 3 = 6 security operational cost, when all SA_i s are present in the network.

In the case, one SA is absent in the path, let's say SA_2, then SA_1 will be establishing 3 * 2 = 6 security associations only, resulting to operational cost of 6 * 3 = 18. Since SA_2 is not available anymore, there will be no operational cost for it. Basically, if one SA is absent, in our case, the total security operations will be relaxed by 4. According to DSIC-S now the other agents could select one lower valued security algorithm for each service, given that the total credit is not higher than the available calculated 18-6=12. In this case, by the absence of SA_2, SA_1's operational cost would be 6 * 2 = 12. Each of other nodes will have an operational cost of 2 * 2 = 4, for a total of 4 * 3 = 12 for the three remaining nodes other than SA_1 and SA_2. Now we can plug in the above calculated values in Equation 6.12 to get

$$P_i(\theta) = P_2(\theta) = 6 + (12 + 12) - (24 + 18) = -12 \qquad (6.14)$$

As described above in Equation 6.13, DSIC-S integrates this value in the calculation of the reimbursement amount for each node as payment and credit to it.

Chapter

7

DSIC-S Properties and Results

7.1 VCG Rule Properties in DSIC-S

The following Theorems, prove the VCG payment's independence of the actual node's valuations but dependence on the number of security associations and number of nodes in the security strategic game. These results prove the most important design aspects of DSIC-S, namely solving the revelation Theorem's intriguing problem of misrepresentation of agent's private information in mechanism design theory.

> **Theorem 1:** *In DSIC-S, an agent's VCG payment is a function of the number of security associations between each node and the consumer's node in the network.*

This Theorem states that as the number of security associations between each two nodes increases, the resulted payment reimbursements will be higher. This is a very preferred property for our incentive model to counteract selfishness of rational players. The agents get more incentives for more work and more resource consumption, i.e. the establishment of more security associations. We

first show this by use of the example in the previous section. The above example is now calculated for the case that only one security service would be required between the consumer and the nodes, i.e. only confidentiality. In this case, the SA_1 would need 4 security operations and each other SA only one. The VCG payment according to Equation 6.12 would be

$$P_i(\theta) = 3 + (6 + 6) - (12 + 9) = -6 \tag{7.1}$$

Proof: We now prove this in a general form. To simplify the proof, let's assume that all nodes select the security services with the same ranking, let's say V. Also, let's assume there are Np agents with each establishing n security associations with the consumer's SA_1. The three expressions of the VCG payment of Equation 6.12 can be written as

$$P_i(\theta) = P_1 + P_2 - P_3 \tag{7.2}$$

The first expression is the sum of all valuations for all the security associations, $n \in N$, a node needs to implement

$$P_1 = \sum_{n \in N} (V) = n * V \tag{7.3}$$

The second expression shows the externality of each node in the absence of the node in the network. DSIC-S also considers a relaxation of choice because of additional operational bandwidth, hence $(V - 1)$. In this case, if a node is not there, it will not have any security associations with SA_1, hence $(Np - 2)$. That means all others will have security associations with SA_1, namely each pair of SA_1 and SA_i will have $(2 * (n * (V - 1)))$. So P_2 is given by

$$P_2 = (Np - 2) * (2 * (n * (V - 1))) \tag{7.4}$$

$$P_2 = 2 * n * [(Np * V) - Np - (2 * V) - 2]$$

For the third expression, we consider the node as present in the network, but calculate the sum of the valuations of all other nodes, (Np ─ 2) * (2 * (n * V)), without that particular node, hence only (n * V) for its association with SA_1

$$P_3 = [(Np ─ 2) * (2 * (n * V))] + (n * V) \quad (7.5)$$

$$P_3 = n * V * [(2 * Np) ─ 3]$$

Now we can put all of the above together to calculate the VCG payment for DSIC-S, yielding

$$P_i(\theta) = (n * V) + (2 * n * [(Np * V) ─ Np ─ (2 * V) ─ 2])$$

$$─ (n * V * [(2 * Np) ─ 3])$$

$$P_i(\theta) = ─ 2 * n * (Np ─ 2) \quad (7.6)$$

As illustrated above and according to the very important result in Equation 7.6, we see that, in DSIC-S, the VCG payment is a function of n, the number of security associations pertaining to the required security associations between each node and the consumer. ■

Now we can prove the dependency of the incentives on the number of agents in this strategic security game.

Theorem 2: *In DSIC-S, an agent's VCG payment is a function of number of nodes in the network.*

This can be first shown by the use of the example in Chapter 6 with an added node. The calculated payment for a network with 6 nodes amounts to

$$P_i(\theta) = 6 + (16 + 16) ─ (30 + 24) = ─ 16 \quad (7.7)$$

Proof: The proof in Theorem 1 is used for this theorem as well. As we can see in Equation 7.6, the payment is a function of Np, the number of players in the security strategic game. This Theorem proves the intuitive assumption that as the number of players, here nodes, increases, each node has to pay taxes to more players for its effect of Clarke's externality. That means, more nodes are affected by its presence or absence. In Clarke's interpretation each node needs to pay a tax for the effect of its externality on each other node. The more nodes are in the game, the more taxes are paid. If this increase of gain with the number of players is used in a well thought design, as here in DSIC-S, it will give more incentives as a reimbursement for the effect of the increase of externalities. This is a very desired property of the designed mechanism to increase motivation of participation in the game. ■

The next Theorem is of a very great importance. It proves that DSIC-S design solves one of the most intriguing problems in design of mechanisms, namely the misrepresentation of agent's private information, the truthful revelation problem. The understanding is that whenever there is private information not known to all the players, it will be misrepresented by the agents to increase their gain and utility of participation. A strong mechanism needs to solve this problem. Theorem 3, proves DSIC-S protocol's solution and its independence of the private information θ.

Theorem 3: *In DSIC-S, an agent's VCG payment is independent of the type profile θ of all agents, the selected security algorithms and their corresponding valuations.*

Proof: The proof in Theorem 1 is once again used for this theorem as well. As we can see in Equation 7.6, the payment does not include V, the valuation of the selected security algorithms! This shows that the Clarke's Payment is independent of θ. ■

On a positive note, this property makes the payments cheat- and strategyproof. This means, as long as the nodes can participate and accommodate the asking security conditions, they will be having an incentive payment regardless of which algorithm they choose, hence strategyproof. Also, there is no need to lie about their true types, and try to misrepresent their private information, because it

does not go into the payment calculations! No truth revelation concerns are needed to be addressed here!

On a negative note, however, only using the VCG payments as the credit payments to the nodes in this case is not enough. The independence of **θ** does not enforce any incentive for the nodes to select a security algorithm with a higher valuation, the one most appealing to the consumer. In the formulation of $t_i(\theta)$, we solve this problem by incorporating the θ-dependent proposed rankings R_{DT} or R_{AT} in the calculation of incentives to the nodes.

In general, the independence of true types θ can be very helpful for the design of cheat- and strategyproof mechanisms. It implies that there would be no need for truth revelation of the private types to calculate the incentive payments. This is the solution for one of most demanding mechanism designs problems, namely the truth manipulation by players, especially in the real-world implementations. The players are rational and as soon as they notice there is a gain, they would misrepresent their true types. It is then up to the mechanism designer to design a novel mechanism to bypass this undesired by-product of selfishness. As we can see this has been achieved in the design of the proposed mechanism, DSIC-S.

7.2 Clarke's Externality of Agents in DSIC-S

Theorems 1 and 2 are also the proof for the very important next Theorem in a security protocol, the degree of externality of each security agent, according to the VCG payment rule as follows,

Theorem 4: *In DSIC-S, the degree of each Security agent's externality is a function of the number of security associations, number of security agents and nodes in the network.*

Proof: See Theorems 1 and 2 for proof. ■

7.3 Agent's Utility Function Properties in DSIC-S

Theorem 5: *In DSIC-S, an agent's utility is a function of the number of security associations between each node and the consumer's node in the network.*

Proof: The proof is based on Theorem 1. As we can see in Equation 7.6, the payment is a function of n, the number of security associations pertaining to the required security services between each node and the consumer.

According to Equation 6.13, $t_i(\theta)$ is a function of $P_i(\theta)$ and subsequently a function of n. This way, according to Equation 6.11, the utility of each agent will be a function of n, the number of security associations, as well.■

This Theorem is very favorable for the proposed incentive model. The more resources the nodes consume, i.e. the more security associations they establish, the more increase they see in their utility payments.

Theorem 6: *In DSIC-S, an agent's utility is a function of the number of nodes in the network.*

Proof: The proof is based on Theorem 2. As we can see in Equation 7.6, the VCG payment $P_i(\theta)$ is a function of Np, the number of nodes in the game. According to Equation 6.13, $t_i(\theta)$ is a function of $P_i(\theta)$ and subsequently a function of Np. According to Equation 6.11 the utility of each agent is calculated as $u_i(\theta) = v_i(k(\theta), \theta_i) + t_i(\theta)$. With the increase of $t_i(\theta)$ as shown above the utility of each agent increases.■

7.4 DSIC-S is Allocatively Efficient

Theorem 7: *DSIC-S is Allocatively Efficient.*

According to the definitions of mechanism design [MyRb_97][WeJn_07][NGNP_09], an allocation function $k(\theta)$ is allocatively efficient if for $\forall\ \theta \in \Theta$, it maximizes the sum of their values $\sum_{i \in N} v_i(k(\theta), \theta_i)$. In other words, it is such defined, that the highest willingness will receive the allocation.

Proof: In DSIC-S, by choosing and defining the allocation function as in Equation 6.7, it naturally maximizes the valuations for each player. Through this rule and the way it is defined, each player is able to participate according to her value of θ_i for the given scenario, the only condition, that maximizes its utility, because otherwise she would have a utility of zero. Thus, it is able to maximize her valuation for that scenario logically by selecting the security algorithm with highest valuation from her SAR table, which is in the list of consumer's requested options for the governing scenario. Since for any given scenario, each player maximizes her valuation naturally, the defined allocation function $k(\theta)$ becomes allocatively efficient for $\forall\ \theta \in \Theta$. ■

7.5 DSIC-S is Individually Rational

Theorem 8: *DSIC-S is Interim Individually Rational*

Individual Rationality is referred to voluntary participation property[NGNP_09]. It means that, each player gains a non-negative utility by participating in the mechanism. "Interim Individual Rationality" implies that the agents can withdraw from the game after knowing their own private types but before responding to the mechanism with their selection of action[NGNP_09]. This implies that in order for the agent to decide to participate in the game, she has to make sure that her calculated utility function is larger than or at least the same as her utility function if not playing.

Proof: Each agent in DSIC-S knows her private type at the beginning of the game before the mechanism starts. Each agent can also calculate her utility function prior to her response according to Equation 6.11. As we now know, through the allocation rules and incentive model discussed in above sections, DSIC-S induces a positive utility naturally, when the player plays. According to Equations 6.7 and 6.13, by not playing, the player has a utility of

zero. This implies that the SA in the path gains a nonnegative utility and is better off playing in the game, hence DSIC-S is Interim Individually Rational. ■

7.6 DSIC-S is Dominant Strategy and Cheat-Proof

According to the definitions of mechanism design [NGNP_09], a Social Choice Function $f: \prod_{i \in N} \Theta_i \rightarrow X$ is said to be Dominant Strategy Incentive Compatible, DSIC, or truthfully implementable in dominant strategy, if the direct mechanism has a weakly dominant strategy equilibrium $s^*(.) = (s^*_1(.), s^*_2(.), \ldots, s^*_n(.))$, where $s^*_i(\theta_i) = \theta_i$, for $\forall\, \theta_i \in \Theta_i$ and $\forall i \in N$. This property is also called cheat- or strategy-proof.

Theorem 9: *DSIC-S Protocol is cheat- and Strategy-proof.*

Proof: As shown above, through DSIC-S incentive model, each agent's optimal response is to participate truthfully to be able to gain a positive utility for a given scenario, regardless of what other agents announce. According to the allocation rule of Equation 6.7, if the agents do not respond with a strategy within the allowable and consumer requested choices, they will get no allocation of the outcome, hence no positive utility. Furthermore, DSIC-S is designed to allocate a positive payment, as shown in Equation 6.13, to each node that adheres to the allocation rule and participates truthfully, regardless of the responses of all other nodes in the game. In addition, the incentive model described in Equation 6.13 is designed so that each node, in order to maximize its utility, finds it to its best interest to choose the most valued security algorithm from the consumer selection, again regardless of other agents' choices. Furthermore, through the proposed unique use of the consumer's valuation table for calculation of these incentives, all agent's can make their decisions regardless of other players and according to SAR table of the consumer. This achieves a strong dominant strategy equilibrium among the nodes with a profile of their equilibrium strategies of $s^*(.) = (s^*_1(.), s^*_2(.), \ldots, s^*_n(.))$, as proved above, we have

$$u_{\theta i}(s^*_i(\theta_i), s_{-i}(\theta_{-i})) > u_{\theta i}(s_i, s_{-i}(\theta_{-i})) \qquad (7.8)$$

$$\forall s_i \in S_i \backslash \{s^*_{-i}(\theta_i)\},\ \forall s_{-i}(\theta_{-i}) \in S_{-i},\ \forall i \in N,\ \forall \theta_i \in \Theta_i,\ \forall \theta_{-i} \in \Theta_{-i}$$

Since according to mechanism design concepts, a strongly dominant strategy equilibrium is automatically a weakly dominant strategy equilibrium, We can always find a weakly dominant equilibrium, which makes DSIC-S strategy-proof and cheat-proof. ■

7.7 DSIC-S is Pareto Optimal and Socially Desirable

The allocation rules, scenario-based structure and incentive model of DSIC-S, are specifically designed to only provide allocative efficiency and individual rationality for the agents, which have the consumer requested security algorithms, or at least one of them, for the specific scenario also in their SAR Table. As we can recall, it is the consumer that determines the level of security. This means, any agent's SAR table according to its preferences, needs to have one non-empty intersection with the consumer's choices for a particular security scenario. That is for $\forall\ i \in N$

(7.9)

$$\begin{cases} H_1 \cap (H_i \cup M_i \cup L_i) : \neq \emptyset & \text{for at least one } \theta_i \in \Theta_i \ \& \text{ high security} \\ M_1 \cap (H_i \cup M_i \cup L_i) : \neq \emptyset & \text{for at least one } \theta_i \in \Theta_i \ \& \text{ medium security} \\ L_1 \cap (H_i \cup M_i \cup L_i) : \neq \emptyset & \text{for at least one } \theta_i \in \Theta_i \ \& \text{ low security} \end{cases}$$

This implies that only agents with at least one of these algorithms in their preference list can be chosen to be a player in the game. For the actual game and credit payments the valuation of the preference of other players other than the consumer are not of importance, as long as the security algorithms are in one of their preference lists. Furthermore, the scenario-based design of DSIC-S requires different security algorithm preferences for different levels of security. The commonality of use, standards acceptance, along with the cryptanalysis proof of fitness for a particular security scenario, also enforces each agent favorably toward having at least one match in her preference list. According to Equation 7.9 and the above assumptions, *Pareto optimality* for DSIC-S can now be defined as

Definition: *An agent's strategy, $s_i = \theta_i$, in DSIC-S is said to be Pareto optimal if $k(\theta) \neq 0$. That means, $\theta_i \in (H_i \cup M_i \cup L_i)$, and $\theta_i \in H_1$ or M_1 or L_1, depending on the governing security scenario .*

That is, if an agent is willing to and by the mechanism's allocation rule is selected to participate, then inherently her selected strategy will always be a Pareto optimal strategy for her. This is according to the fact that our mechanism is allocatively efficient. In general in Mechanism Design Theory, Pareto optimality in an evironment *e* is defined as an allocation, which has to be feasible in that environment and then that each player is at least well off and one is at least better off [GRR_83]. "Formally, x^* is Pareto-efficient in *e* if and only if (i) x^* is feasible for *e* and (ii) if *x* is feasible for *e*, then $u_i(x_i) < u_i(x_i^*)$ for at least one *i*." [GRR_83]. This optimality in our case solves a multi-criteria objective function with regards to security and delay-efficiency through the proposed valuation system, as defined in the type agent θ_i. Our game takes place at the connection establishment phase as a one-stage strategic form game, the simplest class of games, which could be general enough to describe all the complicated games in game theory as Meyerson describes in his Nobel Prize Autobigrophy [MyRb_12]. He also mentions, John von Neumann, notoriously considered as the father of game theory, introduces the notion of *strategic normalization*, where for any dynamic extensive game, one can define an equivalent *strategic form game,* where the players choose strategies independently and whose strategy choices determine their expected payoffs.

Theorem 10: *In DSIC-S all agents' selected strategies are socially desirable and Pareto optimal.*

Proof A: We prove the *Pareto optimality* by contradiction for a given scenario. Let's assume it is a high security scenario and agent i chooses a strategy, $s_i = \theta_i$, that is not Pareto optimal for her but belongs to H_1, this implies that

$$\theta_i \notin M_i \;\&\; \theta_i \notin L_i \;\&\; \theta_i \notin H_i \;\&\; \theta_i \in H_1 \tag{7.10}$$

$$\theta_i \notin (H_i \cup M_i \cup L_i)$$

$$\theta_i \notin \{ H_1 \cap (H_i \cup M_i \cup L_i)\}$$

$$H_1 \cap (H_i \cup M_i \cup L_i) = \emptyset$$

This leads to a contradiction to Equations 7.9 and 7.10, which results to $k(\theta) = 0$. Consequently agent i cannot participate in the game!■

This proves that if an agent is participating in the game, her selected strategy will be Pareto optimal! In case of social desirability and in context of DSIC-S, we define an agent's strategy to be socially desirable, if it can provide the security and delay conditions requested for a particular connection by the consumer. The social planner knows what end-to-end conditions needs to be established and determines the choices of security algorithms best fitting to a particular security and delay scenario in a fair manner for all participants. That is

Definition: *An agent's strategy, $s_i = \theta_i$, in DSIC-S is said to be socially desirable if $\theta_i \in H_1$ or M_1 or L_1, depending on the governing security scenario and $\theta_i \in H_i$ or M_i or L_i.*

This means that $s_i = \theta_i$ needs to belong to the agent's preferred list of security mechanisms, the SAR Table, so that it naturally chooses to implement it.

By the same token of proof A, since through DSIC-S allocation rule only agents will be participating, who are able to select one of the consumer's choices pertaining to Equation 7.9, any selected strategy by any participating node will be socially desirable:

Proof B: Again, we prove *social desirability* by contradiction. Let us assume it is a high security scenario and agent i chooses a strategy, $s_i = \theta_i$, that is not socially desirable. According to the above definition this implies that

$$\theta_i \notin H_1 \,\&\, \theta_i \notin M_i \,\&\, \theta_i \notin L_i \,\&\, \theta_i \notin H_i \qquad (7.11)$$

and from here we can follow the proof above in ProofA. This proves that if an agent is participating in the game, her selected strategy has to be socially desirable! ■

7.8 In DSIC-S SAR Table's elements form the Agent's Pareto Frontier

Theorem 11: *In DSIC-S, the elements of each agent's SAR table form that agent's Pareto Frontier. That is, $(H_i \cup M_i \cup L_i)$ is the agent i's Pareto Set.*

Proof: See above Proof A of Theorem 10 along with the definition of Pareto optimality.■

Chapter

8

DSIC-S Simulations

In the following, we will demonstrate the achieved results in simulation cases interesting in the context of mechanism design theory and its implementation in designing delay efficient network security protocols. In the simulations, we continue to use the assumptions for above examples unless otherwise mentioned. For the analysis, we compare results for multi-agent games of different number players, Np = { 5, 10, 15, 20, 25, 30, 35, 40, 45, 50}. we have used the example consumer SAD and SAR tables in Tables 5.1 and 5.2, which builds upon the performance results in [NaJm_05] and [EAH_10]. For different nodes along the path, we have then adapted delays in range of ±10% of these values randomly.

8.1 Theorems 2 and 3 Results

In order to analyze Equation 7.6, namely the independence of the resulted payments from the selected valuation, we simulated P_{imax}, where all agents would choose the maximum valuation in order to maximize their utility. We also calculated P_{imin}, for the case that they would choose the

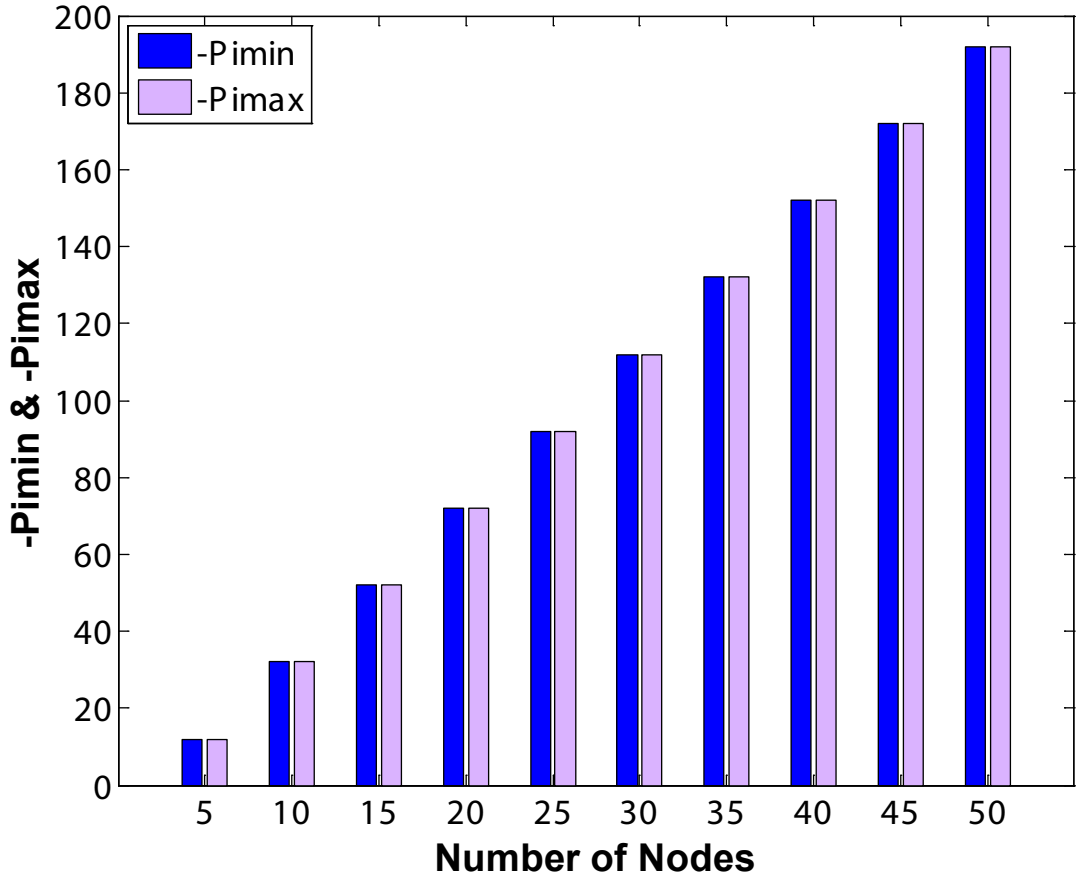


Figure 8.1 Theorems 2 and 3 Results in Medium Security DSIC-S Games

minimum valuation as their strategy. Figure 8.1 compares the results of the simulation for the VCG payment calculations in both cases. As we can see, the simulation results clearly prove Equation 7.6) to be right. They also confirm the statements of Theorem 2, that the payments are a function of the number of nodes and agents in the network. As we can further see, these payments are not a function of selected valuation, which is the confirmation of Theorem 3, $P_{imin} = P_{imax}$.

8.2 Delay and t_i Transfers General Correlation

This research work is proposing a delay-efficient protocol designed by tools of mechanism design theory to be incentive compatible and individually rational. These important properties are achieved through the unique design of DSIC-S protocol, namely the independence of the incentives and outcome from announcement of any agent's private information. We propose a scenario-based

security protocol based on the agent's individual preferences.We also propose the use of a valuation system to calculate the t_i transfers as incentives, which does not require the actual delays to be announced.

In this simulation, through several scenarios, we analyze our proposed design to see if this model truly correlates with the actual delays. It has to also confirm the hypothesis that this correlation has to be reciprocal. Figure 8.2 illustrates the results for the t_i transfers, calculated according to Equation 6.13, and shows the t_i transfer space within the minimum and maximum boundaries. For the calculation of the t_{imin} values, it is assumed that all agents in the game would select the lowest

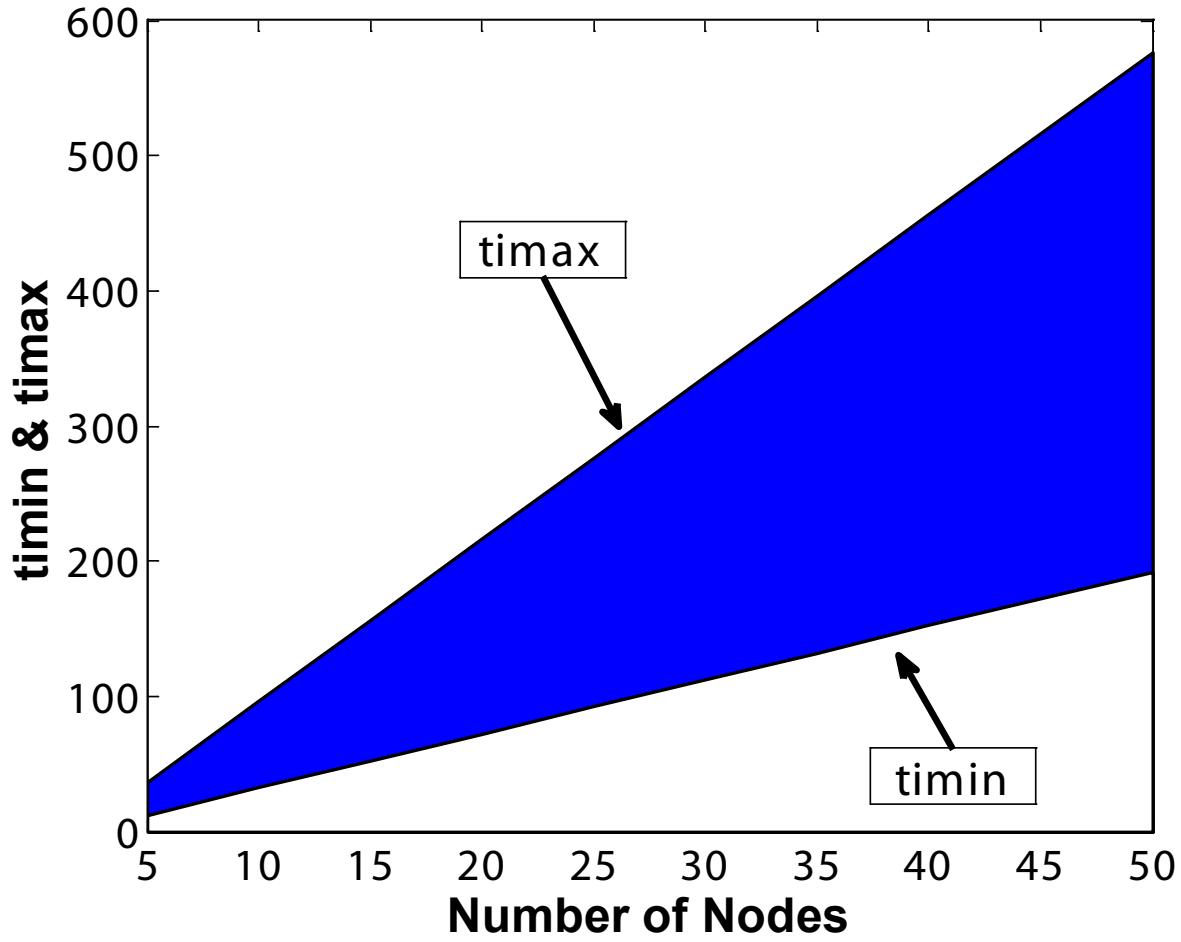


Figure 8.2 t_{imin} & t_{imax} Transfers in a Medium Security DSIC-S Game

valuations. The t_{imax} values are calculated by the assumption that all agents would maximize their utilities by choosing the highest valuations.

Figure 8.3, however, illustrates the actual delays. D_{imax} is the summation of the longest delays at each node for the number of agents in the game. D_{imin} in the same manner shows the lowest delays. The area in between is the permitted delay space of the agents. As we can see, the two graphs in Figures 8.2 and 8.3 follow the same shape and dependence on number of agents. This simulation confirms the general correlation with no granularity. The manner of correlation property is not recognizable yet. Section 8.4 investigates this in another simulation case.

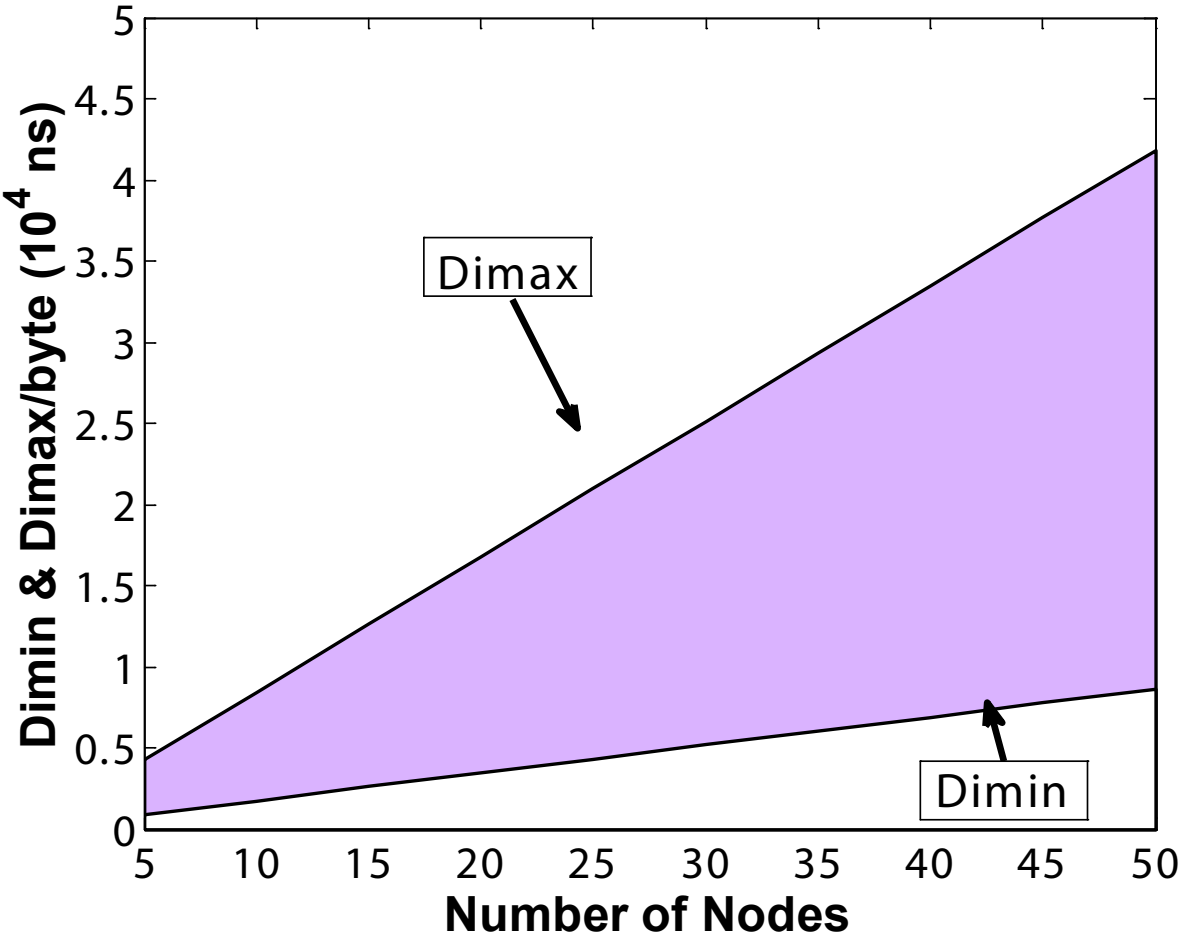


Figure 8.3 D_{imin} & D_{imax} in a Medium Security DSIC-S Game

8.3 Normalized Delay and t_i Transfers

In another simulation case, we investigate this correlation in a more precise manner for better illustration. We compare the incentive distribution through normalized t_{imax}, as depicted in Figure 8.4, which is the ratio of t_{imax} over total t_{imax} of the number of players in each game, that is

$$\text{Normalized } t_{imax} = \frac{t_{imax}}{\sum_{i \in Np} t_{imax}} \tag{8.1}$$

with the delay distribution through normalized D_{imax} , as depicted in Figure 8.5, according to the following:

$$\text{Normalized } D_{imax} = \frac{D_{imax}}{\sum_{i \in Np} D_{imax}} \tag{8.2}$$

8.4 Incentive Compatibility

The manner of correlation is finally clear in another simulation scenario of a 5-Agent DSIC-S Game, illustrated in Figure 8.6. Here, the results confirm the hypothesis and the achievement of the main goal, the reciprocal correlation. They confirm that the protocol gives more incentives to agents producing less delay. As the delay decreases from node to node, see Node 3 and 4, the incentive transfer increases. That is, the design accomplishes one of the most important goals, namely delay efficiency through its ti Transfer and incentive compatibility.

The simulation results furthermore confirm another purposed correlation between t_i and V_D of the selected strategy and security algorithms, as depicted in Figure 8.7. In this scenario, all nodes are choosing the same algorithm for Data Integrity and have chosen different strategies for their confidentiality security service in a 5-Agent DSIC-S game. The result clearly shows more incentives are given to the nodes, that choose algorithms with higher valuations. This way, through the selfishness and apt to maximize their utility, they will always try to select an algorithm, that is the

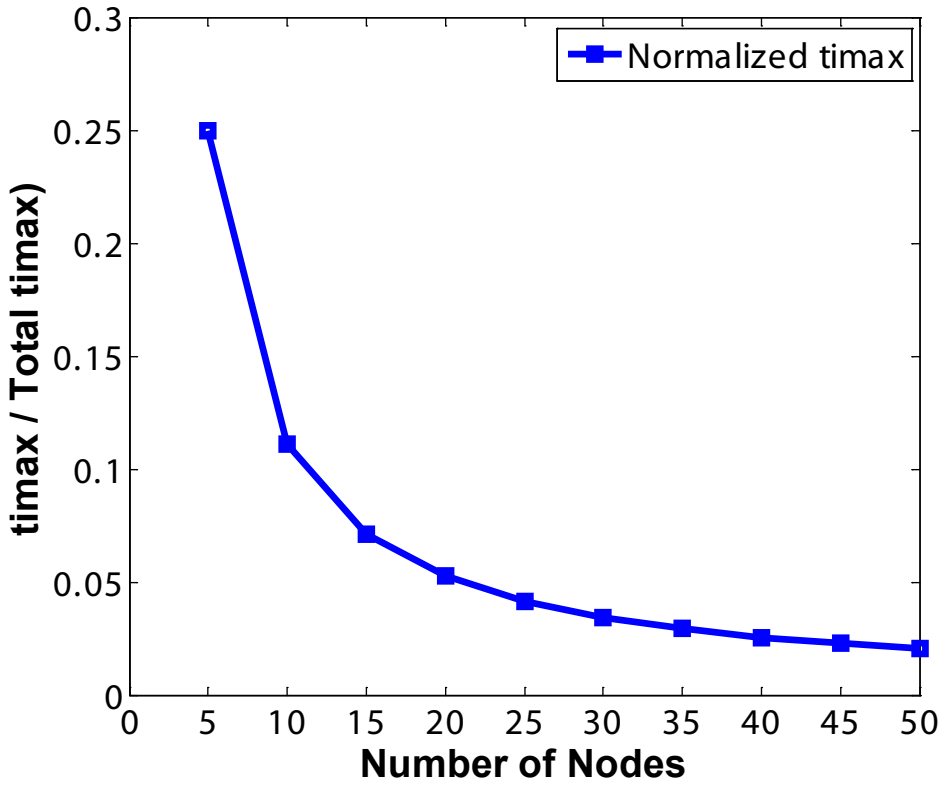


Figure 8.4 Normalized t_{imax} in a Medium Security DSIC-S Game

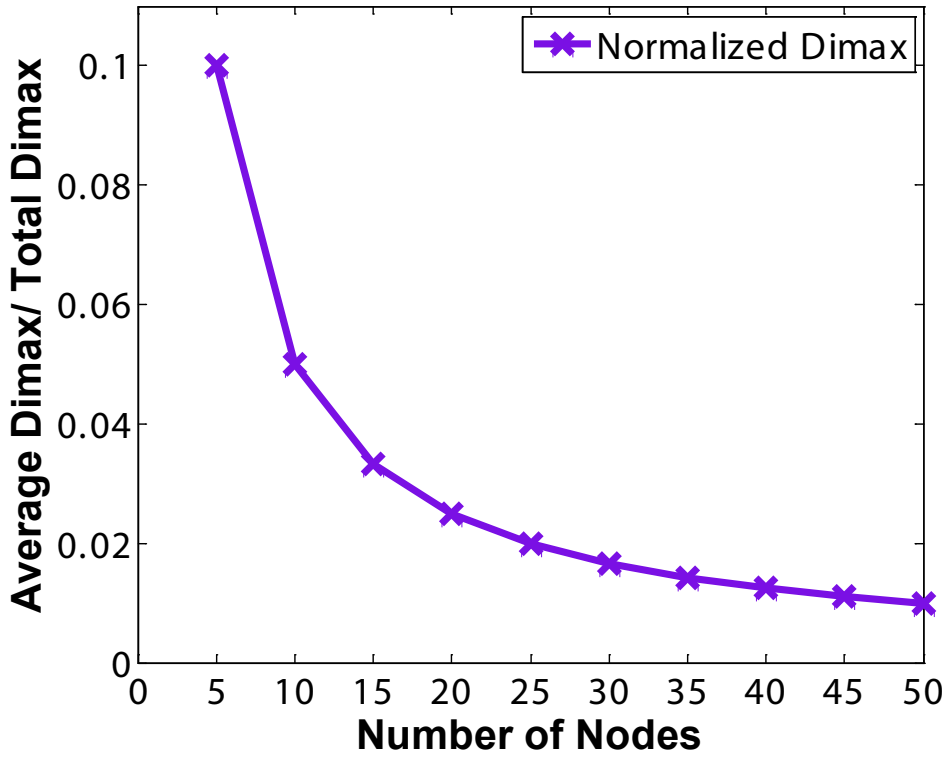


Figure 8.5 Normalized D_{imax} in a Medium Security DSIC-S Game

highest desired by the social planner. Here we can see in case of Nodes 4 and 5, they both choose the highest valuation, which has translated to the same incentive.

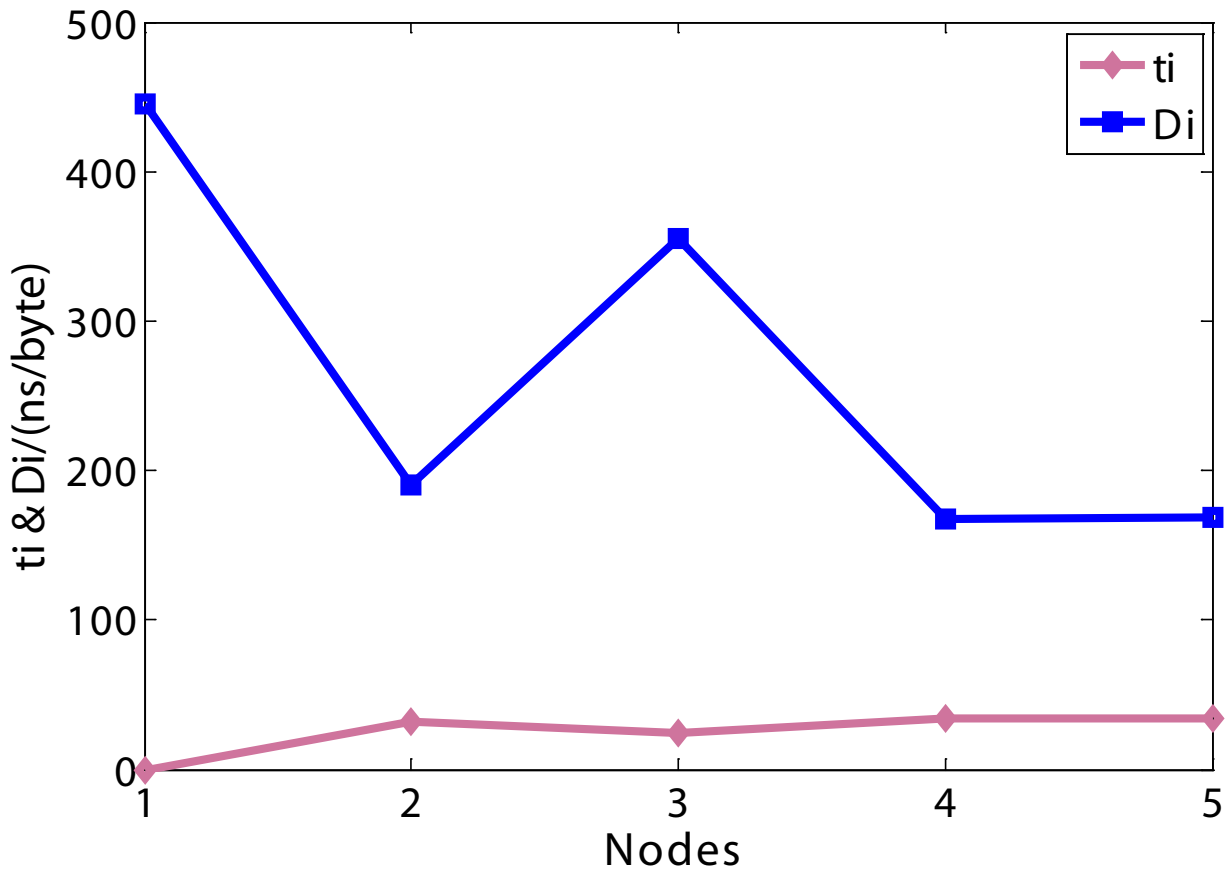


Figure 8.6 D_i Delay Vs. t_i Transfers in a DSIC-S 5-Agent Game

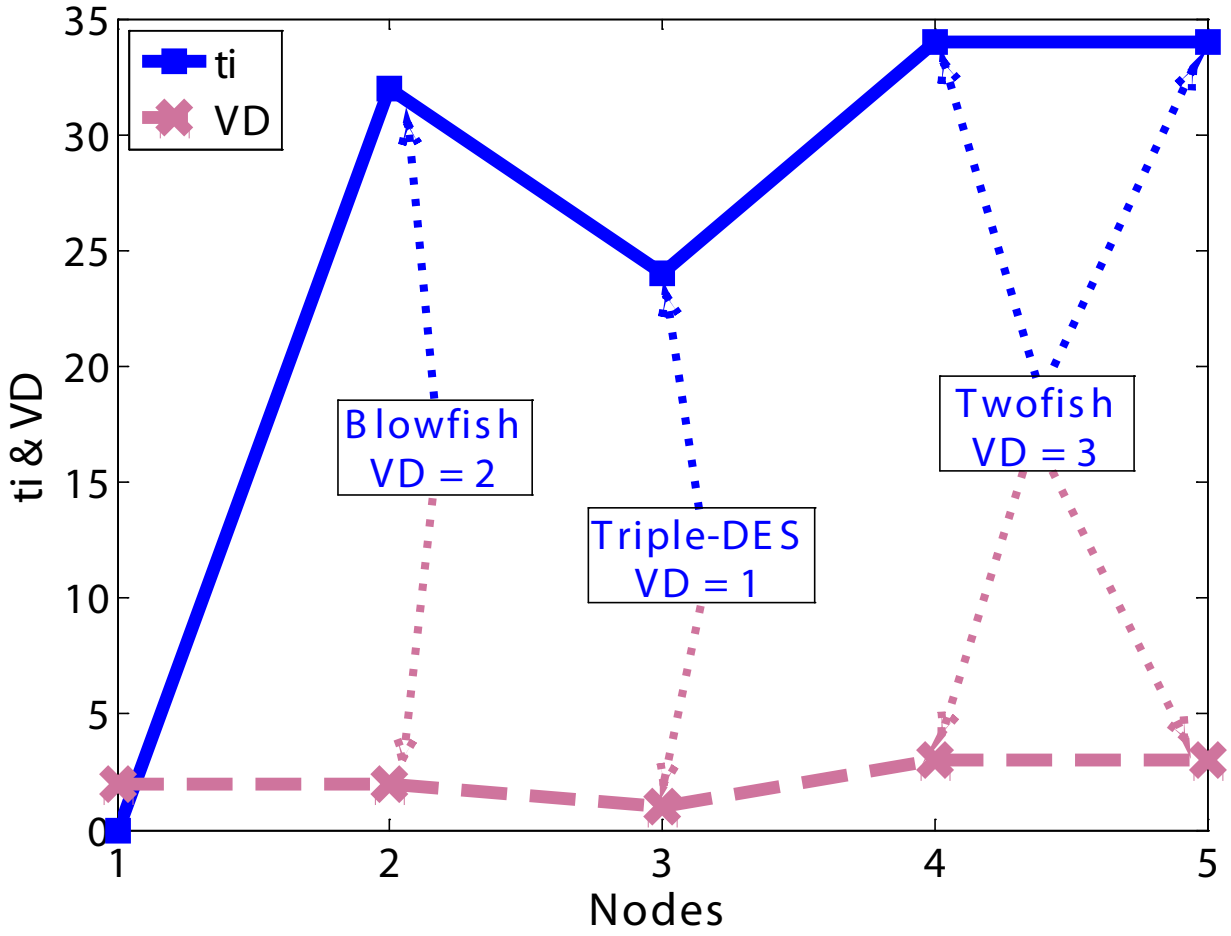


Figure 8.7 t_i **Transfers vs. Valuations in a DSIC-S 5-Agent Game**

Chapter

9

The IPsec-O Protocol

The proposed IPsec-O Protocol has been documented in a paper [SfMi_112] and ready to be submitted for publication.

In the previous chapters, DSIC-S protocol has been introduced as a cross-layer optimum security protocol. In this chapter, DSIC-S is implemented to design a layer-specific protocol for the network layer. In specific, it is implemented for the IP paradigm.

The main contributions of IPsec-O are as follows. Mechanism design theory is implemented for the design of a consumer-centric network security protocol, IPsec-O , as an extension to the current standardized IPsec protocol [RFC_4301] to provide security-delay tradeoffs. IPsec-O induces a cheat-proof consumer-centric strategic game along the connection path. The goal is that the underlying nodes, i.e. ISPs, though influenced by their own self-interest, to naturally find it to their best interest to behave in a network-wide accepted, consumer requested and to-her-promised way. We solve the important problem of misrepresentation of agents' private information in mechanism design and revelation theories for a delay-efficient IP security protocol through a distinctive design. We incorporate a valuation system to integrate the caused delay at each node in selection of security algorithms without consumer's knowledge of any agents' actual delays. Our incentive model

uniquely uses our proposed consumer's preference valuation system based on different security levels as an input for the VCG payment scheme with Clarke's pivotal rule. The credit transfers use this input as well.

IPsec-O achieves end-to-end and individual Pareto optimality, as proved in Theorem 10. This is an extremely desirable feature, which enforces natural adherence of the agents to the mechanism rules. IPsec-O is consumer-centric. The consumer is the social planner of the mechanism to request the desired security and delay level for her data transmission. This is one of the main goals of this research work and a very sought-after proposition. This enforces a natural and automatic control of the behavior of the underlying participants. The consumer can rest assured that her services are provided as expected and promised to her. IPsec-O is cheat-proof and strategy-proof, as shown in Theorem 9. This is a strong property for real life applications, which enables the nodes, i.e. ISPs, to naturally act responsibly and truthfully. Dominant strategy makes each node's best response independent of other nodes' choices of decisions and their belief functions. IPsec-O is scenario based. It considers different levels of security and the resulting tradeoff manifestation. This focal design enforces Pareto optimality, individual rationality and allocative efficiency, as shown in Theorems 10, 8 and 7, respectively.

In the following sections the *Guaranteed Service* protocol [RFC_2212] of the *Integrated Services (IntServ)* architecture [RFC_1633] is researched, which provides guaranteed end-to-end Traffic QoS and therefore is sensitive to the degradations caused by security operations. The Traffic QoS parameters defined in the existing protocols are addressed. IP security architecture [RFC_4301] is introduced. And finally, IPsec-O solution is proposed to incorporate the Traffic QoS degradations caused by security operations using and enhancing the existing IPsec [RFC_4301] and IKEv2 [RFC_5996] protocols.

9.1 QoS in IP Networks

Initially, Internet Protocol was developed with no extensive Quality of Service considerations, as defined in [RFC_791]: "The internet protocol can capitalize on the services of its supporting

networks to provide various types and qualities of service." . The best-effort service is the default service offered by the Internet. The network attempts to deliver the packets in the best possible way. There are, however, no guarantees on the delays and packet losses during the transmission. It is presumed that a very high percentage of packets will be delivered successfully to their destinations. It is further assumed that the transit delay experienced by a very high percentage of packets will not extensively exceed the minimum delay experienced by the successfully delivered packets.

With the emergence of new technologies and use of the Internet for real-time voice and video applications new requirements were imposed to the Quality of Service control over the Internet. Multicasting and capabilities to control the sharing of bandwidth among different traffic classes and to make the unused bandwidth available the rest of the time imposed furthermore upon the need for an extension to the existing unicast (point-to-point) best-effort service [RFC_1633]. The Integrated Services (IntServ) architecture was developed and defined in [RFC_1633] to address these new requirements. IntServ is designed to keep per flow states in the network nodes along the paths. It is an end-to-end protocol offering two different services: "*Guaranteed QoS*" [RFC_2212] and "*Controlled-Load Services*" [RFC_2211]. The following sections introduce guaranteed QoS architectures in more detail.

9.1.1 Integrated Services

As described above, the emergence of new real-time technologies and their implementation over the Internet, introduced the need for more distinct control of the Quality of Service in the Internet Protocol. New architectural extensions were needed to the existing best-effort service to support real-time Traffic QoS and provide control over the end-to-end packet delays [RFC_1633]. The Integrated Services (IntServ) Architecture was developed and introduced in 1994.

IntServ is predominantly concerned with the time-of-delivery of packets. Basically, the only quantitative calculations and service commitments made are the minimum and maximum bounds on delays [RFC_1633].

The protocol classifies applications, according to their timing requirements, into two classes: the elastic and the real-time applications.

Elastic applications are insensitive to introduced delays and delay variations during transmission. The architecture proposes several best-effort classes of service for these applications.

Within the real-time category, the architecture is concerned with the so-called "*Playback*" applications. A datastream is packetized at the source host and sent to the network. The network nodes introduce some variations of delay caused by their queuing behaviors and packet processing. The receiver depacketizes the data and attempts to reassemble and playback with the original timing. In order for this to happen, the network needs some method of time synchronization between the source and destination. The receiver also needs to buffer the receiving data stream and delay the reassembly by a calculated *offset* from the original departure time. The data arriving past this offset time is discarded.

IntServ assumes two classes of real-time applications: the tolerant, which require the knowledge of delay bounds and the intolerant, which require precise playback by using a fixed offset delay. The delay bounds can be either calculated and delivered by the network or predicted and modified from observations of the delays experienced by the previous packets. The intolerant applications should set their offset greater than the maximum delay bound.

Integrated Services Architecture proposes two services to control the Traffic QoS through out the network. For the tolerant application a *Predictive Service* or *Controlled-Load Service* is designed to ensure a fairly but not absolute reliable delay bound. The intolerant applications, however, can take advantage of the *Guaranteed Service* model, which commits to an absolute reliable upper bound on delay. The next section describe *Guaranteed Service* model.

Guaranteed Service

"Guaranteed service provides firm (mathematically provable) bounds on end-to-end datagram queuing delays." [RFC_2212]. It ensures that datagrams arrive within the guaranteed delivery time and there is no packet loss due to queue overflows, provided, the traffic stays within its agreed upon parameters. Guaranteed service is intended for applications with restrict real-time requirements, which cannot tolerate arrival of packets after a certain time. The model does not control the minimal or average delays but the maximum queuing delay.

In this model two network element specific error bounds of C_{tot} and D_{tot} are defined as follows:

C_{tot} is the cumulative total value of error terms C of each network element along the data path. C is the rate-dependent error term. It represents delays caused by rate dependent parameters of the flow. It is a maximum value and is measured in units of bytes.

D_{tot} is the cumulative total value of error terms D of each network element along the data path. D is the rate-independent per-element error term. It represents the worst-case (maximum) delays caused by non-rate-dependent transit time variations through each network element. It is measured in units of one microseconds.

The latter is of interest for the IPsec-O protocol. This parameter is updated for the consideration of network element specific security operational delays.

9.2 IP Security Architecture (IPsec)

IPsec offers security services at the IP layer for IPv4 and IPv6. It provides the capability to select required security protocols, determine algorithms to use for the services and use cryptographic keys required to provide these services [RFC_4301].

IPsec can be implemented on a host or another active network element (to make a *security gateway*) to provide protection to the IP traffic. The corresponding Security Associations (SA) or SA bundles are derived for this particular packet and the security service(s) are implemented. Each packet passing through the IPsec is either provided security services, denied passage and discarded or allowed to bypass through the IPsec, based on the security policy entries in the databases. The security services provided include access control, connectionless integrity, Anti-replay integrity, data origin authentication, confidentiality and limited traffic flow confidentiality. These services are supported by two defined security protocols: the *Authentication Header (*AH*)* and the *Encapsulating Security Payload (*ESP*)*. For each of these protocols at least one Security Association is required, which operates in a tunnel –between a host and a security gateway or two security gateways – or transport mode – between two hosts.

The next sections introduce the security associations and protocols along with the required security policy and association databases in more details.

9.2.1 IP Security Associations (SAs)

Security services are provided by Security Associations, SAs, which are uni-directional connections between two network nodes. Two security protocols: the Authentication Header [RFC_4302], AH, and the Encapsulating Security Payload [RFC_4303], ESP, are defined to offer these services. Each security association is uniquely identifiable by a security protocol index, destination address and the security protocol, AH or ESP. One SA is needed per security protocol. If both protocols are implemented simultaneously two SAs should be created. In this case a combination of SAs or *SA bundle* is required. The SAs, which are created between two hosts, are defined to be in *Transport Mode*. A *Tunnel Mode* SA is a security association applied to an IP tunnel. In this case, one end of the SA is a security gateway. The following sections discuss these modes and the term SA Bundle further. The IPsec-O protocol's security associations are based on this SA bundle model and in general are referred to as nesting security associations.

Tunnel Mode

A tunnel mode SA is a security association applied to an IP tunnel. It could be a SA between either a security gateway and a host or two gateways. Figure 9.1 illustrates these cases. The tunnel mode could also exist between two hosts. There is an outer IP header that specifies the destination for the IPsec processing and an inner IP header with the ultimate destination for the packet.

Transport Mode

A transport mode SA is a security association between two hosts. Figure 9.2 illustrates a security association between two hosts. For transport mode the security protocol header would appear after the IP header and before any optional extensions headers and any upper layer protocols.

SA Bundles

At times a security policy may require protection of traffic flow through a combination of security protocols and thus security associations. These SA combinations are called *SA bundles*. SAs may be combined into bundles in two ways: transport adjacency and iterated tunneling.

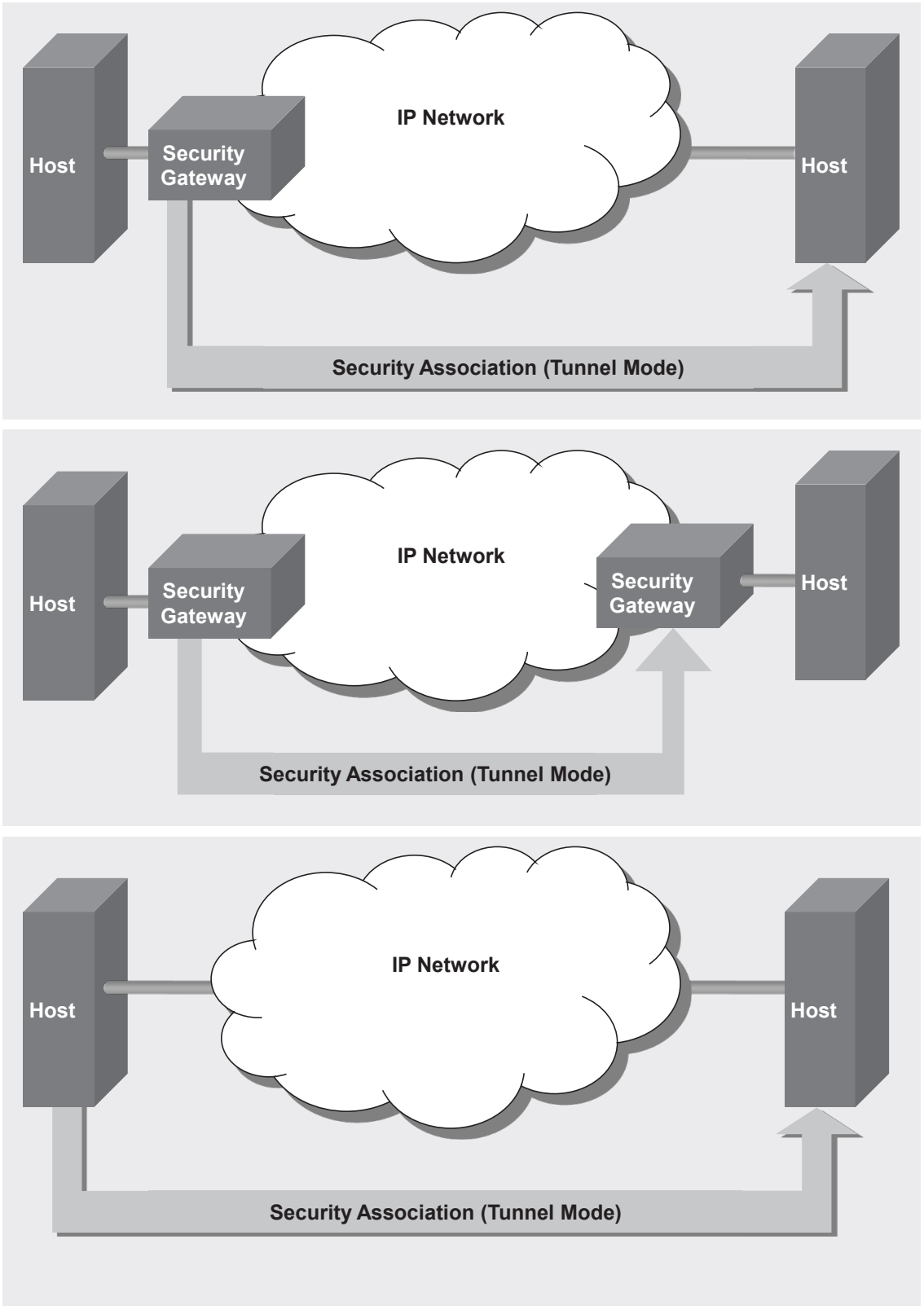


Figure 9.1 The IP Tunnel Mode Security Associations

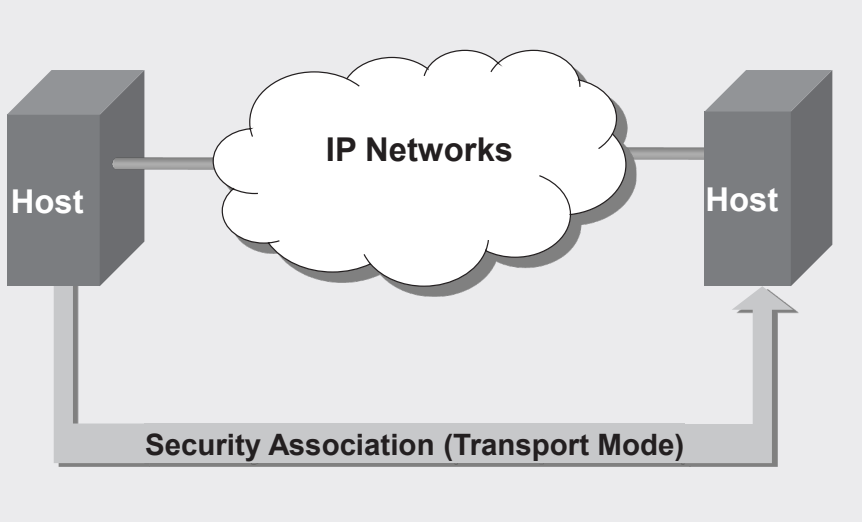


Figure 9.2 The IP Transport Mode Security Association

Transport adjacency is the term used for applying more than one security protocol to the same datagram without using tunneling. This way AH and ESP are used together and offer one level of combination. Figure 9.3 illustrates this case.

Iterated tunneling applies multiple levels of security protocols by establishing different tunnels. There are three cases of iterated tunneling from which the latter two cases are to be supported:

- Case 1: Both ends of the SAs are the same. The inner or outer tunnels could be either security protocols. Figure 9.4 illustrates this case.
- Case 2: One end of the SAs is the same. The inner or outer tunnels could be either security protocols. Figure 9.5 illustrates this case.
- Case 3: Neither ends of the SAs is the same. The inner or outer tunnels could be either security protocols. Figure 9.6 illustrates this case.

9.2.2 IPsec Protocols

IPsec implements two security protocols, the *Authentication Header* [RFC_4302], AH, and the *Encapsulating Security Payload* [RFC_4303], ESP. These protocols can be applied separately or in combination with each other. The Authentication Header offers data origin Authentication,

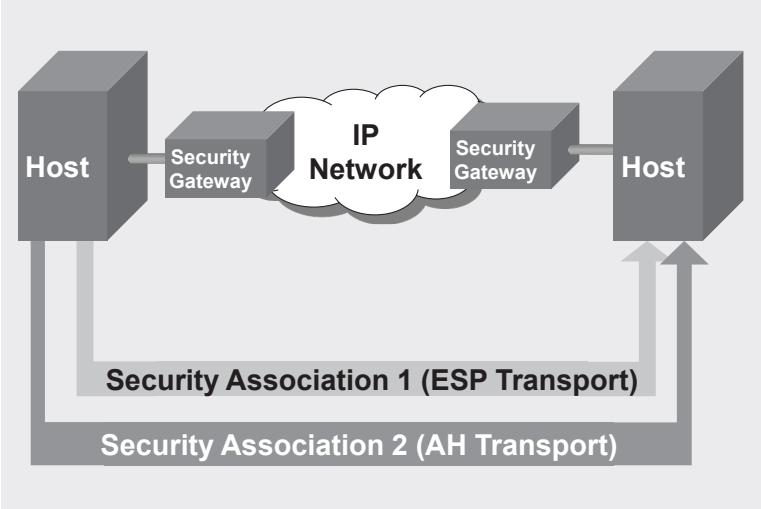


Figure 9.3 Transport Adjacency in SA Bundles

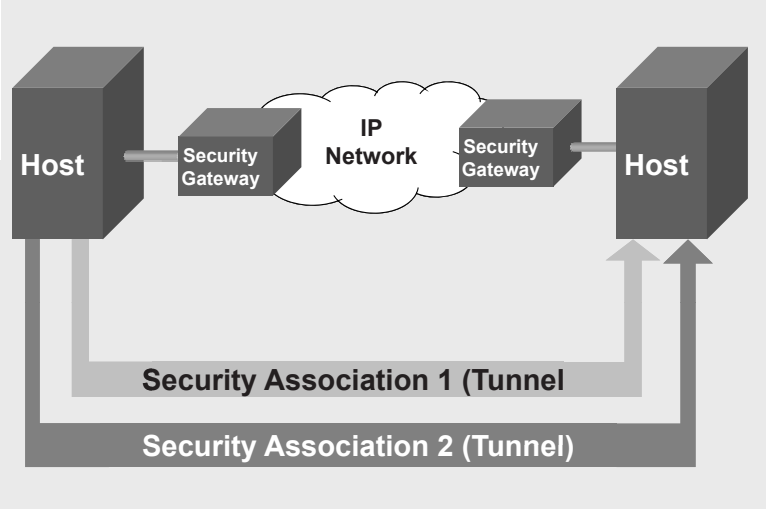


Figure 9.4 Iterating Tunneling in SA Bundles (Case 1)

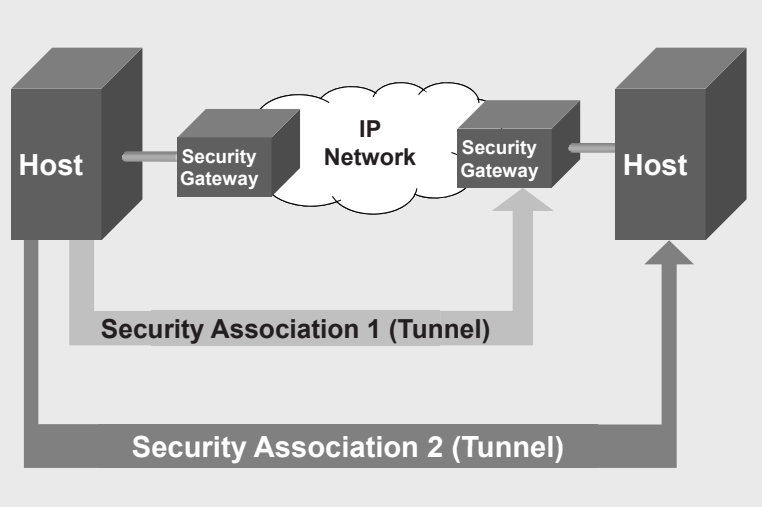


Figure 9.5 Iterating Tunneling in SA Bundles (Case 2)

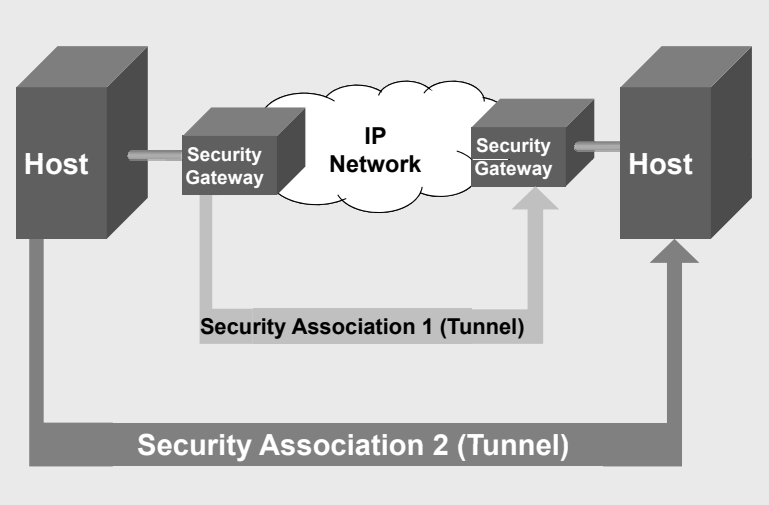


Figure 9.6 Iterating Tunneling in SA Bundles (Case 3)

connectionless integrity and anti-replay integrity. ESP provides in addition to the services offered by AH, confidentiality and limited traffic flow confidentiality. The next sections describe these two protocols in more detail.

Authentication Header (AH)

As mentioned above, AH [RFC_4302] offers data origin Authentication, connectionless integrity and optional anti-replay integrity services. It is used to provide end-to-end protection for the payload (transport mode) as well as partial protection for the IP header [RFC_4302]. AH can be also used to provide access control based on the key distribution mechanism in use.

AH can be used in tunnel or transport mode. This means, it can authenticate transmissions end-to-end between two hosts or between two gateways or a host and a gateway.

Encapsulating Security Payload (ESP)

As mentioned above, ESP [RFC_4303] offers confidentiality and limited traffic flow confidentiality in addition to services offered by AH, namely, data origin Authentication, connectionless integrity and optional anti-replay integrity services. ESP can be also used to provide access control based on the key distribution mechanism in use.

ESP can be used in tunnel or transport mode. This means, it can authenticate transmissions end-to-end between two hosts or between two gateways or a host and a gateway. It is used to provide end-to-end protection for the payload in transport mode but does not offer protection for the IP header in this case [RFC_4303]. If used in tunnel mode, it can provide protection for the entire inner IP packet including its header. Traffic flow confidentiality requires the selection of tunnel mode. It is most effective if applied at the gateways, where traffic can be aggregated according to the source-destination patterns. ESP is designed for use with symmetric algorithms. The SA specifies the encryption algorithm employed.

9.3 IKEv2

IKEv2 [RFC_5996] is a protocol defining the negotiation of security attributes and dynamic communication, establishment and maintenances of the source and destination states for the IPsec protocol.

IKE institutes an authentication between the two parties and then establishes the IKE security associations. This communication and negotiation is established before any secure data communications between the source and destination.

The source defines a set of negotiable attributes and sends it in a *REQUEST* message. The destination sends its selected parameters in a *RESPONSE* message back to the source. All IKE messages consist of a request/response message pair. This pair is also called an *EXCHANGE*. The first exchange of messages establishing an IKE SA are called the *IKE_SA_INIT* and *IKE_AUTH* exchanges. Figure 9.7 illustrates the initial message pair for IKE_SA_INIT. A typical IKE SA is established through 4 messages a pair for the IKE_SA_INIT exchange and a pair for IKE_AUTH.

After the IKE_SA_INIT message all communication is cryptographically protected through the negotiated security algorithms and keys.

9.4 Requirements and Constraints for the IPsec-O

An extension is to be proposed to the existing IPsec and IKEv2 protocols. Its purpose is to consider these Traffic QoS degradations caused by implementing security mechanisms during a Traffic QoS

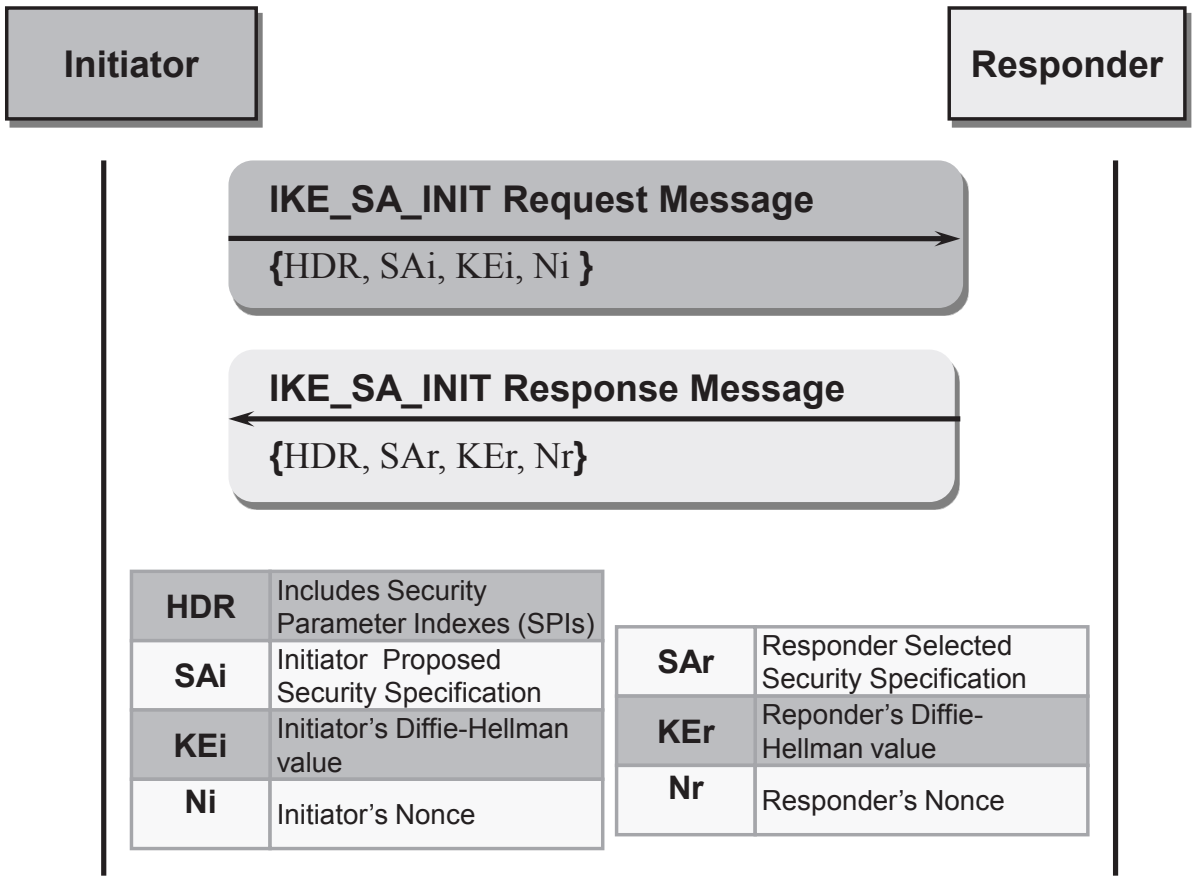


Figure 9.7 IKEv2 Security Specification Negotiation through IKE_SA_INIT Exchange

guaranteed, secure IP transmission. This is achieved by inducing a dominant strategy equilibrium among the participating nodes and implementation of the proposed DSIC-S cross-layer protocol in the previous chapters.

In addition to all assumptions and constraints of the proposed DSIC-S protocol, IPsec-O is designed to comply with the following constraints and assumptions. The players of the induced security strategic game are the security elements residing in different domains, as depicted in Figure 9.8. Since the security elements in one domain would follow the same security policy and the same utility goal and selfishness, we assume one SE per domain as representative and agent for that domain.

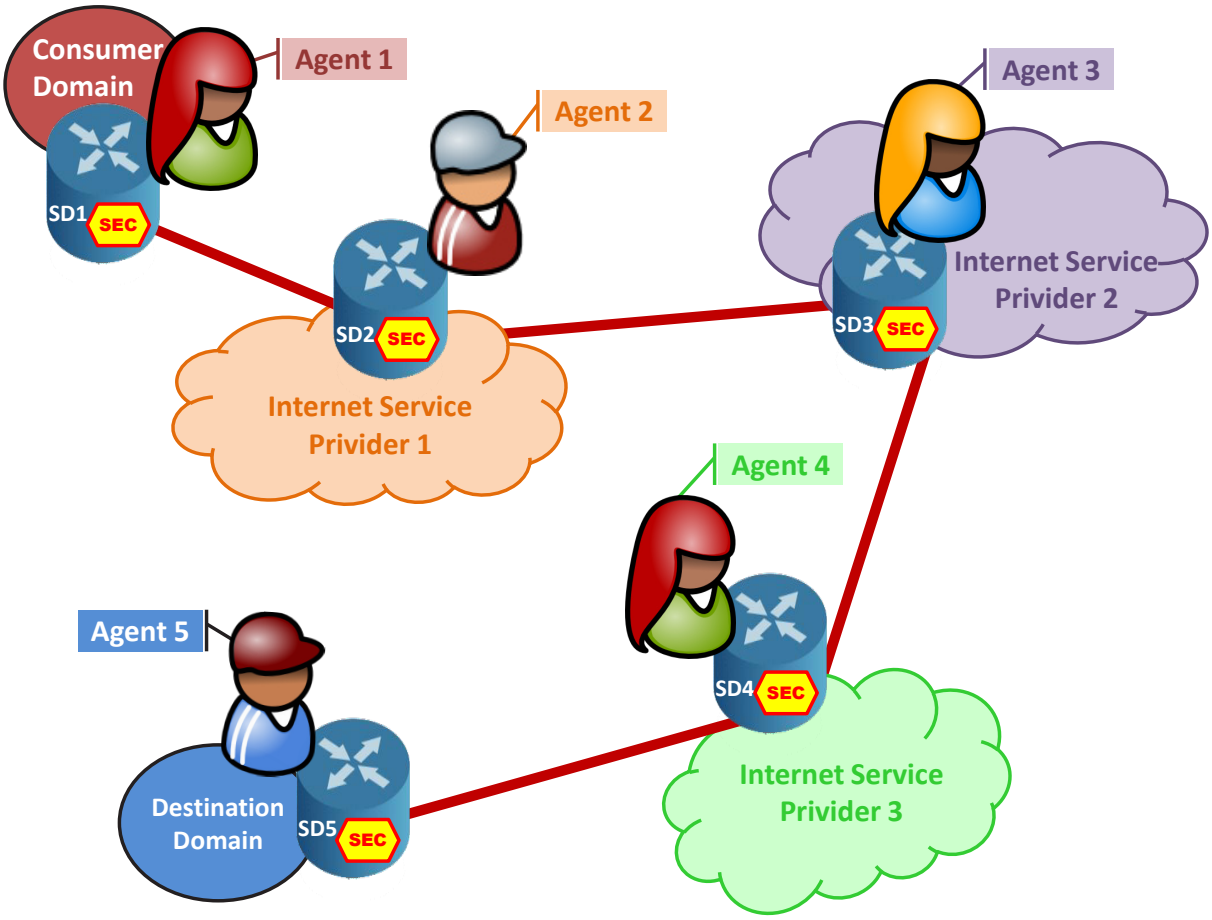


Figure 9.8 Example of an IPsec-O Strategic Game's Components

We assume that the average delay value for the encryption and decryption operations are listed in their SED table, as in Table 9.1. The proposed SED table is kept updated at each Security Element. It depicts the SE's operational delay caused by implementing different security algorithms for different security services and scenarios.

In addition, in order to enforce the security and delay tradeoff in the decision making of the nodes, as descried in the next sections, we propose that the VCG payment calculation be a function of the valuation and rank of the announced security algorithm based on consumer's valuations and desired security or delay. We propose different weighted formulas according to the governing

SED Table			
Security Association	**Security Scenario**	**Security Algorithm A_i**	**D_i ns**
Encapsulating Security Payload	**High-Security**	**AES- Rijndael-256**	**126**
		AES- Rijndael-192	**108**
		AES- Rijndael-128	**90**
	Medium-Security	**Twofish**	**80**
		Blowfish	**100**
		Triple-DES	**345**
	Low-Security	**RC6**	**80**
		RC5	**125**
		DES	**215**
Authentication Header	**High-Security**	**SHA-2-516**	**90**
		SHA-2-256	**80**
		SHA-1	**65**
	Medium-Security	**RIPEMD-160**	**80**
		Triple-DES	**345**
		AES-Rijndael-128	**590**
	Low-Security	**MD-5**	**45**
		DES	**160**
		RC6	**215**

Table 9.1 IP Security Element Delay Table, SED Table

SAR Table				
Security Association	**Security Scenario**	**Security Algorithm A_i**	**V_{Ai}**	**V_{Di}**
Encapsulating Security Payload	**High-Security**	**AES- Rijndael-256**	3	1
		AES- Rijndael-192	2	2
		AES- Rijndael-128	1	3
	Medium-Security	**Twofish**	2	3
		Blowfish	3	2
		Triple-DES	1	1
	Low-Security	**RC6**	3	3
		RC5	2	2
		DES	1	1
Authentication Header	**High-Security**	**SHA-2-516**	3	2
		SHA-2-256	2	3
		SHA-1	1	1
	Medium-Security	**RIPEMD-160**	3	3
		Triple-DES	2	2
		AES-Rijndael-128	1	1
	Low-Security	**MD-5**	3	3
		DES	2	2
		RC6	1	1

Table 9.2 IP Security Algorithm Ranking Table, SAR Table

security scenario for the calculation of these ranks. We assume, these valuations are listed in the consumer's Security Algorithm Ranking table, SAR, shown in Table 9.2.

Our security protocol is an extension to the existing IP protocols of the underlying network. We therefore assume the existence of basic IP security negotiations along with the possibility of transport adjacency and iterating tunneling Security Associations, SAs, to process SA bundles. The availability of IntServ's guaranteed services, RSVP and routing protocols are also assumed.

Moreover, we presume the availability of communication between these systems for resource reservation and routing calculations. Since delay efficiency is one of the main goals, there will be a synergetic effect to use IPsec-O in connection with a least cost routing protocol.

Our protocol does not require any special agreements among the ISPs along the path in terms of security or QoS. To address this, we propose a security model, which requires associations to be established each time between the consumer and a participating node along the path. This means IPsec-O hosts will be using both transport adjacency and iterating tunneling as required and desired by the consumer. This model is illustrated in Figure 9.9 as an example of a connection requiring Encapsulating Security Payload, ESP, security association offering confidentiality and data integrity between each two security elements.

Furthermore, the new extension should be proposed for the existing IP Security Architecture defined in [RFC_4301]. The new extension should be proposed based on the existing IP QoS control mechanisms and signaling protocols. Traffic QoS parameter degradations caused by performing security operations during the data transfer should be considered. The real-time, delay intolerant applications should not experience a loss of Traffic QoS during the data transfer phase because of the security operations. The Traffic QoS should keep its end–to–end characteristic. Traffic QoS degradation values, which are specific to each network element implementing IPsec and specific to each offered security mechanism, are known and available to the new extended IPsec. These values are generated and maintained by the security management system or user.

9.5 The IPsec-O Basics

The purpose of this chapter of the research work is to propose a solution, so that the toll of the implementations of the security mechanisms on the Traffic QoS would be accounted for, when guaranteeing a level of service to the real-time applications. In this view, the Traffic QoS controlling and signaling protocols, which make hard end-to-end guarantees for Traffic QoS and therefore are sensitive to additional sources of degradations, are the protocols of interest for this work. The proposed IPsec-O integrates the *IntServ's Guaranteed Service Protocol* into the current

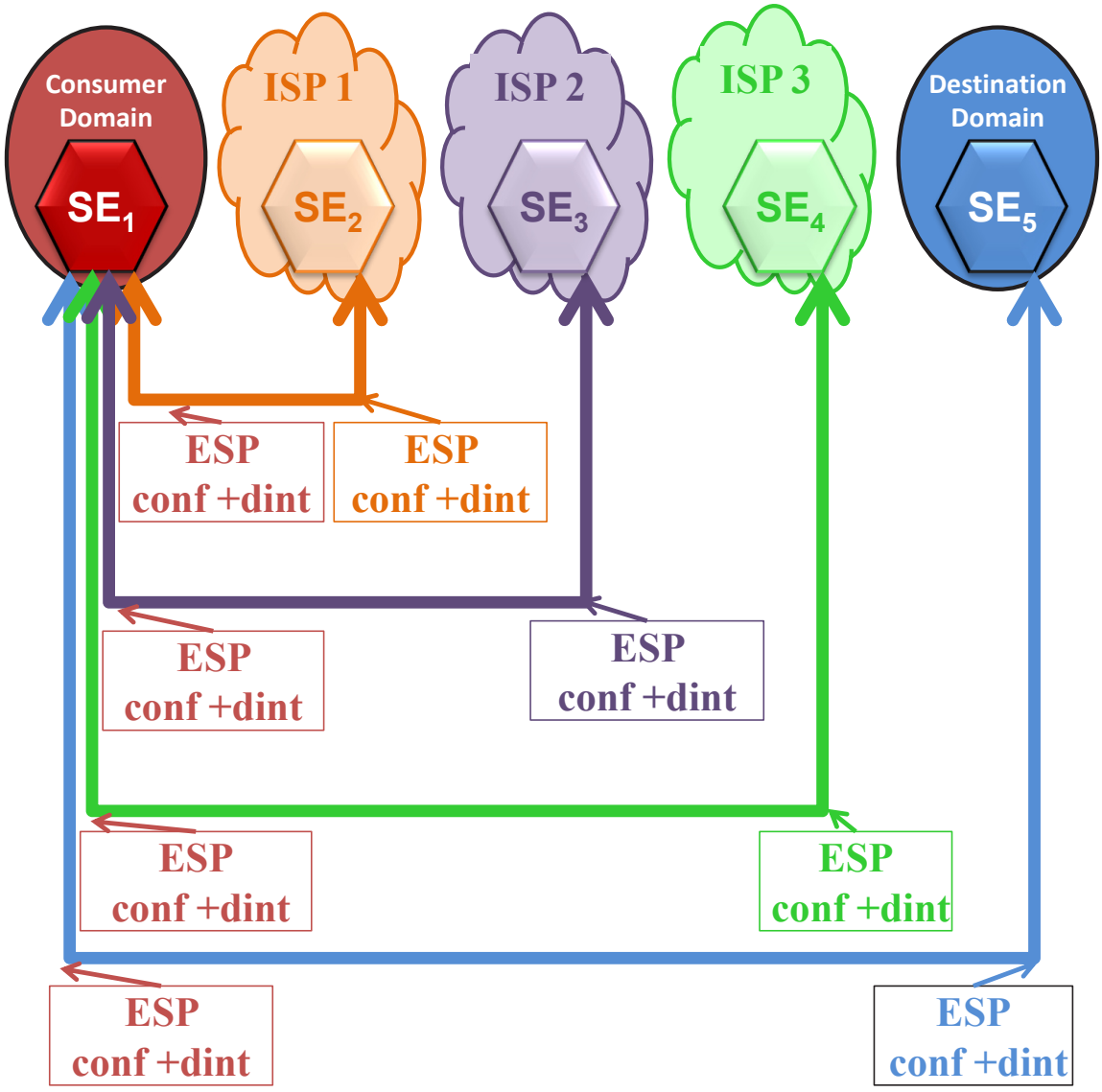


Figure 9.9 Example of IPsec-O Security Associations

IPsec architecture. This section describes the proposed extensions to the current IPsec [RFC_4301] procedures required to achieve this goal.

IKEv2 [3] defines the IP security association establishment procedures within the IPsec [2] framework. Proposing distinctive extensions to these existing protocols, IPsec-O establishes a desired equilibrium and tradeoff outcome between network-wide security level and end-to-end delay. IPsec-O induces a cheat-proof consumer-centric strategic game along the connection path.

The goal is that the underlying nodes, i.e. ISPs, though influenced by their own self-interest, to naturally find it to their best benefit to behave in a network-wide accepted, consumer requested and to-her-promised way.

IPsec-O provides the means to ensure that the nodes participate voluntarily and truthfully by means of "Incentive Compatibility" and "Individual Rationality" through a novel proposed incentive model. the proposed mechanism, also aims a "Dominant Strategy" to guarantee that the best response of each SE is to be truthful regardless of the responses of other SEs in the communication path. This is a very strong property of IPsec-O, which makes it cheat-proof and strategy-proof, see Theorem 9. This is specifically of highly valued importance for the real world implementations.

9.5.1 IP Security Scenario Model

One of the main properties of the IPsec-O protocol is its scenario-based design pertaining to the requested level of security. Three distinct levels of security, namely "High Security", "Medium Security", and "Low Security" are proposed. Table 9.2 depicts an example of a node's SAR table. The rankings are done within each scenario for the preferred available security algorithms pertaining to that particular scenario. The ranking of the valuation VA_i of a security algorithm in SE_i is according to the Equation 5.1 in DSIC-S, with value of 3 for the most preferred security algorithm per scenario, for which they have the highest desire and would select and perform first. The SAR table also has a valuation ranking according to the delay, as listed in the SED table, Table 9.1, caused by that particular security algorithm A_i. The ranking of the valuation of the delay D_i in SE_i is ccording to the Equation 5.2 in DSIC-S, with value of 3 for the lowest delay, most desirable, caused by a security algorithm per scenario. Table 9.2 shows this combination. The proposed customer ranking calculations are described in detail in previous chapters and follow those of DISC. R_{AT} is weighted based on the preference and ranking of the security algorithm and its strength. It is used for the high security case to determine V_A by comparing the other resulted R_{AT} values in the scenario, as listed in Table 9.2. R_{AT} is defined in Equation 5.3.

R_{DT} is weighted based on the occurring delay and used for medium and low security scenarios. V_D is then determined by comparing the other resulted R_{DT} values in the scenario, as listed in Table 9.2. It is defined in Equation 5.3.

This scenario-based structure enforces delay-efficiency and resource consumption management and reduction particularly for cases, where no strong security is needed. This is one of the most impending concerns of the underlying nodes attributed to their selfishness. This way and along with a proper design of an allocation function and incentive model, as described in the previous chapters for DSIC-S, and in the following sections, IPsec-O enforces a natural equilibrium among the players. The agents find it to their best interest to voluntarily follow the consumer's directions and demands in terms of level of security and QoS.

9.5.2 IKEv2-O Security Algorithm Selection

IKEv2 [RFC_5996] defines the initial negotiations for the selection of security algorithms and security associations for the connection. As explained earlier in this chapter, this negotiation is performed within a request/response communication pair through IKE_SA_INIT message to establish the parameters for the desired security within the path.

Our protocol is designed for a multi-agent network, with three or more agents. There are Np = {3, 4, ... , n} players or SEs along the path of a connection. The numbering of the nodes, $i \in N = \{1, 2, 3, \ldots, n\}$, starts with the consumer's node (i = 1) as depicted in Figure 9.8. This node is the social planner and induces its desired outcome of security level and delay through the whole communication path, by establishing transport adjacency and iterating tunneling security associations with all nodes along the path, as depicted previously in Figure 9.9. This way the mechanism and protocol make sure that proper network security between the domains is implied, without requesting any special agreements among the network providers. In her design, the consumer also chooses the nodes along the path of her communication according to a reputation model, which is based on the calculated payments from previous games. It is then imperative for the ISPs to earn excellent incentive credits, so that they can raise revenues by attracting more customers and more consumers wanting to become their permanent clients.

At the initiating phase of a connection, SE_1 first identifies the selected nodes along the path according to their reputation resulted from past games with the help of the routing and RSVP protocols. SE_1 starts the mechanism by requesting a certain level of security from the three proposed high, medium or low security scenarios in the SE_{i1} of IKE_SA_INIT message, as depicted in Figure 9.10, Step 1.

It identifies the security algorithms for both confidentiality and data integrity and their valuations within the desired level from its SAR table and sends these to all SEs along the path. For this process, V_A is calculated as described above and used for the high security scenario algorithms. V_D is calculated as above and is used for the algorithms in medium and low security scenarios. The SEs then according to their own preferences in their SAR table and implication of their selfishness, utility and gained reputation select one security algorithm for each security service and communicate this in a response in Flow2 to SE_1, as depicted in Figure 9.10, Step 2. If they cannot accommodate the conditions asked, they will be dropped from the game according to the allocation rule explained in Chapter 5. Once all Flow 2 replies have been received, the consumer can decide if enough players can qualify for the game. If not the consumer cancels this game and repeats its selection for another game. There will be a negative credit note according to the game and the transfers that those players, not willing to participate or not qualifying to participate, would have earned if they had qualified.

If all or an adequate number of intended players qualify, SE_1 can establish its IPsec-O secure connection with each of the SEs along the path, as seen in Figure 9.10, Step 3. It can also now calculate all the incentives through the VCG mechanism [ViWi_61]and Clarke pivotal rule [ClEd_71] described in previous chapters for each SE for this particular connection's strategic game using its valuation table for the selected security algorithms. Following the proposition that ISPs in real-life need stronger incentives to comply with the theoretical mechanism design models, we recommend the integration of these calculated payments into a reputation based model.

9.5.3 Modeling IPsec-O Strategic Game

To better convey the design, we model the strategic game for IPsec-O in the assumed case that ESP security association with confidentiality and data integrity security services are to be established

for each connection between a node and the consumer's node. Figures 9.9 and 9.10 depict the establishment of these security associations along the path.

The game is modeled as in DSIC-S. The type of agents, θ_i, and allocation functions for IPsec-O are as illustrated in Equations 6.1, 6.2 and 6.7.

The utility function u_i is the motivating factor for each SE_i in selection of its strategy and final decision of truthful participation. According to its general structure in quasi-linear environments, where incentives are calculated as payments, this utility is comprised of the node's valuation, v_i, of the security algorithm given the allocation function. In our case, this translates to its preferences in its SAR table, namely its type, given by Equation 6.10, of course, in addition to its gained transfer $t_i(\theta)$ given in Equation 6.9. The utility function $u_i(\theta)$ is calculated per Equation 6.11.

In the security strategic game, each participating node adhering to the allocation function for the given scenario will be reimbursed by a positive $t_i(\theta)$. In order to enforce the incentive compatibility through these payments and bring the theoretical model closer to the real-life scenario, we suggest the use of a reputation system. These payments should be then applied as an input into the system. This reputation is a vital factor for the participating ISPs to increase the consumer confidence and to attract more customers resulting to an increase in revenues.

9.5.4 SE_i's Strategic Decision

The strategic decision making process is described previously for DSIC-S. In the IPsec-O implementation given a security scenario for a connection, the consumer sends her set of choices and their valuation in the first flow of IKE_SA_INIT to each node. So each node knows which security algorithm has the most consumer valuation and will get the most payment $t_i(\theta)$, resulting to best incentive and reputation for that particular game.

9.5.5 IPsec-O Incentive Payment Model

After the SEs have made their selection of strategies and security algorithms, now the mechanism designer, SE_1, can calculate the payments for each node. As described in previous chapters for DSIC-S also in IPsec-O implementation she uses the VCG with Clarke's pivotal payment rule. $P_i(\theta)$

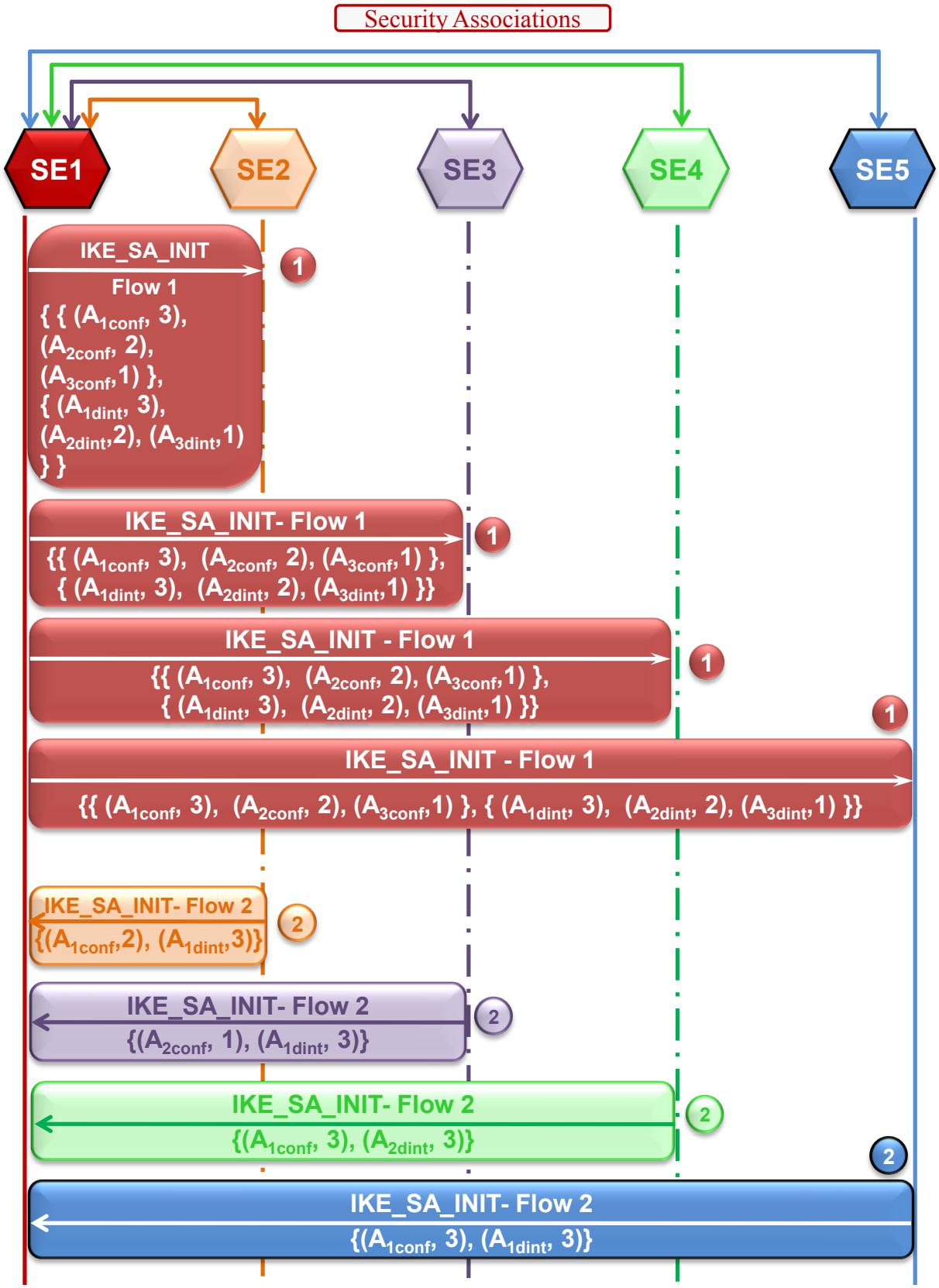


Figure 9.10 Example of IKEv2-O Security Association Establishment

is calculated according to Equation 6.12. IPsec-O design subtracts these taxes from the calculated weighted rankings for each scenario. The calculations and proposals for $t_i(\theta)$ are illustrated in Equation 6.13.This way, additional payments for the selection of most desirable algorithm are guaranteed. Through these incentive payments made at each game, and as means of performance evaluation in the social strategic game, the ISP increases its trustworthiness and attractiveness to draw more customers and increase her revenue.

9.5.6 IPsec-O Properties

IPsec-O inherits the Theorems 1-11 proved in Chapter 7 as properties of DSIC-S. To recap, these theorems prove the dominant strategy, strategy- and cheat-proofness of IPsec-O. They prove the dependency of the Clarke's payments as well as each agents' utility of the number of nodes and security associations (transforms) and the payments' independency of the actual delay and valuation of the agents. They furthermore prove that IPsec-O is allocatively efficient and Interim Individually Rational. They also prove that in IPsec-O all agents' selected strategies are socially desirable and Pareto optimal.

Discussion

To elaborate more on these very strong properties for IPsec-O, the independency of the incentives of the actual delays, makes the management of the network and mechanism cheat-proof, socially desirable and Pareto optimal. There is no need for announcement of misrepresented information from the ISP tainted by their economical selfishness. These properties make IPsec-O consumer-centric. This way, the customer can be rest assured that throughout her connection, the requested and agreed-upon security and QoS values are supported by all participating agents naturally and truthfully. IPsec-O induces a natural and inherently truthful autonomous end-to-end system-wide mechanism.

Chapter

10

ATM Networks

10.1 ATM in Today's and Future Hybrid Network Landscape

The world's network landscape is hybrid comprised of a multitude of technologies from ATM, Frame Relay, SONET, IP to wireless, sensor, just to name a few. These technologies are in use today, and are not to disappear anytime in the foreseeable future. In accordance with the implementation area and their specific competencies and capabilities these technologies get their own preferred niche of application in this colossal web and continue to serve the users. With respect to security research, the complexity of this heterogeneity is welcomed and reinforced to counteract the simplicity of malicious attacks. This fact makes it imperative, to ensure the continuous attention to the security aspects of each of these technologies, hence the additional layer-specific attention of this work.

ATM is an important component of this hybrid network at the datalink layer. It earns its widespread implementation not only from the fact that it is the first technology to effectively integrate voice, data and video over the same communication channel at any speed, but also from its competence to provide guaranteed scalable end-to-end Traffic QoS to the end user. Furthermore, as an advantage over other technologies, ATM offers an efficient backbone for advanced broadband communications networks. For its high traffic engineering capabilities and in addition to its other inherent features,

many IP data carriers, utility companies, Telecom companies and broadband solution providers have maintained ATM as their preferred backbone technology, to compensate the lack of these competencies in the connection-less IP. ATM networks offer high scalability in terms of bandwidth, interface speed, port density, switch size, network size and application support [McSo_99].

The innovators today are fully aware of this vital interconnectivity in the global hybrid network. The current state-of-the-art product development and research activities are a prove of this understanding. The most advanced future-oriented networking and security products and services integrate ATM technology in their designs, specially, where broadband communication is to be implemented [JuNe_10] [ZTE_09].

As the preferred choice of a core and backbone network technology for broadband applications, ATM has made itself a vital part of today's hybrid and heterogeneous network landscape implemented by a variety of organizations and applications. From Smart Girds [DOE_09][DOE_05][SDGE_06] [BrMa_09], Telecom [Ver_11][ATT_11][NCDC_11] to Electric Utility [YaKi_99][MPW_11] [CWLP_11][ZTE_09] companies to telemedicine [UAr_11] and healthcare providers, to distant-learning and DOD organizations widely take advantage of its superior inherent features in the US and worldwide.

ATM networks offer high scalability in terms of bandwidth, interface speed, port density, switch size, network size and application support [McSo_99]. Being the preferred choice of technology implemented by Telecom and Electric Utility companies worldwide as the core and backbone network for broadband applications, ATM makes a vital point of focus for this research work.

One of the to date least researched sources of Traffic QoS degradation is the security operations and implementations along secured network – and in special case of interest here the ATM network – connections. The different security services can be supported by a set of alternative security mechanisms. Each of which introduces a different value of degradation to various negotiable Traffic QoS parameters. This fact builds the core focus of this work. Without any attempt to consider these additional degradations, the already established negotiable Traffic QoS requested by the end user will decrease during the security operations.

The purpose of this work is to address this issue by developing security protocols for two cases – Out-Band and In-Band exchanges – to overcome the negative influences of security operations on the negotiable Traffic QoS parameters. *ATM Forum* Standard *UNI Signaling Specification 4.1* [USIG_41] and *ITU-T Recommendation Q.2931* [ITUQ_2931] define Cell Transfer Delay (CTD), Cell Delay Variation (CDV) and Cell Loss Ratio (CLR) QoS parameters to be negotiable during the ATM connection establishment phase.

In this chapter different network sources of influence are first introduced. The impact of different security services and mechanisms on the negotiable Traffic QoS parameters' degradations is discussed. The design requirements and constraints for the development of the new Out-Band and In-Band security message exchange protocols are derived.

10.2 Network Sources of Traffic QoS Degradation

Negotiable Traffic QoS parameters are mainly influenced by queuing behavior, propagation characteristics, node processing and buffer management of the intervening networks across the connection path. Figure 10.1 depicts an example of possible QoS degradation introduced by various components in ATM devices.

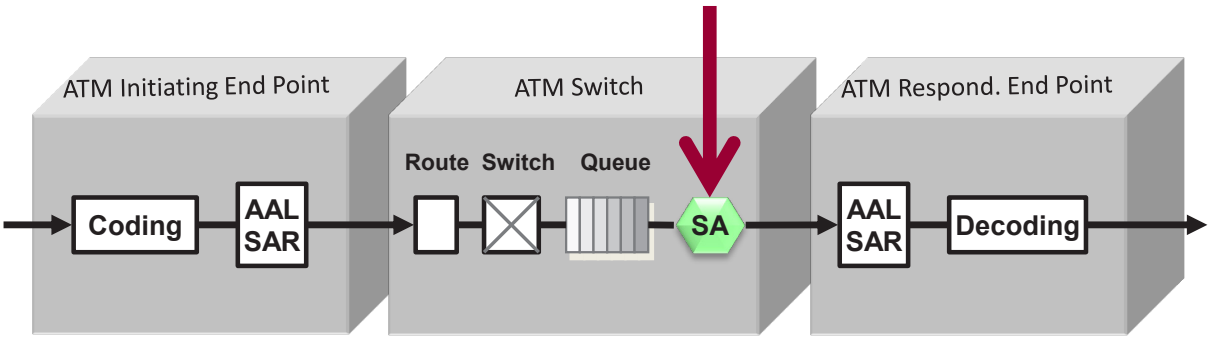


Figure 10.1 QoS Degradation in ATM networks, *Source [McDa_00]*

As illustrated, there are many sources with a high influence on the QoS parameters. The processing delays are an aggregation of values caused by the integrated components at the sending and receiving endpoints, the intermediate network elements, as well as their transmission medium. Figure10.2 depicts the different sources of overall performance influence and the affected Traffic QoS parameters. The influence of the security services and mechanisms implementations are the concern of this work and are further discussed below.

10.3 Traffic QoS Provisioning Requirements for the SME_Q Protocols

The new Out-Band (signaling-based) and In-Band (User Plane) SME_Q protocols are developed to address the Traffic QoS degradation problem caused by security agents providing security services along the ATM connections as discussed above.

It is the objective that security services should not affect the negotiated Traffic QoS of the connection negatively in the data transmission phase. That means these influences should be considered during the connection establishment phase by negotiating different set of security mechanisms. These requirements are derived on the basis of the two security message exchange architectures.

Network Resource of QoS Degradation	CTD	CDV	CLR
Propagation Delay	X		
Switch Architecture	X	X	X
Buffer Capacity	X	X	X
No. of Nodes	X	X	X
Traffic Load	X	X	X
Resource Allocation	X	X	X
SME Processing of SAs	X	X	X

Figure 10.2 Network Sources of QoS Degradation, *Source [TMS_41]*

10.4 Design Requirements and Constraints for the new Out-Band (Signaling_Based) SME_Q Protocol

The new Out-Band Security Message Exchange provisioning Traffic QoS protocol (SME_Q) should comply with the following requirements and constraints:

The Out-Band SME_Q should be a proposed extension to the existing Out-Band SME defined in *ATM Security Specification Version 1.1* [SEC_11]. The security services during the data transfer phase should not negatively impact the user requested Traffic QoS. At the same time, the user should not experience a loss of Traffic QoS during the data transfer phase because of the security operations either. This should be achieved by considering and compensating the Traffic QoS parameter degradations caused by security operations. It is assumed that the Traffic QoS degradation values, which are specific to each network element containing a security agent and selected security mechanism, are known and available to the new protocol in a table – SAC_Q table. This table is generated and maintained by the security management system. Figure 11.1 illustrates an example of a SAC_Q table in the next chapter. $CDV_{(I/R)}$, $CTD_{(I/R)}$ and $CLR_{(I/R)}$ are the respective values of a $SA_{(I)}$ or a $SA_{(R)}$. Furthermore, it is assumed that each SA within an ATM device receives the user defined acceptable and the received cumulative Traffic QoS parameters via an indication from the signaling entity. In addition, each SA within an ATM device should be able to update the cumulative Traffic QoS parameters for the selected security option and send these to the signaling entity via an indication. Each SA in a stand-alone implementation should also process the received signaling message and be able to extract the user defined acceptable and the received cumulative Traffic QoS parameters. It also should process the received signaling message and be able to update the cumulative Traffic QoS parameters according to its SAC_Q table for the selected security option. The Traffic QoS, however, should, at all times, still keep its end-to-end characteristic. Furthermore, the negotiation of different cryptographic algorithms and modes of operation should be transparent to the user. If any of the SME types were selected in a SAS_Q and a *QoS Parameter Information Element* was appended to the SETUP message, the Security Agent would initiate the SME_QC (SME_Q using QoS classes) protocol of the selected type (here Out-Band).

10.5 Design Requirements and Constraints for the new In-Band (User Plane) SME_Q Protocol

The new In-Band Security Message Exchange provisioning Traffic QoS protocol (SME_Q) should comply with the following requirements and constraints.

The In-Band SME_Q should be a proposed extension to the existing In-Band SME defined in *ATM Security Specification Version 1.1* [SEC_11]. The security services during the data transfer phase should not negatively impact the negotiated Traffic QoS. At the same time, the user should not experience a loss of Traffic QoS during the data transfer phase because of the security operations. This should be achieved by considering and compensating the Traffic QoS parameter degradations caused by security operations. It is assumed that the Traffic QoS degradation values, which are specific to each network element containing a security agent and selected security mechanism, are known and available to the new protocol in a table – SAC_Q table. This table is generated and maintained by the security management system. Figure11.1 illustrates an example of a SAC_Q table. Furthermore, it is assumed that each SA within an ATM device receives the user defined acceptable and the received cumulative Traffic QoS parameters via an indication from the signaling entity. In addition, each SA within an ATM device should be able to update the cumulative Traffic QoS parameters according to its SAC_Q table for the selected security option and send these to the signaling entity via an indication. Each SA in a stand-alone implementation should also process the received signaling message andbe able to extract the user defined acceptable and the received cumulative Traffic QoS parameters. It also should process the received signaling message and be able to update the cumulative Traffic QoS parameters for the selected security option. The Traffic QoS, however, should still keep its end-to-end characteristic. Furthermore, the negotiation of different cryptographic algorithms and modes of operation should be transparent to the user and they should be optimized to consider the trade-off between security and QoS. If any of the SME types were selected in a SAS_Q and a *QoS Parameter Information Element* was appended to the SETUP message, the Security Agent would initiate the SME_QC (SME_Q using QoS classes) protocol of the selected type (here In-Band).

Chapter

11

The SME_Q Protocols

The main objective of the SME_Q is to define the capability of providing security services while meeting or exceeding the user requested Traffic QoS parameters along a secured ATM connection. The user should not experience any decrease in Traffic QoS while the required security services are being provided.

As described in the last chapter as one of the requirements for the new protocol, these influences are assumed available in the SAC_Q Table, Table11.1, for all possible security algorithms and in combination with various modes of operation for each supported security service of the Security Agent. As described in previous chapter only the Traffic QoS degradations caused by mechanisms used for the confidentiality and data integrity security services are addressed in this work. According to *ATM Security Specification Version 1.1* [SEC_11], in some security configurations the SA might not be included in the ATM end system. It might be part of another ATM device like a switch. In this case, the ATM device must be capable of performing security operations such as encryption of cell payloads. In the case of data integrity, because of the involvement of the AAL Layer, the SA should be co–located in the ATM end system. Further more, the SA processes the signaling messages directly in a stand–alone implementation. However, it determines the connection

establishment through an indication from the signaling entity if it is co-located in an ATM device. For the simplicity and focus on the core subject the term Security Agent (SA) is used regardless of the hardware implementation.

General approach of the proposed SME_Q is to keep the end-to-end characteristic of the Traffic QoS. SME_Q provides the means for the selection of the appropriate security mechanisms in compliance with the requested Traffic QoS at the establishment phase of a virtual ATM connection.

Out-Band SME_Q complies with the current Out-Band SME protocol. It uses the Two-Way exchange protocol for exchanging the security information at the connection establishment phase. The initiating SA solely determines the security algorithm and mode of operation according to its governing security policy without any participation of its partner responding SA in the selection process. No actual negotiations take place in this case.

The In-Band SME_Q is defined based on the existing In-Band SME, which uses the Three-Way exchange protocol for exchanging the security information at the connection establishment phase. This way, it provides the means for negotiations between partner Security Agents.

The extensions to the existing SME and its information elements needed for the proposed SME_Q protocols are introduced in Section 11.1 – the SME_Q Protocol Basics. In the following Sections, 11.2 and 11.3, the two SME_Q protocols are defined according to different security architectures. Section 11.4 summarizes the developed simulation software and Section 11.5 illustrates the simulation results based on a real-life ATM network. For better understanding of the proposed solutions, the existing standards' style and method of detailed description is maintained and followed.

11.1 SME_Q Protocol Basics

This section defines the proposed changes and additions to the SSIE structure and SME procedures used in the current approved ATM Security Specification [SEC_11]. These additions and proposed changes grant the Traffic QoS provisioning capability to the current SME and as a whole make the SME_Q protocol. SME_Q uses SSIE_Q (enhanced SSIE) to convey the security information between SAs.

11.1.1 SA Characteristics with regards to QoS (SAC_Q)

The security operations in the security agent and/or a network element introduce additional degradation to the Traffic QoS parameters (the negotiable parameters are as of interest for this work) of a connection. Each SA and/or ATM device provide(s) specific rates of Traffic QoS decrease, while implementing different security algorithms and in combination with various modes of operation for each required security service.

As described in the next sections, the SME_Q requires the availability of these Traffic QoS degradation values for each SA and/or ATM device in form of a table. These values build the Security Agent's Characteristics with regard to QoS, SAC_Q, and are specific to each ATM system. Table 11.1 illustrates an example of a SAC_Q table.

SAC_Q $_{(l,R)}$ Table						
Security Service	Encryption	Mode	MAC	CDV$_{(l,R)}$ (µs)	CTD $_{(l,R)}$ (µs)	CLR $_{(l,R)}$ (10^{-n})
Confidentiality	DES	CBC	—	0.2	20	9
	DES	Counter	—	0.15	15	12
	Triple–DES	CBC	—	0.28	28	7
	Triple–DES	Counter	—	0.22	22	8
	FEAL	CBC	—	0.35	35	7
	FEAL	Counter	—	0.3	30	8
Data Integrity	—	—	DES/CBC	0.2	30	9
	—	—	H–MD5	0.25	25	10
	—	—	H–SHA–1	0.28	28	7
	—	—	FEAL/CBC	0.3	30	8

Table 11.1 SAC_Q Table

These Security Agent's Characteristics, SAC_Q parameters, for each initiating SA ($SA_{(I)}$) are, Cell delay Variation – $CDV_{(I)}$, Cell Transfer Delay – $CTD_{(I)}$, and Cell Loss Ratio – $CLR_{(I)}$. And in the same manner, the SAC_Q parameters for each responding SA ($SA_{(R)}$) are appended with (R).

For a required security service in the case of Out-Band SME_Q each initiating SA selects a security algorithm and mode of operation after examining its Traffic QoS values in its SAC_Q table against the governing security policy and the Traffic QoS objectives of the connection. In case of In-Band SME_Q, a SA negotiates the security algorithms and options according to its SAC_Q Table with the partner SA in order to still meet or exceed the requested Traffic QoS.

11.1.2 Security Association Section (SAS_Q) of the SSIE_Q

Figure 11.1 depicts the Security Association Section (SAS_Q) for the In-Band SME_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.

Bits								Octets
8	7	6	5	4	3	2	1	
Security Association Service Identifier								5
Security Association Section Length								5.1
Security Association Section Length (cont.)								5.2
Version			Transport Ind		Flow Indicator		Discard	5.3
Scope								5.4
Scope								5.5
Relative ID								5.6
Relative ID								5.7
Target Security Entity Identifier								5.8*
Security Service Data								**5.9**

Figure 11.1 Security Association Section (SAS_Q) of SME_Q

Security Service Data Section of the SSIE_Q

Figure 11.2 depicts the Security Service Data Section of the Out-Band and In-Band SME_Q. The first three octets of the current Security Service Data Section need modifications to allow the operations of the In-Band SME_Q. In case of the Out-Band SME_Q, however, only the first octet (Security Message Exchange Format) requires an additional type declaration. These changes are described below.

Bits								Octets
8	7	6	5	4	3	2	1	
Security Message Exchange Format								5.9
0	0	1	0	0	1	x	x	
x	x	x	x	x	x	x	x	5.9.1
SME TYPE								
Security Entity Identifier								5.9.2
Security Service Specification Section								5.9.3.x
Confidential Section								5.9.4.x
Authentication Section								5.9.5.x

Figure 11.2 Security Service Data Section of the SME_Q

Security Message Exchange Format of the SSIE_Q

In octet 5.9 two additional codes should be considered for the Two-Way and the Three-Way SME_Q as optional SME types. Figure 11.3 shows the additions to the current SME Format section.

Octet	Bits								Meaning
	8	7	6	5	4	3	2	1	
5.9.1	x	x	x	x	x	T	B	D	Two_way SME_Q
	x	x	x	x	x	T	B	D	Three_way SME_Q

Figure 11.3 Proposed additional SME_Q options for the current SME

Security Entity Identifier of the SSIE_Q

If in the previous section In-Band SME_Q option was 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 of the Three-Way 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. The format of the SA identifier is described in Section 6.2.1 of the [SEC_11].

Security Service Specification Section of the SSIE_Q

The *Security Service Algorithm Section* of this section is a required field for the In-Band SME_Q, which allows the negotiation of the algorithms and modes of operation according to its SAC_Q values. Figure 11.4 depicts the Security Service Specification Section. In the current SME the *Security Service Algorithm Option Section* is usually only used with the In-Band SME.

Bits								Octets
8	7	6	5	4	3	2	1	
1	0	0	0	1	0	0	0	x.1
Security Service Specification Section Identifier								
Security Service Declaration Section								X.2.x
Security Service Option Section								**x.3.x**
Security Service Algorithm Section								**x.4.x**

Figure 11.4 Security Service Option Section of SME_Q

Data Confidentiality Algorithm Primitive of the SSIE_Q

Figure 11.5 illustrates the *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 in [SEC_11]. They carry the Cell Delay Variation $CDV_{(I/R)}$ and Cell Transfer Delay $CTD_{(I/R)}$ parameters of respective $SA_{(R)}$ or $SA_{(R)}$., which are cumulated and calculated for the data transfer phase. Cell Loss Ratio is not a cumulative value, hence, is not proposed to be carried in these primitives.

Bits								Octets
8	7	6	5	4	3	2	1	
1	0	1	0	0	0	0	0	x.1
Data Confidentiality Algorithm Identifier								
Length of Data Confidentiality Algorithm Contents								x.2
Data Confidentiality Algorithm								x.3
OUI								x.4
OUI (continued)								x.4.1
OUI (continued)								x.4.2
Data Confidentiality Mode of Operation								x.5
OUI								x.6
OUI (continued)								x.6.1
OUI (continued)								x.6.2
Data Confidentiality Algorithm Details								x.7
Data Confidentiality Mode Details								x.8
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**

Figure 11.5 Data Confidentiality Algorithm Primitive of SME_Q

Octets x.9.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* [USIG_40] 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* [ITUQ_2931] using two bytes. These values indicate the Traffic 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.

Data Integrity Algorithm Primitive of the SSIE_Q

Figure 11.6 illustrates the *Data Integrity Algorithm* primitive for the In-Band SME_Q. The octets x.6.x and x.7.x are proposed additions to the current structure of this primitive in [SEC_11]. They carry the Cell Delay Variation $CDV_{(I/R)}$ and Cell Transfer Delay $CTD_{(I/R)}$ parameters of respective $SA_{(R)}$ or $SA_{(R)}$., which are cumulated and calculated for the data transfer phase. Cell Loss Ratio is not a cumulative value, hence, is not proposed to be carried in these primitives.

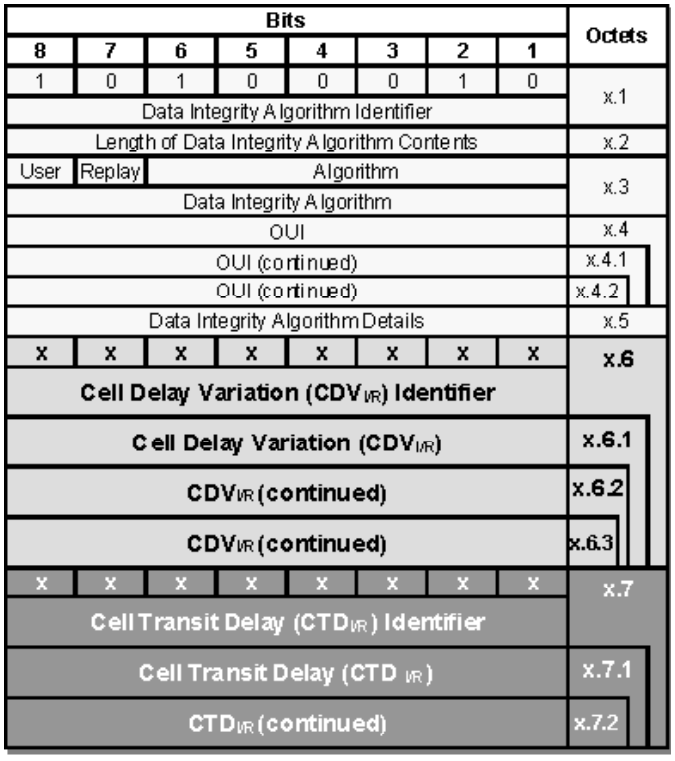


Figure 11.6 Data Integrity Algorithm Primitive of SME_Q

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 4.1* [USIG_41] using three bytes. Octets 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* [ITUQ_2931] using two bytes. These values indicate the Traffic 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.

11.2 The Out-Band SME_Q Protocol

As mentioned earlier, the Out-Band SME_Q takes the current Out-Band SME as framework. It extends SME's Information Elements (IE) and procedures to provide the Traffic QoS provisioning capability for the security operations. These additions are performed if the confidentiality and/or data integrity security service options are selected in combination with the Traffic QoS provisioning function. The following sections define the Out-Band SME_Q for the different network and security topologies.

The proposed Out-Band SME_Q is based on and an enhancement of the current Out-Band SME defined in *ATM Security Specification Version 1.1* [SEC_11]. Figure 11.7 illustrates the selection of security options for the Out-Band SME_Q. It only contains the Information Elements, which are of significance for the new procedures and carry the extensions for the new SME_Q. If the ATM initiating endpoint desires the request for specific Traffic QoS values for its secure point–to–point VC, it accompanies the requested parameters in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element* with the SETUP message and starts the connection establishment process, Figure 11.7, Step 1. The Out-Band SME_Q is processed; a security algorithm and a mode of operation are selected according to the SAC_Q Table of the initiating SA, the end user requested Traffic QoS and the governing security policy. The cumulative Traffic QoS values are corrected, the SSIE_Q is appended and the updated SETUP message is passed on to the network, Figure 11.7, Step 2. The responding SA examines the compliance with the Traffic QoS

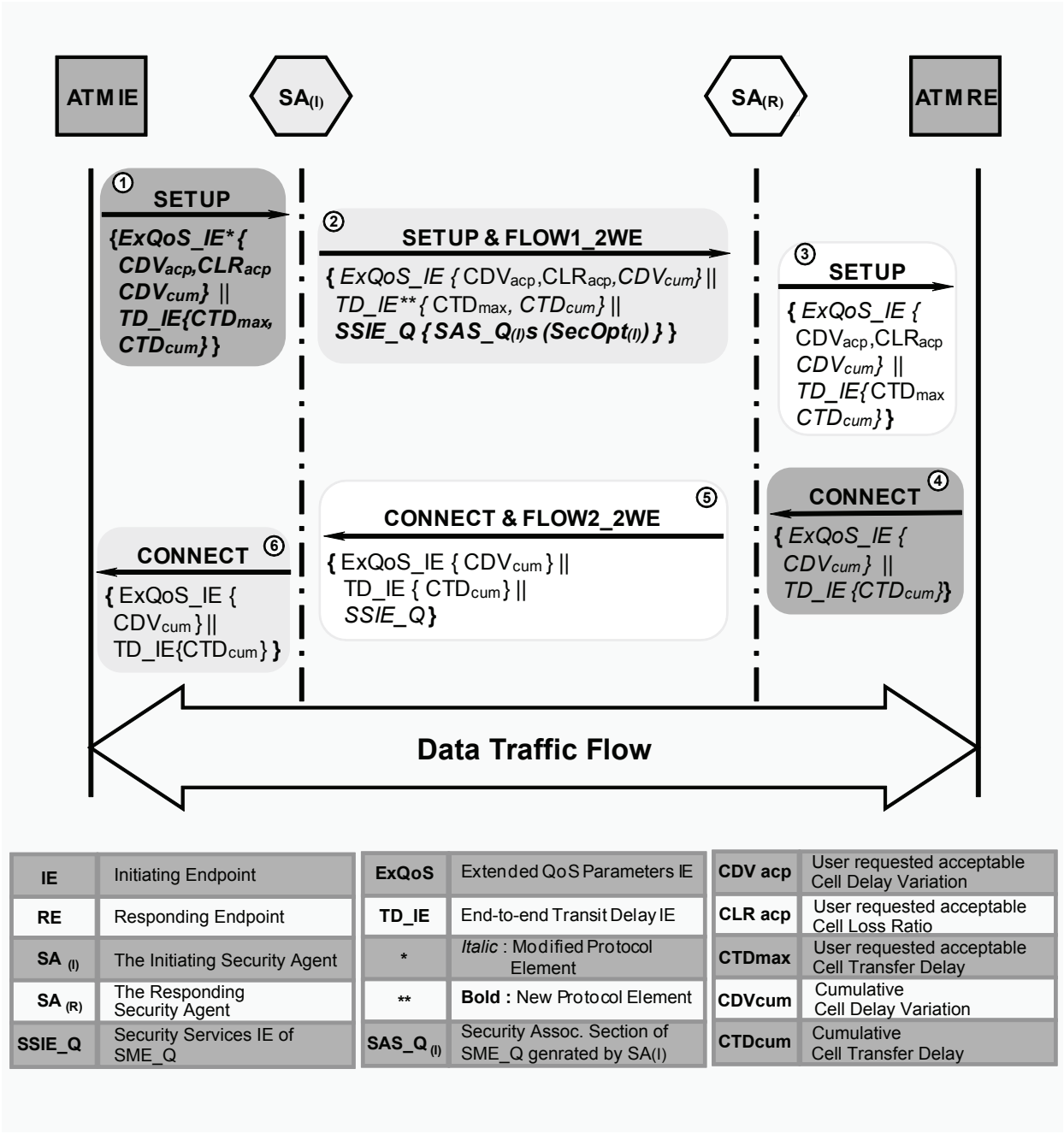


Figure 11.7 The Out-Band SME_Q

objectives according to the received cumulative and acceptable Traffic QoS values and their own SAC_Q table for the selected security algorithm and mode of operation received from their partner initiating SA. It in turn, updates the cumulative values and sends the updated SETUP message toward the ATM responding endpoint , Figure 11.7, Step 3. The responding endpoint updates the cumulative Traffic QoS parameters and sends them in a CONNECT message toward the responding

SA , Figure 11.7, Step 4. The CONNECT message is forwarded to the initiating endpoint, Figure 11.7, Steps 5 and 6, and the connection is established for the user data transfer. The details of the Out-Band SME_Q for the initiating and the responding SA is described in the next sections.

Initiating Security Agent Procedures

The following procedures are proposed in addition to the already existing procedures in Section 5.1.4.4 of the *ATM Security Specification Version 1.1* [SEC_11]. As explained earlier in this chapter, in a stand-alone implementation, the SA processes the signaling messages directly. If it is co-located in an ATM device, however, it communicates the signaling information through an indication with the signaling entity. For the sake of simplicity and focus on the core subject matter, the term *Security Agent (SA)* is used in the description of the procedures regardless of its hardware implementation. Figure 11.8 depicts the Out-Band SME_Q protocol procedure of the $SA_{(I)}$. It illustrates the procedural enhancements to the existing Out-Band SME protocol of *ATM Security Specification Version 1.1* [SEC_11]. Upon receipt of a SETUP message by $SA_{(I)}$, the $SA_{(I)}$ analyzes the values indicated in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element* for this connection. It compares the sum of the received cumulative parameters and the $SAC\_Q_{(I)}$ values of each Traffic QoS parameter for each security service with the indicated acceptable parameters, Figure 11.8, Step l. If none of the $SAC\_Q_{(I)}$ values complies with the acceptable values, the $SA_{(I)}$ rejects the connection request with the cause *SME_Q failure for the requested QoS,* Figure 11.8, Step 6. Otherwise, the $SA_{(I)}$ selects the security algorithm and mode of operation to include in the FLOW1_2WE (SETUP message) for the $SA_{(R)}$. The security algorithm and mode of operation are to be selected, so that, their values of Traffic QoS degradation according to the $SAC\_Q_{(I)}$ table, if added to the received cumulative values, would still result to lower rates than the requested acceptable parameters by the initiating endpoint. The CLR value of the suggested security mechanism is not a cumulative value and should either meet or be lower than the acceptable CLR requested by the initiating endpoint, Figure 11.8, Step 7. If more than one service are requested to be supported between the same SAs, only the parameters for confidentiality and data integrity, if applicable, should be considered in the calculations (S1: first security service

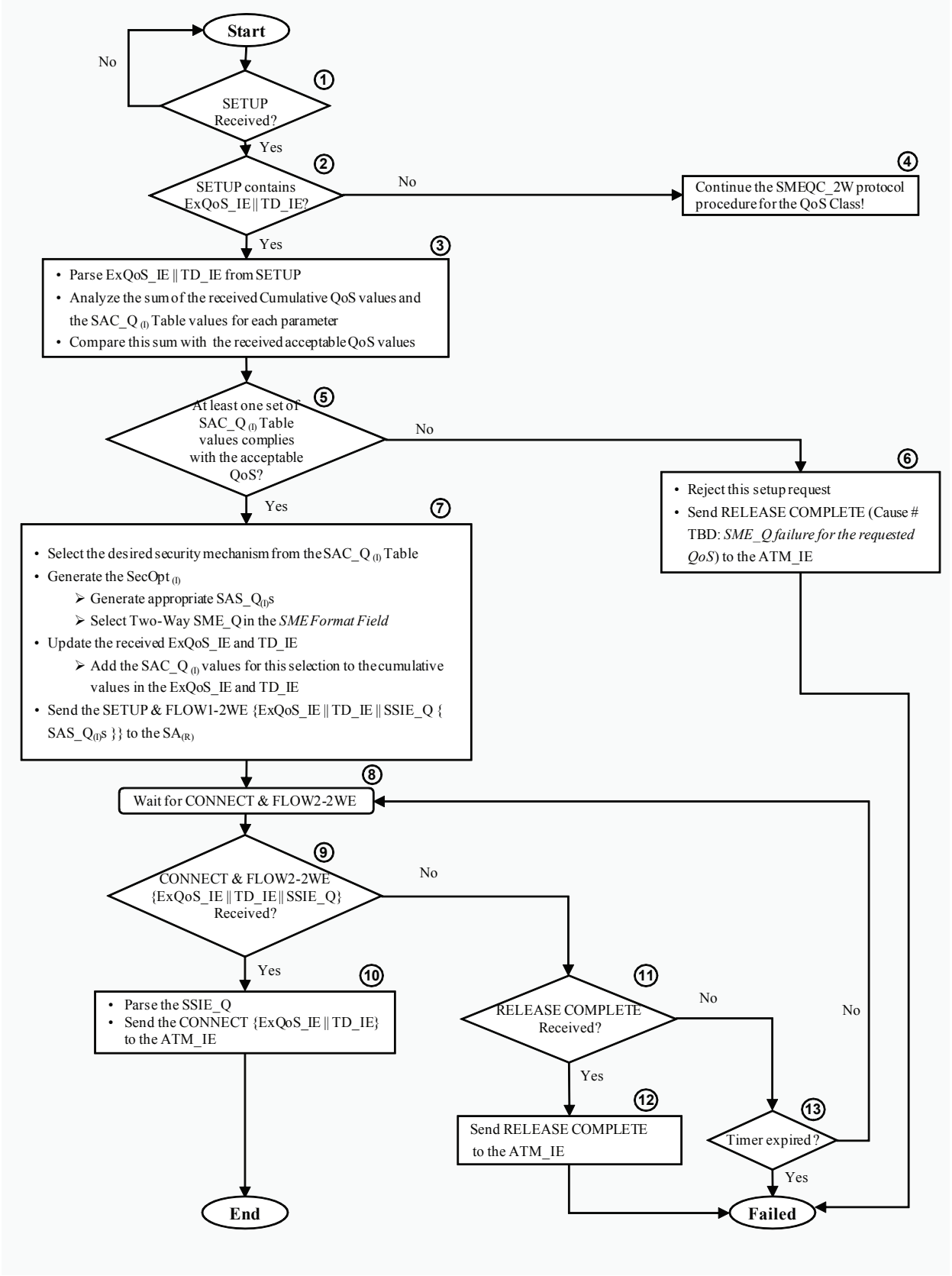


Figure 11.8 The Out-Band SME_Q Protocol Procedure for $SA_{(I)}$

and S2: second security service). Table 11.2 illustrates the nomenclature used in the succeeding equations:

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.
$CDV_{(I)}$	The $SA_{(I)}$ introduced Cell Delay Variation value according to its SAC_Q table.
$CDV_{(I),S1}$	The $SA_{(I)}$ introduced Cell Delay Variation value according to its SAC_Q table for the first service.
$CDV_{(I),S2}$	The $SA_{(I)}$ 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.
$CTD_{(I)}$	The $SA_{(I)}$ introduced Cell Transfer Delay value according to its SAC_Q table.
$CTD_{(I),S1}$	The $SA_{(I)}$ introduced Cell Transfer Delay value according to its SAC_Q table for the first service.
$CTD_{(I),S2}$	The $SA_{(I)}$ 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.
$CLR_{(I),S1}$	The $SA_{(I)}$ introduced Cell Loss Ratio according to its SAC_Q table for the first service.
$CLR_{(I),S2}$	The $SA_{(I)}$ introduced Cell Loss Ratio according to its SAC_Q table for the second service.

Table 11.2 TheQoS Parameter Definition for $SA_{(I)}$

$$CDV_{(I)} = CDV_{(I),S1} + CDV_{(I),S2} \quad (11.1)$$

$$CTD_{(I)} = CTD_{(I),S1} + CTD_{(I),S2} \quad (11.2)$$

$$CLR_{(I),S1} \leq CLR_{acp} \quad (11.3)$$

$$CLR_{(I),S2} \leq CLR_{acp} \quad (11.4)$$

and

$$CDV_{cum} + CDV_{(I)} < CDV_{acp} \quad (11.5)$$

$$CTD_{cum} + CTD_{(I)} < CTD_{max} \quad (11.6)$$

In this case, the $SA_{(I)}$ should first prioritize these services according to the governing security policy. It first selects its preferred option for the service with the higher priority (S1) and then optimizes the selection of the option for the second service (S2) according to the chosen selections for the first option. In addition to Equation 11.4, the following should apply,

$$CDV_{(I),S2} < CDV_{acp} - CDV_{cum} - CDV_{(I),S1} \quad (11.7)$$

$$CTD_{(I),S2} < CTD_{max} - CTD_{cum} - CTD_{(I),S1} \quad (11.8)$$

Naturally, in addition to Equation 11.3, the same equations apply for the first service:

$$CDV_{(I),S1} < CDV_{acp} - CDV_{cum} - CDV_{(I),S2} \quad (11.9)$$

$$CTD_{(I),S1} < CTD_{max} - CTD_{cum} - CTD_{(I),S2} \quad (11.10)$$

If any option does not satisfy the above, it should not be considered and provided for the security measurements of that connection. In case only one of the services (confidentiality or data integrity) is supported, the parameters associated with S2 are equal to zero. This would be a special case of the above equations. After the selection of the security options, the $SA_{(I)}$ selects *two–way SME_Q* in the *SME Format* field of the SAS_Qs. The $SA_{(I)}$ then forwards the SETUP message toward the responding endpoint appending the SSIE_Q with the generated SAS_Qs along with the *Extended QoS Parameters Information Element* and *End-to-End Transit Delay Information Element* after adding its contributions to the cumulative Traffic QoS parameters for the selected option Figures 11.8, Step 2 and 11.9, Step 7,

$$CDV_{cum}\ [:= CDV_{cum\ (received)} + CDV_{(I)}\ ] < CDV_{acp} \quad (11.11)$$

$$CTD_{cum}\ \ [:= CTD_{cum\ (received)} + CTD_{(I)}\ ] < CTD_{max} \quad (11.12)$$

Upon receipt of the CONNECT message (FLOW2_2WE) from the $SA_{(R)}$, the $SA_{(I)}$ proceeds according to Section 5.1.4.4.2.2 of [SEC_11] and forwards the CONNECT message to the initiating endpoint, Figures 11.7, Step 6 and 11.8, Step 10.

Responding Security Agent Procedures

The following procedures are proposed in addition to the already existing procedures in Section 5.1.5.1.1 of the *ATM Security Specification Version 1.1* [SEC_11]. For the simplicity and focus on the core subject, the term *Security Agent (SA)* is used in the description of the procedures regardless of its hardware implementation. The term $SA_{(I)}$ is used for the initiating Security Agent and $SA_{(R)}$

for the responding Security Agent. Figure 11.9 depicts the Out-Band SME_Q protocol procedure of the $SA_{(R)}$. It illustrates the procedural enhancements to the existing Out-Band SME protocol of *ATM Security Specification Version 1.1* [SEC_11].

Upon receipt of a SETUP message on the responding side, the $SA_{(R)}$ analyzes the parameter values indicated in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element* for this connection. It compares the sum of the received cumulative parameters and the local $SAC\_Q_{(R)}$ values for the security algorithm and mode of operation selected by its partner initiating $SA_{(I)}$, with the requested acceptable parameters, Figure 11.9, Step l.

If the calculated values for the security algorithm and mode of operation selected and sent by the partner $SA_{(I)}$ do not satisfy the Traffic QoS objectives, the $SA_{(R)}$ rejects the connection request with the cause *SME_Q failure for the requested QoS*, Figure 11.9, Step 6. If more than one service are requested to be supported between the same SAs, 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 in the last chapter, only these two services impact the network Traffic QoS during the user data transfer phase.

Table 11.3 illustrates the nomenclature used in the succeeding equations for the responding SA. In general, the Equations 11.1-11.6 should be satisfied for the same parameters on the responding side, $CDV_{(R)}$, $CTD_{(R)}$, and $CLR_{(R),}$. In case only one service (confidentiality or data integrity) is supported, the parameters associated with S2 are equal to zero. The $SA_{(R)}$ then proceeds according to Section 5.1.4.4.3.1 of [SEC_11] and forwards the SETUP message toward the responding endpoint along with the *Extended QoS Parameters Information Element* and *End-to-End Transit Delay Information Element* after adding its contributions to the cumulative Traffic QoS parameters for the selected option, Figures 11.9, Step 7 and 11.7, Step l. Here by, satisfying the Equations 11.11-11.12 for the same parameters on the responding side. Upon receipt of the CONNECT message, the $SA_{(R)}$ proceeds according to Section 5.1.4.4.2.2 of [SEC_11] and forwards the CONNECT message toward the initiating endpoint, Figures 11.9, Step 7and 11.7, Step10.

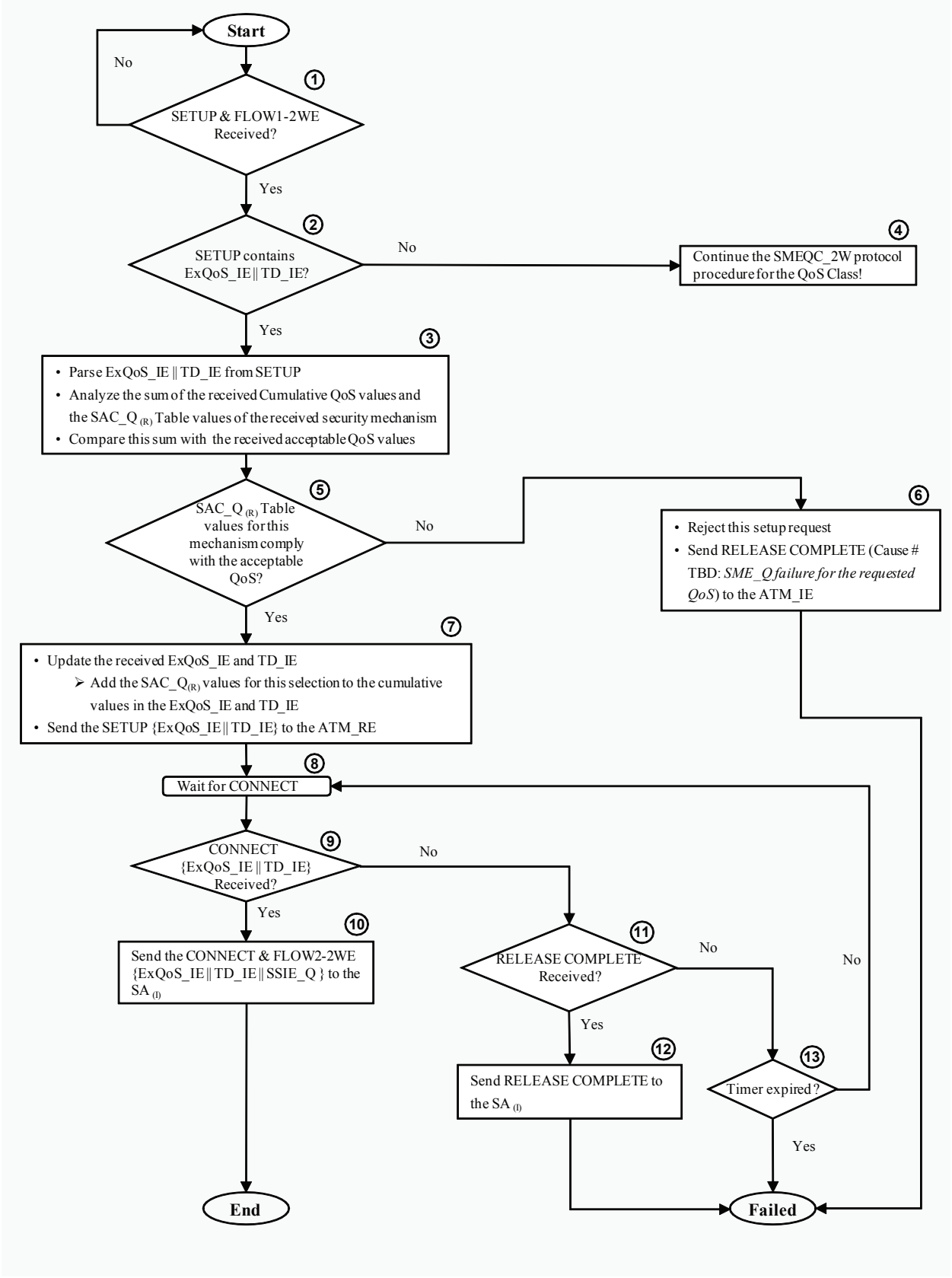


Figure 11.9 The Out-Band SME_Q Protocol Procedure for $SA_{(R)}$

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.
$CDV_{(R)}$	The $SA_{(R)}$ introduced Cell Delay Variation value according to its SAC_Q table.
$CDV_{(R),S1}$	The $SA_{(R)}$ introduced Cell Delay Variation value according to its SAC_Q table for the first service.
$CDV_{(R),S2}$	The $SA_{(R)}$ 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.
$CTD_{(R)}$	The $SA_{(R)}$ introduced Cell Transfer Delay value according to its SAC_Q table.
$CTD_{(R),S1}$	The $SA_{(R)}$ introduced Cell Transfer Delay value according to its SAC_Q table for the first service.
$CTD_{(R),S2}$	The $SA_{(R)}$ 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.
$CLR_{(R),S1}$	The $SA_{(R)}$ introduced Cell Loss Ratio according to its SAC_Q table for the first service.
$CLR_{(R),S2}$	The $SA_{(R)}$ introduced Cell Loss Ratio according to its SAC_Q table for the second service.

Table 11.3 The QoS Parameter Definition for $SA_{(R)}$

11.2.1 Connections with Nesting and Multiple Security Associations

The Out-Band SME_Q also supports the different topologies of security associations: nesting (as well as the one point overlap) and no overlap (sequenced). For each of these different topologies there are always two sides for each security association: the initiating side and the responding side. Further more, according to the *ATM Security Specification Version 1.1* [SEC_11] the nesting level is limited to 16 security associations within an ATM connection path. Figure 11.10 illustrates the selection of security options for the Out-Band SME_Q in the case of nesting security associations. Figure 11.11 illustrates this selection in the case of sequenced security associations. The following sections describe these procedures in detail.

Initiating Security Agent Procedures

The following procedures are proposed in addition to the already existing procedures in Section 5.1.4.4 of the *ATM Security Specification Version 1.1* [SEC_11]. For the simplicity and focus on the

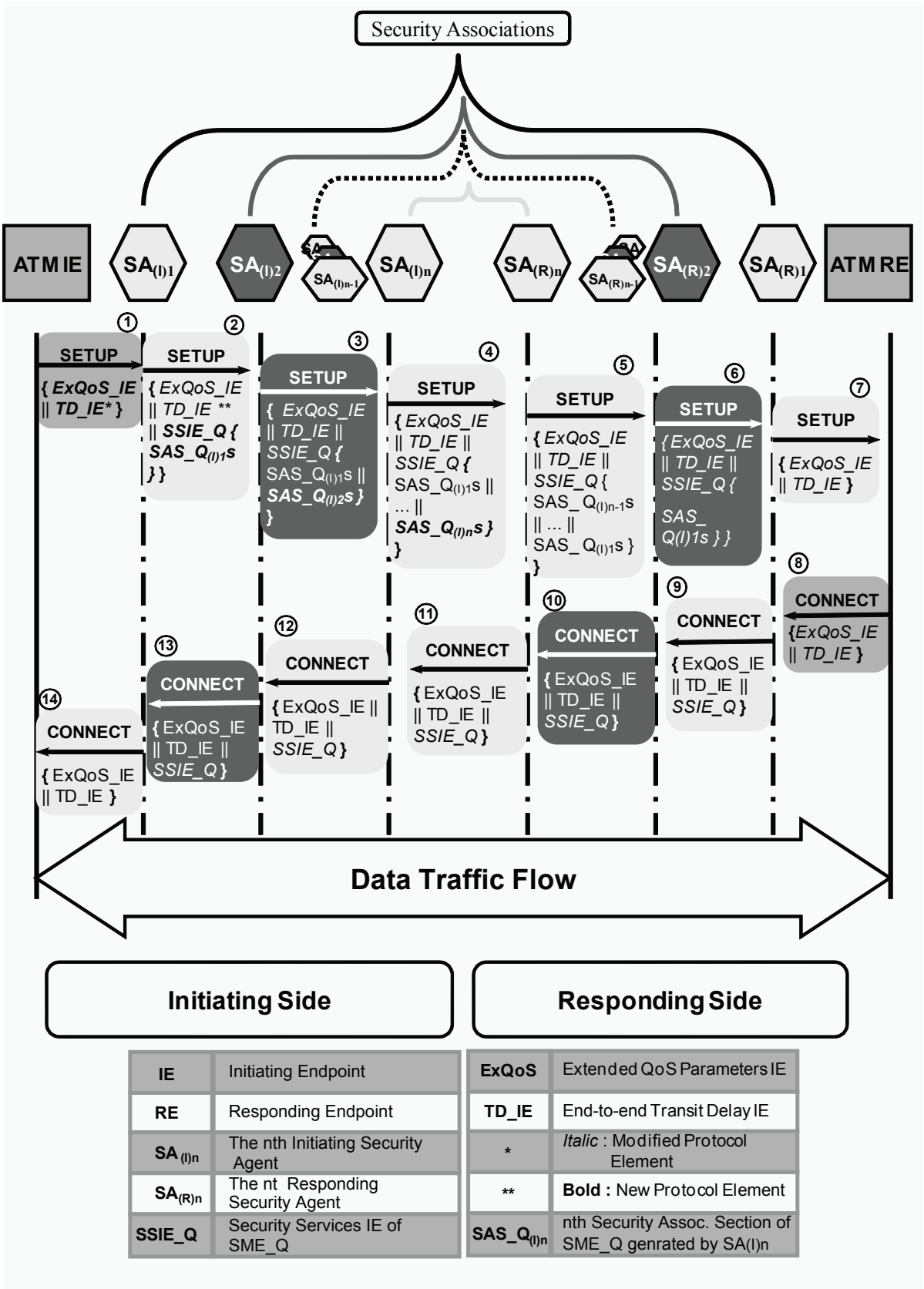
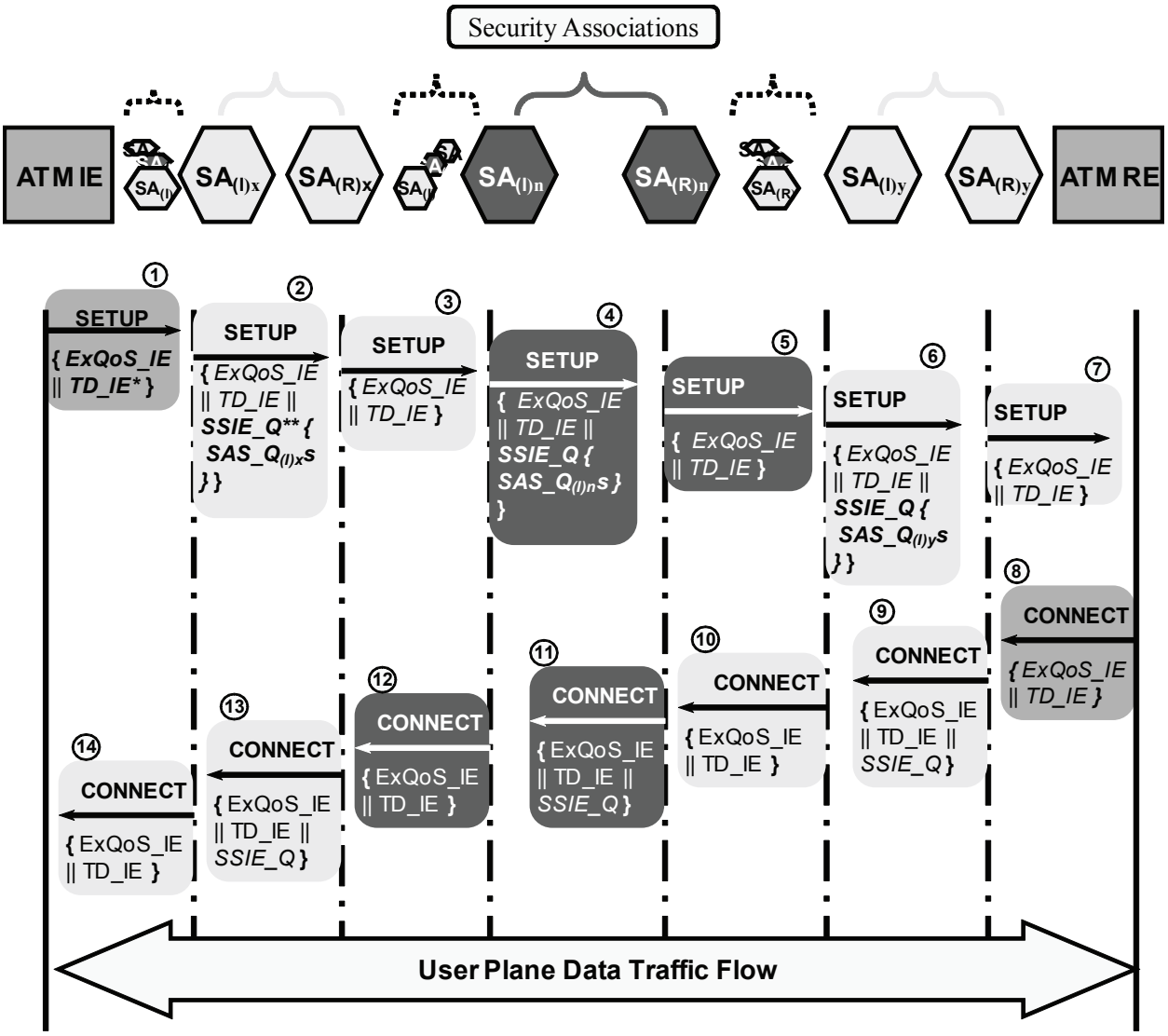


Figure 11.10 Out-Band SME_Q with Nesting Security Associations



IE	Initiating Endpoint	**ExQoS**	Extended QoS Parameters IE
RE	Responding Endpoint	**TD_IE**	End-to-end Transit Delay IE
$SA_{(I)n}$	The nth Initiating Security Agent	*	*Italic* : Modified Protocol Element
$SA_{(R)n}$	The nth Responding Security Agent	**	**Bold :** New Protocol Element
SSIE_Q	Security Services IE of SME_Q	**$SAS\_Q_{(R)n}$**	nth Security Assoc. Section of SME_Q genrated by SA(R)n

Figure 11.11 Out-Band SME_Q with Sequenced Security Associations

core subject, the term *Security Agent (SA)* is used in the description of the procedures regardless of its hardware implementation. The term $SA_{(I)}$ is used for the initiating Security Agent and $SA_{(R)}$ for the responding Security Agent. Figure 11.10 depicts the Out-Band SME_Q protocol procedure for each $SA_{(I)}$ along the path ($SA_{(I)n}$to $SA_{(I)1}$).

Nesting Security Associations

Upon receipt of a SETUP message by the first $SA_{(I)1}$ on the path of a connection with nesting security associations Figure 11.10, the $SA_{(I)1}$ analyzes the values indicated in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element* for this connection. It compares the sum of the received cumulative parameters and the $SAC\_Q_{(I)1}$ values for the security services which are to be provided according to the governing security policy, with the indicated acceptable parameters Figure 11.10;2.

If the SETUP message does not contain the *Extended QoS Parameter Information Element*, the Traffic QoS is determined according to the QoS class for the connection. The SME_QC procedures using the QoS class will be implemented, Figure 11.8, Step 4.

If none of the $SAC\_Q_{(I)1}$ values can complies with the acceptable values, the $SA_{(I)1}$ rejects the connection request with the cause *SME_Q failure for the requested QoS*, Figure 11.8, Step 6.

Otherwise, the $SA_{(I)1}$ selects the security algorithm and mode of operation (in case of Confidentiality security service) to include in the FLOW1_2WE (SETUP message) for the partner $SA_{(R)1}$. The security algorithm and mode of operation are to be selected, so that, the values of Traffic QoS degradation according to the $SAC\_Q_{(I)1}$ table, if added to the received cumulative values, would still result to lower rates than the requested acceptable parameters by the initiating endpoint. The CLR value of the suggested security mechanism is not a cumulative value and should either meet or be lower than the acceptable CLR indicated originally by the initiating endpoint, Figure 11.8, Step 7.

If more than one service are requested to be supported between $SA_{(I)1}$ and $SA_{(R)1}$, 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). Table 11.4 illustrates the nomenclature used for the nesting associations. Here too, the Equations 11.1-11.6 should be satisfied. In this case, the $SA_{(I)1}$ should first prioritize these services according to its governing security policy. It first selects its 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 chosen selections for the first option. In addition to Equatiopn 11.24, the Equations 11.7-11.10 should be satisfied. If any option does not

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.
$CDV_{(I)n}$	The $SA_{(I)n}$ introduced Cell Delay Variation value according to its SAC_Q table.
$CDV_{(I)n,S1}$	The $SA_{(I)n}$ introduced Cell Delay Variation value according to its SAC_Q table for the first service.
$CDV_{(I)n,S2}$	The $SA_{(I)n}$ 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.
$CTD_{(I)n}$	The $SA_{(I)n}$ introduced Cell Transfer Delay value according to its SAC_Q table.
$CTD_{(I)n,S1}$	The $SA_{(I)n}$ introduced Cell Transfer Delay value according to its SAC_Q table for the first service.
$CTD_{(I)n,S2}$	The $SA_{(I)n}$ 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.
$CLR_{(I)n,S1}$	The $SA_{(I)n}$ introduced Cell Loss Ratio according to its SAC_Q table for the first service.
$CLR_{(I)n,S2}$	The $SA_{(I)n}$ introduced Cell Loss Ratio according to its SAC_Q table for the second service.

Table 11.4 The QoS Parameter Definition for $SA_{(I)n}$

satisfy the above, it should not be selected and provided for the security measurements between $SA_{(I)1}$ and $SA_{(R)1}$. In case only one service (confidentiality or data integrity) is supported, the parameters associated with S2 are equal to zero. After the selection of the security options, the $SA_{(I)1}$ selects *two–way SME_Q* in the *SME Format* field of the SAS_Q_1s. The $SA_{(I)1}$ then forwards the SETUP message toward the responding endpoint appending the SSIE_Q with the generated SAS_Q_1s along with the *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element* after adding its contributions to the cumulative Traffic QoS parameters. Here by, satisfying the Equations 11.11 and 11.12 .

Upon receipt of a SETUP message by the second $SA_{(I)2}$ on the path of a connection with nesting security associations Figure 11.10;l, the $SA_{(I)2}$ proceeds with the above described procedures for $SA_{(I)1}$. It makes its selection according to the $SAC\_Q_{(I)2}$ table. It then updates the *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element* by adding its contributions to the cumulative Traffic QoS parameters for the selected option, according to the Equations 11.11and 11.12. The $SA_{(I)2}$ then forwards the updated SETUP message toward the responding endpoint. The same procedures will be processed on the initiating side each time a $SA_{(I)}$

$_n$ of a connection with nesting security associations Figure 11.10;4 receives a SETUP message. New SAS_ns for the security services are generated and addressed to the corresponding partner $SA_{(R)n}$, the *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element* are updated by adding the contributions for the selected security algorithm and mode of operation to the cumulative Traffic QoS parameters.

$$CDV_{cum} \; [:= CDV_{cum\ (received)} + \; CDV_{(I)n}\,] < CDV_{acp} \tag{11.13}$$

$$CTD_{cum} \; [:= CTD_{cum\ (received)} + \; CTD_{(I)n}\,] < CTD_{max} \tag{11.14}$$

The updated SETUP message is forwarded toward the responding endpoint. Upon receipt of the CONNECT message (FLOW2_2WE) from the $SA_{(R)n}$, the corresponding $SA_{(I)n}$ proceeds according to Section 5.1.4.4.2.2 of [SEC_11] and forwards the CONNECT message to the initiating endpoint Figure 11.10; 12-14.

Sequenced Security Associations

In the case of sequenced security associations Figure 11.11, the same procedures as described above for the simple case of Out-Band SME_Q are processed each time between two partner SAs. The ATM connection path is put together from more than one pair of partner $SA_{(I)}$ and $SA_{(R)}$ following each other sequentially. Thus, the security algorithm selection is initiated and completed each time only between the two following partner SAs. This decreases the complexity of the procedure. Upon receipt of a SETUP message by a $SA_{(I)n}$, a new SSIE_Q is generated (if not already available) with new SAS_Q_ns and addressed to the corresponding partner $SA_{(R)n}$ Figure 11.10;2,4,6. The *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element* are updated by adding the contributions for the selected mechanism in the newly generated SAS_Q_ns to the cumulative Traffic QoS parameters according to the Equations 11.13-11.14. The updated SETUP message is forwarded toward the responding endpoint. Upon receipt of the CONNECT message (FLOW2_2WE) from the $SA_{(R)n}$, the corresponding $SA_{(I)n}$ proceeds according to Section 5.1.4.4.2.2 of [SEC_11] and forwards the CONNECT message to the initiating endpoint Figure 11.10; 10,12,14.

Responding Security Agent Procedures

The following procedures are proposed in addition to the already existing procedures in Section 5.1.5.1.1 of the *ATM Security Specification Version 1.1* [SEC_11].

For the simplicity and focus on the core subject, the term *Security Agent (SA)* is used in the description of the procedures regardless of its hardware implementation. The term $SA_{(I)}$ is used for the initiating Security Agent and $SA_{(R)}$ for the responding Security Agent. Figure 11.9 depicts the Out-Band SME_Q protocol procedure for each $SA_{(R)}$ along the path ($SA_{(R)n}$to $SA_{(R)1}$).

Nesting Security Associations

Upon receipt of a SETUP message by the first peer SA ($SA_{(R)n}$) on the path of a connection with nesting security associations Figure 11.10, the $SA_{(R)n}$ examines if there are any $SAS\_Q_n s$ addressed to it. It then analyzes the parameter values indicated in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element* for this connection. It compares the sum of the received cumulative parameters and its SAC_Q table – $SAC\_Q_{(R)n}$ – values for the security mechanism selected by its partner initiating $SA_{(I)n}$, with the indicated acceptable parameters. $SA_{(R)n}$ ignores all other *SAS_Qs,* if available, Figures 11.9, Step l and 11.10, Step 5. If the calculated values for the security algorithm and mode of operation selected and sent by the partner $SA_{(I)n}$ do not comply with the acceptable values, the $SA_{(R)n}$ rejects the connection request with the cause *SME_Q failure for the requested QoS*, Figure 11.9, Step 6. If more than one service are requested to be supported between $SA_{(I)n}$ and $SA_{(R)n}$, 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 in the last chapter, only these two services impact the network Traffic QoS during the user data transfer phase. Table 11.5 illustrates the nomenclature used for sequenced associations. The Equations 11.1-11.6 should be satisfied for the same parameters on the responding side, $CDV_{(R)n}$, $CTD_{(R)n}$, and $CLR_{(R)n}$. In case only one of the services (confidentiality or data integrity) is supported, the parameters associated with S2 are equal to zero. the Equations 11.7 and 11.8 should be satisfied for the same parameters on the responding side. The $SA_{(R)n}$ then proceeds according to Section 5.1.4.4.3.1 of [SEC_11] and forwards the SETUP message toward

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.
$CDV_{(R)n}$	The $SA_{(R)n}$ introduced Cell Delay Variation value according to its SAC_Q table.
$CDV_{(R)n,S1}$	The $SA_{(R)n}$ introduced Cell Delay Variation value according to its SAC_Q table for the first service.
$CDV_{(R)n,S2}$	The $SA_{(R)n}$ 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.
$CTD_{(R)n}$	The $SA_{(R)n}$ introduced Cell Transfer Delay value according to its SAC_Q table.
$CTD_{(R)n,S1}$	The $SA_{(R)n}$ introduced Cell Transfer Delay value according to its SAC_Q table for the first service.
$CTD_{(R)n,S2}$	The $SA_{(R)n}$ 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.
$CLR_{(R)n,S1}$	The $SA_{(R)n}$ introduced Cell Loss Ratio according to its SAC_Q table for the first service.
$CLR_{(R)n,S2}$	The $SA_{(R)n}$ introduced Cell Loss Ratio according to its SAC_Q table for the second service.

Table 11.5 TheQoS Parameter Definition for $SA_{(R)n}$

the responding endpoint along with the *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element* after adding its contributions to the cumulative Traffic QoS parameters for the selected option, Figures 11.9, Step 7 and 11.10 Step 5. The Equations 11.13 and 11.14 should be satisfied for the same parameters on the responding side. Upon receipt of the CONNECT message, the $SA_{(R)}$ proceeds according to Section 5.1.4.4.2.2 of [SEC_11] and forwards the CONNECT message toward the initiating endpoint Figures 11.10; 11.Upon receipt of a SETUP message by the second peer SA ($SA_{(R)n-1}$) on a path with nesting security associations Figure 11.10, the $SA_{(R)n-1}$ proceeds with the above-described procedures for $SA_{(R)n}$. It examines if there are any $SAS\_Q_{n-1}s$ addressed to this SA. It compares the sum of the received cumulative parameters and the $SAC\_Q_{(R)n-1}$ values for the security algorithm and mode of operation selected by its partner initiating $SA_{(I)}$, with the indicated acceptable parameters for the selected option. After updating the *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element* by adding its contributions to the cumulative Traffic QoS parameters for the

selected option, Figure 11.9, Step 7. The Equations 11.13 and 11.14 should be satisfied for $SA_{(R)n-1}$. The $SA_{(R)n-1}$ now proceeds according to Section 5.1.4.4.3.1 of [SEC_11] and forwards the updated SETUP message toward the responding endpoint.

The same procedures will be processed on the responding side of a path with nesting security associations Figure 11.10;5,6,7 each time an $SA_{(R)x}$ receives a SETUP message. The *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element* are updated by adding the contributions to the cumulative Traffic QoS parameters for the received selected security mechanism according to Equations 11.13 and 11.14 for $SA_{(R)x}$ .The $SA_{(R)x}$ now proceeds according to Section 5.1.4.4.3.1 of [SEC_11] and forwards the updated SETUP message toward the responding endpoint. Upon receipt of the CONNECT message, the $SA_{(R)x}$ proceeds according to Section 5.1.4.4.2.2 of [SEC_11] and forwards the CONNECT message toward the initiating endpoint Figure 11.10; 9,10,11.

Sequenced Security Associations

In the case of sequenced security associations, Figure 11.11, the same procedures for the simple case of Out-Band SME_Q are processed each time between two partner SAs as described above. In this case, the ATM connection path is put together from more than one pair of partner $SA_{(I)}$ and $SA_{(R)}$ following each other sequentially. Thus, the security algorithm selection is initiated and completed each time between the two following partner SAs and then passed to the network. This decreases the complexity of the procedure. Upon receipt of a SETUP message, the $SA_{(R)n}$ analyzes the parameter values indicated in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element* for this connection. It compares the sum of the received cumulative parameters and the local $SAC\_Q_{(R)n}$ values for the security services which are to be provided, with the requested acceptable parameters Figure 11.11; l,5,7. If the calculated values for the security algorithm and mode of operation selected and sent by the partner $SA_{(I)n}$ do not satisfy the performance objectives, the $SA_{(R)n}$ rejects the connection request with the cause *SME_Q failure for the requested QoS*, Figure 11.9, Step 6. If more than one service are requested to be supported between the same SAs, 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 in the last chapter, only these two services impact the network Traffic QoS during the user data transfer phase. Table 11.5 illustrates the nomenclature for the sequenced associations. The Equations 11.1-11.4 should be satisfied for $SA_{(R)n}$. In case only one service (confidentiality or data integrity) is supported, the parameters associated with S2 are equal to zero. The Equations 11.5 and 11.6 should be satisfied for $SA_{(R)n}$. The $SA_{(R)n}$ then proceeds according to Section 5.1.4.4.3.1 of [SEC_11] and forwards the SETUP message toward the responding endpoint along with the *Extended QoS Parameters Information Element* and *End-to-End Transit Delay Information Element* after adding its contributions to the cumulative Traffic QoS parameters for the selected option, satisfying Equation 11.13 and 11.14 for $SA_{(R)n}$. Upon receipt of the CONNECT message, the $SA_{(R)n}$ proceeds according to Section 5.1.4.4.2.2 of [SEC_11] and forwards the CONNECT message toward the initiating endpoint Figure 11.11; 9,11,13.

11.3 The In-Band SME_Q Protocol

As mentioned earlier, the In-Band SME_Q takes the current In-Band SME described in chapter 4 as framework. It extends SME's Information Elements (IE) and procedures to provide the Traffic QoS provisioning capability for the security operations. These additions are performed if the confidentiality and/or data integrity security service options are selected in combination with the Traffic QoS provisioning function. The following sections define the In-Band SME_Q for the different network and security topologies.

The proposed In-Band SME_Q is based on and an enhancement of the current In-Band SME defined in *ATM Security Specification Version 1.1* [SEC_11]; therefore providing services for the point–to–point connections only. Figure 11.12 illustrates the selection of security options for the In-Band SME_Q. It only contains the Information Elements, which are of significance for the new procedures and carry the extensions for the new SME_Q. If the ATM initiating endpoint desires the request for specific Traffic QoS values for its secure point–to–point VC, it accompanies the requested parameters in the *Extended QoS Parameter Information Element* and the *End-to-End*

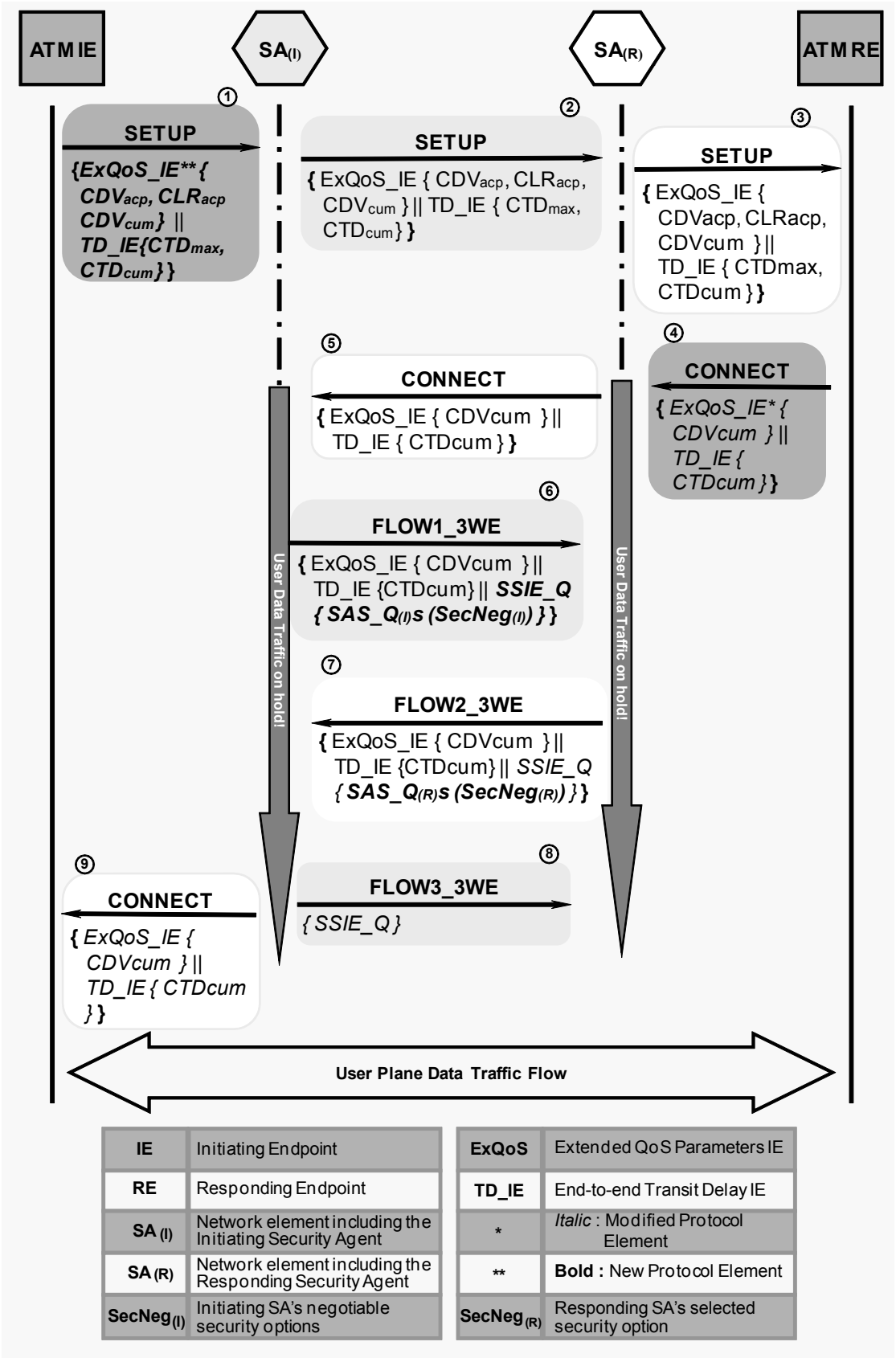


Figure 11.12 The In-Band SME_Q

Transit Delay Information Element, with the SETUP message, Figure 11.12, Step 1 and starts the connection establishment process . Upon receipt of a CONNECT message on the upstream, the user data transfer is blocked and the Three-Way SME_Q is processed, Figure 11.12, Steps 4 and 5. The security algorithms and modes of operation are negotiated, Figure 11.12, Steps 6,7, and 8. The CONNECT message is then forwarded to the initiating endpoint and the connection is unblocked for the user data transfer, Figure 11.12, Step 9. The details of the In-Band SME_Q is described for the initiating and the responding SA in the next sections.

Initiating Security Agent Procedures

The following procedures are proposed in addition to the already existing procedures in Section 5.1.5.1.1 of the *ATM Security Specification Version 1.1* [SEC_11].

The new protocol assumes the communication of the acceptable and cumulative Traffic QoS values between the signaling entity and SA. The calculations are done for the worst case scenario, i.e. the one with the highest values of degradation according to the $SAC\_Q_{(I)}$ table, if added to the received cumulative values, should 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 requested value. Figure 11.13 depicts the In-Band SME_Q protocol procedure of the $SA_{(I)}$. It illustrates the procedural enhancements to the existing In-Band SME protocol of ATM Security Specification Version 1.1 [SEC_11]. Upon receipt of a SETUP message on the initiating side, the $SA_{(I)}$ compares the $SAC\_Q_{(I)}$ values for the security services, which are to be provided with the indicated acceptable parameters, Figure 11.13, Step l. It 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, Figure 11.13, Step 7. If the SETUP message does not contain the *Extended QoS Parameter Information Element*, the QoS is determined according to the QoS class for the connection. The SME_QC procedures using the QoS class will be implemented, Figure 11.13, Step 4.

If none of the security options in the table complies with the acceptable values, the $SA_{(I)}$ rejects the connection request with the cause *SME_Q failure for the requested QoS*, Figure 11.13, Step 6.

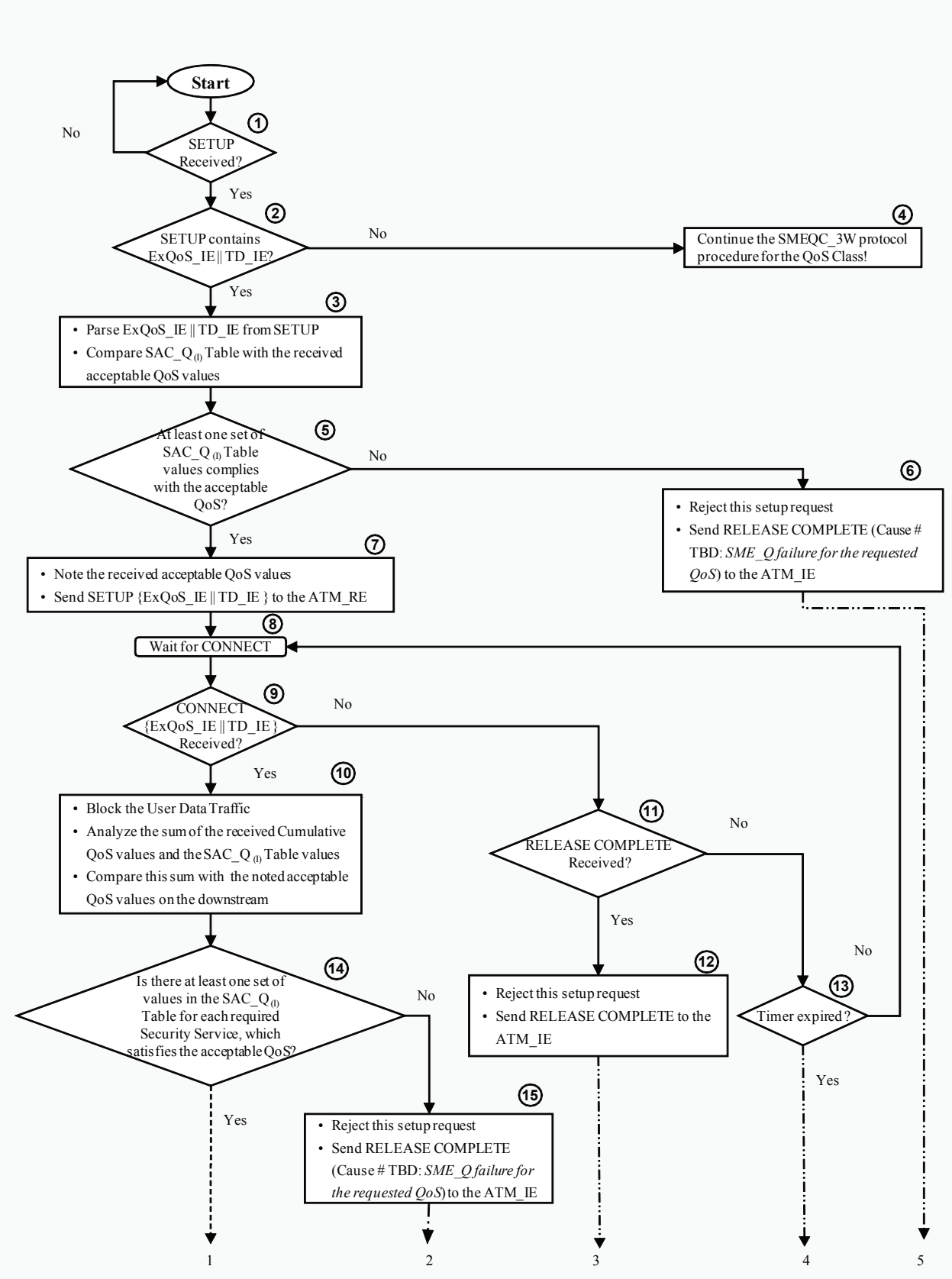


Figure 11.13 The In-Band SME_Q Protocol Procedure for $SA_{(I)}$ (to continue on next page)

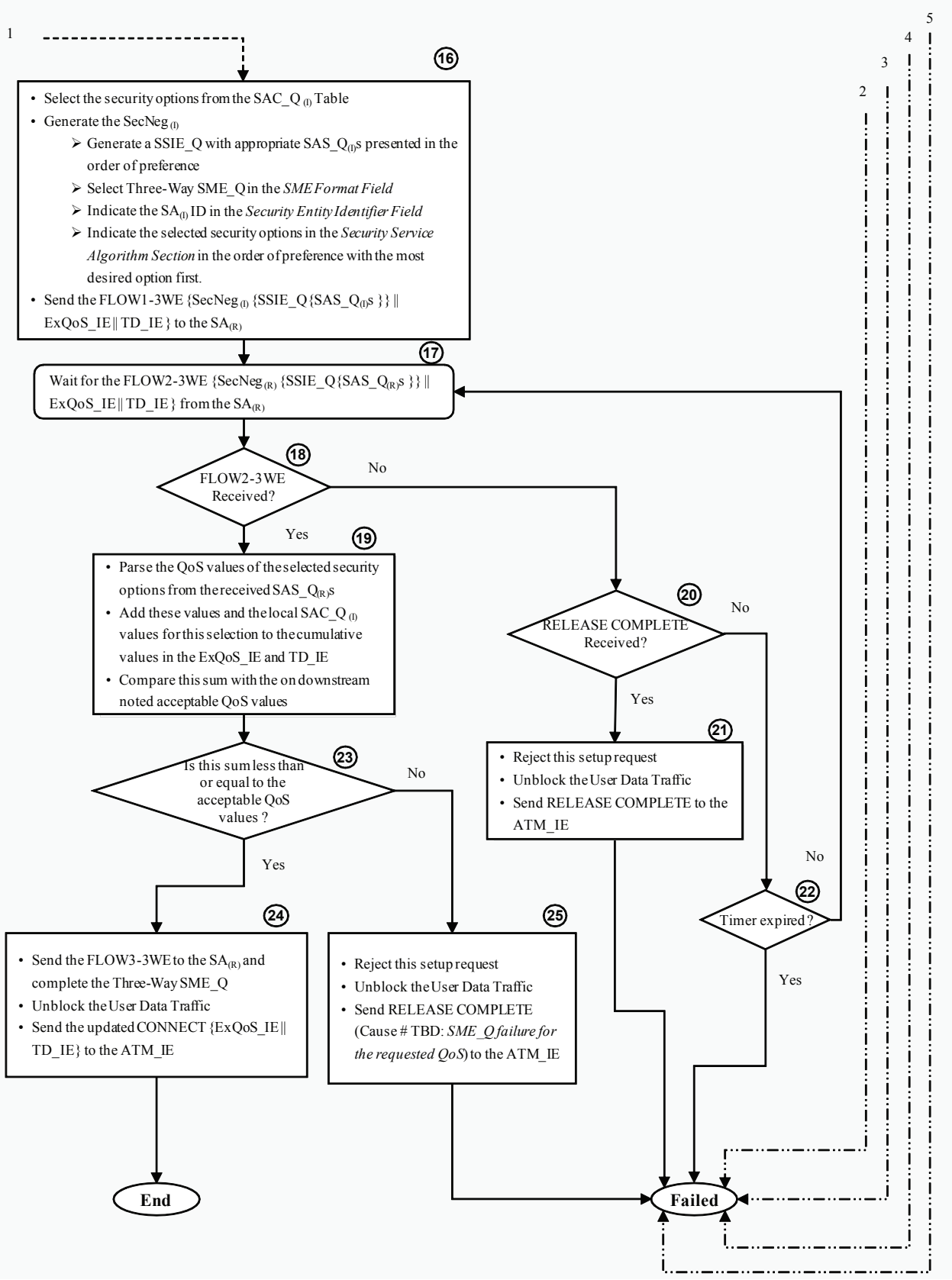


Figure 11.13 The In-Band SME_Q Protocol Procedure for $SA_{(I)}$

Otherwise, the $SA_{(I)}$ forwards the SETUP message toward the responding endpoint, Figure 11.13, Step 7.

Upon receipt of a CONNECT message on the initiating side, the $SA_{(I)}$ blocks the user data traffic and 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 – on the downstream saved – acceptable parameters and the received cumulative values from the responding endpoint, Figure 11.13, Step 10.

The $SA_{(I)}$ decides on a corresponding alternative list of the security algorithms and modes of operation (in case of Confidentiality security service) to include in the FLOW1_3WE for negotiation with the $SA_{(R),}$ Figure 11.13, Step 16. The suggested options are to be selected, so that, the worst option, which isthe 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). Table 11.2 illustrates the nomenclature used. Equations 11.1-11.6 should be satisfied. In this case the $SA_{(I)}$ should first prioritize these services according to the governing security policy. It first selects its 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 chosen selections for the first option. The Equations11.7-11.10 should apply. 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 parameters associated with S2 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 $SAS\_Q_{(I)}s$ as follows:

Three-Way SME_Q is selected in the *SME Format* field. The ID of the initiating SA is included in the *Security Entity Identifier* field. Above selected security service algorithm and mode of operation options are included in the *Security Service Algorithm Section* including the $CDV_{(I)}$ and $CTD_{(I)}$ values for each option. These options are presented in the order of preference listing the

most desired security option for each particular security service first and the least desired last. The *SAS_Q$_{(I)}$s* for the more preferred service are listed first.

The $SA_{(I)}$ starts the In-Band SME_Q with the FLOW1–3WE appending the suggested security options contained in the generated *SAS_Q$_{(I)}$s* with the $SAC\_Q_{(I)}$ values of each option and also the *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element* received in the CONNECT message from the responding endpoint, Figures 11.13, Step16 and 11.12, Step 6. Upon receipt of the FLOW2_3WE from the $SA_{(R)}$, the $SA_{(I)}$ takes the $SAC\_Q_{(R)}$ values of each negotiated security option from the received *SAS_Q$_{(R)}$s* and 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,* Figure 11.13, Step 19. Here, the following equations should apply

$$CDV_{cum}\ [:= CDV_{cum\ (received)} + CDV_{(I)} + CDV_{(R)}\ ] \le CDV_{acp} \qquad (11.15)$$

$$CTD_{cum}\ [:= CTD_{cum\ (received)} + CTD_{(I)} + CTD_{(R)}\ ] \le CTD_{max} \qquad (11.16)$$

If the above equations are satisfied, the $SA_{(I)}$ accepts the chosen options, it completes the In-Band SME_Q by sending the FLOW3_WE to the $SA_{(R)}$, Figures 11.13, Step 24 and 11. 12, Step 8.

The $SA_{(I)}$ now unblocks the user data transfer and forwards the updated CONNECT message with the new cumulative values to the initiating endpoint, Figures 11.13, Step 24 and 11.12, Step 9.

If the above equations were not satisfied, the $SA_{(I)}$ clears the connection with the cause *SME_Q failure for the requested QoS,* Figure 11.13, Step 25.

Responding Security Agent Procedures

The following procedures are proposed in addition to the already existing procedures in Section 5.1.5.1.1 of the *ATM Security Specification Version 1.1* [SEC_11]. For the simplicity and focus on the core subject, the term *Security Agent (SA)* is used in the description of the procedures regardless of its hardware implementation. The term *SA$_{(I)}$* is used for the initiating Security Agent and *SA$_{(R)}$* for the responding Security Agent. Figure 11.14 depicts the In-Band SME_Q protocol procedure of the $SA_{(R)}$. It illustrates the procedural enhancements to the existing In-Band SME protocol

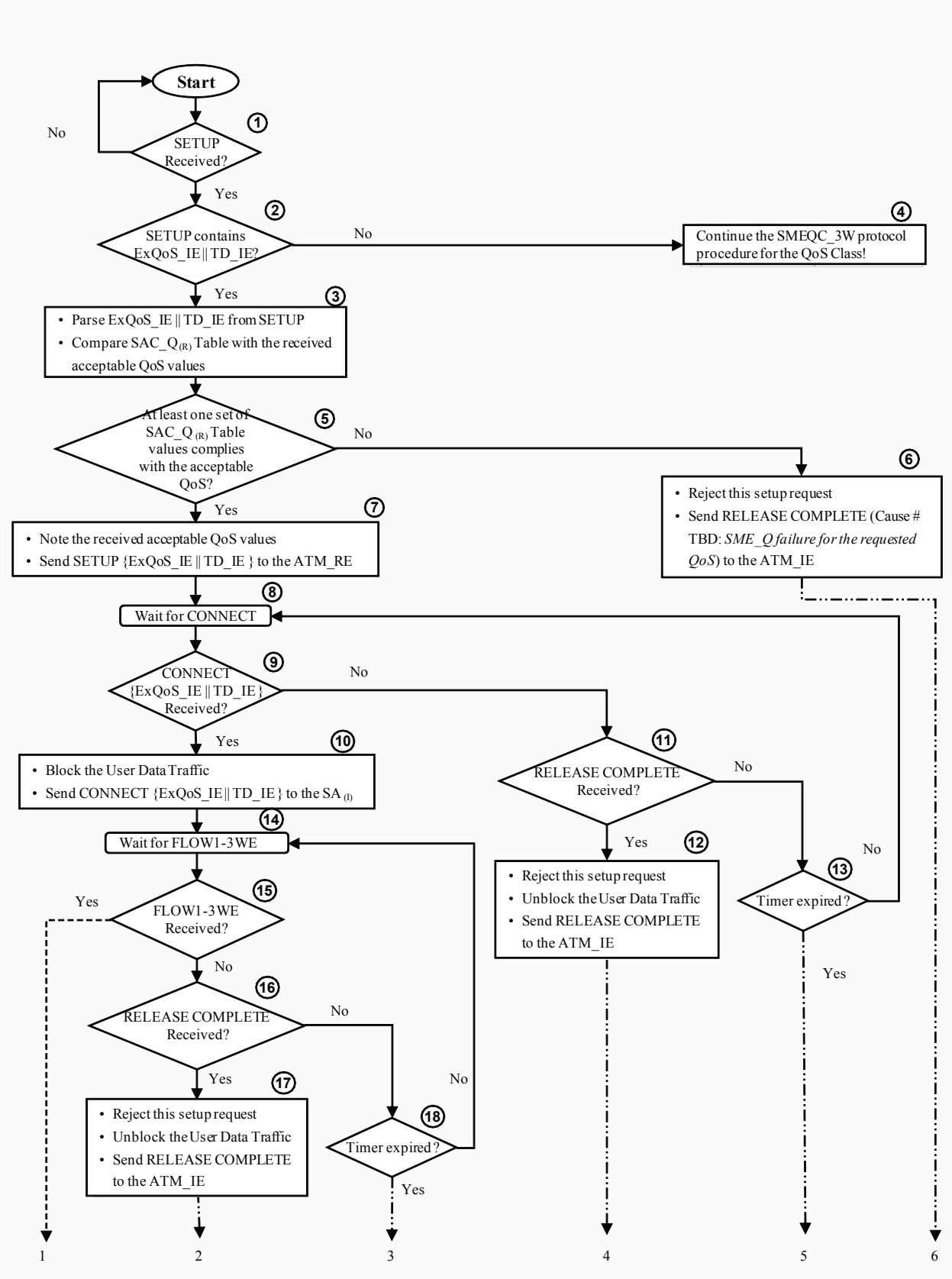


Figure 11.14 The In-Band SME_Q Protocol Procedure for $SA_{(R)}$ (to continue on next page)

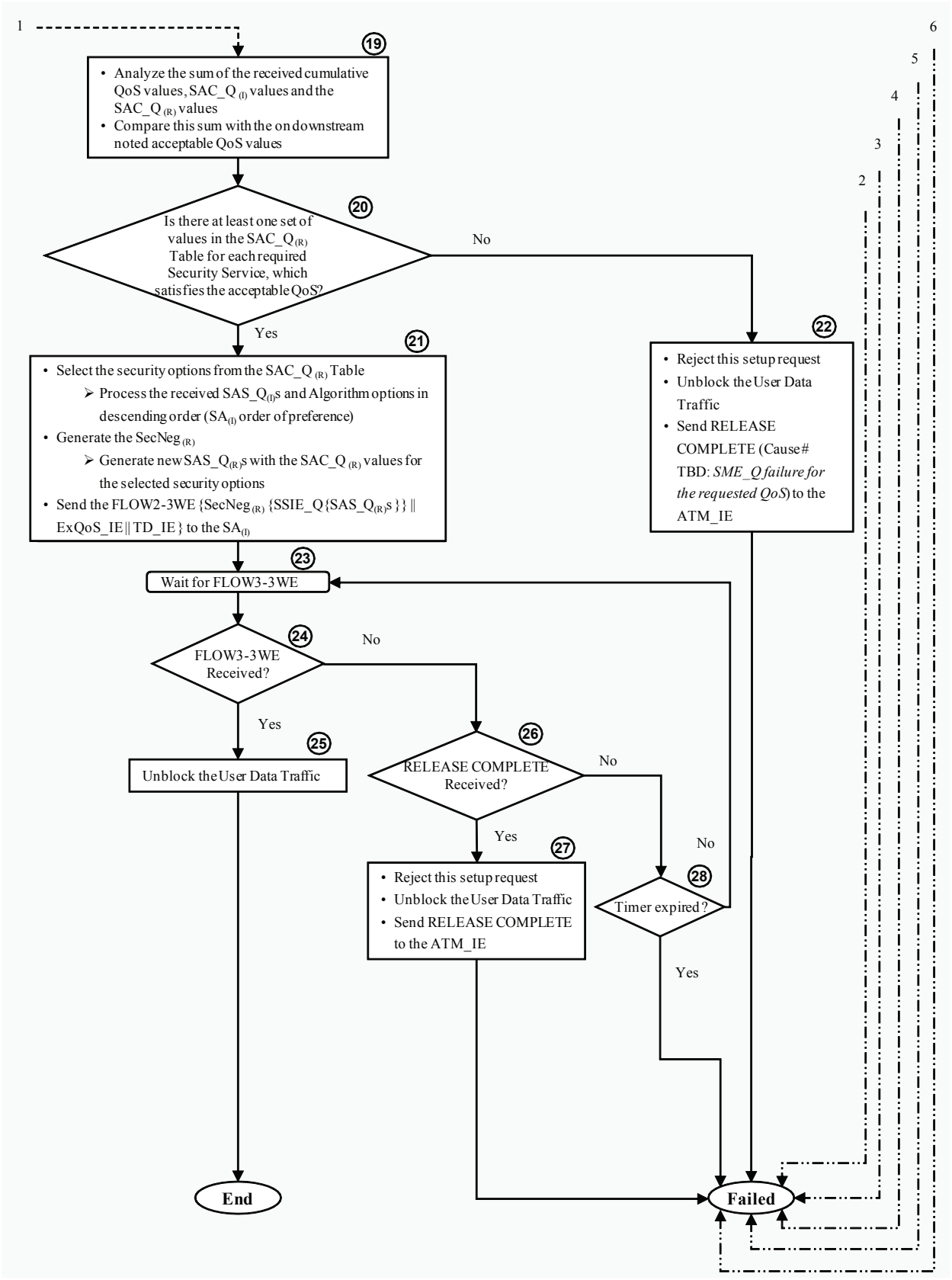


Figure 11.14 The In-Band SME_Q Protocol Procedure for $SA_{(R)}$

of *ATM Security Specification Version 1.1* [SEC_11]. Upon receipt of a SETUP message on the responding side, The $SA_{(R)}$ compares the $SAC\_Q_{(R)}$ values for the security services, which are to be provided, with the indicated acceptable parameters in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element* for that connection, Figure 11.14, Step l. It then notes the requested acceptable Traffic QoS parameter values, Figure 11.14, Step 7. If none of the security options with the $SAC\_Q_{(R)}$ values could comply with the user requested acceptable values, the $SA_{(R)}$ rejects the connection request with the cause *SME_Q failure for the requested QoS,* Figure 11.14, Step 6. Otherwise, the $SA_{(R)}$ forwards the SETUP message toward the responding endpoint, Figures 11.14, Step 7and 11.12, Step l. Upon receipt of a CONNECT message on the responding side, the $SA_{(R)}$ blocks the user data transfer and forwards the CONNECT message toward the initiating endpoint, Figures 11.14, Step 10 and 6.12, Step 5. It then awaits the invocation of the SME_Q from the initiating side over the user connection, Figure 11.14, Step 14. Upon receipt of the FLOW1–3WE from the initiating partner $SA_{(I),}$ the $SA_{(R)}$ decides which security options to accept. FLOW1–3WE includes the $SA_{(I)}$ suggested negotiable security options containing the corresponding $SAC\_Q_{(I)}$ Traffic QoS degradation values of the options and 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 Traffic QoS parameters and $SAC\_Q_{(I)}$ degradation values of suggested options with the previously noted acceptable Traffic QoS parameters values of this connection, Figure 11.14, Step 19.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 $SA_{(I)}$ for that option, it would still meet or result to lower rates than the saved acceptable parameters on the downstream for this connection, Figure 11.14, Step 21. 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 by the initiating endpoint.

If more than one service are requested to be supported, 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). Table 11.3 illustrates the nomenclature used. Equations 11.1-11.4 Should be satisfied for the responding side, in addition to

$$CDV_{cum} + CDV_{(I)} + CDV_{(R)} < CDV_{acp} \quad (11.17)$$

$$CTD_{cum} + CTD_{(I)} + CTD_{(R)} < CTD_{max} \quad (11.18)$$

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

$$CDV_{(R),S2} < CDV_{acp} - CDV_{cum} - CDV_{(I)} - CDV_{(R),S1} \quad (11.19)$$

$$CTD_{(R),S2} < CTD_{max} - CTD_{cum} - CTD_{(I)} - CTD_{(R),S1} \quad (11.20)$$

and of course

$$CDV_{(R),S1} < CDV_{acp} - CDV_{cum} - CDV_{(I)} - CDV_{(R),S2} \quad (11.21)$$

$$CTD_{(R),S1} < CTD_{max} - CTD_{cum} - CTD_{(I)} - CTD_{(R),S2} \quad (11.22)$$

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 parameters associated with S2 are equal to zero. This would be a special case of the above equations. The $SA_{(R)}$ then generates 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 FLOW2_3WE, Figures 11.14, Step 21 and 11.12, Step 7. Upon completion of the In-Band SME_Q the $SA_{(R)}$ unblocks the user data transfer, Figure 11.14, Step 25. If the $SA_{(R)}$ could not support the options according to the Traffic QoS requirements the $SA_{(R)}$ should clear the connection with the cause *SME_Q failure for the requested QoS,* Figure 11.14, Step 22.

11.3.1 Connections with Nesting and Multiple Security Associations

The In-Band SME_Q also supports the different topologies of security associations: nesting (as well as the one point overlap) and no overlap. For each of these different topologies there are always two sides for each security association: the initiating side and the responding side. Further more, according to the *ATM Security Specification Version 1.1* [SEC_11] the nesting level is limited to 16 security associations within an ATM connection path. Figure 11.15 illustrates the negotiation of security options for the In-Band SME_Q in the case of nesting security associations.

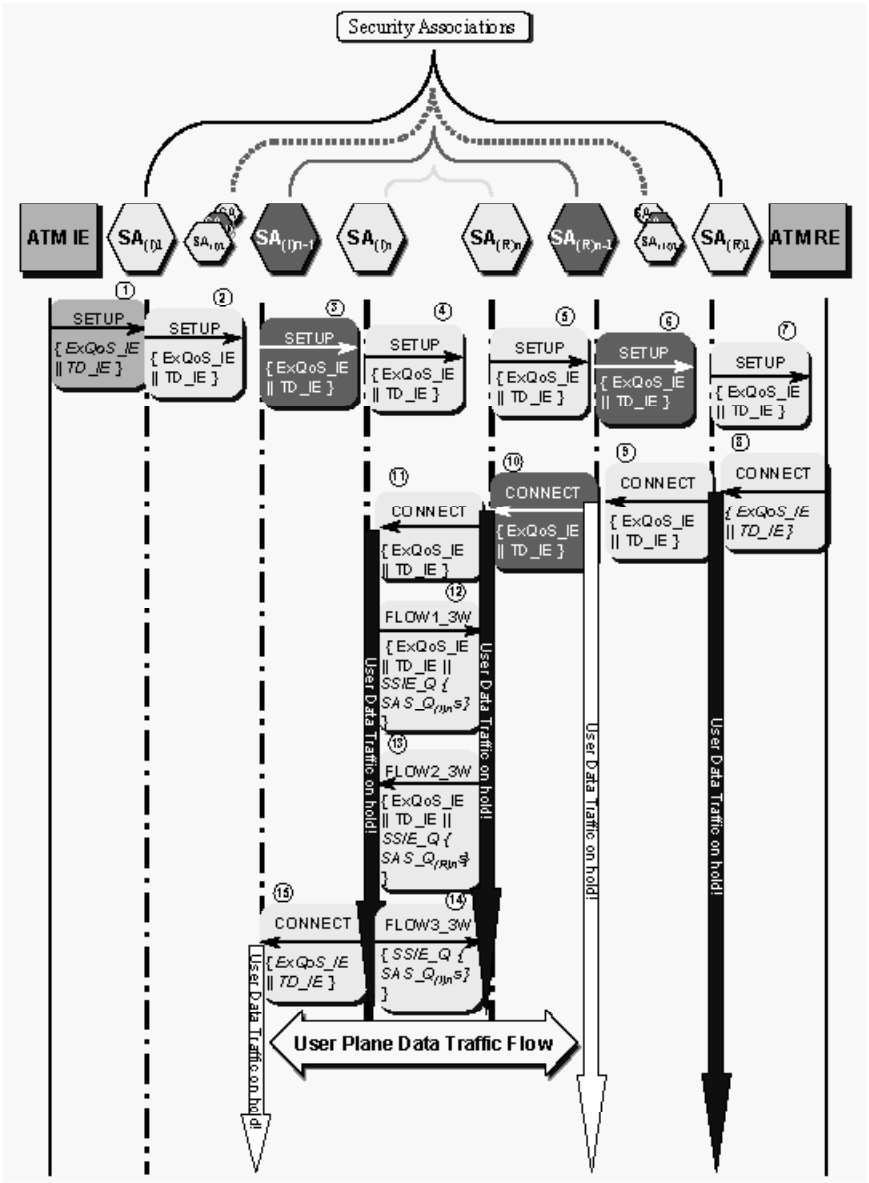


Figure 11.15 In-Band SME_Q with Nesting Security Associations (to continue next page)

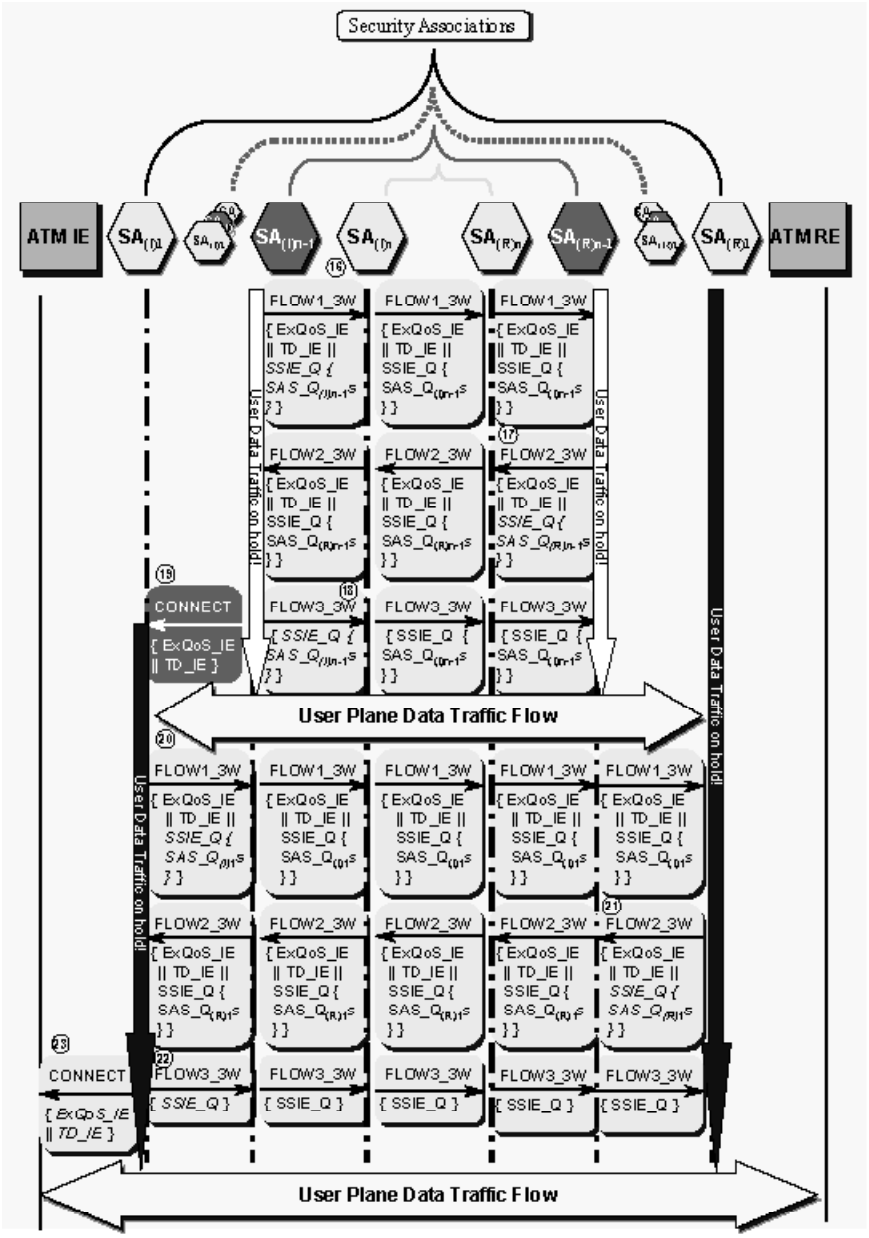


Figure 11.15 In-Band SME_Q with Nesting Security Associations

Figure 11.16 illustrates these negotiations in the case of sequenced security associations. The following describes the procedures for these SAs.

Initiating Security Agents Procedures

The following procedures are proposed in addition to the already existing procedures in Section 6.1.5.1.1 of the *ATM Security Specification Version 1.1* [SEC_11]. For the simplicity and focus on the core subject, the term *Security Agent (SA)* is used in the description of the procedures regardless

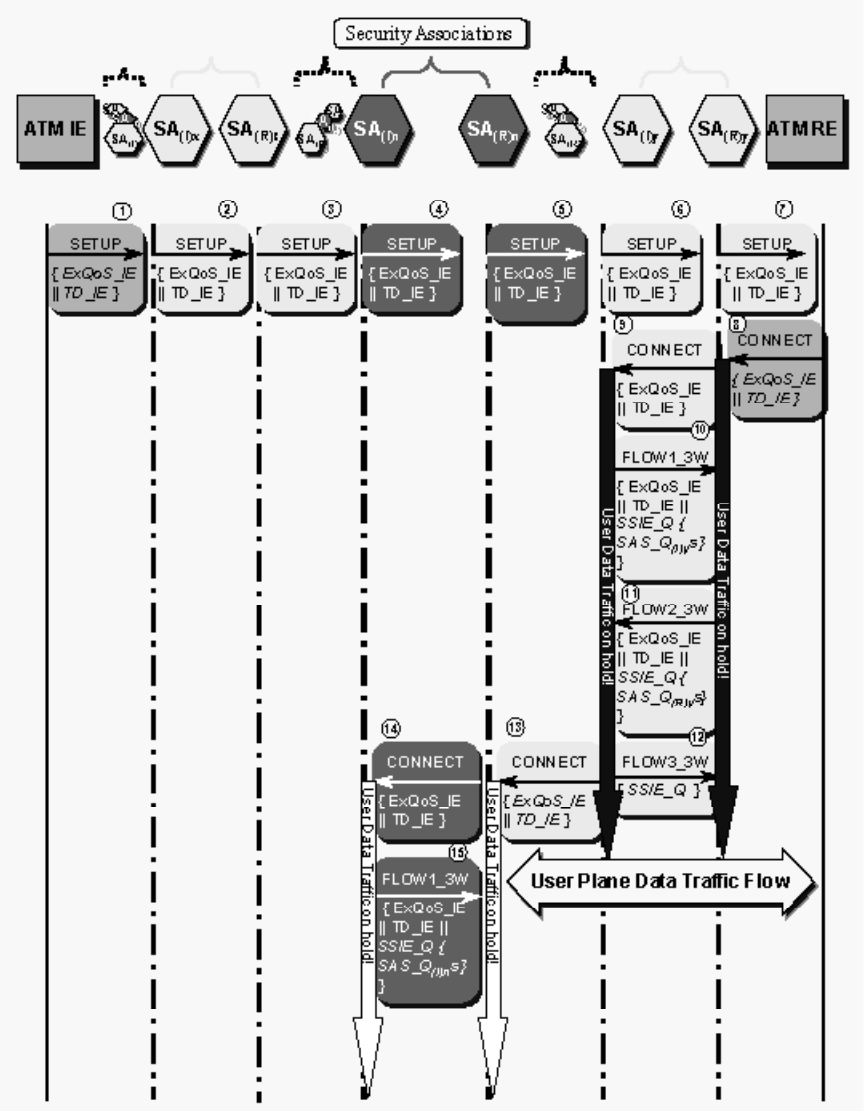


Figure 11.16 In-Band SME_Q with Sequenced Security Associations (to continue next page)

of its hardware implementation. The term $SA_{(I)}$ is used for the initiating Security Agent and $SA_{(R)}$ for the responding Security Agent. Figure 11.13 depicts the In-Band SME_Q protocol procedure for each $SA_{(I)}$ along the path ($SA_{(I)1}$to $SA_{(I)n}$).

Nesting Security Associations

Upon receipt of a SETUP message on the initiating side of a connection with nesting security associations, Figure 11.15, the $SA_{(I)1}$ compares the $SAC\_Q_{(I)1}$ values for the security services, which are to be provided with the indicated acceptable parameters, Figure 11.13, Step l. It notes the requested acceptable Traffic QoS parameter values indicated in the *Extended QoS Parameter*

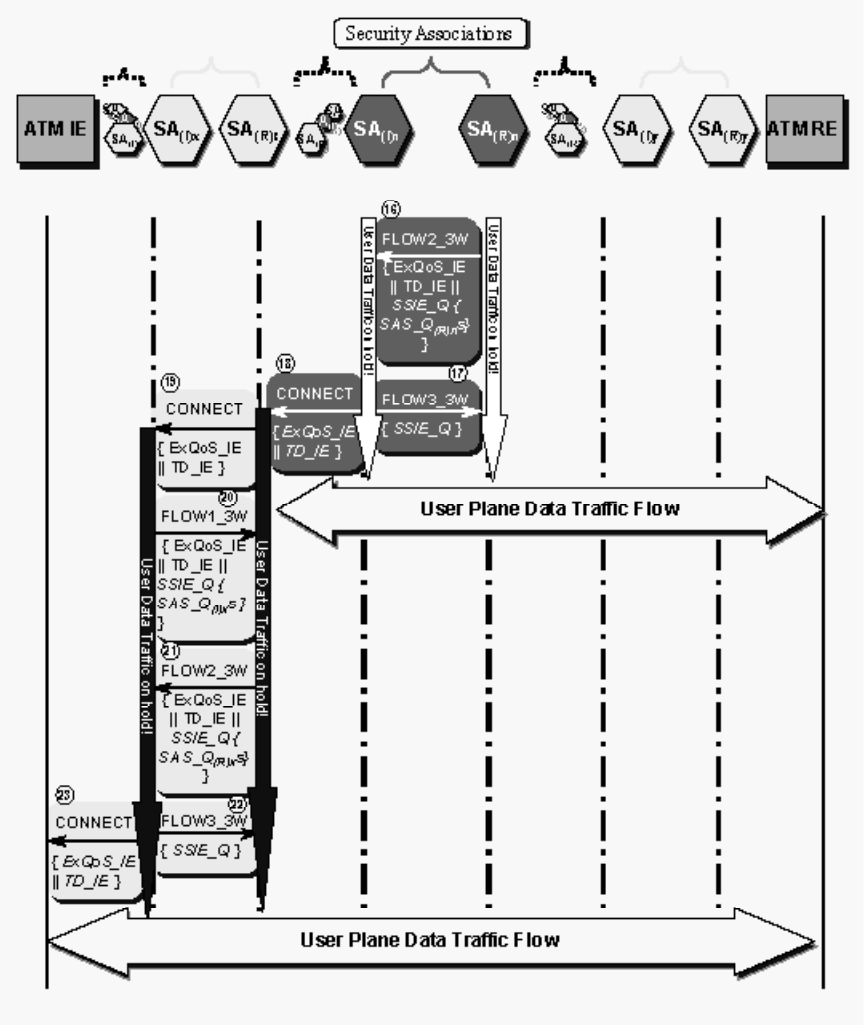


Figure 11.16 In-Band SME_Q with Sequenced Security Associations

Information Element and the *End-to-End Transit Delay Information Element* for that connection Figure 11.13, Step 7.

If none of the security options in the table complies with the acceptable values, the $SA_{(I)1}$ rejects the connection request with the cause *SME_Q failure for the requested QoS*, Figure 11.13, Step 6. Otherwise, the $SA_{(I)1}$ forwards the SETUP message toward the responding endpoint, Figures 11.13;7 and 11.15, Step 2.

Upon receipt of a CONNECT message on the initiating side of a connection with nesting security associations, Figure 11.15, the $SA_{(I)n}$ – the first $SA_{(I)}$ to receive this message – blocks the user data traffic and prevents the CONNECT message to proceed to the initiating endpoint. It analyzes the

cumulative Traffic QoS parameters indicated in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element*. It compares the $SAC\_Q_{(I)n}$ values for the security services, which are to be provided with the – on the downstream noted – acceptable parameters and the received cumulative values from the responding endpoint, Figure 11.13, Step 10. The $SA_{(I)n}$ decides on a corresponding alternative list of the security algorithms and modes of operation to include in the FLOW1_3WE for negotiation with the $SA_{(R)n,}$ Figure 11.13, Step 16. 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 $SAC\_Q_{(I)n}$ values, if added to the received cumulative parameters, 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 noted acceptable CLR indicated originally by the initiating endpoint.

If more than one service are requested to be supported between $SA_{(I)n}$ and $SA_{(R)n}$, 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). Table 11.4 illustrates the nomenclature used. Equations 11.1-11.6 should be satisfied for $SA_{(I)n}$. In this case, the $SA_{(I)n}$ should first prioritize these services. It first selects its 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 chosen selections for the first option. Equations 11.7-11.10 should be satisfied for $SA_{(I)n}$. 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 service is supported, the parameters associated with S2 are equal to zero. This would be a special case of the above equations.

After the selection of the security options, the $SA_{(I)n}$ generates the appropriate $SAS\_Q_{(I)}s$ as follows, Figure 11.13, Step 16, Three–way SME_Q is selected in the *SME Format* field, the ID of the initiating $SA_{(I)n}$ is included in the *Security Entity Identifier* field, above selected security service algorithm and mode options are included in the *Security Service Algorithm Section* including the $CDV_{(I)n}$ and $CTD_{(I)n}$ values for each option. These options are presented in the order of preference

listing the most desired security option for each particular security service first and the least desired last.The $SAS\_Q_{(I)n}$ for the more preferred service are listed first.

The $SA_{(I)n}$ starts the In-Band SME_Q with the FLOW1–3WE appending the generated $SAS\_Q_{(I)n}s$ with the $SAC\_Q_{(I)n}$ values of each option and also the *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element* received in the CONNECT message from the responding endpoint, Figures 11.13, Step 16 and 11.15, Step 12.

Upon receipt of the FLOW2_3WE from the $SA_{(R)n}$ of a connection with nesting security associations, Figure 11.15, the $SA_{(I)n}$ takes the $SAC\_Q_{(R)n}$ values of each negotiated security option from the $SAS\_Q_{(R)n}s$ and adds them along with the local $SAC\_Q_{(I)n}$ values to the cumulative parameters of the *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element,* Figure 11.13; 19. The following should be satisfied

$$CDV_{cum}\ [:= CDV_{cum\ (received)} + CDV_{(I)n} + CDV_{(R)n}\ ] < CDV_{acp} \tag{11.23}$$

$$CTD_{cum}\ [:= CTD_{cum\ (received)} + CTD_{(I)n} + CTD_{(R)n}\ ] < CTD_{max} \tag{11.24}$$

If the above equations are satisfied and the $SA_{(I)n}$ accepts the chosen options, it completes the In-Band SME_Q by sending the FLOW3_WE to the $SA_{(R)n}$, Figure 11.13, Step 24 and 11.15, Step 14). The $SA_{(I)n}$ now forwards the updated CONNECT message with the new cumulative values toward the initiating endpoint. It then unblocks the user data transfer, Figures 11.13, Step 24 and 11.15, Step 15.

The next $SA_{(I)}$ on the path of a connection with nesting security associations, Figure 11.15 toward the initiating endpoint, is the $SA_{(I)n-1}$. This SA negotiates its options in the same way described above with its peer $SA_{(R)n-1}$, Figure 11.15, Step 16 and17). After completing the In-Band SME_Q, it then updates the cumulative parameters of the *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element,* Figure 11.15, Step 18. Equations 11.23 and 11.24 should be satisfied for the (n-1)th SA pair.

It then forwards the updated CONNECT message toward the initiating endpoint and the user data transfer is unblocked, Figure 11.15, Step 19. The last $SA_{(I)}$ on the path of a connection with

nesting security associations, Figure 11.14, is the $SA_{(I)1}$, which also proceeds the same way but this time its cumulative values can also be equal to the accepted parameters, as stated in Equations 11.15and11.16 for $SA_{(I)1}$ and $SA_{(R)1.}$ The updated CONNECT message is forwarded to the initiating endpoint and the user data transfer is unblocked, Figure 11.15, Step 23. If the above equations were not satisfied the $SA_{(I)}$ clears the connection with the cause *SME_Q failure for the requested QoS,* Figure 11.13, Step 25.

Sequenced Security Associations

In the case of sequenced security Associations, Figure 11.15, upon receipt of a SETUP message, the $SA_{(I)}$s compare their $SAC\_Q_{(I)}$ values for the security services, which are to be provided, with the indicated acceptable parameters, Figure 11.13, Step l. They note the requested acceptable Traffic QoS parameter values indicated in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element* for that connection, Figure 11.13, Step 7.

If none of the security options in the table complies with the acceptable values, the $SA_{(I)}$s reject the connection request with the cause *SME_Q failure for the requested QoS*, Figure 11.13, Step 6. Otherwise, the $SA_{(I)}$s forward the SETUP message toward the responding endpoint, Figure 11.13, Step 7. Upon receipt of a CONNECT message from the responding endpoint, the $SA_{(I)y}$ proceeds according to the above described procedures, negotiates the security algorithms and modes of operation and updates the cumulative parameters of the *Extended QoS Parameters Information Element* and the *End-to-End Transit Delay Information Element,* Figure 11.16, Steps 9,10,11, and12). Equations 11.23 and 11.24 should be satisfied for the yth SA pair. It then forwards the updated CONNECT message toward the initiating endpoint and unblocks the user data transfer, Figure 11.16, Step 14.

The last $SA_{(I)}$ on the path of a connection with sequenced security associations, Figure 11.16, is the $SA_{(I)1}$ which updates the cumulative values for the last time, unblocks the user data traffic and forwards the updated Traffic QoS parameters toward the initiating endpoint, Figure 11.16, Step 23.

Responding Security Agents Procedures

The following procedures are proposed in addition to the already existing procedures in Section 5.1.5.1.1 of the *ATM Security Specification Version 1.1* [SEC_11].

For the simplicity and focuss on the core subject, the term *Security Agent (SA)* is used in the description of the procedures regardless of its hardware implementation. The term $SA_{(I)}$ is used for the initiating Security Agent and $SA_{(R)}$ for the responding Security Agent. Figure 11.14 depicts the In-Band SME_Q protocol procedure for each $SA_{(R)}$ along the path ($SA_{(R)n}$to $SA_{(R)1}$).

Nesting Security Associations

Upon receipt of a SETUP message on the responding side, the $SA_{(R)n}$ compares the $SAC\_Q_{(R)n}$ values for the security services, which are to be provided, with the indicated acceptable parameters in the *Extended QoS Parameter Information Element* and the *End-to-End Transit Delay Information Element* for that connection, Figure 11.14, Step l. It then notes the requested acceptable Traffic QoS parameter values, Figure 11.14, Step 7.

If none of the security options with the $SAC\_Q_{(R)n}$ values could comply with the user requested acceptable values, the $SA_{(R)n}$ rejects the connection request with the cause *SME_Q failure for the requested QoS,* Figure 11.14, Step 6. Otherwise, the $SA_{(R)n}$ forwards the SETUP message toward the responding endpoint, Figures 11.14, Step 7and 11.15, Step 5.

Upon receipt of a CONNECT message on the responding side, the $SA_{(R)1}$ blocks the user data transfer and forwards the CONNECT message toward the initiating endpoint, Figures 11.15, Step 10 and 11.16, Step 9. It then awaits the invocation of the SME_Q from the initiating side over the user connection, Figure 11.14, Step 14. Each $SA_{(R)}$ on the path toward the initiating endpoint proceeds the same way, Figure 11.15, Step 10,11. The first $SA_{(R)}$, which receives the FLOW1–3WE on the path of a connection with nesting security associations, Figure 11.15, is the last one forwarding the CONNECT message toward the initiating endpoint. Upon receipt of the FLOW1–3WE, the $SA_{(R)n}$ decides which security options to accept. FLOW1–3WE includes the $SA_{(I)n}$ suggested negotiable security options in the *SAC_Q Information Element*$_{(I)n}$ containing the corresponding $SAS\_Q_{(I)n}$ Traffic QoS degradation values of these options and the received *Extended QoS Parameter Information*

Element and the *End-to-End Transit Delay Information Element* in the CONNECT message from the responding endpoint.

The $SA_{(R)n}$ compares its local $SAS\_Q_{(R)n}$ values for the security services, which are to be provided, in addition to the sum of the received cumulative Traffic QoS parameters and $SAS\_Q_{(I)n}$ degradation values of suggested options with the previously noted acceptable Traffic QoS parameters values of this connection, Figure 11.14, Step 19.

The $SA_{(R)n}$ 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 $SA_{(I)n}$ for that option (derived from the $SAS\_Q_{(I)n}$), it would still result to lower rates than the saved acceptable parameters on the downstream for this connection, Figure 11.14, Step 21. 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 by the initiating endpoint. If more than one service are requested to be supported, 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). Table 11.4 illustrates the nomenclature used. Equations 11.1-11.6 should be satisfied for $SA_{(R)n}$and $SA_{(I)n}$.

In this case, the $SA_{(R)n}$ should process the received $SAS\_Q_{(I)n}s$ in descending order, which is the order of preference of $SA_{(I)n}$ 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 Equations 11.7-11.10 for $SA_{(R)n}$and $SA_{(I)n}$.

The $SA_{(R)n}$ processes the algorithm options also in descending order, which is the order of preference of $SA_{(I)n}$. In case only one service is supported, the parameters associated with S2 are equal to zero. This would be a special case of the above equations. The $SA_{(R)n}$ then generates new $SAS\_Q_{(R)n}s$ and indicates its $SAC\_Q_{(R)n}$ values for the security options chosen. The selected option for each security service is communicated to the $SA_{(I)n}$ in the FLOW2_3WE, Figures 11.14, Step 21 and 11.15, Step 13. Upon completion of the In-Band SME_Q the $SA_{(R)n}$ unblocks the user data transfer, Figure 11.14, Step 25. If the $SA_{(R)n}$ could not support the options according to the Traffic QoS requirements the

$SA_{(R)n}$ should clear the connection with the cause *SME_Q failure for the requested QoS,* Figure 11.14, Step 22. The next $SA_{(R)}$ on the path of a connection with nesting security associations, Figure 11.15 toward the initiating endpoint is the $SA_{(R)n-1}$. This SA negotiates its options in the same way described above with its peer $SA_{(I)n-1}$. The selected option for each security service is communicated to the $SA_{(I)n-1}$ in the FLOW2_3WE, Figure 11.14, Step 17. Upon completion of the In-Band SME_Q the $SA_{(R)n-1}$ unblocks the user data transfer, Figure 11.14, Step 25.

The last $SA_{(R)}$ on the path is the $SA_{(R)1}$, which also proceeds the same way, but this time its decision for selecting an option is based on cumulative values, which can also be *equal to* the accepted parameters:

$$CDV_{(R)1} \leq CDV_{acp} - CDV_{cum} - CDV_{(I)1} \quad (11.25)$$

$$CTD_{(R)1} \leq CTD_{max} - CTD_{cum} - CTD_{(I)1} \quad (11.26)$$

$$CLR_{(R)1,S1} \leq CLR_{acp} \quad (11.27)$$

$$CLR_{(R)1,S2} \leq CLR_{acp} \quad (11.28)$$

Upon completion of the In-Band SME_Q the $SA_{(R)1}$ unblocks the user data transfer. If any $SA_{(R)}$ could not support the options according to the Traffic QoS requirements the $SA_{(R)}$ should clear the connection with the cause *SME_Q failure for the requested QoS,* Figure 11.14, Step 22.

Sequenced Security Associations

In the case of sequenced security associations, Figure 11.16, the first responding SA is the $SA_{(R)y}$, which follows the same procedures described above, Figure 11.16, Steps 9,10,11, and12. After completion of the negotiations and receipt of the FLOW3_3W, it unblocks the user data traffic. In this case, the security negotiations are processed, completed and the Traffic QoS parameters are updated before the next set of SAs could follow these procedures again.

Chapter

12

SMEQ Prototype Simulation – SMEQSIM

SMEQ SIMULATOR (SMEQSIM) is a prototype of the proposed SME_Q protocol described in Chapter 11, which is developed to simulate the Traffic QoS provisioning procedures for different possible security architectures. The software application is designed as a Graphical User Interface (GUI), in particular a Single Document Interface (SDI), based on Windows Operating System (Win32). SMEQSIM is written in C++ object oriented programming language based on Microsoft Foundation Classes (MFC). Figure 8.1 depicts the SMEQ SIMULATOR.

The tabbed property pages provide the user with a user-friendly graphical interface to enter the simulation data for different network elements along an ATM connection path according to a designated simulation scenario. Due to the complexity of some simulation scenarios and the necessity for a large data input, the SMEQSIM is designed to also accept input data in form of data files. The simulator is further designed to save the last applied simulation data and display it for the next simulation for the purpose of regression testing. Entering new data or changing selections are applied each time by pressing the *Apply* button. After entering all data and pressing *Apply*, the *OK* button is pressed and the *SIMPROC* screen appears. On this screen, the actual simulation is run. Pressing *Cancel* terminates the SMEQ SIMULATOR.

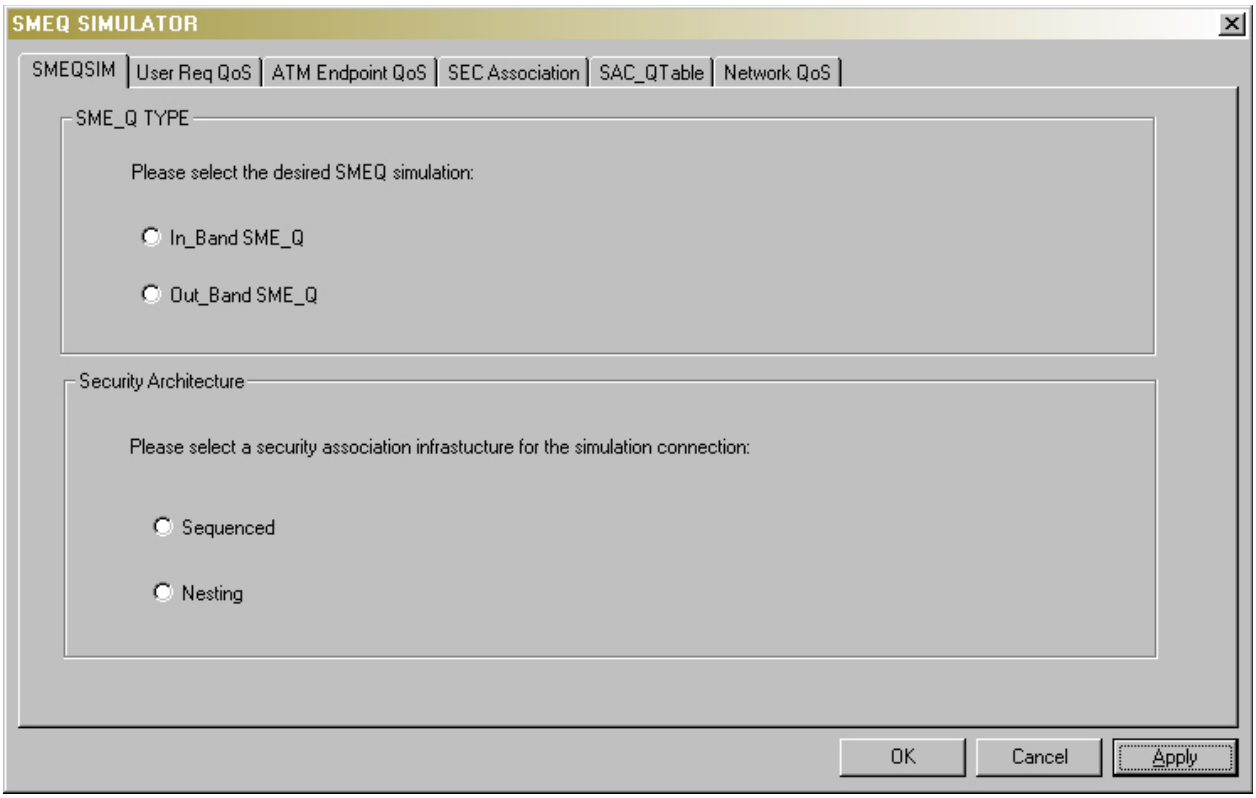


Figure 12.1 The SMEQ SIMULATOR - SMEQSIM

The SMEQ SIMULATOR simulates the ATM connection establishment phase with the Traffic QoS negotiations according to the entered values for the chosen security architecture. The simulation follows the procedures and equations described in detail in Chapter 11. The simulation results and the log of the transactions for each network element are prompted step by step through the course of the ATM connection in the scroll list view on the last simulation screen.

The SMEQ SIMULATOR's design and implementation procedures are described in details for the In-Band SMEQ for two prototype cases of Sequenced and Nesting security architectures. The Out-Band simulation design is identical and follows the procedures described in Chapter 11.

12.1 In-Band SME_Q Prototype Simulation

A prototype of the proposed In-Band SME_Q protocol described in Chapter 11 is developed to simulate the Traffic QoS provisioning procedures for different possible security architectures. Figure 11.7 of the previous chapter illustrates the In-Band SME_Q protocol in the simplest possible

network architecture for a secure ATM connection, where only one pair of SAs are involved in the establishment of security associations. The prototype simulation network in this case consists of only one pair of Security Agents (SA_I, SA_R), two ATM devices, the Initiating Endpoint (ATM_{IE}) and the Responding Endpoint (ATM_{RE}) and the intervening network (NTWK).

The simulation follows the procedures and equations described in detail in the previous chapter. Figures 11.13 and 11.14 summarize these protocol procedures and the program flow for the Initiating SA and Responding SA respectively.

Figure 12.2 depicts the SMEQSIM application workflow for the In-Band SME_Q prototype. It illustrates the required input data and the resulted output for a desired simulation test case.

A simulation scenario is to be designed prior to a simulation process. The network elements and desired simulation security associations along with all required Traffic QoS values for this connection are to be available for input in the SMEQSIM.

The simulation data for different network elements along an ATM connection are entered through a user-friendly graphical interface in form of tabbed property pages for a designated simulation scenario. Due to the complexity of some simulation scenarios and the necessity for a large data input, the SMEQSIM is designed to also accept input data in form of data files. The simulator is further designed to save the last applied simulation data and display it for the next simulation for the purpose of regression testing. The *Apply* button needs to be pressed each time prior to the OK button and after entering new data to apply the changed selections. At the end of the data entry and after pressing *Apply*, the *OK* button is pressed. The *SIMPROC* screen appears, where the actual simulation is run. Pressing *Cancel* terminates the SMEQ SIMULATOR.

The SMEQ SIMULATOR simulates the In-Band ATM connection establishment phase with the Traffic QoS negotiations according to the entered values for the chosen security architecture (Nesting or Sequenced). The simulation results and the log of the transactions for each network element are prompted step by step through the course of the ATM connection in the scroll list view on the last simulation screen.

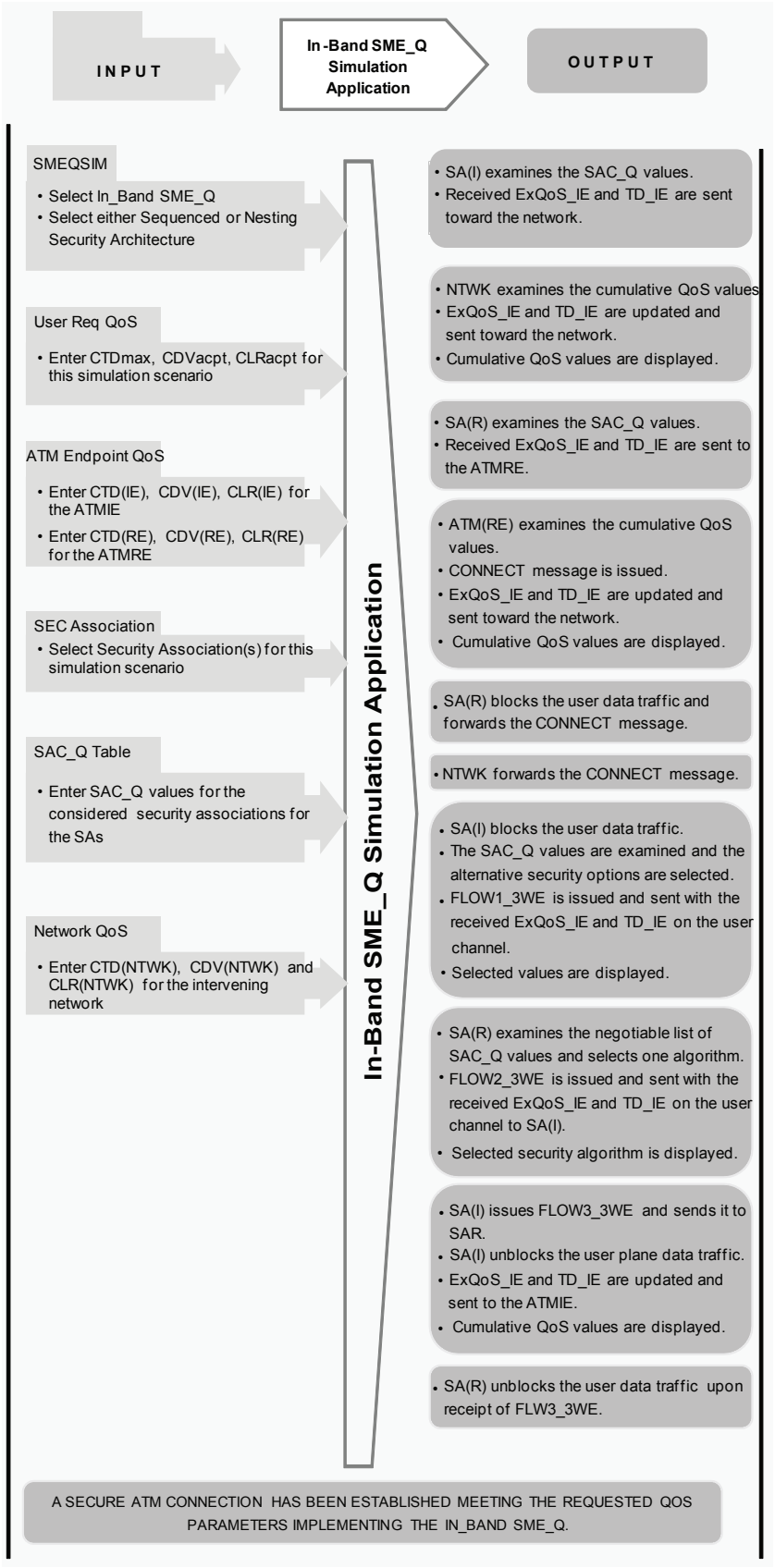


Figure 12.2 In-Band SME_Q Simulation Work Flow Diagram

The log of the simulation transactions and results for each network element is listed step by step through the course of the ATM connection on the last screen (*SIMPROC*) of the SMEQSIM. In case of a negative test scenario where the selected simulation Traffic QoS parameters should not be or are not met or exceeded by the network, the simulation is interrupted and terminated by an Error Message Box. The setup rejection cause is also commented in the simulation results log screen.

To start the In-Band SMEQ simulation process *Simulation* then *Configure* is selected on the menu of the main application window as depicted in Figure 12.3.

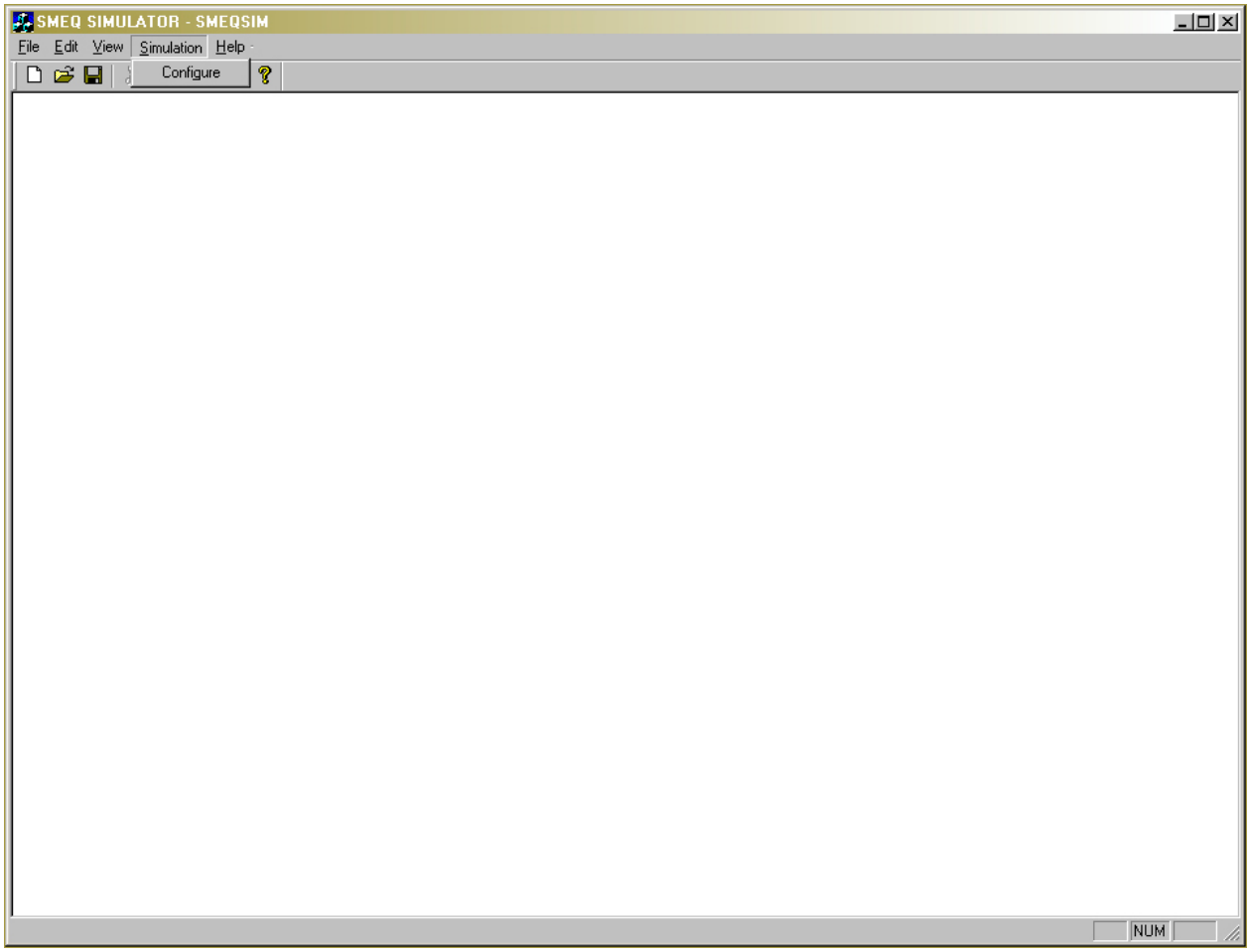


Figure 12.3 The Main Application Window of the SMEQSIM

The SMEQ SIMULATOR property sheet is displayed as depictd in Figure 12.4. In the SME_Q Type section of the SMEQSIM tab – first page – of the SMEQ SIMULATOR property sheet *In-Band SME_Q* is selected.

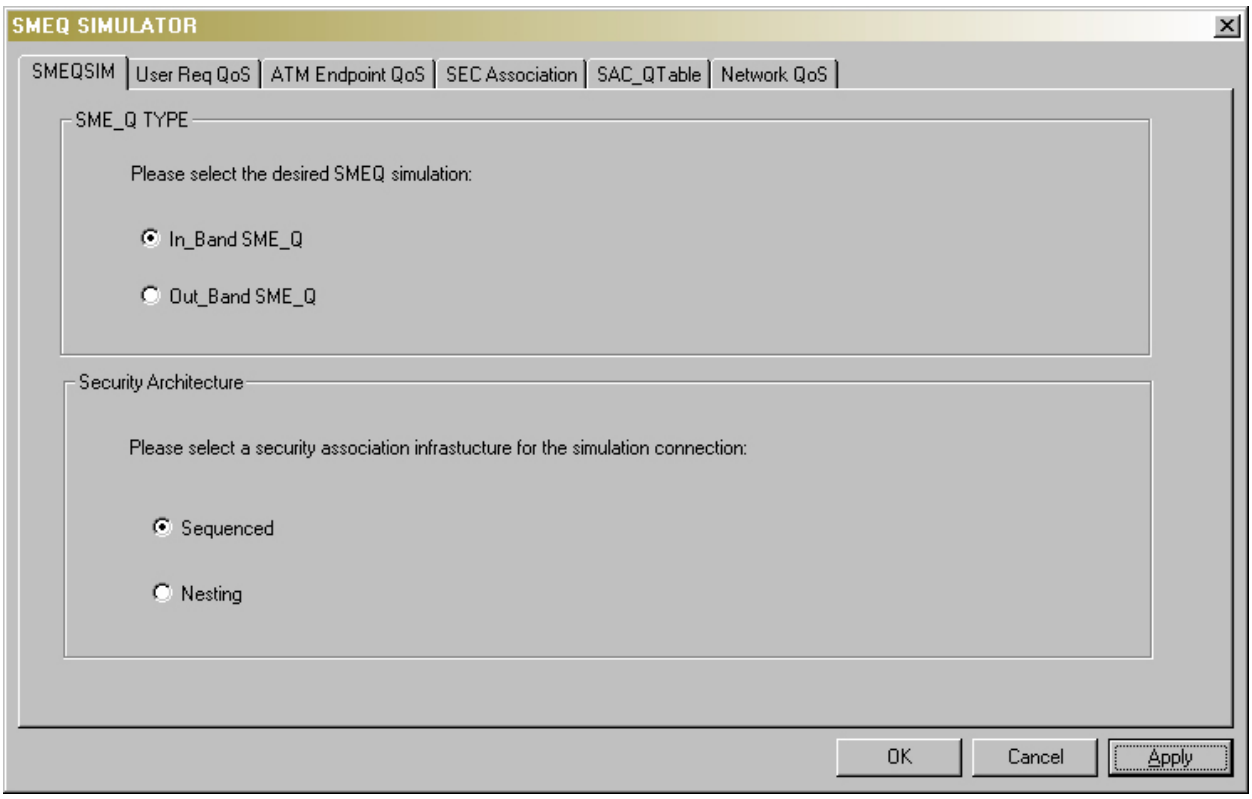


Figure 12.4 *SMEQSIM* **Page of the SMEQSIM**

For the simplest case of a secure In-Band ATM connection, the user can select either security architectures (*Sequenced* or *Nesting*) and choose to enter simulation data for only one pair of SA as depicted in Figure 12.5. After entering new choices the *Apply* button is enabled. The Apply button is pressed and the changes are accepted.

On the *User Req QoS* tab in Figure 12.5, the requested Traffic QoS parameters, maximum Cell Delay Variation, maximum acceptable Cell Delay Variation and the maximum acceptable Cell Loss Ratio (CTD (max), CDV (acpt) and CLR (acpt)) for this particular secure ATM connection simulation scenario are entered.

On the *ATM Endpoint QoS* tab as depicted in Figure 12.6, the Traffic QoS parameters for both of the ATM Endpoints, the ATM Initiating Endpoint (IE) and the Responding Endpoint (RE), are entered. The values for the ATM_{IE} are entered in the *CTD (IE)*, *CDV (IE)* and *CLR (IE)* fields. The values for ATM_{RE} are entered in the *CTD (RE)*, *CDV (RE)* and *CLR (RE)* fields.

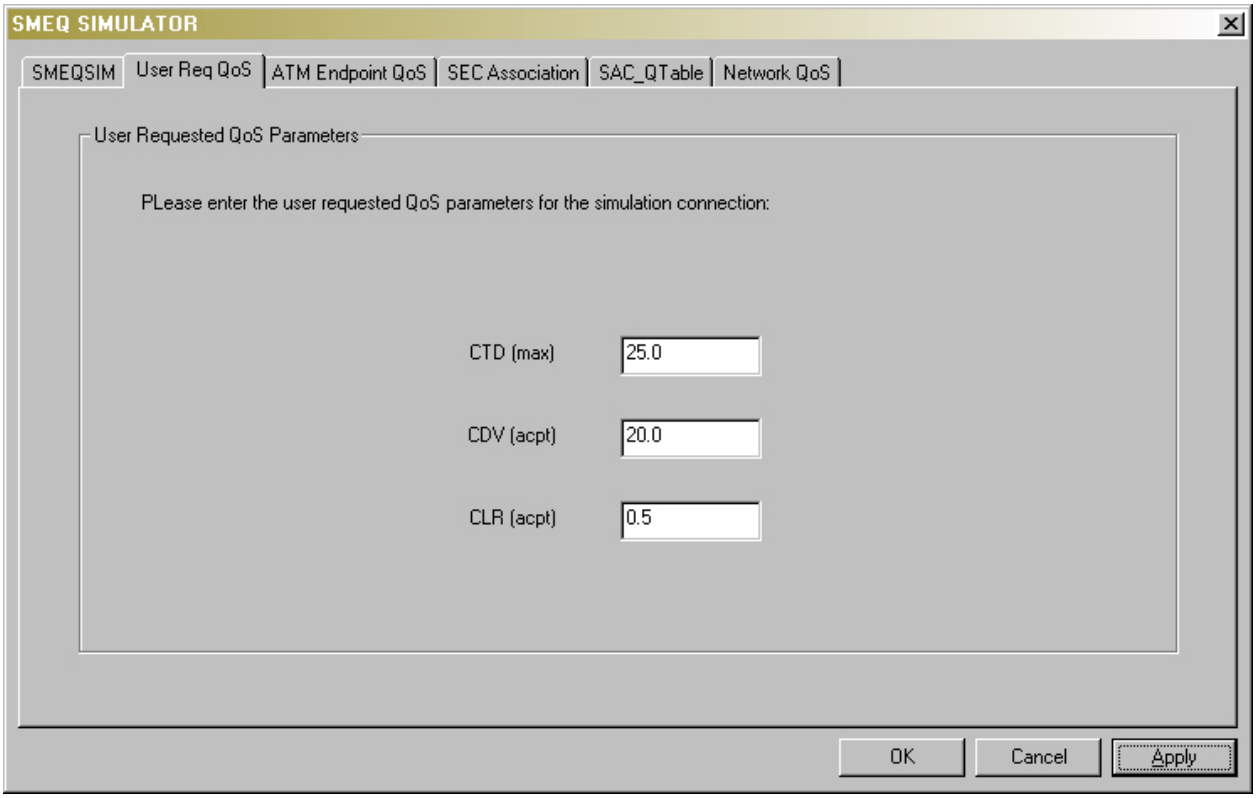


Figure 12.5 *User Req QoS* **Page of the SMEQSIM**

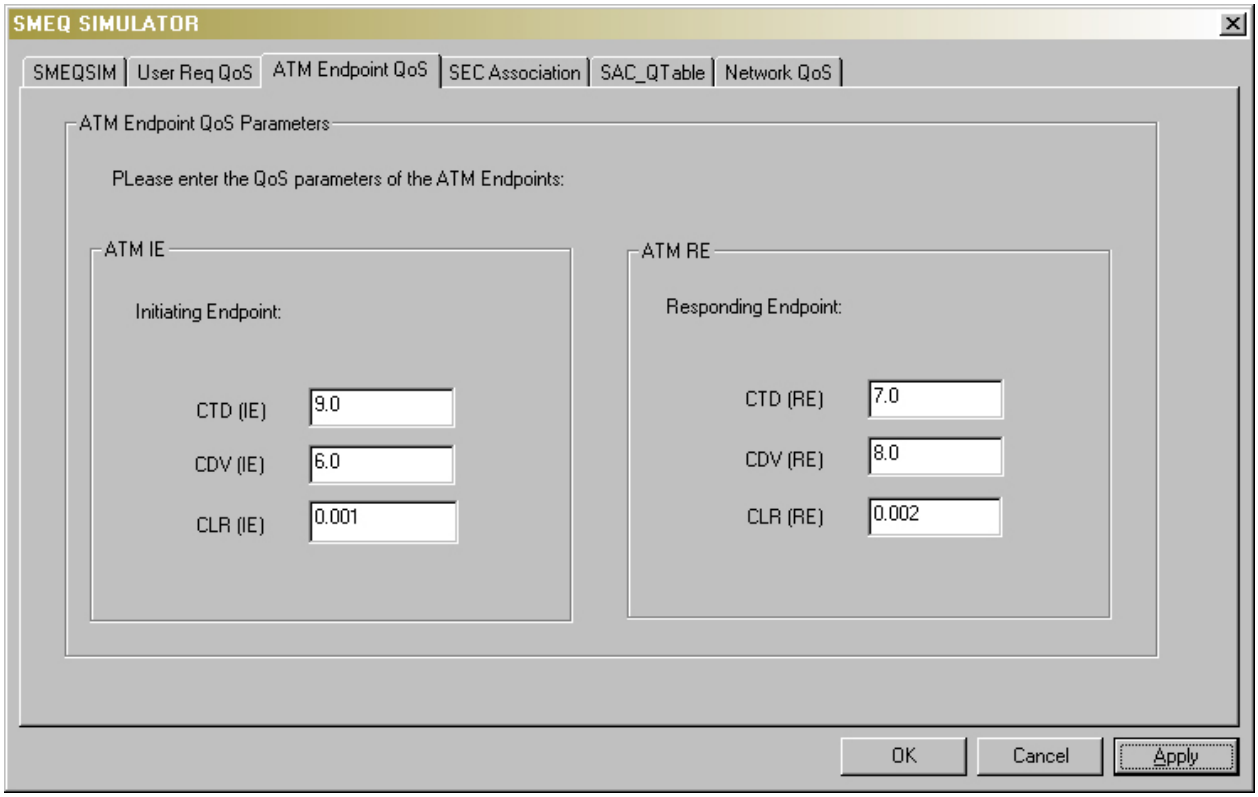


Figure 12.6 *ATM Endpoint QoS* **Page of the SMEQSIM**

On the *SEC Association* tab in Figure 12.7, the desired security associations between the two SAs are entered. To be able to address and identify different SAs, especially in large and complex network simulation cases, SMEQSIM provides the capability of numbering the SAs. The first SA field is considered for the Initiating SAs (SA_I). The second SA field is considered for the corresponding Responding SAs (SA_R).

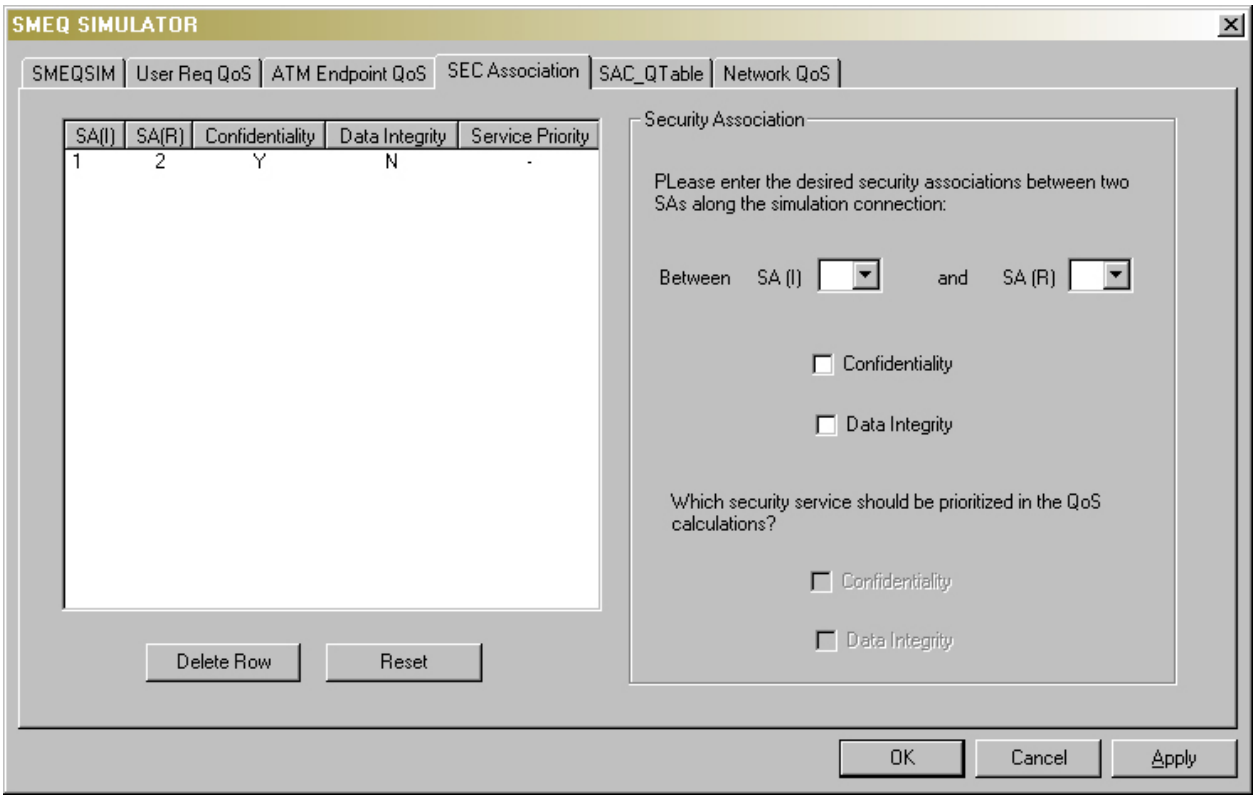


Figure 12.7 *SEC Assosiation* **Page of the SMEQSIM**

As described in Chapter 11, the Confidentiality and Data Integrity security services are of significance for the Traffic QoS calculations in the proposed SMEQ. The SMEQ SIMULATOR provides the means to simulate a security association for either of the services or to simulate two simultaneous security associations for a pair of SAs.

On the *SEC Association* tab, the user can select either *Confidentiality* or *Data Integrity* for simulation of one security association at a time. If both security services are selected, the bottom two

fields are accessible as depicted in Figure 12.8. According to the Traffic QoS calculations described in details in the last chapter, for the negotiation purposes, one of the selected services should be prioritized. The security service with the higher priority is selected first and the second one is chosen according to the remaining quota of the user requested maximum and acceptable values.

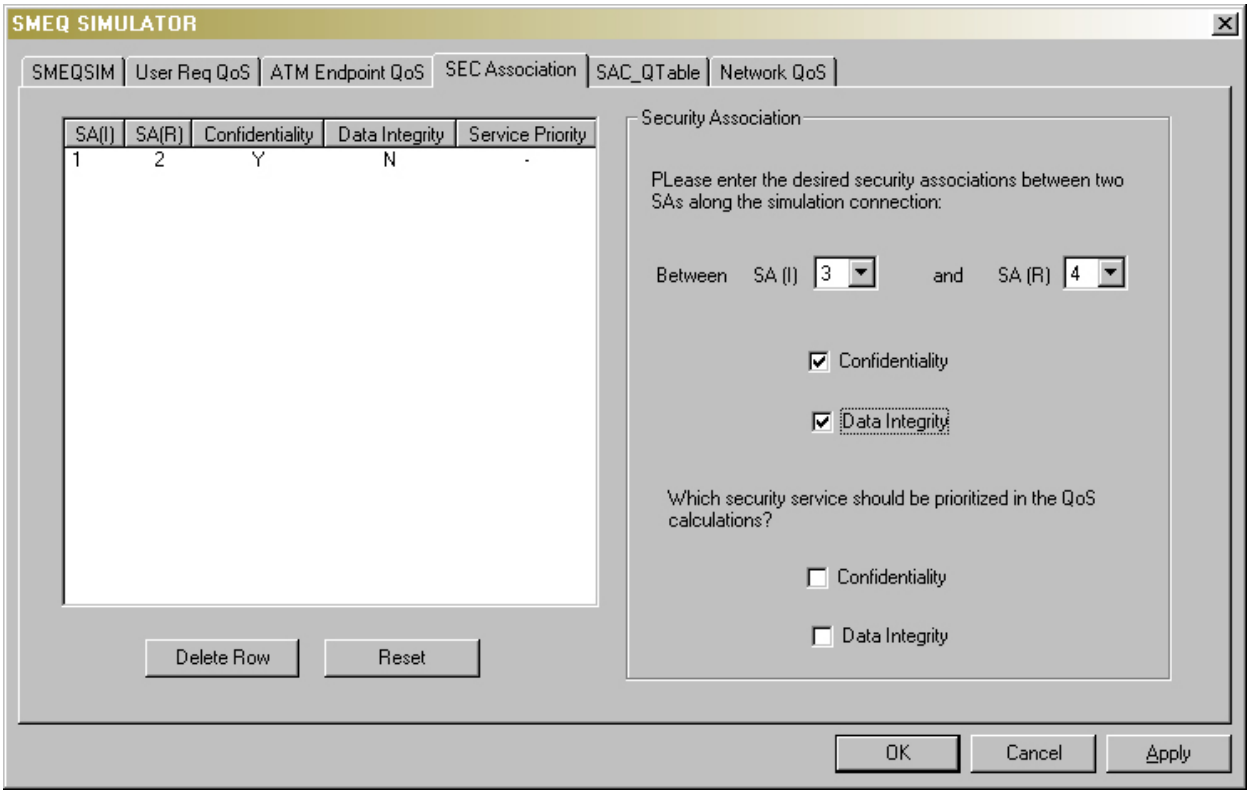


Figure 12.8 *Selection of both security services* **on the SEC Association Page**

After making the desired selections the *Apply* button is pressed. At this time, the entered data is added to the list view on the left of the screen. For the convenience of modifying large number of entries, the SMEQSIM design also provides the capability to delete rows (*Delete Row* button) or delete all entries (*Reset* button) of the list view.

On the *SAC_Q Table* tab as depicted in Figure 12.9, additional Traffic QoS degradation values caused by security operations in each SA are entered for all SAs involved in the secure ATM connection for this simulation scenario. That means for every SA and its security association(s) listed on the previous tab (*SEC Association*) corresponding Traffic QoS degradation values should be entered here. These entries make the SAC_Q Table described in Chapter 11 for that SA.

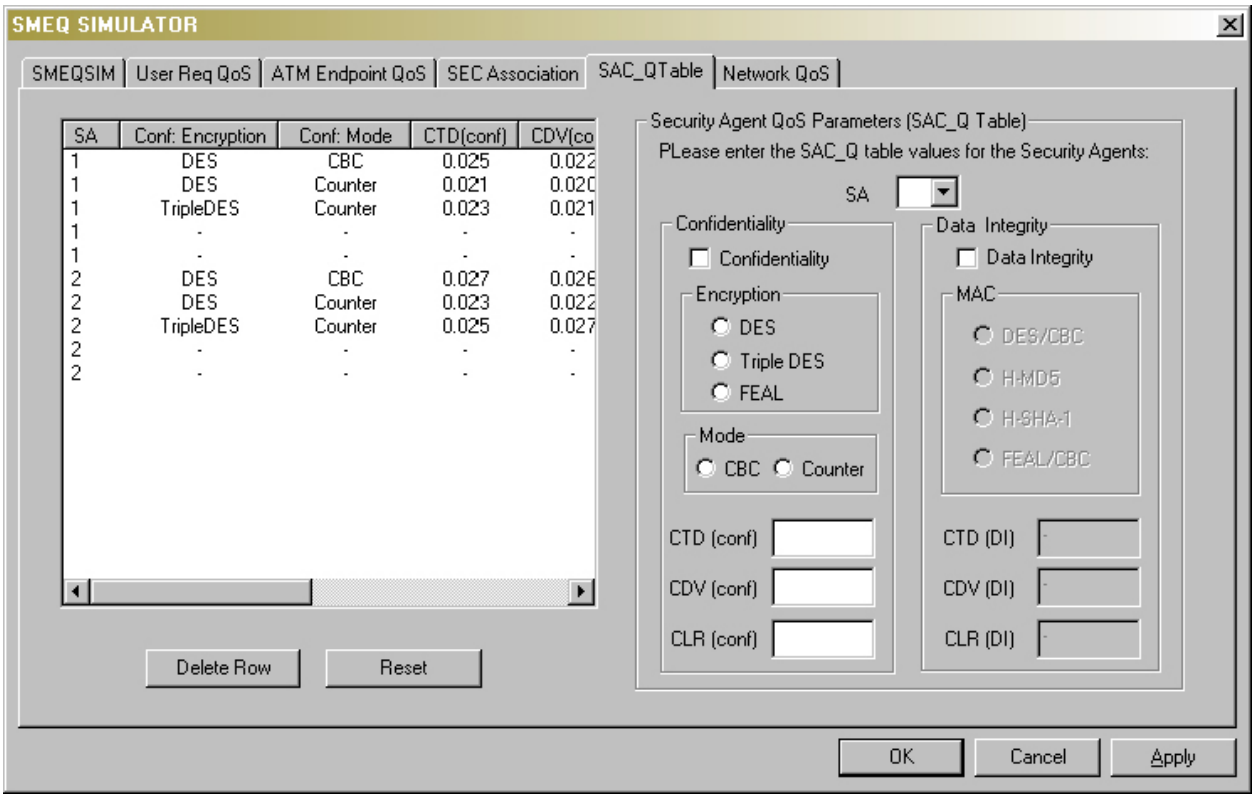


Figure 12.9 *SAC_Q Table* **Page of the SMEQSIM**

To make entries for a particular SA, its number is entered in the SA field. Every time only the values for one security service can be entered. The *Confidentiality* section is by default enabled. Once *Data Integrity* is selected, this section is enabled and the *Confidentiality* section is disabled. The user can make several entries for different choices of security services. This happens by each time entering the data one set at a time and pressing *Apply* before selecting the next entry.

Each time after making the desired selections, the *Apply* button is pressed. At this time, the entered data is added to the list view on the left of the screen. For the convenience of modifying large number of entries, the SMEQSIM design also provides the capability to delete rows (*Delete Row* button) or delete all entries (*Reset* button) of the list view.

On the *Network QoS* tab as depicted in Figure 12.10, the Traffic QoS degradation values caused by intervening networks for this particular simulation network are entered. For example for the simplest secure network containing one pair of SAs only the values for one intervening network are required. To be able to address and identify different intervening networks, especially in large

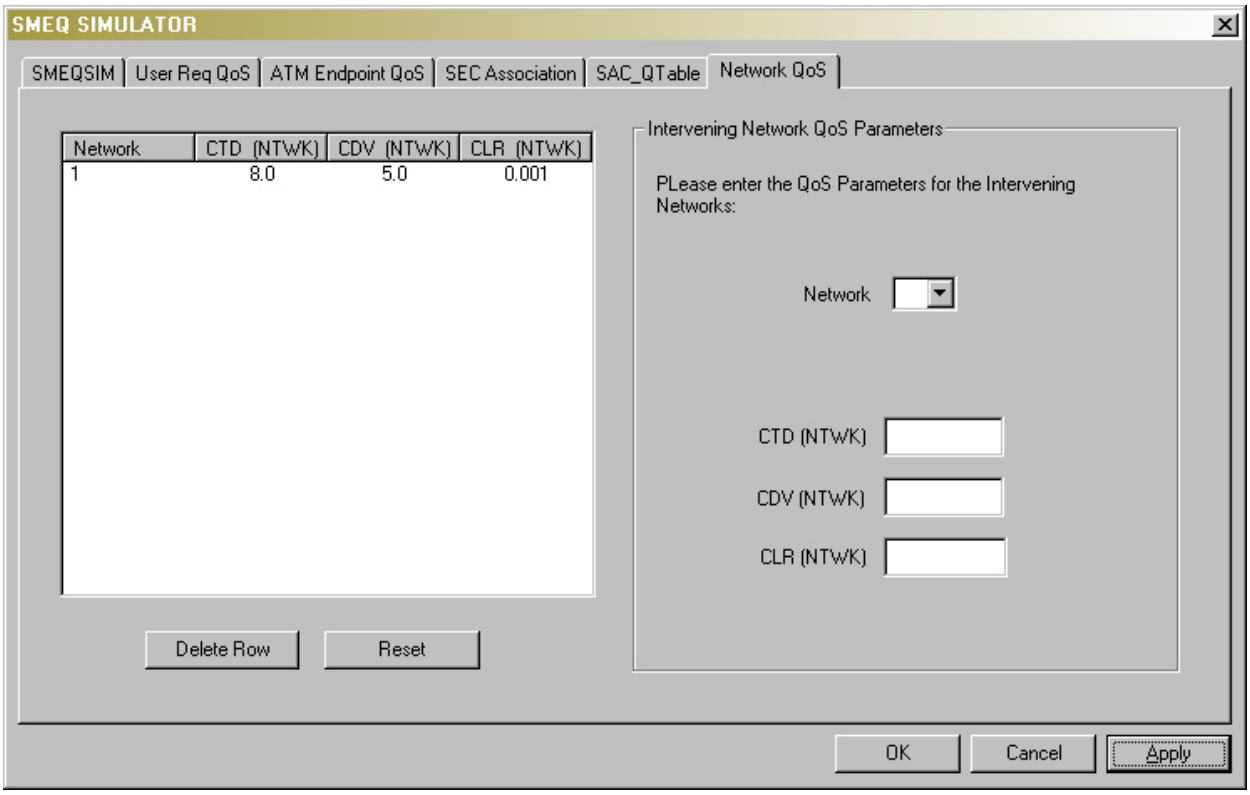


Figure 12.10 *Network QoS* **Page of the SMEQSIM**

and complex network simulation cases, SMEQSIM provides the capability of numbering them. The user enters the network number. Then the Traffic QoS values for this network are entered in the CTD, CDV and CLR fields.

Each time after making the desired selections, the *Apply* button is pressed. At this time, the entered data is added to the list view on the left of the screen. For the convenience of modifying large number of entries, the SMEQSIM design also provides the capability to delete rows (*Delete Row* button) or delete all entries (*Reset* button) of the list view.

After applying the final selections of all SMEQSIM property pages and pressing the *OK* button, all entered data are automatically saved to files (one unique simulation data file per property page) for reuse in the next simulation process. This prevents the reentering of large amount of data for complex scenarios over and over again. If the simulation data files already exist they will be rewritten with the new simulation data. Every time the SMEQSIM is started the program searches for these simulation data files in the current directory. If they exist, the property pages and list views are

initialized with these data. If the files exist and no entries are listed in the list view, the list view should first be reseted before new data can be entered.

The *SIMPROC* screen is displayed after pressing the OK button as depicted in Figure 12.11. On this screen, pressing the *Run Simulation* button will process the actual simulation.

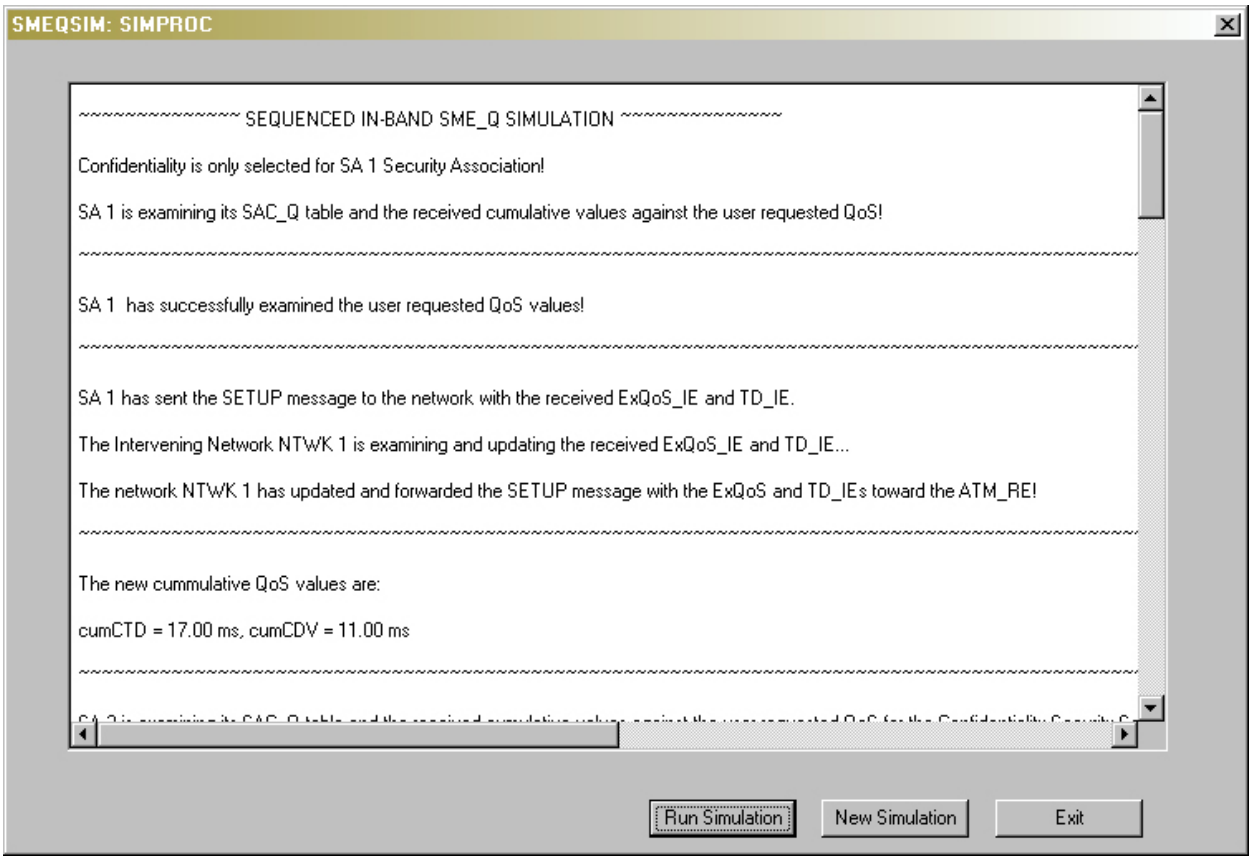


Figure 12.11 *SIMPROC* **Page of the SMEQSIM**

The log of the simulation transactions and results for each network element is listed step by step through the course of the ATM connection on the list view. A copy of the simulation results log file for a sample simulation scenario is illustrated in Appendix A. In case of a negative test scenario, where the selected simulation Traffic QoS parameters should not be or are not met or exceeded by the network elements, the simulation is interrupted and terminated by an Error Message Box as depicted in Figure 12.12. The setup rejection cause is also commented in the simulation results log screen.

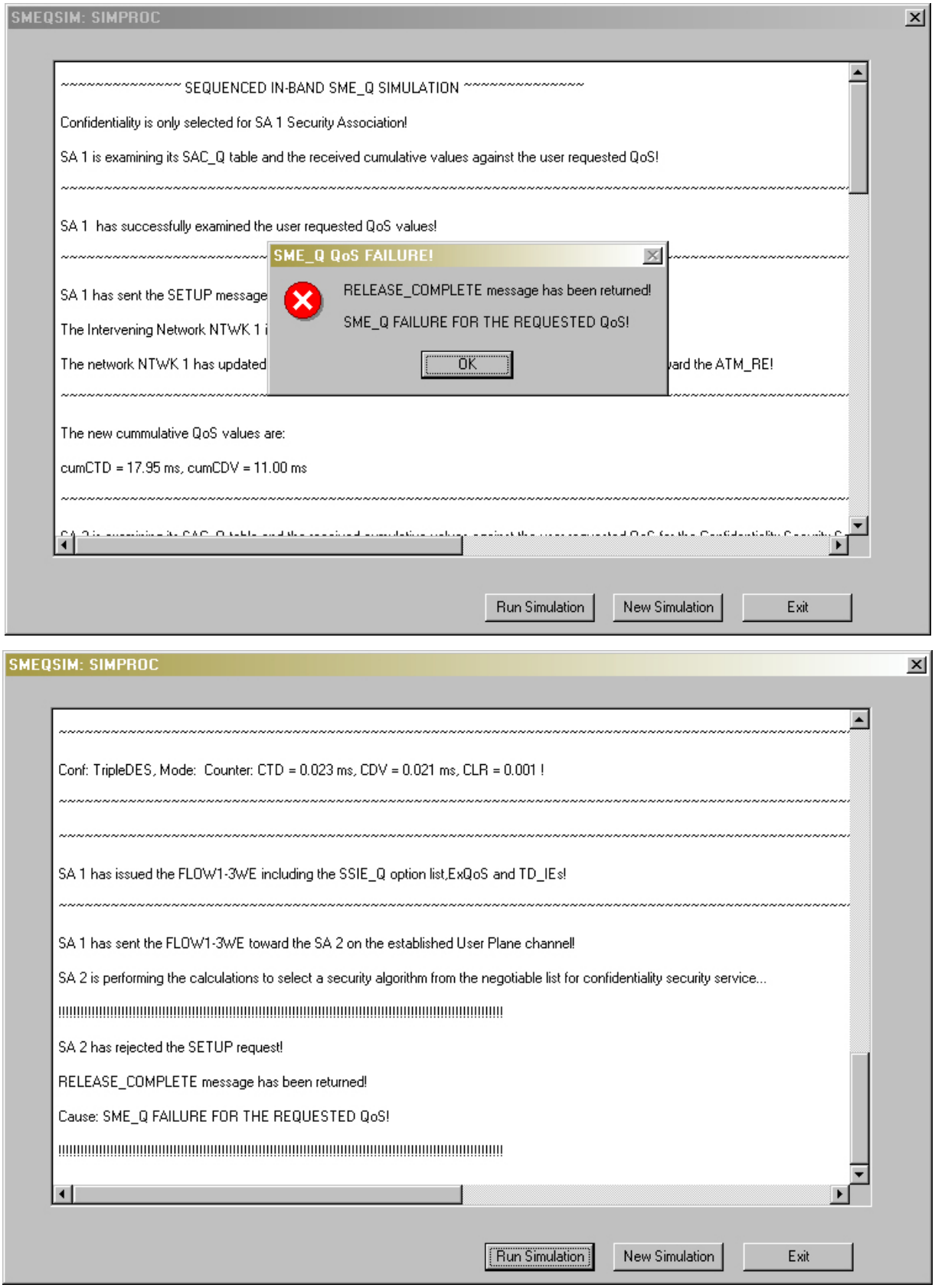


Figure 12.12 *QoS Rejection* **Simulation Scenario for In-Band SME_Q**

Pressing the *New Simulation* button starts a new simulation. The SMEQSIM property pages are displayed and initialized by the last simulation data. Pressing the *Exit* button terminates the simulation. The In-Band SME_Q prototype application is designed to simulate both the sequenced and the nesting security architectures.

12.1.1 Sequenced In-Band SME_Q Prototype Simulation

The Sequenced In-Band SME_Q prototype is designed to simulate the sequenced security architecture for a secure ATM connection with no overlaps. Figure 11.16 illustrates the simulation process flow of the Sequenced In-Band SME_Q. The simulation network in this case consists of more than one pair of neighboring Security Agents ($SA_{(I)1..20}$, $SA_{(R)1..20}$), connected with each other through intervening networks ($NTWK_{1..20}$), as well as two ATM devices, the Initiating Equipment (ATM_{IE}) and the Responding Equipment (ATM_{RE}).

The simulation follows the procedures and equations described in detail in Chapter 11. The application workflow of the prototype including the required input and the resulted output for a desired simulation test case is depicted in Figure 12.13.

12.1.2 Nesting In-Band SME_Q Prototype Simulation

The Nesting In-Band SME_Q prototype is designed to simulate the nesting security architecture for a secure ATM connection with no overlaps. Figure 11.15 illustrates the simulation process flow of the Nesting In-Band SME_Q. The initiating Security Agents in case of a nesting architecture are all on the initiating side of the network and the corresponding responding security agents on the responding side of the network.

The simulation network in this case consists of more than one pair of neighboring Security Agents ($SA_{(I)1..20}$, $SA_{(R)1..20}$), connected with each other through intervening networks ($NTWK_{1..20}$), as well as two ATM devices, the Initiating Equipment (ATM_{IE}) and the Responding Equipment (ATM_{RE}).

The simulation follows the procedures and equations described in detail in Chapter 11. The application workflow of the prototype including the required input and the resulted output for a desired simulation test case is depicted in Figure 12.14.

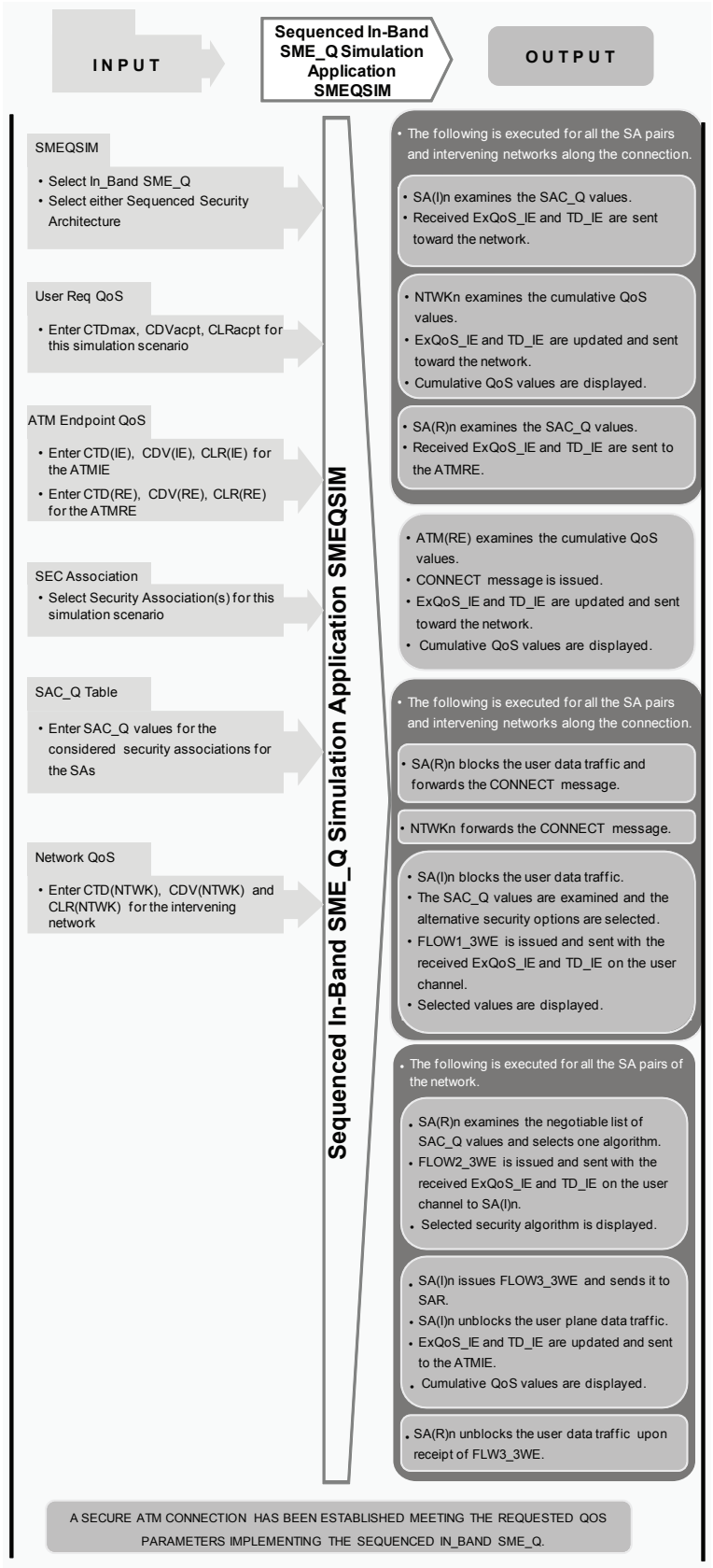


Figure 12.13 The Sequenced In-Band SME_Q Simulation Work Flow Diagram

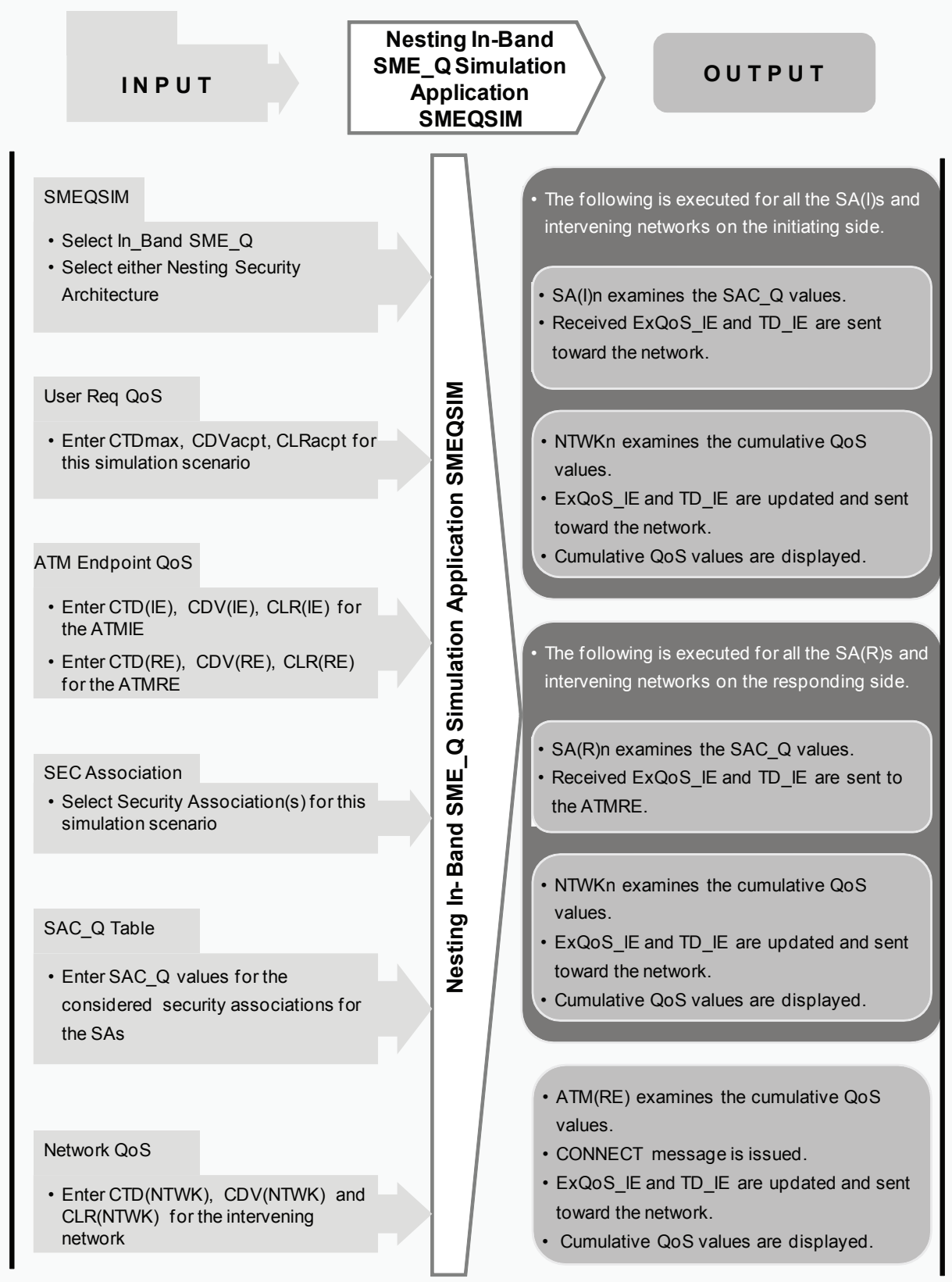


Figure 12.14 The Nesting In-Band SME_Q Simulation Work Flow Diagram
(to continue next page)

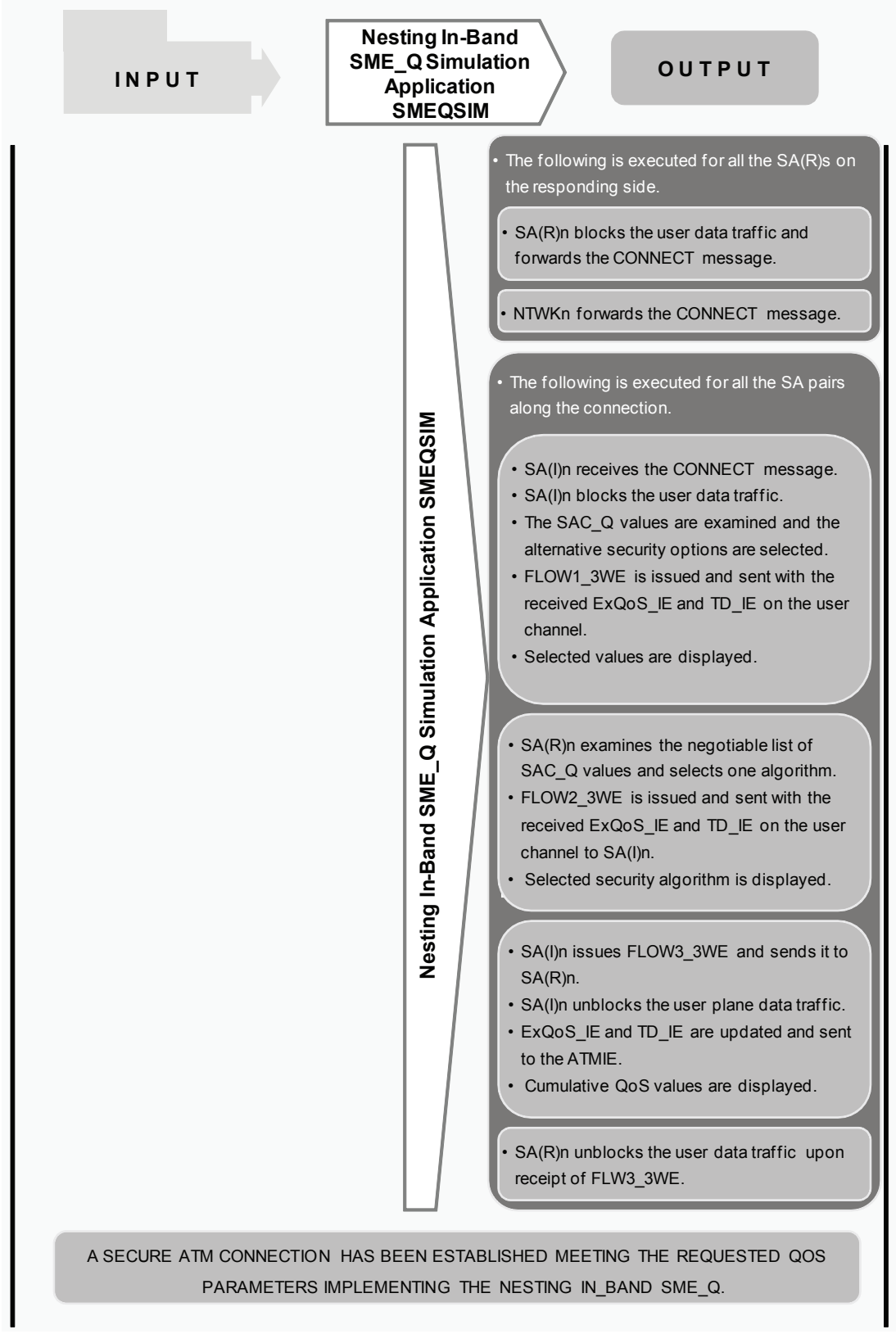


Figure 12.14 The Nesting In-Band SME_Q Simulation Work Flow Diagram
(continued from previous page)

12.1.3 SMEQSIM Operations Summary

The simulation follows the procedures and equations described in detail in Chapter 11. For the negotiation purposes, one of the selected services should be prioritized. The security service with the higher priority is selected first and the second one is chosen according to the remaining quota of the user requested maximum and acceptable values.

After making the desired selections the *Apply* button is pressed. At this time, the entered data is added to the list view on the left of the screen. For the convenience of modifying large number of entries, the SMEQSIM design also provides the capability to delete rows (*Delete Row* button) or delete all entries (*Reset* button) of the list view.

At the end of data entry for the security associations, the number of lines in the list view should correspond to the number of existing SA pairs in the network considered for this simulation scenario. This means, at least two entries (lines), four SAs, are needed for the simulation of a Nesting In-Band SME_Q simulation scenario.

On the *SAC_Q Table* tab, additional Traffic QoS degradation values caused by security operations in each SA are entered for all SAs involved in the secure ATM connection for this simulation scenario. That means for every SA and its security association(s) listed on the previous tab (*SEC Association*) corresponding Traffic QoS degradation values should be entered here. These entries make the SAC_Q Table described in Chapter 11 for that SA.

To make entries for a particular SA, its number is entered in the SA field. Every time only the values for one security service can be entered. The *Confidentiality* section is by default enabled. Once *Data Integrity* is selected, this section is enabled and the *Confidentiality* section is disabled. The user can make several entries for different choices of security services. This happens by each time entering the data one set at a time and pressing *Apply* before selecting the next entry.

Each time after making the desired selections, the *Apply* button is pressed. At this time, the entered data is added to the list view on the left of the screen. For the convenience of modifying large number of entries, the SMEQSIM design also provides the capability to delete rows (*Delete Row* button) or delete all entries (*Reset* button) of the list view.

On the *Network QoS* tab, the Traffic QoS degradation values caused by intervening networks for this particular simulation network are entered. For example the simplest secure network in the case of Nesting In-Band SME_Q contains two pairs of SAs. In this case, the values for three intervening networks are required. To be able to address and identify different intervening networks, especially in large and complex network simulation cases, SMEQSIM provides the capability of numbering them. The user enters the network number. Then the Traffic QoS values for this network are entered in the CTD, CDV and CLR fields.

Each time after making the desired selections, the *Apply* button is pressed. At this time, the entered data is added to the list view on the left of the screen. For the convenience of modifying large number of entries, the SMEQSIM design also provides the capability to delete rows (*Delete Row* button) or delete all entries (*Reset* button) of the list view.

After applying the final selections of all SMEQSIM property pages and pressing the *OK* button, all entered data are automatically saved to files (one unique simulation data file per property page) for reuse in the next simulation process. This prevents the reentering of large amount of data for complex scenarios over and over again. If the simulation data files already exist they will be rewritten with the new simulation data. Every time the SMEQSIM is started the program searches for these simulation data files in the current directory. If they exist, the property pages and list views are initialized with these data. If the files exist and no entries are listed in the list view, the list view should first be reseted before new data can be entered.

The *SIMPROC* screen is displayed after pressing the OK button. On this screen, pressing the *Run Simulation* button will process the actual simulation. In case of a negative test scenario, where the selected simulation Traffic QoS parameters should not be or are not met or exceeded by the network elements, the simulation is interrupted and terminated by an Error Message Box. The setup rejection cause is also commented in the simulation results log screen.

Pressing the *New Simulation* button starts a new simulation. The SMEQSIM property pages are displayed and initialized by the last simulation data. Pressing the *Exit* button terminates the simulation.

Chapter

13

Quantitative Analysis

Impact of Security Operations on the QoS in ATM Networks

This chapter builds upon the qualitative approach of this work up to this point, which assumed the influences of the security operations on the traffic QoS as given and known parameters. The focus of this dissertation has been to develop new security protocols to address the question of *how* one could take these assumed degradations based upon the existing standardized protocols and procedures into account. New SME_Q protocols have been developed to address this problem. A simulation software has also been developed to simulate the traffic QoS provisioning procedures for different possible security architectures in ATM networks.

At this point a real world example of an ATM network is studied to illustrate the effect of these influences in a quantitative and analytical approach. The in-band security operations are used for illustration in this chapter.

The communications of two end systems within this network are analyzed to compare the three following cases in respect to the resulting traffic QoS degradations, a *non-secure* ATM connection, a *secure* ATM connection, and a *secure* ATM connection *offering traffic QoS* (proposed *SME_Q* protocol). The resulting values of the model network for the above cases are then graphically presented and analytically analyzed and compared.

13.1 Model ATM Network

One of the most exciting areas of ATM technology implementations is in the field of healthcare. The ever-increasing need for electronic communications of patient files, medical imaging and in recent years videoconferencing for remote real-time diagnosis, patient care, tele-surgery and radiology has set high demands on simultaneous bandwidth, scalability and security requirements of the underlying networks.

The concurrent delivery of these complex needs makes a real life example of a healthcare system a good candidate for the anticipated study. In addition to the privacy demanding administrative and financial communications, patient files and health records exchanges and last but not least the delay, privacy and accuracy sensitive transmissions of Telemedicine real-time video communications make this field an exciting ground for the implementation of the developed protocols in this work. With the following exercises the impact of security on such an implementation will be explored and concluded.

HealthSystem Minnesota (HSM) located in the Twin Cities (*Minniapolis* and *St. Paul*) suburb of *St. Paul Park* is a merger of the *Methodist Hospital*, the *Park Nicollet Medical Center* and a Primary Physician Network [HSM_97][HsPn_98]. This leading health care provider consists of 25 clinic and hospital locations scattered throughout the Twin Cities within an approximately 17 Miles radius. Figure 13.1 illustrates the geographical spread of these sites [HsCl_98].

An ATM switch is installed in each location and two at the Park Nicollet campus to build the ATM Intranet [HSM_97].

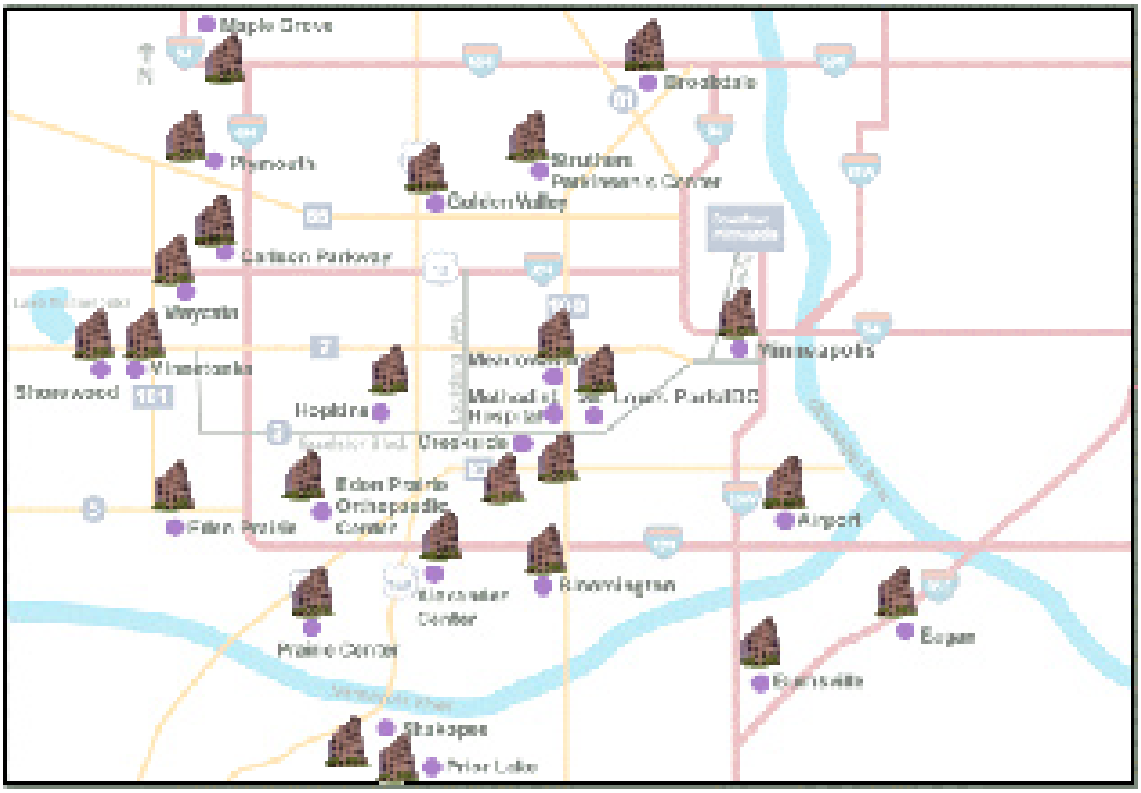


Figure 13.1 HealthSystem Minnesota ATM Network, *Source [HsCl_04]*

With these assumptions of a real life ATM network different scenarios for real time services such as secure and private video transmissions are considered to study the influences of security operations on the QoS of such a network. These degradations are compared first in relative to increasing number of nodes for different scenarios by keeping the configuration and number of the existing security associations constant. Secondly, on scenario is chosen (constant number of ATM switches and nodes) and the result is analyzed in comparison with increasing number of security associations on the same connection path.

Figure 13.2 illustrates an example of the considered model network – used for the scenario no. 2 – with four encryption devices (Security Agents) for this analysis.

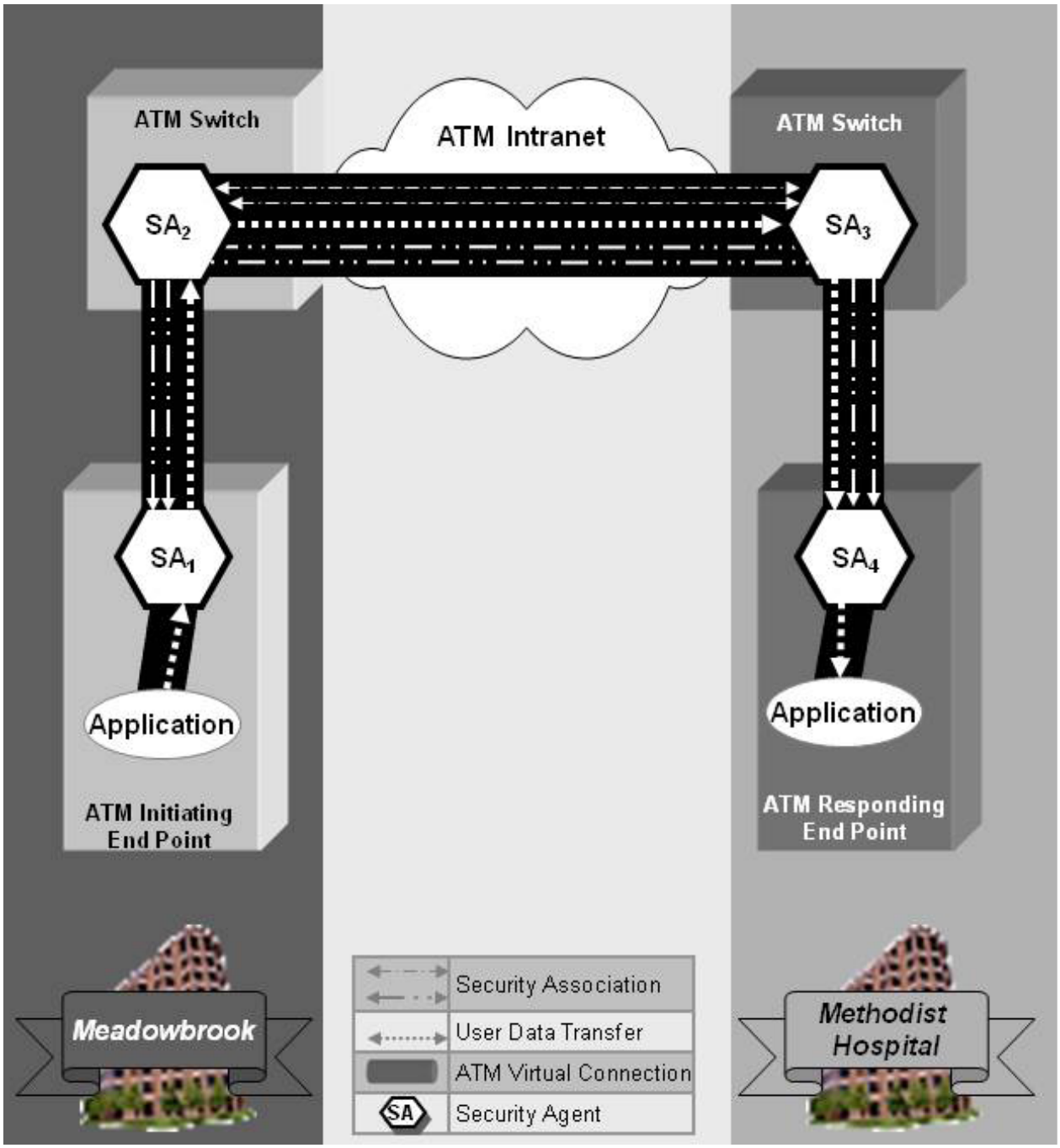


Figure 13.2 Model Network Example

13.2 Real-Time Secure Video Transmission

Real-Time Telemedicine requires transmission of accurate, delay and jitter sensitive high-quality video. ATM Technology with its promises of controlled scalability and high bandwidth is among the forerunners for this pursuit.

The following sections describe the general assumptions for the component characteristics and selected values for the model networks. The analysis scenarios are explained and the achieved results are graphically illustrated and analytically compared.

13.3 Analysis Assumptions

As described above, the study is based on the basic configuration of the *HeatlthSystem Minnesota* real-life ATM intranet network. The geographical spread of this network is only used as an example for the considerations of different scenarios with respect to assumable number of nodes in the following scenarios.

For the characteristic values of the QoS degradations of the ATM switches and encryptors, however, the today available devices in the market, along with the values found in standards and literature are studied and typical values are considered. Any resulting performance evaluations are solely based on the considered scenarios and do not make any statement on those of the real-life *HeatlthSystem Minnesota* ATM intranet network.

As described in Chapter 11, the ATM Forum [ATM_00] takes a deterministic worst-case approach for the calculations and negotiations of the QoS. The worst-case cumulative values of the negotiable delay parameters (CTD and CDV) are added at each traversed network node and should not exceed the user requested value throughout a connection. To date, however, exact manufacturer or literature information on the CDV parameter of encryptor-devices has been almost non-existent. The CLR parameter, which is not a cumulative value, is not addressed in the manufacturer literature either. Although the developed security SME_Q protocols take both of the cumulative degradations into consideration, we base our quantitative study on the accumulation of the CTD value, which also indirectly incorporates the Cell Delay Variations.

Further, although the manufacturer literatures state a CTD value for their devices, they do not distinguish between the encryption algorithms, modes of operations and protocol processing in respect to their individual part of these degradations. They assume a constant maximum delay value for their operations regardless of the complexity and speed of their used security mechanisms. This

leads to the conclusion that the area of quantitative performance and QoS evaluation of security operations in networks introduces a much-needed attention for future research.

For the purpose of focus on the comparison of the three different analysis cases mentioned earlier, various in the market available encryption devices are studied and typical delay values across these devices are assumed for the Cell Transfer Delay parameter [CeSa_00] [CyCe_99] [CyLn_04] [AtCr_03] [SiTl_00] [SaNe_04]. Assumption of the same typical degradation value among Security Agents in the model networks for each analysis scenario makes the comparative analysis of different cases of network structure clearer. The results are then compared for the minimum and maximum assumable values.

Finally, the model networks for each scenario are assumed to have the following component characteristics for the real-time transmission of video for implementations of Telemedicine:

MPEG-2 is considered as the video-coding standard for the study. Each end point is considered to have a codec (coder/decoder) device. The in the market available devices today operate at the rates of 10 Mbps to 80 Mbps, which introduce delays in the range of 4.8 μs to 38.4 μs per cell ($CTD_{(IE,IR)}$). Two end points for each connection introduce a total end-to-end delay of 9.6 μs to 76.8 μs per cell transmission. The ATM switches typical operation rates of 155 Mbps introduce a 2.8 μs delay at the network nodes ($CTD_{(NTWK)}$).

The studied encryptors have reported latencies in the range between 15μs to 30μs ($CTD_{(I,R)}$), which are the considered delays caused by security operations and implementation of each security association [CeSa_00] [CyCe_99] [CyLn_04] [AtCr_03] [SiTl_00] [SaNe_04]. For the overhead calculations a 20% ballpark figure is assumed for the protocol processing and handling of these devices.

The SME_Q proposes seven additional octets to the existing ATMSEC SME protocol [SEC_11]. With the assumption of the 20% ballpark figure and in respect to the considered encryptor delays the SME_Q protocol introduces additional cell transfer delays in range of 0.4μs to 0.79μs. This results to a protocol overhead processing increase of 2.6 % compare to the current rates.

13.4 Case Study: End-to-End CTD over Increasing No. of Nodes

Figures 13.3, 13.4, 13.5, 13.6 illustrate the model network configurations and security associations for the first four analysis scenarios. Here the number of security associations and encryption devices are held constant and the resulting cell transfer delay behavior is studied in respect to the above mentioned cases (non-secure, secure, SME_Q secured connections) with increasing number of traversed network nodes. A total of four encryptors implementing both confidentiality and data integrity services resulting to four security associations are considered.

It is further assumed that in addition to its network switch each end point includes an encryptor for the absolute end-to-end security. Table 13.1 illustrates the Input data for these scenarios. As depicted the cell transmission delay caused by security operations are the same for all cases and have resulted from 120μs to 240μs for four security associations per connection.

Table 13.2 displays the resulted values corresponding to these input data for two cases of non-secure transmission and secure transmission. Table 13.3 illustrates these values for the case that SME_Q is implemented to establish the secure connection while offering the QoS.

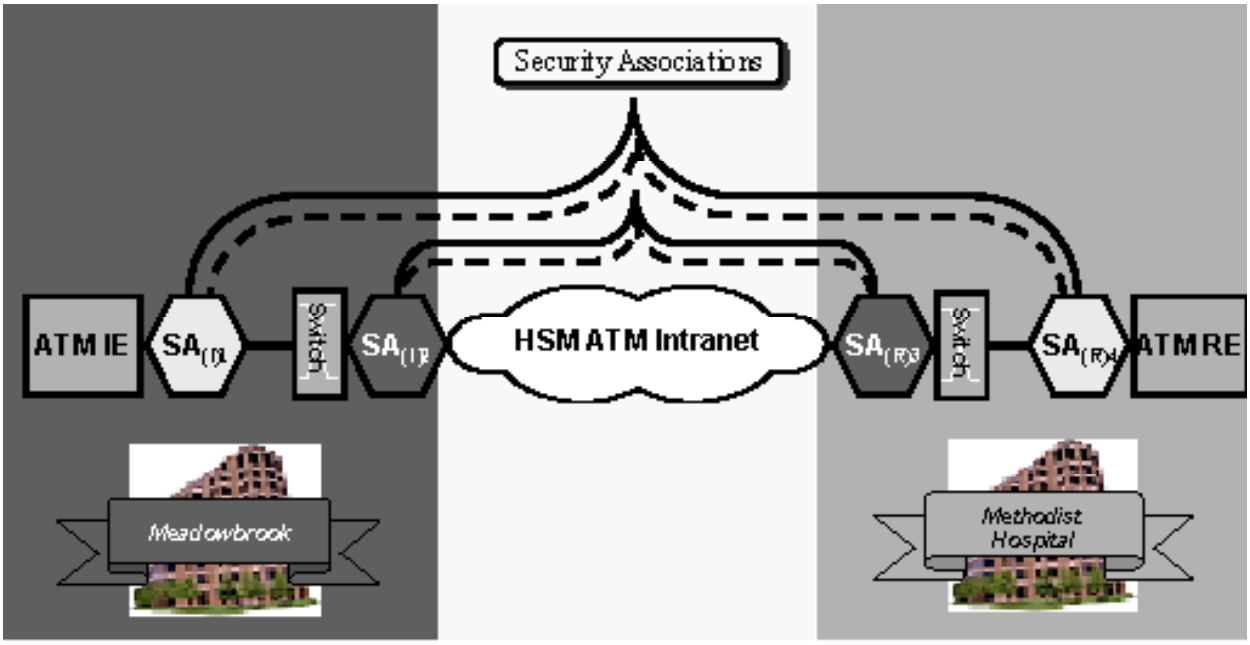


Figure 13.3 Model Network Configuration for Scenario 1

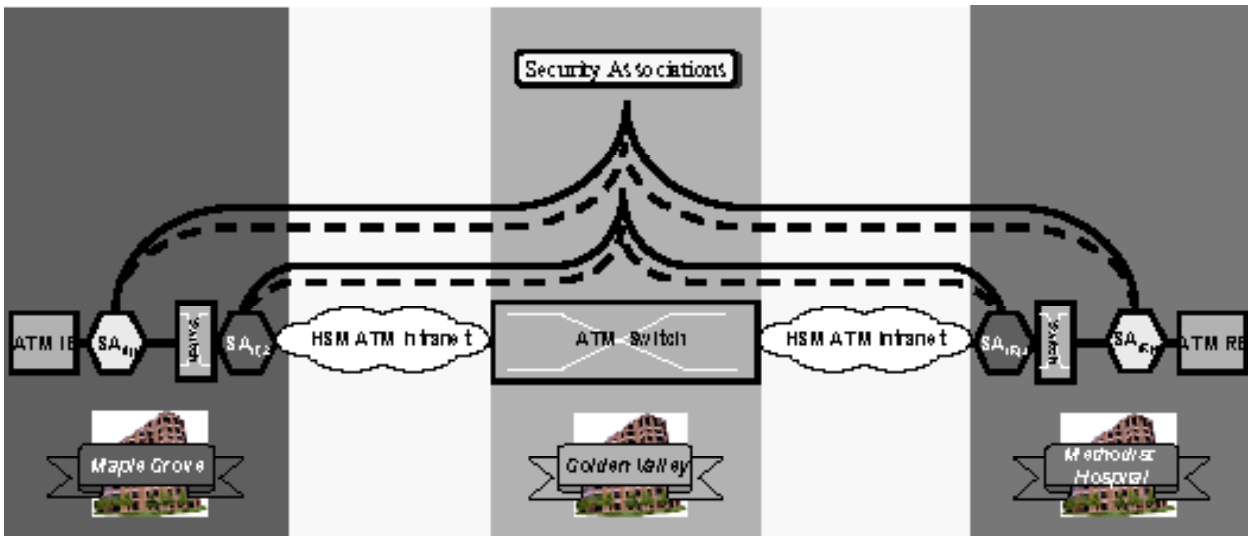


Figure 13.4 Model Network Configuration for Scenario 2

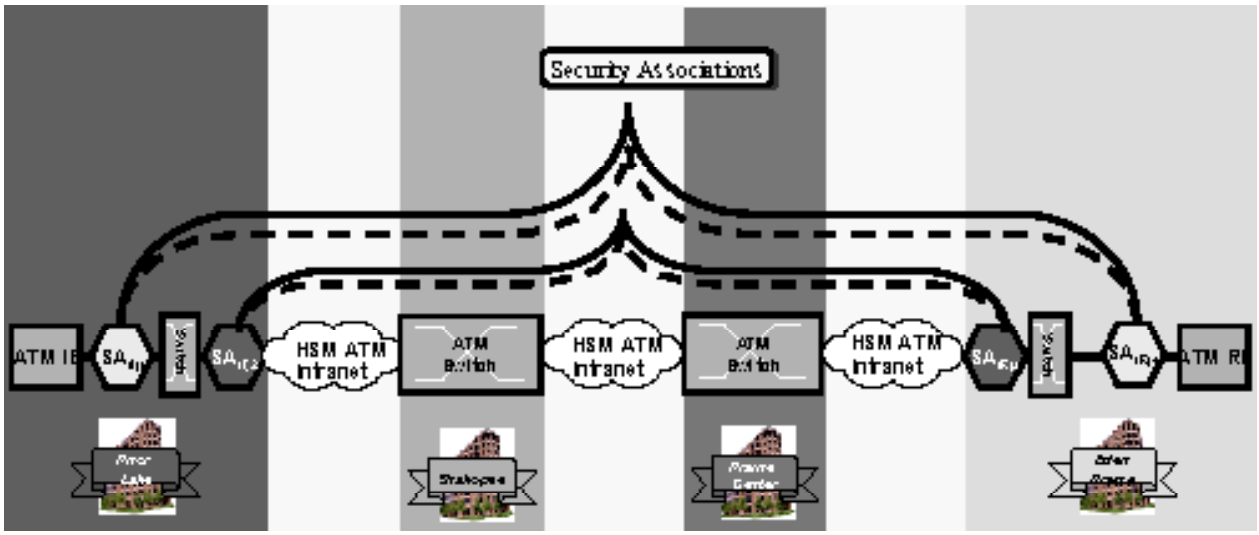


Figure 13.5 Model Network Configuration for Scenario 3

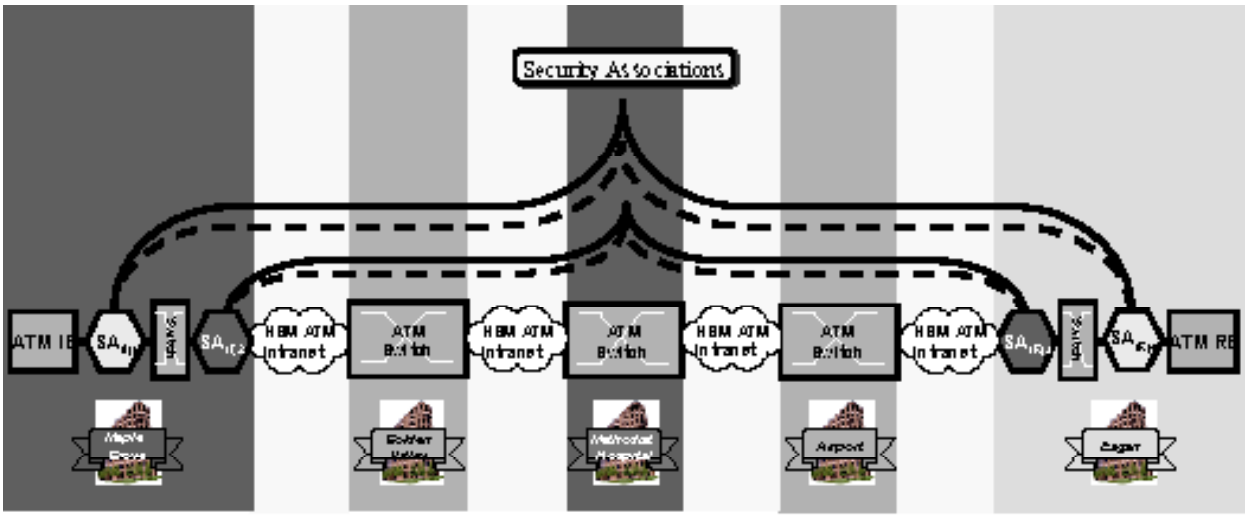


Figure 13.6 Model Network Configuration for Scenario 4

Scenario #	Communication Path	No. of Nodes	Distant km	No. of ATM Switches	No of ATM Encryptor	No of Security Assoc.	Total EndPoint $CTD_{IE,RE}$ in µs for MPEG-2 Codec rates in Mbps				Total end-to-end CTD_{NTWK} of the network nodes (Switch) in µs	CTD_{SEC} Secure Connection for encrptor's CTD(I,R) in µs		
							10	15	50	80				
							38.4	25.6	7.68	4.8	2.8	15	24	30
1	Meadowbrook to Methodist Hospital (MB --> MH)	2	0.02	2	4	4	76.80	51.20	15.36	9.60	5.60	120.00	192.00	240.00
2	Maple Grove to Methodist Hospital (MG --> MH)	3	26.00	3	4	4	76.80	51.20	15.36	9.60	8.40	120.00	192.00	240.00
3	Prior Lake to Eden Prairie (PL --> EP)	4	45.00	4	4	4	76.80	51.20	15.36	9.60	11.20	120.00	192.00	240.00
4	Maple Grove to Eagan (MG --> EG)	5	82.00	5	4	4	76.80	51.20	15.36	9.60	14.00	120.00	192.00	240.00

Table 13.1 Input Data for Scenarios1 to 4

Scenario #	Communication Path	Total end-to-end CTD for a Non-Secure Connection in µs for MPEG-2 Codec rates in Mbps				Total end-to-end CTD Secure Connection in µs for encrptor's CTD(I,R) in µs and MPEG-2 Codec rates in Mbps											
		10	15	50	80	15				24				30			
		38.40	25.6	7.68	4.8	38.40	25.60	7.68	4.80	38.40	25.60	7.68	4.80	38.40	25.60	7.68	4.80
1	Meadowbrook to Methodist Hospital (MB --> MH)	82.40	56.80	20.96	15.2	202.40	176.80	140.96	135.2	274.40	248.80	212.96	207.20	322.40	296.80	260.96	255.20
2	Maple Grove to Methodist Hospital (MG --> MH)	85.20	59.60	23.76	18.00	205.20	179.60	143.76	138.00	277.20	251.60	215.76	210.00	325.20	299.60	263.76	258.00
3	Prior Lake to Eden Prairie (PL --> EP)	88.00	62.40	26.56	20.80	208.00	182.40	146.56	140.80	280.00	254.40	218.56	212.80	328.00	302.40	266.56	260.80
4	Maple Grove to Eagan (MG --> EG)	90.80	65.20	29.36	23.60	210.80	185.20	149.36	143.60	282.80	257.20	221.36	215.60	330.80	305.20	269.36	263.60

Max Delays
Min Delays

Table 13.2 End-to-end CTD Outsome for Secure and Non-Secure Networks by Increasing No. of Nodes and Const. No. of Sec. Associations

Scenario #	Communication Path	Total end-to-end CTD_{SMEQ} for a Secure Connection in µs using SME_Q protocol (+ overhead) for delays in µs caused by different MPEG-2 Codec rates: Assumption: 1: 20% of the encrptor's delay is for processing of the security protocol 2: SME_Q adds 7 bytes of overhead to the existing ATMSECSME protocol											
		0.40				0.63				0.79			
		38.40	25.60	7.68	4.80	38.40	25.60	7.68	4.80	38.40	25.60	7.68	4.80
1	Meadowbrook to Methodist Hospital (MB --> MH)	202.80	177.20	141.36	135.60	275.03	249.43	213.59	207.83	323.19	297.59	261.75	255.99
2	Maple Grove to Methodist Hospital (MG --> MH)	205.60	180.00	144.16	138.40	277.83	252.23	216.39	210.63	325.99	300.39	264.55	258.79
3	Prior Lake to Eden Prairie (PL --> EP)	208.40	182.80	146.96	141.20	280.63	255.03	219.19	213.43	328.79	303.19	267.35	261.59
4	Maple Grove to Eagan (MG --> EG)	211.20	185.60	149.76	144.00	283.43	257.83	221.99	216.23	331.59	305.99	270.15	264.39

Table 13.3 End-to-end CTD Outsome for Secure Networks using SME_Q Protocol by Increasing No. of Nodes and Const. No. of Sec. Associations

These resulting values for the minimum delays are demonstrated according to the assumed input values in Figures 13.7 and 13.8. As depicted security operations have a very high impact on the QoS of the networks.

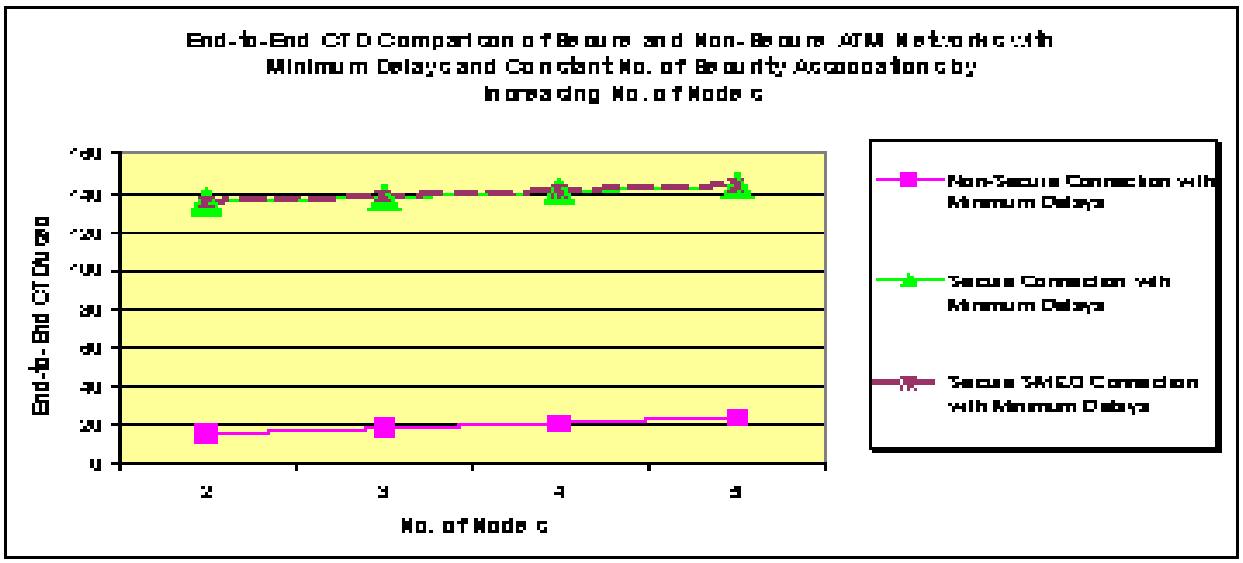


Figure 13.7 End-to-End CTD Comparison for Secure and Non-Secure Networks Assuming Minimum Delays by Increasing No. of Nodes and Const. No. of Sec. Associations

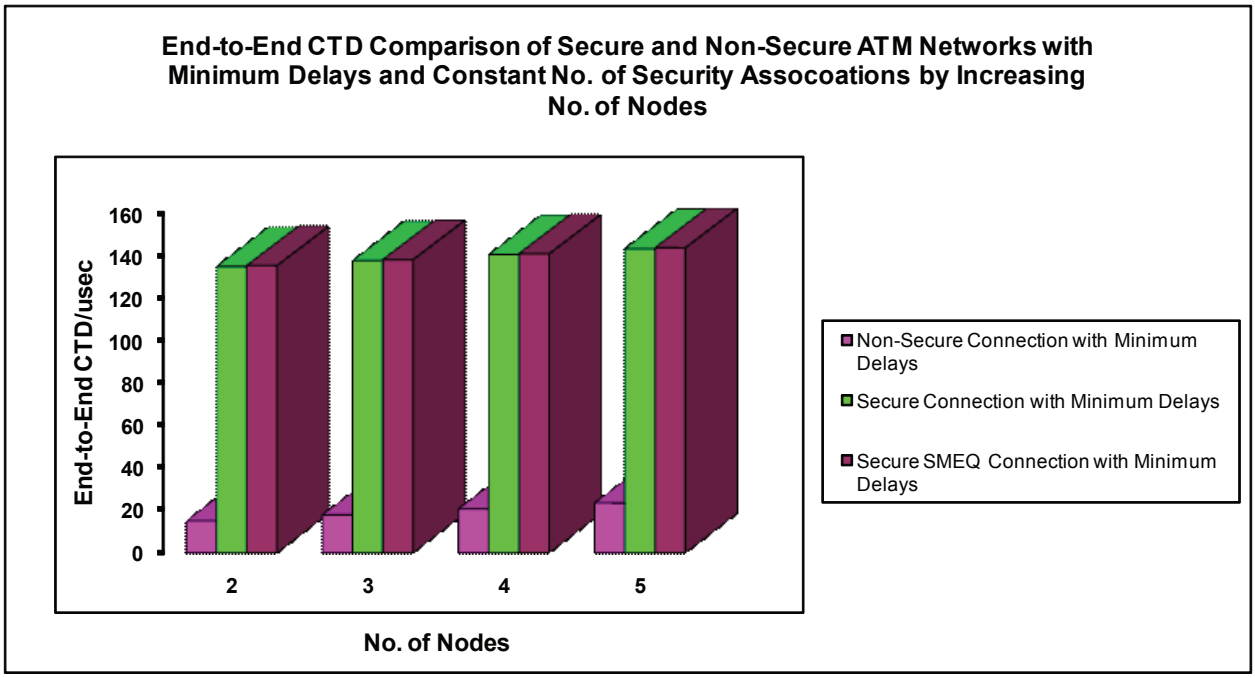


Figure 13.8 End-to-end CTD 3-D Comparison for Secure and Non-Secure Networks Assuming Minimum Delays by Increasing No. of Nodes and Const. No. of Sec. Associations

The scenarios demonstrate an increase of total cell transmission delays caused by security operations for the given number of security associations in the range from 11.2% to 16.2% in compare with the non-secure connections. As noticed this incongruity increases with increasing number of network nodes.

This analysis confirms the necessity for measures to consider these degradations of QoS in new protocols, hence the in this work presented SME_Q protocols.

According to the assumptions described earlier in this chapter, the additional overhead of the proposed SME_Q protocol is 2.6% in compare with the existing security protocols. This would be a minimal toll to pay to consider the 11.2% discrepancy to the achieved values of QoS, which today is not considered in secure communications. In addition to considering these degradations, the SME_Q protocol offers mechanisms to negotiate and consider security algorithms with smaller amount of contributing degradations, which can among others consequently decrease the number of network interruptions in extreme cases.

Figures 13.9 and 13.10 illustrate the same exercise assuming the highest delay values in the networks. The resulted degradation values from security operations here are even a lot higher and

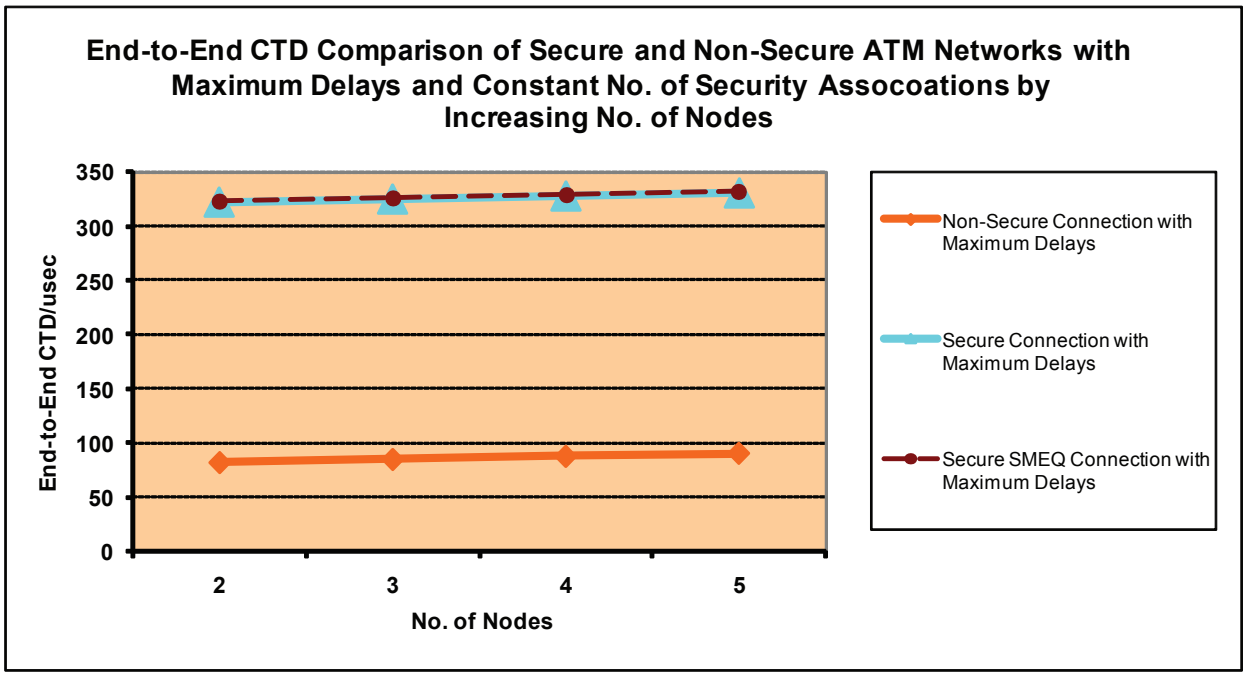


Figure 13.9 End-to-end CTD Comparison for Secure and Non-Secure Networks Assuming Maximum Delays by Increasing No. of Nodes and Const. No. of Sec. Associations

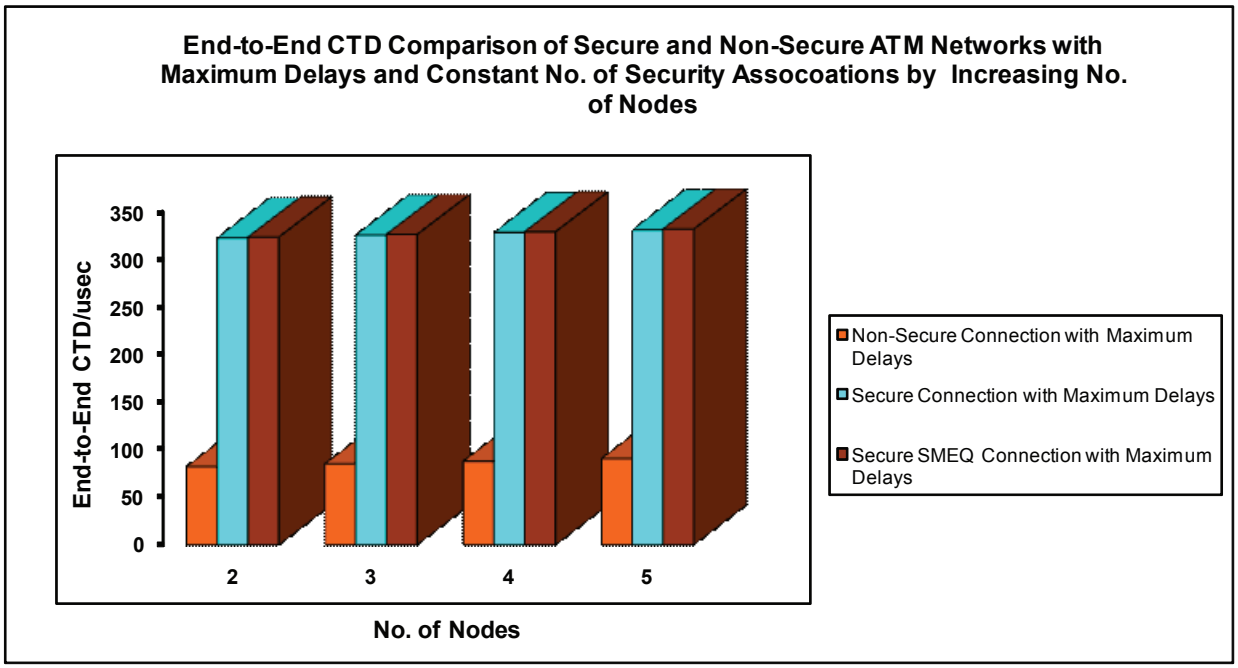


Figure 13.10 End-to-end CTD 3-D Comparison for Secure and Non-Secure Networks Assuming Maximum Delays by Increasing No. of Nodes and Const. No. of Sec. Associations

the increase ranges from 25.5% to 27.4%. In addition to confirming the conclusions from above analysis of minimum values, it shows that the overall selection of network components plays a vital roll in the importance and necessity of capturing these degradations while offering QoS in secure connections.

13.4.1 Case Study: End-to-End CTD over Increasing No. of Security Associations

Figures 13.11 through 13.16 illustrate the model network configurations and security associations for analysis scenarios no. 5 to 10.

For these scenarios the connection path of scenario no. 4 is considered with five traversing network nodes. On this path starting with two encryption devices, which implement one security association, the number of security associations are increased and corresponding necessary components are added to the connection path.

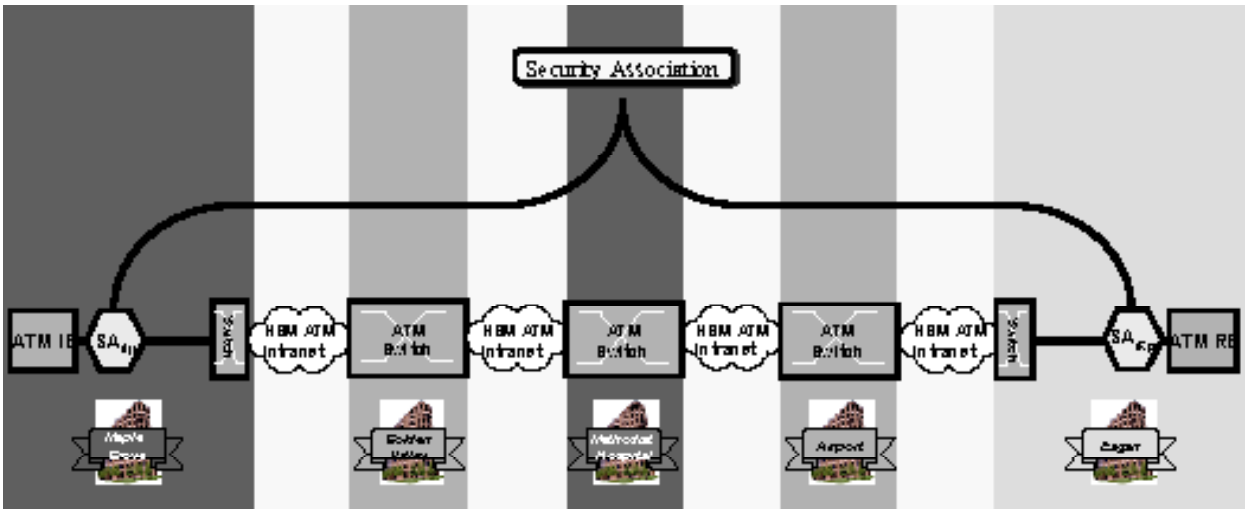


Figure 13.11 Model Network Configuration for Scenario 5

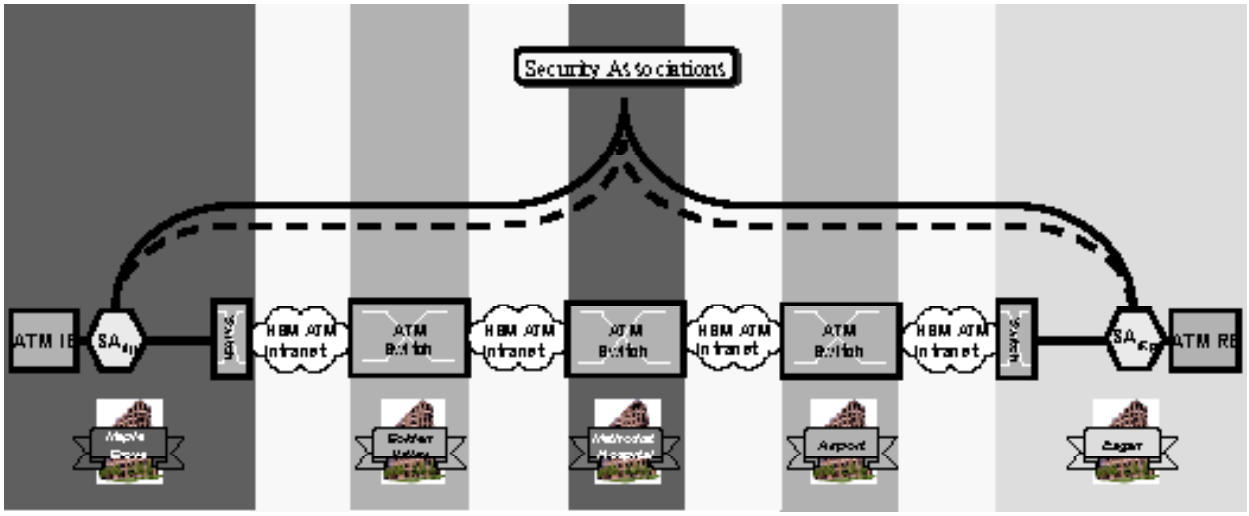


Figure 13.12 Model Network Configuration for Scenario 6

Figure 13.13 Model Network Configuration for Scenario 7

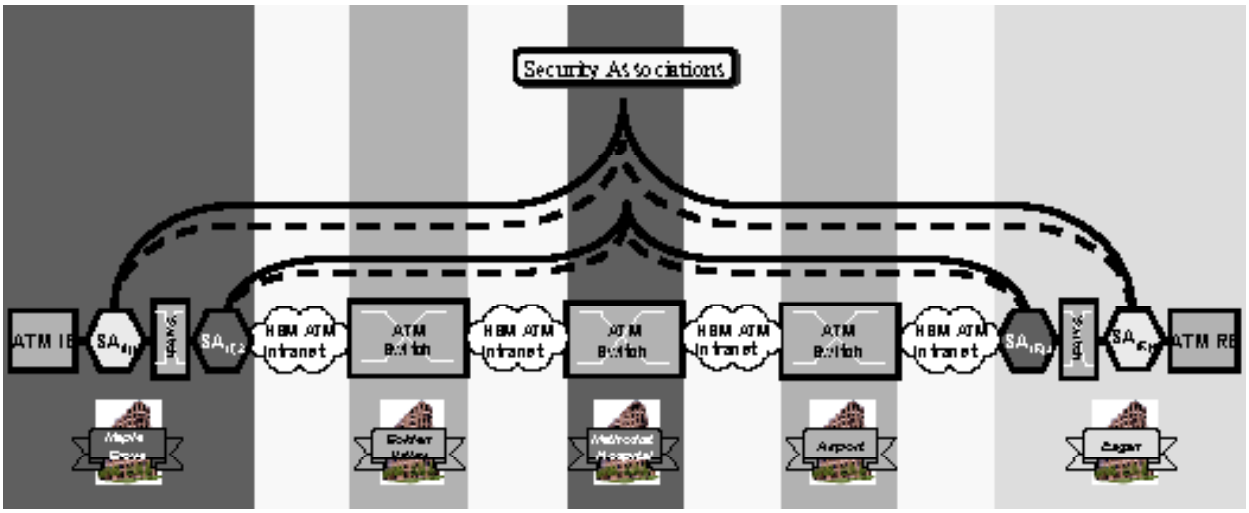


Figure 13.14 Model Network Configuration for Scenario 8

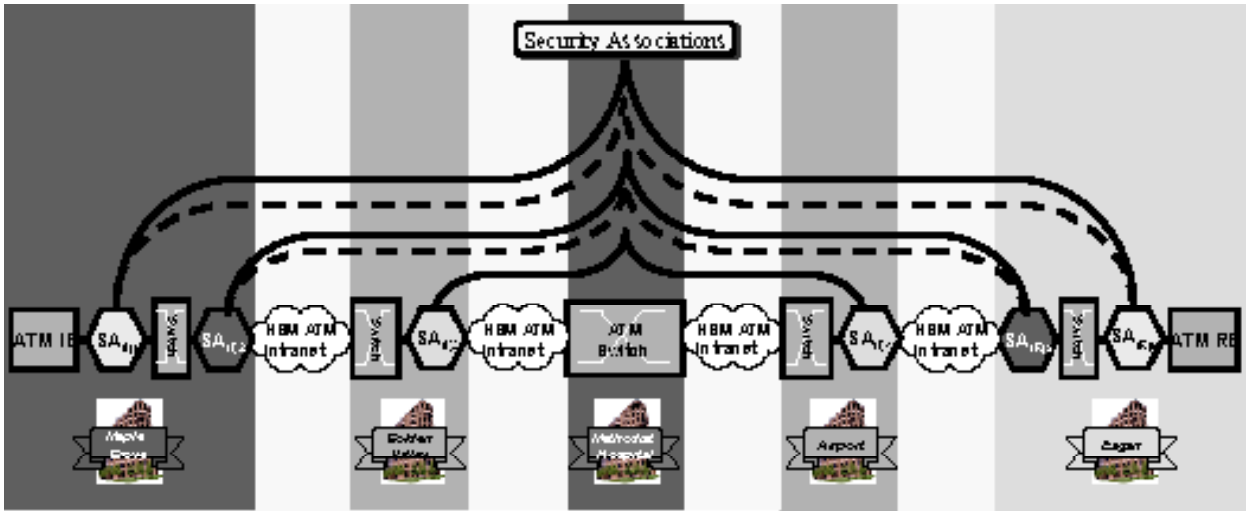


Figure 13.15 Model Network Configuration for Scenario 9

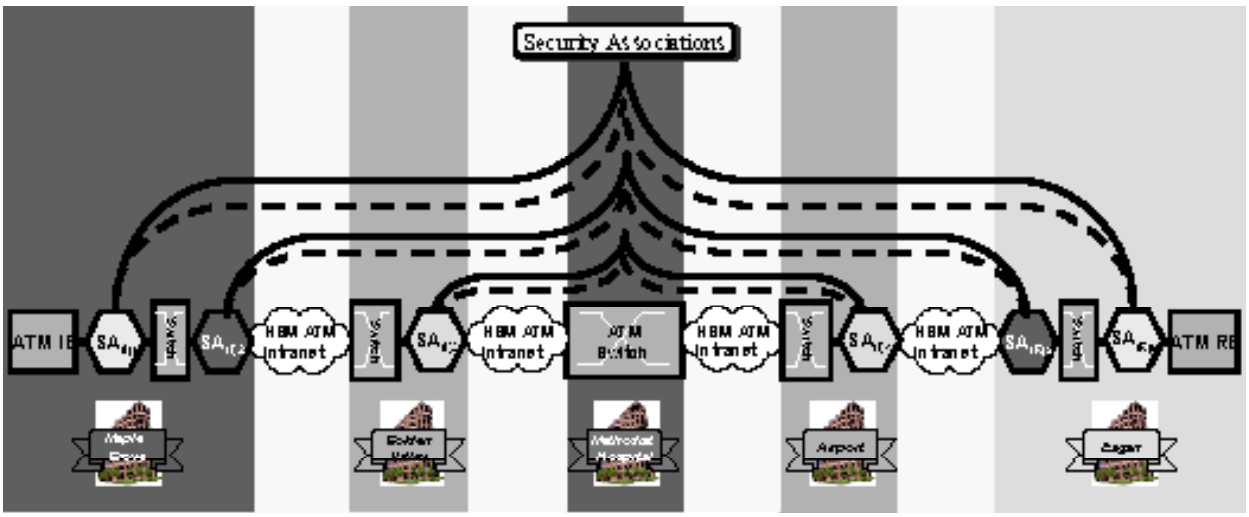


Figure 13.16 Model Network Configuration for Scenario 10

The resulting cell transfer delay behavior is studied in respect to the above-mentioned cases (non-secure, secure, SME_Q secured connections) with increasing number of security associations.

It is further assumed that in addition to its network switch each end point includes an encryptor for the absolute end-to-end security. Table 13.4 illustrates the Input data for these scenarios. As depicted having the same number of switches in the network path the network delays are constant. The resulting cell transmission delay caused by security operations, however, increase according to the number of security associations and the type of encryption devices used. This figure ranges from 30µs to 360µs for the considered configurations.

Scenario #	Communication Path	No. of Nodes	Distant km	No. of ATM Switches	No. of ATM Encryptor	No. of Security Assoc.	Total EndPoint $CTD_{E,E}$ in µs for MPEG-2 Codec rates in Mbps				Total end-to-end CTD_{NTWK} of the network nodes (Switch) in µs	CTD_{sec} Secure Connection for encrptor's CTD(I,R) in µs		
							10	15	50	80				
							38.4	25.6	7.68	4.8	2.8	15	24	30
5	Maple Grove to Eagan (MG --> EG)	5	82.00	5	2	1	76.80	51.20	15.36	9.60	14.00	30.00	48.00	60.00
6	Maple Grove to Eagan (MG --> EG)	5	82.00	5	2	2	76.80	51.20	15.36	9.60	14.00	60.00	96.00	120.00
7	Maple Grove to Eagan (MG --> EG)	5	82.00	5	4	3	76.80	51.20	15.36	9.60	14.00	90.00	144.00	180.00
8	Maple Grove to Eagan (MG --> EG)	5	82.00	5	4	4	76.80	51.20	15.36	9.60	14.00	120.00	192.00	240.00
9	Maple Grove to Eagan (MG --> EG)	5	82.00	5	6	5	76.80	51.20	15.36	9.60	14.00	150.00	240.00	300.00
10	Maple Grove to Eagan (MG --> EG)	5	82.00	5	6	6	76.80	51.20	15.36	9.60	14.00	180.00	288.00	360.00

Table 13.4 Input Data for Scenarios 5 to 10

Table 13.5 displays the resulted values corresponding to these input data for two cases of non-secure transmission and secure transmission. Table 13.6 illustrates these values for the case that SME_Q is implemented to establish the secure connection while offering the QoS.

These resulting values for the minimum delays are demonstrated according to the assumed input values in Figures 13.17 and 13.18. As depicted security operations have a very high impact on the QoS of the networks.

Scenario #	Communication Path	Total end-to-end CTD for a Non-Secure Connection in µs for MPEG-2 Codec rates in Mbps				Total end-to-end CTD Secure Connection in µs for encrptor's CTD(I,R) in µs and MPEG-2 Codec rates in Mbps											
		10	15	50	80	15				24				30			
		38.40	25.6	7.68	4.8	38.40	25.60	7.68	4.80	38.40	25.60	7.68	4.80	38.40	25.60	7.68	4.80
5	Maple Grove to Eagan (MG --> EG)	90.80	65.20	29.36	23.60	120.80	95.20	59.36	53.60	138.80	113.20	77.36	71.60	150.80	125.20	89.36	83.60
6	Maple Grove to Eagan (MG --> EG)	90.80	65.20	29.36	23.60	150.80	125.20	89.36	83.60	186.80	161.20	125.36	119.60	210.80	185.20	149.36	143.60
7	Maple Grove to Eagan (MG --> EG)	90.80	65.20	29.36	23.60	180.80	155.20	119.36	113.60	234.80	209.20	173.36	167.60	270.80	245.20	209.36	203.60
8	Maple Grove to Eagan (MG --> EG)	90.80	65.20	29.36	23.60	210.80	185.20	149.36	143.60	282.80	257.20	221.36	215.60	330.80	305.20	269.36	263.60
9	Maple Grove to Eagan (MG --> EG)	90.80	65.20	29.36	23.60	240.80	215.20	179.36	173.60	330.80	305.20	269.36	263.60	390.80	365.20	329.36	323.60
10	Maple Grove to Eagan (MG --> EG)	90.80	65.20	29.36	23.60	270.80	245.20	209.36	203.60	378.80	353.20	317.36	311.60	450.80	425.20	389.36	383.60

Max Delays
Min Delays

Table 13.5 End-to-end CTD Outsome for Secure and Non-Secure Networks by Increasing No. of Sec. Associations and Const. No. of Nodes

Scenario #	Communication Path	Total end-to-end CTD_{BMEG} for a Secure Connection in µs using SME_Q protocol (+ overhead) for delays in µs caused by different MPEG-2 Codec rates: Assumption: 1: 20% of the encrptor's delay is for processing of the security protocol 2: SME_Q adds 7 bytes of overhead to the existing ATMSECSME protocol											
		0.40				0.63				0.79			
		38.40	25.60	7.68	4.80	38.40	25.60	7.68	4.80	38.40	25.60	7.68	4.80
5	Maple Grove to Eagan (MG --> EG)	121.20	95.60	59.76	54.00	139.20	113.60	77.76	72.00	151.20	125.60	89.76	84.00
6	Maple Grove to Eagan (MG --> EG)	151.20	125.60	89.76	84.00	187.20	161.60	125.76	120.00	211.20	185.60	149.76	144.00
7	Maple Grove to Eagan (MG --> EG)	181.20	155.60	119.76	114.00	235.20	209.60	173.76	168.00	271.20	245.60	209.76	204.00
8	Maple Grove to Eagan (MG --> EG)	211.20	185.60	149.76	144.00	283.20	257.60	221.76	216.00	331.20	305.60	269.76	264.00
9	Maple Grove to Eagan (MG --> EG)	241.20	215.60	179.76	174.00	331.20	305.60	269.76	264.00	391.20	365.60	329.76	324.00
10	Maple Grove to Eagan (MG --> EG)	271.20	245.60	209.76	204.00	379.20	353.60	317.76	312.00	451.20	425.60	389.76	384.00

Table 13.6 End-to-end CTD Outsome for Secure Networks using SME_Q Protocol by Increasing No. of Sec. Associations and Const. No. of Nodes

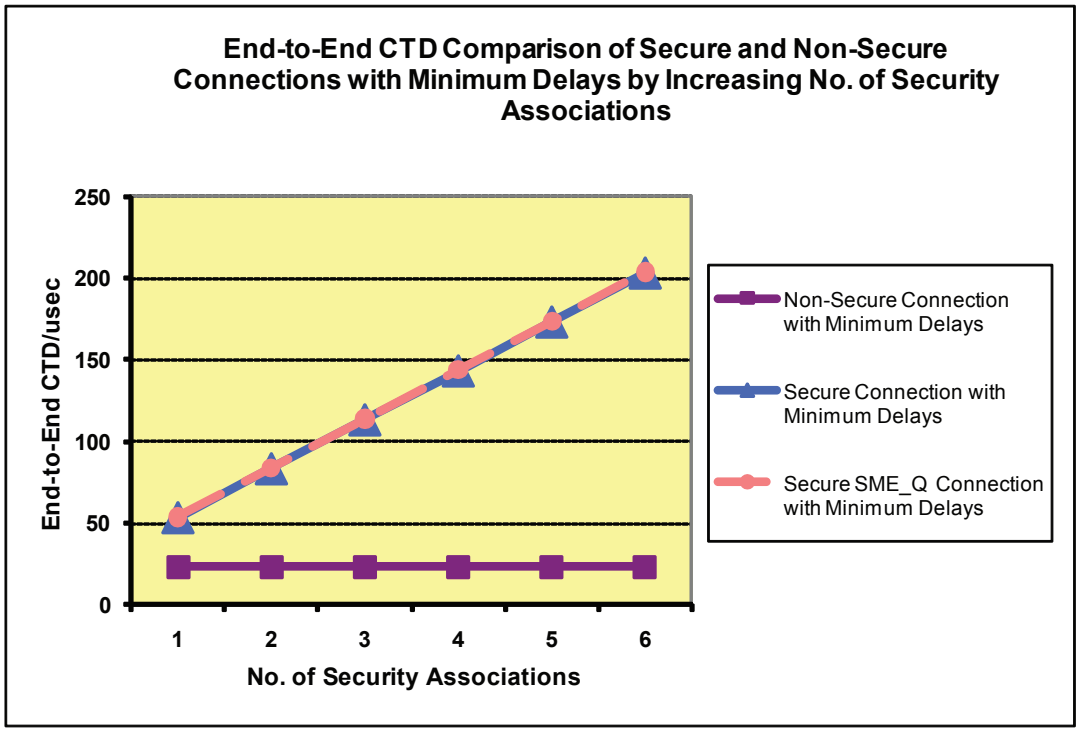


Figure 13.17 End-to-End CTD Comparison for Secure and Non-Secure Networks Assuming Minimum Delays by Increasing No. of Sec. Associations and Const. No. of Nodes

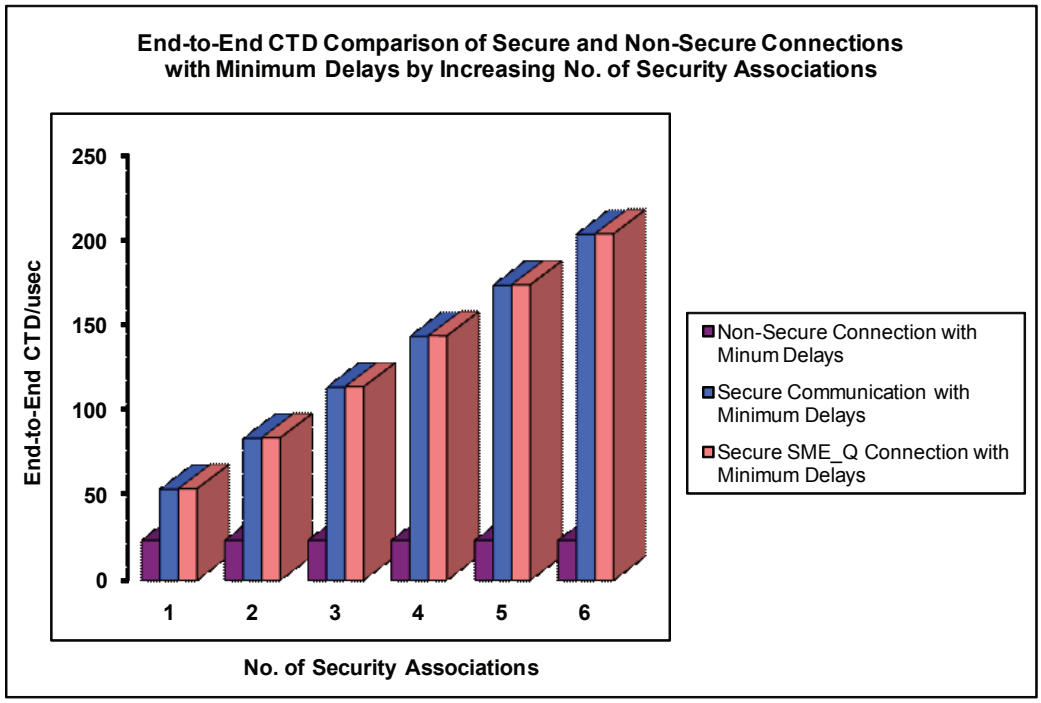


Figure 13.18 End-to-end CTD 3-D Comparison for Secure and Non-Secure Networks Assuming Minimum Delays by Increasing No. of Sec. Associations and Const. No. of Nodes

The resulting values of cell transmission delay for the non-secure network connection are constant for all scenarios for a given codec devices rate. They only increase per connection for decreasing coding rates of these devices.

The scenarios demonstrate a huge increase of total cell transmission delays caused by security operations for increasing number of security associations in the range from 227.1% to 862.7% in compare with the non-secure connections.

This analysis confirms the enormous necessity for measures to consider these degradations of QoS in new protocols specially by increasing requirements for tighten security measures for a given network, hence the in this work presented SME_Q protocols.

According to the assumptions described earlier in this chapter, the additional overhead of the proposed SME_Q protocol is 2.6% in compare with the existing security protocols. This would be a minimal toll to pay to consider the range of discrepancy from 227.1% to 862.7% in compare with the achieved values of QoS, which today is not considered in secure communications. In addition to considering these degradations, the SME_Q protocol offers mechanisms to negotiate and consider security algorithms with smaller amount of contributing degradations, which can among others consequently decrease the number of network interruptions in extreme cases.

Figures 13.19 and 13.20 illustrate the same exercise assuming the highest delay values in the networks. The resulted degradation values from security operations here also show also a high increase from 166.1% to 496.5%. These results are not as high as the above case because of the greater difference between the minimum and maximum coding rates of the codecs in relation to the stepwise increase of the number of security associations. These figures, in addition to confirming the conclusions from above analysis of minimum values, they also show that the overall selection of network components plays a vital roll in the importance and necessity of capturing these degradations while offering QoS in secure connections.

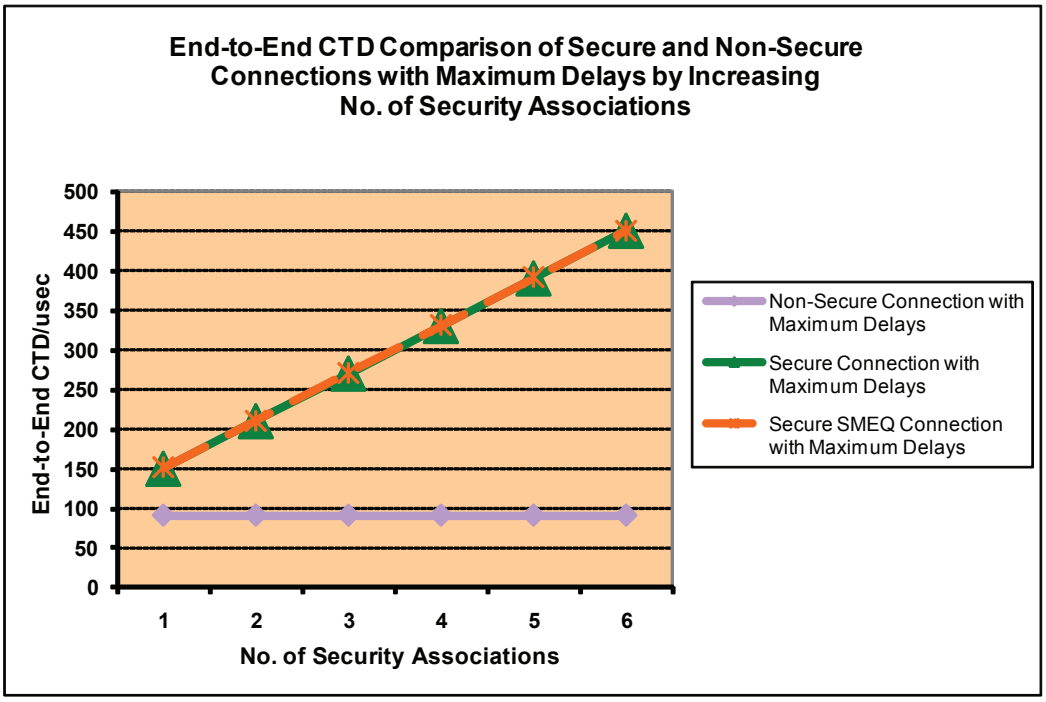


Figure 13.19 End-to-end CTD Comparison for Secure and Non-Secure Networks Assuming Maximum Delays by Increasing No. of Sec. Associations and Const. No. of Nodes

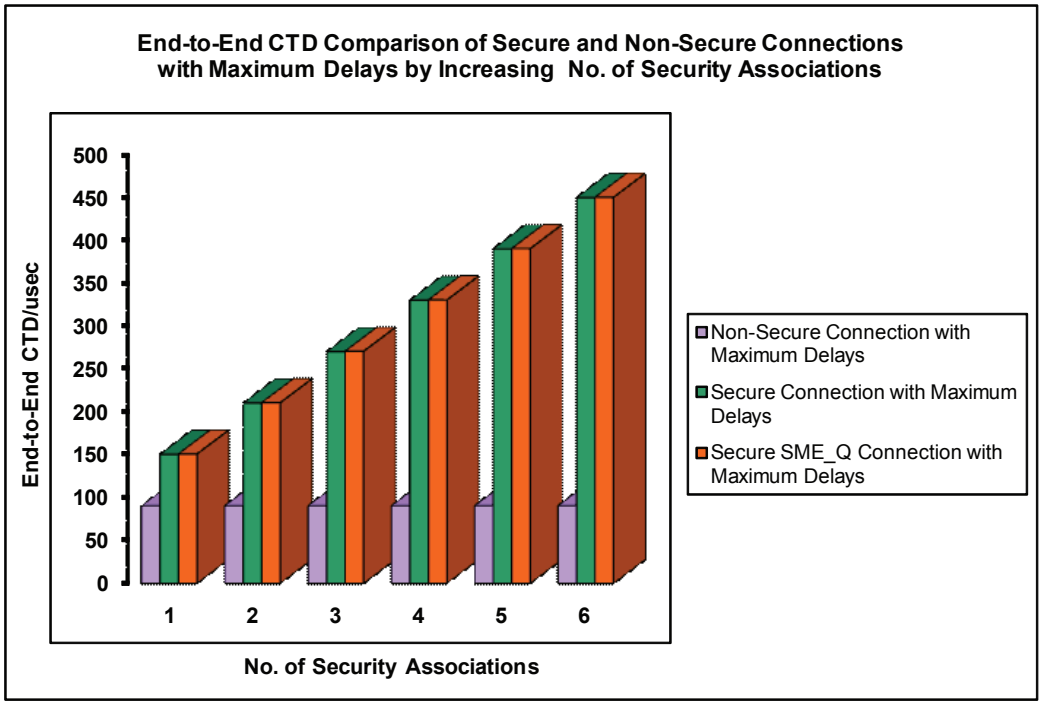


Figure 13.20 End-to-end CTD 3-D Comparison for Secure and Non-Secure Networks Assuming Maximum Delays by Increasing No. of Sec. Associations and Const. No. of Nodes

Chapter

14

Conclusion & Future Research

14.1 Conclusion and Significant Contributions

In this research work, we have designed and proposed new security protocols to take the degradation of QoS based on security operations into account, while aiming optimality for a secure connection integrating the social behavioral and economic characteristics for real-world implementations.

In the first 9 Chapters, we have proposed delay-efficient security protocols by implementing game and mechanism design theoretic principles, a cross-layer protocol, DSIC-S, and a layer-specific for the network layer in the IP paradigm [RFC_4301][RFC_5996], IPsec-O.

The protocols are incentive compatible and individually rational. We have proved that these protocols are network-wide socially desirable and Pareto optimal. They are consumer-centric and guarantee the delivery of consumer's security services within desired levels, naturally and trustfully. The proposed nesting security model reinforces this goal. We have addressed resource management and delay-efficiency through synergy of several design aspects. We have proposed a scenario-based security model with different security levels. We have incorporated a valuation system to integrate

the caused delay at each node in selection of security algorithms without consumer's knowledge of the actual delays. We have achieved this by incorporating the valuation and preference system, in particular that of the consumer's, in the calculation of the Vickrey-Clarke-Groves (VCG) payments [ViWi_61]with Clarke's pivotal rule [ClEd_71] and the credit transfers. We have proved that this enforces independence of private types. This way, DSIC-S and IPsec-O designs solve the revelation theory problem of misrepresentation of agents' private information in mechanism design theory and achieve delay-efficiency in a security protocol. To counteract the selfishness, we also proposed an incentive model and incorporated the valuation and rankings in the incentives.

The simulations have confirmed the theoretical results. They confirmed the suitability of our valuation system, scenario-based structure and incentive model to enforce the desired outcome. We can see the correlation between the credit transfers and actual delays, and that of the incentives to the valuations.

In Chapters 10-13, we have proposed yet another layer-specific security protocol, the SME_Q, for the datalink layer as an example of and based on ATM. Through our proposed protocol, the desired Quality of Service of an ATM connection will no longer have to be sabotaged by 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 as shown in a real-life ATM network simulation. In addition, we have also developed an extensive simulation software, SMEQSIM, to simulated ATM security negotiations and the possibility of incorporating the QoS degradations.

The developed Out-Band and In-Band SME_Q protocols are extensions to the *ATM Security Specification Version 1.1*[SEC_11]. These protocols were further expanded to satisfy the two special cases of nesting and sequenced security associations for each case. The protocols propose extensions to the current security association section of the SSIE to carry the Traffic QoS degradation of the selected security mechanism for the respective security agent. Within the design, we have proposed that these degradation values be available for each implementation of SA in the SA Characteristics with regard to QoS (SAC_Q) Table, as shown in Figure 11.1.

As analyzed and shown in the quantitative study of a real-life example of an ATM network in Chapter 13, the scenarios demonstrate an increase of total cell transmission delays caused by security operations for a given number of security associations in the range from 11.2% to 16.2% considering minimum device delays and 25.5% to 27.4% considering maximum device delays in compare to the non-secure connections for the assumed network model. The results also confirm that his incongruity increases with increasing number of network nodes. The same study demonstrates an even higher increase of total cell transmission delays caused by security operations for increasing number of security associations for the same connection in the range from 227.1% to 862.7% (considering minimum device delays) in compare with the non-secure connections for the assumed network model.

This analysis confirms the enormous necessity for measures to consider these degradations of QoS in new protocols, especially by increasing requirements for tighten security measures for a given network; hence, the in this work presented SME_Q protocols. The existing standardized security protocols to date, however, do not take these degradations of requested and agreed upon Quality of Service caused by security implementations into account.

14.2 Future Research

The study of the bounded rationality and its impact to achieve such an optimal consumer-centric and delay-efficient security protocol would be an interesting continuation of this research work and thought process. Study of the impact of repeated games and cooperation on DSIC-S would be a great extension as well.

Another good direction for future research would bethe implementation of DSIC-S as an extension to the existing security protocols at different layers. Other interesting areas of application would be VPNs, Cloud Computing, and different areas of wireless communications. An additional great idea would be in the area of reputation modeling and design. Following a study of how reputations would influence rational behavior in real world, one could design a proprietary reputation model based on the calculated incentives of DSIC-S.

A good direction for the network-layer specific future research would be the proposal of an extension to the current IntServ's Guaranteed Service [RFC_2212]and RSVP protocol [RFC_2205] for the implementation of IPsec-O.

Appendix A

References

ANSI_00 American National Standards Institute, www.ansi.org

AtCr_03 ATMedia GmbH, "ATM Crypt Data Sheet", www.atmedia.de,2003.

ATM_00 The ATM Forum, (www.atmforum.com) now: www.broadband-forum.org.

ATM_99 ATM Forum, "Speaking Clearly with ATM - A practical guide to carrying voice over ATM",1999.

ATT_11 AT&T, www.business.att.com.

BAX_11 M. Barua, M.S. Alam, L. Xiaohui, S. Xuemin, "Secure and Quality of Service assurance scheduling scheme for WBAN with Application to eHealth", IEEE Wireless Communications and Networking Conference,WCNC, 2011.

BeGa_92 D. Bertsekas, R. Gallager, "Data Networks", Second edition, Prentice Hall, 1992.

BHLLO_07 S. Bohacek, J. P. Hespanha, J. Lee, C. Lim, and K. Obraczka, "Game Theoretic Stochastic Routing for Fault Tolerance and Security in Computer Networks", IEEE Transaction on Parallel and Distributed Computing, Vol. 18, No. 9, September 2007.

BiEl_99 E. Biham,"A Note Comparing the AES Candidates", Isreal Institute of Technology, comments submitted to NIST, 1999.

BiMm_02 I. Bisio, M. Marchese, "QOS-Constrained MOP-Based Bandwidth Allocation Over Space Networks", Satellite Communications and Navigation Systems, ISBN: 978-0-387-47522-6 , December 2007.

BlUy_00 U. Black, "QoS in Wide Area Networks", Prentice Hall, 2000.

BlUy_91 U. Black, "OSI, A Model for Computer Communications Standards", Prentice Hall, 1991.

BrMa_09 M. Bryan, "The Economic Benefits of Smart Grids and the Role of Communications", Alcatel-Lucent Australia, EEA Conference & Exhibition 2009, Christchurch, June 2009.

CaCo_03 C. F. Camerer, "Behavioral Game Theory, Experiments in Strategic Interaction", Princeton University Press, ISBN 0-691-09039-4, 2003.

CeSa_00 Cellware Broadband, "Cell-Safe Data Sheet", www.cellware.de, 2000.

ChLi_99 Ray-Guang Cheng, Chung-Ju Chang and Li-Fong Lin,"A QoS-Provisioning Neural Fuzzy Connection Admission Controller for Multimedia High-Speed Networks", IEEE/ACM Transactions on Networking, Vol. 7, No. 1, February 1999.

Cj_2004 J. L. Cohen, "Multiobjective Programming And Planning", Dover Publications, 2004.

ClEd_71 E. H. Clarke, "Multi-Part Pricing of Public Goods", Journal of Public Choice, p. 11, 17-23, 1971.

ClEd_80 E. H. Clarke, "Demand Revelation and the Provision of Public Goods", Ballinger Publishing Company, ISBN 0-88410-686-1, 1980.

ClLj_09 L. Chen, and J. Leneutre, " A Game Theoretical Framework on Intrusion Detection in Heterogeneous Networks", IEEE Transactions On Information Forensics and Security, Vol. 4, No. 2, June 2009.

CNET_01 CNET, "Broadband Milestones and Developments in Singapore", www.cnet.com, 2001.

CsWm_10 S. Chen and M. Wu, "Game Theoretic Approach in Multipath Routing for Tradeoff between Routing Security and Performance", Proceedings of the 2010 14th International Conference on Computer Supported Cooperative Work in Design, April 2010.

CvHy_83 V. Chankong, Y. Y. Haimes, "Multiobjective Decision Making: Theory and Methogology", Elsevier Science Publishing Co., ISBN:0-444-00710-5, 1983.

CWLP_11 City Water Light & Power, Springfield, illinois, www.cwlp.com.

CWZ_99 W. Chen, M. M. Wiecek, and J. Zahng, "Quality Utility - A Compromise Programming Approach to Robust Design", Journal of Mechanical Design, Vol 121, pp. 179-187, June 1999.

CyCe_99 CTAM, "CypherCell Specifications",www.ctam.com.au,1999.

CyLn_04 CYLINK (SAFENET), "ATM Encryptor User's Guide", www.cylink.com, 2004.

DBP_96 H. Dobbertin, A. Bosselaers, and B. Preneel, "RIPEMD-160: A strengthened version of RIPEMD", Proceedings of 3rd International Workshop on Fast Software Encryption, Springer-Verlag, p. 71-82, 1996.

DiDj_98 I. Das, and J. Dennis, "Normal-Boundary Intersection: A New Method for Generating Pareto Optimal Points in Multicriteria Optimization Problems", SIAM Journal on Optimization, Vol. 8, No. 3, p. 631–657, August 1998.

DmDr_96 M.A. Dimand, R. W. Dimand,"A History of Game Theory, Volume 1", Routledge, ISBN 0-415-07257-3, 1996.

DOE_05 DOE, "A Summary of Control System Security Standards Activities in the Energy Sector", DOE: Office of Electricity Delivery and Energy Reliability, October 2005.

DOE_09 DOE, "Study of Security Attributes of Smart Grid Systems – Current Cyber Security Issues", DOE: Office of Electricity Delivery and Energy Reliability, April 2009.

Dr_76 R. A. Van Dusseldorp et al, "Applications of Goal Programming to Education", AEDS, May 1976.

EAH_10 D. S. Abd Elminaam, H. M. Abdual Kader, M. M. Hadhoud, "Evaluating The Performance of Symmetric Encryption Algorithms", International Journal of Network Security, Vol.10, No.3, PP.213-219, May 2010.

ElGa_85 T. Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms", IEEE Transactions on Information Theory, IT–31, 469–472, 1985.

ETSI_00 European Telecommunications Standards Institute, www.etsi.org

FeHu_98 P. Ferguson, G. Huston, "Quality of Service, Delivering QoS on the Internet and in Corporate Networks", Wiley Computer Publishing, 1998.

FGH_98 E. C. Fink, S. Gates, B. D. Humes, "Gamer Theory Topics, Incomplete Information, Repeated Games, N-Player Games", SAGE Publications, ISBN 0-7619-1016-6, 1998.

FIPS_463 Federal Information Processing Standards, "Data Encryption Standard (DES)", NIST, FIPS PUB 46-3, October 1999.

FIPS_197 Federal Information Processing Standards, "Advanced Encryption Standard (AES)", NIST, FIPS PUB 197, November 2001.

FIPS_1801 Federal Information Processing Standards, "Secure Hash Algorithm (SHA-1)", NIST, FIPS PUB 180-1, April 1995.

FIPS_1803 Federal Information Processing Standards, "Secure Hash Standard (SHS)", NIST, FIPS PUB 180-3, October 2008.

FIPS_1861 Federal Information Processing Standards, "Digital Signature Standard (DSS)", NIST, FIPS PUB 186-1, December, 1998.

GiAp_10 I. K. Geckil, P. L. Anderson, "Applied Game Theory and Strategic Behavior", CRC Press, 2010.

GoMa_98 M. Goncalves, "FIREWALLS Complete", McGraw–Hill, 1998.

GoPa_98 R. Gopalakrishnan and G. M. Parulkar, "Efficient User-Space Protocol Implementations with QoS Guarantees Using Real-Time Upcalls", IEEE/ACM Transactions on Networking, Vol. 6, No. 4, August 1998.

GRR_83 T. Groves, R. Radner, S. Reiter, "Information, Incentives, and Economic Mechanisms", University of Minnesota Press, ISBN 0-8166-1340-0, 1983.

GrRo_76 R. E. Grieson, "Public and Urban Economics, Essays in Honor of William S. Vickrey", D.C. Health and Company, ISBN 0-669-98400-0, 1976.

GrTh_73 T. Groves, "Incentives in Teams", Econometrica,p. 41, 6187-631, 1983.

GT_10 Georgia Tech, "Optimization in Engineering Design", Systems Realization Lab Presentation Slides.

HaJC_67 J.C. Harsanyi, “Games with Incomplete Information Played by “Bayesian” Players, I-III”, Management Science Journal, 1967.

HiNa_09 I. Haugen, A. Nilsen,”Game Theory: Strategies, Equilibria, and Theorems”, Nova Science Publishers, Inc., ISBN 978-1-60456-844-8, 2009.

HsCl_04 Park Nicollet Health Services,”Clinic Locations Website”, www.parknicollet.com /clinic/clinic_locations.cfm, 2004.

HSM_97 ATM Forum, “Case Study: Health System of Minnesota”, 1997.

HsPn_98 NORDX/CDT Cable Design Technoloies,”Real World Applications:Health System MinnesotaPark Nicollet Clinic”,1998.

HsVy_95 S. P. Hargreaves Heap, Y. Varoufakis, “Game Theory, A Critical Introduction”, Routledge, ISBN 0-415-09402-X, 1995.

HuGe_00 Geoff Huston, Internet Performance Survival Guide, QoS Strategies for Multiservice Networks, Wiley, 2000.

Hy_09 Y. Y. Haimes, “Risk Modeling, Assessment, and Management”, Wiley, ISBN: 978-0-470-28237-3, 2009.

IETF_00 The Internet Engineering Task Force, www.ietf.org.

ISO_00 International Organization for Standardization, www.iso.org/iso/en/ISOOnline.frontpage.

ISO_10116 International Organization for Standardization, “Modes of operation for an n-bit block cipher”,1997.

ITU_00 International Telecommunication Union, www.itu.int/home.

ITUI_321 ITU-T Recommendation I.321, “B-ISDN Protocol Reference Model and its Application”, April 1991.

ITUI_356 ITU-T Recommendation I.356, “B-ISDN ATM Layer Cell Transfer Performance”, March 2000.

ITUI_361 ITU-T Recommendation I.361, “B-ISDN ATM layer specification”, February 1999.

ITUI_3632 ITU-T Recommendation I.363.2, “B-ISDN ATM Adaptation Layer specification: Type 2 AAL”, November 2000.

ITUI_3633 ITU-T Recommendation I.363.3, “Type 3/4 AAL”, August 1996.

ITUQ_2931 ITU–T, “ITU-T Recommendation Q.2931, Digital Subscriber Signalling System No. 2 - User-Network Interface (UNI) layer 3 specification for basic call/connection control”, February 1995.

ITUQ_2971 ITU–T Recommendation Q.2931, “Digital Subscriber Signalling System No. 2 - User-Network Interface (UNI) layer 3 specification for point-to-multipoint call/connection control”, February 1995.

ITUT_00 ITU Telecommunication Standardization Sector (ITU-T), www.itu.int/ITU-T

JoHe_08 H. L. Jones II, “Public Wireless for the Smart Grid Secure, Scalable, Reliable, Smart Synch Inc.”, 2008.

JuNe_10 Juniper Networks, “Juniper Networks Smart Grid Networking Solution”, Solution Brief, January 2010.

KaMi_96 Micheal E. Kabay, Ph.D., "The NCSA Guide to Enterprise Security, Protecting Information Assets", McGraw–Hill, 1996.

KaPeSp_95 Charlie Kaufman, Radia Perlman, Mike Speciner,"Network Security, Private Communication in a public World", Prentice Hall 1995.

KiWo_04 I.Y. Kim, O.L. de Weck, "Adaptive weighted-sum method for bi-objective optimization: Pareto Front Generation", MIT, Dept. of Aeronautics & Astronautics, Engineering Systems Division, Springer-Verlag, 2004.

KkSs_06 K. Kang, S. H. Son, "Toward Security and QoS Optimization in Real-Time Embedded Systems", ACM SIGBED Review, Volume 3 , Issue 1, January 2006.

KuSv_02 Dipl-Ing. Sven Kuhn, "Einfluss der Verschluesselung auf Quality of Service in ATM-Netzen bei Betriebsarten ohne zusaetzlichen Bandbereitenbedarf", February 2002.

LeRo_10 R. Leonard, "Von Neumann, Morgenstern, and the Creation of Game Theory", Cambridge University Press, ISBN 978-0-521-56266-9, 2010.

LoPe_99 Pete Loshin, "IPv6 Clearly Explained", Morgan Kaufmann Publishers, 1999.

LZY_05 P. Liu, W. Zang, and M. Yu, "Incentive-Based Modeling and Inference of Attacker Intent, Objectives, and Strategies", ACM Transactions on Information and System Security, Volume 8 Issue 1, February 2005.

MaDi_05 D. Maringer, "Advances in Computational Management Science Portfolio Management with Heuristic Optimization", Springer US, ISBN: 978-0-387-25852-2, pages 38-76, Volume 8, 2005.

MaMc_02 A. Messac, C. A. Mattson, "Generating Well-Distributed Sets of Pareto Points for Engineering Design using Physical Programming", Optimization and Engineering, Vol. 3, pp. 431–450, December 2002.

MaMc_04 A. Messac, C. A. Mattson, "Normal Constraint Method with Guarantee of Even Representation of Complete Pareto Frontier", Rensselaer Polytechnic Institute, Multidisciplinary Design and Optimization Lab, AIAA Journal, 2004.

McDa_00 D. McDysan, "QoS & Traffic Management in IP & ATM Networks", McGraw–Hill, 2000.

McSo_99 D. McDysan, D. Spohn, "ATM Theory and Applications", McGraw–Hill, 1999.

MdS_10 D.C. F. Mackie, Stone, "Optimization Techniques", Sam Houston State University, www.shsu.edu.

MeA_96 A. Messac, "Physical Programming: Effective Optimization for Computational Design", AIAA Journal, Vol. 34, No. 1, pp. 149–158, January 1996.

MIM_03 A. Messac, A. Ismail-Yahaya, C. A. Mattson, "The Normalized Normal Constraint Method for generating the Pareto Frontier", Rensselaer Polytechnic Institute, Structural and Multidisciplinary Optimization, Vol. 25, No. 2, pp. 86-98, 2003.

MiMa_00 M. A. Miller,"Implementing IPv6, Supporting the Next Generation Internet Protocols", M&T Books, 2000.

MPW_11 Muscatine Power and Water, www.mpw.org.

MRC_04 Marconi Press Release, www.marconi.com, 2004.

MSL_10 Y. Mao, B. Sun, D. Li,"Design and Implementation of a trust-based QoS Management Scheme", 2nd IEEE International Conference on Network Infrastructure and Digital Content, 2010.

MyRb_97 R.B. Myerson, "Game Theory: Analysis of Conflict", Harvard University Press, Massachusetts, 1997.

MyRb_12 R.B. Myerson, "Roger B. Myerson - Autobiography", Nobelprize.org. http://www.nobelprize.org/nobel_prizes/economics/laureates/2007/myerson-autobio.html, 2012.

NaJm_05 A. Nadeem, M.Y. Javed, "A Performance Comparison of Data Encryption Algorithms", First International Conference on Information and Communication Technologies, ICICT 2005.

NBDFR_00 J. Nechvatal, E. Barker, L. Bassham, W. Burr, M. Dworkin, J. Foti, E. Roback, "Report on the Development of the Advanced Encryption Standard (AES) ", Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, October, 2000.

NCDC_11 North Carolina Department of Commerce, North Carolina, The State of Minds: Utilities-Telecommunications, www.nccommerce.com.

NGNP_09 Y. Narahari, D. Garg, R. Narayanam, H. Prakash, "Game Theoretic Problems in Network Economics and Mechanism Design Solutions", Springer; 2nd Printing edition, ISBN-10: 1848009372, ISBN-13: 978-1848009370, February, 2009.

NgTs_05 Ng, C.Y. Tse, H.M. Lok, T.M., "A goal programming model and schemes for channel assignment in general downlink transmission system", ICC2005, May 2005.

NISTCM_07 NIST," Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC", NIST Special Publication 800-38D, November 2007.

NiJaZh_97 K. Nichols, V. Jacobson, and L. Zhang, "A Two-bit Differentiated Services Architecture for the Internet", December 1997.

NRTV_08 N. Nisan, T. Roughgarden, E. Tardos, V. Vazirani, "Algorithmic Game Theory", Cambridge University Press, ISBN 978-0-521-87282-9, 2008.

OMWDB_08 H. Otrok, N. Mohammed, L. Wang, M. Debbabi, and P. Bhattacharya, " A Moderate to Robust Game Theoretical Model for Intrusion Detection in MANETs", IEEE International Conference on Wireless & Mobile Computing, Networking & Communication, 2008.

PNNI_10 ATM Forum, "PNNI Signaling Specification Version 1.0", af–pnni–0055.000, March 1996.

RFC_1321 R. Rivest, "The MD5 Message-Digest Algorithm", IETF, April 1992.

RFC_1633 R. Braden, D. Clark, S. Shenker, etal, "Integrated Services in the Internet Architecture: an Overview", June 1994.

RFC_2205 R. Braden, L. Zhang, S. Berson, etal, "Resource ReSerVation Protocol (RSVP) — Version 1 Functional Specification", September 1997.

RFC_2210 J. Wroclawski, MIT LCS, "The Use of RSVP with IETF Integrated Services", September 1997.

RFC_2211 J. Wroclawski, MIT LCS," Specification of the Controlled-Load Network Element Service", September 1997.

RFC_2212 S. Shenker, C. Partridge, R. Guerin, etal, "Specification of Guaranteed Quality of Service", September 1997.

RFC_2215 S. Shenker, J. Wroclawski, Xerox PARC/MIT LCS, "General Characterization Parameters for Integrated Service Network Elements", September 1997.

RFC_2216 S. Shenker, J. Wroclawski, Xerox PARC/MIT LCS, "Network Element Service Specification Template", September 1997.

RFC_2401 S. Kent, R. Atkinson, etal, "Security Architecture for the Internet Protocol", November 1998.

RFC_2402 S. Kent, R. Atkinson, etal, "IP Authentication Header", November 1998.

RFC_2406 S. Kent, R. Atkinson, etal, "IP Encapsulating Security Payload (ESP)", November 1998.

RFC_2460 S. Deering, R. Hinden, etal, "Internet Protocol, Version 6 (IPv6) Specification", December 1998.

RFC_2474 K. Nichols, S. Blake, F. Baker, etal, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", December 1998.

RFC_2475 S. Blake, D. Black, M. Carlson, etal, "An Architecture for Differentiated Services", December 1998.

RFC_2990 G. Huston, Telstra, "Next Steps for the IP QoS Architecture", November 2000.

RFC_4301 S. Kent, K. Seo, "Security Architecture for the Internet Pro-tocol" , IETF RFC 4301, December 2005.

RFC_4302 S. Kent, R. Atkinson, etal, "IP Authentication Header", December 2005.

RFC_4303 S. Kent, "IP Encapsulating Security Payload (ESP)", IETF RFC 4303, December 2005.

RFC_5996 C. Kaufman, P. Hoffman, Y. Nir, P. Eronen, "Internet Key Exchange Protocol Version 2 (IKEv2)", IETF RFC 5996, ISSN: 2070-1721, September 2010.

RFC_791 Information Sciences Institute of USC, "Internet Protocol, DARPA Internet Program Protocol Specification", September 1981.

RiKl_02 K. Ritzberger, "Foundations of Non-Cooperative Game Theory", Oxford University Press, ISBN 0-19-924785-4, 2002.

RoYi_97 M.J.B. Robshaw, Yiqun Lisa Yin, "Elliptic Curve Cryptosystems, An RSA Laboratories Technical Note", Revised June 1997.

RSA_11 RSA Labratories, www.rsa.com/rsalabs, 2011.

RSA_78 R. L. Rivest, A. Shamir, L. M. Adleman, "A method for obtaining digital signatures and public key cryptosystems", Communications of ACM, 21(2): 120-126, February 1978.

RuSc_99 C. Ruland, "Kryptographische Verfahren & Anwendungen", 1999.

RuSEC_99 C. Ruland, "ATM Security", half–day Tutorial, Annual Computer Security Applications Conference (ACSA), 1999.

SaMe_97 G. C. Sackett, C. Y. Metz, "ATM and Multiprotocol Networking", McGraw–Hill, 1997.

Sane_04	SafeNet,"SafeEnterprise Security System Specifications",www.safenet-inc.com, 2004.
Sb_11	Bruce Schneier, http://www.schneier.com/twofish.html, 2011.
SbWd_00	B.Schneier, D. Whiting, "A Performance Comparison of the Five AEC Finalists",Counterpance Internt Security, Inc., March 2000.
ScMi_88	M. Schwartz, "Telecommunication Networks: Protocols, Modeling and Analysis", Addison–Wesley Publishing Company, 1988.
SDGE_06	SDG&E, "Capital Budget Workpapers Information Technology", Sandiego Gas & Electric, December 2006.
SEC_10	ATM Forum,"ATM Security Specification Version 1.0", af-sec-0100.001, February 1999.
SEC_11	ATM Forum, "ATM Security Specification Version 1.1", af-sec-0100.001, March 2001.
SECAdd_11	ATM Forum, "Security Services Renegotiation Addendum to Security Version 1.1", fb-sec-0180.000, March 2002.
SfMi_111	F. Schlake, L. Mili, "Efficient Network Security as a Strategic Game", to be published in the International Journal of Critical Infrastructures, 2012.
SfMi_112	F. Schlake, L. Mili, "IPsec-O, Optimal, Delay-Efficient and Cheat-Proof; A Mechanism Design Theoretic Approach", to be submitted for peer-review, 2012.
SfRc_02	F. Schlake, C. Ruland, "A Security Protocol Providing QoS in ATM Networks", The 8th International Conference on Communication Systems, ICCS2002, IEEE conference, IEEE Catalog: 02EX585, Vol. 2, p. 933-937, November 2002.
ShPa_01	P. Shaw, "E-Business Privacy And Trust, Planning and Management Strategies", John Wiley & Sons, Inc., ISBN 0-471-41444-1, 2001.
SiKa_10	K. Sigmund, "The Calculus of Selfishness", Princeton University Press, ISBN 978-0-691-14275-3, 2010.
SiTl_00	Rohde&Schwarz, "SITLine ATM Product Brochure", www.sit.rohde-schwarz.com, 2004.
SM_95	Marc J. Schneiderjans, "Goal Programming Methodology and Applications", Kluwer Academic Publishers, ISBN:0-7923-9558-1, 1995.
SnNy_08	N. Rama Suri and Y. Narahari,"Design of an Optimal Bayesian Incentive Compatible Broadcast Protocol for Ad Hoc Networks with Rational Nodes", IEEE Journal of Selected Areas in Communications, Vol. 26, No. 7, September 2008.
SpHaPa_91	J.D. Spragins, J.L. Hommand, K. Pawlikowski, "Telecommunications Protocols and Design", Addison Wesley, 1991.
StR_86	R. Steuer, "Multiple Criteria Optimization: Theory, Computation, and Application", John Wiley & Sons, New York, Chapter 3, 1986.
TgBj_08	G. Theodorakopoulos and J. S. Baras, "Game Theoretic Modeling of Malicious Users in Collaborative Networks", IEEE Journal of Selected Areas in Communications, Vol. 26, No. 7, September 2008.

TMS_40 ATM Forum, "Traffic Management Specification Version 4.0", af-tm-0056.000, April 1996.

TMS_41 ATM Forum, "Traffic Management Specification Version 4.1", af-tm-0121.000, March 1999.

UAr_11 The University of Arizona, "Arizona Telemedicine Program", www.telemedicine.arizona.edu.

USEC_98 ATM Forum, "UNI Signaling Specification 4.0 Security Addendum Draft", BTD_CS_UNI_SEC_01.06, December 1998.

USIG_31 ATM Forum, "UNI Signaling Specification 3.1", af-uni-0010.002.

USIG_40 ATM Forum, "UNI Signaling Specification 4.0", af-sig-0061.000, July 1996.

USIG_41 ATM Forum, "UNI Signaling Specification 4.1",af-sig-0061.001 April 2002.

Ver_11 Verizon, www22.verizon.com.

ViWi_61 W. Vickrey. "Counterspeculation, auctions, and competitive sealed tenders", Journal of Finance,16, 8–37, 1961.

WaBe_98 B. Walke, "Mobilfunknetze und ihre Protokolle", B.G. Teubner Stuttgart, 1998.

WeJa_07 J. N. Webb, "Game Theory, Decisions, Interactions, Evolution", Springer-Verlag London Limited, ISBN-10 1-8428-423-6, 2007.

WeJn_07 James N. Webb, "Game Theory, Decisions, Interactions and Evolution", Springer-Verlag, ISBN-13: 978-1-84628-423-6, London, 2007

WiSt_08 S. R. Williams, "Communication in Mechanism Design, A Different Approach", Cambridge University Press, ISBN 978-0-521-85131-2, 2008.

WoKa_02 K. Wollenweber, "Einfluss der Verschluesselung auf Quality of Service in ATM-Netzen bei Betriebsarten mit zusaetzlichem Bandbereitenbedarf", May 2002.

YaKi_99 K. K. Yackovich," Muscatine Utility Meets New Service Demands", UTC Journal, October 1999.

ZhZh_08 Y. Zhu, W. Zhang, "Trust model regarding QoS using FCMs and D-S Evidence Theory", IEEE International Conference on Service Operations and Logistics, 2008.

ZTE_09 ZTE Corporation, "ZTE Electric Utility Communications Transmission Network Solution", wwwen.zte.com.cn, July 2009.