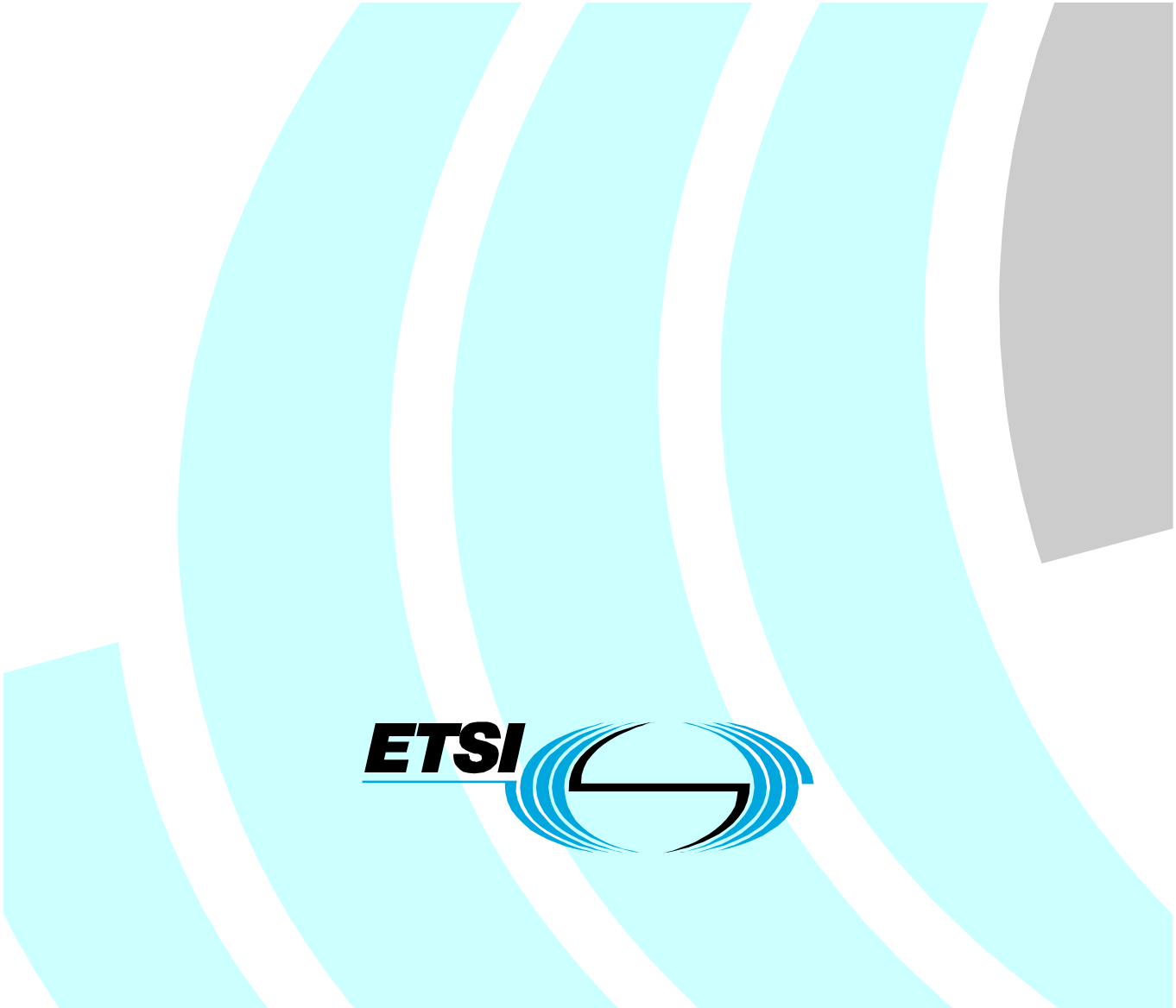
ETSI EG 202 549 V1.1.1 (2006-12)

ETSI Guide

Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN);
Design Guide;
Application of security countermeasures to service capabilities



Reference
DEG/TISPAN-07004-Tech

Keywords
internet, IP, protocol, security, service, VoIP

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This ETSI Guide (EG) has been produced by ETSI Technical Committee Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN).

Introduction

The present document is one of a set of documents that addresses standardization of security protocols and mechanisms within the context of the *e*Europe 2005 programme. The suite of documents in this suite is composed as follows:

- EG 202 387: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Security Design Guide; Method for application of Common Criteria to ETSI deliverables".
- ES 202 383: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Security Design Guide; Method and proforma for defining Security Targets".
- ES 202 382: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Security Design Guide; Method and proforma for defining Protection Profiles".
- TS 102 165-1: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Protocol Framework Definition; Methods and Protocols for Security; Part 1: Threat Analysis".
- TS 102 165-2: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Protocol Framework Definition; Methods and Protocols for Security; Part 2: Counter Measures".
- DTS/TISPAN-07008-Tech: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Protocol Framework Definition; Protection Profile".
- EG 202 549: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Design Guide; Application of security countermeasures to service capabilities".

These documents are developed based on the objectives of the *e*Europe programme and are also developed to ensure they comply with the overall objectives of the European regulatory framework as defined in the following documents:

- Directive 2002/19/EC of the European Parliament and of the council of 7 March 2002 on access to, and interconnection of, electronic communications networks and associated facilities (Access Directive).
- Directive 2002/20/EC of the European Parliament and of the council of 7 March 2002 on the authorisation of electronic communications networks and services (Authorisation Directive).
- Directive 2002/21/EC of the European Parliament and of the council of 7 March 2002 on a common regulatory framework for electronic communications networks and services (Framework Directive).
- Directive 2002/22/EC of the European Parliament and of the council of 7 March 2002 on universal service and users' rights relating to electronic communications networks and services (Universal Service Directive).

- Directive 2002/58/EC of the European Parliament and of the council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector (Directive on privacy and electronic communications).

In particular the present document forms part of the standardization initiative for the Next Generation Network (NGN) platform to be used in *e*Europe and upon which the trust and viability of the *e*-enabled community will, to a very large part, depend on.

The *e*Europe 2005 action plan has been drawn up to focus on "*the widespread availability and use of broadband networks throughout the Union … and the security of networks and information, eGovernment, eHealth and eBusiness*" requiring a supporting infrastructure, which is truly pan-European. To quote COM(2002)263: "*By 2005 Europe should have … a secure information infrastructure*".

1 Scope

The present document gives guidance on the application of security countermeasures to service capabilities. It covers the construction of services from service capabilities and how a security evaluation of a service capability should be performed. The present document examines and gives guidance on the use of the Composition assurance class defined by the Common Criteria working group in order to be able to answer the question: "if components A and B are evaluated as having security ratings X and Y what is the security rating that can be assigned to the combination of A and B?"

The present document builds on the guidance to the Common Criteria for Information Technology Security Evaluation given in EG 202 387 [3] with a particular view to assessing the security of the NGN. In the NGN context, where services are not explicitly defined but are made from combining service capabilities, the present document gives guidance on the means to apply effective security to both service capabilities in isolation, and to service capabilities in combination.

The guidance reviews the service capability model in clause 4 and examines the requirements for security arising from the service capability requirements defined for NGN-R1 in clause 5. The analysed security requirements are presented in the form of ISO/IEC 15408-2 [17] functional models. Clause 6 presents a review of the Common Criteria Composition assurance class and describes its impact on the ETSI standardization process. Annex A reviews the use of cryptographic techniques in the NGN.

A number of assumptions of the design of NGN for security analysis to take place are made on the NGN development process. The assumption in the present document is that the NGN has been developed using top-down decomposition of the specification, using techniques of planned validation of the specification, with careful recording of design decisions and validation results.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

[1] ETSI TR 181 004: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); NGN Generic capabilities and their use to develop services".

[2] ETSI TR 181 003: "Telecommunications and Internet Converged Services and Protocols for Advanced Networking (TISPAN); Services capabilities, requirements and strategic direction for NGN services ".

[3] ETSI EG 202 387: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Security Design Guide; Method for application of Common Criteria to ETSI deliverables".

[4] ETSI TS 102 165-1 (2003): "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 4; Protocol Framework Definition; Methods and Protocols for Security; Part 1: Threat Analysis".

[5] ETSI TS 102 165-2 (2003) "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 4; Protocol Framework Definition; Methods and Protocols for Security; Part 2: Counter Measures".

[6] ETSI TS 133 210: "Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); 3G security; Network Domain Security (NDS); IP network layer security (3GPP TS 33.210 version 7.0.0 Release 7)".

[7] ETSI TS 133 203 (V7.0.0): "Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); 3G security; Access security for IP-based services (3GPP TS 33.203 version 7.0.0 Release 7)".

[8] ETSI TR 101 878: "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 5; Service Capability Definition; Service Capabilities for a Multi Media Call".

[9] ETSI TR 101 882 (V5.1.1): "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) Release 5; Protocol Framework Definition and Interface Requirement Definition; General".

[10] ETSI EG 202 238: "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON); Evaluation criteria for cryptographic algorithms".

[11] ETSI EG 202 107: "Methods for Testing and Specification (MTS); Planning for validation and testing in the standards-making process".

[12] ETSI EG 201 015: "Methods for Testing and Specification (MTS); Specification of protocols and services; Validation methodology for standards using Specification and Description Language (SDL); Handbook".

[13] ETSI ETR 184: "Methods for Testing and Specification (MTS); Overview of validation techniques for European Telecommunication Standards (ETSs) containing SDL".

[14] ITU-T Recommendation X.509: "Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks".

[15] ITU-T Recommendation I.130: "Method for the characterization of telecommunication services supported by an ISDN and network capabilities of an ISDN".

[16] ITU-T Recommendation I.210: "Principles of telecommunication services supported by an ISDN and the means to describe them".

[17] ISO/IEC 15408-2: "Information technology - Security techniques - Evaluation criteria for IT security - Part 2: Security functional requirements".

[18] ISO/IEC 15408-3: "Information technology - Security techniques - Evaluation criteria for IT security - Part 3: Security assurance requirements".

[19] ISO/IEC 15408: "Information technology - Security techniques - Evaluation criteria for IT security".

NOTE: When referring to all parts of ISO/IEC 15408 the reference above is used.

[20] W. Diffie and M.E. Hellman: "New directions in cryptography", IEEE Transactions on Information Theory, IT-22: 644-654, 1976.

[21] "Common Criteria Portal": http://www.commoncriteriaportal.org.

[22] "Common Criteria for Information Technology Security Evaluation - Part 3: Security assurance components" July 2005 Version 3.0 Revision 2.

[23] IETF RFC 4306: "Internet Key Exchange (IKEv2) Protocol".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

example 1: text used to clarify abstract rules by applying them literally

NOTE: This may contain additional information.

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

ACO	Assurance Composition Class
ADV	Assurance class DeVelopment
AES	Advanced Encryption Standard
AH	Authentication Header
AKA	Authentication and Key Agreement
ALC	Assurance class Life Cycle
ASE	Assurance class Security target Evaluation
ASN.1	Abstract Syntax Notation
ATE	Assurance class TEsting
CA	Certification Authority
CAP	Composition Assurance Level
CC	Common Criteria
EAL	Evaluation Assurance Level
ESP	Encapsulating Security Payload
IKE	Internet Key Exchange
IM	Instant Messaging
IMS	IP Multimedia Subsystem
IP	Internet Protocol
ISDN	Integrated Services Digital Network
MMS	Multimedia Messaging Services
MSC	Message Sequence Chart
NGN	Next Generation Network
OID	Object IDentifier
PKI	Public Key Infrastructure
PSTN	Public Switched Telephone Network
QoS	Quality of Service
SA	Security Association
SDL	Specification and Description Language
TOE	Target Of Evaluation
TSDS&TP	Test Suite Structure and Test Purposes
TVRA	Threat Vulnerability and Risk Analysis
UML	Unified Modelling Language

4 Service capabilities as building blocks in the NGN

4.1 General

In the NGN standardization environment services and applications are not fully standardized, rather the building blocks for services are standardized to act as a service development toolkit. As the security provided by a service composed of discretely protected service capabilities is only as good as the interaction between the service capabilities it is important to ensure that a service capability is sufficiently protected, and sufficiently clear in its operation, that when it is deployed alongside other capabilities that the system behaviour is correct and the system security is as good as can be achieved. Service capabilities are therefore designed to be re-usable with only an outline concept of where they will be reused. Some service capabilities will be designed as specializations of others but in the main are quite generic, relying on the value of their data to perform a job. For example a capability to establish a point-to-point connection for duplex voice communication (a telephone call) may be little different from the capability to establish a point-to-point connection for duplex video communication (a videophone call) so a single capability to "establish point to point connection for streaming media" may be sufficient where the details of the media connection are held in the data. This may make the job of a service designer simpler but may complicate that of the security designer when compared with "stovepipe" application development.

A service is, however, not only a composition of service capabilities but also has data and service logic that determines how the service capabilities are joined. In many instances the service logic and much of the service data will not be subject to standardization. Furthermore, as a result of this there is no certainty that two services composed of service capabilities A, B and C will act in the same way.

NOTE: Service capabilities may not be the only capabilities defined in the NGN.

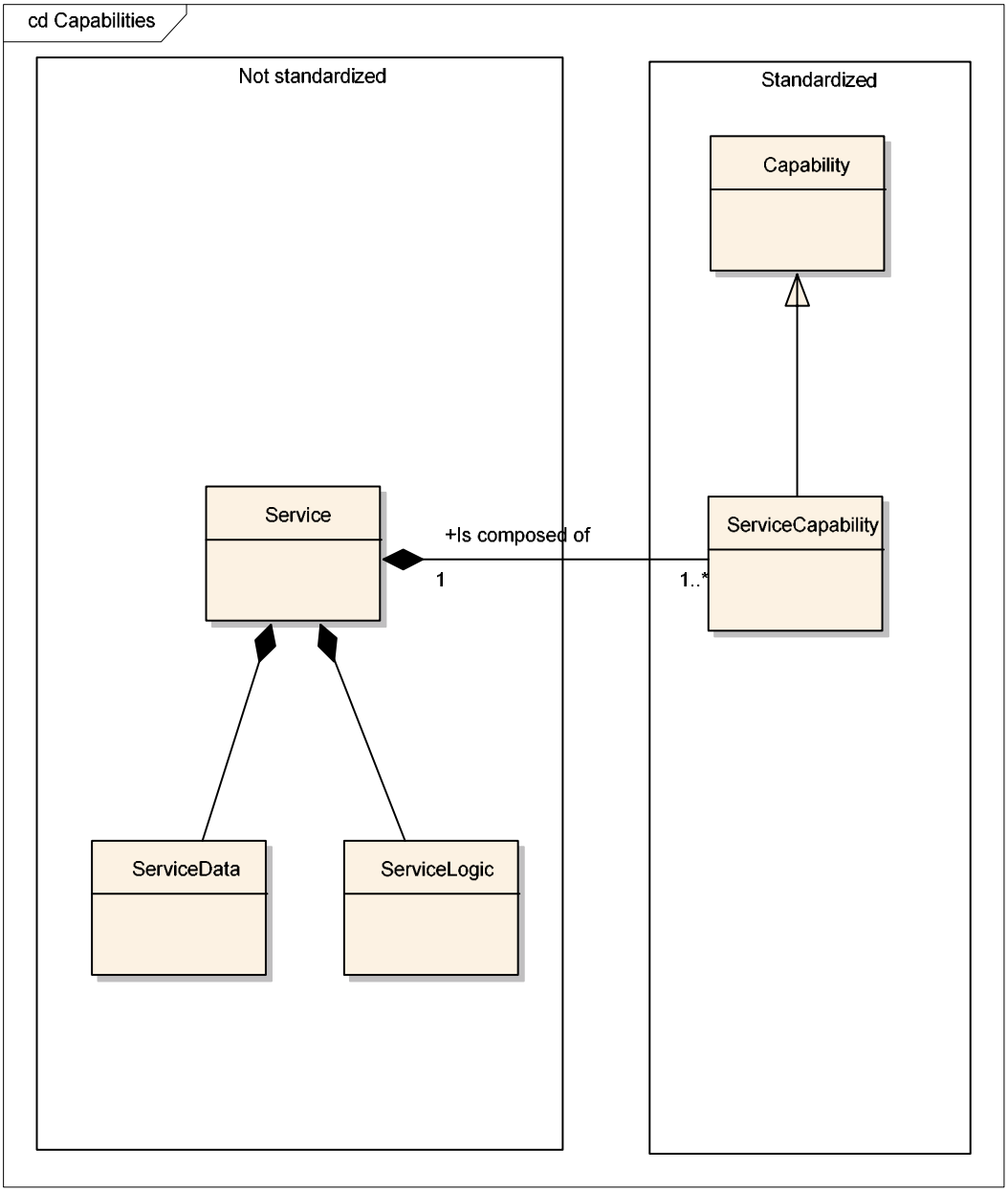


Figure 1: The service, service data and service logic are not standardized, the service capability is

4.2 Security requirements with respect to service capabilities

The security question that needs to be addressed in complex systems such as the NGN is of the form: "if components A and B are evaluated as having security ratings X and Y what is the security rating that can be assigned to the combination of A and B?" When the components of the system are service capabilities the question is directly applicable to the NGN when the NGN is composed of discrete combinations of these capabilities, particularly when the manner in which the capabilities are combined is not standardized.

As recommended in TS 102 165-1 [4] (*reference WI-07006*) the security requirements of any entity should be written with reference to the security requirements statements found in ISO/IEC 15408-2 [17] and which have been translated to ETSI format and interpretation in TS 102 165-2 [5] (*reference WI-07007*). When service capabilities are considered the main concern is to inhibit unauthorized use and to also inhibit unauthorized disclosure of information held by the service capability.

Each invocation of a service capability should follow the following simple guidelines:

- The <<*service capability invoking user*>> is not allowed to <<*invoke the service capability*>> prior to successful identification (FIA_UID.2).

NOTE 1: Identification may be achieved by a number of schemes.

- The <<*service capability invoking user*>> is not allowed to <<*invoke the service capability*>> prior to successful authentication (FIA_UAU.2).

NOTE 2: Authentication may be achieved by a number of schemes.

Where data is transferred between objects (instances of service capabilities) the integrity of the data should be assured and in most cases the requested service capability should not be invoked if there is any doubt in the integrity of the received data. This is particularly important for detection of attacks caused by a man-in-the-middle modifying capability invocations.

- When <<*service capability*>> transmits <<*data in a signal*>> to a user the system shall provide that user the means to detect modification anomalies (FCO_IED.1).

- When <<*service capability*>> transmits <<*data in a signal*>> to a user the system shall provide that user the means to detect deletion anomalies (FCO_IED.1).

- When <<*service capability*>> transmits <<*data in a signal*>> to a user the system shall provide that user the means to detect insertion anomalies (FCO_IED.1).

- When <<*service capability*>> transmits <<*data in a signal*>> to a user the system shall provide that user the means to detect replay anomalies (FCO_IED.1).

NOTE 3: Detection of integrity errors may be achieved by a number of schemes.

4.3 Service capability design considerations

4.3.1 General model

Service capabilities are intended to be re-used and as such have to be designed in such a way that the interface is clear and the operation is clear. It is suggested in TR 101 882 [9] that this is achieved by application of the staged design process outlined in ITU-T Recommendation I.130 [15] and ITU-T Recommendation I.210 [16] updated to reflect component rather than service design. In this way data and signalling should be separated and isolated by means of fully defined interfaces.

The guidance given in EG 202 387 [3] for development (summarized below) should stand as the guidelines for design of each service capability. The design process should be approached by deploying the following sequential activities:

- decomposition of the system into subsystems;

- decomposition of the subsystems into modules (system capabilities);

- description of the behaviour of the modules (system capabilities); and

- demonstration of correspondence between all decompositions.

In the NGN and for security analysis to take place it is necessary for the NGN development process to follow strict guidelines. These involve top-down decomposition of the specification, the use of specification languages such as SDL or UML, planned validation of the specification and the careful recording of design decisions and validation results. In addition, it is also necessary for a vulnerability analysis to be undertaken for all standards specifying security-related aspects. This analysis will provide essential information which can be used in the development of security services and requirements. The vulnerability, threat and risk analysis method documented in TS 102 165-1 [4] should be followed for each service capability.

For many NGN capabilities there is a single root capability that is extended and in general there should be no assumption that if the root capability is secure that the extended capability is also secure. In UML terms a session control class may exist in a number of specialized forms for voice calls, for video calls, for conferencing, and so forth which may have different relationships to other service capabilities in other classes. Figure 2 illustrates the problem by consideration of the authentication process which has a number of different specializations, at different depths, which have quite different means of implementation but which provide a solution to the same problem: Is the entity asserting to be Bob provably Bob?

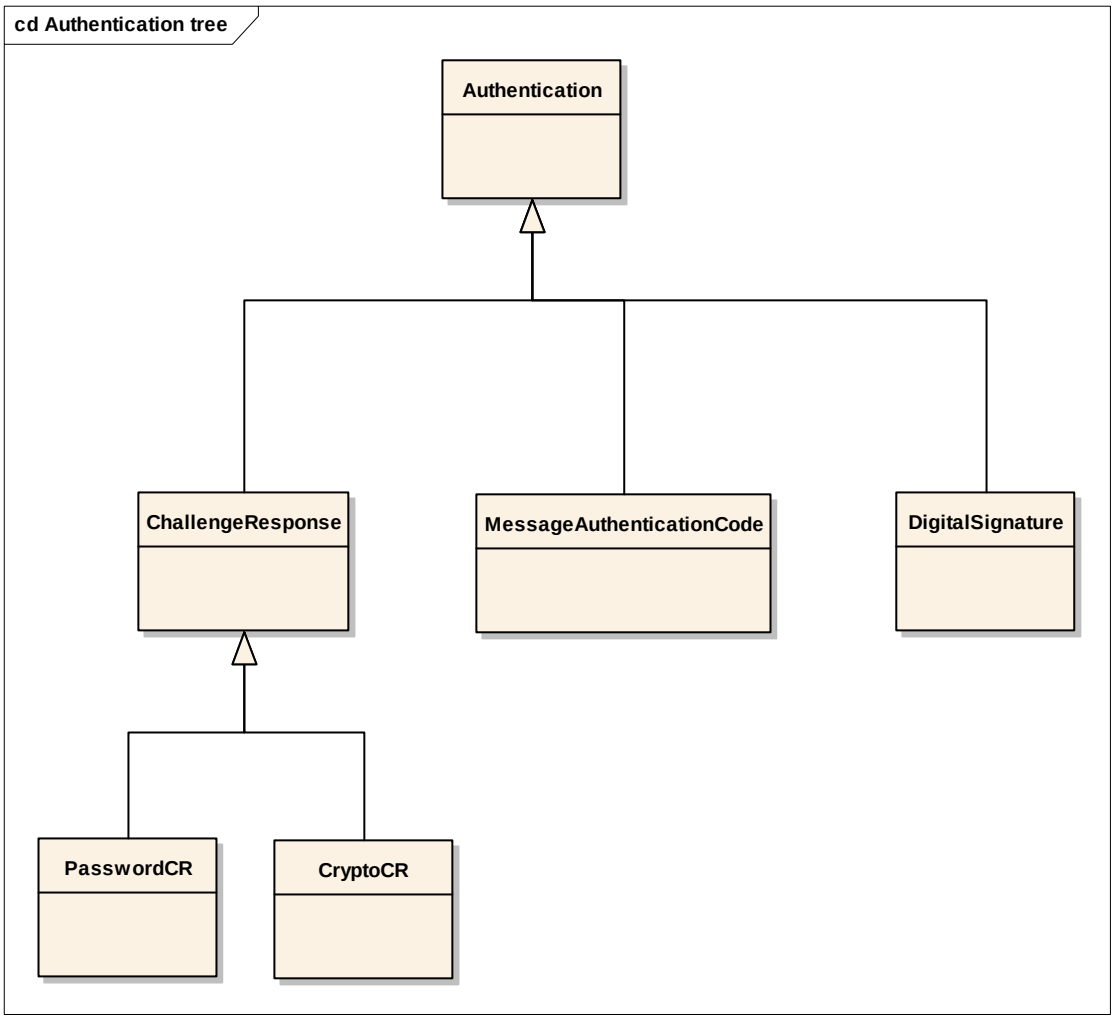


Figure 2: Specializations of authentication capability

Although the development process implied in ISO/IEC 15408-3 [18] is not identical to that generally assumed for communication standards, the mapping of activities between them is straightforward. Figure 3 shows how the component families within the CC Development class relate to specifications produced during the standardization of a communication protocol. Both the vulnerability analysis and the specification of functional requirements (equivalent to a stage 1 protocol specification) are expected to take place prior to the start of the development process itself.

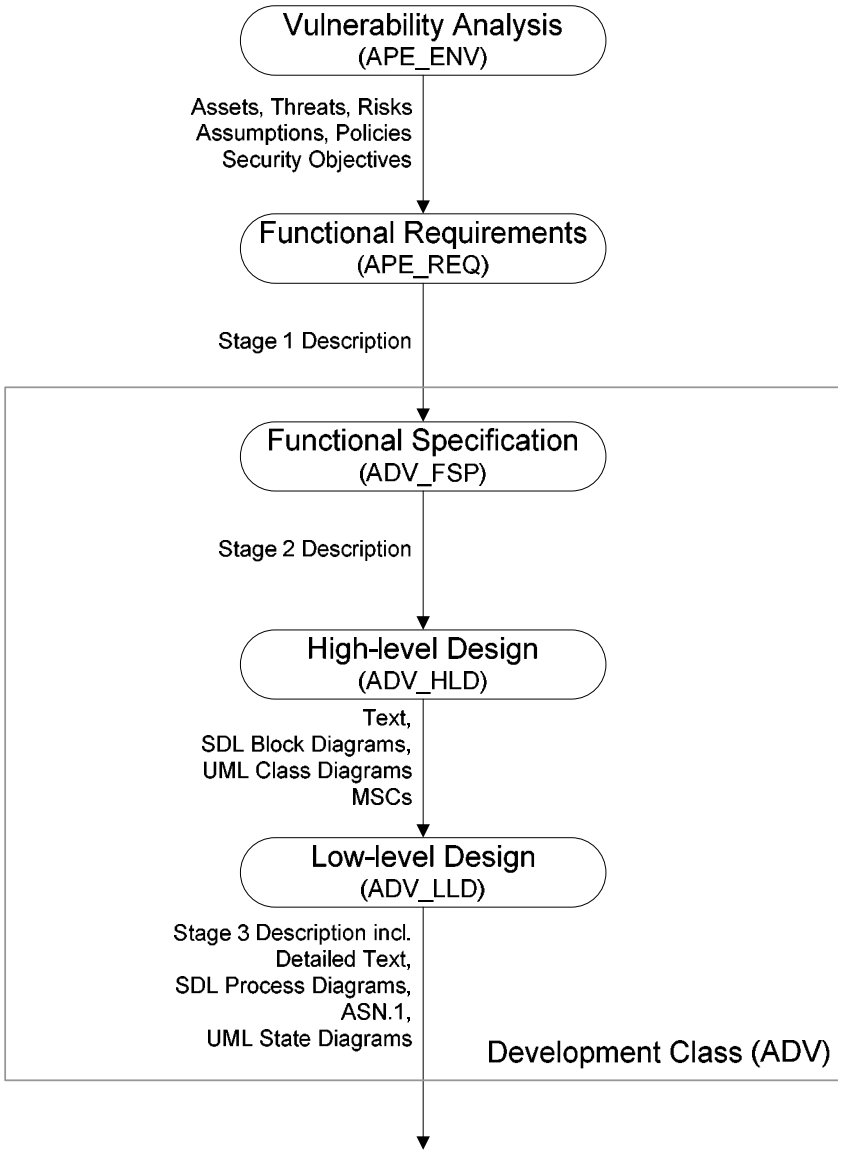


Figure 3: Relationships between CC development activities and the standardization process

4.3.2 Security countermeasures

Security tends to be considered as a suite of services rather than as service capabilities although it is possible to build complex composite services wherein the component services may be considered as capabilities. The most extreme example is probably that of non-repudiation which requires all of the basic security building blocks (authenticity, integrity, confidentiality and trust).

Where a service is composed of service capabilities as shown in figure 1 and where the service logic and service data are in the non-standardized domain then the security countermeasures have to apply to both the standardized and no-standardized domains although some common countermeasures may be applied within the set of standardized service capabilities. The present document identifies those security capabilities required to ensure that isolated service capabilities are safe to use and gives guidance on how to ensure that the deployed service maintains this level of safety/security.

5 Security analysis of NGN service capabilities

5.1 Introduction

The use of service capabilities in the NGN is considered in TR 181 004 [1] and the service capabilities requirements for the NGN are defined in TR 181 003 [2]. The service capability requirements are summarized in table 1.

NOTE: Service capabilities are also described in TR 101 878 [8] in text derived from analysis of a multimedia call, and in TR 101 882 [9] using UML to fully specify those capabilities introduced in TR 101 878 [8], which may also meet the NGN requirements.

Table 1: Service capability requirement review from TR 181 004

Service capability requirement	Short description
Communication control	basic means to establish, maintain and tear-down a communication session and performs accounting.
MMS Session Control	Provides session control for multimedia messaging systems
IM Session Control	Provides session control for Instant Message Services
CHAT Session Control	Provides session control for CHAT Services
User profile storage agent	entity that stores information about the profile data in a hierarchical way
Routing database	Static routing database
Communication Routing	Perform session routing
Logon/SignOn	Perform user logon/sign-on
Real Time charging calculation	Perform amount to charge to the user
User presence delivery	Delivers user presence information
User location delivery	Delivers user geographic location information
User presence delivery watcher	Queries for user presence information
User location delivery watcher	Queries for user geographic location information
Conference service control	Enables multiple users to have a common communication session
Profile Agents	Enable user to manipulate the user profiles
Telephony Conversation Bearer Topology	Enables a bidirectional bi-party single-media flow for narrowband audio
Telephony Conference Bearer Topology	Enables a bidirectional multi-party single-media flow for narrowband audio
Multimedia Conversation Bearer Topology	Enables a bidirectional bi-party multi-media flow of any supported type
Multimedia Conference Conversation Bearer Topology	Enables a bidirectional multi-party multi-media flow of any supported type
MMS Bearer Topology	Enables the delivery of MMS messages to the appropriate MMS end-points
Bidirectional Narrowband Conversational Voice Streaming	Ensures bidirectional narrowband audio streaming between appointed points in the network
Bidirectional Broadband Conversational Audio Streaming	Ensures bidirectional wideband audio streaming between appointed points in the network
In-call session events	Enables in-call events to be delivered with appropriately low to an appointed end-point
Unidirectional audio streaming	Ensures unidirectional wideband audio streaming from appointed points in the network to appointed end-user end-points
Unidirectional video streaming	Ensures unidirectional video streaming from appointed points in the network to appointed end-user end-points
Bidirectional conversational video streaming	Ensures bidirectional video streaming between appointed points in the network
Conference Bridge	Enables multiple media streams to be appropriately mixed
Narrowband audio stream transcoder	Transcodes between two narrowband audio streams
Media Gateway	Translates between transport mechanisms for narrowband audio
Media Streaming Forwarder	Forward media streams
Media Encryption	Encrypts/decrypts/transcripts media flows
Broadband audio stream transcoder	Transcodes between two broadband audio codecs
Video Stream Transcoder	Transcodes between two video codecs
MMS Submission	Enables the submission of MMS messages into the network
MMS Storage	Enables the storage of MMS messages in the network
MMS -type conversion	Coverts between two MMS content types
MMS format conversion	Transcodes between two MMS formats
MMS forwarding	Forwards an MMS message

Service capability requirement	Short description
MMS mass delivery	Delivers MMS messages to a list
MMS notification	Notifies the user of new MMS messages
IM forwarding	Enables the near-real time forwarding of IM messages in the network
IM storage	Enables the storage of IM messages in the network
IM delivery	Delivers Instant Messages to the recipient
MMS delivery push	Delivers MMS messages to the recipient's terminal
MMS delivery pull	Enables the recipient's terminal to retrieve MMS messages
MMS delivery streaming	Streams MMS messages to the recipient's terminal
CHAT messaging	Delivers CHAT messages to the designated chat group
CHAT private messaging	Delivers CHAT messages to the designated recipient
CHAT storage	Stores CHAT messages
UNI transport capabilities	Best-effort transport, QoS tagged packet transport, QoS enabled media transport
End to end transport capabilities	Packet switching, packet routing
Audio narrowband media presentation	Enables audio flows to be received and presented to the user
Bearer Topology end point	Enables media streams to be established as part of a communication session
Communication initiation	Enables communication sessions to be established
Communication termination	Enables a designated communication session to be terminated
IM Session client	Enables an end-point to communicate with an IM session control service capability
MMS Session Client	Enables an end-point to communicate with an MMS session control Service Capability
CHAT Session Client	Enables an end-point to communicate with an CHAT session control Service Capability
Video Media Presentation	Enables video flows to be received and presented to the user
Terminal/USIM storage of user profile	Enables the user profile for a particular service to be stored
MMS delivery control	Enables user control over MMS delivery options
MMS creation	Enables MMS creation by the user
MMS presentation	Enables MMS presentation to the user
MMS storage	Enables MMS storage in the terminal
MMS notification presentation	Enables MMS notification presentation to the user
IM creation	Enables IM creation by the user
IM presentation	Enables IM presentation to the user
IM storage	Enables IM storage in the terminal
IM notification presentation	Enables IM notification presentation to the user
Stored Message Manipulation	Enables manipulation or locally stored Instant Messages/MMS/CHAT messages by the user
CHAT session establishment/joining/leaving	Enables the user to establish/join/leave a CHAT session
CHAT Session presentation	Enables presentation of a chat session to the user
CHAT Message creation	Enables creation of a chat message by the user
CHAT invitation creation	Enables creation of a chat invitation by the user to another user
CHAT group creation	Enables the user to establish a CHAT group
User profile editing	Enables the user to edit their user profile
User sign-on	Allows the user to sign-on and authenticate to the network
User presence setting	Enables the user set their presence values
User location setting	Enables the user set their geographic location values
Media encryption	Enables media encryption/decryption
Media Transport	Enables media to be stored in packets and sent
QoS tagging	Enables packets to be tagged for appropriate QoS
Transport packet encryption	Enables packets to be encrypted

When a service is built it may be built by combining service capabilities together and a number of examples are given in clause 6 of TR 181 004 [1]. Figure 4 shows the relevant Service Capabilities when an NGN makes a call to a user on the PSTN. The straight lines indicate direct communication. In this scenario the Bearer Topology Service Capability has the simple job of selecting the appropriate media gateway.

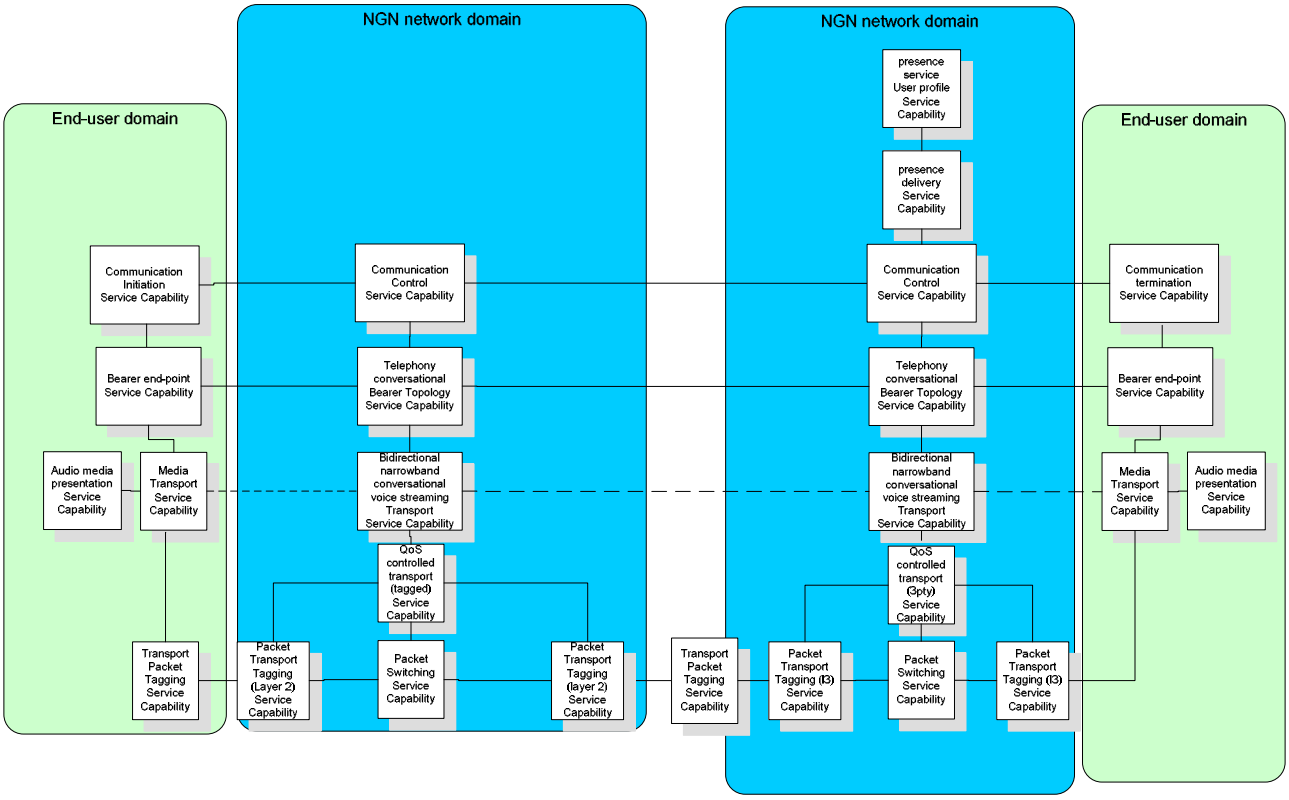


Figure 4: NGN QoS controlled telephone service (from TR 181 004, clause 6.2.6 [1])

5.2 Service capabilities

5.2.1 Service capability data model

The service capability model distinguishes data and action. A class of data, say a user profile, may allow a number of discrete actions. This is shown when service capabilities are defined using some object based design and implementation languages (e.g. UML and Smalltalk) where the specification language may offer some built in protection of the service capability by restriction of the visible interface. This is shown visually in figure 5 for a typical architectural arrangement for an object based on the Profile class as defined in TR 101 882 [9].

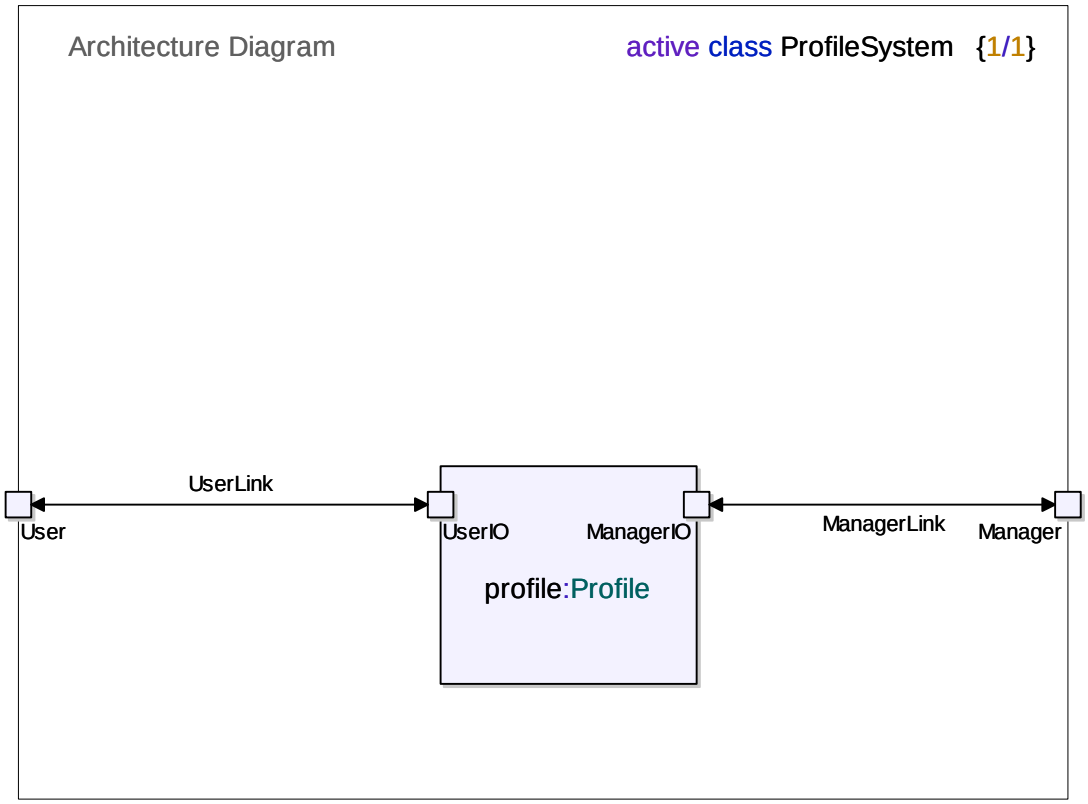


Figure 5: Typical profile service architecture

The ability to restrict some capabilities of the profile class is shown in figure 6 where the ability to add and delete services from the user profile, and the ability to switch authentication on and off, is restricted to the manager interface.

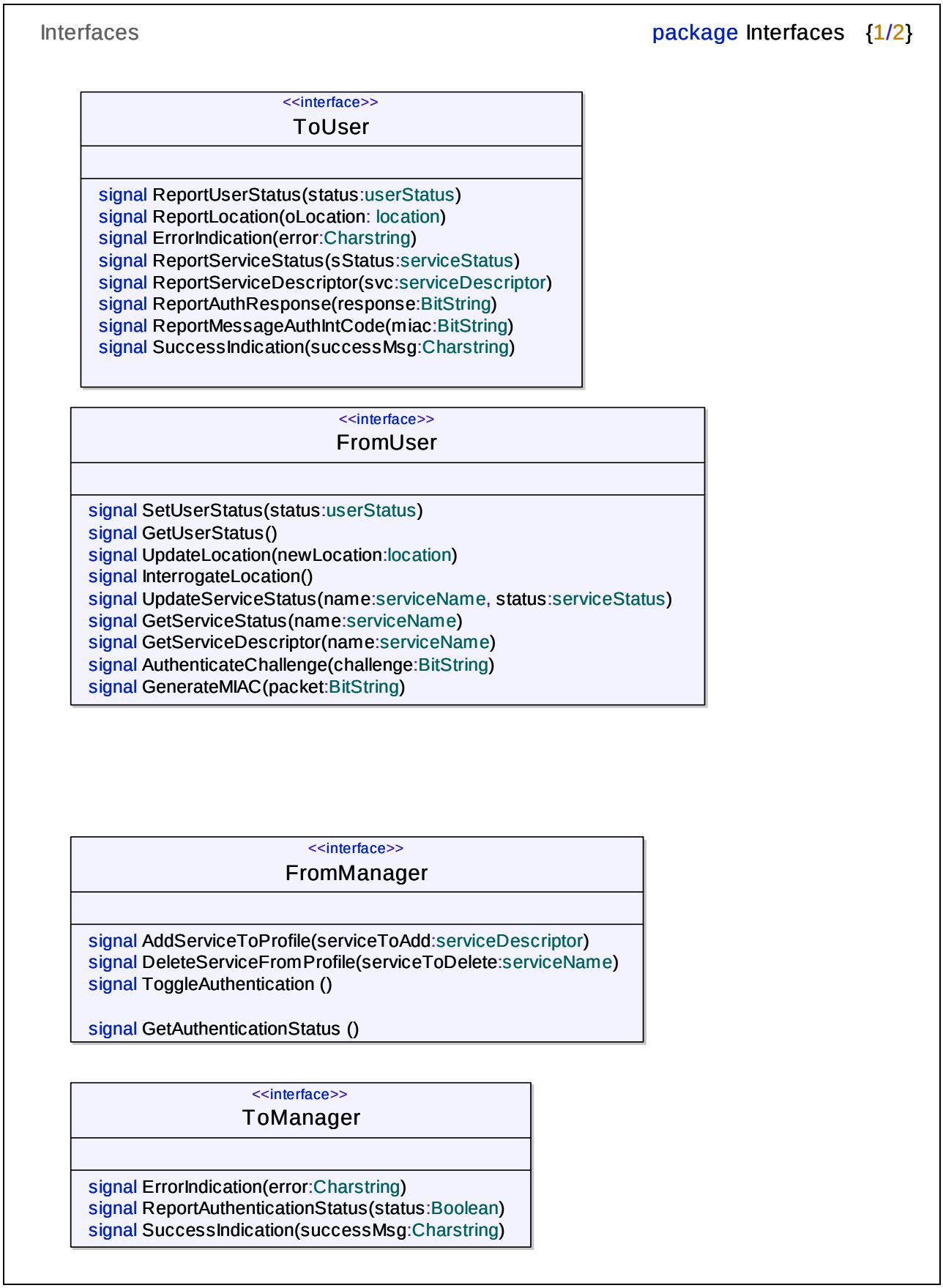


Figure 6: Class diagram showing the signals belonging to each interface

Similar groupings of functionality for the service capabilities defined in TR 181 004 [1] and restriction of access to the capabilities by interfaces and ports (a firewall form of protection) is assumed as a first level of protection to the service capabilities in each class.

5.2.2 Service capability model

The service capabilities shown in table 2 are taken from TR 101 882 [9] and reflect the approach recommended in clause 5.2.1 whereby service capabilities are discrete objects.

Table 2: Service capabilities as defined in TR 101 882 [9]

Service capability	Short description
Authenticate	The *authenticate* service capability is described by a set of capabilities allowing Challenge-Response authentication and Message Authentication Integrity Code authentication forms. The service capability supports symmetric and asymmetric keying methods, single and multi-pass protocols, and both unilateral and mutual authentication.
Get user status	The *get user status* service capability allows an authorized user to query the current status of a user (the requesting user or another).
Set user status	The *set user status* service capability allows an authorized user to set the current status of a user.
Interrogate location	The *interrogate location* service capability allows an authorized user to query the location of a user.
Update location	The *update location* service capability allows an authorized user to set the location of a user.
Update service status	The *update service status* service capability modifies the service status where the service status may take values including available and unavailable.
Add service to profile	The *add service to profile* service capability adds a service to the profile of the user.
Remove service from profile	The *remove service from profile* service capability removes a service from the profile of the user.
Get service status	The *get service status* service capability allows a user to query the current status of a service.
Get service descriptor	The *get service status* service capability allows a user to retrieve service descriptor information.
Call setup	
Call cleardown	
Calling party identity information delivery	
Call redirect	
Set call priority	
Call join	
Call interrogate	
Bearer Create	The Bearer Create capability is a composition of two sub-capabilities: Reserve Bearer and Allocate Bearer. The Reserve Bearer service capability assigns the necessary bearer resources, if available, to a call but does not complete the connection. The Allocate Bearer service capability completes the connection of previously reserved bearer resources.
Modify bearer	The *Modify Bearer* service capability assigns the bearer resources, if available, required to alter the bearer capabilities of an established call (see note).
Delete bearer	The *Delete Bearer* service capability releases all previously reserved and allocated bearer resources associated with a particular call.
Set media encode	The *set media encode* service capability establishes the media encoding and decoding requirements for a particular media type. These requirements are characterized by the information elements in the supplied media attributes.
Clear media encode	The *clear media encode* service capability releases any media encoding and decoding resources allocated by the *set media encode* service capability.
Create message	The *create message* service capability creates a new message on request from a suitably authorized user or application.
Retrieve message	The *message retrieve* service capability delivers the contents of an existing message to a suitable authorized user or application (normally the message recipient).
Set message status	The *set message status* service capability modifies the current status of an existing message. The only valid values of message status shall be "Read" and "Unread".
Get message status	The *get message status* service capability returns the current status of an existing message to a suitably authorized user or application.
Delete message	The *delete message* service capability removes an existing message on request from a suitably authorized user or application.

Service capability	Short description
Set condition	The *set condition* service capability sets a trigger based upon a condition related to the monitored group. The supplied event descriptor specifies the service capability to be invoked and the parameters to use when the condition is met.
Clear condition	The *clear condition* service capability clears a previously set condition, identified by the supplied event identity.
NOTE: This service capability does not complete the connection of the modified bearer resources. This can be achieved by invocation of the *Allocate Bearer* service capability.	

5.3 Formal statement of security requirements

5.3.1 Identification and authentication

Service capabilities are offered to an external service logic and should be protected from invocation from unauthorized entities. In many cases formal identification and authentication of the invoking party will not be possible, in such cases the invoker should be considered as anonymous and the capabilities offered to anonymous users should be extremely restricted, i.e. should not be able to create or delete permanent (long-life) data. Management actions should be fully accountable and therefore the invoking entity should be identified and should also be authenticated (to counter masquerade).

Each invocation of a service capability should follow the following simple guidelines introduced in clause 4:

- The <<*service capability invoking user*>> is not allowed to <<*invoke the service capability*>> prior to successful identification (FIA_UID.2).

NOTE 1: Identification may be achieved by a number of schemes.

- The <<*service capability invoking user*>> is not allowed to <<*invoke the service capability*>> prior to successful authentication (FIA_UAU.2).

NOTE 2: Authentication may be achieved by a number of schemes.

5.3.2 Integrity of data

Where data is transferred between objects (instances of service capabilities) the integrity of the data should be assured and in most cases the requested service capability should not be invoked if there is any doubt in the integrity of the received data. This is particularly important for detection of attacks caused by a man-in-the-middle modifying capability invocations.

- When <<*service capability*>> transmits <<*data in a signal*>> to a user the system shall provide that user the means to detect modification anomalies (FCO_IED.1).

- When <<*service capability*>> transmits <<*data in a signal*>> to a user the system shall provide that user the means to detect deletion anomalies (FCO_IED.1).

- When <<*service capability*>> transmits <<*data in a signal*>> to a user the system shall provide that user the means to detect insertion anomalies (FCO_IED.1).

- When <<*service capability*>> transmits <<*data in a signal*>> to a user the system shall provide that user the means to detect replay anomalies (FCO_IED.1).

NOTE: Detection of integrity errors may be achieved by a number of schemes.

6 Consideration of Common Criteria Composition class

6.1 Composition assurance classes

In traditional Common Criteria Evaluation as described in ISO/IEC 15408 [19] and addressed for the ETSI standardization process in EG 202 387 [3] there is a single concept of Evaluation Assurance Levels (EALs) where a higher level indicates to a reasonable approximation that the design of the object, and the documenting and testing of the design, has been undertaken with greater rigour. In such cases the evaluation and hence the EAL awarded is done for each object in isolation. Combining the results of many evaluations is not trivial and the conventional EALs have been considered inappropriate. In recognizing this the Common Criteria study group has defined instead a set of 3 composition assurance classes:

- Composition assurance level A (CAP-A) - Structurally composed
- Composition assurance level B (CAP-B) - Methodically composed
- Composition assurance level C (CAP-C) - Methodically composed, tested and reviewed

6.1.1 CAP-A: Structurally composed

CAP-A is applicable when the user requires a low to moderate level of independently assured security in the absence of the complete development record. Under a CAP-A evaluation the security requirements and provisions are analysed based upon the evaluations of each service capability component in isolation and by examination of the behaviour expected of each service capability across its interface. Core to the examination is the content of the reliance class (see later in this clause for a detail breakdown of each member of the assurance family as it applies to the standards development process).

In addition to the examinations to be performed in the ACO class the following examinations also apply for the composed system (the reader is referred to Common Criteria [22] for a complete description of these assurance classes as these are not described further in the present document):

- Assurance class Life Cycle Support:
  - Configuration management capabilities, Labelling of the TOE, ALC_CMC.1.
  - Configuration management scope, Parts of the TOE CM coverage, ALC_CMS.2.
- Assurance class Security Target Evaluation:
  - Conformance claims, ASE_CCL.1.
  - Extended components definition, ASE_ECD.1.
  - ST introduction, ASE_INT.1.
  - Security objectives, ASE_OBJ.1.
  - Security requirements, ASE_REQ.1.
  - TOE Summary specification, ASE_TSS.1.

NOTE: The assurance classes ALC and ASE have no relevance in the standardization process as they refer to Security Targets (products) and not to the Protection Profiles which standards more closely reflect.

6.1.2 CAP-B: Methodically composed

CAP-B extends CAP-A and is applicable when the user requires a moderate level of independently assured security by thorough investigation of the composed system and its development record without requiring substantial re-engineering to achieve the desired security assurances.

6.1.3 CAP-C: Methodically composed, tested and reviewed

CAP-C extends CAP-B and is applicable when the user requires a high level of independently assured security by thorough investigation of the composed system and its development record and where additional re-engineering costs are acceptable to achieve the desired security assurances.

6.2 Class description

Work in the Common Criteria for Security Assurance Composition Class is closely aligned to identifying the answer to the problem "if components A and B are evaluated as having security ratings X and Y what is the security rating that can be assigned to the combination of A and B?" (see http://www.commoncriteriaportal.org [21]). The goal of the Composition assurance activity in Common Criteria [22] is to determine whether components can be integrated in a secure manner by examination and testing of the interfaces between the components, supported by examination of the design of the components and the conduct of vulnerability analysis (in ETSI terms the use of the TVRA process described in TS 102 165-1 [4] should be adopted).

The approach of using discrete interfaces and closing down the capabilities being offered as described in clause 5 eases the task of performing a TVRA and the visualization approach recommended in TS 102 165-1 [4] allows examination of the scenario that is represented by the composition. In particular the dependency relationship between service capabilities should be highlighted (e.g. call-setup requires bearer-setup to be completed before it itself can be completed) and the cardinality of each relationship should be highlighted (e.g. a call requires exactly one bearer to be established).

The Common Criteria Assurance Composition Class (ACO) is defined in Common Criteria [22] such that the developer of a system that is composed of two or more components which have evaluated using the CC, can determine if they can be integrated in a secure manner. This is achieved by the following steps in the integration:

a) determine that the required assurance is provided by the base component where the base component is the service capability;

b) determine that the base component and dependent component are compatible; and

c) search for any vulnerabilities introduced through composing the base and dependent components into a single composed entity.

6.3 Implications for the standardization process

Composition is a natural occurrence arising from standardization and the evaluation of security in systems composed from many models is a core requirement in system design where modular design is employed and where designs are reused. Such models are at the root of the service capability model of development.

The design of service capabilities (including those providing security capability) has to follow a number of rules such that the interfaces to each service capability can be examined and the operation of the service capability can be tested.

The use of UML, in particular UML2, addresses both the rigour requirements of the design process and can be applied to the entire lifecycle. The use of deployment diagrams for example allows the designer to show how service capabilities are deployed in real life systems thus setting the stage for the composition class evaluation.

6.4 Families and components

6.4.1 Composition class evaluation levels

Table 3: "Composition" family evaluation levels

Evaluation component		CAP-A	CAP-B	CAP-C
Composition rationale				
ACO_COR.1	Composition rationale	✓	✓	✓
Development evidence				
ACO_DEV.1	Functional description	✓		
ACO_DEV.2	Basic evidence of design		✓	
ACO_DEV.3	Detailed evidence of design			✓
Reliance of dependent component				
ACO_REL.1	Basic reliance information	✓		
ACO_REL.2	Reliance information		✓	
ACO_REL.3	Detailed reliance information			✓
Base TOE testing				
ACO_TBT.1	Interface testing	✓	✓	✓
Composition vulnerability analysis				
ACO_VUL.1	Composition vulnerability review	✓		
ACO_VUL.2	Composition vulnerability analysis		✓	
ACO_VUL.3	Extended-basic Composition vulnerability analysis			✓

6.4.2 Composition rationale family (ACO_COR)

Applicable to: CAP-A to CAP-C.

The Composition rationale (ACO_COR) family is used to determine whether or not the appropriate assurance measures have been applied to the base for successful integration in the composed TOE. That is, the Security Assurance Requirements claimed for the component service capability are consistent with those being claimed in the composite system (e.g. if the assurance package for the composite system included ACO_DEV.3 Detailed evidence of design, a base component that was evaluated against Assurance Class Development Functional Specification level 2 (ADV_FSP.2) would not have had the appropriate assurance measures applied, as insufficient design evidence would have been examined.)

6.4.3 Development evidence family (ACO_DEV)

6.4.3.1 Functional description (ACO_DEV.1)

Applicable to: CAP-A.

In the functional description it is necessary to show the interfaces of the service capabilities used in the composed system and specifically identify which of the interfaces are actively involved during as a result of the composition. In the design terms recommended wherein a service capability belongs to a data object with access to the data object (or parts of it) restricted to particular signals this level of evidence should identify the decomposition and rationale for the decomposition made by the designer.

6.4.3.2 Basic evidence of design (ACO_DEV.2)

Applicable to: CAP-B.

The developer is required to give a description of the interfaces in each service capability that is used by another service capability. For example a bearer-setup capability may be used by the call-setup capability and therefore the developer has to fully describe each of these capabilities. The dependency between capabilities has to be shown in order for an evaluator to be able to determine if the requirements being placed on the composition of capabilities can be satisfied from the design. In more detail the development information has to achieve the following:

- describe the purpose and method of use of each interface of the service capability used in the composed system;
- describe all parameters associated with each interface for the service capability;
- describe all operations associated with each interface for the service capability;
- describe the error messages resulting from processing associated with all operations;
- identify those interfaces of the service capability that are provided to support functions of any dependent service capability in the composed system.

6.4.3.3 Detailed evidence of design (ACO_DEV.3)

Applicable to: CAP-C.

ACO_DEV.3 extends ACO_DEV.2 by also requiring that the development information has to achieve the following:

- describe the structure of the base service capability in terms of components;
- describe the architecture of those components that provide the interfaces of the base service capability that are relied upon to support the TSF of the dependent component.

6.4.4 Reliance of dependent component family (ACO_REL)

The purpose of the ACO_REL family is to show that there is a genuine dependency between the components being evaluated. Whilst this may appear trivial and of little consequence as in figure 7 the practical application is much greater.

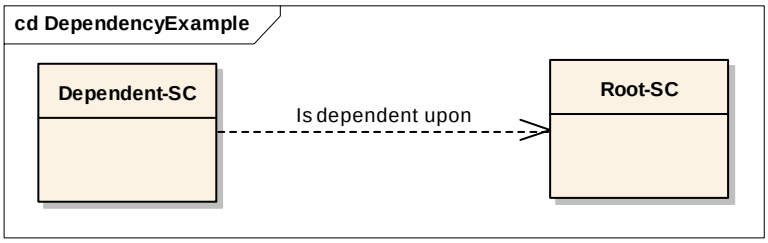


Figure 7: Simple/trivial dependency relationship

6.4.4.1 Basic reliance information (ACO_REL.1)

Applicable to: CAP-A.

The developer has to provide functional reliance information addressed to system integrators (i.e. those building the system from the service capabilities) and has to include the following elements:

- the functionality of the base component hardware, firmware and/or software that is relied upon by the dependent component (i.e. a description of the service capability and in particular its deployment environment (if known), invocation signals and the data associated with it);

- identification of all interfaces through which the dependent component requests services from the base component (i.e. the signals and ports offering services from the service capability's class to any calling service capability);
- the purpose and method of use of each interface.

6.4.4.2 Reliance information (ACO_REL.2)

Applicable to: CAP-B.

ACO_REL.2 extends ACO_REL.1 by also requiring that the reliance information has to achieve the following:

- describe the expected operations and results associated with each security function-enforcing interface;
- describe the error handling performed as a result of the dependent component's use of each SFR-enforcing interface.

6.4.4.3 Detailed reliance information (ACO_REL.3)

Applicable to: CAP-C.

ACO_REL.3 extends ACO_REL.2 by requiring that the reliance information is extended to cover all the interfaces and not only those that enforce the security functional requirements.

6.4.5 Base TOE testing

6.4.5.1 Interface testing (ACO_TBT.1)

Applicable to: CAP-A to CAP-C.

In the ACO_TBT.1 class it is necessary for the developer to show that the service capability (base and dependent components) have been developed and tested. Furthermore the test results from the developer execution of the tests have to demonstrate that the base component interface relied upon by the dependent component behaves as specified. This has to be shown by developing a full suite of test documentation consisting of test plans (In ETSI terms these are the Test Suite Structure and Test Purposes (TSS&TP)), test procedure descriptions (in ETSI terms these are test cases), expected test results and actual test results.

The analysis of this class is as for the ATE class described in clause 6.7 of EG 202 387 [3] and the recommendations made in that document also apply:

- Evidence of coverage should be carried out and documented according to the "walk-through" method described in EG 202 107 [11].
- Analysis of coverage should make use of simulation techniques to validate formal models as described in EG 201 015 [12], EG 202 107 [11] and ETR 184 [13].

6.4.6 Composition vulnerability analysis

6.4.6.1 Composition vulnerability review (ACO_VUL.1)

Applicable to: CAP-A.

The ACO_VUL class is intended to show to an evaluator that the residual vulnerabilities in a composed system are no exploitable when the system is deployed. The class itself builds on the AVA_VLA class described in clause 6.8.3.4 of EG 202 387 [3] and for which a method of analysis is described in TS 102 165-1 [4]. The method of eTVRA described in TS 102 165-1 [4] identifies the suite of assumptions and objectives for the service capability and shows how to determine the level of risk in the system. The eTVRA method also assists in the activity of analysis for combinations of assets (the system capabilities in the NGN).

6.4.6.2 Composition vulnerability analysis (ACO_VUL.2)

Applicable to: CAP-B.

The requirements placed on the developer for ACO_VUL.2 are the same as for ACO_VUL.1 but the evaluator will perform an independent vulnerability analysis to cross check that of the developer.

6.4.6.3 Extended basic composition vulnerability analysis (ACO_VUL.3)

Applicable to: CAP-C.

The requirements placed on the developer for ACO_VUL.3 are the same as for ACO_VUL.1 but the evaluator conducts further testing to determine the susceptibility of the service capability and the composed system to attacks where the attacker is more motivated (extended basic capability).

Annex A (informative): Use of Cryptographic techniques

A.1 Introduction

There are several steps in cryptography, confidentiality and integrity decision trees in clause 5 that have references to this annex. Some steps in the authentication decision tree in clause 5 need this annex too when doing for example manual key management. When considering an encryption technique, restrictions in the use of cryptographic techniques in clause A.5 should be taken into account. Manual, automatic, symmetric and asymmetric key management (e.g. using certificates and CA) must also be considered when selecting cryptographic mechanisms.

A.2 Key management overview

Where cryptographic methods are used to support security the primary element of achieving security is in the key. The general assumptions for any system relying on cryptology are:

- Knowledge of how algorithms work is in the public domain.
- Knowledge of protocols for authentication and key establishment are in the public domain.

The only means of assuring security remains in place, over and above the known limitations of the algorithm and protocol, is in the secrecy of the key. A secret is by definition not a secret when it is widely known and so a shared secret is not really secret. Symmetric key cryptography works only by control of the number of entities who know the secret and generally, for telecommunications, the intention is to limit this to two parties only. However in public communication where secrecy may be required of communication to a large number of unknown parties the normal definition of secrecy cannot apply. The challenge of this is met by a set of techniques based on non-secret cryptology, or asymmetric keying, whereby a key has two components one of which is private and the other is public. The success is built on the mathematics of the key construction and on the algorithms that make use of the key, but essentially it consists of a pair of one-way functions and the view that is computationally infeasible from knowledge of the public key to find the matching private key. A public key can then be distributed either freely or traceably to either receive data encrypted by the private key, or to encrypt data to be sent to the holder of the private key.

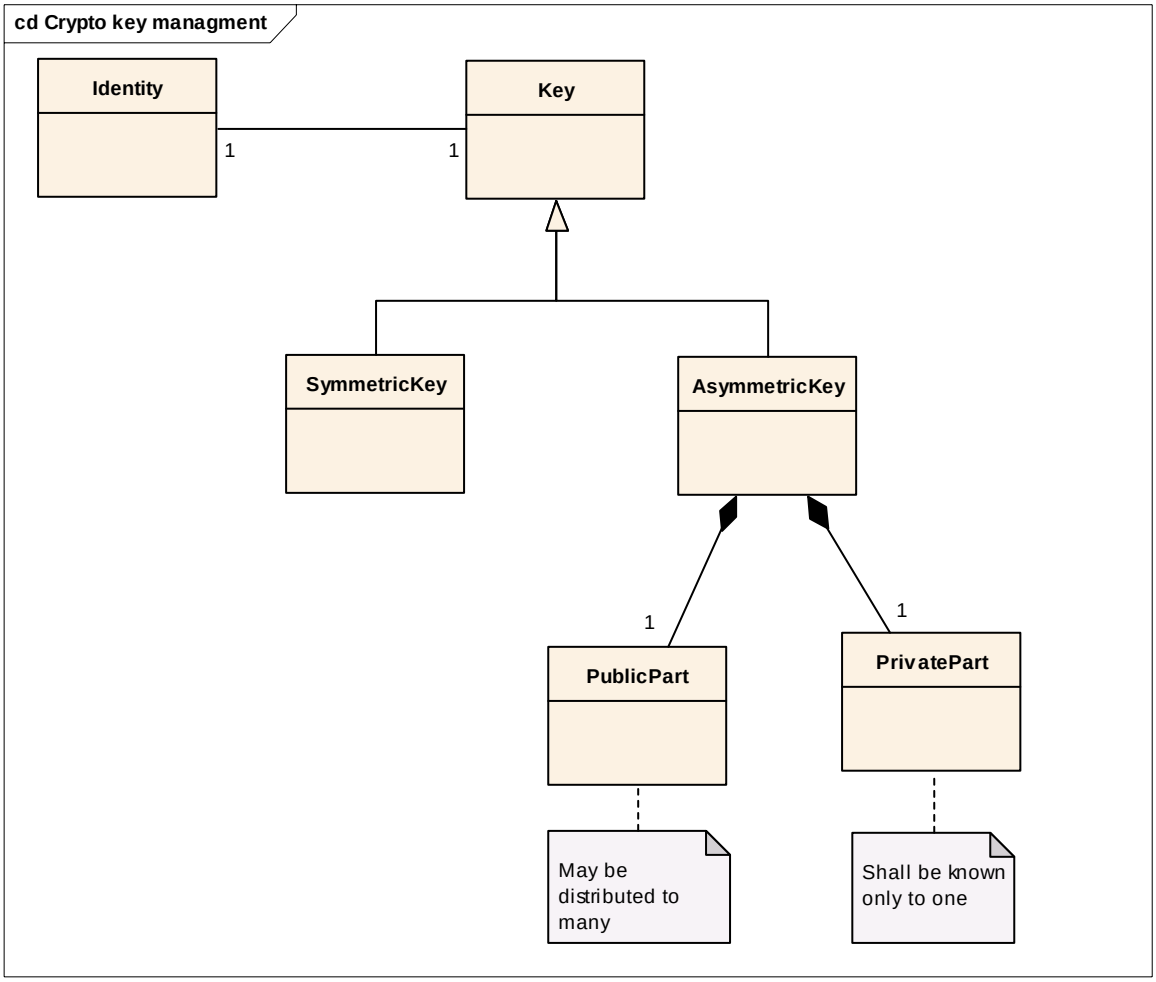


Figure A.1: Simplified model of key relationships

A.3 Symmetric key management

A.3.1 Overview

In symmetric key cryptography there is one mandatory requirement:

- Only 2 parties have access to the key.

In order to maintain compliance with this requirement there are a number of approaches to key distribution that may be taken. In each case the key should be delivered in a tamper proof format and in a manner that leaves an audit trail. Tamper proofing may be achieved in either software or hardware.

A.3.2 Key expiry

As only two parties have access to the key, the deletion by any one party of the key implies that the key is no longer valid. Operational policy may dictate the lifetime of a key and the provisions if either party loses the key (a lost key should in most instances be treated as non-recoverable but if one party is assigned as key owner with the second party as key user it may be feasible to re-deliver the key to the second party).

A.4 Asymmetric key management

A.4.1 Overview

In asymmetric cryptography a public key can be distributed to either receive data encrypted by the private key, or to encrypt data to be sent to the holder of the private key. However there is a legitimate concern that whilst the mathematical relationship is understood to work there is often only a weak relationship between the two communicating parties hence trust that the data is visible to the correct party has to be assured. The counter to the trust problem is to distribute public keys through a trusted source within public key certificates according to clause 7 of ITU-T Recommendation X.509 [14].

NOTE: The present document does not replace ITU-T Recommendation X.509 [14] but is intended to assist users of public key cryptosystems in its use.

A.4.2 Certificate generation

A certificate is a signed data object that contains the data elements outlined in table A.1.

The content of a digital certificate can be summarized as follows:

- The identifier of the certification authority.
- The unique identifier of the user of the certificate.
- Some attributes of the user, like address, company, tax code, etc.
- Public key, generated with the private key, to be used to verify digital signature.
- Period of validity of the certificate, defined by a start date and a end date.
- Unique identity code of the certificate.
- Digital signature of the certification authority.
- Environment in which the certificate is valid.
- Non-mandatory attributes.

Table A.1: Contents of X.509 certificate

Information element	M/O	Notes
Version	M	Default value of v1
Serial number	M	
Signature	M	
Issuer	M	
Validity	M	
Subject	M	
Subject public key information	M	
Issuer unique identifier	O	
Subject unique identifier	O	
Extension	O	

A.4.3 Certificate revocation

A certificate may expire naturally, i.e. when the value of the validity information element is no longer valid, and may also be revoked, i.e. to force a certificate where the value of the validity information element although still valid is to be treated as if it were invalid.

A.4.4 Certificate extension

Certificate extensions can be used to provide service and service capabilities authorization as explained above.

As defined in ITU-T Recommendation X.509 [14] extensions provide methods for associating additional attributes with users or public keys and for managing the hierarchy. It also allows communities to define private extensions. Extensions can be defined in a certificate as critical and non-critical. A system that uses a certificate must reject the certificate if it encounters a critical extension it does not recognize. Each extension includes an ASN.1 Object Identifier (OID) and an ASN.1 structure. Only one instance of a particular extension may appear in a particular certificate.

A.4.5 Certification authority

This is a trusted third party that issues certificates. In PKIs (Public Key Infrastructure) the CA verifies identity. CAs (certification authorities) can issue different kinds of certificates:

- Identity.
- Authorization.
- Transaction.
- Time Stamp.

Repudiation services may use a CA as a Trusted Third Party.

A.5 Manual and automatic key management

A.5.1 Manual key management

Manual key management maybe achieved by means of a token card and a token server. Each token card, about the size of a credit card, is programmed to a specific user, and each user has a unique PIN that can generate a password keyed strictly to the corresponding card. The password is then entered into the password field during a remote authentication. The server sends a challenge to the user. The user uses the token card to get a response of that challenge that sends to the server which compares it with the result stored in it. If both are the same the authentication is successful.

Smartcards maybe used for authentication of identity. The most common example is in conjunction with a Public Key Infrastructure (PKI). The smart card will store an encrypted digital certificate issued from the PKI along with any other relevant or needed information about the card holder. Smart cards are a privacy-enhancing technology, and when used in conjunction with appropriate security and privacy policies, can be part of a highly effective authentication system.

There are several protocols to automatically manage the keys that are used to provide the key that the cryptographic algorithms use.

A.5.2 Automatic key management

The keys and cryptographic algorithms and control parameters used in IPsec can be set manually by means of a SA negotiation protocol. IKE (RFC 4306 [23]) is used to set those keys and establish the SA. IKE is not limited to be used with IPsec, it can also manage the keys for other protocols. There are two phases in the negotiation to establish a secure channel to establish an SA between two entities. IKE allows to change the keys in a process that is called rekeying.

The scheme for authentication and key agreement in the IMS is called IMS AKA as defined in TS 133 203 [7].

A.5.3 Key exchange algorithms and protocols

A.5.3.1 Ellis and non-secret cryptography

James Ellis and colleagues at the United Kingdom's GCHQ proposed the possibility of "non-secret encryption" in 1973 as a response to the problem of key distribution for symmetric cryptography. This predates the more widely known approach taken by Diffie and Hellman that was proposed in 1976 but set the groundwork for public-key, or asymmetric key, cryptography. In particular Malcolm Williamson and Clifford Cocks developed algorithms that practically demonstrated the value of the system.

A.5.3.2 Diffie-Hellman algorithm

The Diffie-Hellman key agreement protocol (also called exponential key agreement) was developed by Diffie and Hellman in 1976 and published in the ground-breaking paper "New Directions in Cryptography" [20]. The protocol allows two users to exchange a secret key over an insecure medium without any prior secrets.

The protocol has two system parameters p and g. They are both public and may be used by all the users in a system. The protocol depends on the discrete logarithm problem for its security which assumes that it is computationally infeasible to calculate the shared secret key $k = g^{ab} \bmod p$ given the two public values $g^a \bmod p$ and $g^b \bmod p$ when the prime p is sufficiently large.

The Diffie-Hellman key exchange has known vulnerabilities, in particular it is vulnerable to a man-in-the-middle attack. In this attack, an opponent Carol intercepts Alice's public value and sends her own public value to Bob. When Bob transmits his public value, Carol substitutes it with her own and sends it to Alice. Carol and Alice thus agree on one shared key and Carol and Bob agree on another shared key. After this exchange, Carol simply decrypts any messages sent out by Alice or Bob, and then reads and possibly modifies them before re-encrypting with the appropriate key and transmitting them to the other party. This vulnerability is present because Diffie-Hellman key exchange does not authenticate the participants but can be mitigated by the use of authentication to identify the presence of a man-in-the-middle although if this is necessary some authentication credentials may need to be known which complicates the protocol of key exchange.

A.5.3.3 Internet Key Exchange

NOTE: Real environments are composed of applications and machines that are not based on IP. Real environments are more complex and have lots of interconnection and security implications further than the ones described in this example.

Confidentiality, authentication and integrity services can be provided by IPsec in the IP layer for applications that are based on IP and they are independent of the network layer technology (FR, xDSL, ATM, PPP). So in a bank environment where lots of offices in the company Intranet and remote users that need to connect from a visited public IP network with different access technologies have to be connected with a main data centre or other applications that are located in smaller data centres. There are also communication sessions with other different companies, all of them over IP protocol. In all of these cases, IPsec provides communications confidentiality and integrity and they are independent of the access technology.

The equipment responsible for establishing the tunnels may have the following methods to establish the safe channel of the first IKE phase. These are some examples of the IKE algorithms negotiated and the equipment should allow them to be configured.

- Pre-shared key authentication. In this case the key is manually configured in the communication equipment. This configuration can be done by typing the commands manually in the console of the communications equipment or by writing configuration files that can be downloaded to the equipment. Special care must be taken when doing this download as a more or equal safe channel than the intended one must be used.

- Authentication with digital signature. In this case the key exchange to establish the safe channel is done be means of the Diffie-Hellman algorithm and the authentication be means of digital signature. The same practical considerations than in the previous point must be applied.

- Authentication with asymmetric key. In this case asymmetric cryptography is used to do authentication and to establish the safe channel. Better authentication mechanism is provided in this case than in the previous one. The same practical considerations than in the first point must be applied when distributing and configuring the private key of the asymmetric algorithms. A full description of asymmetric key management in annex A.

In table A.2 some sets of security parameters to define IKE security policy to establish the safe channels are detailed.

Table A.2: Example IKE security policy

Parameter description	Values	Predetermined value
Confidentiality algorithm	DES 3DES	DES
Integrity algorithm	MD5 SHA-1	SHA-1
Authentication algorithm	Preshared keys RSA	RSA
Key exchange algorithm	Diffie-Hellman 768 Diffie-Hellman 1024	Diffie-Hellman 768
SA timeout	Any number of seconds	80 000

Once the safe channel is established IPsec can negotiate the security parameters to be used later.

During negotiation of the SA several sets of protocols, algorithms and security parameters can be applied to IPsec tunnels to protect the traffic between the nodes. Each proposal includes one or more protocols Each protocol contains one or more transforms (each specifying a cryptographic algorithm). Each transform contains zero or more attributes (attributes are needed only if the transform identifier does not completely specify the cryptographic algorithm).

In the table below there are some examples of the building components of these sets.

Table A.3: Example IPsec transforms

Transform set
AH-HMAC-MD5
AH-HMAC-SHA
ESP-DES
ESP-3DES
ESP-HMAC-MD5
ESP-HMAC-SHA

In those two phases TS 133 210 [6] recommends some mandatory attributes for each of the security parameters as shown in tables A.4 and A.5.

Table A.4: Recommended parameters for IKE Phase-1

Parameter description	Value
Confidentiality algorithm	3DES
Integrity algorithm	SHA-1
Key exchange algorithm	Diffie-Hellman group 2
SA timeout	Any number of seconds
Authentication algorithm	Preshared keys

Some more values of parameters are recommended as mandatory (e.g. AES and Main Mode) in TS 133 210 [6] for IKE phase 1. Only those that are of relevance for this example are shown.

Once the SAs are established (IKE phase 2) the following parameters are mandatory for that SA (as recommended in TS 133 210 [6] clause 5.4.

Table A.5: Recommended parameters for IKE SAs

Parameter description	Values
	IP Addresses or subnet identity
Notifications	SHA-1
Key exchange algorithm	Diffie-Hellman group 2
Key Length in AES-CBC transform	128 bits

A.6 Restrictions on use of cryptographic techniques

The reader is referred to EG 202 238 [10] for advice on the evaluation of cryptographic algorithms.

Annex B (informative): Bibliography

- "The possibility of Non-Secret digital encryption" J H Ellis, CESG Research Report, January 1970.
- "Non-Secret Encryption Using a Finite Field", M J Williamson, CESG Report, 21 January 1974.
- "Thoughts on Cheaper Non-Secret Encryption", M J Williamson, CESG Report, 10 August 1976.
- "A Note on Non-Secret Encryption", C C Cocks, CESG Report, 20 November 1973.
- Bruce Schneier: "Secrets & Lies. Digital Security in a networked world" (ISBN 0-471-25311-1).
- http://www.rsasecurity.com/rsalabs/node.asp?id=2248
- ETSI TR 180 001: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); NGN Release 1; Release definition".
- ETSI TS 181 001: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Videotelephony over NGN".
- ETSI TS 181 002: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); NGN Basic Supplementary services; General aspects".
- ETSI TS 181 006: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Services and Capabilities Requirements for TISPAN NGN Release 1".
- 3GPP TR 33.900 (V1.2.0): "3rd Generation Partnership Project; Technical Specification Group SA WG3; A Guide to 3rd Generation Security (3GPP TR 33.900 version 1.2.0)".
- 3GPP TS 33.120 (V4.0.0): "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G security; Security principles and objectives (Release 4)".
- 3GPP TS 33.102 (V7.0.0): "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Security architecture (Release 7)".
- 3GPP TR 33.908 (V4.0.0): "General Report on the Design, Specification and Evaluation of 3GPP Standard Confidentiality and Integrity Algorithms (Release 4)".

History

Document history		
V1.1.1	October 2006	Membership Approval Procedure MV 20061201: 2006-10-03 to 2006-12-01
V1.1.1	December 2006	Publication