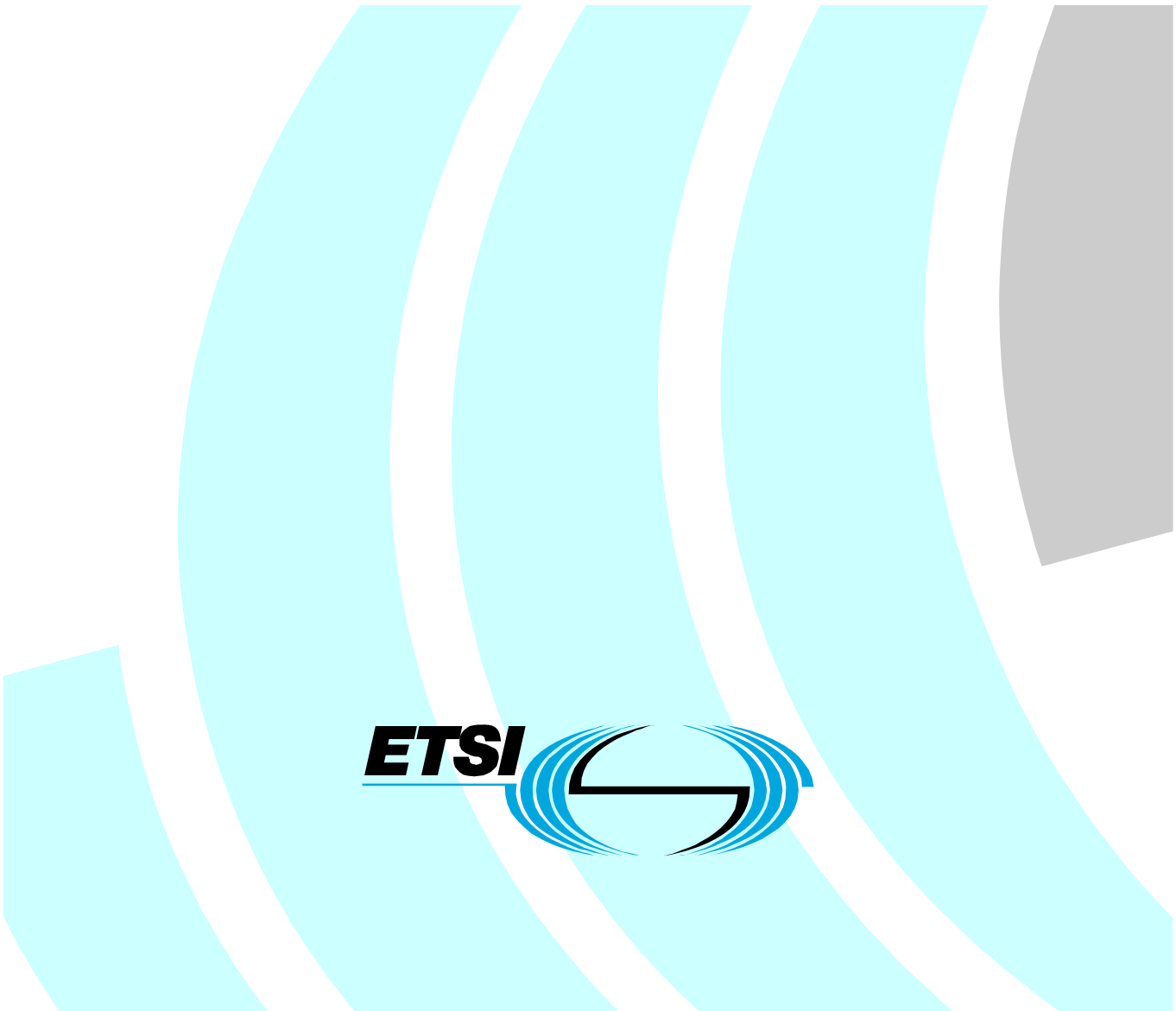
ETSI TR 102 780 V1.1.1 (2009-02)

Technical Report

Methods for Testing and Specification (MTS);
Security;
Guide to the use of methods in development of ETSI security standards



Reference
DTR/MTS-00109 OCG_SEC_DOC

Keywords
methodology, security

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 Technical Report (TR) has been produced by ETSI Technical Committee Methods for Testing and Specification (MTS).

1 Scope

The present document is a guide to the use of ETSI security standardization methods. The document identifies existing process documents and illustrates their use in order to provide a unified method for the preparation of security documents (guides, standards, algorithms).

NOTE: The present document is a companion to the "Security" pages on ETSI's "Making Better Standards" website [i.15].

2 References

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.
- Non-specific reference may be made only to a complete document or a part thereof and only in the following cases:
  - if it is accepted that it will be possible to use all future changes of the referenced document for the purposes of the referring document;
  - for informative references.

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.

2.1 Normative references

The following referenced documents are indispensable for the application of the present document. For dated references, only the edition cited applies. For non-specific references, the latest edition of the referenced document (including any amendments) applies.

Not applicable.

2.2 Informative references

The following referenced documents are not essential to the use of the ETSI deliverable but they assist the user with regard to a particular subject area. For non-specific references, the latest version of the referenced document (including any amendments) applies.

[i.1] ETSI EG 200 234: "Telecommunications security; A guide to specifying requirements for cryptographic algorithms".

[i.2] ETSI EG 202 238: "Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON); Evaluation criteria for cryptographic algorithms".

[i.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".

[i.4] ETSI ES 202 382: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Security Design Guide; Method and proforma for defining Protection Profiles".

[i.5] ETSI ES 202 383: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Security Design Guide; Method and proforma for defining Security Targets".

[i.6] ETSI EN 300 392-7: "Terrestrial Trunked Radio (TETRA); Voice plus Data (V+D); Part 7: Security".

[i.7] ETSI ETR 232: "Security Techniques Advisory Group (STAG); Glossary of security terminology".

[i.8] ETSI ETR 237: "Security Techniques Advisory Group (STAG); Baseline security standards; Features and mechanisms".

[i.9] ETSI ETR 332: "Security Techniques Advisory Group (STAG); Security requirements capture".

[i.10] ETSI TR 187 011: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); NGN Security; Application of ISO-15408-2 requirements to ETSI standards - guide, method and application with examples".

[i.11] ETSI TS 102 165-1: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Methods and protocols; Part 1: Method and proforma for Threat, Risk, Vulnerability Analysis".

[i.12] ETSI TS 102 165-2: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Methods and protocols; Part 2: Protocol Framework Definition; Security Counter Measures".

[i.13] ETSI TS 133 203: "Digital cellular telecommunications (Phase 2+); Universal Mobile Telecommunications System (UMTS); 3G Security; Access security for IP-based services".

[i.14] ETSI White Paper OCG/SEC.

[i.15] ETSI Portal - Making Better Standards.

NOTE: Available from http://portal.etsi.org/mbs/.

[i.16] ISO/IEC 15408-1: "Information technology - Security techniques - Evaluation criteria for IT security - Part 1: Introduction and general model".

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

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

[i.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.

[i.20] ISO/IEC 27001:2005: "Information technology - Security techniques - Information security management systems - Requirements".

[i.21] ISO/IEC 10181-2: "Information technology - Open Systems Interconnection - Security frameworks for open systems: Authentication framework".

[i.22] ISO/IEC 10181-3: "Information technology - Open Systems Interconnection - Security frameworks for open systems: Access control framework".

[i.23] ISO/IEC 10181-4: "Information technology - Open Systems Interconnection - Security frameworks for open systems: Non-repudiation framework".

[i.24] ISO/IEC 10181-5: "Information technology - Open Systems Interconnection - Security frameworks for open systems: Confidentiality framework".

[i.25] ISO/IEC 10181-6: "Information technology - Open Systems Interconnection - Security frameworks for open systems: Integrity framework".

[i.26] ISO/IEC 10181-7: "Information technology - Open Systems Interconnection - Security frameworks for open systems: Security audit and alarms framework" .

NOTE: Equivalent to ITU-T Recommendation X.815.

[i.27] ISO/IEC 13335-1: "Information technology - Security techniques - Management of information and communications technology security - Part 1: Concepts and models for information and communications technology security management".

[i.28] ISO 14977: "Extended Backus-Naur Form (EBNF) syntactic meta-language".

[i.29] Common Criteria Portal.

NOTE: Available from http://www.commoncriteriaportal.org/.

[i.30] ANSI Homeland Security Standards Panel Emergency Communications Report .

NOTE: Available from http://publicaa.ansi.org/sites/apdl/Documents/Standards%20Activities/Homeland%20Security%20Standards%20Panel/Workshop%20Reports/Emergency%20Communications.pdf.

[i.31] Federal Information Processing Standard (FIPS) 140-2: "Security Requirements For Cryptographic Modules".

[i.32] Federal Information Processing Standard (FIPS) 140-3: "Security Requirements For Cryptographic Modules".

NOTE: FIPS 140-3 will supersede FIPS 140-2 when it is completed.

[i.33] Federal Information Processing Standard (FIPS) 180-2: "Secure Hash Signature Standard (SHS)".

[i.34] Federal Information Processing Standard (FIPS) 197: "Advanced Encryption Standard (AES)".

[i.35] NIST guide to conformance: "Conformance Resources and Information".

NOTE: Available from "http://www.itl.nist.gov/div897/ctg/conformProject.html".

[i.36] NIST Special Publication 800-115: "Technical Guide to Information Security Testing and Assessment".

[i.37] NIST Computer Security Division; Security Testing and Metrics group.

NOTE: Available from "http://csrc.nist.gov/groups/STM/index.html".

[i.38] OASIS: "Security Assertion Markup Language (SAML)".

NOTE: Available from http://www.oasis-open.org/specs/#samlv2.0).

[i.39] IETF RFC1321: " The MD5 Message-Digest Algorithm".

[i.40] C.E. Shannon: "A Mathematical Theory of Communication", Bell System Technical Journal, vol. 27, pp. 379-423, 623-656, July, October, 1948.

[i.41] IETF RFC 4301: "Security Architecture for the Internet Protocol".

[i.42] IETF RFC 4303: "IP Encapsulating Security Payload (ESP)".

[i.43] IETF RFC 4306: "Internet Key Exchange (IKEv2) Protocol".

[i.44] IETF RFC 4305: "Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)".

[i.45] IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".

[i.46] IETF RFC 3566: "The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec".

[i.47] IETF RFC 3602: "The AES-CBC Cipher Algorithm and Its Use with IPsec".

[i.48] IETF RFC 3686: "Using Advanced Encryption Standard (AES) Counter Mode With IPsec Encapsulating Security Payload (ESP)".

[i.49] ITU-T Recommendation X.509: "Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks".

[i.50] ISECOM - The Institute for Security and Open Methodologies: "The Open Source Security Testing Methodology Manual".

[i.51] Information Systems Security Assessment Framework (ISSAF) Penetration Testing Framework.

NOTE: Available from http://www.oissg.org/issaf.

[i.52] Auguste Kerckhoffs: "La cryptographie militaire", Journal des sciences militaires, vol. IX, pp. 5-38, Jan. 1883, pp. 161-191, Feb. 1883.

[i.53] IEEE Std 802.11-2007: "IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications".

[i.54] IEEE Std 1394-1995: "IEEE Standard for High Performance Serial Bus Bridges".

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in ETR 232 [i.7], ETR 237 [i.8], ISO/IEC 13335-1 [i.27] apply.

3.2 Abbreviations

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

AMI	Authority Management Infrastructure
ANSI	The American National Standards Institute
API	Application Programming Interface
CRC	Cyclic Redundancy Check
DoS	Denial of Service
EBNF	Extended Backus-Naur Form
FIPS	The Federal Information Processing Standards
ICT	Information and Communication Technologies
IEEE	Institute of Electrical and Electronics Engineers
ISSAF	The Information Systems Security Assessment Framework
MAC	Message Authentication Code
MBS	Making Better Standards
NIST	National Institute of Standards and Technology
OSSTMM	Open Source Security Testing Methodology Manual
PMI	Privilege Management Infrastructure
SA	Security Association
SAML	Security Assertion Markup Language
TIA	Telecommunications Industry Association
TLS	Transaction Layer Security
ToE	Target of Evaluation
TVRA	Threat, Vulnerability and Risk Analysis
TVRA	Threat, Vulnerability and Risk Analysis
UML	Unified Modelling Language

4 Security design guidelines

4.1 Introduction

The main aim of communications security is the reduction or elimination of unwanted incidents. Practical security involves a sufficient understanding of risk in order to be able to contain it, manage it and, with care, to eliminate it.

The fundamental purpose of security standards is to specify countermeasures that can protect a system against certain forms of exploitation. Standardized countermeasures depend on an understanding of the attack (exploitation) context and the vulnerabilities that might exist within a system. In the standards environment there are then a number of concepts and a set of process documents that support the development of better security standards and it is these that are described in the present document.

ISO/IEC 15408 [i.19] (often referred to as "the Common Criteria") defines a means of achieving a level of security assurance. However, whilst the Common Criteria describes a key documentary approach with some underlying concepts, it is insufficient, by itself, to assist in the task of making better security standards.

Security technologies exist to manage risk which is measured by the likelihood of unwanted incidents arising. Potential unwanted incidents are determined by the environment within which the system exists and the countermeasures required to manage the risk at an acceptable level will, therefore, be driven by the environment or context.

TS 102 165-2 [i.12] describes a number of simple relationships in communications systems which together help to identify the conditions that can lead to unwanted incidents arising. These relationships are shown graphically as a UML class diagram in figure 1. The challenge to the standards developer is to identify vulnerabilities in the various system components.

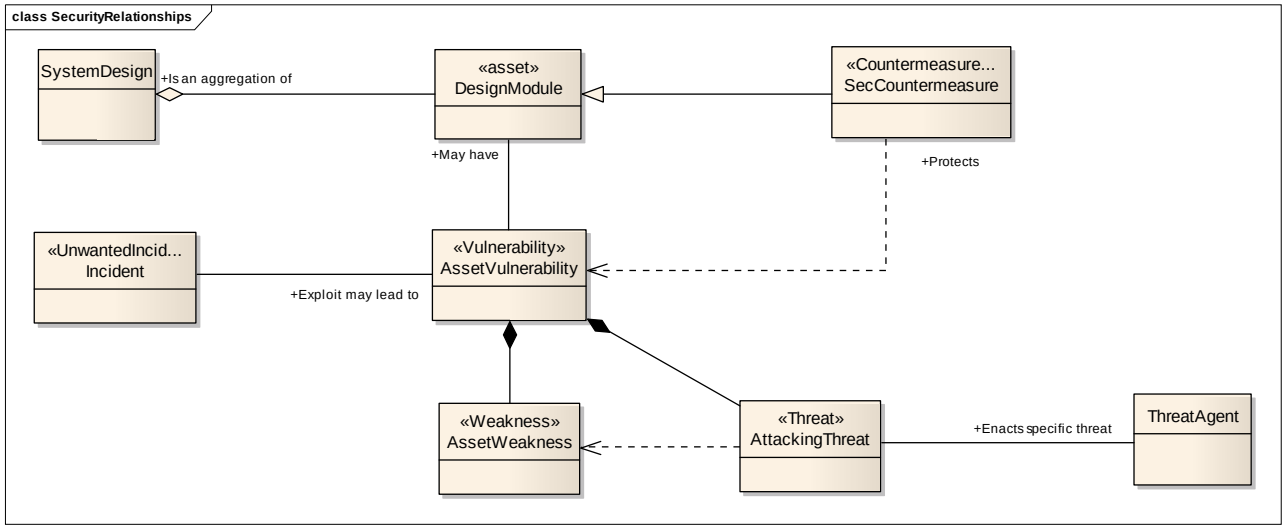


Figure 1: Security relationship model from TS 102 165-2 [i.12]

Standards for security address a wide range of aspects of the security domain as shown in Figure 2

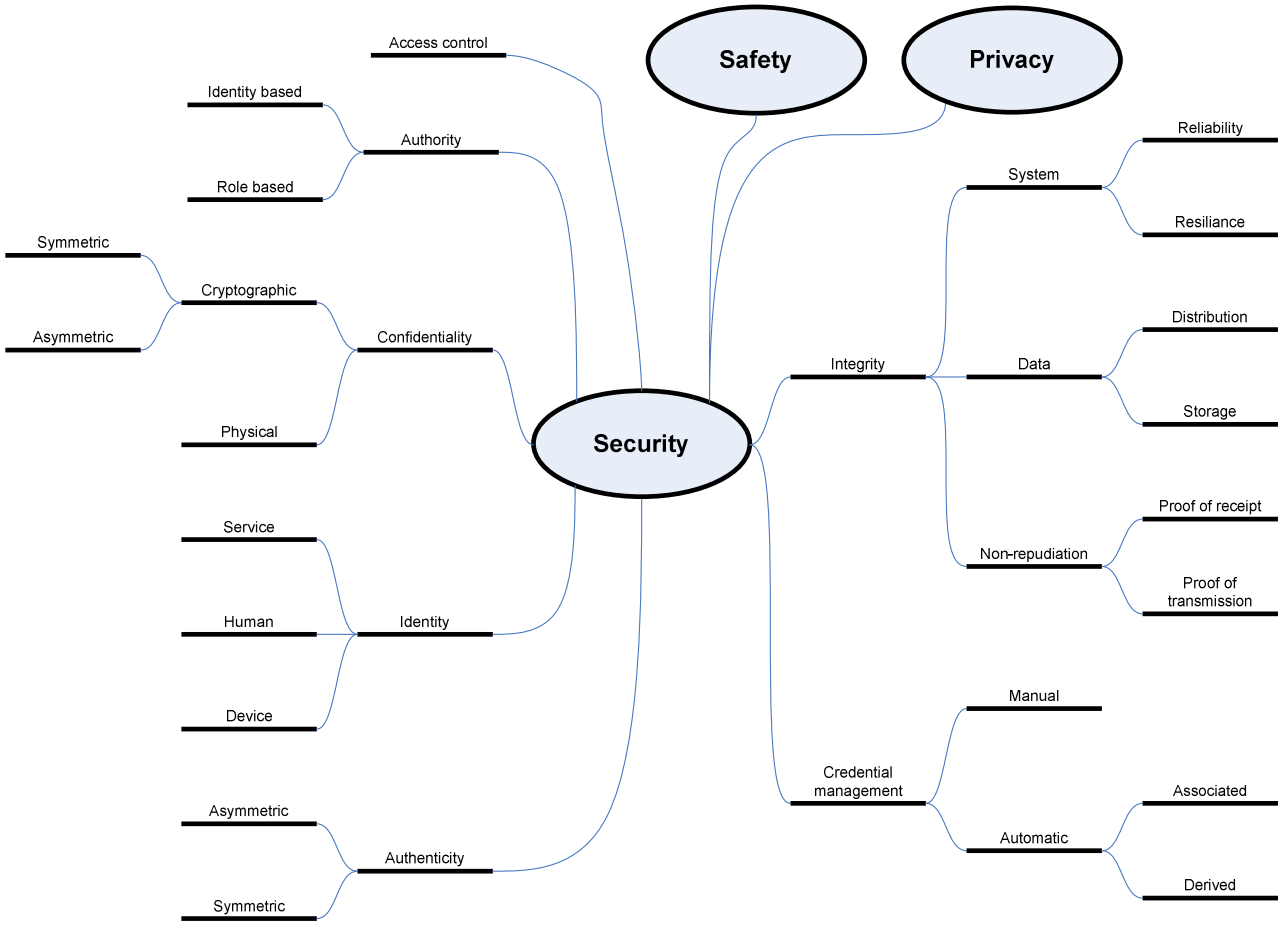


Figure 2: Security relationships and functions

4.2 Standards and requirements

Standards specify requirements that enable products to be developed with a high assurance of interoperability and interworking. As communications technologies and networks have become more open, the need for security functions has increased. It is no longer the case that security can be specified separately from base protocols and each specification will have to take security seriously. A general approach to the development of such standards is described in TR 187 011 [i.10]. The process that TR 187 011 [i.10] establishes is as follows:

1) define objectives for the target of standardization;

2) derive functional (high-level) requirements from the objectives;

3) derive detailed (implementation) requirements from the functional requirements.

It is important that the testability of requirements is considered as they are being developed. As a rule of thumb any requirement that cannot be tested probably cannot be implemented either. The guidance in TR 187 011 [i.10] is quite simple and stresses simplicity in requirement statements. The structure of *precondition-stimulus-response* is the recommended form but others may be valid, particularly for the specification of algorithms and physical parameters where, for example, a key has to be of a certain length to provide a particular level of security.

4.3 Communications security

A communications standard specifies detailed requirements that will have to be met by implementations of the standard in order to be compliant. Depending on the range and complexity of the specified requirements, such standards might be implemented by whole systems or by individual components of the systems. In those cases where implementations are likely to exist within a secure environment, the standard will specify additional security related requirements derived from a thorough Threat, Vulnerability and Risk Analysis (TVRA) as defined in TS 102 165-1 [i.11].

One of the keys to successful system design is the ability to show the relationships which exist between objectives, requirements and the system itself. The distinction between security objectives and security requirements is an important one to make:

- An objective is a broad expression of what a security system should be able to do:
  - A desire.
- A requirement is a detailed specification of how an objective is achieved:
  - A mandate.

Information held by and transmitted between IT products and systems is a critical resource that enables organizations to succeed. It is also reasonable to expect that personal information contained in IT products or systems should:

- remain private;
- be available to authorized users as needed;
- not be subject to unauthorized modification.

IT products and systems will have to exercise proper control of information to ensure that it is protected against hazards such as unwanted or unwarranted dissemination, alteration or loss. The term IT security is used to cover prevention and mitigation of these and similar hazards.

Standards writers have a responsibility to ensure that their standards do not compromise the security capabilities of any product or system implementing them. To achieve this, the following aspects should be considered when writing any communications standard:

- Appropriate development methods are followed to ensure all security issues are addressed.
- The requirements for IT security expressed in ISO/IEC 15408 [i.19] are followed.
- Security requirements are included in any validation of the standard.
- Any requirements for cryptographic algorithms are processed.
- Regulatory issues related to security are addressed.

4.4 Primary security technologies

4.4.1 Confidentiality

Confidentiality measures ensure that communication between two legitimate parties remains confidential if intercepted by a third party. ISO defines confidentiality as "ensuring that information is accessible only to those authorized to have access", in this case the original two parties and not the third.

In the majority of standardization confidentiality is provided by encryption although in many discrete systems confidentiality is also achieved by physical means. Standards may require the use of one or both of these techniques to assure confidentiality.

4.4.2 Integrity

Integrity measures provide assurance that data has not been modified. Whilst a Cyclic Redundancy Check (CRC) is ideal as a means of detecting and correcting non-malicious data errors, it is not an acceptable method for proving the ongoing integrity of an established communications path. If the transmitted data is maliciously modified en route, it is also possible that the CRC value is altered to match the modified data.

Any integrity protection and validation mechanism should involve the following steps:

- Preparation of a digest of the data to be transmitted at the source and appending the digest to the transmission.
- Preparation of a digest of the data received at the destination.
- Comparison of the two digests at the destination.

If the digests are the same there is a high assurance that there has been no manipulation of the data in transit.

There are two forms of integrity check digests: those that use a keyed algorithm and those that do not. The former may also reinforce authentication and thus is referred to as a Message Authentication Code (MAC).

- Keyed digest example SHA-1 [i.33].
- Non-keyed digest example MD5 (RFC1321 [i.39]).

Cryptographic integrity check algorithms, commonly called hashing algorithms, create a digest (mathematical summary of the message) and are considered secure where it is computationally infeasible to find a message that corresponds to a given message digest or to find two different messages that produce the same message digest. The design criteria are such that any change to a message will, with a very high probability, result in a different message digest. This will result in a verification failure when the secure hash algorithm is used with a digital signature algorithm or a keyed-hash message authentication algorithm.

4.4.3 Authenticity

Authenticity measures are used to validate the identity of a legitimate user to the extent that it is difficult for another user to illegally masquerade as the validated user. The person or entity being authenticated is termed "the principal" and authentication methods rely upon:

- something that the principal is;
  - e.g. biometric data.
- something that the principal has;
  - e.g. a token or smartcard.
- something that the principal knows;
  - e.g. a password.

This is sometimes supplemented by how the principal behaves. The methods of achieving authentication fall into two root classes (for cryptographic authentication):

- Challenge - response:
  - Generally used to authenticate a principal on a session-by-session basis, i.e. the user is authenticated by the network and is then trusted for the duration of the user's current session.
  - The authenticator sends a challenge to the principal who responds with a pre-arranged or calculated response which is then checked by the authenticator. The method relies on the inability of an attacker to guess the correct response even with knowledge of the challenge and the algorithm used to generate the response even though the authenticator can predict the correct response. This method relies on a secret shared by the authenticator and the principal.

- Keyed digest:
  - Generally used to authenticate a principal as the source of data on a packet-by-packet basis.
  - The principal processes the data to be transmitted using cryptographic tools to form a digest which is then appended to the transmitted packet. The receiver uses similar tools to produce a digest of the received data and only accepts the data if the two digests match. This method relies on the inability of an attacker to create a match with random tools and data (see also Message Authentication Code (MAC) in clause 0) and the use of asymmetric key protocols such as Transaction Layer Security (TLS) [i.44].

There are a large number of examples of authentication methods in ETSI standards using both the challenge-response and the keyed digest approaches.

EXAMPLE 1: Challenge response authentication is described for TETRA in clause 4 of EN 300 392-7 [i.6].

EXAMPLE 2: MAC authentication is described for infrastructure authentication in 3G in clause 5.1.1 of TS 133 203 [i.13].

As a prerequisite for authentication the user identity has to be known and any key material used in the authentication protocol has to be distributed.

4.4.4 Authority

Authority measures are used to determine to what degree a particular user is permitted access to specific services or data. The degree of access may range from "No Access" to "Full Access" with various partial access levels in between. As an example, individual users may have access to stored data based on a combination of read, write, edit and delete capabilities depending on their level of authority. In a more distributed environment such as in telecommunications the assertions of authority are more complex and require some form of Authority Management Infrastructure (AMI) which can be found in two main suites of protocols and objects:

- Security Assertion Markup Language (SAML) [i.38].
- Privilege Management Infrastructure (PMI) in X.509 Attribute Certificates [i.49].

Authority is dependent on Authenticity and Identity which are both necessary to establish and validate a specific user's authority within a system.

4.5 Secondary security attributes

4.5.1 Availability

Availability refers to the fraction of time a component is providing its full set of services in a manner consistent with its design and intended usage. Availability of a system is related in security in a number of ways. It is possible that a component forms a critical part of the overall security infrastructure. If that component becomes unavailable, either permanently or intermittently, it may be that the security of the system is compromised. In this case the availability of the component will have to be monitored and if it becomes unavailable the system may need to take corrective and/or protective action to avoid creating vulnerability within the system. An example of such a component is a good quality source of entropy for random number generation. If that source is disrupted, it is possible that random numbers can be predicted or easily guessed, undermining the basis of many cryptographic algorithms.

From another perspective, a certain level of availability of a component may be necessary in order to maintain correct system behaviour. As such, mechanisms need to be in place to protect that component from external attacks which may attempt to compromise the integrity of the overall system by reducing the availability of a given component. A well known example of this is the "Denial of Service" (DoS) attack class, where the capability of a computer system (for example, a network switch) is compromised by overloading that system with communication or connections with the intent of overwhelming it.

Standards for algorithms, system architectures, and APIs all need to consider availability of components from these two perspectives.

4.5.2 Reliability

Considering the required degree of reliability within a system and for individual components is an important part of a security review. Failed components are not available, and present the security issues mentioned in the preceding clause. Protocols and architectures need to consider what mechanisms may be necessary to gracefully handle errors and failures, either internally generated, or as a result of external interference (malicious or otherwise). This will typically require a definition of reliability for the affected components and possibly mechanisms to monitor the level of reliability.

4.5.3 Repeatability

Secure systems and protocols will have to behave in a consistent, well defined way. As such, it is essential that standards are reviewed to confirm they define behaviour that is repeatable: given the same set of inputs applied in the same way, a given output should result. Put differently, systems will have to be deterministic.

4.5.4 Resilience

Resilience refers to a system or components ability to cope with hazards and failures gracefully. Determining the necessary level of resilience follows on from a treat and vulnerability analysis, deciding how to cope with various situations. Providing resilience in an ICT environment rarely refers to a physical system and therefore “static” solutions are unlikely to be suitable. Instead, resilience is achieved through reactive components which monitor system behaviour and detect anomalous conditions. In protocols, resilience is typically built-in via graceful handling of out-of-order or garbled messages in a protocol exchange sequence. Building these in to a system requires the designers to think not of what ought to happen, but of what could happen, in particular given a malicious interfacing system.

4.6 Security associations

For each of the prime security attributes there is a link to the technology that provides assurance:

- Authenticity assurance is provided by authentication technologies.
- Integrity assurance is provided by integrity proof creation and validation technologies.
- Confidentiality assurance is provided by encryption technologies.

NOTE: In the present document there is a strong distinction between attributes (ending in "ity" (e.g. authenticity, integrity) and the means to provide them (ending in "tion", e.g. authentication, encryption).

In security for authentication to work (or for confidentiality or integrity or authority to work) there has to be some form of security association between the entities offering (say) an integrity proof and verifying that proof. The strategy of keying (i.e. the secrets that provide security) will have an impact on the form of the security association but essentially a security association exists between the two parties offering a capability and the security association may be described as a link:

- that determines assurance;
- that determines security functionality.

The particular security association has to then define for each linked pair (the partners of the security association):

- Algorithms used for each security capability (e.g. AES-128).
- What security capabilities are available (e.g. authenticity plus integrity (as in a Message Authentication Code)).
- What keys are to be used (e.g. shared secret with key index 9982).

When creating security associations there are a number of rules of thumb to be considered:

- For cryptographic operations One key = one purpose.
- One operation = one purpose.

4.6.1 IPsec and SAs

In many security groups the term security association is itself associated (inaccurately) with the IP security suite as defined in RFC 4301 [i.41] and extended in RFC 4303 [i.42] and RFC 4305 [i.44]. SAs in IPsec can be established both manually or using a number of protocols of which the Internet Key Exchange (IKE) currently at version 2 in RFC 4306 [i.43] is the version cited in the IPsec suite.

5 Risk analysis

5.1 Attacks and attack vectors

5.1.1 Conventional attacks

5.1.1.1 Masquerade

Masquerade is an attack on identity whereby the attacker asserts the identity of a legitimate system user (there is a broad assumption that the attacker is not a legitimate system user). Masquerade is preventable by authentication.

5.1.1.2 Manipulation

Manipulation is an attack in which an object is illicitly modified in some way. Manipulation is detectable, and may be correctable, using forms of message digests (in themselves these are sophisticated cyclic redundancy checks as are normally found in forward error correction encoding schemes).

5.1.1.3 Eavesdropping

Eavesdropping is an attack in which a communication is overheard. In any form of broadcast communication eavesdropping is inevitable, e.g. in a radio communication both wanted and unwanted recipients will receive the message. Whilst it is not possible with certain media (e.g. radio) to prevent eavesdropping it is possible to restrict the ability of the eavesdropper to gain access to the content of the communication by use of encryption.

5.1.2 Social and combination attacks

Hard security attacks are headline gathering but often difficult and specialist. In practice most security is broken by manipulation of the people who run the security systems. Whilst protection against social attacks is not provided in technical standards there are a number of guides to staff training, staff selection, data storage and distribution that if followed may limit the likelihood of social attacks occurring. ISO/IEC 27001 [i.20] covers some of these issues and guidance documents are available from most security or business agencies of the European governments.

Whilst much attention is given to single attacks often an attacker will use combinations of attacks on a system. This is not particularly unrelated to the way in which any battle is fought with attacks on many fronts, some to weaken, some to divert, some to break. There are some tools available to simulate combination attacks (see clause 9.2 on penetration testing) but essentially a system cannot and should be considered secure when it protects only against attacks in isolation. This is also covered in the TVRA [i.11] approach by analysing risk by attack intensity (e.g. a system may be able to protect itself and recover if an attack occurs every 10 seconds but if the intensity of attack is increased to 10 per second the system may prove catastrophically vulnerable).

6 Security boundary analysis and establishment

To borrow from Donald Rumsfeld "… there are known knowns … there are known unknowns … there are also unknown unknowns …" which whilst being unwieldy political speak points to a key problem in security work, that of establishing and proving, a security boundary.

The security boundary is most often referred to for the purposes of a TVRA in TS 102 165-1 [i.11] in terms of a Target of Evaluation as defined in ISO/IEC 15408 [i.19]. Both TS 102 165-1 [i.11] and ISO/IEC 15408 [i.19] assist the reader in identifying and documenting the security boundary.

The rationale for this is probably obvious: A clear description of the ToE or system boundary is necessary in order to build the analysis and to illustrate the rationale for any resulting countermeasure.

In security standardization development the ToE may be a set of software interfaces, a communications protocol, or an architecture, or any combination of them.

7 Countermeasure patterns and specialization

A key conceit of security standards is that there is little new: Masquerade is always countered by authentication; Manipulation is always countered by some form of integrity check value. Moving on from this there are a set of common models or patterns for the security countermeasures that can be specialized for each implementation.

The biggest source of security feature patterns (or templates or outlines or frameworks) is the suite of ISO documents as below:

- ISO/IEC 10181-2: "Information technology -- Open Systems Interconnection -- Security frameworks for open systems: Authentication framework" [i.21]
- ISO/IEC 10181-3: "Information technology -- Open Systems Interconnection -- Security frameworks for open systems: Access control framework" [i.22]
- ISO/IEC 10181-4: "Information technology - Open Systems Interconnection - Security frameworks for open systems: Non-repudiation framework" [i.23]
- ISO/IEC 10181-5: "Information technology -- Open Systems Interconnection -- Security frameworks for open systems: Confidentiality framework" [i.24]
- ISO/IEC 10181-6: "Information technology - Open Systems Interconnection - Security frameworks for open systems: Integrity framework" [i.25]
- ISO/IEC 10181-7: "Information technology -- Open Systems Interconnection -- Security frameworks for open systems: Security audit and alarms framework" [i.26]

In ETSI a number of the patterns from these documents have been adopted and specialized for specific applications and this is shown in TS 102 165-2 [i.12].

8 Cryptographic selection and design

Cryptography is the mathematical toolkit for providing security and has a long history only some of which has been in the public domain. The present document does not aim to guide readers on the mathematical basis for cryptography nor on the details of how algorithms work however it does aim to assist readers in where to find material that may be useful and how to incorporate the result into standards. However it does point to some factors that should be addressed irrespective of the system and which a standards designer should consider carefully in specifying where cryptography is to be applied.

The bulk of cryptography should be driven by Kerckhoffs' principle [i.52] that there should be separation of algorithm (method) and key such that "a cryptosystem should be secure even if everything about the system, except the key, is public knowledge". This principle has been taken to its extreme in the open algorithm design of AES [i.34].

8.1 Specification of algorithms and other cryptographic processes

EG 202 238 [i.2] and EG 200 234 [i.1] define an approach to developing requirements for cryptographic elements that enforce some parts of the security domain. The salient points are brought out here with some additional illustration and comment on where they apply in the act of making better security standards.

Algorithms specifications should be considered complete when the following are documented:

- Use of the algorithm:
  - who are the users of the algorithm;
  - the purposes for which the algorithm is used;
  - the places where the algorithm is used; and
  - the types of implementation (hardware/software).
- Use of the algorithm specification:
  - who will own the algorithm and related test data specification;
  - who will be entitled to use the algorithm specification;
  - any procedures and requirements with respect to licensing and confidentiality agreements; and
  - any procedures needed for distribution and management of the specification.
- Functional requirements:
  - the type of the algorithm and its relevant parameters;
  - the detailed interfaces to the algorithm;
  - modes of operation (if applicable);
  - implementation complexity and operational constraints; and
  - requirements for the strength of the algorithm.
- Algorithm specification and test data:
  - Deliverables might include the algorithm specification (including simulation code), conformance test data (for detailed testing of correctness of implementations), integration test data (for testing the correct response to interface data) and a design and evaluation report (outlining the procedures and results of the design and evaluation work, but not containing technical information on the algorithm).

8.2 Attacks on cryptographic implementations

8.2.1 Brute force attacks

A brute force attack essentially refers to trying every key in turn until the plain text data of an encrypted message is revealed. Illustrative figures of immunity to a brute force attack are based on the key length and the number of key attempts available in a period of time. If there were 1 000 000 000 machines that could try 1 000 000 000 keys/s, it would take all these machines longer than the lifetime of the universe to find the key (i.e. 10^{18} keys/second for a key space of approximately 10^{42} would require 10^{24} seconds to search in order to give a greater than 50 % probability of finding the key (as there are about 3×10^7 seconds/year this is roughly equal to 3×10^{16} years).

Therefore an algorithm with a key space of 128 bits is not vulnerable to brute-force attack using current technology.

NOTE: Key space should be sufficient to evade brute force attack in the lifetime of the transmitted message.

8.2.2 Birthday attacks and cryptographic hash security

A birthday attack is a name used to refer to a class of brute-force attacks. It gets its name from the result that the size of group is only 23 where there is a greater than 50 % probability of two or more people sharing a birthday.

If some function, when supplied with a random input, returns one of *k* equally-likely values, then by repeatedly evaluating the function for different inputs, the same output should be obtained after about $1{,}2k^{1/2}$ attempts. As an example for a message of length 20 bits (i.e. 2^{20}-1 equally likely values) there will be a collision after only 1 223 messages. The entropy of the original message has a significant impact as for a message of length 20 bits the encoding or structure of the message may be such that not all values are equally likely.

NOTE: The birthday problem is solved using the $1{,}2k^{1/2}$ equation with k = 365 that equates to 22,8.

Birthday attacks are often used to find collisions of hash functions and in the context of message digest functions (SHA-1 [i.33], MD5 [i.39]) can be used to give a measure of the relative strength of digests where a birthday attack on a message digest of size *n* produces a collision with a work factor of approximately $2^{n/2}$.

So if the state of the art for brute force cryptanalysis is M bits then a hash should be at least 2M bits in length to give the same factor of "mathematically infeasible to find a collision". However the way in which the actual algorithm is defined may make the strength somewhat less than M.

8.2.3 Message entropy and cryptography

Whilst the present document is not intended to be a security concepts tutorial there are a number of concepts that need to be addressed as is evident in the previous parts of this clause. Any transmitted message will have some variation to all other transmitted messages, however in some communications systems there may be very strong levels of commonality between transmitted messages that may be exploited. A goal of cryptography is to mask the similarity between messages and commonly this is referred to (in the mathematical language of cryptography at least) as maximizing the entropy of a transmitted message. If a message is to be encrypted and the message has low entropy the cryptographer has to raise the entropy prior to encryption or as part of the encryption process.

There are a number of examples of messages with inherently low entropy:

- Short text messages.
- Telematics status messages.
- Call setup messages.

There are a number of mathematical sources that discuss message entropy but at the root is Shannon's "A Mathematical Theory of Communication" [i.40]. Essentially if the attacker knows or guesses that the message can take a small set of values the probability of correctly guessing bit N+1 after receiving bit N tends towards 1 whereas for a random binary alphabet the probability of a correct guess should always be 0,5. In a cryptographic context, where **Alice** is sending a message *m* to **Bob** in the form of a binary string the rule of thumb is that the bigger the entropy of the message *m* the more guesses required by an attacker to guess *m*. After encryption of message *m* to generate message *c* the entropy of *c* should be as high as possible.

9 Security testing

9.1 Protocol testing

For security standards the tools of protocol testing defined in ETSI Technical body MTS apply and cover both interoperability and conformance test types. However whilst testing of a protocol (say for authentication) will show that (for example) an authentication challenge is followed with an authentication response, the result is dependent on the algorithms that are invoked at each end of the link.

9.2 Penetration testing

Penetration tests refer to a particular class of testing of system security carried out in the guise of a hostile agent. In terms of process this class of tests evaluate the security by means of an active analysis of the system for any potential vulnerabilities that may result from poor or improper system configuration, known and/or unknown hardware or software flaws, or operational weaknesses in process or technical countermeasures. The intent of a penetration test is to determine feasibility of an attack and the impact of a successful exploit. Penetration testing should form part of the TVRA although this has obvious difficulties in a standards development environment.

Penetration tests can be conducted in several ways where the most significant difference is the amount of knowledge of the implementation details of the system being tested that are available to the testers.

NOTE: Some of the parameterization of vulnerability defined in TS 102 165-1 [i.11] and in ISO/IEC 15408 [i.19] looks at aspects of the penetration testing environment where the attack is not that of a "friendly" tester acting in the guise of a hostile agent, but the hostile agent himself.

Black box testing assumes no prior knowledge of the system to be tested thus requiring that the testers make some determination of the capability of the system before starting their analysis and tests. White box testing by contrast assumes (by direct provision or in some other way) that the testers have complete knowledge of the system under test

9.2.1 Penetration standards and methods

There are a number of standards and guidelines in the area of penetration testing the methods of which are also to be found in the TVRA approach [i.11] under the guidance to evaluation of the ToE and the attack vectors relevant to exploit weaknesses:

- The Open Source Security Testing Methodology Manual (OSSTMM) [i.50] defines a set of methods for performing security tests and their corresponding metrics. OSSTMM focuses on the technical details of exactly which items need to be tested, what to do before, during, and after a security test, and how to measure the results.
- National Institute of Standards and Technology (NIST) SP 800-115 [i.36] defines a methodology that whilst less detailed than the OSSTMM covers the same ground often by direct reference.
- The Information Systems Security Assessment Framework (ISSAF) Penetration Testing Framework [i.51] is in a draft state (published as a draft in May 2006) and covers a number of areas that are also covered in the TVRA approach tasks of assessing risk by assessing the viability of attacks.

Annex A: Review of US Standards Development Organizations relating to ICT Security Requirements

A.1 ANSI

The American National Standards Institute (ANSI) is not directly responsible for the development of standards but accredits American SDOs (e.g. TIA, NIST, IEEE) and provides oversight of their standards development process. In light of this each SDO has to develop its own development process and ensure it is approved by ANSI. The key elements of the accreditation process however do not address methods of development but covers the following aspects:

1) Openness.

2) Balance and Lack of Dominance.

3) Discrete interest category definitions.

4) Public comment opportunity.

5) Consideration of Views and Objections.

A.2 IEEE

The IEEE has no specific guidelines concerning approaches to security issues in standards development. Inspection of IEEE standards suggests that the *modus operandi* is to use the following techniques:

- Extended Backus-Naur Form (EBNF) syntactic meta-language ISO 14977 [i.28] for defining formal languages.

NOTE 1: This is similar to the ETSI MBS recommendation of using ASN.1 to define data.

- Sequence diagrams to communicate protocol steps.

NOTE 2: This is similar to the ETSI MBS recommendation to use Message Sequence Charts to illustrate protocol sequences.

- Pseudo-code to define particular algorithms.

- Flow-charts to describe processing logic.

NOTE 3: This is similar to the ETSI MBS recommendation to use SDL or UML to define processing logic in protocol specifications.

- Bit-maps to define the bit-level data layout in a data frame.

- Block diagrams to describe processing flow between independent modules.

EXAMPLE: IEEE Std 802.11-2007 [i.53] (Wi-Fi) and IEEE Std 1394-1995 [i.54] (Firewire) provide examples of the employment of the above approach.

The IEEE does not specify conformance testing for implementations or provide accreditation. Typically, specific industry groups take responsibility for testing and accreditation.

A.3 NIST

The National Institute of Standards and Technology (NIST), provides the largest source of security-related standards and standards guidelines within US SDOs. It is a federal agency within the US Department of Commerce. The FIPS-140 standard [i.31] provides an example of comprehensive requirements for cryptographic modules. NIST produces numerous ITC security guides, but these primarily focus on the deployment or operation of ITC systems, rather than on the standardization, design, or implementation of such systems.

A.3.1 FIPS

The Federal Information Processing Standards (FIPS) provide guidelines on systems used within the US Federal government. These are also adopted more widely due to their rigorous nature, verification tests, and implementation certification system. FIPS 140-3 (draft) [i.32], which is scheduled to supersede the 2001 FIPS 140-2 [i.31] addresses cryptographic modules, providing requirements on algorithm design, functionality, physical design constraints, and cryptographic security. It specifies the process for verification/conformance testing, and the requirement for a finite state machine description of the module. It is intended for sensitive but unclassified information and provides four levels of security as follows:

- Level 1:
  - allows for un-tested commodity electronic components (such as a standard personal computer) and software-based cryptographic modules running on un-tested operating systems. It is intended to provide the lowest level of recognized security.
- Level 2:
  - introduces the requirement for tamper-evident casing around an isolated crypto-module and role-based authentication of operators who wish to utilize a set of services provided by the module. The associated operating system which interfaces to the crypto-module will have to meet Common Criteria [i.19] evaluation assurance level 2 (CC-EAL2).
- Level 3:
  - requires that the crypto-module on detection of physical access erases all stored critical security parameters. In addition this level requires physical separation of ports for input and output of plaintext critical security parameters. The interfacing system will have to be designed to meet CC-EAL3.
- Level 4:
  - introduces the requirement to monitor environmental conditions and respond to any dramatic fluctuations in temperature, pressure, or input voltage by erasing CSPs. The interfacing operating system will have to meet CC-EAL4.

NOTE 1: The current FIPS 140-3 [i.32] drafts add a Level 5 which includes radiation protection requirements and inactivity monitoring.

The objective of the FIPS 140-2 [i.31] standard is to define functional requirements that will limit the possibility of disclosure or modification of secure information contained within the crypto-module. The requirements are defined across eleven areas for each of the four levels as below:

1) Cryptographic module specification.
2) Cryptographic module ports and interfaces.
3) Roles, services, and authentication.
4) Finite state model.
5) Physical security.
6) Operational environment.
7) Cryptographic key management.

8) EMI/EMC.

9) Self-tests.

10) Design assurance.

11) Mitigation of other risks.

NOTE 2: The proposed draft of FIPS 140-3 [i.32] retains a multi-area security concept but does not retain the 11 areas (or categories) from FIPS 140-2 [i.31].

There is no specific guidance in FIPS 140-2 [i.31] on how to achieve either the FIPS 140-2 [i.31] levels or the equivalent CC levels although the guidance from ETSI in EG 202 387 [i.3] should apply.

A.4 TIA

TIA does not provide any specific guidance on security standards or security considerations in any of its standards authoring manuals. TIA has several standards for cryptographic algorithms and IMS security frameworks although these are based on the 3GPP specifications.

A.5 IETF

The Internet Engineering Task Force mandates that every RFC includes a section on "Security Considerations" but does not give explicit guidance on how these are to be derived, verified and presented.

History

Document history		
V1.1.1	February 2009	Publication