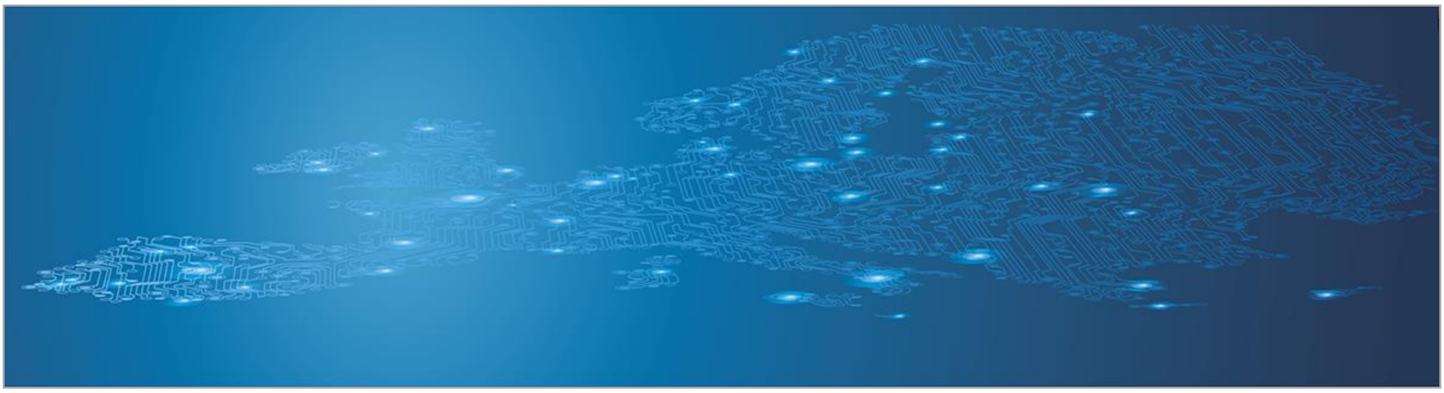

Standardisation in the field of Electronic Identities and Trust Service Providers

Inventory of activities

Version 1.0, December 2014

European Union Agency for Network and Information Security www.enisa.europa.eu

About ENISA

The European Union Agency for Network and Information Security (ENISA) is a centre of network and information security expertise for the EU, its member states, the private sector and Europe's citizens. ENISA works with these groups to develop advice and recommendations on good practice in information security. It assists EU member states in implementing relevant EU legislation and works to improve the resilience of Europe's critical information infrastructure and networks. ENISA seeks to enhance existing expertise in EU member states by supporting the development of cross-border communities committed to improving network and information security throughout the EU. More information about ENISA and its work can be found at www.enisa.europa.eu.

Editor

Sławomir Górniak – ENISA

Contact

For contacting the authors please use sta@enisa.europa.eu

For media enquires about this paper, please use press@enisa.europa.eu.

Acknowledgements

ENISA would like to thank the company CryptoExperts SAS and the following experts for their valuable input to this paper, consisting in the proposal of a new standard: Anne Canteaut (Inria), Antoine Joux (CryptoExperts), David Pointcheval (CNRS/ENS/Inria).

Legal notice

Notice must be taken that this publication represents the views and interpretations of the authors and editors, unless stated otherwise. This publication should not be construed to be a legal action of ENISA or the ENISA bodies unless adopted pursuant to the Regulation (EU) No 526/2013. This publication does not necessarily represent state-of the-art and ENISA may update it from time to time.

Third-party sources are quoted as appropriate. ENISA is not responsible for the content of the external sources including external websites referenced in this publication.

This publication is intended for information purposes only. It must be accessible free of charge. Neither ENISA nor any person acting on its behalf is responsible for the use that might be made of the information contained in this publication.

Copyright Notice

ISBN 978-92-95032-40-8

Executive summary

In order to remove barriers for cross-border trust services and having regard to results from European projects like STORK[1], which have shown that technical issues of interoperability can be overcome, on 27 July 2014 the European Parliament and the Council of the European Union adopted the Regulation on electronic identification and trusted services for electronic transactions in the internal market that replaced the Directive 1999/93/EC on a community framework for electronic signatures, which provided for the legal recognition of electronic signatures. This Regulation strengthens the provisions for interoperability and mutual recognition of electronic identification schemes across borders, enhances current rules for electronic signatures and provides a legal framework for other types of trust services (electronic seals, electronic delivery services, electronic documents, time stamping services and web site authentication).

At the same time, in the field of promoting a Single Market for cybersecurity products, the cyber security strategy underlines the importance of CSCG and ENISA, by stating: "*the Commission will support the development of security standards*"; "*Such work should build on the on-going standardisation work of the European Standardisation Organisations (CEN, CENELEC and ETSI), of the Cybersecurity Coordination Group (CSCG) as well as on the expertise of ENISA, the Commission and other relevant players*".

This paper explains why standards are important for cyber security, specifically in the area of electronic identification and trust services providers. A number of challenges associated with the definition and deployment of standards in the area of cyber security are discussed. This is followed by a brief overview of several key EU initiatives in this area.

The paper also discusses concrete standardisation activities associated with electronic IDs and trust service providers, providing an overview of standards developed under the mandate m460 from the European Commission and others, related to eIDAS Regulation. It concludes with a proposal of a standard on cryptographic suites for electronic signatures and infrastructures, put forward by ENISA and related to the ETSI TS 119 312.

[1] https://www.eid-stork.eu/

Table of Contents

Executive summary iii

1 Introduction 1

2 Importance of standards in information security 2

3 Standardisation challenges in Cyber Security 3

3.1 Organisational challenges 3

3.2 Areas of standardisation 3

3.3 Lack of agility 3

3.4 Competing sets of standards 4

3.5 Economic considerations 4

3.6 Lack of awareness 4

4 Cyber security strategy of the European Union 5

5 Cyber Security Coordination Group 7

6 Strategy towards standardisation options 8

7 Standardisation activities in the area of Electronic Signatures and Trust Service Providers 9

8 Proposal for replacement of ETSI TS 119 312 13

Annex 1 – Proposal for replacement of ETSI TS 119 312 15

Annex 2 – References 24

1 Introduction

In the *Cyber Security Strategy of the EU*[2], the European Union reaffirms the importance of all stakeholders in the current Internet governance model and supports the multi-stakeholder governance approach. Indeed, the multi-stakeholder approach is fundamental to the development of successful standards, particularly in the area of cyber security where public sector requirements are implemented to a large extent by private sector service providers.

In the field of promoting a Single Market for cybersecurity products, the cyber security strategy underlines the importance of the Cybersescurity Coordination Group[3] and ENISA. It states, among others, that: "*the Commission will support the development of security standards*"; "*Such work should build on the on-going standardisation work of the European Standardisation Organisations (CEN, CENELEC and ETSI), of the Cybersecurity Coordination Group (CSCG) as well as on the expertise of ENISA, the Commission and other relevant players*".

A number of EU governments are now promoting the broader adoption and use of standards. A good example is a standardisation policy for software interoperability, data and document formats in government IT specifications, published by the UK government.[4] Standards also play an important role in the EU's Digital Agenda. Quoting the then European Commission's Vice President Neelie Kroes, they "*create competition, lead to innovation, and save money*".

Within the private sector, industrial interest in standardisation activities in the area of NIS tends to be driven by areas of work that are in line with the core interests of product developers or service providers. Aligning public sector goals with standardisation priorities of the private sector remains challenging.

Where information security is concerned, there is clearly room for improvement in identifying and responding to evolving risks and technology developments. In particular, the time lag between the appearance of a new technology or technically driven business model and the availability of applicable standards is still too long.

[2] http://ec.europa.eu/information_society/newsroom/cf/dae/document.cfm?doc_id=1667

[3] http://www.cencenelec.eu/standards/Sectors/DefenceSecurityPrivacy/Security/Pages/Cybersecurity.aspx

[4] Like https://www.gov.uk/government/publications/open-standards-principles

2 Importance of standards in information security

There are many reasons why standards have an important role to play in improving approaches to information security that involve different geographical regions or different communities. Some of the more important reasons include:

- Improving efficiency and effectiveness of key processes.
- Facilitating systems integration and interoperability
- Enabling different products or methods to be compared in a meaningful manner.
- Providing a means for users to assess new products or services.
- Structuring the approach to deploying new technologies or business models.
- Simplification of complex environments.
- Promoting economic growth.

Standardising processes and procedures is an essential part of achieving successful cooperation in a cross-border or cross-community multi-vendor environment. Without such standardisation, communication is likely to be inefficient and could result in a process that is ineffective. An illustrative example is provided by the way in which different countries would react to a significant cyber incident. Here, in line with the principle of subsidiarity and the need to preserve sovereign state control, decision making is made in a distributed environment and the processes that support this process must be optimal. Standardised operating procedures could help ensure that various countries can interact with each other according to one set of predefined and agreed procedures.

Similarly, specifications such as ISO 27001 encourage the adoption of a standard organization structure, which makes it easier for customers to understand how processes work and also reduces the costs of auditing and due diligence. This is largely due to the fact that these organisational standards provide a blue-print for setting up a management system for security, but also for a blue-print for auditing and checking compliance of an organisation to security best practices.

Standards play a key role in ensuring that security products can be put together into systems capable of detecting and responding to real events. In particular, standard interfaces and protocols make systems integration much simpler and allow products to interoperate in heterogeneous environments. Standardisation of testing methods also makes it possible to compare security products in a meaningful manner ('benchmarking') and provide a means for the end user to assess new products or services. The level of compatibility of cryptographic modules with the FIPS 140-2 standard (which is used to accredit such products) for instance is used to assess the ability of such products to meet certain security requirements.

By structuring the approach to deploying new technologies or business models, standards help to reduce the complexity of the business environments that deploy them, which in turn makes it easier to secure the resulting environment. Although there is also an argument against standardisation in this respect, notably that any vulnerabilites associated with such systems will also be 'standardised', making it possible to conduct attacks against large numbers of systems in a short timescale. The usual way of dealing with this however is not to avoid standardisation but to ensure that the defences used to protect information systems are not critically dependent on a single system or type of system – this is the principle of defence in depth.

Last but not least, the use of standards encourages information exchange among developers and it is likely to result in greater competition among product developers.

All these factors have a great impact on the overall preparedness of the governments to the cyber threat. Standardised technologies and approaches enhance harmonisation among cooperating countries, ensure a larger pool of experts available and higher level of knowledge of systems deployed.

3 Standardisation challenges in Cyber Security

Despite the fact that an appropriate use of standards is clearly beneficial to achieving a strong approach to security in a cross-border environment, there are also many challenges to achieving this in practice.

3.1 Organisational challenges

Over the last ten years a plethora of SDOs (Standard Development Organisations) have been created. In many occasions these organisations have been initiated by industry (e.g. Oasis, W3C, Open Data Center, IETF, Adobe, ITIL and many others) to a certain extent as a reaction of the industry to the large investment in terms of time and people required by 'traditional' SDOs (such as ETSI, CEN-CENELEC, ISO, ITU) and partially the result of convergence where standardisation fora that traditionally focused on a specific sector (e.g. IEEE) found applicability in many different business sectors. The number of SDOs and the number of published standards has increased, which can be a source of confusion to end users.

3.2 Areas of standardisation

Industrial interest in standardisation activities in the area of NIS tends to be driven by areas of work that lay in line with the core interests of service providers (for example authentication, billing, etc.). Although an increased general interest in the area of privacy is observed, specific interest of industry is expected to become lower since privacy enhancing technologies are perceived as being in conflict with commercial expectations.

At the time of writing, there is no single, continuous "line of standards" related to cyber security, but rather a number of discrete areas which are the subject of standardisation:

- Technical standards
- Metrics (related mostly to business continuity)
- Definitions
- Organisational aspects

Some areas could be potentially considered as over-standardised. There are several standards on information security governance and risk management.

In some areas standards are lacking, for example there are relatively few standards that deal with compliance to privacy and data protection legislation. Similarly, there are not many standards covering service levels, or more broadly, service agreements and service contracts, terms of use and conditions, et cetera. A quick look across the different offerings of cloud providers will show that every provider has a different lengthy legal text describing the terms of use and exceptions to obligations.

3.3 Lack of agility

Designing and agreeing standards is a lengthy process, measured in years. The IT landscape on the other hand evolves rapidly and, in order to remain useful, standards need to evolve at a comparable pace. Failure to do so will result in standards that are either obsolete or only partially applicable to real life environments.

One solution to this issue could be sought in the direction of using 'good practices' as precursors for standards. Good practicesare generally subjected to change control procedures that are much less stringent than those applied to candidate standards and could therefore be developed to maturity more quickly. Good practices that are sufficiently mature could then be used as a basis for a corresponding standard.

With regard to standards, a 'fast track' mechanism could be developed and agreed among interested parties, to be able to publish non-controversial standards in a quicker manner.

3.4 Competing sets of standards

In some areas of information security there are several different groups of standards that are defined. To some extent, these standards are competing with each other for adoption and it is often difficult for the end user to judge which standards are the best choice for their particular requirements. Occasionally, it is necessary to mix and match standards from different families in order to achieve the goal. When implementing Public key Infrastructure (PKI) for instance, it is not unusual to see organisations adopt such a combination of standards (for example X.509 (ITU) for the certificate format, PKIX (IETF) standards for core PKI and PKCS (RSA) standards for interfacing to secure devices).

3.5 Economic considerations

Although some providers see their use of recognised standards as a unique selling point, there are also many cases of vendors who have a dominant position, who insist on their own proprietary standards and fail to constructively support and implement standards for their products. For instance, the fact that every mobile phone vendor uses different charger plugs is annoying for consumers, and it is wasteful in terms of resources. In order to resolve this situation, the EU followed up an industry initiative to adopt a single standard universal mobile phone charger plug.

Companies with a dominant position have few incentives to adopt interoperable standards, because it would only reinforce the position of competitors. For a dominant vendor there are advantages to using proprietary standards, because they lock-in the customer. Lock-in means that:

- The customer cannot buy or integrate with compatible products from competitors, which generates more revenue for the provider.
- It is hard for customers to switch to another supplier, because they cannot easily move their data and processes to a competitor.

3.6 Lack of awareness

Despite the clear disadvantages associated with the use of proprietary standards, there are still many examples of cases where customers (especially in this context we consider as 'customers' national authorities, governmental organisations, etc.) fail to demand open standards. This may well be due to a lack of awareness of the existence of such standards.

4 Cyber security strategy of the European Union

The European Commission published the Cybersecurity strategy of the European Union[5] on 4 February 2013. This strategy provides a harmonised framework for the evolution of three different aspects of cyber security, which until recently had been evolving independently. In so doing, the Commission recognised and responded to the need to bring different communities together to improve the approach to cyber security across the EU and laid the foundations for a more coordinated approach. The Cyber Security Strategy of the EU also includes a proposal for a Directive on Network and Information Security (NIS) requiring the Member States (MS) to have minimum NIS capabilities in place, to cooperate and exchange information within a dedicated network and requiring the private sector to adopt NIS enhancing actions.

- The EU reaffirms the importance of commercial and non-governmental entities, involved in the day-to-day management of Internet standards
- A prime focus should be to create incentives to carry out appropriate risk management and adopt security standards and solutions, as well as possibly establish voluntary EU-wide certification schemes building on existing schemes in the EU and internationally
- the Commission will support the development of security standards and assist with EU-wide voluntary certification schemes in the area of cloud computing

Under strategic objective 4, the Commission asks ENISA to '*develop, in cooperation with relevant national competent authorities, relevant stakeholders, International and European standardisation bodies and the European Commission Joint Research Centre, technical guidelines and recommendations for the adoption of NIS standards and good practices in the public and private sectors*.'

This is a timely recommendation as the new ENISA mandate gives the Agency a more proactive role in this area. The task assigned to ENISA by the new ENISA regulation in this area is to 'support research and development and standardisation, by facilitating the establishment and take up of European and international standards for risk management and for the security of electronic products, networks and services'.

There are also recommendations for public and private stakeholders. In particular '*The Commission invites public and private stakeholders to:*

- *Stimulate the development and adoption of industry-led security standards, technical norms and security-by-design and privacy-by-design principles by ICT product manufacturers and service providers, including cloud providers; new generations of software and hardware should be equipped with stronger, embedded and user-friendly security features.*
- *Develop industry-led standards for companies' performance on cybersecurity and improve the information available to the public by developing security labels or kite marks helping the consumer navigate the market.*
- *An important part of the cyber security strategy is the proposal for a Network and Information Security (NIS) Directive. This Directive asks the Member States to support standardisation in the area of NIS:*
- *Given the global nature of NIS problems, there is a need for closer international cooperation to improve security standards and information exchange, and promote a common global approach to NIS issues.*

[5] http://ec.europa.eu/information_society/newsroom/cf/dae/document.cfm?doc_id=1667

- *Standardisation of security requirements is a market-driven process. To ensure a convergent application of security standards, Member States should encourage compliance or conformity with specified standards to ensure a high level of security at Union level. To this end, it might be necessary to draft harmonised standards*.

Article 16 on standardisation states the following:

- *…..Member States shall encourage the use of standards and/or specifications to networks and information security.*
- *The Commission shall draw up, by means of implementing acts a list of the standards referred to in paragraph 1. The list shall be published in the Official Journal of the European Union*

5 Cyber Security Coordination Group

In 2011, following a request of the Commission, the Standards Development Organizations (SDOs) CEN, CENELEC and ETSI have created the CEN-CENELEC-ETSI 'Cyber Security Coordination Group' (CSCG) for strategic advice in the field of IT security, Network and Information Security and cyber security. The main objectives of the CSCG are to

- Establish a European standardisation roadmap in the above mentioned areas
- Act as the main contact point for all questions by EU institutions related to standardisation issues
- Define and propose to the Commission a cooperation strategy between the EU and the US for the establishment of a framework, relating to standardisation of cyber security.

The European Union Agency for Network and Information Security (ENISA) has participated and contributed to the activities of the CSCG since its launch. A first white paper was addressed by the members of CSCG to the Commission with strategic advice on the priorities for R&D of EU funded research in the area and how to optimise EU research with mandates for cyber security standardisation.

6 Strategy towards standardisation options

One of the issues that the European Union needs to address is the strategy towards standardisation in the area of ICT. The current approach is not consistent and lacks a unified vision. In this light the EC has taken an initiative in 2011 in order to promote a coordinated approach at EU level. For this purpose the Commission (DG CONNECT) supported also by ENISA has identified possible alternatives and options briefly summarized below.[6] High level strategic options for recommendations on security standards could be:

1. General recommendations

They can only be applied to specific cases, otherwise they are not considered by any communities, no more than very generic recommendations on security in general.

2. Recommendations targeting organizations (such as ISO 27000 for the management of information security or ISO 31000 for risk assessment within organizations)

Very costly and possibly limiting innovations. This option has a lot of potential if implemented in a correct (and acceptable by industry) way. A European framework for standards would be 'nice to have' on one hand, but on the other would be very costly, would require a lot of resources (in terms of research and following-up related activities). However, the adoption of standards could be enforced by the European legislation and national competent authorities (for example requiring defined standards to be applied in order to get authorization to perform certain activities, like provision of ICT services).

3. Specific recommendations for products / services with dedicated standards (similar to Common Criteria)

Complicated approach presenting (among others) a problem in the definition of specific products or services. In the world where most ICT services are converging, identifying a 'class' of products is a challenge.

4. Recommendations on functions / products / services using a mash-up approach

Such a “mash-up" approach could be an ad hoc solution, where functions, products, services would need to be selected following an appropriate process

At the EU level it is important to take advantage of Framework Programs of EU funded R&D (FP7) by funding flagship project/initiatives with clear standardization objectives. The additional benefit of this approach is that by definition these research projects have strong industrial participation that could be also 'channeled' towards strategic standardization initiatives. For example in the area of Attribute Based Credentials (ABC) the Commission is funding an interesting Integrated Project that makes use of IPRs of US based companies (mainly MS and Intel). Even in such (difficult) cases all efforts should be made for strategic contributions at ETSI.

[6] *Strategic options for recommendations on the introduction of security standards*, draft, European Commission, December 2011

7 Standardisation activities in the area of Electronic Signatures and Trust Service Providers

In order to create a rationalised framework of the existing European eSignature standardisation deliverables, supporting also the realisation of the items of the Action Plan related to eSignature and future adoption of the Regulation on eIDs and TSPs (now known under number 910/2014), the Commission issued standardisation mandate (m460) to CEN and ETSI in 2010. In July 2012 these two standardisation bodies jointly published the "Rationalised Framework for electronic signature"[7].

The tables below constitute an inventory of standardisation activities in the areas covered by the adopted eIDAS Regulation. Most of these standards are in a phase of development or update, their publication is expected between 2015 and 2016.

Standards on policy requirements

Reference	Short Title	Publisher
TS 102 042	Policy requirements for Certification Authorities issuing public key certificates	ETSI
TS 102 023	Policy requirements for time-stamping authorities	ETSI
TS 102 158	Policy requirements for Certification Service Providers issuing attribute certificates usable with Qualified certificates	ETSI
EN 319 411-1	Common policy requirements for certification authorities	ETSI
EN 319 411-2	Policy requirements for certification authorities issuing qualified certificates	ETSI
EN 319 411-3	Policy requirements for Certification Authorities issuing public key certificates	ETSI
EN 319 421	Policy Requirements for Trust Service Providers providing Time-Stamping Services	ETSI
EN 319 101	Policy requirements for certification authorities issuing qualified certificates	ETSI
EN 319 511	Policy & Security Requirements for Registered Electronic Mail (REM) Service Providers	ETSI
EN 319 521	Policy & Security Requirements for Data Preservation Service Providers (DPSPs)	ETSI

[7] http://www.e-signatures-standards.eu/reference-documentation/standardisation-mandate-and-framework/rationalised-structure-for-electronic-signature-standardisation-version-09-2013

Standards on certificate profiles

Reference	Short Title	Publisher
EN 319 111	Protection Profiles for Signature Creation & Validation Applications	ETSI
EN 319 412-1	Profiles for Trust Service Providers issuing certificates; Part 1: Overview and common data structures	ETSI
EN 319 412-2	Profiles for Trust Service Providers issuing certificates; Part 2: Certificate Profile for certificates issued to natural persons	ETSI
EN 319 412-3	Profiles for Trust Service Providers issuing certificates; Part 3: Certificate profile for certificates issued to legal persons	ETSI
EN 319 412-4	Profiles for Trust Service Providers issuing certificates; Part 4: Certificate profile for web site certificates issued to organizations	ETSI
EN 319 412-5	Profiles for Trust Service Providers issuing certificates; Part 5: Extension for Qualified Certificate profile	ETSI

Standards on electronic signatures

Reference	Short Title	Publisher
SR 019 530	Rationalised framework of Standards for Electronic Delivery Applying Electronic Signatures	ETSI
TR 119 300	Electronic Signatures and Infrastructures (ESI) – Business Guidance on Cryptographic Suites	ETSI
TS 119 312	Electronic Signatures and Infrastructures – Cryptographic Suites	ETSI
EN 319 102	Procedures for Signature Creation and Validation	ETSI
EN 319 122	CMS Advanced Electronic Signatures (CAdES)	ETSI
EN 319 132	Electronic Signatures and Infrastructures (ESI); XML Advanced Electronic Signatures (XAdES)	ETSI
EN 319 142	Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signatures (PAdES)	ETSI
EN 319 152	Advanced Electronic Signatures in Mobile Environments	ETSI
EN 319 162	Associated Signature Containers (ASiC)	ETSI
EN 319 172	Signature Policies	ETSI
EN 319 441	Policy and Security Requirements for TSPs providing Signature validation Services	ETSI
EN 419 211	Protection profiles for secure signature creation device CEN	ETSI
EN 419 221	Security requirements for trustworthy systems managing certificates for electronic signature	ETSI

Standards on the Trusted List

Reference	Short Title	Publisher
EN 319 601	General Policy & Security Requirements for Trust Service Status Lists Providers	ETSI
EN 319 602	Trust Service Status Lists Format	ETSI
EN 319 611	Policy & Security Requirements for Trusted List Providers	ETSI
EN 319 612	Trusted list format	ETSI

Standards on Time Stamping

Reference	Short Title	Publisher
EN 319 422	Profile for Trust Service Providers providing Time-Stamping Services	ETSI
EN 419 231	Security requirements for trustworthy systems supporting time-stamping	ETSI

Standards related to Conformity Assessment Bodies

Reference	Short Title	Publisher
SR 003 091	Recommendations on Governance and Audit Regime for CAB Forum Extended Validation and Baseline Certificates	ETSI
TR 101 564	Guidance on ETSI TS 102 042 for Issuing Extended Validation Certificates for Auditors and CSPs	ETSI
TR 103 123	Guidance for Auditors and CSPs on ETSI TS 102 042 for Issuing Publicly-Trusted TLS/SSL Certificates	ETSI
TS 103 090	Conformity Assessment for Trust Service Providers issuing Extended Validation Certificates	ETSI
TS 119 403	Trust Service Provider Conformity Assessment - General requirements and guidance	ETSI
EN 319 103	Conformity Assessment for Signature Creation & Validation Applications	ETSI
EN 319 403	Trust Service Provider Conformity Assessment	ETSI
ISO 17065	Conformity assessment - Requirements for bodies certifying products, processes and services	ISO
ISO 17020	Conformity assessment -- Requirements for the operation of various types of bodies performing inspection	ISO

Other standards to take into consideration

Reference	Short Title	Publisher
RFC 3161	Internet X.509 Public Key Infrastructure – Time Stamp Protocol	IETF
RFC 3647	Internet X.509 Public Key Infrastructure – Certificate Policy and Certification Practices Framework	IETF
RFC 5280	Internet X.509 Public Key Infrastructure – Certificate and CRL profile	IETF
RFC 6960	Internet X.509 Public Key Infrastructure – Online Certificate Status Protocol – OCSP	IETF
ISO 15408	Evaluation criteria for IT security (Qualified Signature Creation Devices security evaluation)	ISO
ISO 18045	Methodology for IT security evaluation (Qualified Signature and Seal Creation Devices security evaluation	ISO
ISO 27000	Information security management systems	ISO
ISO 31000	Family of standards related to risk management	ISO
EN 419 241	Security requirements for trustworthy systems supporting Server Signing	ETSI

8 Proposal for replacement of ETSI TS 119 312

The technical specification on cryptographic suites for use with electronic signatures and seals was developed by ETSI under the name TS 119 312. The goal of this standard is to provide guidance on which algorithms and key sizes should be used for the creation of electronic signatures. It has been the subject of modifications and updates during the years. The current ETSI document in some places shows a lack of adaptation to the current developments.

In 2013 ENISA has published a report "Algorithms, Key Sizes and Parameters"[8] ("ENISA document") in which sufficient cryptographic solutions to use at the moment of publication are discussed. This document can be related to the ETSI TS 119 312. In order to support the standardisation in the field of electronic identification and trust service providers, having regard to the developed capabilities of ENISA, the Agency has commented on the TS 119 312 in 2014 and hereby proposes a new, unified version of the possible standard (in Annex 1). The rationale for changes and for the new version is as explained below.

- The document provides now three lists:
  - Algorithms and key sizes to be used for the creation of electronic signatures
  - Algorithms and key sizes still in use, but to be phased out
  - Algorithms and key sizes not to be used anymore
- A new SHA-3 standard[9] was added to the document, addressing also the French[10] and German[11] agencies recommendations
- Security requirements at the 128-bit level has been homogenized, following, in particular, the recommendations of the ENISA document. One typical consequence is the removal for the white list of the hash functions on 224 bits.
- Extendable hash functions, which are included in the SHA-3 standard, were added. Such functions deserve consideration because they simplify the description of padding schemes such as the full-domain-hash and PSS. Indeed, these schemes require such functions and before their inclusion in the SHA-3 standard, therefore it was necessary to explain how they can be derived from ordinary hash functions.
- Section on RSA has been modified in order to remove some technical inconsistencies, to take into account the shared prime factors attacks. A proposal has been made that the public exponent should be prime, which has no effect on the most common choice e=65537.
- It was proposed to remove DSA from the standard due to the absence of a security proof and also to the attack of Vaudenay from PKC 2003[12]. This is in line with the ENISA document, which recommends to prefer Schnorr algorithm to DSA. Instead, EC-Schnorr was added.

[8] Reference XXX

[9] Draft FIPS Publication 202 (2014). SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions. Available at http://csrc.nist.gov/publications/PubsDrafts.html#FIPS-202

[10] Agence nationale de la sécurité des systèmes d'information, Référentiel Général de Sécurité version 1.0, 2010-01. Available at http://www.ssi.gouv.fr/IMG/pdf/RGS_B_1.pdf

[11] Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen, Übersicht über geeignete Algorithmen, 2014-01. Available at http://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/QES/Veroeffentlichungen/Algorithmen/2014Algorithmenkatalog.pdf

[12] Serge Vaudenay, The security of DSA and ECDSA. Advances in Cryptology — PKC 2003. Lecture Notes in Computer Science 2567. Pages 309-323. Springer; http://infoscience.epfl.ch/record/99503?ln=en

- Technical criteria that prevent the use of CM-curve were added to the construction of elliptic curves, to make them simpler and easier to test. A paper of Sutherland[13] was considered, which improves the construction of CM-curves.
- Description and presentation of the numerous EC-DSA variants was simplified.
- PCKS#1 v 1.5 was removed from the document, in line with the ENISA document.
- Introduction of the full domain hash padding was added to the document.[14]
- In the proposal of the new document, the use of Nextprime in parameter generation was added. For efficiency reasons, Nextprime is usually based on pseudo-primality tests.[15]

[13] Andrew V. Sutherland, Computing Hilbert class polynomials with the Chinese remainder theorem. Mathematics of Computation 80 (2011), Pages 501-538

[14] Saqib Kakvi, Eike Kiltz, Optimal Security Proofs for Full Domain Hash, Revisited, Advances in Cryptology — EUROCRYPT 2012. Lecture Notes in Computer Science 7237. Pages 537-553. Springer

[15] In this case, the probability threshold given in the Note 1 of the first section is sufficient.

Annex 1 – Proposal for replacement of ETSI TS 119 312

1 Expected security level

As generally recommended by the academic cryptography, the minimum security level for medium or long-term security corresponds to 128-bit symmetric keys. With this key size, exhaustive search remains out of range for at least 50 years, even assuming that Moore's law continues during the full 50 years period with a doubling of computing power for unit cost every year.

In addition, cryptographic schemes offering some formal security argument should be preferred to schemes whose security is only based on the fact that they have not yet been attacked.

However, this disregards the possibility of quantum computers becoming available during this timeframe. To account for this possibility, the recommended key size is 256 bits.

For shorter-term security, a 100-bit key is sufficient. Thus, it might be considered for short-term interoperability purposes.

NOTE 1: This security level is required to resist adversaries. To protect against bad random events that may occur naturally but cannot be manipulated by an adversary, requirements are less drastic. Typically, for aircrafts, it is often required that the probability of a catastrophic event should remain below 10^{-9} per hour. As a consequence, after taking into account the long time period and the large number of users, a probability threshold of 2^{-80} per individual event is sufficient for the purpose of the document.

2 Hash Functions

Hash functions are keyless cryptographic primitives that need to satisfy several security properties. In particular, they should resist pre-image and collision attacks. Collision resistance is the strongest of the two requirements and, due to a generic birthday paradox attack, it requires the output of the function to be at least twice as big as the desired security level. For 128-bit security, the minimal output size is 256 bits.

In addition, the recommended hash functions are well known and have shown resistance against the best cryptanalytic effort of the academic cryptographic community.

The following table includes three lists of hash functions. The first list contains recommended algorithm. The second list contains algorithms that might be usable in specific applications. These algorithms are not recommended either because of their output size or because preferable alternatives are recommended. The third list is a blacklist of algorithms that should no longer be used. This list is not exhaustive; it only mentions frequently encountered hash functions, which are not considered secure with current standards.

White List **Recommended algorithms**	**Grey List** **Usable only for legacy or interoperability purposes**	**Black List** **Don't use for secure applications**
SHA-512; SHA-512/256	SHA-224; SHA-512/224	SHA(-0), SHA-1
SHA-384	WHIRLPOOL	MD2, MD4, MD5
SHA-256	SHA3-224	RIPEMD

SHA-3 (Waiting for standard) SHA3-256; SHA3-384; SHA3-512	SHA-3 finalists (BLAKE, Grostl, JH, Skein)	HAS-160 (from EC-KCDSA)
SHAKE-128, SHAKE-256 (waiting for SHA-3)		SipHash

2.1 Recommended Hash Functions

This section details the main properties of the recommended hash functions.

2.1.1 SHA-2 family

The SHA-2 family of algorithms is described in FIPS Publication 180-4.

The recommended algorithms SHA-256, SHA-384, SHA-512 and SHA-512/256 are part of this family.

SHA-256 can be used for messages with length up to 2^{64}-1 bits.

SHA-384, SHA-512 and SHA-512/256 apply to messages with length up to 2^{128}-1 bits.

The output size of the message digest, i.e. of the hash function output, is 256 bits for SHA-256 and SHA-512/256, 384 bits for SHA-384 and 512 bits for SHA-512.

All these algorithms shall be implemented as defined in FIPS Publication 180-4.

NOTE 1: Whenever possible, SHA-512/256 should be preferred to SHA-256, due to its larger inner state and its apparently higher security margins.

NOTE 2: FIPS Publication 180-4 also specifies SHA-1, SHA-224 and SHA-512/224, which are not part of the recommended list of hash functions.

2.1.2 SHA-3 family

The SHA-3 family of algorithms is described in DRAFT FIPS Publication 202.

The recommended algorithms SHA3-256, SHA3-384 and SHA3-512 are part of this family.

SHA-3 algorithms do not have a limit on the length of input messages.

The output size of the message digest is 256 bits for SHA3-256, 384 bits for SHA3-384 and 512 bits for SHA3-512.

All these algorithms shall be implemented as defined in DRAFT FIPS Publication 202.

NOTE: DRAFT FIPS Publication 202 also specify SHA3-224, which is not part of the recommended list of hash functions.

2.1.3 SHA-3 extendable output functions

In addition to the SHA-3 algorithms, DRAFT FIPS Publication 202 also specifies two extendable hash functions SHAKE-128 and SHAKE-256.

These algorithms can be used whenever longer message digest are requested. In particular, they are well suited for use with the Full-Domain Hash signature padding.

SHAKE-128 and SHAKE-256 shall be implemented as defined in DRAFT FIPS Publication 202.

NOTE 1: Using SHAKE-128 for output length up to 256 bits is not recommended. Using SHAKE-256 for output length up to 512 bits is not recommened.

NOTE 2: Using SHAKE on the same message to produce two digests of different sizes must be avoided at all costs. Indeed, in this case, the shorter digest is a truncation of the longest one. This can lead to various attacks against cryptographic protocols.

As explained in Appendix A of DRAFT FIPS Publication 202, to avoid this problem, tags can be added into the input message to indicate the use and length of the output (in order to provide a form of domain separation).

3 Signature Suites

A signature suite is a public key cryptosystem that can be used to sign or verify arbitrary messages, possibly with a length limit. For efficiency, it combines a hash function that is used to represent the given message by a much shorter digest and a public key algorithm for signing and verifying. This document includes two large families of signature suites, one based on RSA and the hardness of factoring and the other based on the discrete logarithm problem.

Every signature scheme consists of three algorithms, one for generating public/private key-pairs, one for signing an element of the message set and one for verifying the signature of such an element. These algorithms include calls to hash functions and the security and validity of signatures are deeply linked to the hash function that is used. Implementers should ensure that attackers cannot manipulate the users into using weak hash functions when signing messages.

The signature algorithm of a given suite takes as input a private key and a message, it outputs a signature; the verification algorithm takes as input the corresponding public key, the message and signature to verify, it outputs Valid or Invalid.

NOTE: There also exists a different flavor of signature schemes, signature with message recovery, which are not considered in this document.

3.1 RSA Signatures

RSA signature suites include two main components: a basic signature scheme and a padding method.

The basic signature scheme is a public key cryptosystem that can be used to sign or verify special messages that follow a uniform probability distribution in the set of integers modulo the RSA modulus. This scheme cannot be directly used to sign arbitrary messages. For this purpose, it needs to be integrated into a signature suite by adding a padding method.

3.1.1 Basic signature scheme

The RSA algorithm is a trapdoor one-way function whose security requires large and hard-to-factor numbers. An RSA key pair is constructed by first selecting a public exponent e larger than 2^{16}, odd and preferably prime. Then, two large primes p and q, such that p-1 and q-1 are coprime to e, are selected. The product N=pq is then computed.

The public key of the algorithm is the pair (N,e).

The private key of the algorithm is the pair (N,d) where d is the smallest representative of the inverse of e modulo (p-1)(q-1).

The signature algorithm takes a uniformly selected random element m from the interval [1,N-1] and outputs m^d modulo N.

The verification algorithm takes an element m and a signature s, it outputs Valid when $m=s^e$ modulo N and Invalid otherwise.

NOTE: The modulus N is said to be an n-bit number if and only if, $2^{n-1} \leq N < 2^n$. To construct an n-bit RSA key, each of the two primes p and q should be selected in the interval $[2^{(n-1)/2};2^{n/2}[$. Ideally, one should select p and q uniformly at random in the interval. However, since this distribution is not so easy to sample efficiently, it is considered acceptable practice to select for each the first prime following a randomly selected element of the interval, i.e., p=Nextprime(u) and q=Nextprime(v).

It is extremely important to start for two uncorrelated values u and v. Otherwise, the process is flawed and N can often be factored.

TYPICAL ERROR: One typical mistake is to let p=Nextprime(u) and q=Nextprime(p+1), then N is trivially factored. Similarly, if p and q are too close from each other factoring N is also easy. For this reason, it is sometimes recommended to check that the distance between p and q is not too small. However, the probability that this happens when the primes are correctly generated is well below the probability of 2^{-80} that is accepted for non-adversarial errors. As a consequence, there is no need to implement this test. Indeed, it only protects against one of the many implementation errors that are possible.

RSA key-size:

- For short-term security, corresponding to the **100-bit** security level, the RSA key should have at least **2048 bits.**
- For longer-term security, corresponding to the **128-bit** security level, the RSA key should have at least **3072 bits.**
- The 256-bit security level required to protect against quantum computers cannot be achieved with RSA, due to the quantum polynomial time factoring algorithm of Shor.

3.1.2 Padding methods

As explained above, the basic RSA signature scheme cannot be used directly to sign arbitrary messages. The main reason is that, due to the multiplicativity of RSA, it would not be secure. Indeed, with the basic scheme, the signature of a product of two numbers is just the product of the two signatures.

To avoid this attack, messages to be signed should first be transformed into numbers modulo N in a way that prevents an attack for being able to efficiently construct multiplicative relations between these numbers. This transformation is called a padding method.

3.1.2.1 Full domain hash

The simplest method to prepare a message M for RSA signature is simply to hash it, i.e. to compute m=H(M) and then apply the basic RSA signature to m.

However, if H is a usual hash function with short output, this is not secure. For security, one needs to use a hash function with a large output. This is the full domain hash.

In theoretical papers, one simply assumes that the hash function H outputs digests which are integers in [1;N-1]. For practical purposes, three variations are possible:

- Choose H_0 that outputs bitstrings of the length of N, and define $H(M)=H_0(\text{mincount}||H_0(M))$, where mincount is the smallest possible counter value such that H(M) belongs to [1;N-1]. This corresponds to the theoretical model but does not permit constant time implementation.

- Choose H that outputs bitstrings shorter than N by one bit. The distribution is no longer uniform. The full domain hash is not proven in this case but no attack is known.
- Choose H_0 that outputs bitstrings longer than N by a margin of the order of the security level and let $H(M)=H_0(N||M)$ modulo N. The output of H becomes close to uniform. One might also consider the simpler variation $H(M)=H_0(M)$ modulo N. However, potentially, the reduction modulo N could cause collision search to become easier, even if it is unclear how this would apply in the context of RSA. Adding N into the hash computation prevents attacks based on Vaudenay's DSA attack.

In the three variations, the hash function with a long output can be derived from an extendable hash function, preferably using a message header to ensure domain separation.

NOTE: In order for the security proof of full domain hash RSA to be tight, one need to make sure than the public RSA exponent e is smaller than $N^{1/4}$.

3.1.2.2 PSS encoding

By contrast with the full domain hash method, this padding technique is probabilistic. As a consequence, if the same message is signed twice, it will lead to different signatures. Depending on the application, this can be either an advantage or a drawback.

In addition to the hash function H (with fixed size), PSS encoding also requires a mask generating function G. Note that G is, in fact, a variable output length function adapted to N.

PSS encoding:

- **Input:** message M, salt size sLen, desired output length nBits
- **Output:** encoding to be signed by the basic RSA scheme

- Let m=H(M) of size hLen
- If nBits < hLen + sLen + 2 then Abort
- Generate random string *salt* of size sLen
- Let M'= 0||0||0||0||0||0||0||0||m||salt (concatenation of 8 zero bytes, m and salt)
- Let h=H(M') of size hLen
- Let DB = 0||…||0||1||salt, with the number of zeroes chosen to make the length of DB equal to nLen-hLen-1, where nLen is nBits/8 rounded up.
- Compute dbmask=G(h) with variable output size set to nBits-hLen-1
- Let maskedDB= DB XOR dbmask and clear the 8nLen-nBits higher order bits of maskedDB.
- Let encoding= maskedDB||h||0xbc

The signature itself is simply the encoding raised to the private exponent modulo N.

During verification, the signature is raised to the public exponent and the output is considered as an encoding to be verified.

PSS encoding verification:

- **Input:** message M, salt size sLen, desired output length nBits
- **Output:** encoding to be verified

- Let m=H(M) of size hLen
- If nBits < hLen + sLen + 2 then Abort
- Parse encoding into maskedDB || h || Oxbc. Abort if failure
- Check that the 8nLen-nBits higher order bits of maskedDB are zeroes
- Compute dbmask=G(h) with variable output size set to nBits-hLen-1

- Let DB= maskedDB XOR dbmask and clear the 8nLen-nBits higher order bits of DB.
- Parse DB as 0||…||0||1||salt, abort if it fails.
- Let M’= 0||0||0||0||0||0||0||0||m||salt (concatenation of 8 zero bytes, m and salt)
- Let h’=H(M’) of size hLen
- If h=h’ accept signature

3.2 Discrete logarithm-based signatures

3.2.1 Possible discrete logarithm groups

3.2.1.1 Finite fields

Historically, the first groups considered for discrete logarithm based encryption and signature where the multiplication groups of integers modulo a prime p, GF(p).

Other finite fields, in particular GF(2^m), were also considered. However, the fields GF(2^m) have recently been shown to be insecure for cryptographic purposes.

In addition to GF(p), other fields GF(p^k) for intermediate values of p and k could be considered. However, these are not standardized and rarely used in applications.

As a consequence, in this document, prime fields GF(p) are considered as potential candidates for discrete logarithm based signatures.

For these fields, the key sizes should be chosen with the same rules as for RSA, namely:

- For short-term security, corresponding to the **100-bit** security level, the prime p should have at least **2048 bits.**
- For longer-term security, corresponding to the **128-bit** security level, the prime p should have at least **3072 bits.**
- The 256-bit security level required to protect against quantum computers cannot be achieved with discrete logarithms, due to the quantum polynomial time discrete logarithm algorithm of Shor.

NOTE: Whenever possible, elliptic curve discrete logarithm signatures should preferred to finite field discrete logarithm signatures.

3.2.1.2 Elliptic curves over finite fields

Another possibility is to use the group of points of an elliptic curve defined over a finite field. Here, only fields of the form GF(p) or GF(2^m) are considered. The main advantage of elliptic curves is that they permit to choose much smaller key sizes, which very positively impacts their performance.

For elliptic curves, the general rules for key sizes are :

- For short-term security, corresponding to the **100-bit** security level, the number of points on the curve should have at least **200 bits.**
- For longer-term security, corresponding to the **128-bit** security level, the number of points on the curve should have at least **256 bits.**
- The 256-bit security level required to protect against quantum computers cannot be achieved with discrete logarithms, even on elliptic curves, due to the quantum polynomial time discrete logarithm algorithm of Shor.

3.2.2 Signing Equations

Once a group is chosen, the way signatures can be computed and verified needs to be described. Since it's possible to choose groups that are either written multiplicatively (finite fields) or additively (elliptic curves), the equations would, in theory, have to be written twice. For compactness, since elliptic curves are the preferred choice, only the additive description is given.

Global Parameters:

Description of the group, order q and generator G. (By construction qG=0)

Hash function H, with consistent security level.

Key pairs:

Secret key integer d in [1;q-1]. Public Key Q=dG.

3.2.2.1 DSA style

The ECDSA signature on a string M works as follows:

1. Select a random integer k in [1; q-1]
2. Compute (x_1, y_1)=kG, lift x_1 to its smallest positive representative integer.
3. Compute r=x_1 mod q, if r=0 restart
4. Compute k^{-1} mod q
5. Let e=H(M), where e is an integer obtained by converting the bitstring H(M)
6. Let s= k^{-1}(e+dr) mod q, if s=0 restart
7. Output the signature (r,s)

The verification of a ECDSA signature works as follows given M, r and s as input:

1. Check that both r and s are in [1; q-1]
2. Let e=H(M)
3. Let w=s^{-1} mod q
4. Compute u_1=ew mod q and u_2=rw mod q
5. Let X=u_1 G+u_2 Q
6. If X=0, reject signature, otherwise write X=(x_1,y_1)
7. Compute x_1 mod q, accept signature if r= x_1 mod q, reject otherwise

For plain DSA, the adaptation to additive notation is done in the straightforward way. There is an additional difference, instead of letting x_1 denote the point abscissa, it represents the group element itself as an integer.

There are variations of ECDSA with similar properties, such as EC-GDSA, EC-KCDSA, …

NOTE: There exists an attack by Vaudenay against DSA signature that derives from the fact that collisions on H(M) mod q are enough to forge signatures and are much easier to produce if the adversary has full control of the parameter generation process. This attack can be mitigated by a careful validation of the public parameters.

3.2.2.2 Schnorr style

The Schnorr signature on a string M works as follows:

1. Select a random integer k in [1; q]
2. Let R=kG
3. Let e=H(M || R), where M||R indicates that a string representing the point R is appended in a non-ambiguous representation to the message M.
4. Let s= k+de mod q, if s=0 restart
5. Output the signature (R, s)

The verification of a Schnorr signature works as follows given M, R and s as input:

1. Check that R is a point on the curve
2. Let e=H(M || R)
3. If R+eQ=sG accept signature else reject

Thanks to the inclusion of the point R into the hash function computation, Schnorr signatures are more robust and easier to prove than their DSA counterparts. In particular, they are not vulnerable to Vaudenay's attack.

Wherever possible, Schnorr signatures should be preferred to DSA signatures.

4 Generating keys and parameters

4.1 RSA

RSA key pair is constructed by first selecting a public exponent e larger than 2^{16}, odd and preferably prime. Then, two large primes p and q, such that p-1 and q-1 are coprime to e, are selected. The product N=pq is then computed. Moreover, for tightness of the full domain hash, it is recommended that $e<N^{1/4}$.

Typically, one chooses e=65537.

NOTE: For signature schemes, the restriction $e>2^{16}$ is not essential. However, since the same key generators are used both for encryption and signature scheme, it is preferable to keep this restriction here.

Recommended method for generating the primes p and q:

- Call the randomness generator to produce a seed S of length at least equal to the desired security level in bits.
- Let counter=0 be a fixed length counter (4 bytes are enough)
- Let n be the desired bit size of N (for simplicity assume that n is even)
- Repeat:
    - p_0=G(counter||S) where G is a variable output function on n/2 bits
    - If $p_0 < 2^{(n-1)/2}$, increment counter and loop
    - p=Nextprime(p_0)
    - If $p>2^{n/2}$, increment counter and loop
    - If e divides (p-1), increment counter and loop [if e is not a prime, replace this test by gcd(e,p-1) different from 1]
- Increment counter
- Repeat:
    - q_0=G(counter||S)
    - If $q_0 < 2^{(n-1)/2}$, increment counter and loop
    - q=Nextprime(q_0)
    - If $q>2^{n/2}$, increment counter and loop
    - If e divides (q-1), increment counter and loop [if e is not a prime, replace this test by gcd(e,q-1) different from 1]
- Let N=pq and $d = e^{-1}$ mod (p-1)(q-1)

The goal of tying the generation of p and q together (from a single seed S) is to prevent factoring attacks on keys that share a common factor.

4.2 Elliptic Curves

For elliptic curve based signatures, the choice of an elliptic curve is usually a system wide parameter. As a consequence, to avoid any suspicion that the curve has been chosen in a adversarial manner by the managing authority, it is important to give the elliptic curve together with a certificate that allow users to replicate the parameter generation.

For using elliptic curve over prime fields GF(p), suggested parameter generation algorithm is presented for this case.

Recommended method for generating GF(p) and elliptic curve E:

- Input : seed S of length at least equal to the desired security level in bits.
- Let counter=0 be a fixed length counter (4 bytes are enough)
- Let n be the desired bit size of p
- Repeat:
    - p_0=G(counter||S) where G is a variable output function on n bits
    - If $p_0 < 2^{(n-1)}$, increment counter and loop
    - p=Nextprime(p_0)
    - If $p>2^n$, increment counter and loop
- Increment counter
- Repeat:
    - a=G(counter||S) mod p, increment counter, b= G(counter||S) mod p
    - Count the number of points q on the elliptic curve y^2=x^3+a x+b (mod p)
    - Test whether q is prime, if not increment counter and loop
- Increment counter
- Repeat:
    - x_0=G(counter||S) mod p
    - if x_0^3+a x_0+b is a square mod p, let y_0 be the square root in the interval [1,(p-1)/2]
- Output: S (to permit verification), p, q and the basepoint (x_0,y_0)

NOTE: This generation method protects against selecting weak curves from a very small subset of all elliptic curves. It does not protect against an adversarial parameter generation authority that knows an attack that works against a small but not negligible fraction of curves. For example, if the authority knows how to attack 1 curve in 10^9, it chooses many values of S until such a curve is obtained.

To protect against this, it is possible to determine the seed S in a way that cannot be manipulated by a single dishonest authority. For example, shared generation of S between several authorities can be used. Alternatively, one may announced in advance the process that will be followed to generate S and make sure that process is auditable and hard to tamper with. Typically, taking the hash of all stock values in some fixed market and format at a future date is a possible (but cumbersome) option.

Annex 2 – References

ENISA papers

[1] Algorithms, key size and parameters report 2014, https://www.enisa.europa.eu/activities/identity-and-trust/library/deliverables/algorithms-key-size-and-parameters-report-2014

Standards

[2] ETSI TS 319 112: "Electronic Signatures and Infrastructures (ESI); Cryptographic Suites"

[3] ETSI TS 101 861: "Electronic Signatures and Infrastructures (ESI); Time stamping profile", Version 1.4.1 (2011-07).

[4] ETSI TR 119 300: "Electronic Signatures and Infrastructures (ESI); Business Guidance on Cryptographic Suites".

[5] ETSI TR 119 100: "Electronic Signatures and Infrastructures (ESI); Business Guidance on Signature Creation and Validation".

[6] ETSI TR 119 000: "Electronic Signatures and Infrastructures (ESI); Rationalised structure for Electronic Signature Standardisation".

[7] ETSI TS 101 733: "Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAdES)".

[8] ETSI TS 101 903: "Electronic Signatures and Infrastructures (ESI); XML Advanced Electronic Signatures (XAdES)".

[9] ETSI TS 102 778: "Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signatures (PAdES)".

[10] ETSI TS 102 176-1: Electronic Signatures and Infrastructures (ESI); Algorithms and Parameters for Secure Electronic Signatures; Part 1: Hash functions and asymmetric algorithms; v2.1.1 (2011-07).

[11] ISO/IEC 14888-3 (2006) "Information technology - Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms".

[12] ISO/IEC 14888-3:2006/Amd.1 (2010) "Amendment 1: Elliptic Curve Russian Digital Signature Algorithm, Schnorr Digital Signature Algorithm, Elliptic Curve Schnorr Digital Signature Algorithm, and Elliptic Curve Full Schnorr Digital Signature Algorithm"

[13] ISO/IEC 18031 (2011): "Information technology - Security techniques - Random bit generation".

[14] ISO/IEC 18032 (2005): "Information technology - Security techniques - Prime number generation".

[15] ISO/IEC 10118-3 (2004): "Information technology - Security techniques - Hash functions - Part 3: Dedicated hash functions".

[16] ANSI X9.62 (2005): "Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)".

[17] ANSI X9.82 (2006): "Random Number Generation Parts 1".

[18] ANSI X9.17: "Pseudo Random Number Generator (RNG)".

[19] IETF RFC 3447 (2003): "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1".

[20] IETF RFC 5639 (2010): "Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation".

[21] IETF RFC 3279 (2002): "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile" and updates by RFC 4055, RFC 4491, RFC 5480, and RFC 5758.

[22] IETF RFC 4055 (2005): "Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet X.509 Public Key Infrastructure - Certificate and Certificate Revocation List (CRL) Profile".

[23] IETF RFC 5753: "Use of Elliptic Curve Cryptography (ECC) Algorithms in Cryptographic Message Syntax (CMS)".

[24] IETF RFC 6931 (2013): "Additional XML Security Uniform Resource Identifiers (URIs)".

[25] IETF RFC 3161 (2001): "Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP)", updated by RFC 5816.

[26] IETF RFC 6960 (2013): "X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP", updated by RFC 2560, RFC 6277.

[27] IETF RFC 5280 (2008): "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile".

[28] FIPS Publication 180-4 (2012): "Secure Hash Standard (SHS)".

[29] FIPS Publication 186-4 (July 2013): "Digital Signature Standard (DSS)".

[30] Draft FIPS Publication 202 (2014): “SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions”. Available at http://csrc.nist.gov/publications/PubsDrafts.html#FIPS-202National Institute of Standards and Technology SHA-3 Competition (2007-2012)

[31] AIS 20/31: "Application Notes and Interpretation of the Scheme: Functionality classes and evaluation methodology for deterministic random number generators", Version 2.

[32] W3C Recommendation: "XML Encryption Syntax and Processing Version 1.1", Apr 2013

[33] W3C Recommendation: "XML-Signature Syntax and Processing Version 1.1", Apr 2013

Reference books, good practices

[34] Agence nationale de la sécurité des systèmes d’information, Référentiel Général de Sécurité version 1.0, 2010-01. Available at http://www.ssi.gouv.fr/IMG/pdf/RGS_B_1.pdf

[35] Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen, Übersicht über geeignete Algorithmen, 2014-01. Available at http://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/QES/Veroeffentlichungen/Algorithmen/2014Algorithmenkatalog.pdf

[36] Bundesamt für Sicherheit in der Informationstechnik, TR-02102-1 Kryptographische Verfahren: Empfehlungen und Schlüssellängen, 2014-01. Available at https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR02102/BSI-TR-02102_pdf.html

[37] European Network of Excellence in Cryptology, ECRYPT II Yearly Report on Algorithms and Keysizes (2011-2012), 2012-09. Available at http://www.ecrypt.eu.org/documents/D.SPA.20.pdf

[38] NIST Special Publication SP 800-90A Recommendation for Random Number Generation Using Deterministic Random Bit Generators, Jan 2012.

Research papers

[39] Mihir Bellare and Phillip Rogaway, The Exact Security of Digital Signatures-How to Sign with RSA and Rabin. Advances in Cryptology — EUROCRYPT '96. Lecture Notes in Computer Science 1070. Pages 399-416. Springer

[40] Jean-Sébastien Coron, Optimal Security Proofs for PSS and Other Signature Schemes. Advances in Cryptology — EUROCRYPT 2002. Lecture Notes in Computer Science 2332. Pages 272-287. Springer

[41] Saqib Kakvi, Eike Kiltz, Optimal Security Proofs for Full Domain Hash, Revisited, Advances in Cryptology — EUROCRYPT 2012. Lecture Notes in Computer Science 7237. Pages 537-553. Springer

[42] Nadia Heninger, Zarik Durumeric, Eric Wustrow, J. Alex HAlderman, Mining your Ps and Qs: Detection of widespread weak keys in network devices. Proc. 21st USENIX Security Symposium.

[43] Andrew V. Sutherland, Computing Hilbert class polynomials with the Chinese remainder theorem. Mathematics of Computation 80 (2011), Pages 501-538.

[44] Serge Vaudenay, The security of DSA and ECDSA. Advances in Cryptology — PKC 2003. Lecture Notes in Computer Science 2567. Pages 309-323. Springer

ENISA
European Union Agency for Network and Information Security
Science and Technology Park of Crete (ITE)
Vassilika Vouton, 700 13, Heraklion, Greece

Athens Office
1 Vass. Sofias & Meg. Alexandrou
Marousi 151 24, Athens, Greece

PO Box 1309, 710 01 Heraklion, Greece
Tel: +30 28 14 40 9710
info@enisa.europa.eu
www.enisa.europa.eu