

Australian Standard™

**Information technology—Security
techniques—Entity authentication**

**Part 5: Mechanisms using zero-
knowledge techniques**



This Australian Standard was prepared by Committee IT-012, Information Systems, Security and Identification. It was approved on behalf of the Council of Standards Australia on 31 March 2006.
This Standard was published on 1 May 2006.

The following are represented on Committee IT-012:

Attorney General's Department
Australia Post
Australian Association of Permanent Building Societies
Australian Bankers Association
Australian Chamber of Commerce and Industry
Australian Electrical and Electronic Manufacturers Association
Australian Information Industry Association
Certification Forum of Australia
Consumers' Federation of Australia
Department of Defence (Australia)
Department of Social Welfare New Zealand
Government Communications Security Bureau, New Zealand
Internet Industry Association
NSW Police Service
New Zealand Defence Force
Reserve Bank of Australia

Keeping Standards up-to-date

Standards are living documents which reflect progress in science, technology and systems. To maintain their currency, all Standards are periodically reviewed, and new editions are published. Between editions, amendments may be issued. Standards may also be withdrawn. It is important that readers assure themselves they are using a current Standard, which should include any amendments which may have been published since the Standard was purchased.

Detailed information about Standards can be found by visiting the Standards Web Shop at www.standards.com.au and looking up the relevant Standard in the on-line catalogue.

Alternatively, the printed Catalogue provides information current at 1 January each year, and the monthly magazine, *The Global Standard*, has a full listing of revisions and amendments published each month.

Australian Standards™ and other products and services developed by Standards Australia are published and distributed under contract by SAI Global, which operates the Standards Web Shop.

We also welcome suggestions for improvement in our Standards, and especially encourage readers to notify us immediately of any apparent inaccuracies or ambiguities. Contact us via email at mail@standards.org.au, or write to the Chief Executive, Standards Australia, GPO Box 476, Sydney, NSW 2001.

This Standard was issued in draft form for comment as DR 06034.

Australian Standard™

**Information technology—Security
techniques—Entity authentication**

**Part 5: Mechanisms using zero-
knowledge techniques**

First published as AS ISO/IEC 9798.5—2006.

COPYRIGHT

© Standards Australia

All rights are reserved. No part of this work may be reproduced or copied in any form or by any means, electronic or mechanical, including photocopying, without the written permission of the publisher.

Published by Standards Australia GPO Box 476, Sydney, NSW 2001, Australia

ISBN 0 7337 7386 9

PREFACE

This Standard was prepared by the Joint Standards Australia/Standards New Zealand Committee IT-012, Information Systems, Security and Identification.

After consultation with stakeholders in both countries, Standards Australia and Standards New Zealand decided to develop this Standard as an Australian Standard rather than an Australian/New Zealand Standard.

This Standard is identical with, and has been reproduced from ISO/IEC 9798-5:2004, *Information technology—Security techniques—Entity authentication—Part 5: Mechanisms using zero-knowledge techniques*.

The objective of this Standard is to provide the information security management community with detailed guidance on the background, techniques and procedures of entity authentication mechanisms using zero-knowledge techniques.

This Standard is Part 5 of AS 9798, *Information technology—Security techniques—Entity authentication*, which, when complete, will consist of the following:

Part 1: General

Part 2: Mechanisms using symmetric encipherment algorithms

Part 3: Mechanisms using digital signature techniques

Part 4: Mechanisms using a cryptographic check function

Part 5: Mechanisms using zero-knowledge techniques (this Standard)

Part 6: Mechanisms based on manual data transfer

As this Standard is reproduced from an international standard, the following applies:

- (a) Its number appears on the cover and title page while the international standard number appears only on the cover.
- (b) In the source text ‘this International Standard’ should read ‘this Australian Standard’.
- (c) A full point substitutes for a comma when referring to a decimal marker.

The terms ‘normative’ and ‘informative’ are used to define the application of the annex to which they apply. A normative annex is an integral part of a standard, whereas an informative annex is only for information and guidance.

CONTENTS

	<i>Page</i>
1 Scope	1
2 Normative references	1
3 Terms and definitions	2
4 Symbols and abbreviated terms	4
5 Mechanisms based on identities	7
6 Mechanisms based on integer factorization	12
7 Mechanisms based on discrete logarithms with respect to prime numbers	15
8 Mechanisms based on discrete logarithms with respect to composite numbers	17
9 Mechanisms based on asymmetric encipherment systems	20
Annex A (normative) Object identifiers	23
Annex B (informative) Principles of zero-knowledge techniques	25
Annex C (informative) Guidance on parameter choice and comparison of the mechanisms	28
Annex D (informative) Numerical examples	38
Bibliography	49

NOTES

AUSTRALIAN STANDARD

Information technology — Security techniques — Entity authentication —**Part 5:
Mechanisms using zero-knowledge techniques****1 Scope**

This part of ISO/IEC 9798 specifies entity authentication mechanisms using zero-knowledge techniques.

- Clause 5 specifies mechanisms (already present in the first edition, ISO/IEC 9798-4:1999) based on identities and providing unilateral authentication. They have been repaired after the withdrawal of ISO/IEC 9796:1991.
- Clause 6 specifies mechanisms (inserted in this second edition) based on integer factorization and providing unilateral authentication.
- Clauses 7 and 8 specify mechanisms based on discrete logarithms with respect to numbers that are either prime (see Clause 7, mechanisms already present in the first edition) or composite (see Clause 8, mechanisms inserted in the second edition), and providing unilateral authentication.
- Clause 9 specifies mechanisms based on asymmetric encipherment systems and providing either unilateral (see 9.3, mechanisms already present in the first edition), or mutual (see 9.4, mechanisms inserted in the second edition) authentication.

The verifier associates the correct verification key with the claimant by any appropriate procedure, for example, by retrieving it from a certificate. Such procedures are outside the scope of this part of ISO/IEC 9798.

To identify each mechanism, Annex A specifies object identifiers in accordance with ISO/IEC 8825-1.

These mechanisms are constructed using the principles of zero-knowledge techniques, but they will not be zero-knowledge according to the strict definition sketched in Annex B for every choice of parameters.

Annex C compares the mechanisms and provides guidance on parameter choices.

Annex D provides numerical examples.

2 Normative references

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

ISO/IEC 8825-1:2002, *Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*

ISO/IEC 10118 (all parts), *Information technology — Security techniques — Hash-functions*

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1

accreditation exponent

secret number related to the verification exponent and used in the production of private numbers

3.2

adaptation parameter

public number specific to the modulus and used in the definition of public numbers in the GQ2 mechanisms

3.3

asymmetric cryptographic technique

cryptographic technique that uses two related operations: a public operation defined by a public data item, key or number, and a private operation defined by a private data item, key or number (the two operations have the property that, given the public operation, it is computationally infeasible to derive the private operation)

3.4

asymmetric encipherment system

system based on asymmetric cryptographic techniques whose public operation is used for encipherment and whose private operation is used for decipherment

3.5

asymmetric pair

two related data items, keys or numbers, where the private data item defines a private operation and the public data item defines a public operation

3.6

challenge

procedure parameter used in conjunction with secret parameters to produce a response

3.7

claimant

entity whose identity can be authenticated, including the functions and the private data necessary to engage in authentication exchanges on behalf of a principal

3.8

claimant parameter

public data item, number or bit string, specific to a given claimant within the domain

3.9

decipherment

reversal of a corresponding encipherment
[ISO/IEC 9798-1]

3.10

domain

collection of entities operating under a single security policy, e.g., public key certificates created by a single certification authority, or by a collection of certification authorities using the same security policy

3.11

domain parameter

public number, or function, agreed and used by all entities within the domain

3.12

encipherment

reversible operation by a cryptographic algorithm converting data into ciphertext, so as to hide the information content of the data

3.13**entity authentication**

corroboration that an entity is the one claimed
[ISO/IEC 9798-1]

3.14**exchange multiplicity parameter**

number of exchanges of information involved in one instance of an authentication mechanism

3.15**hash-function**

function that maps strings of bits to fixed-length strings of bits, satisfying the following two properties:

- for a given output, it is computationally infeasible to find an input that maps to this output;
- it is computationally infeasible to find two distinct inputs that map to the same output

[ISO/IEC 10118-1]

3.16**identification data**

set of public data items (e.g., an account number, an expiry date and time, a serial number, etc.) assigned to an entity and used to identify it

3.17**mutual authentication**

entity authentication that provides both entities with assurance of each other's identity
[ISO/IEC 9798-1]

3.18**number**

natural integer, i.e., a non-negative integer

3.19**pair multiplicity parameter**

number of asymmetric pairs of numbers involved in one instance of an authentication mechanism

3.20**private key or private number**

that data item, key or number, of an asymmetric pair, that shall be kept secret and should only be used by a claimant in accordance with an appropriate response formula, thereby establishing its identity

3.21**procedure parameter**

public data item involved with a transient value in one instance of an authentication mechanism, e.g., witness, challenge, response

3.22**public key or public number**

that data item, key or number, of an asymmetric pair, that can be made public and shall be used by every verifier for establishing the claimant's identity

3.23**random number**

time variant parameter whose value is unpredictable
[ISO/IEC 9798-1]

3.24**response**

procedure parameter produced by the claimant, and processed by the verifier for checking the identity of the claimant

3.25**secret parameter**

number or bit string that does not appear in the public domain, only used by a claimant, e.g., a private number

3.26**token**

message consisting of data fields relevant to a particular communication and which contains information that has been produced using a cryptographic technique

3.27**unilateral authentication**

entity authentication that provides one entity with assurance of the other's identity but not vice versa [ISO/IEC 9798-1]

3.28**verification exponent**

public number used as exponent by the claimant and the verifier

3.29**verifier**

entity including the functions necessary for engaging in authentication exchanges on behalf of an entity requiring an entity authentication

3.30**witness**

procedure parameter that provides evidence of the claimant's identity to the verifier

4 Symbols and abbreviated terms

For the purposes of this document, the following symbols and abbreviated terms apply.

$(a \mid n)$ Jacobi symbol of a positive integer a with respect to an odd composite integer n

NOTE By definition, the Jacobi symbol of any positive integer a with respect to any odd positive composite integer n is the product of the Legendre symbols of a with respect to each prime factor of n (repeating the Legendre symbols for the repeated prime factors). The Jacobi symbol ^[10, 13] can be efficiently computed without knowledge of the prime factors of n .

$(a \mid p)$ Legendre symbol of a positive integer a with respect to an odd prime integer p

NOTE By definition, the Legendre symbol of any positive integer a with respect to any odd positive prime integer p is set equal to $a^{(p-1)/2} \bmod p$. This means that $(a \mid p)$ is zero if a is a multiple of p , and either +1 or -1 otherwise, depending on whether or not a is a square modulo p .

$|A|$ bit size of the number A if A is a number (i.e., the unique integer i so that $2^{i-1} \leq A < 2^i$ if $A > 0$, or 0 if $A = 0$, e.g., $|65\,537| = |2^{16} + 1| = 17$), or bit length of the bit string A if A is a bit string

NOTE The binary representation of a number A as a string of $|A|$ bits is straightforward. For representing a number A as a string of α bits with $\alpha > |A|$, $\alpha - |A|$ bits set to 0 are appended on the left of the $|A|$ bits.

$\lfloor A \rfloor$ the greatest integer that is less than or equal to the real number A

$B \parallel C$ bit string resulting from concatenating the two bit strings B and C in that order

CRT Chinese Remainder Theorem

d challenge (procedure parameter)

D response (procedure parameter)

f	number of prime factors
$\gcd(a, b)$	the greatest common divisor of the two integers a and b
G, G_i	public number (domain parameter)
$G(A), G_i(A)$	public number (claimant parameter)
h	hash-function
$ h $	bit length of the hash-code produced by the hash-function h
H, HH	hash-codes
$Id(A)$	identification data (claimant parameter)
$Id_i(A)$	part of the identification data (claimant parameter)
$j \bmod n$	the unique integer i from $\{0, 1, \dots, n-1\}$ so that n divides $j - i$
$j \bmod^* n$	the unique integer i from $\{0, 1, \dots, (n-1)/2\}$ so that n divides either $j - i$ or $j + i$
$\text{lcm}(a, b)$	the least common multiple of the two integers a and b
m	pair multiplicity parameter (domain parameter)
n	composite modulus (domain parameter)
$n(A)$	composite modulus (claimant parameter)
$p_1, p_2 \dots$	prime factors of the modulus in ascending order, i.e., $p_1 < p_2 < \dots$ (secret parameters)
Q, Q_i	private number (secret parameter)
r	fresh random number or fresh string of random bits (secret parameter)
v	verification exponent (domain parameter)
W	witness (procedure parameter)
'XY'	notation using the hexadecimal digits '0' to '9' and 'A' to 'F', equal to XY to the base 16
α	modulus size in bits, i.e., $2^{\alpha-1} \leq \text{modulus} < 2^\alpha$, also denoted $ \text{modulus} $ (domain parameter)
δ	length of fresh strings of random bits for representing challenges (domain parameter)
ρ	length of fresh strings of random bits for representing random numbers (domain parameter)
$\{3, 5, 6\}$	set of the integers 3, 5 and 6

For the purposes of clause 5 (identity-based mechanisms), the following symbols and abbreviated terms apply.

F	bit string
t	exchange multiplicity parameter (domain parameter)
u	accreditation exponent with respect to the modulus (secret parameter)

u_j accreditation exponent with respect to the prime factor p_j (secret parameter)

For the purposes of clause 6 (integer factorization based mechanisms), the following symbols and abbreviated terms apply.

b adaptation parameter (specific to the modulus)

D_j response component with respect to the prime factor p_j (secret parameter)

g_i basic number (domain parameter)

$g(A)$ basic number (claimant parameter)

k security parameter (domain parameter)

$Q_{i,j}$ private component with respect to the basic number g_i and the prime factor p_j (secret parameter)

r_j fresh random number with respect to the prime factor p_j (secret parameter)

u_j accreditation exponent with respect to the prime factor p_j (secret parameter)

W_j witness component with respect to the prime factor p_j (secret parameter)

For the purposes of clause 7 (mechanisms based on discrete logarithms with respect to prime numbers), the following symbols and abbreviated terms apply.

g base of the discrete logarithms (domain parameter)

p modulus (domain parameter)

q prime number (domain parameter)

For the purposes of clause 8 (mechanisms based on discrete logarithms with respect to composite numbers), the following symbols and abbreviated terms apply.

g base of the discrete logarithms (domain parameter)

$g(A)$ base of the discrete logarithms (claimant parameter)

σ number of bits for private numbers in the first mode (domain parameter)

For the purposes of clause 9 (mechanisms based on asymmetric encipherment systems), the following symbols and abbreviated terms apply.

P_A public operation, i.e., encipherment (claimant parameter)

S_A private operation, i.e., decipherment (secret parameter)

x private RSA exponent (secret parameter)

5 Mechanisms based on identities

5.1 Security requirements for the environment

These mechanisms enable a verifier to check that a claimant knows private number(s) that are related to identification data by a verification key.

NOTE These mechanisms implement schemes due either to Fiat and Shamir ^[4] and denoted FS, or to Guillou and Quisquater ^[8] and denoted GQ1.

Within a given domain, the following requirements shall be satisfied.

- 1) Domain parameters shall be selected, which will govern the operation of the mechanism. They include a hash-function, e.g., one of the functions specified in ISO/IEC 10118-3. The selected parameters shall be made known in a reliable manner to all entities within the domain.
- 2) Every claimant shall be equipped with a modulus that is either a domain parameter or a claimant parameter. Each number used as modulus is set equal to the product of two or more distinct prime factors so that knowledge of its value shall not feasibly enable any entity to deduce its prime factors, where feasibility is defined by the context of use of the mechanism.

— If the modulus is a domain parameter, then it is denoted n . A trusted authority has selected it and only this authority can use the corresponding prime factors. The authority guarantees the identities of every claimant within the domain.

NOTE For example, a card issuer has a modulus. A delegated entity signs identification data for issuing smart cards; it uses the issuer's prime factors. In each card, the delegated entity stores appropriate identification data and private number(s). During its life, the card uses its private number(s) in accordance with an identity-based mechanism.

— If the modulus is a claimant parameter, then it is denoted $n(A)$. A principal has selected it and the corresponding prime factors are the principal's long-term secret. For each session, the principal creates a claimant. The claimant uses private number(s) as a short-term secret.

NOTE For example, in a local area network, an authority supervises each login operation within the domain and manages a directory where every verifier can obtain a trusted copy of a modulus for each principal.

— During each login operation, i.e., when a computer opens a session, it uses a principal's prime factors for a "single-sign-on" of session identification data including identifier, expiry date and time, rights, etc.

— During the session, the computer cannot use the prime factors because it does not know them any more. It uses the private number(s) in accordance with an identity-based mechanism. The private numbers only last for a few hours: their utility disappears after the session.

- 3) Every claimant shall be provided with identification data and with one or more private numbers by some means. In this context, the identification data is a string of bits, nor all equal, that uniquely and meaningfully identifies the claimant in accordance with an agreed convention.

NOTE The presence of an expiry date and time in the identification data enforces their expiry; the presence of a serial number simplifies their revocation.

- 4) Every verifier shall obtain a trusted copy of the correct modulus of the claimant.

NOTE The exact means by which the verifier obtains a trusted copy of the correct modulus is beyond the scope of this document. This may, for example, be achieved by the use of public-key certificates or by some other environment-dependent means.

- 5) Every claimant and every verifier shall have the means to produce random numbers.

5.2 Key production

5.2.1 Asymmetric key pair

A verification exponent, a pair multiplicity parameter and an exchange multiplicity parameter shall be selected. Unless otherwise specified, they are domain parameters respectively denoted v , m and t .

— Certain values of v , such as the prime numbers 2, 257, $2^{16}+1$, $2^{32}+15$, $2^{36}+2^{13}+1$ and $2^{40}+15$, have some practical advantages.

- The value of m shall be at most eight if $v = 2$ and set equal to one if v is an odd prime.
- The value of $v^{-m \times t}$ fixes a mechanism security level (see C.1.4). A value from 2^{-8} to 2^{-40} is appropriate for most applications.

A number, denoted α , fixes the modulus size in bits, i.e., $2^{\alpha-1} < \text{modulus} < 2^\alpha$, in accordance with the context of use of the mechanism (for further details, see C.1.1). It is a domain parameter.

The authority or the principal shall keep secret two or more distinct large prime factors denoted $p_1, p_2 \dots$ in ascending order, the product of which is the modulus.

- If $v = 2$ (the Rabin scheme), there shall be only two prime factors (i.e., $f = 2$), both congruent to 3 mod 4, but not congruent to each other mod 8.
- If v is an odd prime (the RSA scheme), there may be more than two prime factors. For each prime factor p_j , $p_j - 1$ shall be co-prime to v .

If α is a multiple of the number of prime factors, denoted f , then the bit size of each prime factor shall be α / f (for further details, see C.1.2). The modulus is set equal to either $p_1 \times p_2$ if $v = 2$, or $p_1 \times \dots \times p_f$ if v is odd. In accordance with the second requirement in 5.1, the modulus is either a domain parameter denoted n , or a claimant parameter denoted $n(A)$.

With respect to each prime factor p_j , an accreditation exponent, denoted u_j , is set equal to the least positive integer so that $u_j \times v + 1$ is a multiple of either $(p_j - 1)/2$ if $v = 2$, or $p_j - 1$ if v is an odd prime.

With respect to the modulus, an accreditation exponent, denoted u , is set equal to the least positive integer so that $u \times v + 1$ is a multiple of either $\text{lcm}(p_1 - 1, p_2 - 1)/2$ if $v = 2$, or $\text{lcm}(p_1 - 1, \dots, p_f - 1)$ if v is an odd prime.

5.2.2 Asymmetric pair(s) of numbers

5.2.2.1 Case where $v = 2$

The identification data $Id(A)$ shall be converted into m parts by appending sixteen bits representing the numbers 1 to m , namely '0001', '0002', and so on, in turn to the string $Id(A)$.

$$Id_x(A) = Id(A) \parallel '000X'$$

NOTE The mechanism below derives from the first format mechanism specified in ISO/IEC 14888-2^[21], known as PSS (PSS reads Probabilistic Signature Scheme) and due to Bellare and Rogaway^[1].

For converting each part, from $Id_i(A)$ to $Id_m(A)$, into a string of α bits, denoted F_1 to F_m , the following computational steps are performed.

- 1) The string $Id_x(A)$ shall be hashed to obtain a hash-code denoted H_x .

$$H_x = h(Id_x(A))$$
- 2) A string of $(64 + |h|)$ bits is constructed from left to right by concatenating 8 octets set to '00' and the hash-code H_x . This string shall be hashed to obtain a hash-code denoted HH_x .

$$HH_x = h('00000000 00000000' \parallel H_x)$$
- 3) Named a mask, a string of $(\alpha - |h| - 8)$ bits is constructed from the hash-code HH_x . The procedure makes use of two variables: a bit string of variable length, denoted *String*, and a 32-bit counter, denoted *Counter*.
 - a) Set *String* to the empty string.
 - b) Set *Counter* to 0.
 - c) Replace *String* by *String* $\parallel h(HH_x \parallel \text{Counter})$.
 - d) Replace *Counter* by *Counter* + 1.
 - e) If $|h| \times \text{Counter} < \alpha - |h| - 8$, then go to step c.

$Mask_x$ equals the leftmost $(\alpha - |h| - 8)$ bits of *String* where the leftmost bit has been forced to 0.

- 4) A string denoted F_x is constructed from left to right by concatenating the $(\alpha - |h| - 8)$ bits of the mask where the rightmost bit has been reversed, the $|h|$ bits of the hash-code HH_x and one octet set to 'BC'.

$$F_x = \text{Format}(Id_x(A)) = (\text{Mask}_x \oplus (000 \dots 000 \parallel 1)) \parallel HH_x \parallel \text{'BC'}$$

A public number denoted $G_x(A)$ is derived from the number represented by the bit string F_x (also denoted F_x , this number is even, non-zero and less than the modulus), as follows.

- If the Jacobi symbol $(F_x | n)$ is +1, then $G_x(A) = F_x$.
- If the Jacobi symbol $(F_x | n)$ is -1, then $G_x(A) = F_x / 2$.

The authority or the principal shall provide claimant A with m private numbers denoted Q_1 to Q_m . The private number denoted Q_x is set equal to the u -th modular power of the public number $G_x(A)$.

$$Q_x = G_x(A)^u \pmod{\text{either } n \text{ or } n(A)}$$

NOTE 1 The CRT technique (see C.2.3) may be used for converting each public number into a private number.

— For each prime factor p_j , a component Z_j is set equal to $G_x(A)^{u_j} \pmod{p_j}$.

— A CRT composition converts the set of components $\{Z_1, Z_2 \dots\}$ into a number Z .

$$Q_x = Z \pmod{\text{either } n \text{ or } n(A)}$$

NOTE 2 Each asymmetric pair of numbers verifies a relationship governed by the verification key.

$$G_x(A) \times Q_x^2 \equiv 1 \pmod{\text{either } n \text{ or } n(A)}$$

NOTE 3 Consequently, any number $G_x(A)$ or Q_x may be replaced by the modulus minus the number.

5.2.2.2 Case where v is an odd prime

NOTE The mechanism below derives from the first format mechanism specified in ISO/IEC 14888-2^[21], known as PSS (PSS reads Probabilistic Signature Scheme) and due to Bellare and Rogaway^[1].

For converting the identification data $Id(A)$ into a string of α bits, denoted F , the following computational steps are performed.

- 1) The string $Id(A)$ shall be hashed to obtain a hash-code denoted H .

$$H = h(Id(A))$$

- 2) A string of $(64 + |h|)$ bits is constructed from left to right by concatenating 8 octets set to '00' and the hash-code H . This string shall be hashed to obtain a hash-code denoted HH .

$$HH = h('00000000 00000000' \parallel H)$$

- 3) Named a mask, a string of $(\alpha - |h|)$ bits is constructed from the hash-code HH . The procedure makes use of two variables: a bit string of variable length, denoted *String*, and a 32-bit counter, denoted *Counter*.

- a) Set *String* to the empty string.
- b) Set *Counter* to 0.
- c) Replace *String* by *String* $\parallel h(HH \parallel \text{Counter})$.
- d) Replace *Counter* by *Counter* + 1.
- e) If $|h| \times \text{Counter} < \alpha - |h|$, then go to step c.

The mask equals the leftmost $(\alpha - |h|)$ bits of *String* where the leftmost bit has been forced to 0.

- 4) A string denoted F is constructed from left to right by concatenating the $(\alpha - |h|)$ bits of the mask where the rightmost bit has been reversed and the $|h|$ bits of the hash-code HH .

$$F = \text{Format}(Id(A)) = (\text{Mask} \oplus (000 \dots 000 \parallel 1)) \parallel HH$$

A public number, denoted $G(A)$, is set equal to the number represented by the bit string F (also denoted F , this number is non-zero and less than the modulus).

$$G(A) = F$$

The authority or the principal shall provide claimant A with a private number, denoted Q , set equal to the u -th modular power of the public number $G(A)$.

$$Q = G(A)^u \pmod{\text{either } n \text{ or } n(A)}$$

NOTE 1 The CRT technique (see C.2.3) may be used for converting the public number into the private number.

— For each prime factor p_j , a component Q_j is set equal to $G(A)^{u_j} \pmod{p_j}$.

— A CRT composition converts the set of components $\{Q_1, Q_2, \dots\}$ into the number Q .

NOTE 2 The asymmetric pair of numbers (the private number is the modular inverse of the RSA signature, see ISO/IEC 14888-2^[21]) verifies a relationship governed by the verification key.

$$G(A) \times Q^v \equiv 1 \pmod{\text{either } n \text{ or } n(A)}$$

5.3 Unilateral authentication exchange

The bracketed numbers in Figure 1 correspond to the steps of the mechanism, including the exchanges of information, described in detail below. The claimant is denoted A . The verifier is denoted B .

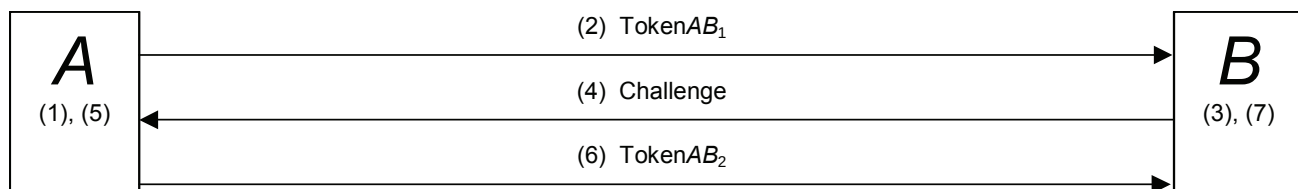


Figure 1 — Identity-based mechanism

In addition to identification data $Id(A)$, a verification exponent v (a prime number), a pair multiplicity parameter m and an exchange multiplicity parameter t , the claimant shall store a modulus n or $n(A)$ and either

- m private numbers Q_1 to Q_m if $v = 2$, or
- a single private number Q if v is an odd prime.

In addition to identification data $Id(A)$, a verification exponent v (a prime number), a pair multiplicity parameter m and an exchange multiplicity parameter t , the verifier shall be provided with a trusted copy of a modulus n or $n(A)$. If not already known by B , a copy of $Id(A)$, v , m and t shall be sent along with $TokenAB_1$; however, such a copy needs not be trusted.

For each application of the mechanism, the following procedure shall be performed t times. The verifier B shall only accept the claimant A as valid if all t iterations of the procedure complete successfully.

- 1) For each iteration of the procedure, a fresh number shall be uniformly selected at random, so that it is non-zero and less than the modulus. Denoted r , it shall be kept secret.

The fresh random number r shall be converted into a witness, denoted W , as the v -th modular power.

- Witness formula if $v = 2$: $W = r^2 \pmod{\text{either } n \text{ or } n(A)}$
- Witness formula if v is an odd prime: $W = r^v \pmod{\text{either } n \text{ or } n(A)}$

The number W is represented by a string of α bits, also denoted W .

- 2) A sends $TokenAB_1$ = either witness W or a hash-code of W and $Text$, one of four hash variants, to B .

The four hash variants are $h(W \parallel Text)$, $h(W \parallel h(Text))$, $h(h(W) \parallel Text)$, and $h(h(W) \parallel h(Text))$, where h is a hash-function and $Text$ is an optional text field (it may be empty). If the text field is non-empty, then B shall have the means to recover the value of $Text$; this may require A to send all or part of the text field at this point. The text field is available for use in applications outside the scope of this document. Annex A of ISO/IEC 9798-1 gives information on the use of text fields. The hash variant is a domain parameter.

- 3) On receipt of $TokenAB_1$, the following computational steps are performed.

- a) If the value of $v^{m \times t}$ is less than 2^{40} and/or if $m > 8$ when $v = 2$, and/or if $m > 1$ when v is an odd prime, then the procedure fails.

- b) If the identification data $Id(A)$ is invalid (e.g., expired or revoked), then the procedure fails.
- c) A fresh string of δ bits shall be uniformly selected at random.
 - If $v = 2$, then $\delta = m$ and the string consists of m bits, denoted d_1 to d_m .
 - If v is an odd prime, then $\delta = |v| - 1$ and the string represents a number less than v , possibly zero, denoted d .

NOTE The total number of possible challenges per iteration of the procedure should be limited to 2^{40} . If this recommendation is not followed, then special care should be taken to prevent the verifier using the claimant as a signing oracle.

- 4) B sends the fresh string as a challenge to A .

NOTE Optimizations may induce constraints on the Hamming weight of the challenges, with an impact on the total number of possible challenges and on the mechanism security level.

- 5) On receipt of the challenge, the following computational steps are performed.

- a) If the challenge is not a string of δ bits, then the procedure fails.
- b) A response denoted D shall be computed from the random number r and
 - the m private numbers $Q_1, Q_2, \dots, Q_m$ and the m challenge bits $d_1, d_2, \dots, d_m$ if $v = 2$.

Response formula if $v = 2$:
$$D = r \times \prod_{i=1}^m Q_i^{d_i} \pmod{\text{either } n \text{ or } n(A)}$$

- the single private number Q and the challenge number d if v is an odd prime.

Response formula if v is an odd prime:
$$D = r \times Q^d \pmod{\text{either } n \text{ or } n(A)}$$

- 6) A sends $\text{TokenAB}_2 = \text{response } D$ to B .

- 7) On receipt of TokenAB_2 , the following computational steps are performed.

- a) If the response D is **zero** or equal to or more than the modulus, then the procedure fails.
- b) The identification data $Id(A)$ shall be converted into
 - m public numbers (see 5.2.2.1), denoted $G_1(A), G_2(A), \dots, G_m(A)$, if $v = 2$.
 - a single public number (see 5.2.2.2), denoted $G(A)$, if v is an odd prime.
- c) Denoted W^* , a witness shall be computed.
 - Verification formula if $v = 2$:
$$W^* = D^2 \times \prod_{i=1}^m G_i(A)^{d_i} \pmod{\text{either } n \text{ or } n(A)}$$
 - Verification formula if v is an odd prime:
$$W^* = D^v \times G(A)^d \pmod{\text{either } n \text{ or } n(A)}$$
- d) If either witness W^* or a hash-code of W^* and Text , one of the four hash variants, is identical to TokenAB_1 received in step (2), then the iteration of the procedure is successful. Otherwise the procedure fails.

NOTE 1 Other information may be sent with any exchange of the procedure. B might use such information to help compute the value of the optional text field.

NOTE 2 B can compute the public number(s) for A at any stage, i.e., B need not wait until the receipt of response D before computing them. If B verifies A frequently, then B may cache the public number(s).

NOTE 3 The t iterations of the procedure can be performed in parallel, i.e., in the first step, A may choose t random numbers $r_1, r_2, \dots, r_t$, compute t witnesses $W_1, W_2, \dots, W_t$, send them simultaneously to B , and so on. If this parallel implementation is adopted, the total number of message exchanges will be equal to three, regardless of the value of t .

NOTE 4 The use of a hash-code instead of witness W in the first exchange of the procedure can achieve efficiency gains by reducing the number of bits in TokenAB_1 .

6 Mechanisms based on integer factorization

6.1 Security requirements for the environment

These mechanisms enable a verifier to check that a claimant knows a decomposition of a claimed modulus.

NOTE These mechanisms implement schemes due to Guillou and Quisquater^[9] and denoted GQ2.

Within a given domain, the following requirements shall be satisfied.

- 1) Domain parameters shall be selected, which will govern the operation of the mechanism. The selected parameters shall be made known in a reliable manner to all entities within the domain.
- 2) Every claimant shall be equipped with distinct prime factors so that knowledge of their product, i.e., the modulus (a claimant parameter), shall not feasibly enable any entity to deduce them, where feasibility is defined by the context of use of the mechanism.

NOTE When opening a session (see 5.1), a computer may randomly select two prime factors to be used during the session (a few hours). Using the principal's long-term secret in a "single-sign-on" of session identification data, the computer signs an "ephemeral" certificate covering an "ephemeral" modulus, product of the "ephemeral" prime factors.

- 3) Every verifier shall obtain a trusted copy of the modulus specific to the claimant.

NOTE The exact means by which the verifier obtains a trusted copy of the modulus specific to the claimant is beyond the scope of this document. This may, for example, be achieved by the use of public-key certificates or by some other environment-dependent means.

- 4) Every claimant and every verifier shall have the means to produce random numbers.
- 5) If the mechanism makes use of a hash-function, then all entities within the domain shall agree on a hash-function, e.g., one of the functions specified in ISO/IEC 10118-3.

6.2 Key production

A number, denoted α , fixes the modulus size in bits, i.e., $2^{\alpha-1} < \text{modulus} < 2^\alpha$, in accordance with the context of use of the mechanism (for further details, see C.1.1). It is a domain parameter.

A security parameter and a pair multiplicity parameter, denoted k and m , together fix a mechanism security level set to the value of $2^{-k \times m}$ in accordance with the needs of the application (see C.1.4). They are domain parameters. A value of $k \times m$ from 8 to 40 is appropriate for most applications.

NOTE The total number of possible challenges should be limited to 2^{40} . If this recommendation is not followed, then special care should be taken to prevent the verifier using the claimant as a signing oracle.

Claimant A shall keep secret two or more distinct large prime factors denoted $p_1, p_2 \dots$ in ascending order. If α is a multiple of the number of prime factors, denoted f , then the bit size of each prime factor shall be α / f (for further details, see C.1.2).

Each prime factor p_j determines a number, denoted b_j , so that $p_j - 1$ is divisible by 2^{b_j} , but not by 2^{b_j+1} , i.e., the b_j+1 least significant bits of $p_j - 1$ are one bit set to 1 followed by b_j bits set to 0 and $(p_j-1)/2^{b_j}$ is an odd number.

NOTE The number b_j is set equal to one if $p_j \equiv 3 \pmod{4}$, and to two or more if $p_j \equiv 1 \pmod{4}$.

For the equivalence with a decomposition of the modulus, the first 54 prime numbers, namely $\{2, 3, 5, 7, 11, \dots, 251\}$, i.e., of bit size equal to eight or less, are searched for an appropriate number g .

— The Legendre symbol of a candidate number g is evaluated with respect to each prime factor from p_1 to p_f . The candidate number g is appropriate if there are two prime factors p_j and p_i as follows.

- If $b_j = b_i$, the Legendre symbols are different, i.e., $(g | p_j) = -(g | p_i)$.
- If $b_j > b_i$, the Legendre symbol with respect to p_j is -1 , i.e., $(g | p_j) = -1$.

NOTE In average, each candidate number has one chance out of 2^{f-1} of being appropriate. Consequently the probability is negligible of not finding an appropriate number g within the first 54 prime numbers.

The m basic numbers are the number g , completed by as many numbers as needed from the first 54 prime numbers. They are either domain parameters, denoted g_1 to g_m in ascending order if they are the first m prime numbers, or claimant parameters, denoted $g_1(A)$ to $g_m(A)$ in ascending order otherwise.

NOTE If the m basic numbers are systematically the first m prime numbers without checking the Legendre symbols, then for f large prime factors randomly generated, the probability that the knowledge of the set of the private numbers does not imply the knowledge of a decomposition of the modulus is in average less than $2^{-m \times (f-1)}$.

An adaptation parameter denoted b is set equal to $\max(b_1 \text{ to } b_f)$. It is a claimant parameter. For each basic number g_i or $g_i(A)$, a public number denoted G_i is set equal to the b -th square of the basic number.

$$G_i = \text{Either } g_i^{2^b} \text{ or } g_i(A)^{2^b}$$

A verification exponent denoted v is set equal to 2^{k+b} . With respect to each prime factor p_j , an accreditation exponent, denoted u_j , is set equal to the least positive integer so that $v \times u_j + 1$ is a multiple of $(p_j - 1)/2^{b_j}$.

For each basic number g_i or $g_i(A)$ and each prime factor p_j , a private component denoted Q_{ij} is set equal to the u_j -th modular power of the public number G_i .

$$Q_{i,j} = G_i^{u_j} \bmod p_j$$

The modulus is set equal to the product of the large prime factors, i.e., $p_1 \times \dots \times p_f$. It is a claimant parameter denoted $n(A)$.

NOTE The same modulus may be used for the GQ2 mechanisms and for the RSA mechanisms.

6.3 Unilateral authentication exchange

The bracketed numbers in Figure 2 correspond to the steps of the mechanism, including the exchanges of information, described in detail below. The claimant is denoted A . The verifier is denoted B .

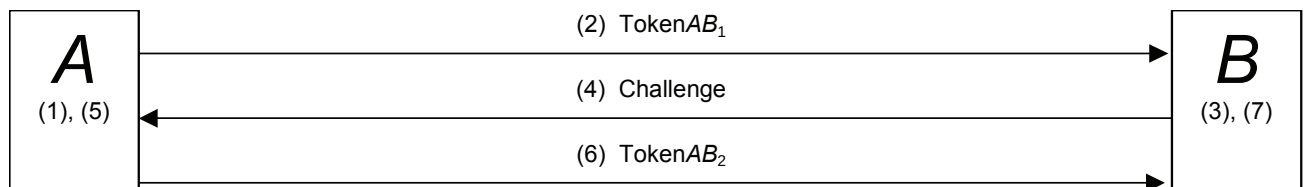


Figure 2 — Mechanism based on the factorization of a modulus

In addition to parameters b , k and m , and m basic numbers g_1 to g_m or $g_1(A)$ to $g_m(A)$, the claimant shall store either

- a modulus $n(A)$ and m private numbers Q_1 to Q_f , or
- f prime factors p_1 to p_f , $f \times m$ private components $Q_{1,1}$ to $Q_{m,f}$ and $(f-1)$ CRT coefficients (see C.2.3).

In addition to parameters b , k and m , and m basic numbers g_1 to g_m or $g_1(A)$ to $g_m(A)$, the verifier shall be provided with a trusted copy of the claimant's modulus $n(A)$. If not already known by B , a copy of b , k , m and $g_1(A)$ to $g_m(A)$ shall be sent along with TokenAB_1 ; however, such a copy needs not be trusted.

For each application of the mechanism, the following procedure shall be performed. The verifier B shall only accept the claimant A as valid if the procedure completes successfully.

- 1) For each iteration of the procedure, for each prime factor p_j , a fresh number shall be uniformly selected at random, non-zero and less than p_j . Denoted r_j , it shall be kept secret.

Each fresh random r_j number shall be converted into a witness component, denoted W_j .

Witness component formula: $W_j = r_j^v \bmod p_j$

Involving the set of prime factors and CRT coefficient(s), a CRT composition (see C.2.3) shall convert the set of witness components $\{W_1, W_2, \dots\}$ into a witness denoted W . The number W is represented by a string of α bits, also denoted W .