This is a Chapter from the **Handbook of Applied Cryptography**, by A. Menezes, P. van Oorschot, and S. Vanstone, CRC Press, 1996.
For further information, see `www.cacr.math.uwaterloo.ca/hac`

Chapter 15

Patents and Standards

Contents in Brief

15.1 Introduction

This chapter discusses two topics which have significant impact on the use of cryptography in practice: patents and standards. At their best, cryptographic patents make details of significant new processes and efficient techniques publicly available, thereby increasing awareness and promoting use; at their worst, they limit or stifle the use of such techniques due to licensing requirements. Cryptographic standards serve two important goals: facilitating widespread use of cryptographically sound and well-accepted techniques; and promoting interoperability between components involving security mechanisms in various systems.

An overview of patents is given in §15.2. Standards are pursued in §15.3. Notes and further references follow in §15.4.

15.2 Patents on cryptographic techniques

A vast number of cryptographic patents have been issued, of widely varying significance and use. Here attention is focused on a subset of these with primary emphasis on unexpired patents of industrial interest, involving fundamental techniques and specific algorithms and protocols. In addition, some patents of historical interest are noted.

Where appropriate, a brief description of major claims or disclosed techniques is given. Inclusion herein is intended to provide reference information to practitioners on the existence and content of well-known patents, and to illustrate the nature of cryptographic patents in general. There is no intention to convey any judgement on the validity of any claims.

Because most patents are eventually filed in the United States, U.S. patent numbers and associated details are given. Additional information including related filings in other countries may be found in patent databases. For further technical details, the original patents should be consulted (see §15.2.4). Where details of patented techniques and algorithms appear elsewhere in this book, cross-references are given.

Expiry of patents

U.S. patents are valid for 17 years from the date of issue, or 20 years from the date a patent application was filed. For applications filed before June 8 1995 (and unexpired at that point), the longer period applies; the 20-year rule applies for applications filed after this date.

Priority data

Many countries require that a patent be filed before any public disclosure of the invention; in the USA, the filing must be within one year of disclosure. A large number of countries are parties to a patent agreement which recognizes *priority dates*. A patent filed in such a country, and filed in another such country within one year thereof, may claim the date of the first filing as a priority date for the later filing.

Outline of patents section

The discussion of patents is broken into three main subsections. §15.2.1 notes five fundamental patents, including DES and basic patents on public-key cryptography. §15.2.2 addresses ten prominent patents including those on well-known block ciphers, hash functions, identification and signature schemes. §15.2.3 includes ten additional patents addressing various techniques, of historical or practical interest. Finally, §15.2.4 provides information on ordering patents.

15.2.1 Five fundamental patents

Table 15.1 lists five basic cryptographic patents which are fundamental to current cryptographic practice, three involving basic ideas of public-key cryptography. These patents are discussed in chronological order.

Inventors	Patent #	Issue date	Ref.	Major claim or area
Ehrsam et al.	3,962,539	Jun. 08 1976	[363]	DES
Hellman-Diffie-Merkle	4,200,770	Apr. 29 1980	[551]	Diffie-Hellman agreement
Hellman-Merkle	4,218,582	Aug. 19 1980	[553]	public-key systems
Merkle	4,309,569	Jan. 05 1982	[848]	tree authentication
Rivest-Shamir-Adleman	4,405,829	Sep. 20 1983	[1059]	RSA system

Table 15.1: *Five fundamental U.S. cryptographic patents.*

(i) DES block cipher

The patent of Ehrsam et al. (3,962,539) covers the algorithm which later became well-known as DES (§7.4). Filed on February 24 1975 and now expired, the patent was assigned to the International Business Machines Corporation (IBM). Its background section comments briefly on 1974 product cipher patents of Feistel (3,798,359) and Smith (3,796,830), respectively filed June 30 1971 and November 2 1971. It notes that while the Feistel patent discloses a product cipher which combines key-dependent linear and nonlinear transformations, it fails to disclose specific details including precisely how key bits are used, regarding the nonlinear transformation within S-boxes, and regarding a particular permutation. In addition, the effect of key bits is limited by the particular grouping used. The background section comments further on the cipher of Smith's patent, noting its inherently serial nature as a performance drawback, and that both it and that of Feistel have only two types of substitution boxes, which are selected as a function of a single key bit. Thus, apparently, the need for a new cipher. The patent contains ten (10) claims.

(ii) Diffie-Hellman key agreement

The first public-key patent issued, on April 29 1980, was the Hellman-Diffie-Merkle patent (4,200,770). Filed on September 6 1977, it was assigned to Stanford University (Stanford, California). It is generally referred to as the *Diffie-Hellman patent*, as it covers Diffie-Hellman key agreement (§12.6.1). There are two major objects of the patent. The first is a method for communicating securely over an insecure channel without *a priori* shared keys; this can be done by Diffie-Hellman key agreement. The second is a method allowing authentication of an identity over insecure channels; this can be done using authentic, long-term Diffie-Hellman public keys secured in a public directory, with derivation and use of the resulting Diffie-Hellman secret keys providing the authentication. The patent contains eight (8) claims including the idea of establishing a session key by public-key distribution, e.g., using message exchanges as in two-pass Diffie-Hellman key agreement. Claim 8 is the most specific, specifying Diffie-Hellman using a prime modulus q and exponents x_i and x_j in $[1, q-1]$.

(iii) Merkle-Hellman knapsacks and public-key systems

The Hellman-Merkle patent (4,218,582) was filed October 6 1977 and assigned to the Board of Trustees of the Leland Stanford Junior University (Stanford, California). It covers public-key cryptosystems based on the subset-sum problem, i.e., Merkle-Hellman trapdoor knapsacks (now known to be insecure – see §8.6.1), in addition to various claims on public-key encryption and public-key signatures. The objects of the invention are to allow private conversations over channels subject to interception by eavesdroppers; to allow authentication of a receiver's identity (through its ability to use a key only it would be able to compute); and to allow data origin authentication without the threat of dispute (i.e., via public-key techniques, rather than a shared secret key). There are seventeen (17) claims, with Claims 1–6 broadly applying to public-key systems, and Claims 7–17 more narrowly focused on knapsack systems. The broad claims address aspects of general methods using public-private key pairs for public-key encryption, public-key signatures, and the use of public-key encryption to provide authentication of a receiver via the receiver transmitting back to the sender a representation of the enciphered message.

(iv) Tree authentication method of validating parameters

Merkle's 1982 patent (4,309,569) covers tree authentication (§13.4.1). It was filed September 5 1979, and assigned to the Board of Trustees of the Leland Stanford Junior University (Stanford, California). The main motivation cited was to eliminate the large storage requirement inherent in prior one-time signature schemes, although the idea has wider application. The main ideas are to use a binary tree and a one-way hash function to allow authentication of leaf values Y_i associated with each user i. Modifications cited include: use of a ternary or k-ary tree in place of a binary tree; use of the tree for not only public values of one-time signatures, but for authenticating arbitrary public values for alternate purposes; and use of a distinct authentication tree for each user i, the root R_i of which replaces Y_i above, thereby allowing authentication of all values in i's tree, rather than just a single Y_i. The epitome of conciseness, this patent contains a single figure and just over two pages of text including four (4) claims.

(v) RSA public-key encryption and signature system

The Rivest-Shamir-Adleman patent (4,405,829) was filed December 14 1977, and assigned to the Massachusetts Institute of Technology. It covers the RSA public-key encryption (§8.2.1) and digital signature method (§11.3.1). Also mentioned are generalizations, including: use of a modulus n which is a product of three or more primes (not necessarily distinct); and using an encryption public key e to encrypt a message M to a ciphertext C by evaluating a polynomial $\sum_{i=0}^{t} a_i M^e \bmod n$ where e and a_i, $0 \le i \le t$, are integers, and recovering the plaintext M by "utilizing conventional root-finding techniques, choosing which of any roots is the proper decoded version, for example, by the internal redundancy of the message". Other variations mentioned include using RSA encipherment in CFB mode, or as a pseudorandom number generator to generate key pads; signing a compressed version of the message rather than the message itself; and using RSA encryption for key transfer, the key thereby transferred to be used in another encryption method. This patent has the distinction of a claims section, with forty (40) claims, which is longer than the remainder of the patent.

15.2.2 Ten prominent patents

Ten prominent patents are discussed in this section, in order as per Table 15.2.

Inventors	Patent #	Issue date	Ref.	Major claim or area
Okamoto et al.	4,625,076	Nov. 25 1986	[952]	ESIGN signatures
Shamir-Fiat	4,748,668	May 31 1988	[1118]	Fiat-Shamir identification
Matyas et al.	4,850,017	Jul. 18 1989	[806]	control vectors
Shimizu-Miyaguchi	4,850,019	Jul. 18 1989	[1125]	FEAL cipher
Brachtl et al.	4,908,861	Mar. 13 1990	[184]	MDC-2, MDC-4 hashing
Schnorr	4,995,082	Feb. 19 1991	[1095]	Schnorr signatures
Guillou-Quisquater	5,140,634	Aug. 18 1992	[523]	GQ identification
Massey-Lai	5,214,703	May 25 1993	[791]	IDEA cipher
Kravitz	5,231,668	Jul. 27 1993	[711]	DSA signatures
Micali	5,276,737	Jan. 04 1994	[861, 862]	'fair' key escrow

Table 15.2: *Ten prominent U.S. cryptographic patents.*

(i) ESIGN signatures

The Okamoto-Miyaguchi-Shiraishi-Kawaoka patent (4,625,076) covers the original ESIGN signature scheme (see §11.7.2). The patent was filed March 11 1985 and assigned to the Nippon Telegraph and Telephone Corporation (Tokyo), with priority data listed as March 19 1984 (Japanese patent office). The objective is to provide a signature scheme faster than RSA. The patent contains twenty-five (25) claims.

(ii) Fiat-Shamir identification and signatures

The Shamir-Fiat patent (4,748,668) covers Fiat-Shamir identification (§10.4.2) and signatures (§11.4.1). It was filed July 9 1986, and assigned to Yeda Research and Development Co. Ltd. (Israel). For identification, the inventors suggest a typical number of rounds t as 1 to 4, and parameter selections including $k = 5$ (secrets), $t = 4$ for a 2^{-20} probability of forgery, and $k = 6$, $t = 5$ for 2^{-30}. A range of parameters k, t for $kt = 72$ is tabulated for the corresponding signature scheme, showing tradeoffs between key storage, signature size, and real-time operations required. Noted features relative to prior art include being able to pipeline computations, and being able to change the security level after the key is selected (e.g., by changing t). Generalizations noted include replacing square roots by cubic or higher roots. There are forty-two (42) claims.

(iii) Control vectors for key management

The Matyas-Meyer-Brachtl patent (4,850,017) is one of several in the area of control vectors for key management, in this case allowing a sending node to constrain the use of keys at a receiving node. It was filed May 29 1987 and assigned to the IBM Corporation. Control vectors reduce the probability of key misuse. Two general methods are distinguished. In the first method, the key and a control value are authenticated before use through verification of a special authentication code, the key for which is part of the data being authenticated. In the second method (see §13.5.2), the key and control value are cryptographically bound at the time of key generation, such that recovery of the key requires specification of the correct control vector. In each method, additional techniques may be employed to control which users may use the key in question. The patent contains twenty-two (22) claims.

(iv) FEAL block cipher

The Shimizu-Miyaguchi patent (4,850,019) gives the originally proposed ideas of the FEAL block cipher (see §7.5). It was filed November 3 1986 and assigned to the Nippon Telegraph and Telephone Corporation (Tokyo), with priority data listed as November 8 1985 (Japanese patent office). Embodiments of FEAL with various numbers of rounds are described, with figures including four- and six-round FEAL (now known to be insecure – see Note 7.100), and discussion of key lengths including 128 bits. The patent makes twenty-six (26) claims.

(v) MDC-2/MDC-4 hash functions

The patent of Brachtl et al. (4,908,861) covers the MDC-2 and MDC-4 hash functions (§9.4.1). It was filed August 28 1987 and assigned to the IBM Corporation. The patent notes that interchanging internal key halves, as is done at a particular stage in both algorithms, is actually required for security in MDC-2 but not MDC-4; however, the common design was nonetheless used, to allow MDC-4 to be implemented using MDC-2 twice. A preliminary section of the patent discusses alternatives for providing message authentication (see §9.6), as well as estimates of the security of the new hash functions, and justification for fixing certain bits within the specification to avoid effects of weak DES keys. There are twenty-one (21) claims, mainly on building $2N$-bit hash functions from N-bit block ciphers.

(vi) Schnorr identification and signatures

The Schnorr patent (4,995,082) covers Schnorr's identification (§10.4.4) and signature (§11.5.3) schemes, and optimizations thereof involving specific pre-processing. It was filed February 23 1990, with no assignee listed, and priority data given as February 24 1989 (European patent office). There are eleven (11) claims. Part of Claim 6 covers a specific variation of the Fiat-Shamir identification method using a prime modulus p, such that $p-1$ is divisible by a prime q, and using a base β of order q.

(vii) GQ identification and signatures

The Guillou-Quisquater patent (5,140,634) addresses GQ identification (Protocol 10.31) and signatures (Algorithm 11.48). It was filed October 9 1991, as a continuation-in-part of two abandoned applications, the first filed September 7 1988. The original assignee was the U.S. Philips Corporation (New York). The disclosed techniques allow for authentication of so-called *accreditation information*, authentication of messages, and the signing of messages. The central authentication protocol involves a commitment-challenge-response method and is closely related to the zero-knowledge-based identification technique of Fiat and Shamir (Protocol 10.24). However, it requires only a single protocol execution and single accreditation value, rather than a repetition of executions and a plurality of accreditation values. The cited advantages over previous methods include smaller memory requirements, and shorter overall duration due to fewer total message exchanges. The main applications cited are those involving chipcards in banking applications. There are twenty-three (23) claims, including specific claims involving the use of chipcards.

(viii) IDEA block cipher

The Massey-Lai patent (5,214,703) covers the IDEA block cipher (§7.6), proposed as a European or international alternative to DES offering greater key bitlength (and thereby, hopefully greater security). It was filed May 16 1991, and assigned to Ascom Tech AG (Bern), with priority data given as May 18 1990 from the original Swiss patent. A key concept in the cipher is the use of at least two different types of arithmetic and logical operations, with emphasis on different operations in successive stages. Three such types of operation are proposed: addition mod 2^m, multiplication mod $2^m + 1$, and bitwise exclusive-or (XOR). Symbols denoting these operations, hand-annotated in the European version of the patent (WO 91/18459, dated 28 November 1991, in German), appear absent in the text of the U.S. patent, making the latter difficult to read. There are fourteen (14) figures and ten (10) multi-part claims.

(ix) DSA signature scheme

The patent of Kravitz (5,231,668), titled "Digital Signature Algorithm", has become widely known and adopted as the DSA (§11.5.1). It was filed July 26 1991, and assigned to "The United States of America as represented by the Secretary of Commerce, Washington, D.C." The background section includes a detailed discussion of ElGamal signatures and Schnorr signatures, including their advantage relative to RSA – allowing more efficient on-line signatures by using off-line precomputation. Schnorr signatures are noted as more efficient than ElGamal for communication and signature verification, although missing some "desirable features of ElGamal" and having the drawback that cryptanalytic experience and confidence associated with the ElGamal system do not carry over. DSA is positioned as having all the efficiencies of the Schnorr model, while remaining compatible with the ElGamal model from an analysis perspective. In the exemplary specification of DSA, the hash function used was MD4. The patent makes forty-four (44) claims.

(x) Fair cryptosystems and key escrow

Micali's patent (5,276,737) and its continuation-in-part (5,315,658), respectively filed April 20 1992 and April 19 1993 (with no assignees listed), cover key escrow systems called "fair cryptosystems" (cf. §13.8.3). The subject of the first is a method involving a public-key cryptosystem, for allowing third-party monitoring of communications (e.g., government wiretapping). A number of shares (see secret-sharing – §12.7) created from a user-selected private key are given to a set of trustees. By some method of verifiable secret sharing, the trustees independently verify the authenticity of the shares and communicate this to an authority, which approves a user's public key upon receiving all such trustee approvals. Upon proper authorization (e.g., a court order), the trustees may then subsequently provide their shares to the authority to allow reconstruction of a user private key. Exemplary systems include transforming Diffie-Hellman (see paragraph below) and RSA public-key systems into fair cryptosystems. Modifications require only k out of n trustees to contribute shares to recover a user secret and prevent trustees from learning the identity of a user whose share is requested. The patent contains eighteen (18) claims, the first 14 being restricted to public-key systems.

A fair cryptosystem for Diffie-Hellman key agreement modulo p, with a generator g and n trustees, may be constructed as follows. Each user A selects n integers $s_1, \dots, s_n$ in the interval $[1, p-1]$, and computes $s = \sum_{i=1}^{n} s_i \bmod p$, public shares $y_i = g^{s_i} \bmod p$, and a public key $y = g^s \bmod p$. Trustee T_i, $1 \le i \le n$, is given y, public shares $y_1, \dots, y_n$, and the secret share s_i to be associated with A. Upon verifying $y_i = g^{s_i}$, T_i stores (A, y, s_i), and sends the authority a signature on $(i, y, y_1, \dots, y_n)$. Upon receiving such valid signatures from all n trustees, verifying the y_i in the signed messages are identical, and that $y = \prod y_i \bmod p$, the authority authorizes y as A's Diffie-Hellman public key.

The continuation-in-part pursues time-bounded monitoring in greater detail, including use of tamper-proof chips with internal clocks. Methods are also specified allowing an authority (hereafter, the government) access to session keys, including users employing a master key to allow such access. A further method allows verification, without monitoring content, that transmitted messages originated from government-approved devices. This may involve tamper-proof chips in each communicating device, containing and employing a government master key K_M. Such devices allow verification by transmitting a redundant data string dependent on this key. The continuation-in-part has thirteen (13) claims, with the first two (2) restricted to public-key systems. Claims 11 and 12 pursue methods for verifying that messages originate from a tamper-proof device using an authorized encryption algorithm.

15.2.3 Ten selected patents

Ten additional patents are discussed in this section, as listed in Table 15.3. These provide a selective sample of the wide array of existing cryptographic patents.

Inventors	Patent #	Issue date	Ref.	Major claim or area
Feistel	3,798,359	Mar.19 1974	[385]	Lucifer cipher
Smid-Branstad	4,386,233	May 31 1983	[1154]	key notarization
Hellman-Pohlig	4,424,414	Jan. 03 1984	[554]	Pohlig-Hellman cipher
Massey, Omura	4,567,600	Jan. 28 1986	[792, 956]	normal basis arithmetic
Hellman-Bach	4,633,036	Dec. 30 1986	[550]	generating strong primes
Merkle	4,881,264	Nov. 14 1989	[846]	one-time signatures
Goss	4,956,863	Sep. 11 1990	[519]	Diffie-Hellman variation
Merkle	5,003,597	Mar. 26 1991	[847]	Khufu, Khafre ciphers
Micali et al.	5,016,274	May 14 1991	[864]	on-line/off-line signing
Brickell et al.	5,299,262	Mar. 29 1994	[203]	exponentiation method

Table 15.3: *Ten selected U.S. cryptographic patents.*

(i) Lucifer cipher

Feistel's patent (3,798,359) is of historical interest. Filed June 30 1971 and assigned to the IBM Corporation, it has now expired. The background section cites a number of earlier cipher patents including ciphering wheel devices and key stream generators. The patent discloses a block cipher, more specifically a product cipher noted as being under the control of subscriber keys, and designed to resist cryptanalysis "not withstanding ... knowledge of the structure of the system" (see Chapter 7 notes on §7.4). It is positioned as distinct from prior art systems, none of which "utilized the advantages of a digital processor and its inherent speed." The patent has 31 figures supporting (only) six pages of text plus one page of thirteen (13) claims.

(ii) Key notarization

The Smid-Branstad patent (4,386,233) addresses key notarization (§13.5.2). It was filed September 29 1980, with no assignee listed. A primary objective of key notarization is to prevent key substitution attacks. The patent contains twenty-one (21) claims.

(iii) Pohlig-Hellman exponentiation cipher

The Hellman-Pohlig patent (4,424,414) was filed May 1 1978 (four and one-half months after the RSA patent), and assigned to the Board of Trustees of the Leland Stanford Junior University (Stanford, California). It covers the Pohlig-Hellman symmetric-key exponentiation cipher, wherein a prime q is chosen, along with a secret key K, $1 \leq K \leq q-2$, from which a second key D, $1 \leq D \leq q-2$, is computed such that $KD \equiv 1 \bmod (q-1)$. A message M is enciphered as $C = M^K \bmod q$, and the plaintext is recovered by computing $C^D \bmod q = M$. Two parties make use of this by arranging, *a priori*, to share the symmetric-keys K and D. The patent contains two (2) claims, specifying a method and an apparatus for implementing this block cipher. Although of limited practical significance, this patent is often confused with the three well-known public-key patents of Table 15.1.

(iv) Arithmetic in $\mathbb{F}_{2^m}$ using normal bases

Two patents of Massey and Omura are discussed here. The Omura-Massey patent (4,587,627) teaches a method for efficient multiplication of elements of a finite field $\mathbb{F}_{2^m}$ by exploiting normal bases representations. It was filed September 14 1982, with priority data November 30 1981 (European patent office), and was issued May 6 1986 with the assignee being OMNET Associates (Sunnyvale, California). The customary method for representing a field element $\beta \in \mathbb{F}_{2^m}$ involves a polynomial basis $1, x, x^2, x^3, \ldots, x^{m-1}$, with $\beta = \sum_{i=0}^{m-1} a_i x^i$, $a_i \in \{0,1\}$ (see §2.6.3). Alternatively, using a normal basis $x, x^2, x^4, \ldots, x^{2^{m-1}}$ (with x selected such that these are linearly independent) allows one to represent β as $\beta = \sum_{i=0}^{m-1} b_i x^{2^i}$, $b_i \in \{0,1\}$. The inventors note that this representation "is unconventional, but results in much simpler logic circuitry". For example, squaring in this representation is particularly efficient (noted already by Magleby in 1963) – it requires simply a rotation of the coordinate representation from $[b_{m-1} \ldots b_1 b_0]$ to $[b_{m-2} \ldots b_1 b_0 b_{m-1}]$. This follows since $x^{2^m} \equiv 1$ and squaring in $\mathbb{F}_{2^m}$ is a linear operation in the sense that $(B+C)^2 = B^2+C^2$; furthermore, $D = B \times C$ implies $D^2 = B^2 \times C^2$. From this, the main object of the patent follows directly: to multiply two elements B and C to yield $D = B \times C = [d_{m-1} \ldots d_1 d_0]$, the same method used for computing d_{m-1} can be used to sequentially produce d_i, $m-2 \leq i \leq 0$, by applying it to one-bit rotations of the representations of B and C. Alternatively, m such identical processes can be used to compute the m components d_i in parallel. The patent makes twenty-four (24) claims.

The closely related Massey-Omura patent (4,567,600) includes claims on exponentiation in $\mathbb{F}_{2^m}$ using normal bases. It was likewise filed September 14 1982 and assigned to OMNET Associates (Sunnyvale, California), with priority date February 2 1982 (European patent office). Its foundation is the observation that using a normal basis representation allows efficient exponentiation in $\mathbb{F}_{2^m}$ (Claim 16), since the cost of squaring (see above) in the customary square-and-multiply exponentiation technique is eliminated. A second subject is the implementation of Shamir's three-pass protocol (Protocol 12.22) using modular exponentiation in $\mathbb{F}_{2^m}$ as the ciphering operation along with a normal basis representation for elements; and subsequently employing a shared key, established by this method, as the key in an $\mathbb{F}_{2^m}$ exponentiation cipher (cf. Hellman-Pohlig patent) again using normal bases. A further object is a method for computing pairs of integers e, d such that $ed \equiv 1 \bmod 2^m - 1$. Whereas customarily e is selected and, from it, d is computed via the extended Euclidean algorithm (which involves division), the new technique selects a group element H of high order, then chooses a random integer R in $[1, 2^m - 2]$, and computes $e = H^R$, $d = H^{-R}$. The patent includes twenty-six (26) claims in total.

(v) Generation of strong primes

The Hellman-Bach patent (4,633,036) covers a method for generating RSA primes p and q and an RSA modulus $n = pq$ satisfying certain conditions such that factoring n is believed to be computationally infeasible. The patent was filed May 31 1984 and assigned to Martin E. Hellman. The standard strong prime conditions (Definition 4.52) are embedded: $p - 1$ requiring a large prime factor r; $p + 1$ requiring a large prime factor s; and $r - 1$ requiring a large prime factor r'. A new requirement according to the invention was that $s - 1$ have a large prime factor s', with cited justification that the (then) best known factoring methods exploiting small s' required s' operations. The patent includes twenty-four (24) claims, but is now apparently of historical interest only, as the best-known factoring techniques no longer depend on the cited properties (cf. §4.4.2).

(vi) Efficient one-time signatures using expanding trees

Merkle's 1989 patent (4,881,264), filed July 30 1987 with no assignee listed on the issued patent, teaches how to construct authentication trees which may be expanded arbitrarily, without requiring a large computation when a new tree is constructed (or expanded). The primary cited use of such a tree is for making available public values y (corresponding to secret values x) of a user A in a one-time signature scheme (several of which are summarized). In such schemes, additional public values are continually needed over time. The key idea is to associate with each node in the tree three vectors of public information, each of which contains sufficient public values to allow one one-time signature; call these the LEFT, RIGHT, and MESSAGE vectors. The combined hash value H_i of all three of these vectors serves as the hash value of the node i. The root hash value H_1 is made widely available, as per the root value of ordinary authentication trees (§13.4.1). A new message M may be signed by selecting a previously unused node of the tree (e.g., H_1), using the associated MESSAGE vector for a one-time signature thereon. The tree may be expanded downward from node i (e.g., $i = 1$), to provide additional (verifiably authentic) public values in a new left sub-node $2i$ or a right sub-node $2i + 1$, by respectively using the LEFT and RIGHT vectors at node i to (one-time) sign the hashes H_{2i} and H_{2i+1} of the newly created public values in the respective new nodes. Full details are given in the patent; there are nine (9) claims.

The one-time signatures themselves are based on a symmetric cipher such as DES; the associated one-way function F of a private value x may be created by computing $y = F(x) = DES_x(0)$, i.e., encrypting a constant value using x as key; and a hash function for the authentication tree may also be constructed using DES. Storage requirements on user A for its own tree are further reduced by noting that only x values need be stored; and that these may be pseudorandomly generated, for example, letting $J = 0, 1, 2$ denote the LEFT, RIGHT, and MESSAGE vectors, and assuming that K public values are needed per one-time signature, the K^{th} value x in a vector of public values at node I may be defined as $x[I, J, K] = DES_{K_A}(I||J||K)$, where K_A is A's secret key and "||" denotes concatenation.

(vii) Goss variation of Diffie-Hellman

The patent of Goss (4,956,863) covers a variation of Diffie-Hellman key agreement essentially the same as Protocol 12.53. It was filed April 17 1989 and assigned to TRW Inc. (Redondo Beach, California). The primary application cited is an authenticated key establishment technique, completely transparent to end-users, for facsimile (FAX) machines on existing telephone networks. At the time of manufacture, a unique device identifier and a signed certificate binding this to a long-term Diffie-Hellman public key (public exponential) is embedded in each device. The identity in the certificate, upon verification, may be used as the basis on which to accept or terminate communications channels. Such a protocol allows new session keys for each FAX call, while basing authentication on long-term certified keys (cf. Remark 12.48; but regarding security, see also Note 12.54). The patent makes sixteen (16) claims.

(viii) Khufu and Khafre block ciphers

Merkle's 1991 patent (5,003,597) covers two symmetric-key block ciphers named Khufu and Khafre (see §7.7.3). These were designed specifically as fast software-oriented alternatives to DES, which itself was designed with hardware performance in mind. The patent was filed December 21 1989 and assigned to the Xerox Corporation. Khufu and Khafre have block size 64 bits and a user-selectable number of rounds. Khufu has key bitlength up to 512 bits, and S-boxes derived from the input key; it encrypts 64-bit blocks faster than Khafre. Khafre has fixed S-boxes, and a key of selectable size (with no upper bound), though larger keys impact throughput. The majority of the patent consists of C-code listings specifying the ciphers. The patent contains twenty-seven (27) claims.

(ix) On-line/off-line digital signatures

The Micali-Goldreich-Even patent (5,016,274) teaches on-line/off-line digital signature schemes. The patent was filed November 8 1988, with no assignee listed. The basic idea is to carry out a precomputation to reduce real-time requirements for signing a particular message m. The pre-computation, executed during idle time and independent of m, involves generation of matching one-time public and private keying material for a fast (one-time) first signature scheme, and using a second underlying signature scheme to create a signature s_2 over the one-time public key. This key from the first scheme is then used to create a signature s_1 on m. The overall signature on m is (s_1, s_2). Appropriate hash functions can be used as usual to allow signing of a hash value $h(m)$ rather than m. In the exemplary method, Rabin's scheme is the underlying signature scheme, and DES is used both to build a one-time signature scheme and for hashing. Regarding security of the overall scheme, a one-time scheme, if secure, is presumed secure against chosen-text attack (since it is used only once); the underlying scheme is secure against chosen-text attack because it signs only strings independent of a message m. The method thus may convert any signature scheme into one secure against chosen-text attacks (should this be a concern), or convert any underlying signature scheme to one with smaller real-time requirements. The patent contains thirty-three (33) claims.

(x) Efficient exponentiation for fixed base

The Brickell-Gordon-McCurley patent (5,299,262) teaches a method for fast exponentiation for the case where a fixed base is re-used; see also page 633. This has application in systems such as the ElGamal, Schnorr, and DSA signature schemes. The patent was filed August 13 1992, issued March 29 1994, and assigned to "The United States of America as represented by the United States Department of Energy, Washington, D.C." The method is presented in Algorithm 14.109. The patent contains nine (9) claims.

15.2.4 Ordering and acquiring patents

Any American patent may be ordered by patent number from the U.S. Patent and Trademark Office (PTO). Written requests should be posted to: PTO, Washington, D.C., 20231, USA. Telephone requests may also be made at +703-305-4350, with payment by credit card. A nominal fee applies (e.g., US$3 for patents returned by postal mail; or US$6 for returns by fax, usually the same day). For on-line information on recent patents, consult URL `http://www.micropatent.com` (e.g., specifying patent class code 380 for cryptography).

15.3 Cryptographic standards

This section summarizes cryptographic and security standards of practical interest. These facilitate widespread use of cryptographically sound techniques, and interoperability of systems and system components. Tables 15.4–15.11 present an overview allowing relevant standards to be located and identified, and access to formal title information allowing acquisition of particular standards. These tables may also be used to locate standards addressing particular areas (e.g., key management). For specific details of techniques and algorithms, the original standards should be consulted. Where relevant technical details appear elsewhere in the book, cross-references are given.

Outline of standards section

§15.3.1 presents international (ISO and ISO/IEC) application-independent standards on cryptographic techniques. §15.3.2 summarizes banking security standards, subdivided into ANSI and ISO standards. §15.3.3 considers international security architectures and frameworks (ISO and X.509). §15.3.4 summarizes security-related standards for use by U.S. federal government departments. §15.3.5 addresses selected Internet specifications, while §15.3.6 notes selected de facto industry standards. §15.3.7 provides information allowing acquisition of standards.

15.3.1 International standards – cryptographic techniques

The International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) develop standards individually and jointly. Joint standards are developed under the joint technical committee ISO/IEC JTC 1. ISO and ISO/IEC standards progress through the following draft stages before maturing to the International Standard status: Working Draft (WD); Committee Draft (CD); and Draft International Standard (DIS). Each ISO and ISO/IEC standard is reviewed every five years, at which time it is either reaffirmed, revised, or retracted. The ISO/IEC subcommittee responsible for standardizing generic cryptographic techniques is SC 27 (ISO/IEC JTC 1 SC 27). Table 15.4 lists selected ISO and ISO/IEC standards on cryptographic techniques.

ISO 8372: This standard specifies the four well-known modes of operation of a block cipher – electronic codebook (ECB), cipher block chaining (CBC), cipher feedback (CFB), and output feedback (OFB). These modes were originally standardized for DES in FIPS 81 (1980) and ANSI X3.106 (1983). ISO 8372 (first published in 1987) specifies these modes for general 64-bit block ciphers (cf. ISO/IEC 10116).

ISO #	Subject	Ref.
8372	modes of operation for a 64-bit cipher	[574]
9796	signatures with message recovery (e.g., RSA)	[596]
9797	data integrity mechanism (MAC)	[597]
9798–1	entity authentication – introduction	[598]
9798–2	— using symmetric encipherment	[599]
9798–3	— using public-key techniques	[600]
9798–4	— using keyed one-way functions	[601]
9798–5	— using zero-knowledge techniques	[602]
9979	register of cryptographic algorithms	[603]
10116	modes of operation for an n-bit cipher	[604]
10118–1	hash functions – introduction	[605]
10118–2	— using block ciphers	[606]
10118–3	— customized algorithms	[607]
10118–4	— using modular arithmetic	[608]
11770–1	key management – introduction	[616]
11770–2	— symmetric techniques	[617]
11770–3	— asymmetric techniques	[618]
13888–1	non-repudiation – introduction	[619]
13888–2	— symmetric techniques	[620]
13888–3	— asymmetric techniques	[621]
14888–1	signatures with appendix – introduction	[622]
14888–2	— identity-based mechanisms	[623]
14888–3	— certificate-based mechanisms	[624]

Table 15.4: *ISO and ISO/IEC standards for generic cryptographic techniques.*

ISO/IEC 9796: This standard specifies a generic mechanism for digital signature schemes giving message recovery (see §11.3.5 and ANSI X9.31–1; cf. ISO/IEC 14888). Examples are given in its Annex B corresponding to RSA and Rabin's variant thereof (with encryption exponent 2). The main part of the standard is a redundancy scheme, intended to be generically applicable to a large class of signature schemes, although specifically designed to preclude attacks on schemes such as RSA and Rabin which have a multiplicative property.

ISO/IEC 9797: This standard defines a message authentication code (MAC) based on the CBC mode of operation of a block cipher, similar to the MAC algorithms of ISO 8731–1, ISO 9807, ANSI X9.9, and ANSI X9.19 (see Algorithm 9.58).[1] Relative to these, in 9797 the m-bit MAC result is constrained only by $m \leq n$ (the leftmost or most significant bits are retained), the block cipher is unspecified but has n-bit blocks, and a second padding method is specified. These other MAC algorithms may be viewed as special cases of 9797; for example, the specific values $n = 64$ and $m = 32$ along with use of the first padding method (see below) and DES as the block cipher yields the MAC of X9.9.

In 9797, one of two specified padding methods must be selected (Algorithms 9.29, 9.30). The first pads the data input by appending zero or more 0-bits, as few as necessary, to obtain a string whose bitlength is a multiple of n. The second method always appends to the data input a single 1-bit, and then zero or more 0-bits, as few as necessary, to obtain

[1]Specific technical details are provided for MAC standards in this chapter moreso than for other standards, in an attempt to clarify the differences between the large number of CBC-MAC standards which differ only in fine details.

a string whose bitlength is a multiple of n. Annex A specifies two optional processes; Annex B provides examples. The first optional process is the optional process as described under ANSI X9.19 in §15.3.2; this reduces the threat of exhaustive key search and chosen-plaintext attacks, and is recommended when $m = n$ (see Remark 9.59). The alternative second optional process, providing protection against chosen-plaintext attacks, employs a second key K' (possibly derived from K) to encrypt the (previously final) output block, before extracting the m-bit MAC result.

ISO/IEC 9798: Parts subsequent to the introduction (9798–1) of this standard specify entity authentication mechanisms based on: symmetric encryption algorithms (9798–2); public-key signature algorithms (9798–3); a cryptographic check function or MAC (9798–4); and other customized techniques (9798–5), historically referred to by academics as zero-knowledge techniques. The mechanisms use timestamps, sequence numbers, and random numbers as time-variant parameters (§10.3.1). The 9798-3 mechanisms are functionally analogous to those of X.509, and the 9798-3 two-pass and three-pass techniques based on random number challenge-response are the source for those in FIPS 196.

9798-2 specifies four entity authentication mechanisms (as given in §10.3.2) involving two parties A and B and requiring that they share a symmetric key *a priori*, for use in a symmetric encryption algorithm. When timestamps or sequence numbers are used, these mechanisms require one and two messages, respectively, for unilateral and mutual entity authentication; using challenge-response based on random numbers, one additional message is required in each case. 9798-3 includes four analogous mechanisms (see §10.3.3) wherein the role of the symmetric encryption algorithm is replaced by a digital signature algorithm, and the requirement of shared symmetric keys is replaced by that of possession of authentic (or the capability to authenticate) public keys. 9798-4 specifies four analogous mechanisms (again see §10.3.2) where symmetric encryption as used in 9798-2 is replaced by a cryptographic check function or MAC. 9798-2 specifies two additional mutual authentication mechanisms for the case that A and B do not share a key *a priori*, but each does share a key with a trusted third party T; these require two further messages (for communication with T) beyond those for the respective mutual entity authentication mechanisms above. 9798-5 (draft) includes an identity-based identification protocol of which Fiat-Shamir (cf. Protocol 10.24) and GQ identification (Protocol 10.31) are special cases, and a protocol based on public-key decryption with witness (see §10.3.3).

ISO/IEC 9979: This standard specifies procedures allowing certain entities (e.g., ISO member bodies and liaison organizations) to register encryption algorithms in an official ISO register of such algorithms. Registration involves no security evaluation or assessment (the policy of ISO/IEC is to not standardize encryption algorithms themselves). The standard specifies the formats required for such register entries, and registration results in the assignment of a unique identifier to each algorithm, e.g., to allow interoperability. For further information, see page 660.

ISO/IEC 10116: This standard specifies the same four modes of block-cipher operation as ISO 8372, but subsumes that standard by allowing general n-bit block ciphers. ISO/IEC 10116 also provides greater detail regarding various properties of the modes, and sample calculations based on DES.

ISO/IEC 10118: This is a multi-part standard on cryptographic hashing algorithms. 10118–1 specifies common definitions and general requirements. 10118–2 specifies two generic constructions based on n-bit block ciphers: the Matyas-Meyer-Oseas hash function (Algorithm 9.41) and a block-cipher independent MDC-2 (cf. Algorithm 9.46). The draft standard 10118–3 includes SHA–1 (Algorithm 9.53), RIPEMD-128 and RIPEMD-160 (Algorithm 9.55). The draft 10118–4 includes MASH-1 and MASH-2 (see Algorithm 9.56).

ISO/IEC 11770: This multi-part standard addresses generic key management and specifies key establishment mechanisms. 11770–1 is a key management framework and overview including discussion of the key life cycle, protection requirements for keying material, and roles of third parties in key establishment. 11770-2 specifies key establishment mechanisms based on symmetric techniques, including those wherein two parties communicate point-to-point (as in §12.3.1), those similar to the Kerberos and Otway-Rees protocols involving a trusted server or key distribution center (§12.3.2), and those involving a key translation center (e.g., Protocol 13.12). 11770-3 specifies key establishment mechanisms based on asymmetric techniques. These are divided into key agreement protocols, practical instantiations of which are based on Diffie-Hellman and similar techniques (§12.6.1); and key transfer protocols, which typically involve both public-key encryption and digital signatures (§12.5.2) including adaptations of the random number based ISO/IEC 9798-3 mechanisms involving transfer of an embedded encrypted key.

ISO/IEC 13888: This multi-part (draft) standard addresses non-repudiation services (protection against false denials) related to the transfer of a message from an originator to a recipient. Mechanisms are specified for non-repudiation of origin (denial of being the originator of a message), non-repudiation of delivery (denial of having received a message), and non-repudiation associated with the actions of a third party acting as a transfer agent on behalf of others. 13888–1 (draft) provides a non-repudiation model and overview. 13888-2 (draft) specifies mechanisms involving symmetric techniques (encipherment and keyed one-way functions). 13888-3 (draft) specifies mechanisms involving asymmetric techniques and the use of digital signatures.

ISO/IEC 14888: This multi-part (draft) standard addresses schemes for signature with appendix (see §11.2.2 and ANSI X9.30–1; cf. ISO/IEC 9796). 14888–1 (draft) provides common definitions and a general overview including models outlining the steps required for signature generation and various classes of verification processes. 14888–2 (draft) addresses identity-based signature mechanisms, wherein the signature verification key is a public function of the signer's identity. 14888–3 (draft) addresses certificate-based mechanisms, wherein this public key is explicitly specified and, for example, distributed by means of a certificate. These may include DSA and similar signature mechanisms such as ElGamal, Schnorr signatures, and RSA.

15.3.2 Banking security standards (ANSI, ISO)

This section considers banking security standards developed by ANSI and by ISO. Banking security standards are typically divided into wholesale and retail banking (see Table 15.5). *Wholesale banking* involves transactions between financial institutions. *Retail banking* involves transactions between institutions and private individuals, including automated teller machine (ATM) and point-of-sale (POS) transactions, and credit authorizations.

category	transaction volume	average transaction value
retail	high (millions per day)	\$50
wholesale	low (thousands per day)	\$3 million

Table 15.5: *Retail vs. wholesale banking characteristics.*

(i) ANSI encryption standards

The American National Standards Institute (ANSI) develops standards through various Accredited Standards Committees (ASCs). Accreditation implies that standards developed under a particular committee become ANSI standards. Accredited committees include ASC X3 – Information Processing Systems; ASC X9 – Financial Services; and ASC X12 – Electronic Business Data Interchange. Table 15.6 lists selected ANSI encryption and banking security standards developed under X3 and X9.

ANSI X3.92: This standard specifies the DES algorithm, which ANSI standards refer to as the Data Encryption Algorithm (DEA). X3.92 is technically the same as FIPS 46.

ANSI X3.106: This standard specifies DES modes of operation, or DEA modes of operation as referred to in ANSI standards. X3.106 is technically the same as FIPS 81 (cf. ISO 8372). An appendix in FIPS 81 contains additional background information on the various modes.

(ii) ANSI banking security standards

ASC X9 subcommittee X9F develops information security standards for the financial services industry. Banking security standards include cryptographic and operational requirements, with a heavy emphasis on controls, audit, sound business practices, and interoperability. Among the working groups under X9F, most of the cryptographic work is in X9F1 (public key cryptography and cryptographic tools) and X9F3 (security in wholesale financial telecommunications).

ANSI #	Subject	Ref.
X3.92	data encryption algorithm (DEA)	[33]
X3.106	data encryption algorithm (DEA) modes	[34]
X9.8	PIN management and security	[35]
X9.9	message authentication (wholesale)	[36]
X9.17	key management (wholesale; symmetric)	[37]
X9.19	message authentication (retail)	[38]
X9.23	encryption of messages (wholesale)	[39]
X9.24	key management (retail)	[40]
X9.26	sign-on authentication (wholesale)	[41]
X9.28	multi-center key management (wholesale)	[42]
X9.30–1	digital signature algorithm (DSA)	[43]
X9.30–2	secure hash algorithm (SHA) for DSA	[44]
X9.31–1	RSA signature algorithm	[45]
X9.31–2	hashing algorithms for RSA	[46]
X9.42	key management using Diffie-Hellman	[47]
X9.45	attribute certificates and other controls	[49]
X9.52	triple DES and modes of operation	[50]
X9.55	certificate extensions (v3) and CRLs	[51]
X9.57	certificate management	[52]

Table 15.6: *ANSI encryption and banking security standards.*

ANSI X9.8: This standard addresses PIN management and security. It consists of ISO 9564 reproduced in its entirety, with clearly marked “X9 Notes” added where required to adapt the text for use as an ANSI X9 standard. A standard means for interchanging PIN data is specified. Annex A of 9564 (procedures for the approval of an encipherment algorithm) is included; the only currently specified approved algorithm is DES. Annex B (general principles for key management) is also retained from 9564, but noted as superseded by X9.24 (retail key management).

ANSI X9.9: This standard specifies a DES-based message authentication code (MAC) algorithm for wholesale banking as summarized below (cf. X9.19 for retail banking). If data is protected by both authentication and encryption mechanisms, a different key is required for each purpose. Message replay is precluded by use of date and message identifier fields. Appendix B includes sample MAC computations. X9.9 requires key management in accordance with ANSI X9.17, and also addresses implementation issues including coded character sets and representations, field delimiters, and message normalization (e.g., replacing carriage returns or line feeds by space characters, and multiple spaces by single spaces), and notes other practical concerns such as escape sequences beyond the scope of a MAC causing over-writing of authenticated data fields on display devices.

The X9.9 MAC algorithm may be implemented using either the cipher-block chaining (CBC) or 64-bit cipher feedback (CFB-64) mode, initialized to produce the same result (see Note 15.1). Final data blocks with fewer than 64 bits are left-justified and zero-bits are appended to complete the block before processing. The MAC result is specified to be the leftmost 32 bits of the final DES output. X9.9 states that the capability to generate 48-bit and 64-bit MAC values should also exist.

15.1 Note (*CBC-MAC and equivalent CFB-64 MAC*) For data blocks $D_1, \dots, D_t$ and a fixed MAC key K, equivalent MACs may be generated using either the CBC or 64-bit cipher feedback (CFB-64) modes. In the CBC case, the MAC C_t is defined by $C_i = E_K(D_i \oplus C_{i-1})$ for $1 \le i \le t$ and $C_0 = IV = 0$. For the CFB-64 case, let $O_i = E_K(I_i)$ be the output from the block encryption at stage i for $1 \le i \le t$, where $I_i = D_i \oplus O_{i-1}$ for $2 \le i \le t$ and $I_1 = D_1$ (the first 8 data bytes serve as IV). Note $O_t = C_t$ from above. (A block $D_{t+1} = 0$ may be introduced if the CFB implementation interface requires the final output O_t be XORed to a data block before release.)

ANSI X9.17: This standard, which was the basis for ISO 8732, specifies manual and automated methods (symmetric-based) for wholesale banking key management, including key establishment techniques and protection of keys in key management facilities. A key management hierarchy is defined consisting of manually-distributed key-encrypting keys, electronically-distributed key-encrypting keys, and electronically-distributed data or transaction keys for authentication or encryption. Key management techniques include the use of key counters, key offsetting, and key notarization. Key establishment settings include direct exchange between two nodes (point-to-point), and both key distribution centers (KDCs) and key translation centers (KTCs).

ANSI X9.19: This standard specifies a DES-based message authentication code (MAC) algorithm for retail banking (cf. X9.9 for wholesale banking). Implementation and other issues are addressed as per X9.9, and the MAC algorithm itself is essentially the same as X9.9, differing in that the MAC result is the leftmost m bits of the final 64-bit output, where m is to be specified by the application. An optional X9.19 procedure using a second key K' is specified for increased protection against exhaustive key determination: the (previously) final output is decrypted using K' and then re-encrypted under the original key. The resulting algorithm is widely referred to as the *retail MAC*; see Figure 9.6.

ANSI X9.23: This standard addresses message formatting and representation issues related to the use of DES encryption in wholesale banking transactions. These include field delimiting and padding, as well as filtering methods required to prevent ciphertext bit sequences from interfering with communications protocols when inadvertently interpreted as control characters (e.g., end-of-transmission).

ANSI X9.24: This standard, which motivated ISO 11568, specifies manual and automated methods for retail key management, addressing authentication and (DES-based) encryption of PINs, keys, and other data. Guidelines include protection requirements at various stages in the key management life cycle. Appendices provide additional information, including (Appendix D) methods providing unique per-transaction keys, updated after each transaction as a one-way function of the current key and transaction-specific details; and (Appendix E) how to derive a large number of different terminal keys (for distinct terminals) from a common base key, simplifying key management for servers which must communicate with all terminals. Such derived keys may be combined with the unique per-transaction key methods.

ANSI X9.26: This standard specifies two main classes of entity authentication mechanisms of use for access control. The first involves user passwords. The second involves cryptographic keys used in DES-based challenge-response protocols (e.g., a time-variant parameter challenge must be ECB-encrypted). The latter class is subdivided, on the basis of granularity, into user-unique and node-unique keys.

ANSI X9.28: This standard extends X9.17 to allow the distribution of keying material (using X9.17 protocols) between entities (subscriber nodes) which neither share a common key, nor share a key with a common central server (KDC or KTC). Two or more key centers form a *multiple-center group* to provide a more general key distribution service allowing the establishment of keying material between any two subscribers sharing a key with at least one center in the group. As there are no known or proposed implementations of this standard, it appears destined to be withdrawn from the ANSI suite.

ANSI X9.30: The first in a suite of ANSI public-key standards, X9.30–1 and X9.30–2 specify DSA and SHA for the financial services industry, as per FIPS 186 and FIPS 180, respectively.

ANSI X9.31: The (draft) standard X9.31–1 parallels X9.30–1, and specifies a signature mechanism based on an RSA signature algorithm, more specifically the ISO/IEC 9796 variant combined with a hashing algorithm. The (draft) standard X9.31–2 defines hash functions for use with Part 1, including MDC-2.

ANSI X9.42: This (draft) standard specifies several variations of unauthenticated Diffie-Hellman key agreement, providing shared symmetric keys for subsequent cryptographic use.

ANSI X9.45: This (draft) standard employs a particular type of attribute certificate (§13.4.2) called an *authorization certificate*, and other techniques from ANSI X9.57, to allow a party to determine whether a received message or signed document is authorized with respect to relevant rules or limits, e.g., as specified in the authorization certificate.

ANSI X9.52: This (draft) standard for encryption offers improvements over DES security by specifying a number of modes of operation for triple-DES encryption, including the four basic modes of ISO 8372, enhanced modes intended to provide additional protection against advanced cryptanalytic attacks, and message-interleaved and pipelined modes intended to allow increased throughput in multi-processor systems.

ANSI X9.55: This (draft) standard specifies extensions to the certificate definitions of ANSI X9.57 corresponding to, and aligned with, ISO certificate extensions for ITU-T X.509 Version 3 certificates (see page 660).

ANSI X9.57: This (draft) certificate management standard includes both technical specifications defining public-key certificates (based on ITU-T X.509) for electronic commerce, and business controls necessary to employ this technology. The initial version is defined for use with DSA certificates, in conjunction with ANSI X9.30–1.

(iii) ISO banking security standards

ISO banking security standards are developed under the ISO technical committee TC68 – Banking and Related Financial Services. TC68 subcommittees include TC68/SC2 (wholesale banking security) and TC68/SC6 (retail banking security and smart card security). Table 15.7 lists selected ISO banking security standards.

ISO #	Subject	Ref.
8730	message authentication – requirements (W)	[575]
8731–1	message authentication – CBC-MAC	[576]
8731–2	message authentication – MAA	[577]
8732	key management/symmetric (W)	[578]
9564	PIN management and security	[579]
9807	message authentication – requirements (R)	[581]
10126	message encipherment (W)	[582]
10202–7	key management for smart cards	[584]
11131	sign-on authentication	[585]
11166–1	key management/asymmetric – overview	[586]
11166–2	key management using RSA	[587]
11568	key management (R), in 6 parts	[588]

Table 15.7: *ISO banking security standards (W–wholesale; R–retail).*

ISO 8730: Together with ISO 8731, this wholesale banking standard for message authentication code (MAC) algorithms forms the international equivalent of ANSI X9.9. ISO 8730 is algorithm-independent, and specifies methods and requirements for the use of MACs including data formatting and representation issues, and a method by which specific algorithms are to be approved.

ISO 8731: ISO 8731–1 and 8731–2 specify particular MAC algorithms complementary to the companion standard ISO 8730. 8731–1 specifies a DES-based CBC-MAC with $m = 32$ (cf. ISO/IEC 9797). 8731–2 specifies the Message Authenticator Algorithm, MAA (Algorithm 9.68).

ISO 8732: This standard for key management in wholesale banking was derived from ANSI X9.17, and is its international equivalent.

ISO 9564: This standard, used as the basis for ANSI X9.8, specifies minimum measures for the management and security of Personal Identification Numbers (PINs). Part 1 specifies principles and techniques to protect against disclosure of PINs to unauthorized parties during the PIN life cycle. Part 2 specifies encipherment algorithms approved to protect PINs.

ISO 9807: This standard for message authentication in retail banking is analogous to ANSI X9.19 (cf. ISO 8730/8731–1 vs. ANSI X9.9), but does not address data representation issues, and names two approved algorithms in Annex A – the CBC-MAC of 8731–1 (allowing optional final processing as per X9.19), and the MAA of 8731-2.

ISO 10126: This multi-part standard is the international equivalent of X9.23 addressing confidentiality protection of (parts of) financial messages. ISO 10126–1 provides general principles; 10126–2 defines a specific algorithm – DES.

ISO 10202: This eight-part standard addresses security architecture issues for integrated circuit cards (chipcards) used for financial transactions. In particular, ISO 10202-7 specifies key management aspects.

ISO 11131: This standard for sign-on authentication is the international (non-DES specific) analogue of ANSI X9.26.

ISO 11166: This multi-part standard for banking key management specifies asymmetric techniques for distributing keys for symmetric algorithms. It was developed from ISO

8732, which uses symmetric techniques only. Part 1 specifies general principles, procedures, and formats, including background regarding key protection during its life cycle, certification of keying material, key distribution by either key exchange (e.g., Diffie-Hellman) or key transport, and cryptographic service messages. Further parts are intended to define approved algorithms for use with the procedures of Part 1. Part 2 specifies the RSA algorithm for both encipherment and digital signatures; RSA formatting differs from both ISO/IEC 9796 and PKCS #1.

ISO 11568: This multi-part standard addresses retail key management and life cycle issues. It originated from X9.24, but is generalized for international use (e.g., it is no longer DES-specific), and addresses both symmetric and public-key techniques.

15.3.3 International security architectures and frameworks

Table 15.8 lists selected ISO standards on security frameworks and architectures. Some of these are developed by SC21 (ISO/IEC JTC 1 SC21), which includes activities on Open Systems Interconnection (OSI) projects. The International Telecommunication Union (ITU) develops common-text specifications with JTC 1 for some standards in this area.

ISO #	Subject	Ref.
7498-2	OSI security architecture	[573]
9594-8	authentication framework (X.509)	[595]
10181	OSI security frameworks	[609]

Table 15.8: *ISO and ISO/IEC security architectures and frameworks.*

ISO 7498-2 (X.800): The OSI basic reference model of ISO 7498 defines a communications protocol stack with seven layers: application (layer 7), presentation (6), session (5), transport (4), network (3), data-link (2), and physical layers (1). ISO 7498-2 specifies the security architecture for the basic reference model, including the placement of security services and mechanisms within these layers. It also provides a general description of the basic OSI security services: authentication (peer-entity and data-origin); access control; data confidentiality; data integrity; and non-repudiation (with proof of origin, or with proof of delivery). Specific mechanisms are used to implement these services; for example, encipherment is a mechanism for providing confidentiality.

ISO/IEC 9594-8 (X.509): This standard is the same as ITU-T (formerly CCITT) Recommendation X.509. It defines both simple authentication techniques (based on passwords) and so-called strong authentication techniques (wherein secret values themselves are not revealed to the verifier). The strong techniques included are the two-pass and three-pass X.509 exchanges (see §12.5.2) based on digital signatures and the use of time-variant parameters. An implicit assumption is the use of an algorithm such as RSA which may serve as both an encryption and a signature mechanism; the specification may, however, be modified (e.g., to use DSA). The standard also specifies techniques, including X.509 certificates, for acquiring or distributing authentic public keys; and addresses cross-certificates, and the use of certificate chains (§13.6.2(i)).

ISO/IEC 10181 (X.810 through X.816): This specification is a series of security frameworks intended to provide context and background, consisting of the following parts: security frameworks overview (1); authentication framework (2); access control framework (3); non-repudiation framework (4); confidentiality framework (5); integrity framework (6); security audit and alarms framework (7).

15.3.4 U.S. government standards (FIPS)

Table 15.9 lists selected security-related Federal Information Processing Standards (FIPS) publications. These are developed under the National Institute of Standards and Technology (NIST), for use by U.S. federal government departments.

FIPS #	Subject	Ref.
FIPS 46–2	DES	[396]
FIPS 74	guidelines for using DES	[397]
FIPS 81	DES modes of operation	[398]
FIPS 112	password usage	[399]
FIPS 113	data authentication (CBC-MAC)	[400]
FIPS 140–1	cryptomodule security requirements	[401]
FIPS 171	key management using X9.17	[402]
FIPS 180–1	secure hash standard (SHA–1)	[404]
FIPS 185	key escrow (Clipper & SKIPJACK)	[405]
FIPS 186	digital signature standard (DSA)	[406]
FIPS 196	entity authentication (asymmetric)	[407]

Table 15.9: *Selected security-related U.S. FIPS Publications.*

FIPS 46: This standard specifies the DES algorithm (cf. ANSI X3.92).

FIPS 74: This standard provides guidelines for implementing and using DES.

FIPS 81: This standard specifies 4 basic DES modes of operation (cf. ANSI X3.106).

FIPS 112: This standard provides guidelines on password management and usage.

FIPS 113: This standard specifies the customary DES-based CBC-MAC algorithm (see ISO/IEC 9797), referring to it as the Data Authentication Algorithm (DAA). The MAC result is called a Data Authentication Code (DAC). The last data bock, if incomplete, is left-justified and zero-padded before processing; the result is the leftmost m output bits, where m is a multiple of 8, and $16 \leq m \leq 64$. Implementation may be either by the CBC mode with $IV = 0$, or CFB-64 mode with $IV = D_1$, the first data block (see Note 15.1). 7-bit ASCII-coded data to be authenticated by the DAA is preprocessed into 8-bit characters with leading bit 0.

FIPS 140–1: This standard specifies security requirements for the design and implementation of cryptographic modules for protecting (U.S. government) unclassified information, including hardware, firmware, software modules, and combinations thereof. Four grades of increasing security are specified as Levels 1 through 4, covering a wide range of security applications and environments. A FIPS 140–1 validation program is run by NIST to determine if cryptomodules meet the stated requirements.

FIPS 171: FIPS 171 specifies, for use by (U.S.) federal government departments, a subset of the key distribution techniques of ANSI X9.17. The objective of specifying a subset is to increase interoperability and decrease system costs.

FIPS 180 and 180–1: The hash algorithm specified in the original standard FIPS 180 is the Secure Hash Algorithm, SHA. A revised version was specified shortly thereafter in FIPS 180–1 (Algorithm 9.53), and denoted SHA–1. SHA–1 differs from SHA as noted in §9.8.

FIPS 185: This Escrowed Encryption Standard (EES) specifies the parameters and use of the SKIPJACK symmetric-key block cipher, and a method of creating Law Enforcement Access Fields (LEAFs) for use with the Clipper key escrow system (§13.8.3). The purpose is to allow wiretapping under lawful authorization. Internal details of the SKIPJACK algorithm are not publicly available, although its interface specification is (§13.8.3(i)).

FIPS 186: This standard is the Digital Signature Standard (DSS), which specifies the Digital Signature Algorithm (DSA). The hash function originally mandated for use with DSA is defined in FIPS 180 (SHA), which was superseded by FIPS 180–1 (SHA–1).

FIPS 196: This standard on entity authentication using asymmetric techniques was derived from the two-pass and three-pass random-number based mechanisms of ISO/IEC 9798-3. It includes additional expository and implementation details.

15.3.5 Internet standards and RFCs

Documents called *Requests for Comments* (RFCs) are official working notes of the Internet research and development community. A subset of these are specifications which are candidates for standardization within the community as Internet Standards.

The Internet Engineering Steering Group (IESG) of the Internet Engineering Task Force (IETF) is responsible for making recommendations regarding progression of "standards-track" specifications from Proposed Standard (PS) to Draft Standard (DS) to Standard (STD). RFCs may also correspond to the following types of documents: Experimental (E) protocols which may be part of early research efforts; Informational (I) protocols published for convenience of the community; and Historical (H) protocols which have been superseded, expired, or abandoned.

The E, I, and H categories are not on the standards track, and the IESG does not make recommendations on these. Less mature, less stable, or less widely circulated documents are typically available as an Internet-Draft (I-D); these are considered to be "work in progress", and should be cited as such.

RFC	Status	Subject	Ref.
1319	I	MD2 hash function	[1033]
1320	I	MD4 hash function	[1034]
1321	I	MD5 hash function	[1035]
1421	PS	PEM – encryption, authentication	[1036]
1422	PS	PEM – certificates, key management	[1037]
1423	PS	PEM – algorithms, modes, identifiers	[1038]
1424	PS	PEM – key certification and services	[1039]
1508	PS	Generic Security Service API (GSS-API)	[1040]
1510	PS	Kerberos V5 network authentication	[1041]
1828	PS	keyed MD5 (as a MAC)	[1044]
1847	PS	security multiparts for MIME	[1045]
1848	PS	MIME Object Security Services (MOSS)	[1046]
1938	PS	one-time password system	[1047]

Table 15.10: *Selected Internet RFCs (May 1996 status).*

Table 15.10 lists selected security-related Internet RFCs. The hashing algorithms MD2, MD4, and MD5 are specified in RFCs 1319-1321, respectively. The Internet Privacy-Enhanced Mail (PEM) specifications are given in RFCs 1421-1424.

The Generic Security Service Application Program Interface (GSS-API) of RFC 1508 is a high-level security API which isolates application code from implementation details; for example, the interface provides functions such as *sign* and *seal* (e.g., as opposed to "seal using a 32-bit DES CBC-MAC and this particular key"). Specific implementation mechanisms must be provided beneath GSS-API; options include Kerberos V5 as per RFC 1510 for symmetric-based techniques, and SPKM for public-key based techniques (see page 661).

RFC 1828 specifies a method for using keyed MD5 as a MAC (cf. §9.5.2). RFC 1848 defines MIME Object Security Services (MOSS), where MIME denotes Multipurpose Internet Mail Extensions. MOSS makes use of the RFC 1847 framework of multipart/signed and multipart/encrypted MIME messages, and facilitates encryption and signature services for MIME including key management based on asymmetric techniques. RFC 1938 specifies an authentication technique based on Lamport's one-time password scheme (Protocol 10.6).

15.3.6 De facto standards

Various security specifications arising through informal processes become de facto standards. This section mentions one such class of specifications: the PKCS suite.

PKCS specifications

A suite of specifications called *The Public-Key Cryptography Standards* (PKCS) has parts as listed in Table 15.11. The original PKCS #2 and PKCS #4 have been incorporated into PKCS #1. PKCS #11 is referred to as *CRYPTOKI*.

No.	PKCS title
1	RSA encryption standard
3	Diffie-Hellman key-agreement standard
5	Password-based encryption standard
6	Extended-certificate syntax standard
7	Cryptographic message syntax standard
8	Private-key information syntax standard
9	Selected attribute types
10	Certification request syntax standard
11	Cryptographic token interface standard

Table 15.11: *PKCS specifications.*

15.3.7 Ordering and acquiring standards

ISO and ISO/IEC standards may be obtained from (member body) national standards organizations such as ANSI, the British Standards Institution (BSI), and the Standards Council of Canada (SCC). To purchase standards directly from ISO, contact ISO Central Secretariat, Case postale 56, CH-1211 Geneva 20, Switzerland; telephone +41.22.749.01.11.

ANSI X9 standards are published by EDI Support Services Incorporated; to purchase standards, telephone 1-800-334-4912 (from within the USA) or +216-974-7650 (from outside the USA).

FIPS PUBS may be purchased from the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Road, Springfield, Virginia 22161 (USA); telephone +703-487-4650, fax +703-321-8547. To obtain copies of specifications of proposed (draft) FIPS, contact the Standards Processing Coordinator, National Institute of Standards and Technology, Technology Building, Room B–64, Gaithersburg, Maryland 20899 (USA); telephone +301-975-2816. Alternatively, consult URL `http://csrc.ncsl.nist.gov/`.

Internet RFCs and Internet-Drafts are available on-line via anonymous FTP from numerous ftp sites (e.g., `ds.internic.net`); further information can be obtained by sending an email message to `rfc-info@isi.edu` with the message body "help: ways_to_get_rfcs". RFCs are typically under the directory `rfc/` as `rfcXXXX.txt` (e.g. rfc1321.txt), and an RFC index is available as `rfc-index.txt`. RFCs can also be obtained via electronic mail by sending an email message to `rfc-info@isi.edu` whose body includes "Retrieve: RFC" and "Doc-ID: RFCnnnn" on separate lines.

The PKCS suite is published by RSA Laboratories, 100 Marine Parkway, Suite 500, Redwood City, California 94065-1031 (telephone +415-595-7703), and is available by anonymous FTP from `rsa.com` under the directory `pub/pkcs/`.

15.4 Notes and further references

§15.1

Levine [762] compiled a comprehensive list of American cryptographic patents issued between 1861 and 1981, citing patent number, name of principal inventor, date granted, and patent title; this provides an insightful perspective of the history of cryptography over this period. Kahn [648] discusses many patents in his historical tour, including many related to rotor machines (cf. Chapter 7). Contact information regarding the current assignees of some cryptographic patents may be found throughout the book of Schneier [1094].

Davies and Price [308] provide both general discussion of standards, and detailed technical discussion of selected standards. Preneel [1001] gives background on worldwide, European, and North American standardization organizations, and an overview of activities therein. Ford [414] provides a comprehensive overview of information security standards including extensive background information on various standardization processes and organizations, including technical committees ISO TC 68 and ISO/IEC JTC 1 and their subcommittees; ITU; ANSI; and national, regional, and international standardization bodies. For a more recent overview of security standards for open systems, see Fumy and Rietenspiess [432]. A status update of selected standards is also provided by Ford [415].

§15.2

One of the earliest and most important cryptographic patents was U.S. Patent No. 1,310,719 [1221] issued to Vernam on July 22 1919 for the *Vernam cipher* (cf. the one-time pad, Chapter 7; see also Kahn [648, p.401]). Two other patents by Vernam, titled "Ciphering device", were granted May 23 1922 (1,416,765) and January 8 1924 (1,479,846).

In consideration of ANSI making DES a standard, IBM made the DES patent of Ehrsam et al. (3,962,539) [363] available free of license fees in the U.S. when used to implement ANSI standards.

The first widespread published disclosure of public-key cryptography was through the conference paper of Diffie and Hellman [344], presented June 8 1976, fifteen months prior to the filing of the Hellman-Diffie-Merkle patent [551]. Merkle independently conceived the idea of deriving a secret key over a public channel in 1974 (see §12.10); his paper [849], first submitted to *Communications of the ACM* in 1975, was rejected several times before final publication in 1978. Meanwhile, the 1976 Diffie-Hellman conference paper introduced the concept of a digital signature as well as public-key cryptography and public-key authentication. Although Diffie and Hellman noted: "At present we have neither a proof that public key systems exist, nor a demonstration system", the existence of public-key systems was postulated, and three suggestions were offered supporting the general idea. The first involved matrix inversion, which is more difficult than multiplication by a factor $O(n)$ for $n \times n$ matrices; this offers a degree of security for very large n. The second involved compiling a function described in a high-level language into machine code; this makes it difficult to recover the original function. The third suggestion involved obscuring the input-output relationships between, e.g., 100 input and 100 output bits (wires) in an invertible hardware circuit originally implementing the identity mapping, by, e.g., inserting 4-by-4 bit invertible S-boxes into randomly selected sets of 4 wires; re-arranging the particular mappings of input lines into S-boxes then makes inverting the resulting circuit difficult.

The Hellman-Merkle patent [553] was filed sixteen months after the above Diffie-Hellman conference paper was presented. A major reason why the RSA patent [1059] took almost 6 years from application filing to issue date was so-called interference proceedings between it and some of the Stanford patents. The subject of the authentication trees patent of Merkle [848] is discussed in his thesis [851, p.126-131] and in the open literature [852, 853].

The signature technique of the ESIGN patent [952] is discussed in the literature by Okamoto [948]; see also Fujioka, Okamoto, and Miyaguchi [428]. The identification and signature technique of the Shamir-Fiat patent [1118] is described by Fiat and Shamir [395]. Regarding the Guillou-Quisquater patent [523], see Guillou and Quisquater [524]. The identification and signature schemes patented by Schnorr [1095] are discussed in the literature by Schnorr [1097, 1098]; the preprocessing scheme proposed therein, however, was shown to be insecure by de Rooij [314, 315].

In its announcement of the proposed FIPS for DSS (*Federal Register* vol.56 no.169, August 30 1991, 42980-42982), NIST noted its intent to make the DSA patent of Kravitz [711] available world-wide on a royalty-free basis. In a letter to the Director of the Computer System Laboratories at NIST dated October 30 1991, Schnorr stated that DSA infringed on Claim 6 of his patent (4,995,082). FIPS 186 itself (1994) states that "The Department of Commerce is not aware of any patents that would be infringed by this standard".

MDC-2 and MDC-4 [184] (see also Bosselaers and Preneel [178]) are discussed in §9.4.1. For further discussion of FEAL [1125], see §7.5. A patent on IDEA was originally filed in Switzerland and subsequently as a European patent [790], prior to being filed as a U.S. patent [791]; for literature references, see Chapter 7.

Related to the Matyas-Meyer-Brachtl patent [806] on control vectors, the October 7 1980 patent of Ehrsam et al. (4,227,253), "Cryptographic communication security for multiple domain networks", describes use of a master key and two variants obtained by inverting designated bits of the master key, equivalent to an XOR of the master with fixed mask values. Also related is the key notarization method of the patent by Smid and Branstad [1154], which controls which parties use a key, but not the uses. The key notarization technique is essentially identical – involving concatenation of various quantities (user identities), which are then XOR'd with a key-encryption key – but control vectors have broader functionality.

Fair cryptosystems [861, 862] are discussed in the literature by Micali [863]; but see also Kilian and Leighton [671], who remark on a critical weakness.

Interest in product cipher systems was stimulated by the product ciphers described in Shannon's 1949 paper [1121]. Meyer and Matyas [859] note that Lucifer was the name of the cryptographic system in which the product cipher of Feistel's patent (3,798,359) [385] was implemented, and from which the IBM team lead by Tuchman derived DES. The 1974 patent of Smith [1159] is also related to Lucifer. A second 1974 patent of Feistel [386] on a "step code ciphering system" was filed and issued with dates matching the Lucifer algorithm patent. Sorkin [1165] states that Lucifer is the subject of all three of these patents, plus a fourth: "Centralized verification system" (3,798,605) granted March 19 1974 to H. Feistel. Feistel gives a high-level background discussion on a first variation of Lucifer in his 1973 *Scientific American* article [387], which appeared prior to his 1974 patents being issued. A description of the second variation of Lucifer (which lead to the design of DES) is given by Sorkin [1165]; see also Biham and Shamir [138]

Related to the Massey-Omura [792] and Omura-Massey [956] patents is that of Onyszchuk, Mullin, and Vanstone [959]. It was filed May 30 1985 and issued May 17 1988 with no assignee listed. The patent teaches the construction of a multiplier for elements in $\mathbb{F}_{2^m}$, stated to be a significant improvement over the method of Omura-Massey. The patent also tabulates those values m, $2 \leq m \leq 2493$, for which so-called optimal normal bases exist; in these fields, the disclosed normal-basis multipliers for $\mathbb{F}_{2^m}$ are more efficient than in others. Shamir's three-pass protocol was first proposed by Shamir, as indicated by Konheim [705]. Massey [786] notes that Shamir also specifically proposed implementing the three-pass protocol using exponentiation as the ciphering operation, an idea later independently proposed by Omura (cf. §12.3 notes on page 535).

In contrast to the prime generation methods of Shawe-Taylor and Maurer (§4.4.4) which result in guaranteed primes, the prime generation method of the Hellman-Bach patent [550] uses probabilistic primality tests, and is related to that presented by Gordon at Eurocrypt in April of 1984 [514], and which appeared (dated April 26 1984) in the June 7 1984 issue (vol.20 no.12) of *Electronics Letters* [513].

The protocol patented by Goss [519], filed April 17 1989, combines exponentials by an XOR operation. An essentially identical protocol published in 1986 by Matsumoto, Takashima, and Imai [800] uses modular multiplication (cf. Protocol 12.53).

The exponentiation cipher of the Hellman-Pohlig patent [554] is discussed in the literature by Pohlig and Hellman [982]. The ciphers Khufu and Khafre [847] are similarly discussed by Merkle [856]; on-line/off-line digital signatures [864] by Even, Goldreich, and Micali [377, 378]; and the techniques of the patent on efficient exponentiation [203] are presented by Brickell et al. [204] (for more recent work, see Hong, Oh, and Yoon [561]).

A patent by Crandall (5,159,632) [286] includes twelve (12) claims on specific implementations of elliptic curves using primes p of special form (e.g., $p = 2^q - C$ for C small) allowing fast multiplication using shifts and adds alone (cf. Mohan and Adiga, 1985), and specific use of Fast Fourier Transforms (FFT) for optimized modular multiplication in this case. The patent, filed September 17 1991, was issued October 27 1992 and assigned to NeXT Computer, Inc. (Redwood City, California); see also its continuation-in-part, (5,271,061) [287]. Another patent in this area is the Miyaji-Tatebayashi patent (5,272,755) [888] filed June 26 1992, with priority data June 28 1991 (Japanese patent office). Issued December 21 1993, and assigned to the Matsushita Electric Industrial Co. (Osaka), it contains six (6) claims in the area of selecting elliptic curves over $\mathbb{F}_p$ whose order is precisely p. This covers a small subset of possible curves of this order over $\mathbb{F}_p$, and one particular method for selecting from among these; see also its continuation-in-part, (5,351,297) [889].

Regarding other block ciphers discussed in this book, a patent application has been filed for the RC5 cipher (§7.7.2). Adams [3] is the inventor for a patent on the CAST block cipher design procedure (see p.281); the assignee, Northern Telecom Limited (Montreal), will, however, make a CAST cipher available free of license fees.

The SEAL stream cipher (§6.4.1) of Coppersmith and Rogaway is also patented [281].

§15.3

A draft standard in development under the IEEE Microprocessor Standards Committee group is *IEEE P1363: Standard for RSA, Diffie-Hellman and related public-key cryptography*, which includes specifications for elliptic curve systems.

Theoretical justification for the redundancy scheme used in ISO/IEC 9796 is given by Guillou et al. [525]. The customary 5-year review of this standard in 1996 resulted in a title change and the creation of a second part. The original standard (with content unchanged) will be re-titled *Digital signature schemes giving message recovery – Part 1: Mechanisms using redundancy*. The second part, a working draft (WD) as of April 1996 titled *Part 2: Mechanisms using a hash function*, specifies mechanisms utilizing the idea that when a signature algorithm such as RSA is used with a hash function, and the RSA modulus (say 1024 bits) is much larger than a hash value (say 160 bits), the remaining bits may be used to carry message text which can be recovered upon signature verification. This *partial message recovery* mode of the signature algorithm decreases the amount of accompanying cleartext required, which is of interest in bandwidth or memory-limited applications, and those wherein the text being signed is relatively small.

The Registration Authority designated by ISO/IEC to maintain the register of cryptographic algorithms of ISO/IEC 9979 is the National Computer Centre, Oxford Road, Manchester, M1 7ED, United Kingdom (telephone +44-161-228-6333, fax +44-161-228-1636). Twelve algorithms were registered as of October 1995: BARAS, B-Crypt, CDMF, DES, FEAL, IDEA, LUC, MULTI2, RC2, RC4, SXAL/MBAL, and SKIPJACK. An alternative for obtaining unique algorithm identifiers is the *object identifier* (OID) and registration scheme of the Abstract Syntax Notation One (ASN.1) standard ISO/IEC 8824; for more information, see Ford [414, pp.478-480].

For a history of DES-related standards from an American perspective, including ANSI standards, see Smid and Branstad [1156]. ANSI X9.24, Annex C contains a convenient six-page summary of ANSI X9.17. A revision of X9.30–2:1993 is to specify FIPS 180–1 (SHA–1) in place of SHA. An ANSI standard in development, but currently "on hold" pending resolution of patent issues, is (draft) X9.44 [48], which specifies a key transport technique based on RSA. An enhanced mode of triple-DES encryption included in the draft ANSI X9.52 [50] is *cipher block chaining with output feedback masking*. The draft ANSI X9.57 [52] is intended for use with X9.30 and (draft) X9.31, although the initial version addresses X9.30 (DSA) certificates. ITU-T X.509 v3 certificates and certificate extensions to which ANSI X9.55 is aligned are discussed below. Both (draft) X9.45 and (draft) X9.55 may eventually be incorporated into X9.57. Related to attribute certificates, see Fischer [410] regarding electronic document authorization and related patents [408, 409].

The ISO 11568 retail key management project includes six parts [588, 589, 590, 591, 592, 593]. Among these, 11568-3 specifies the key life cycle for symmetric encryption algorithms; 11568–4 addresses key management techniques for public-key cryptosystems, including certificate management and (in Annex C) attribute certificates; and 11568–5 addresses key life cycle for public-key cryptosystems.

ISO/IEC 9594-8 (X.509) is one part of a series of specifications outlining directory services for Open Systems Interconnection (OSI) and other systems. The *Directory* is a logical database of information with directory entries arranged in a tree structure, the *Directory Information Tree* (DIT), as introduced in ISO/IEC 9594–1 (ITU-T Recommendation X.500) [594], which also provides an overview of directory services. For extension discussion, see Chapter 14 of Ford [414]. The 1988 version of X.509 (equivalent to ISO/IEC 9594-8:1990) was updated in 1993 [626] (equivalent to ISO/IEC 9594-8:1995). A 1995 technical corrigendum [627] added a certificate *extensions* field, yielding Version 3 (v3) certificates. Standard extensions for v3 certificates are defined in a further amendment [628] (see §13.9). The OSI security frameworks project is specified in seven parts of ISO 10181 [609, 610, 611, 612, 613, 614, 615].

FIPS 140–1 [401] supersedes FIPS 140, *General Security Requirements for Equipment Using the Data Encryption Standard* (formerly Federal Standard 1027, April 1982). Information on FS 1027 is provided by Davies and Price [308]. In May 1994, NIST announced a weakness in SHA [403], resulting from unpublished analysis carried out by the U.S. National Security Agency; the formal revision was published as FIPS 180–1 [404].

The PKCS standards, developed by industrial collaboration lead by RSA Laboratories (a Division of RSA Data Security Inc.), are widely used in practice, and periodically updated. PKCS #1,3,5,6,7,8,9,10 [1072] and PKCS #11 [1071] are currently available (e.g., from URL `http://www.rsa.com/`).

For an overview of Internet security standards, see Kent [667]. Linn's GSS-API (RFC 1508) [1040] is an API suitable for session-oriented applications. An analogous specification for store-and-forward applications is the IDUP-GSS-API (Independent Data Unit Protection GSS-API) interface. Implementation mechanisms which have been specified to plug in beneath GSS-API include a symmetric-key mechanism based on Kerberos (the Kerberos Version 5 GSS-API mechanism), and a public-key based mechanism SPKM (*Simple Public-Key Mechanism*). For an overview of these work-in-progress items under development in the Common Authentication Technologies (CAT) group of the IETF, see Adams [4].

Work-in-progress in the IP Security (IPSEC) working group of the IETF includes two items using Diffie-Hellman key exchange for session key establishment over the Internet – the Photuris protocol of Karn and Simpson, and the SKIP protocol of Aziz. Krawczyk [718] notes these and presents an alternative (SKEME).

MIME, specified in RFC 1521 [1042], is designed to facilitate multipart textual and non-textual mail, i.e., mail messages whose bodies may contain multiple objects of a variety of content types including non-ASCII text, multi-font text, and audio and image fragments. An alternative to the MOSS proposal of RFC 1848 [1046] is S/MIME [1191], which adds signature and/or encryption services to MIME messages, using PKCS specifications.

Many other standards, both formal and informal, have been developed or are undergoing development. A collection of cryptographic algorithms and protocols recommended for use in Europe is that resulting from the European RACE Integrity Primitives Evaluation (RIPE) project; see Bosselaers and Preneel [178]. Pretty Good Privacy (PGP) is a popular, widely available software package originally developed by Zimmermann [1272] (see Garfinkel [442] for additional perspective), currently employing RSA signatures, MD5 hashing, and IDEA encipherment.

Examples of pseudorandom number generators (PRNGs) which appear in U.S. standards include a DES-based PRNG in ANSI X9.17 (Appendix C), and two further methods in FIPS 186 (Appendix 3) based on both the Secure Hash Algorithm (SHA) and DES.