CHAPTER 8

Cryptography

Cipher Methods, Algorithms, Tools & Secure Protocols

Principles of Information Security

Whitman & Mattord — 6th Edition

Topic	Edition	Chapter
Cryptography	6th (2018)	8 of 12

LEARNING OBJECTIVES

- Define cryptology, cryptography, and cryptanalysis, and explain their roles in information security
- Describe the foundations of cryptology: plaintext, ciphertext, keys, key space, and work factor
- Explain cipher methods: substitution, transposition, XOR, Vernam, book, and running key ciphers
- Understand symmetric encryption algorithms: DES, 3DES, AES, Blowfish, RC4/5, and IDEA
- Understand asymmetric (public key) encryption: RSA, Diffie-Hellman, Elliptic Curve, and El Gamal
- Describe cryptographic hash functions: MD5, SHA-1, SHA-2, and their security status
- Explain cryptographic tools: digital signatures, digital certificates, PKI, and key management
- Identify protocols for secure communications: SSL/TLS, S/MIME, PGP, HTTPS, SSH, IPSec, and WPA

8.1 Foundations of Cryptology

The science of secret communication

Cryptology is the science of making and breaking secret codes. In information security, cryptography provides the mathematical foundation for confidentiality, integrity, authentication, and non-repudiation. Understanding both how cryptographic systems work and how they can be attacked is essential for any security professional.

8.1.1 Core Terminology

Term	Definition
Cryptology	The science encompassing both cryptography (making secret codes) and cryptanalysis (breaking secret codes)
Cryptography	The art of creating and using encryption systems to secure information from unauthorized access
Cryptanalysis	The process of deciphering ciphertext without the key — "code breaking"
Plaintext (Cleartext)	The original, readable message before encryption is applied
Ciphertext	The scrambled, unreadable output produced by encrypting plaintext
Cipher	The algorithm used to transform plaintext into ciphertext and back again
Key	A value (number or string) used by the cipher algorithm to control encryption and decryption
Key Space	The total set of possible key values for a given key length — 2^n for an n-bit key
Work Factor	The effort required for cryptanalysis to break a cipher; measures cryptographic strength
Encryption	The process of converting plaintext to ciphertext using a key and algorithm
Decryption	The reverse process: converting ciphertext back to plaintext using a key
Steganography	Hiding the existence of a message inside another medium (image, audio, text) — different from encryption

8.1.2 Historical Context

Cryptography has been used for thousands of years. Key historical milestones:

- **Caesar cipher (50 BC):** Julius Caesar shifted each letter in the alphabet by 3 positions. Simple substitution; broken by frequency analysis
- **Vigenère cipher (1553):** Polyalphabetic substitution using a keyword; resistant to simple frequency analysis
- **Enigma machine (WWII):** Mechanical electro-rotor cipher machine used by Nazi Germany; broken by Alan Turing at Bletchley Park
- **DES (1977):** US government first standardized encryption algorithm; 56-bit key; now considered broken
- **Public key cryptography (1976):** Diffie and Hellman introduced asymmetric cryptography; revolutionized key exchange
- **AES (2001):** Advanced Encryption Standard replaced DES as the US federal standard; 128/192/256-bit keys

8.1.3 Cryptographic Strength

Key Space	The range of all possible key values. A 64-bit key has a key space of 2^64 (about 18 quintillion possible keys). Larger key spaces require exponentially more effort to break by brute force — adding one bit doubles the key space.

Work Factor	The computational effort required to break a cipher. Expressed in time or operations. A cipher with a work factor of 2^128 operations is considered computationally infeasible with current technology.

Key Length	Possible Keys	Approximate Brute-Force Time
40-bit	~1 trillion (2^40)	Seconds with modern hardware (WEP used 40-bit RC4 — BROKEN)
56-bit	~72 quadrillion (2^56)	Hours to days (DES — considered BROKEN since 1999)
128-bit	~340 undecillion (2^128)	Billions of years with all current computing power
256-bit	~1.16 × 10^77 (2^256)	Computationally infeasible even with quantum computers (Grover halves to 128-bit)

EXAM TIP

Key length determines brute-force resistance. DES (56-bit) is BROKEN. AES-128 is the current minimum standard. AES-256 is quantum-resistant. Remember: adding 1 bit doubles the key space — so 57-bit is twice as strong as 56-bit.

8.2 Cipher Methods

Fundamental techniques for transforming plaintext

Before studying specific algorithms, it is essential to understand the fundamental methods that all ciphers use to transform plaintext into ciphertext. Most modern encryption algorithms combine several of these methods.

8.2.1 Substitution Ciphers

Substitution Cipher	A cipher that replaces each element of the plaintext with another element (character, number, or symbol) according to a fixed rule. The positions of elements are preserved but the identity is changed.

Substitution Type	Description	Example	Weakness
Monoalphabetic	Each plaintext letter maps to exactly one ciphertext letter throughout the entire message	Caesar cipher: A→D, B→E, C→F (shift by 3)	Vulnerable to frequency analysis — common letters remain identifiable
Polyalphabetic	Uses multiple substitution alphabets; the substitution changes based on position or keyword	Vigenère cipher: keyword determines which shift to apply at each position	Kasiski examination can determine key length; then

Substitution Type	Description	Example	Weakness
			frequency analysis breaks each alphabet
Homophonic	Maps each plaintext letter to multiple possible ciphertext symbols	A might map to {3, 17, 42, 95} — chosen randomly	Harder to break but still vulnerable to advanced cryptanalysis

8.2.2 Transposition Ciphers

Transposition (Permutation) Cipher	A cipher that rearranges the positions of plaintext characters without changing the characters themselves. The same letters appear in the ciphertext but in a scrambled order. The key determines the rearrangement pattern.

Common transposition methods:

- **Rail fence cipher:** Write plaintext diagonally across "rails" (rows), then read across each rail in order
- **Columnar transposition:** Write plaintext in rows of a fixed width; read off columns in a key-determined order
- **Double transposition:** Apply columnar transposition twice; significantly more secure than single transposition

COMBINED CIPHERS

Modern encryption algorithms combine both substitution and transposition to create product ciphers. DES and AES use multiple rounds of substitution-permutation networks (SPN) — each round applies substitution (S-boxes) and permutation (P-boxes) to the data.

8.2.3 XOR and Exclusive OR Operations

XOR (Exclusive OR)	A binary operation where the output is 1 if the inputs differ and 0 if they are the same. XOR is the fundamental operation in most modern symmetric encryption. Properties: A XOR A = 0; A XOR 0 = A; (A XOR B) XOR B = A (reversible with same key).

XOR ENCRYPTION

```
Ciphertext = Plaintext XOR Key     |     Plaintext = Ciphertext XOR Key
```

XOR with the same key encrypts and decrypts. This reversibility makes XOR ideal for stream ciphers. RC4, ChaCha20, and one-time pads all use XOR as their core operation.

8.2.4 Vernam Cipher and One-Time Pad

Vernam Cipher	A stream cipher invented by Gilbert Vernam (1917) that XORs plaintext bits with a key stream. When the key stream is truly random, as long as the message, and used only once, the result is the one-time pad — the only provably unbreakable encryption scheme.
One-Time Pad (OTP)	Theoretically perfect encryption: truly random key of same length as message, used only once. Information-theoretically secure — even with unlimited computing power, ciphertext reveals no information about plaintext. Impractical for general use due to key distribution challenges.

OTP CONDITIONS

The one-time pad is provably unbreakable ONLY if: (1) The key is truly random (not pseudorandom), (2) The key is at least as long as the message, (3) The key is used only once, (4) The key is kept completely secret. Violating any condition destroys security.

8.2.5 Book Cipher and Running Key Cipher

Book Cipher	A cipher where the key is derived from a specific book or document. The sender and receiver agree on the same text; page, line, and word numbers serve as the key. Security depends entirely on the secrecy of which book is used.
Running Key Cipher	Similar to a Vigenère cipher but uses a very long key (often a passage of text) that runs as long as the message. The long key avoids periodic repetition, making Kasiski examination ineffective. Security depends on the randomness and secrecy of the key text.

8.3 Cryptographic Algorithms

Symmetric and asymmetric encryption systems

Modern cryptographic algorithms fall into two major categories: symmetric (secret key) algorithms where the same key is used for both encryption and decryption, and asymmetric (public key) algorithms where mathematically related key pairs are used. Most secure systems use both in combination.

8.3.1 Symmetric Encryption

Symmetric Encryption (Secret Key Cryptography)	Encryption in which the SAME key is used for both encryption and decryption. Both parties must share the same secret key before communicating securely. Fast and efficient for bulk data encryption but requires a secure key exchange mechanism.

Symmetric encryption characteristics:

- **Speed:** 100–1,000× faster than asymmetric encryption; efficient for large volumes of data
- **Key distribution problem:** How to securely share the secret key before communicating? This is the fundamental challenge.
- **Scalability:** n users require n(n–1)/2 unique keys — scales poorly as the group grows
- **Block ciphers:** Process fixed-size blocks of data (64-bit or 128-bit blocks). DES, AES, Blowfish
- **Stream ciphers:** Encrypt data one bit or byte at a time, generating a keystream. RC4, ChaCha20

8.3.2 Symmetric Algorithm: DES

DES (Data Encryption Standard)	The first US federal encryption standard (FIPS 46, adopted 1977). 64-bit block cipher with a 56-bit effective key (8 bits used for parity). Uses a Feistel network with 16 rounds of substitution and permutation. Now considered BROKEN — a 56-bit key is too short for modern computers.
3DES (Triple DES / TDEA)	Applies DES three times with two or three different keys: C = E_K3(D_K2(E_K1(P))). Provides 112-bit (two-key) or 168-bit (three-key) equivalent security. Much slower than AES. NIST deprecated 3DES in 2017; disallowed after 2023 for most new applications.

DES WEAKNESS

In 1998, the EFF built "Deep Crack" for $250,000 and cracked DES in 56 hours. Today DES can be broken in minutes with consumer hardware. The fundamental flaw: a 56-bit key space of 2^56 (≈72 quadrillion) is too small for modern brute-force attacks.

8.3.3 Symmetric Algorithm: AES

AES (Advanced Encryption Standard)	The current US federal encryption standard (FIPS 197, adopted 2001). Based on the Rijndael algorithm. 128-bit block cipher with 128, 192, or 256-bit keys. Uses a Substitution-Permutation Network (SPN) with 10, 12, or 14 rounds respectively. Considered secure and efficient.

AES Variant	Key Length	Rounds	Security Level	Use Cases
AES-128	128 bits	10 rounds	Strong — standard for most applications	WPA2, TLS, disk encryption, general use
AES-192	192 bits	12 rounds	Higher security	Government, sensitive data
AES-256	256 bits	14 rounds	Highest; quantum-resistant	Top secret, long-term data, post-quantum preparedness

8.3.4 Other Symmetric Algorithms

Algorithm	Key Size	Block/Stream	Status	Notes
Blowfish	32–448 bits	Block (64-bit)	Legacy — not recommended	Bruce Schneier (1993); fast; patent-free; 64-bit block vulnerable to SWEET32 birthday attack
RC4	Variable	Stream	BROKEN — never use	Ron Rivest; used in WEP and early SSL; RC4 bias attacks (NOMORE); RFC 7465 prohibits in TLS
RC5	0–2040 bits	Block (variable)	Legacy	Parameterizable; precursor to RC6; academic interest
RC6	128–256 bits	Block (128-bit)	Legacy	AES finalist; based on RC5; not widely deployed
IDEA	128 bits	Block (64-bit)	Legacy — patent expired	Used in PGP 2.x; patent expired 2012; 64-bit block is a limitation
Skipjack	80 bits	Block (64-bit)	Legacy — classified	NSA-designed for Clipper chip; key escrow controversy

8.3.5 Asymmetric (Public Key) Encryption

Asymmetric Encryption (Public Key Cryptography)	Encryption using a mathematically related KEY PAIR: a public key (freely distributed) and a private key (kept secret). What one key encrypts, only the other can decrypt. Solves the key distribution problem but is much slower than symmetric encryption.

Two fundamental uses of asymmetric cryptography:

- **Encryption for confidentiality:** Sender encrypts with recipient's PUBLIC key → only recipient's PRIVATE key decrypts
- **Digital signatures for authentication:** Sender signs with own PRIVATE key → anyone with sender's PUBLIC key can verify

Aspect	Symmetric	Asymmetric
Key(s) used	One shared secret key	Key pair: public + private
Speed	Very fast (100–1000× faster)	Much slower (computationally expensive)
Key distribution	Secure channel required	Public key freely shared; no problem
Scalability	n(n–1)/2 keys for n users	Each user needs only 1 key pair
Primary use	Bulk data encryption	Key exchange, digital signatures, certificates
Examples	AES, 3DES	RSA, Diffie-Hellman, ECC, El Gamal

8.3.6 RSA (Rivest-Shamir-Adleman)

RSA	The most widely used public key algorithm. Security based on the mathematical difficulty of factoring the product of two large prime numbers. Created by Ron Rivest, Adi Shamir, and Leonard Adleman (1977). Used for key exchange, digital signatures, and encrypting session keys.

RSA Key Length	Security Equivalent	Status
1024-bit	~80-bit symmetric	WEAK — no longer recommended; do not use for new systems
2048-bit	~112-bit symmetric	Minimum acceptable; adequate through approximately 2030
3072-bit	~128-bit symmetric	Recommended — equivalent strength to AES-128
4096-bit	~140-bit symmetric	High security; slower; used for certificate authority root keys

8.3.7 Diffie-Hellman Key Exchange

Diffie-Hellman (DH) Key Exchange	A mathematical protocol (Whitfield Diffie and Martin Hellman, 1976) that allows two parties to establish a shared secret key over an insecure channel WITHOUT transmitting the key itself. Based on the discrete logarithm problem. This was the first practical solution to the key distribution problem.

DH LIMITATION

Diffie-Hellman provides KEY EXCHANGE only — it does not encrypt data. It also does not authenticate the parties. Without authentication, DH is vulnerable to man-in-the-middle attacks. In practice, DH is combined with digital certificates (as in TLS) to provide both key exchange and authentication.

8.3.8 Elliptic Curve Cryptography (ECC)

Elliptic Curve Cryptography (ECC)	Public key cryptography based on the algebraic structure of elliptic curves over finite fields. Provides equivalent security to RSA with significantly shorter key lengths. A 256-bit ECC key provides similar security to a 3072-bit RSA key. More efficient in processing and bandwidth — ideal for mobile devices and IoT.

ECC Key Length	RSA Equivalent
160 bits	~1024-bit RSA (now considered weak)
224 bits	~2048-bit RSA
256 bits	~3072-bit RSA (current standard recommendation)
384 bits	~7680-bit RSA

8.3.9 El Gamal

El Gamal Encryption	An asymmetric algorithm (Taher Elgamal, 1985) based on the discrete logarithm problem. Can be used for both encryption and digital signatures. The basis for the Digital Signature Algorithm (DSA) used in DSS (Digital Signature Standard). Produces ciphertext approximately twice the size of the plaintext.

8.4 Cryptographic Tools

Hash functions, digital signatures, certificates, and PKI

8.4.1 Hash Functions

Hash Function (Message Digest)	A one-way mathematical function that transforms an input of any length into a fixed-length output called a hash, digest, or fingerprint. Cryptographic hash functions are NOT encryption — they cannot be reversed. Used for integrity verification, password storage, and digital signatures.

Properties of a cryptographic hash function:

- Variable-length input → fixed-length output
- One-way: computationally infeasible to reverse (pre-image resistance)
- Collision resistance: computationally infeasible to find two inputs with the same hash
- Avalanche effect: small change in input produces drastically different hash
- Deterministic: same input always produces same output

Algorithm	Output Size	Status	Notes
MD5 (Message Digest 5)	128 bits	BROKEN — collision attacks (2004)	Do NOT use for security; acceptable only for non-security checksums
SHA-1 (Secure Hash Algorithm)	160 bits	BROKEN — deprecated (2017)	SHAttered attack demonstrated full collision; banned in certificates
SHA-256 (SHA-2 family)	256 bits	Secure — current standard	Used in TLS, code signing, Bitcoin, most modern security applications
SHA-384 (SHA-2 family)	384 bits	Secure — high security	NSA Suite B; high-security TLS configurations
SHA-512 (SHA-2 family)	512 bits	Secure — high security	Faster on 64-bit systems than SHA-256
SHA-3 (Keccak)	224–512 bits	Secure — modern	Different design from SHA-1/2 (sponge construction); 2015 NIST standard

HASH EXAM KEY

MD5 = BROKEN (1996 weaknesses, 2004 full collision). SHA-1 = BROKEN (2017). SHA-256 = current minimum standard. For password storage, use bcrypt, scrypt, or Argon2 — these are deliberately slow and include salting, making brute-force attacks impractical.

8.4.2 HMAC — Hash-Based Message Authentication Code

HMAC (Hash-Based Message Authentication Code)	A construction that uses a cryptographic hash function combined with a secret key to produce a Message Authentication Code (MAC). HMAC provides both data integrity AND authentication. Anyone with the shared key can verify the HMAC; without the key, the HMAC cannot be forged.

HMAC FORMULA

$$HMAC(K, m) = H[(K \oplus opad) \,||\, H[(K \oplus ipad) \,||\, m]]$$

Where K=key, m=message, H=hash function, opad/ipad=padding constants. Used in TLS (MAC for each record), IPSec (AH and ESP), and JWT authentication tokens.

8.4.3 Digital Signatures

Digital Signature	A cryptographic value computed from a message using the sender's private key. Provides message integrity, sender authentication, and non-repudiation. Verifiable by anyone with the sender's public key. Created by hashing the message and then encrypting the hash with the private key.

Digital signature process:

- SIGNING: Hash the message (H = SHA-256(message)) → Encrypt the hash with sender's private key → Attach signature to message
- VERIFYING: Decrypt signature with sender's public key → recover hash → independently hash the received message → compare hashes; if equal, signature is valid

SIGN vs ENCRYPT

Sign with PRIVATE key → verify with PUBLIC key. Encrypt with PUBLIC key → decrypt with PRIVATE key. These use the keys in OPPOSITE directions. Digital signatures provide authentication and non-repudiation; encryption provides confidentiality.

8.4.4 Digital Certificates and PKI

Digital Certificate	An electronic document that binds a public key to an identity (person, organization, server). Contains the subject's name, public key, issuer (Certificate Authority), validity period, and the CA's digital signature. X.509 is the standard format used in TLS/HTTPS.
Certificate Authority (CA)	A trusted third party that issues and digitally signs certificates, vouching for the identity of certificate holders. Root CAs are pre-installed in browsers and operating systems. The CA's signature on a certificate allows recipients to verify its authenticity without contacting the CA directly.
Public Key Infrastructure (PKI)	The complete system of policies, procedures, hardware, software, and people needed to create, manage, distribute, use, store, and revoke digital certificates. PKI enables trustworthy public key cryptography at scale.

PKI Component	Role
Root CA	Trust anchor; self-signed certificate; kept offline for security; signs intermediate CA certificates
Intermediate CA	Signs end-entity certificates; online; if compromised, only its certificates need revocation
Registration Authority (RA)	Handles identity verification and enrollment on behalf of the CA
Certificate Revocation List (CRL)	List of revoked certificate serial numbers published by the CA
OCSP (Online Certificate Status Protocol)	Real-time certificate validity checking; more efficient than downloading full CRL
Certificate Repository	LDAP directory storing certificates and CRLs for retrieval

8.4.5 Key Management

Cryptographic keys are valuable — the security of encrypted data depends entirely on keeping keys secret and well-managed. Key management encompasses the full lifecycle of cryptographic keys:

Phase	Activity	Best Practice
Generation	Create cryptographically strong keys using a CSPRNG (Cryptographically Secure Pseudo-Random Number Generator)	Use hardware random number generators for high-value keys; adequate entropy is critical
Distribution	Securely deliver keys to authorized parties	Use asymmetric encryption (key encapsulation) or Diffie-Hellman for symmetric key exchange
Storage	Protect keys at rest from unauthorized access	Hardware Security Modules (HSMs); encrypt keys at rest; separate keys from data they protect
Use	Apply keys according to policy; limit exposure	Access controls; audit key usage; never store keys in plaintext in application code
Rotation	Replace keys periodically or after suspected compromise	Rotate before expiry; document rotation schedule; re-encrypt data with new key
Destruction	Securely delete keys when no longer needed	Cryptographic erasure; multiple overwrites; HSM key zeroization

8.5 Protocols for Secure Communications

Applying cryptography in network security

Cryptographic algorithms are rarely used in isolation. Secure communication protocols combine multiple cryptographic primitives (symmetric encryption, asymmetric encryption, hash functions, digital signatures) into coherent systems that provide confidentiality, integrity, and authentication over networks.

8.5.1 SSL/TLS (Secure Sockets Layer / Transport Layer Security)

TLS (Transport Layer Security)	The successor to SSL, TLS is the primary protocol for securing communications over the Internet. Used in HTTPS, SMTPS, IMAPS, and many other protocols. TLS uses a HYBRID approach: asymmetric cryptography for key exchange and authentication, symmetric cryptography for bulk data encryption.

TLS Version	Status	Key Features
SSL 2.0	BROKEN — never use	Multiple critical vulnerabilities; deprecated 2011
SSL 3.0	BROKEN — POODLE attack	POODLE (Padding Oracle On Downgraded Legacy Encryption); deprecated 2015
TLS 1.0	Deprecated (RFC 8996)	BEAST, POODLE attacks; PCI DSS prohibited since 2018
TLS 1.1	Deprecated (RFC 8996)	Removed some weaknesses; still vulnerable; PCI DSS prohibited
TLS 1.2	Current minimum standard	Secure when configured correctly; avoid RC4, export ciphers, 3DES

TLS Version	Status	Key Features
TLS 1.3	Modern standard — preferred	Simplified handshake; forward secrecy mandatory; removed all weak ciphers; faster

TLS handshake process (TLS 1.2 simplified):

- 1. Client Hello: Client sends supported cipher suites and TLS version
- 2. Server Hello + Certificate: Server sends chosen cipher suite and its X.509 certificate
- 3. Key Exchange: Client verifies certificate; generates/exchanges key material (RSA or DHE/ECDHE)
- 4. Session Keys: Both sides derive symmetric session keys from the key exchange
- 5. Finished: Both sides send encrypted Finished messages; all subsequent data is symmetric-encrypted

8.5.2 S/MIME (Secure/Multipurpose Internet Mail Extensions)

S/MIME	A standard for public key encryption and signing of MIME data (email). S/MIME uses X.509 certificates from a PKI to authenticate senders and encrypt email content. Provides confidentiality (encryption), integrity (digital signature), authentication, and non-repudiation for email.

S/MIME capabilities:

- **Signed email:** Sender signs with private key; recipient verifies with sender's public key from certificate → authentication + non-repudiation
- **Encrypted email:** Sender encrypts a symmetric key with recipient's public key; message encrypted with symmetric key → confidentiality
- **Certificate requirement:** Both sender and recipient need valid X.509 certificates; requires PKI infrastructure

8.5.3 PGP (Pretty Good Privacy)

PGP (Pretty Good Privacy)	An encryption program providing cryptographic privacy and authentication for data communication, particularly email. Created by Phil Zimmermann (1991). PGP uses a "web of trust" model instead of hierarchical PKI — users vouch for each other's public keys. OpenPGP is the open standard; GnuPG (GPG) is the free implementation.

Feature	S/MIME	PGP/GPG
Trust Model	Hierarchical PKI (Certificate Authorities)	Web of Trust (users vouch for each other)
Certificate Format	X.509 (same as TLS)	OpenPGP key format
Enterprise Suitability	Better — centrally managed PKI	Decentralized — harder to manage at scale
Individual Use	Requires certificate from CA	Easy to generate your own key pair
Interoperability	Built into most email clients (Outlook, iOS)	Requires plugin or separate client (Thunderbird + Enigmail)

8.5.4 Secure Email Protocols at a Glance

Protocol	Port	Purpose
SMTPS (SMTP + TLS)	465 (implicit TLS) or 587 (STARTTLS)	Secure SMTP for sending email
IMAPS (IMAP + TLS)	993	Secure IMAP for retrieving email
POP3S (POP3 + TLS)	995	Secure POP3 for retrieving email
S/MIME	N/A (message-level)	Encrypts and signs individual email messages
PGP/GPG	N/A (message-level)	Encrypts and signs email using web-of-trust keys

8.5.5 HTTPS and Web Security

HTTPS (HTTP Secure)	HTTP transported over TLS. Uses port 443. The browser verifies the server's TLS certificate (issued by a trusted CA) to authenticate the website. All data exchanged is encrypted. HTTPS provides confidentiality, integrity, and server authentication. Modern browsers mark HTTP sites as "Not Secure."
HSTS (HTTP Strict Transport Security)	A security header that instructs browsers to only connect to a website using HTTPS, never HTTP. Prevents SSL stripping attacks (where an attacker downgrades HTTPS to HTTP). Browsers remember HSTS policy for a set period (max-age directive).

8.5.6 SSH (Secure Shell)

SSH (Secure Shell)	A cryptographic network protocol for secure remote login, command execution, and file transfers over untrusted networks. SSH replaced insecure protocols: Telnet (unencrypted terminal), FTP (unencrypted file transfer), and rsh/rlogin. SSH uses port 22.

SSH provides:

- **Secure remote administration:** Encrypted command-line access to remote systems
- **Secure file transfer:** SFTP (SSH File Transfer Protocol) and SCP (Secure Copy Protocol)
- **Port forwarding / tunneling:** Encrypt other protocols by tunneling them through SSH
- **Authentication options:** Password authentication (weaker) or public key authentication (stronger — recommended)

8.5.7 IPSec (Internet Protocol Security)

IPSec	A suite of protocols providing security at the IP layer (Layer 3). Used for VPNs and protecting IP communications. IPSec operates in two modes: Transport mode (protects payload only) and Tunnel mode (protects entire IP packet — creates new outer packet). Consists of AH and ESP protocols.

IPSec Component	Purpose	Details
AH (Authentication Header)	Authentication and integrity ONLY — no encryption	Authenticates the entire packet including outer IP header; incompatible with NAT
ESP (Encapsulating Security Payload)	Encryption, authentication, and integrity	Encrypts the payload; optional authentication; NAT-compatible (most common)
IKE (Internet Key Exchange)	Negotiate security associations and exchange keys	IKEv2 is current standard; uses Diffie-Hellman for key exchange
SA (Security Association)	One-way logical connection defining crypto parameters	Separate SAs for inbound and outbound; identified by SPI (Security Parameter Index)

IPSec Mode	What is Protected	Outer Header	Use Case
Transport Mode	IP payload (data) only	Original IP header visible	Host-to-host; protecting specific communications between two systems
Tunnel Mode	Entire original IP packet (header + payload)	New outer IP header added	Site-to-site VPN; gateway-to-gateway; entire packet hidden from eavesdroppers

8.5.8 Wireless Security Protocols

Protocol	Year	Encryption	Key Exchange	Status
WEP (Wired Equivalent Privacy)	1999	RC4 (40/104-bit)	Shared static key	COMPLETELY BROKEN — 24-bit IV reuse allows key recovery in minutes
WPA (Wi-Fi Protected Access)	2003	TKIP (RC4-based)	PSK or 802.1X	Deprecated — TKIP weaknesses; PSK vulnerable to offline dictionary attacks
WPA2 (Wi-Fi Protected Access 2)	2004	AES-CCMP (128-bit)	PSK or 802.1X	Current minimum — KRACK vulnerability (2017) patched; WPA2-Enterprise preferred
WPA3	2018	AES-GCMP (256-bit)	SAE (Dragonfly)	Recommended — SAE eliminates offline dictionary attacks; PMF mandatory

8.6 Chapter Summary & Exam Review

Key concepts, algorithms, and review questions

Algorithm Comparison — Quick Reference

Algorithm	Type	Key/Output	Status	Primary Use
DES	Symmetric block	56-bit key	BROKEN	Legacy — avoid
3DES	Symmetric block	112/168-bit	Deprecated	Being phased out
AES-128	Symmetric block	128-bit key	✓ Current standard	General encryption, TLS, disk
AES-256	Symmetric block	256-bit key	✓ Quantum-resistant	High security, long-term data
RC4	Stream	Variable	BROKEN	Banned in TLS; legacy WEP/WPA
RSA-2048	Asymmetric	2048-bit key	✓ Minimum	Key exchange, digital signatures
ECC P-256	Asymmetric	256-bit key	✓ Recommended	Mobile, IoT, TLS key exchange
Diffie-Hellman	Key exchange	2048+ bits	✓ (DHE for PFS)	Key exchange in TLS, IPSec
MD5	Hash	128-bit output	BROKEN	Checksums only — not security
SHA-1	Hash	160-bit output	BROKEN	Deprecated — do not use
SHA-256	Hash	256-bit output	✓ Current standard	TLS, code signing, integrity
HMAC-SHA256	MAC	256-bit output	✓ Standard	Message authentication in TLS, IPSec

Secure Protocol Port Reference

Protocol	Port	Description
HTTPS	443	HTTP over TLS — secure web browsing
SMTPS	465 / 587	SMTP over TLS — secure email sending
IMAPS	993	IMAP over TLS — secure email retrieval
POP3S	995	POP3 over TLS — secure email retrieval
FTPS	990 (implicit) / 21 (explicit)	FTP over TLS — secure file transfer
SSH / SFTP / SCP	22	Secure remote login and file transfer
LDAPS	636	LDAP over TLS — secure directory services
SNMPv3	161/162	SNMP with authentication and encryption
DNSSEC	53	DNS with digital signatures (not encryption, but integrity)

Key Terms Summary

Cryptology	Science of both creating (cryptography) and breaking (cryptanalysis) secret codes
Plaintext	Original readable message before encryption
Ciphertext	Encrypted, unreadable output
Key Space	All possible key values; 2^n for n-bit key; larger = more brute-force resistant
Work Factor	Computational effort to break a cipher; measures cryptographic strength
Substitution Cipher	Replace plaintext elements with other elements (Caesar, Vigenère)
Transposition Cipher	Rearrange positions of plaintext elements without changing them
XOR	Fundamental operation in modern ciphers; reversible with same key
One-Time Pad	Theoretically perfect encryption; random key same length as message; used once only
DES	56-bit symmetric block cipher; BROKEN; replaced by AES
3DES	Applies DES three times; 112/168-bit effective; deprecated; phasing out
AES	128-bit block, 128/192/256-bit key; NIST FIPS 197; current standard
Symmetric Encryption	Same key encrypts and decrypts; fast; key distribution problem
Asymmetric Encryption	Public/private key pair; slow; solves key distribution
RSA	Asymmetric; based on prime factorization; 2048-bit minimum
ECC	Asymmetric; elliptic curves; 256-bit ≈ 3072-bit RSA; efficient for mobile/IoT
Diffie-Hellman	Key exchange (not encryption); discrete logarithm; DH ⟶ DHE (with PFS)
El Gamal	Asymmetric; discrete logarithm; basis for DSA; used in PGP

MD5	128-bit hash; BROKEN (collision attacks 2004); checksums only
SHA-1	160-bit hash; BROKEN (SHAttered 2017); deprecated in all security contexts
SHA-256	256-bit hash; current minimum standard; TLS, certificates, Bitcoin
HMAC	Hash + secret key = integrity AND authentication (not just integrity)
Digital Signature	Private key signs, public key verifies; integrity + authentication + non-repudiation
Digital Certificate	X.509; binds public key to identity; signed by CA; used in TLS/HTTPS
PKI	Public Key Infrastructure; CA hierarchy; certificate lifecycle management
Certificate Authority	Trusted issuer of digital certificates; root CA (offline) → intermediate CA → end-entity
CRL	Certificate Revocation List; list of revoked certificate serial numbers
TLS	Transport Layer Security; HTTPS uses TLS; v1.2 minimum; v1.3 preferred
S/MIME	Email encryption/signing using X.509 certificates and PKI
PGP	Email encryption/signing using web-of-trust model; OpenPGP standard
SSH	Secure Shell; port 22; replaces Telnet; remote admin + SFTP
IPSec AH	Authentication Header: integrity/auth only; no encryption; incompatible with NAT
IPSec ESP	Encapsulating Security Payload: encryption + auth; most common; NAT-compatible
Transport Mode	IPSec: encrypts payload only; original IP header visible; host-to-host
Tunnel Mode	IPSec: encrypts entire packet; new outer header; site-to-site VPN
WEP	BROKEN wireless security; 24-bit IV reuse; cracked in minutes

WPA2	Current minimum wireless standard; AES-CCMP; KRACK (patched)
WPA3	Best wireless standard; SAE eliminates offline dictionary attacks; PMF mandatory

Review Questions

#	Question	Answer / Guidance
Q1	What is the difference between cryptography and cryptanalysis?	Cryptography: the science of creating encryption systems to secure information. Cryptanalysis: the science of breaking encryption systems (deciphering without the key). Together they form cryptology.
Q2	Why is DES considered broken? What replaced it?	DES uses only a 56-bit key (2^56 possible keys), which can be exhausted by brute force in hours with modern hardware. AES (128/192/256-bit keys) replaced DES as the NIST standard in 2001.
Q3	What is the key distribution problem and how does asymmetric cryptography solve it?	Symmetric encryption requires securely sharing a secret key before communication — but how do you share the key without a secure channel? Asymmetric cryptography solves this by using a public key (freely distributed) for encryption, requiring only the private key to remain secret.
Q4	When signing a document, which key do you use? When verifying?	Sign with your PRIVATE key (only you can sign). Verify with the signer's PUBLIC key (anyone can verify). Remember: signing and encryption use keys in opposite directions.
Q5	What is the difference between IPSec Transport mode and Tunnel mode?	Transport mode: encrypts the IP payload only; original IP header remains visible; used for host-to-host. Tunnel mode: encrypts the entire original IP packet and adds a new outer header; used for site-to-site VPN (entire packet is hidden).
Q6	What is the difference between AH and ESP in IPSec?	AH (Authentication Header): provides authentication and integrity ONLY — no encryption; incompatible with NAT. ESP (Encapsulating Security Payload): provides encryption plus optional authentication; NAT-compatible; the most commonly used IPSec protocol.
Q7	Why is SHA-1 no longer acceptable for security purposes?	The SHAttered attack (2017) demonstrated a practical collision attack — two different documents with the same SHA-1 hash. Collisions make digital signatures unreliable (attacker could forge documents). SHA-256 or higher is now required.
Q8	What makes the One-Time Pad theoretically unbreakable?	The OTP is information-theoretically secure because: (1) the key is truly random, (2) the key is the same length as the message, (3) the key is never reused, and (4) the key is kept secret. Every possible plaintext is equally likely, so ciphertext reveals nothing about the message.

FINAL EXAM TIP

Cipher methods: Substitution (replace elements) + Transposition (rearrange positions). Modern ciphers combine both.
DES = BROKEN (56-bit); 3DES = deprecated; AES-128 = current standard; AES-256 = quantum-resistant.
Symmetric = same key, fast, key distribution problem. Asymmetric = key pair, slow, solves key distribution.
Sign with PRIVATE key → verify with PUBLIC key. Encrypt with PUBLIC key → decrypt with PRIVATE key.
MD5 = BROKEN (collisions). SHA-1 = BROKEN (SHAttered). SHA-256 = current minimum. HMAC = hash + key.
Diffie-Hellman = key exchange ONLY (not encryption). AH = auth only (no encryption). ESP = encryption + auth.
IPSec transport mode = payload only (host-to-host). Tunnel mode = entire packet (site-to-site VPN).
TLS 1.0/1.1 = deprecated. TLS 1.2 = minimum. TLS 1.3 = best. WEP = BROKEN. WPA3 = best wireless.

IPS

PKI: Root CA (offline) → Intermediate CA → End-entity cert. CRL = revocation list. OCSP = real-time check.
PGP uses web-of-trust. S/MIME uses X.509 PKI. SSH port 22 (replaces Telnet). HTTPS port 443.