

Pick by what you need

- Confidentiality only → AEAD cipher.
- Integrity, shared secret → HMAC.
- Integrity + non-repudiation → signature.
- Password storage → slow KDF (*not* a cipher, not a fast hash).
- Key agreement → ephemeral (EC)DH.

Hashes

- **Use:** SHA-256, SHA-512, SHA-3, BLAKE2/3.
- **Broken (collisions):** MD5, SHA-1 — never for signatures/certs.
- Properties: pre-image, 2nd pre-image, collision. Collisions fall first, at $2^{n/2}$.
- A hash proves *corruption*, not *tampering*, unless it arrives over an independent, trusted path.

Password KDFs

- **Use:** Argon2id, scrypt, bcrypt, PBKDF2 (high iterations).
- Per-user **salt** (stored in the clear); optional secret **pepper** (stored apart).
- Tune cost so one verification ≈ 100 ms.
- Never a plain fast hash — GPUs do billions/sec.

Symmetric encryption

- **Cipher:** AES (128-bit block) or ChaCha20.
- **Mode:** AEAD — AES-GCM or ChaCha20-Poly1305. *Default to this.*
- **Never** ECB (structure survives). CBC needs a random IV and has padding-oracle risk.

- **Never** reuse a (key, nonce) pair — catastrophic for stream/CTR modes.

Asymmetric

- **RSA:** needs OAEP (encrypt) / PSS (sign) padding. Textbook RSA is unsafe.
- **ECC:** Ed25519 (sign), X25519 (key exchange) — smaller, faster, preferred for new work.
- Slow → used to wrap a symmetric session key (hybrid).
- Unauthenticated DH is trivially MITM'd — authenticate it.

PKI & TLS

- Certificate binds a public key to a name, signed by a CA.
- Validate: chain to trusted root, signatures verify, in date, name matches (SAN), not revoked.
- TLS 1.3: 1-RTT, mandatory forward secrecy, no static-RSA, no CBC, no compression.
- Certificate Transparency logs every issuance publicly.

Post-quantum

- Shor breaks RSA/DH/ECC; Grover only halves symmetric strength (AES-256 fine).
- NIST: ML-KEM (Kyber), ML-DSA (Dilithium), SLH-DSA (SPHINCS+).
- Deploy *hybrid*; migrate long-lived secrets first (harvest-now-decrypt-later).

Rules of thumb

- Do not implement primitives — use a reviewed library.
- Kerckhoffs: public algorithm, secret key.
- The key leaks from Git history, not from AES.

Injection (Ch 12) — one bug

Attacker data reaches an interpreter as *instructions*.

- XSS — browser HTML parser.
- SQLi — database.
- Command injection — shell.
- Path traversal — file system.
- XXE / SSTI / Log4Shell — XML parser / template engine / logger.
- **Fix:** validate on input (allowlist), encode on output by context, parameterise.

Memory safety (Ch 11)

- Stack smashing — overwrite the return address.
- Function-pointer / vtable / GOT overwrite.
- Format string (%n) — arbitrary write.
- Integer over/underflow → bad allocation size.
- **Fix:** memory-safe language; else canary + NX + ASLR + CFI.

Network (Ch 8), by OSI layer

- L2 — ARP spoof, MAC flood, rogue AP.
- L3 — IP spoof, BGP hijack.
- L4 — SYN flood, session hijack, port scan.
- L7 — DNS poisoning, HTTP attacks.
- Amplification (DNS/NTP/memcached) for volumetric DDoS.

Authentication (Ch 2)

- Credential stuffing, brute force, rainbow tables.
- Phishing, MFA fatigue, SIM swap (SMS OTP).

- **Fix:** passkeys/FIDO2, rate limiting, slow salted hashes.

Web session (Ch 12)

- CSRF — forged request, real session. Fix: SameSite, anti-CSRF token.
- XSS defeats CSRF defences — fix XSS first.

Malware (Ch 14)

- Virus / worm / trojan / ransomware / rootkit / botnet / APT.
- Polymorphic + packed → signatures fail.
- Lifecycle: recon, deliver, exploit, persist, C2, act.

Cloud (Ch 15)

- Public bucket / over-broad policy — top breach cause.
- SSRF → instance metadata → role creds.
- IAM wildcards, long-lived keys, escalation via PassRole.
- Privileged container → host (shared kernel).

Physical / social (Ch 2, 13)

- Tailgating, shoulder surf, evil maid, cold boot.
- BadUSB, RFID clone, juice jacking.
- Pretexting, baiting, vishing.

AI/ML (Ch 10)

- Evasion (adversarial examples), poisoning/backdoor.
- Model extraction, membership inference.
- Prompt injection (direct + indirect); the “lethal trifecta”.

Golden rules

- Preserve first, work on a *copy*.
- Hash at acquisition; re-verify at every handover.
- Collect most-volatile first (RAM before disk).
- Document every command — chain of custody.
- Use a hardware **write blocker**.

Imaging

```
dd if=/dev/sdb of=ev.img bs=4M \  
    conv=noerror,sync status=progress  
dcflddd if=/dev/sdb of=ev.img \  
    hash=sha256 hashlog=ev.hash  
sha256sum /dev/sdb ev.img # MUST match
```

Partition / file recovery

```
testdisk ev.img      # recover partitions  
photorec ev.img     # carve by signature  
mmls ev.img         # partition layout  
fls -r -o 2048 ev.img # list files
```

Metadata / EXIF

```
exiftool photo.jpg      # all metadata  
exiftool -gps* photo.jpg # GPS only  
identify -verbose img.jpg # ImageMagick  
stat file               # MAC times
```

Timeline

```
log2timeline.py out.plaso src/  
psort.py -o l2tcsv out.plaso  
# Normalise everything to UTC first.
```

Memory

```
# Acquire BEFORE shutdown.  
vol -f mem.raw windows.pslist  
vol -f mem.raw windows.netscan  
vol -f mem.raw windows.malfind
```

Static malware triage (safe)

```
file sample; strings -n 8 sample  
sha256sum sample      # reputation lookup  
# entropy up -> packed/encrypted  
# Never execute outside an isolated VM.
```

Watch for anti-forensics

- Timestomping — compare \$STANDARD_INFO vs \$FILE_NAME on NTFS.
- Cleared/truncated logs (the *absence* is evidence).
- Secure-wipe / full-disk encryption.
- Living off the land (signed system binaries).

Network Scanning & Capture Flags

Nmap — host & port discovery

```
nmap -sn 10.0.0.0/24 # ping sweep
nmap -sS -p- TARGET # SYN, all ports
nmap -sU --top-ports 50 TARGET # UDP
nmap -sV -sC TARGET # versions+scripts
nmap -O TARGET # OS fingerprint
nmap -A TARGET # aggressive (all)
nmap -T2 TARGET # slower, quieter
nmap -oA out TARGET # save all formats
```

tcpdump — capture

```
tcpdump -i eth0 -nn -w cap.pcap
tcpdump -i eth0 host 10.0.0.5
tcpdump -i eth0 port 53 # DNS
tcpdump -i eth0 'tcp[13]&2!=0' # SYNs
tcpdump -r cap.pcap -A # read+ASCII
```

Reconnaissance (passive)

```
whois example.com
dig example.com ANY
dig +short mx example.com
```

```
host -t txt example.com
# CT logs, Shodan: no packets to target.
```

Common ports

- 22 SSH 23 Telnet 25 SMTP
- 53 DNS 80 HTTP 443 HTTPS
- 110 POP3 143 IMAP 389 LDAP
- 445 SMB 3306 MySQL 3389 RDP
- 5432 Postgres 6379 Redis 9200 Elastic

Scan types (nmap)

- -sS SYN (half-open).
- -sT full connect (no raw sockets).
- -sU UDP (slow; often forgotten).
- -sn no port scan (discovery only).
- -Pn skip host discovery (assume up).

Authorisation

Scan only hosts you own or are authorised in *writing* to test. Unauthorised scanning is a criminal offence regardless of outcome.