

Lesson 11 — Security Fundamentals

Attacks, network-layer defenses, compute security, and the defense-in-depth model.

IT Landscape

Module 4

Why security at all?

Every system you've learned about in this course can be turned against its users.

- **Data theft** — emails, addresses, passwords, payment info.
- **Ransomware** — your files encrypted, payment demanded.
- **Account takeover** — someone logs in as you.
- **Fraud** — fake invoices, wire transfers, refund scams.
- **Espionage** — state actors and corporate spies.

Global cost of cybercrime *as of 2026*: measured in **trillions of dollars annually** (per Cybersecurity Ventures' running estimate).

Three intuitions to take away

- **1. Attackers go for the cheapest path** — not the most technically impressive one.
- **2. Humans, not algorithms, are the most-attacked surface** — phishing beats zero-days.
- **3. Security is layered** — no single fix is enough. *Defense in depth* wins.

If you remember nothing else, remember these three.

Attack 1 — Phishing (the cheapest path)

Trick a human into giving you credentials or running your code.

Still the **#1 initial-access method** in 2026 across every major incident report (Verizon DBIR, Mandiant M-Trends).

What a phishing email looks like:

```
From: support@arnazon.com    <-- look closely: arn, not ama
Subject: Your account will be locked in 24 hours
Click here to verify: http://arnazon-security.co/verify-now
```

*You've seen one of these. Probably this week. **Spear-phishing** (targeted) is now AI-generated and indistinguishable from a colleague's tone.*

Attack 2 — Credential stuffing & password reuse

- Attackers buy databases of **leaked username/password pairs** — billions exist.
- They try those pairs against **every service** — banks, email, SaaS.
- Because most people **reuse passwords**, a 1% hit rate is enormous.

Mitigation: unique passwords per site (use a **password manager**) plus **MFA** (multi-factor auth) everywhere.

Attack 3 — Supply-chain attacks

Why attack you, when I can attack the library you depend on?

Famous cases:

- **SolarWinds** (2020) — trojaned update reached ~18,000 organizations.
- **Log4Shell** (2021) — one logging library, internet-wide panic.
- **xz-utils backdoor** (2024) — caught hours from landing in every Linux distro.
- Various **npm** / **PyPI** package compromises ongoing in 2026.

A modern app pulls in hundreds of dependencies; any one of them is a foothold.

Attack 4 — Ransomware

Encrypt the victim's data, demand payment for the key.

- As of 2026 still one of the most **disruptive** attack categories — average ransom and average downtime cost both up year-on-year (per IBM X-Force and the annual Sophos *State of Ransomware* report).
 - High-impact targets: **hospitals, municipalities, manufacturing, schools**.
 - Modern ransomware is usually **double-extortion**: encrypt AND leak the data publicly if unpaid.
-

Attack 5 — Web app attacks (the OWASP Top 10)

The **OWASP Top 10** is the long-running canonical list of the most common web-application vulnerabilities.

Selected entries (2021 edition, current heading into 2026's refresh):

- **Broken access control** — #1 for years now.
- **Cryptographic failures** — weak or misused encryption.
- **Injection** (incl. **SQL injection**) — ranked highly for 25 years.
- **Vulnerable & outdated components** — unpatched libraries.
- **Security logging & monitoring failures** — can't see what you're not logging.

You don't need to memorize — you do need to know this list exists.

Defense 1 — Identity & access (the new perimeter)

- **MFA** (multi-factor auth) on everything important — password + a second factor.
- **SSO** (single sign-on) — reduces password proliferation.
- **Least privilege** — give each account only what it strictly needs.
- **Zero trust** — don't trust by network location; verify every request by identity.

Identity is the modern firewall.

Defense 2 — Patching & updates

- Most successful breaches exploit vulnerabilities **with patches already available**.
- **"Patch Tuesday"** (Microsoft's monthly cadence) and the **CVE** registry are how vulnerabilities get tracked publicly.
- Severity is graded by the **CVSS** score (0–10).
- **CISA's KEV** catalog lists vulnerabilities known to be actively exploited — patch those first.

Auto-update where possible; prioritize the rest by severity and known-exploited status.

Defense 3 — Backups, encryption, monitoring

- **Backups** — offline, immutable, *tested*. Most ransomware victims discover their backups didn't work **during** the incident.
- **Encryption at rest and in transit** — HTTPS, TLS 1.3, disk encryption (BitLocker, FileVault, LUKS).
- **Monitoring / SIEM** — detect intrusions in hours, not months.

*Median attacker **dwell time** in 2026 is still measured in **weeks** — per Mandiant M-Trends.*

Defense at the network layer — the first line

- **Firewalls** — perimeter (at the network edge) and host-based (on the machine itself). Both matter.
- **Network segmentation** — split the network into zones (DMZ, internal segments) so a breach in one zone can't freely reach another.
- **NAC** (Network Access Control) — devices must meet health checks before joining the network.
- **IDS / IPS** (Intrusion Detection / Prevention) — watch traffic patterns for known-bad behavior; IDS alerts, IPS blocks in real time. Open-source examples: Snort, Suricata.
- **VPN patch discipline** — unpatched remote-access VPNs are a top ransomware entry point every year.
- **Encrypted DNS (DoH / DoT)** — prevents eavesdropping on and forging DNS lookups.

Network controls don't replace identity — they're the outer ring on the defense onion.

Security in computing & virtualization — shared compute is shared risk

- **Hypervisor isolation** — VMs are meant to be isolated from each other; real cracks existed (VENOM 2015, L1TF/Foreshadow 2018). Hypervisor CVEs are treated as emergencies.
- **Container escapes** — containers share the host kernel; a kernel bug or misconfiguration can let a container break out to the host. Good hardening (seccomp, non-root, SELinux) makes this very hard in practice.
- **Secrets management** — never commit API keys, passwords, or certificates to version control. Bots scan public repos in minutes. Store secrets in Vault / AWS Secrets Manager / Azure Key Vault; inject at runtime.
- **RBAC at the orchestrator** — in Kubernetes, control who can `kubectl exec` into pods, who can read secrets, who can deploy workloads. Wildcard permissions are a common audit finding.

Most compute-layer breaches in 2026 are secrets leaks and misconfigured RBAC — not spectacular VM escapes.

Who does security in a company?

- **Security engineers** — build defenses (IAM, hardening, scanners).
- **SOC analysts** — Security Operations Center; watch the alerts in real time.
- **Incident responders** — handle a breach end-to-end.
- **Penetration testers / red teams** — paid to attack you, before real attackers do.
- **CISO** — Chief Information Security Officer; owns the program.

*And in 2026: **every developer** has security responsibilities — **DevSecOps**.*

Recap

- Attackers take the **cheapest path** — usually a human.
- Humans are the **most-attacked surface** — phishing still wins.
- Defense is **layered**: identity, **network controls** (firewalls, IDS/IPS, segmentation), **compute hygiene** (secrets management, RBAC), patching, encryption, monitoring.
- The **network is the first line of defense** — and shared compute means shared risk.

Coming up: Module 4 — Lesson 12, the AI era begins.

