Security Cheat Sheet

Security · Arun R Kaushik · 2026-06-15

OWASP Top 10, security headers, hashing vs encryption, and TLS basics—condensed for engineers and reviewers.

OWASP Top 10 (2021)

1. Broken Access Control
2. Cryptographic Failures
3. Injection
4. Insecure Design
5. Security Misconfiguration
6. Vulnerable and Outdated Components
7. Identification and Authentication Failures
8. Software and Data Integrity Failures
9. Security Logging and Monitoring Failures
10. Server-Side Request Forgery (SSRF)

Essential HTTP Security Headers

Header	Purpose
`Strict-Transport-Security`	Forces HTTPS, prevents downgrade attacks
`Content-Security-Policy`	Restricts sources for scripts, styles, etc.
`X-Content-Type-Options: nosniff`	Stops MIME-type sniffing
`X-Frame-Options: DENY`	Mitigates clickjacking
`Referrer-Policy`	Controls referrer info sent with requests
`Permissions-Policy`	Restricts browser feature access

Hashing vs Encryption vs Encoding

	Reversible?	Purpose
Hashing	No	Integrity checks, password storage
Encryption	Yes (with key)	Confidentiality
Encoding	Yes (no key)	Data representation (e.g. Base64)

Password storage: use `bcrypt`, `scrypt`, or `argon2`—never plain SHA-256/MD5.

Common Hash Algorithms

Algorithm	Output Size	Status
MD5	128-bit	Broken—do not use
SHA-1	160-bit	Deprecated
SHA-256	256-bit	Current standard
SHA-3	variable	Modern alternative

TLS Handshake (TLS 1.3, simplified)

1. Client → Server: `ClientHello` (supported ciphers, key share)
2. Server → Client: `ServerHello`, certificate, key share
3. Both derive session keys; encrypted application data begins

Symmetric vs Asymmetric Encryption

	Symmetric	Asymmetric
Keys	Single shared key	Public/private key pair
Speed	Fast	Slower
Examples	AES, ChaCha20	RSA, ECC
Typical use	Bulk data encryption	Key exchange, signatures

Quick Checks

```
# Check TLS certificate details
openssl s_client -connect example.com:443 -servername example.com

# Generate a SHA-256 hash
sha256sum file.txt

# Check open ports on a host
nmap -sV example.com

# Decode a JWT (header/payload only)
echo "<jwt-part>" | base64 -d
```

Defense-in-Depth Mnemonic

P-D-R-D: Prevent, Detect, Respond, Deter—layer controls so no single failure leads to a breach.