Interview Questions – Security Focused

1️⃣ How would you design a secure authentication system for a distributed application?

Answer:

- Use **OAuth 2.0 / OpenID Connect** for delegation and identity federation.
- **Authentication Server (e.g., Auth0, custom)** issues **access and refresh tokens** (usually JWT).
- Secure **token transmission via HTTPS**.
- Implement **token expiration** (~15 mins) and **refresh tokens** (stored securely with short TTL).
- Use **stateless token validation** (JWT with HMAC or RSA signature).
- Store **minimal session state** on client or via encrypted HTTP-only cookies.

Mitigation & Security Layers:

- Use MFA for added protection.
- Rotate secrets and signing keys periodically.
- Rate-limit login attempts.

2️⃣ Explain how the CIA triad applies to system design.

Answer:

Principle	Description	Example in System Design
Confidentiality	Prevent unauthorized access to data	HTTPS, encryption at rest, IAM roles
Integrity	Prevent unauthorized modification of data	HMACs, digital signatures, input validation

Availability	Ensure services are accessible when needed	Load balancing, autoscaling, DDoS protection

Tip: Always balance trade-offs. Example: Too much availability can compromise confidentiality (open endpoints).

3️⃣ What are common security threats in a microservices architecture, and how would you mitigate them?

Answer: Threats:

- Unauthorized inter-service calls
- Sensitive data leakage over internal APIs
- No centralized audit logging
- API gateway spoofing

Mitigations:

- **Service-to-service authentication** using mTLS or identity tokens
- Implement a **zero-trust model**: each service authenticates every call
- **Encrypt all traffic** inside the network (TLS everywhere)
- Use **API gateways** for authentication, throttling, and schema validation
- Centralized **logging and monitoring**

4️⃣ How would you protect your system from a DDoS attack?

Answer:

- **Rate limiting** at edge (e.g., via API gateway or CDN like Cloudflare)
- **Web Application Firewall (WAF)** to block malicious patterns
- **Auto-scaling infrastructure** (e.g., Kubernetes horizontal pod autoscaling)

- **Global load balancers** to distribute traffic
- Set up **traffic scrubbing** services (Cloudflare Spectrum, AWS Shield Advanced)

Tip: Detect early via monitoring spikes in request volume, latency, or failed auth.

5️⃣ What role does TLS/HTTPS play in system security?

Answer:

- **TLS ensures confidentiality** (encrypted data in transit)
- **TLS ensures integrity** (detects tampering)
- **Authenticates server identity** using certificates

How to manage at scale:

- Use **automated cert rotation** (e.g., Let's Encrypt with Certbot)
- Offload TLS at the **load balancer** or **API gateway**
- Enforce **HSTS (HTTP Strict Transport Security)**

Important: Always disable older protocols like TLS 1.0 and 1.1.

6️⃣ How can you ensure secure data storage in a cloud-based system?

Answer:

- **Encrypt data at rest** using KMS-managed keys (e.g., AWS KMS, GCP KMS)
- Use **role-based access control (RBAC)** for access management
- Store secrets in a **secure vault** (AWS Secrets Manager, HashiCorp Vault)
- **Audit logs** to track who accessed what and when

Follow the principle of least privilege. E.g., DB access should be granted only to required services.

7️⃣ What is threat modeling and how would you incorporate it into your design process?

Answer:

- **Threat modeling** is identifying:
    - What you're protecting (assets)
    - From whom (threat actors)
    - Where you’re vulnerable (attack vectors)
    - How to defend (mitigations)

How to integrate:

- Conduct during **design phase** of SDLC
- Use models like **STRIDE** or **DREAD**
- Document each component and assess risk
- Iterate with every new feature or major change