2.0 Security Architecture

2.1 Given a scenario, analyze requirements to design resilient systems.

- **Component placement and configuration**
  - Firewall
  - Intrusion prevention system (IPS)
  - Intrusion detection system (IDS)
  - Vulnerability scanner
  - Virtual private network (VPN)
  - Network access control (NAC)
  - Web application firewall (WAF)
  - Proxy
  - Reverse proxy
  - Application programming interface (API) gateway
  - Taps
  - Collectors
  - Content delivery network (CDN)
- **Availability and integrity design considerations**
  - Load balancing
  - Recoverability
  - Interoperability
  - Geographical considerations
  - Vertical vs. horizontal scaling
  - Persistence vs. non-persistence

2.2 Given a scenario, implement security in the early stages of the systems life cycle and throughout subsequent stages.

- **Security requirements definition**
  - Functional requirements
  - Non-functional requirements
  - Security vs. usability trade-off
- **Software assurance**
  - Static application security testing (SAST)
  - Dynamic application security testing (DAST)
  - Interactive application security testing (IAST)
  - Runtime application self-protection (RASP)
  - Vulnerability analysis
  - Software composition analysis (SCA)
  - Software bill of materials (SBoM)
  - Formal methods
- **Continuous integration/ continuous deployment (CI/CD)**
  - Coding standards and linting
  - Branch protection
  - Continuous improvement
  - Testing activities
    - Canary
    - Regression
    - Integration
    - Automated test and retest
    - Unit
- **Supply chain risk management**
  - Software
  - Hardware
- **Hardware assurance**
  - Certification and validation process
- **End-of-life (EOL) considerations**

2.3 Given a scenario, integrate appropriate controls in the design of a secure architecture.

- **Attack surface management and reduction**
  - Vulnerability management
  - Hardening
  - Defense-in-depth
  - Legacy components within an architecture
- **Detection and threat-hunting enablers**
  - Centralized logging
  - Continuous monitoring
  - Alerting
  - Sensor placement
- **Information and data security design**
  - Classification models
  - Data labeling
  - Tagging strategies
- **DLP**
  - At rest
  - In transit
  - Data discovery
- **Hybrid infrastructures**
- **Third-party integrations**
- **Control effectiveness**
  - Assessments
  - Scanning
  - Metrics

2.4 Given a scenario, apply security concepts to the design of access, authentication, and authorization systems.

- Provisioning/deprovisioning
  - Credential issuance
  - Self-provisioning
- Federation
- Single sign-on (SSO)
- Conditional access
- Identity provider
- Service provider
- Attestations
- Policy decision and enforcement points
- Access control models
  - Role-based access control
  - Rule-based access control
  - Attribute-based access control (ABAC)
  - Mandatory access control (MAC)
  - Discretionary access control (DAC)
- Logging and auditing
- Public key infrastructure (PKI) architecture
  - Certificate extensions
  - Certificate types
  - Online Certificate Status Protocol (OCSP) stapling
  - Certificate authority/registration authority (CA/RA)
  - Templates
  - Deployment/integration approach
- Access control systems
  - Physical
  - Logical

2.5 Given a scenario, securely implement cloud capabilities in an enterprise environment.

- Cloud access security broker (CASB)
  - API-based
  - Proxy-based
- Shadow IT detection
- Shared responsibility model
- CI/CD pipeline
- Terraform
- Ansible
- Package monitoring
- Container security
- Container orchestration
- Serverless
  - Workloads
  - Functions
  - Resources
- API security
  - Authorization
  - Logging
  - Rate limiting
- Cloud vs. customer-managed
  - Encryption keys
  - Licenses
- Cloud data security considerations
  - Data exposure
  - Data leakage
  - Data remanence
  - Insecure storage resources
- Cloud control strategies
  - Proactive
  - Detective
  - Preventative
- Customer-to-cloud connectivity
- Cloud service integration
- Cloud service adoption

2.6 Given a scenario, integrate Zero Trust concepts into system architecture design.

- Continuous authorization
- Context-based reauthentication
- Network architecture
  - Segmentation
  - Microsegmentation
  - VPN
  - Always-on VPN
- API integration and validation
- Asset identification, management, and attestation
- Security boundaries
  - Data perimeters
  - Secure zone
  - System components
- Deperimeterization
  - Secure access service edge (SASE)
  - Software-defined wide area network (SD-WAN)
  - Software-defined networking
- Defining subject-object relationships