SAST & Dependency Scan

Static Application Security Test

Analyze source code + Scan 3rd party dependencies and libraries

Container/Image Scan

Scans container/images to identify vulnerabilities within containers and their components

DAST

Dynamic Application Security Test

Simulate attacks on running application by actively interacting with them by manipulating inputs etc.

Pen Test

Attack by a third party team of security experts to discover vulnerabilities by using manual/automated tools & techniques

Assurance Through Evaluation: Enhancing Security and Compliance

A comprehensive overview of industry-recognized frameworks and architectural solutions to enhance security, software quality, and regulatory compliance across an organization.

Introduction to Security Assurance

- **Assurance Through Evaluation**
  Structured methodologies to validate security controls, assess software maturity, and certify systems based on industry-recognized frameworks.
- **Common Criteria Evaluation Assurance Levels (EAL)**
  A scale from EAL1 (Basic) to EAL7 (Advanced) that determines the depth of security testing and analysis for IT products.
- **ISO/IEC 27000 Series**
  International standards for information security management, providing guidance on security policies, risk management, and compliance.
- **Capability Maturity Model (CMMI-DEV)**
  A framework to assess and improve software development processes, enhancing security, quality, and efficiency.
- **Intergroup Coordination**
  Aligning security policies with business objectives, regulatory requirements, and risk management strategies across IT, development, and security teams.
- **Peer Reviews**
  Independent evaluation of software code, security designs, and risk assessments by subject-matter experts to identify vulnerabilities and ensure compliance.

The Common Criteria Evaluation Assurance Scale

- Structured Approach to Security Assessment

  The Common Criteria Evaluation Assurance Level (EAL) scale provides a standardized methodology for evaluating the security strength of IT products.

- EAL1: Functionally Tested

  Basic security testing without formal design analysis, suitable for low-risk IT products.

- EAL2: Structurally Tested

  Security testing with documented design reviews, appropriate for products with moderate security requirements.

- EAL3: Methodically Tested and Checked

  More rigorous functional and vulnerability testing, intended for products with significant security concerns.

- EAL4: Methodically Designed, Tested, and Reviewed

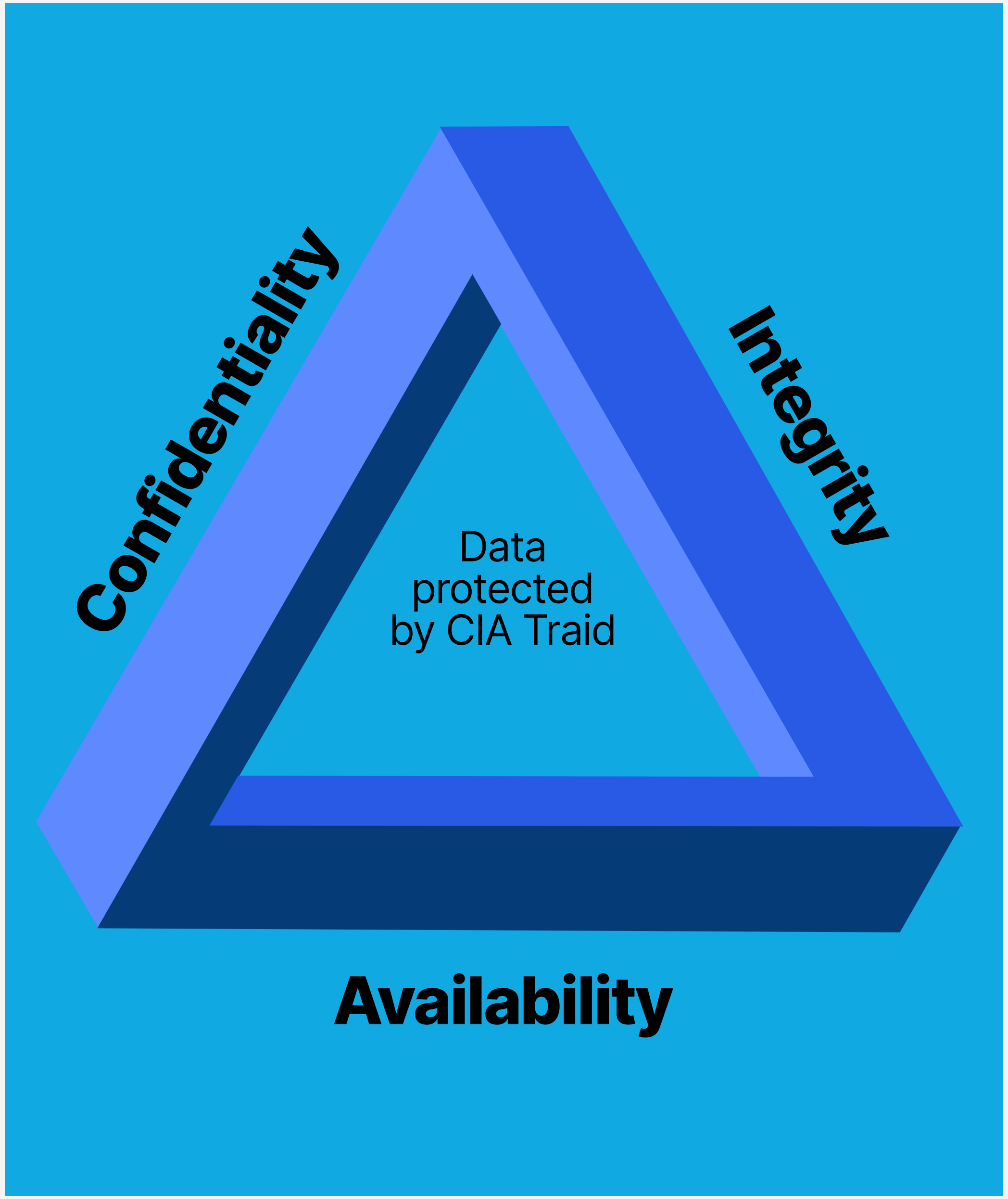
  The highest level achievable without specialized security design methods, suitable for products with high security requirements.

- EAL5-EAL7: Advanced Assurance Levels

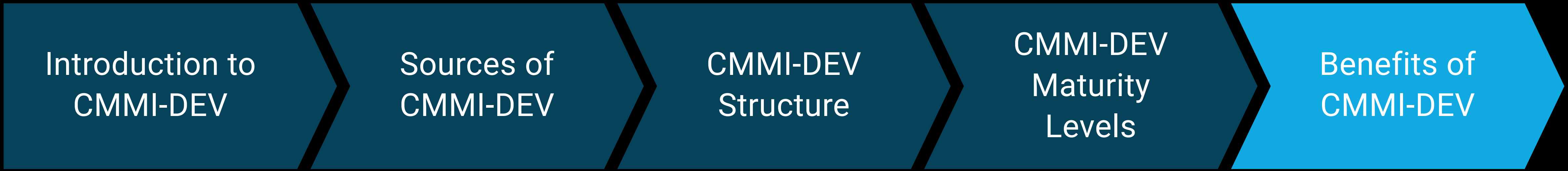
  Require formal design verification, mathematical modeling, and security-proof validation, primarily for highly critical systems like military and national security applications.

ISO/IEC 27000 Series: Information Security Standards

The ISO/IEC 27000 series consists of international standards that provide a comprehensive framework for information security management systems (ISMS). These standards define security policies, risk management practices, and compliance measures for organizations handling sensitive data.



The Capability Maturity Model (CMMI-DEV)



Introduction to CMMI-DEV

The Capability Maturity Model Integration for Development (CMMI-DEV) is a framework developed by the Software Engineering Institute (SEI) to assess and improve software development processes. It provides structured guidelines for organizations to enhance software security, quality, and efficiency.

Sources of CMMI-DEV

The CMMI-DEV model is based on best practices from software engineering, risk management, and cybersecurity frameworks, integrating knowledge from ISO 9001, IEEE Software Engineering Standards, and Defense Industrial Base (DIB) Security Frameworks.

CMMI-DEV Structure

CMMI-DEV is organized into process areas that guide organizations in developing secure and reliable software, including Project Planning & Risk Management, Configuration Management, Process & Product Quality Assurance, Measurement & Analysis, and Causal Analysis & Resolution.

CMMI-DEV Maturity Levels

The CMMI-DEV model consists of five maturity levels that reflect an organization's ability to manage, control, and continuously improve software development processes. Higher maturity levels demonstrate more mature and well-documented security practices, reducing vulnerabilities in software development lifecycles.

Benefits of CMMI-DEV

Organizations that implement CMMI-DEV can enhance software security, quality, and efficiency, reduce vulnerabilities, and demonstrate compliance with industry standards and regulatory requirements.

Intergroup Coordination in Security Architecture

DevSecOps Implementation

Integrating security into software development cycles by fostering collaboration between development, security, and operations teams.

Cross-Department Security Awareness

Training employees across IT, development, and business departments on threat detection, incident response, and compliance requirements.

Security Governance & Incident Response Planning

Establishing centralized security frameworks and communication channels to respond to cyber threats efficiently across the organization.

Aligning Security Policies with Business Objectives

Ensuring security architecture and controls are tailored to support the organization's strategic goals and regulatory compliance needs.

Risk Management Coordination

Involving cross-functional teams in identifying, assessing, and mitigating security risks across the software development lifecycle.

Peer Reviews: Validating Security and Compliance

Formal Code Reviews

Security analysts conduct in-depth reviews of software code to identify vulnerabilities in cryptographic implementations and secure coding practices.

Threat Modeling Reviews

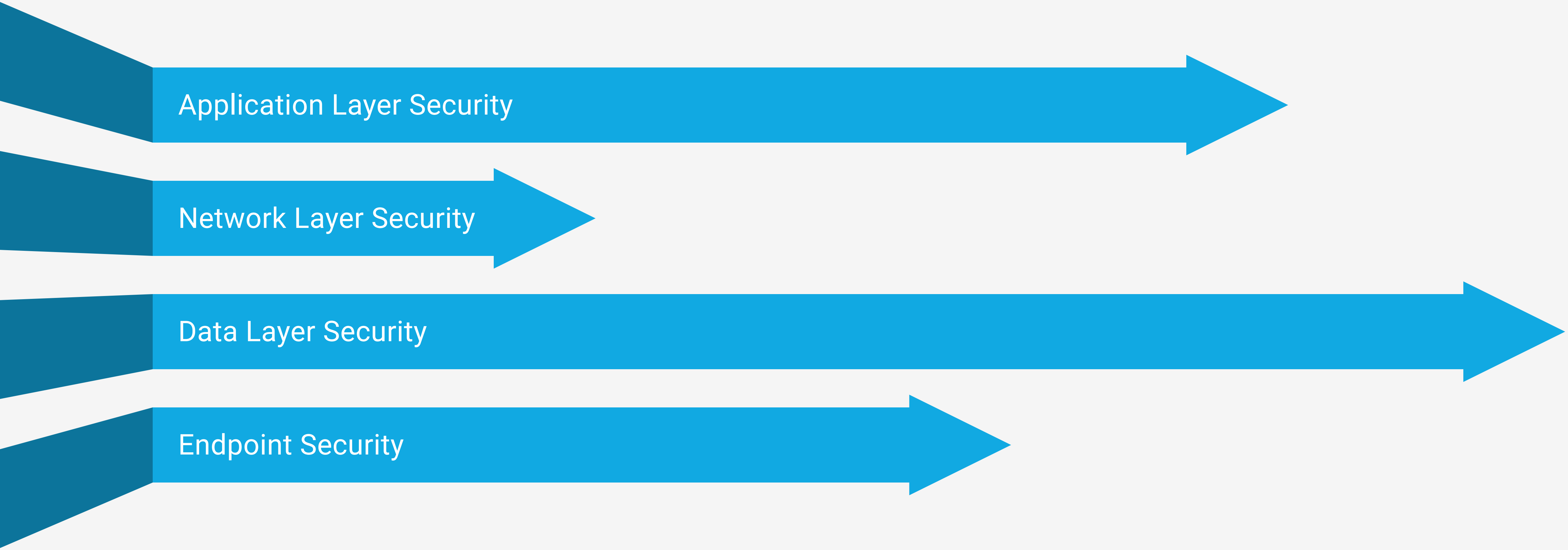
Evaluating system designs for potential attack vectors and security flaws to enhance architectural decisions.

Compliance Audits

Ensuring adherence to industry standards like NIST, ISO/IEC 27001, and PCI-DSS to meet regulatory requirements.

Peer reviews help reduce software flaws, enhance security posture, and improve resilience against cyber threats by identifying vulnerabilities, validating architectural decisions, and ensuring compliance with industry standards before software deployment.

The OSI Reference Model: Layered Security Approach



Application Layer Security

Network Layer Security

Data Layer Security

Endpoint Security

PCI-DSS: Securing Payment Card Transactions

Requirement	Description
Encrypt Cardholder Data	Mandate the use of AES-256 or RSA encryption for storing and transmitting payment card information
Implement Multi-Factor Authentication (MFA)	Enforce strong access controls by requiring at least two forms of authentication for payment processing systems
Conduct Regular Security Audits and Penetration Testing	Regularly assess the security posture of payment systems and identify potential vulnerabilities through rigorous testing
Monitor and Log Security Events	Implement Security Information and Event Management (SIEM) solutions to detect and respond to security incidents

Architectural Solutions for Security Assurance

Zero Trust Architecture (ZTA)

Enforces continuous authentication and access verification to reduce attack surfaces and minimize lateral movement within the network.

Microservices Security

Implements granular API security, container isolation, and secure DevOps practices to enhance application-level security in distributed architectures.

Cloud Security Models

Leverages Identity and Access Management (IAM), data encryption, and multi-cloud security policies to ensure consistent security controls across hybrid and multi-cloud environments.

AI-Driven Threat Detection

Uses machine learning and advanced analytics to identify anomalies, detect advanced persistent threats (APTs), and enable automated incident response for enhanced security monitoring and resilience.

Conclusion: Continuous Evaluation and Alignment

Continuous Evaluation

Regularly assess security architectures, software, and IT systems to ensure they meet evolving security and compliance requirements.

Refinement of Security Frameworks

Continuously review and update security frameworks to address emerging threats, vulnerabilities, and industry best practices.

Alignment with Regulatory Changes

Ensure security frameworks and controls are aligned with new or updated regulatory requirements, such as GDPR, HIPAA, and PCI-DSS.

Effective security assurance requires a proactive and adaptive approach, where security architectures, software, and IT systems are continuously evaluated, refined, and aligned with evolving threats and regulatory changes. This ensures an organization's security posture remains robust and compliant over time.