Software Development Lifecycle (SDLC)

Understand and integrate security throughout the software development lifecycle (SDLC)

Development Methodologies

Build and fix	• No key architecture design • Problems fixed as they occur • No formal feedback cycle • Reactive not proactive
Waterfall	• Linear sequential lifecycle • Each phase is completed before moving on • No formal way to make changes during cycle • Project ends before collecting feedback and re-starting
V-shaped	• Based on the waterfall model • Each phase is complete before moving on • Verification and validation after each phase • No risk analysis phase
Prototyping	• Rapid prototyping - quick sample to test the current project • Evolutionary prototyping - incremental improvements to a design • Operational prototypes - incremental improvements intended for production
Incremental	• Multiple cycles (~ multiple waterfalls) • Restart at any time as a different phase • Easy to introduce new requirements • Delivers incremental updates to software
Spiral	• Iterative • Risk analysis during development • Future information and requirements considered for risk analysis • Allows for testing early in development
Rapid Application Development (RAD)	• Rapid prototyping • Designed for quick development • Analysis and design are quickly demonstrated • Testing and requirements are often revisited
Agile	• Umbrella term - multiple methods • Highlights efficiency and iterative development • User stories describe what a user does and why • Prototypes are filtered down to individual features

DevOps (Development & Operations)

Software Development • Quality Assurance • IT Operations

Software Development Methods

Database Systems

Database	Define storing and manipulating data
DBMS (database management system)	Software program control access to data stored in a database.
DBMS Types	Hierarchical • Network • Mesh • Object-orientated • Relational
DDL	Data definition language defines structure and schema DML
Degree of Db	number of attributes (columns) in table
Tuple	row
DDE	Dynamic data exchange
DCL	Data control language. Subset of SQL.
Semantic integrity	ensure semantic rules are enforced between data types
Referential integrity	all foreign keys reference existing primary keys
Candidate Key	an attribute that is a unique identifier within a given table, one of the candidates key becomes primary key and others are alternate keys
Primary Key	unique data identification
Foreign Key	reference to another table which include primary key. Foreign and primary keys link is known as referential integrity.
DBMS terms	• Incorrect Summaries • Dirty Reads • Lost Updates • Dynamic Lifetime Objects: Objects developed using software in an Object Oriented Programming environment. • ODBC - Open Database Connectivity. Database feature where applications to communicate with different types of databases without a program code. • Database contamination - Mixing data with different classification levels • Database partitioning - splitting a single database into multiple parts with unique contents • Polyinstantiation - two or more rows in the same relational database table appear to have identical primary key and different data in the table.

Programming Language Types

Machine Languages	Direct instructions to processor - binary representation
Assembly Language	Use of symbols, mnemonics to represent binary codes - ADD, PUSH and POP
High-Level Language	Processor independent programming languages - use IF, THEN and ELSE statements as part of the code logic
Very high-level language	Generation 4 languages further reduce amount of code required - programmers can focus on algorithms. Python, C++, C# and Java
Natural language	Generation 5 languages enable system to learn and change on its own - AI

Database Architecture and Models

Relational Model	Uses attributes (columns) and tuples (rows) to organize data
Hierarchical Model	Parent child structure. An object can have one child, multiple children or no children.
Network Model	Similar to hierarchical model but objects can have multiple parents.
Object-Oriented Model	Has the capability to handle a variety of data types and is more dynamic than a relational database.
Object-Relational Model	Combination of object oriented and relational models.

Database Interface Languages

Open Database Connectivity (DOBC)	Local or remote communication via API
Java Database Connectivity (JDBC)	Java API that connects to a database, issuing queries and commands, etc
XML	DB API allows XML applications to interact with more traditional databases
Object Linking and Embedding Database (OLE DB)	is a replacement for ODBC

Knowledge Management

Expert Systems	Two main components: 'Knowledge base' and the 'Inference engine' • Use human reasoning • Rule based knowledge base • If-then statements • Interference system
Expert Systems (Two Modes)	• Forward chaining: Begins with known facts and applies inference rule to extract more data unit it reaches to the goal. A bottom-up approach. Breadth-first search strategy. • Backward chaining: Begins with the goal, works backward through inference rules to deduce the required facts that support the goal. A top-down approach. Depth-first search strategy.
Neural Networks	Accumulates knowledge by observing events, measuring their inputs and outcome, then predicting outcomes and improving through multiple iterations over time.

Covert Channels (Storage & Timing)

Executable content Mobile code	ActiveX controls, Java applets, browser scripts
Virus	Propagates with help from the host
Worm	Propagates without any help from the host
Logic Bomb/Code Bomb	Run when a specific event happens
Buffer Overflow	Memory buffer exhaustion
Backdoor	Malicious code install at back end with the help of a front end user
Covert Channel	Unauthorized information gathering
Botnet	Zombie code used to compromise thousands of systems
Trojan	Malicious code that outwardly looks or behaves as harmless or necesary code

Data Warehousing and Data Mining

Data Warehousing	Combine data from multiple sources.
Data Mining	Arrange the data into a format easier to make business decisions based on the content.

Database Threats

Aggregation	The act of combining information from various sources.
Inference	Process of information piecing
Access Control	• Content Dependent Access Control: access is based on the sensitivity of the data • Context Dependent Access Control: access via location, time of day, and previous access history.
Access Control Mechanisms	• Database Views: set of data a user or group can see • Database Locks: prevent simultaneous access • Polyinstantiation: prevent data interference violations in databases

A • C • I • D

Atomicity	Database roll back if all operations are not completed, transactions must be completed or not completed at all
Consistency	Preserve integrity by maintaining consistent transactions
Isolation	Transaction keeps separate from other transactions until complete
Durability	Committed transaction cannot be roll backed

Traditional SDLC

Steps	Analysis, High-level design, Detail Design, Construction, testing, Implementation
Phases	• Initiation: Feasibility, cost analysis, risk analysis, Management approval, basic security controls • Functional analysis and planning: Requirement definition, review proposed security controls • System design specifications: detailed design specs, Examine security controls • Software development: Coding. Unit testing Prototyping, Verification, Validation • Acceptance testing and implementation: security testing, data validation

Object-oriented technology (OOT) - Terminology

Objects contain both data and the instructions that work on the data.

Encapsulation	Data stores as objects
Message	Informs an object to perform an action.
Method	Performs an action on an object in response to a message.
Behavior	Results shown by an object in response to a message. Defined by its methods, which are the functions and subroutines defined within the object class.
Class	Set of methods which defines the behavior of objects
Object	An instance of a class containing methods
Inheritance	Subclass accesses methods of a superclass
Multiple Inheritance	Inherits characteristics from more than one parent class
Polyinstantiation	Two or more rows in the same relational database table appear to have identical primary key elements but contain different data
Abstraction	Object users do not need to know the information about how the object works
Process isolation	Allocation of separate memory spaces for process's instructions and data by the operating system.

Trusted Computer Base (TCB)

The set of all hardware, firmware, and/or software components that are critical to its security. Any compromises here are critical to system security.

Input/output operations	May need to interact with higher rings of protection - such communications must be monitored
Execution domain switching	Applications that invoke applications or services in other domains
Memory protection	Monitoring of memory references to verify confidentiality and integrity in storage
Process activation	Monitor registers, process status information, and file access lists for vulnerabilities

Security Assessment & Testing Terms

Cross-site request forgery (CSRF / XSRF)	Browser site trust is exploited by trying to submit authenticated requests forcefully to third-party sites.	Penetration Testing	A process of identifying and determining the true nature if system vulnerabilities
Cross-site scripting (XSS)	Uses inputs to pretend a user's browser to execute untrusted code from a trusted site	Patch management system	Manages the deployment of patches to prevent known attack vectors
Session Hijacking	Attempts to obtain previously authenticated sessions without forcing browser requests submission	Open system	System with published APIs - third parties can use system
SQL Injection	Directly attacks a database through a web app	Closed system	Proprietary system - no third-party involvement
Hotfix / Update / Security fix	Updates to operating systems and applications	Open-source	Source code can be viewed, edited and distributed free or with attribution or fees
Service Pack	Collection of patches for a complete operating system	API Keys	Used to access API. Highly sensitive - same as passwords

Change Management Process

Request Control	Develop organizational framework where users can request modifications, conduct cost/ benefit analysis by management, and task prioritization by developers
Change Control	Develop organizational framework where developers can create and test a solution before implementation in a production environment.
Release Control	Change approval before release

Configuration Management Process

Software Version Control (SVC)	A methodology for storing and tracking changes to software
Configuration Identification	The labelling of software and hardware configurations with unique identifiers
Configuration Control	Verify modifications to software versions comply with the change control and configuration management policies.
Configuration Audit	Ensure that the production environment is consistent with the accounting records

Capability Maturity Model

Reactive	1. Initiating – informal processes, 2. Repeatable – project management processes
Proactive	3. Defined – engineering processes, project planning, quality assurance, configuration management practices 4. Managed – product and process improvement 5. Optimizing – continuous process improvement

Project Management Tools

Gantt chart	Type of bar chart that illustrates the relationship between projects and schedules over time.
Program Evaluation Review Technique (PERT)	Project-scheduling tool used to measure the capacity of a software product in development which uses to calculate risk.

Phases of object-oriented design

OORA (Requirements Analysis)	Define classes of objects and interactions
OOA (Analysis)	Identify classes and objects which are common to any applications in a domain - process of discovery
OOD (Design)	Objects are instances of classes
OOP (Programming)	Introduce objects and methods
ORBs (Object Request Brokers)	Work as middleware locators and distributors for the objects
CORBA (Common object request)	Architecture and standards that use ORBS to allow different systems and software on a system to interfce with eachother
Cohesion	Work independently without help from other programs • High cohesion – No integration or interaction with other modules • Low cohesion – Have interaction with other modules • Coupling - Level of interaction between objects

Virus Types

Boot sector	Boot record infectors, gain the most privaleged access and can be the most damaging
System infector	Infects executable system files, BIOS and system commands
UEFI	Infects a system's factory installed UEFI (firmware)
Companion	Virus stored in a specific location other than in the main system folder. Example NOTEPAD.EXE
Stealth	Any modifications to files or boot sector are hidden by the virus
Multipart	Infects both boot sector and executable files
Self-garbling	Attempts to hide from anti-virus by changing the encoding of its own code, a.k.a. 'garbling'
Polymorphic	The virus modifies the "garble" pattern as it spreads
Resident	Loads as and when a program loads to the memory
Master boot record / sector (MBR)	Infects the bootable section of the system

Anti-Virus Types

Signature based	Not able to detect new malware a.k.a. Zero-day attacks
Heuristic based	Static analysis without relying on signatures

Protection Rings

Layer 0	Operating system kernel
Layer 1	Parts of the operating system other than the kernel
Layer 2	I/O drivers and utilities
Layer 3	Applications and programs