Welcome to Copernicus CPA Review!

ISC

Copernicus CPA Review

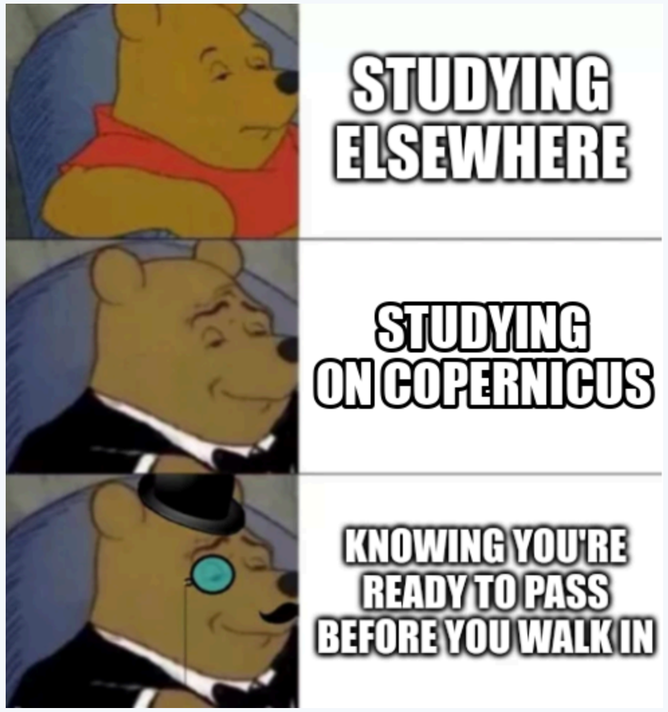
Hey there! This is the version of our memorization guide for CPA candidates who aren't yet actively studying on our platform. It has everything the other version does, and includes two extra slides including this one to let you know why you might want to give Copernicus CPA Review a try.

What makes Copernicus different?

- ✔ **Affordable:** Each course costs less than a single exam retake
- ✔ **ReadyRating Readiness Predictions:** a 5-level rating tells you when to sit or reschedule, to help you avoid retake fees
- ✔ **6,700+ Questions:** a deep practice bank across all six exams, including AICPA materials
- ✔ **200+ Task-Based Simulations:** realistic practice for the simulation testlets you will see on every section of the exam
- ✔ **Pass Guarantee:** Refund if you fail after reaching ReadyRating Level 5

Copernicus CPA Review is also super efficient. Our platform maximizes your readiness by actively discovering what you know, targeting gaps, reinforcing memory, and giving you a clear picture of what you need to do to pass. Our free account is free forever, with 50+ lesson videos, 200+ practice questions, 100+ flashcards and 10 task-based simulations.

We hope you'll join us on your CPA journey! **Create your free account**



ISC Exam Memorization Guide

Guide version 5.3.1

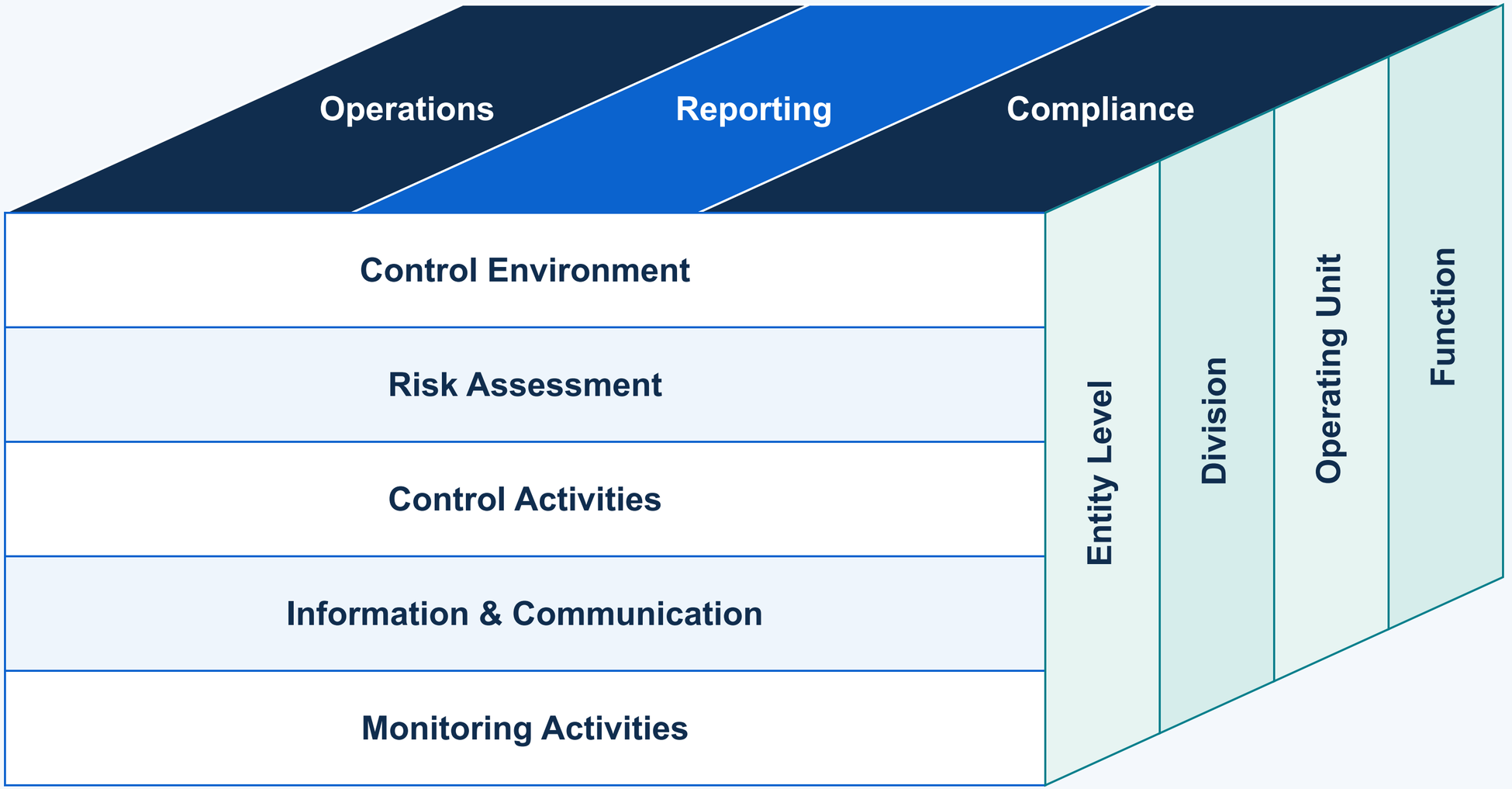
Memorization Guide - What's Current

This guide reflects the current AICPA Exam Blueprint for ISC, covering information systems and data management; security, confidentiality and privacy; and considerations for System and Organization Controls (SOC) engagements. Content aligns with the AICPA Trust Services Criteria (SOC 2), COSO Internal Control Framework (2013), COBIT governance framework, and NIST Cybersecurity Framework. Coverage includes IT general controls (ITGCs), application controls, SOC 1/SOC 2/SOC 3 report structures, and current data governance and privacy considerations. Framework and standard versions follow the references in the current AICPA ISC blueprint. Every page is checked line by line against primary authority before each version is released.

Each of the CPA exams is different. Similar to the other exams, the ISC exam, while not easy, has the benefit of containing significant memorization. Why is this a benefit? If you were able to take the exam with a textbook, you would likely pass. By memorizing the essentials covered in this guide your mind becomes a concise ISC textbook that you can bring in on exam day.

When it comes to the content included here, my sincere advice is to not enter the exam until it is all memorized. Do not view this as a cause for concern; rather, view it as a lifesaving guide that will earn you free and stress-free points on the exam. Practice reinforcement of these points through our flashcards. Do the heavy lifting now while you have time, rather than a week before the exam when you would have to cram.

COSO cube


Operations
Reporting
Compliance
Control Environment
Risk Assessment
Control Activities
Information & Communication
Monitoring Activities
Entity Level
Division
Operating Unit
Function

The five components of the COSO internal control integrated framework

Control Environment

- Commitment to Ethics & Integrity
- Board Independence & Oversight
- Organizational Structure
- Commitment to Competence
- Accountability

Risk Assessment

- Specify Objectives
- Identify & Analyze Risks
- Consider Potential for Fraud
- Identify & Assess Changes

Control Activities

- Select & Develop Control Activities
- Select & Develop General Controls over Technology
- Deployment of Policies & Procedures

Information & Communication

- Obtain & Use Information
- Internally Communicate Information
- Communicate with External Parties

Monitoring Activities

- Ongoing & Separate Evaluations
- Communication of Deficiencies

Effective internal control: all five components and the relevant principles are present, functioning and operating together; it gives reasonable, not absolute, assurance.
SOC 2 (System and Organization Controls) link: the Trust Services Criteria common criteria CC1 to CC5 follow these five components and 17 principles.

The five components of the COSO (ERM) enterprise risk management framework

Governance and Culture

- Exercises board oversight
- Establishes operating structure
- Defines desired culture
- Demonstrates commitment to core values
- Attracts, develops, and retains capable employees

Strategy and Objective-Setting

- Analyzes business context
- Defines risk appetite
- Evaluates alternative strategies
- Formulates business objectives

Performance

- Identifies risks
- Assesses severity of risk
- Prioritizes risk
- Implements risk responses
- Develops portfolio view

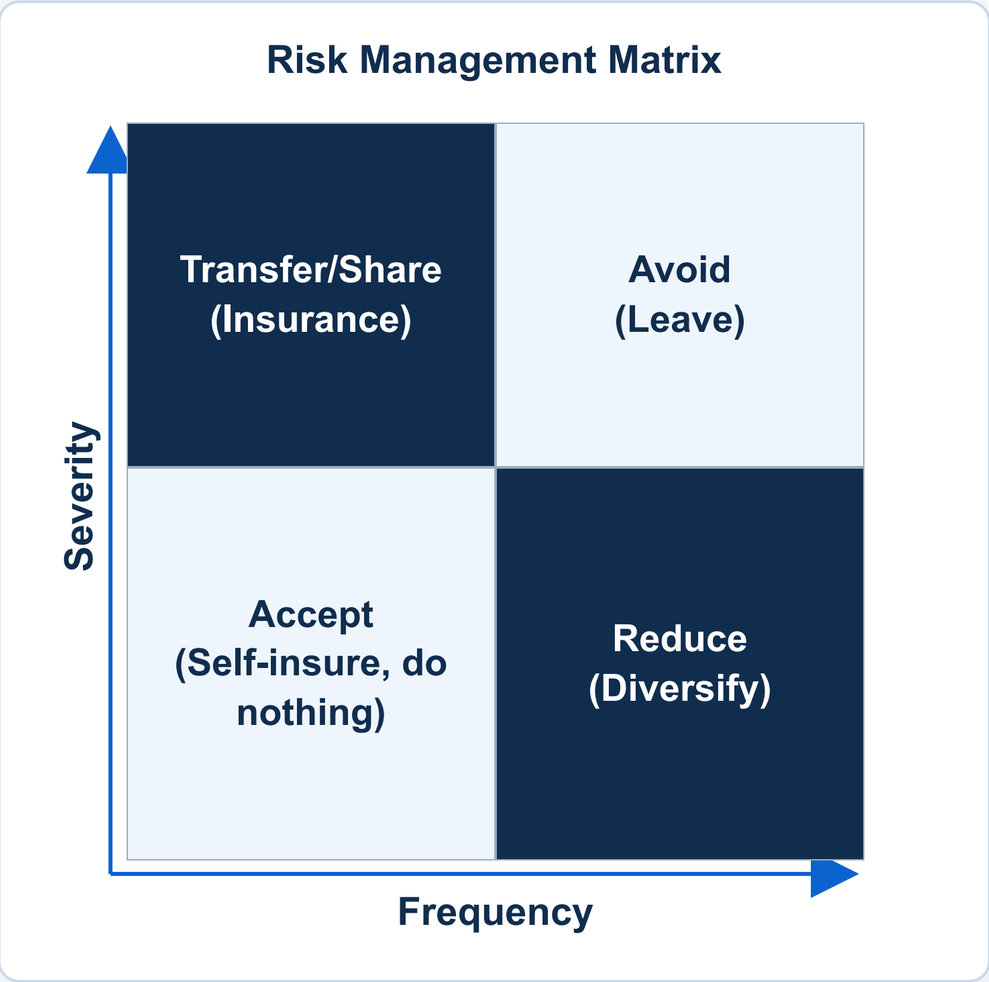
Review and Revision

- Assesses substantial change
- Reviews risk and performance
- Pursues improvement in ERM

Information, Communication, and Reporting

- Leverages information and technology
- Communicates risk information
- Reports on risk, culture, and performance

Responses to risk



Residual Risk - The risk that remains after management efforts to respond to the risk and alter its severity have been made.

Inherent Risk - This is the risk that exists without any action being taken to curb it.

Residual Risk (conceptual) = Inherent Risk - Effect of management's risk responses

COSO (Committee of Sponsoring Organizations of the Treadway Commission) Enterprise Risk Management (2017) lists five risk responses: accept, avoid, pursue, reduce and share. Pursue means accepting more risk to achieve improved performance.

The value cycle

Value Creation:

When the benefits of resources deployed exceed their cost, e.g., launching a new product.

Value Preservation:

Day-to-day operations sustain created benefits, e.g., high customer satisfaction and the profitability of the new product.

Value Realization:

When stakeholders derive the benefits from Value Creation (monetary or non-monetary), for shareholders generally through dividends or stock appreciation.

Value Erosion:

When strategy or day-to-day execution fails and value is lost, such as lower profitability, lower customer satisfaction, lack of demand, legal issues, competition, etc.

Fraud triangle

Incentive

Financial pressure:

- Greed
- Addictions
- Poor credit rating or cash management

Work pressure:

- Dissatisfaction with pay
- Overlooked for promotion

Opportunity

- Weak internal controls
- Ineffective monitoring of controls
- Assets susceptible to fraud

Rationalization

- “I’ll pay it back.”
- “I deserve a pay raise.”
- “It’s for a good purpose.”

Separation of duties

Separation of Duties

Authorization of Transactions

- Approving transactions and changes to records
- Approving (signing off on) the daily cash balancing report

Recording of Transactions

- Performing reconciliation
- Entering payment information and relieving accounts receivable

Custody of Assets

- Physical access to cash
- Preparation of bank deposit
- Physical access to vehicle inventory

Personnel should never have more than one of these activities within their role. The risk/reward tradeoff is difficult at a smaller company which may need one employee to cover responsibilities in two of these areas.

Collusion and/or management override are the most common ways to circumvent separation of duties. Additional controls need to be put in place to contain these concerns.

Types of internal controls

Internal Controls Classification

Preventive: Prevent errors and fraud before they occur.

For example:

- Segregation of duties
- Job rotation
- Training
- Dual access
- Authorization and approval
- Preformatted input
- Input edit checks (check digits, limit and validity checks)

Detective: Uncover errors and fraud after they have occurred.

For example:

- Transmittal documents
- Batch control totals
- Hash totals
- Batch balancing

Corrective: Correct the negative effects of unwanted events.

For example:

- Upstream resubmissions
- Repairing damage
- Remanufacturing
- Isolating and removing viruses
- Disciplinary action

Cyber examples: preventive = system hardening, multifactor authentication (MFA), intrusion prevention; detective = log review, intrusion detection; corrective = quarantining infected devices, patching, restoring from backup.

IT governance personnel

Board of Directors: Fiduciaries who oversee and appoint executive positions. The Board evaluates IT governance policies for the organization.

Executive Management: C-Suite management responsible for the “tone at the top” and for execution of IT governance policies.

Middle Management: Branch, regional, store, and department managers who act as an extension of Executive Management.

Network Administrators/Engineers: Design and maintain the network including configuring network hardware and remote access.

Executive-Level IT Staff: Roles such as CTO, CIO, or IT Director responsible for IT strategy and governance execution.

Help Desk and Lower-Level IT Staff: First responding troubleshooters for end-users.

Cybersecurity Staff: Ensure security of the company network and IT-related assets.

Function-Specific Staff: Work on one specific role such as database management and have a narrow focus.

IT governance personnel (continued)

Accountants: Responsible for the confidential data such as banking and financial records processed through the entity's IT systems.

Data Librarian: The person who should maintain the custody of an entity's data in a large firm.

Software Developer/Programmer: Designs and/or writes programs to collect, process, store, transform and distribute the data and information that are entered by the end users. Developers have a larger scope of a project whereas programmers specifically write code.

Systems Analyst: Designs systems, prepares specifications for programmers, and serves as intermediary between users and programmers.

End Users: Those who are best equipped to understand the day-to-day practical uses of organizational technological needs as the End Users are the ones utilizing the technology.

External Stakeholders: Customers, vendors, shareholders, auditors, regulators, and more.

Project Development Teams: Responsible for ground-level functions such as designing IT infrastructure, planning, tracking, and monitoring project performance. Lower level of personnel than the Steering Committee. These teams are formed for new IT projects as they arise.

Steering Committees: Responsible for overseeing information systems functions. Has a more holistic and big picture view than the Project Development Team. Able to review and approve long-term plans that include budget allocation, strategic goals, monitoring progress, and addressing concerns.

Business information systems

Management Information System (MIS) - Routine reports that help managers plan and control operations.

Decision Support System (DSS) - Management uses to make decisions. Interactive system for “What-if” scenarios and forecasting.

Accounting Information System (AIS) - Customizable but useable for any type of business. Designed to create an audit trail.

- Transaction Processing System (TPS)
- Management Reporting System (MRS)
- Financial Reporting System (FRS) / General Ledger System (GLS)

Executive Information System (EIS) or Executive Support System (ESS) - Provides high-level, summarized reports and dashboards for senior executives to support strategic decision-making. EIS serves a small user base (top management only) and draws data from other subsystems like MIS and TPS. The Transaction Processing System (TPS) is typically the most widely used subsystem as it processes the highest volume of routine business transactions.

Enterprise Resource Planning (ERP) - Integrates whole company data in one shared database. Multiple systems such as sales, accounting, operations, etc. are centralized.

The greatest impact on the design of any management information system is the types of decisions that need to be made with it.

IT outsourcing through cloud computing

Infrastructure as a Service (IaaS) - Amazon Web Services

- Outsourced hardware infrastructure.
- A cloud computing service where enterprises rent or lease servers for computational ability and storage in the cloud.

Platform as a Service (PaaS) - Heroku, Google App Engine

- Outsourced hardware and software infrastructure.
- Offers hardware and software tools over the internet, usually for application development. PaaS provides a platform allowing customers to develop, run, and manage applications without the complexity of building and maintaining infrastructure.

Software as a Service (SaaS) - Microsoft 365

- Outsourced applications delivered over the internet.
- Applications are hosted by a vendor or service provider and made available to customers over the Internet.

Business Process Re-Engineering	Business Process Management
Implementation is radical and one-step change.	Implementation is evolutionary and continuous.
Implementation time is long.	Implementation time is short and smooth takeover.
One major process at a time.	It is flexible.
Business and process experts involved.	Process experts and all related people involved.
Risk is high.	Risk is low.
Outcome sometimes is drastic.	Outcome is improved.
Cultural issues are a major concern.	Cultural issues are not much concern.
Implementation stress and concern is high.	Implementation stress and concern is low.

Data processing

Data Mining: Can be used to sort through data in order to discover previously unknown information. Data mining is the process of finding anomalies, patterns and correlations within large data sets to predict outcomes. Using a broad range of techniques, you can use this information to increase revenues, cut costs, improve customer relationships, reduce risks and more. Data mining is one form of data analytics, and it assists managers in making decisions and strategic plans.

Data Warehouse: Specially designed for data analytics, which involves reading large amounts of data to understand relationships and trends across the data.

Real-Time Processing: The processing of each transaction as it occurs, providing near-instantaneous output.

Robotic process automation (RPA): software bots do repetitive, rule-based tasks (e.g., invoice matching); control bot changes, bot credentials and access like any user. Outsourcing moves a process to a service organization, but the entity keeps responsibility and monitors the provider (e.g., through System and Organization Controls (SOC) reports).

Data processing (continued)

System Documentation: Includes things like source code, testing documentation and API documentation (programmers' documentation or instructions). It describes the requirements and capabilities of the software and informs the reader about what the software can and can't do – in other words, its functionality.

API (Application Programming Interface): A set of definitions and protocols for building and integrating application software. An API delivers a request from one application to another system and sends that system's response back.

Data extraction methods: SQL (Structured Query Language) queries, APIs, report or flat-file export, and ETL (extract, transform, load) into a data warehouse. Check extracted data for completeness and accuracy (record counts, control totals).

IT attacks

Malware

A file or code, typically delivered over a network, that infects, explores, steals or conducts virtually any behavior an attacker wants.

Denial-of-Service (DoS) attack

A malicious, targeted attack that floods a network with false requests in order to disrupt business operations. In a DoS attack, users are unable to perform routine and necessary tasks, such as accessing email, websites, online accounts or other resources that are operated by a compromised computer or network. While most DoS attacks do not result in lost data and are typically resolved without paying a ransom, they cost the organization time, money and other resources in order to restore critical business operations.

Trojan Horse

Malware disguised as or hidden in a legitimate-looking program that infects a system while appearing to behave as expected.

Physical Intrusion

A breach of physical controls such as an intruder sneaking into an office to steal data.

Probability using data

Expected Monetary Value (EMV) of a Risk Event = (Cost of Damages x Probability of Risk) + (Cost with no Damages x Probability of No Risk)

Given

XYZ Company helps to manage its risk by utilizing risk analysis. For each new project, a risk probability is estimated. For Project A, the company has estimated a risk probability of .20 that it will see significant damages. Project A has a budget of $50,000 and if the risk occurs, the cost will be $15,000 to remedy the damages. What is the expected monetary value of this risk event?

Solution

20% x $15,000 + 80% x $0 = $3,000

IT controls

General Controls

Ensure that an organization's IT environment is well managed and stable. These include information technology infrastructure, security management, and software acquisition. Convenience is not a priority when establishing security controls.

Security Testing

Verifies that system protection mechanisms prevent improper penetration or data alteration.

Audit Trail

Detailed, chronological record whereby accounting records, project details, or other financial data are tracked and traced. Used to verify and track all kinds of transactions, work processes, accounting details, trades in brokerage accounts, and more.

IT controls (continued)

Reasonableness Check

Checks whether data is logical when compared with other data (e.g., hours worked versus pay). It is a common-sense check that you can use to ensure that your data is what is expected.

Batch Totals vs Hash Totals

Both are methods of validating data. Batch totals use a meaningful field (e.g., dollar amount, record count, quantity). Hash totals add fields that normally aren’t summed (e.g., invoice #s) to detect missing, extra, or altered records (e.g., transpositions).

Reviewing the Systems-Access Log

A way to detect fraud.

Cold, Warm and Hot Disaster Recovery Models

Type of Site	Cold Site	Warm Site	Hot Site
Secondary Location	Yes	Yes	Yes
Equipment at Location		Yes	Yes
Connectivity at Location		Yes	Yes
Active before Incident			Yes
Incident Outage Measured in	***WEEKS***	***DAYS/HOURS***	***HOURS/MINUTES***

These sites exist to allow a business to continue despite any damage to the existing workplace.

IT security terms

Encryption - The method by which information is converted into secret code that hides the information's true meaning. The science of encrypting and decrypting information is called cryptography. In computing, unencrypted data is also known as plaintext, and encrypted data is called ciphertext.

Digital Signature - A mathematical algorithm routinely used to validate the authenticity and integrity of a message (e.g., an email, a credit card transaction, or a digital document). It is created with the signer's private key and verified with the signer's public key, which also supports nonrepudiation.

Virtual Private Network (VPN) - Creates an encrypted communication tunnel across the Internet for the purpose of allowing a remote user secure access into the network.

Physical Controls - Controls over physical access to IT assets, mostly preventive, such as biometric devices (fingerprint scanner), key cards, and locks.

Logical Controls/Technical Controls - May include user log-ins, data encryption, antivirus software, and firewalls.

Administrative Controls - Includes administrative activities such as separation of duties and business continuity planning.

Firewall - Network security device that monitors and filters incoming and outgoing network traffic. Barrier that sits between a private internal network and the public Internet.

1. Operating Systems (OS):

- **Purpose**: Operating systems act as the backbone of IT systems, providing a platform for applications to run and facilitating interactions between hardware and software. They manage hardware resources and offer essential services to support various applications.
- **Examples**: Windows Server and Linux distributions (such as Ubuntu, Red Hat) for servers. Windows 11, macOS, Linux distributions (like Ubuntu, Fedora), and Android or iOS for mobile devices.

2. Servers:

- **Purpose**: Servers are powerful computers that provide resources and services to other computers over a network. They are designed to manage, store, send, and process data 24/7 in a secure and reliable manner.
- **Examples**: Web servers like Apache HTTP Server and Nginx that host websites. Database servers such as Microsoft SQL Server and Oracle Database that store and manage data.

Understanding Key Components of IT Architecture (continued)

3. Network Infrastructure:

- **Purpose**: The network infrastructure consists of hardware and software resources that facilitate connectivity, communication, operations, and management of an enterprise network. It ensures secure and reliable data transmission between various IT components.
- **Examples**: Physical devices like routers, switches, and modems. Network protocols (TCP/IP), firewalls for security, and wireless access points.

4. End-User Devices:

- **Purpose**: End-user devices are the interface through which users interact with the IT environment. They enable users to access, input, and manipulate data in the IT system.
- **Examples**: Personal computers (desktops, laptops), smartphones, and tablets. Specialized devices like point-of-sale terminals and barcode scanners.

Understanding Cloud Computing: Models and Deployment Types

Cloud computing has revolutionized the way businesses and individuals use technology. It involves delivering various services over the internet, including data storage, servers, databases, networking, and software. Central to cloud computing are its service models and deployment models, which offer different levels of control, flexibility, and management.

Cloud Computing Service Models:

Infrastructure as a Service (IaaS):

- **Description**: Provides virtualized computing resources over the internet. IaaS is the most basic and each component (like servers, storage, and networking) is provided as a separate service.
- **Control**: Users control operating systems, storage, and deployed applications but don't manage the underlying cloud infrastructure.
- **Examples**: Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform.

Platform as a Service (PaaS):

- **Description**: Offers hardware and software tools over the internet, usually for application development. PaaS provides a platform allowing customers to develop, run, and manage applications without the complexity of building and maintaining infrastructure.
- **Control**: Users control the deployed applications and possibly the application hosting environment configurations.
- **Examples**: Heroku, Google App Engine, Microsoft Azure App Service.

Software as a Service (SaaS):

- **Description**: Delivers software applications over the internet, on-demand, and typically on a subscription basis. With SaaS, cloud providers host and manage the software application and underlying infrastructure.
- **Control**: Users don’t manage the infrastructure or platform but can configure limited user-specific settings.
- **Examples**: Google Workspace, Microsoft 365, Salesforce.

Cloud Computing Deployment Models (NIST adds Community Cloud: shared by organizations with common concerns):

Public Cloud:

- **Description**: Services are provided over the public internet and are available to anyone who wants to use or purchase them. They are owned and operated by third-party cloud service providers.
- **Use Cases**: Suits organizations of any size, especially applications with variable demand.
- **Examples**: AWS, Azure, Google Cloud.

Private Cloud:

- **Description**: The cloud infrastructure is used exclusively by a single business or organization. The private cloud can be located on the company's on-site data center or hosted by a third-party service provider.
- **Use Cases**: Suitable for businesses with high data security, privacy, and control requirements.
- **Examples: VMware Cloud Foundation, OpenStack.**

Hybrid Cloud:

- **Description**: Combines two or more distinct clouds (typically public and private), bound together by technology that allows data and applications to be shared between them. It gives businesses greater flexibility and more deployment options.
- **Use Cases**: Ideal for businesses that want to maintain a private infrastructure for sensitive processes or data while leveraging public cloud for less-critical resources.
- **Examples**: Hybrid deployments using AWS and on-premises resources, Azure Arc.

Shared responsibility, community cloud and connected systems

- Shared responsibility: the provider secures the layers it supplies (facilities, hardware, virtualization, and more as you move from infrastructure to platform to software as a service); the customer always keeps responsibility for its data, user access and configurations, and outsourcing does not transfer accountability. COSO's Enterprise Risk Management for Cloud Computing applies the enterprise risk management components to cloud governance.

- Community Cloud: shared by several organizations with common concerns (e.g., mission, security or compliance requirements); managed by them or a third party.

- An accounting information system (AIS) may be a module of the enterprise resource planning (ERP) system or a separate system connected to it. Other systems they interact with: electronic data interchange (EDI), the computer-to-computer exchange of business documents such as purchase orders and invoices with trading partners; and blockchain, a shared, append-only distributed ledger whose records are linked cryptographically.

Understanding Enterprise Resource Planning (ERP) and Accounting Information Systems: Encompassment and Interaction

Enterprise Resource Planning (ERP) Systems: ERP systems are comprehensive software platforms used by organizations to manage and integrate various business processes. They cover a wide range of functions such as finance, human resources, supply chain management, customer relationship management, and more.

Key Features:

- **Integration**: ERP systems integrate various business processes into a single unified system, enhancing data accuracy and facilitating decision-making.
- **Automation**: Many routine tasks are automated, improving efficiency and reducing the likelihood of errors.
- **Real-Time Data**: Provides real-time insights into business operations, enabling quick responses to changes in the business environment.

Benefits: Streamlines business processes, improves data accuracy, enhances productivity, and provides strategic insights through data analytics.

Accounting information systems (AIS)

Accounting Information Systems (AIS): AIS are specialized subsystems within an organization's overall information system, specifically designed to manage financial data. They handle tasks such as accounting, payroll, accounts receivable/payable, and financial reporting.

Key Features:

- **Data Collection and Processing**: Efficiently collects and processes financial data.
- **Financial Reporting**: Generates accurate financial reports for internal and external use.
- **Regulatory Compliance**: Supports compliance with financial regulations and standards.

Benefits: Enhances the accuracy of financial data, streamlines financial processes, and supports compliance with accounting standards and regulations.

Management Assertions in SOC Engagements

SOC 1® Engagements

- **Focus**: On controls at a service organization likely to be relevant to the user entities' internal control over financial reporting.
- **Type 1**: Management asserts that the description of the system is fairly presented and that the controls are suitably designed to achieve the control objectives stated in the description as of a specific date.
- **Type 2**: Same assertions, but covering a period rather than a date, and management also asserts that the controls were operating effectively over a specified period.

Management Assertions in SOC Engagements (continued)

SOC 2® Engagements

- **Focus**: On controls at a service organization relevant to security, availability, processing integrity, confidentiality, and privacy.
- **Type 1**: Management asserts that the description of the system is presented in accordance with the description criteria and that the controls were suitably designed to provide reasonable assurance that the service commitments and system requirements would be achieved based on the applicable Trust Services Criteria, as of a specific date.
- **Type 2**: Similar to SOC 1® Type 2, with an additional assertion that the controls operated effectively over a defined period to achieve the service commitments and system requirements based on the applicable Trust Services Criteria.

SOC 3® Engagements

- **Focus**: Similar to SOC 2®, but intended for a wider audience with a general report on controls related to the Trust Services Criteria without detailed descriptions.
- **SOC 3® Report Basis:** A SOC 3® report is a general-use report on the same Trust Services Criteria subject matter as a SOC 2® Type 2 examination and is often issued alongside one. It provides assurance on the effectiveness of controls over a period but does not include the detailed descriptions of tests and results found in a SOC 2® report. There is not a "Type 1" equivalent for a SOC 3® report.

Trust Services Criteria and SOC for Cybersecurity

- Service commitments: management's promises to user entities about the system (e.g., in contracts or service level agreements). System requirements: the specifications for how the system must work to meet those commitments and applicable laws and regulations.
- Trust Services Criteria: the common criteria (CC1 to CC9) apply to every category; CC1 to CC5 follow the 17 COSO 2013 principles, and CC6 to CC9 add supplemental criteria for logical and physical access, system operations, change management and risk mitigation. Availability, processing integrity, confidentiality and privacy add their own criteria. Points of focus guide judgment; they are not a required checklist.
- The Trust Services Criteria are used in SOC 2 and SOC 3 examinations, in SOC for Cybersecurity (an entity's cybersecurity risk management program), and in other attest engagements on an entity's systems or controls.
- SOC 1 reports are restricted use: management of the service organization, user entities, and the user entities' financial statement auditors.
- SOC for Cybersecurity: a general-use examination of an entity's cybersecurity risk management program (management's description and whether its controls achieved the entity's cybersecurity objectives). Users include management, boards, investors, analysts, business partners and regulators. It covers the entity (or a unit of it), not services provided to user entities.

Intended Users of SOC 1®, SOC 2®, and SOC 3® Reports

System and Organization Controls (SOC) reports are designed for different user groups, based on the nature of the report and the information it contains. Here's an overview of the intended users for SOC 1®, SOC 2®, and SOC 3® reports:

SOC 1® Reports

- **Primary Users**: These reports are primarily intended for the auditors of the user entities (the entities that use the services of the service organization).
- **Purpose**: SOC 1® reports are used in the audit of a user entity’s financial statements, particularly for assessing the impact of the service organization’s controls over financial reporting.
- **User Entities' Management**: Management of the user entities also uses these reports for oversight and vendor management purposes.

Intended Users of SOC 1®, SOC 2®, and SOC 3® Reports (continued)

SOC 2® Reports

- **Broader Audience**: While still somewhat technical, SOC 2® reports are intended for a broader audience than SOC 1® reports.
- **Specific Users**: SOC 2 reports are restricted-use reports for specified parties with sufficient knowledge of the service organization's system and controls, such as management of the service organization and of user entities, prospective user entities, business partners, the practitioners who serve them, and regulators, who need assurance about the service organization's controls related to security, availability, processing integrity, confidentiality, or privacy.
- **Versatility in Use**: They are often used in risk management decisions, vendor management programs, and internal corporate governance and risk management processes.

SOC 3® Reports

- **General Users**: SOC 3® reports are designed for a wide range of users who need assurance about the controls at a service organization but do not require the detailed information contained in a SOC 2® report.
- **Public Distribution**: These reports can be freely distributed and are often used for marketing purposes, such as on a company’s website, to demonstrate the organization’s commitment to maintaining a strong control environment.
- **Lay Audience Appeal**: They are particularly useful for users who do not have the need or expertise to understand the detailed content of a SOC 2® report.

Independence Considerations in SOC Engagements

In System and Organization Controls (SOC) engagements, the service auditor's independence from the service organization (and any included subservice organization) is crucial for the integrity and reliability of the examination.

Service Auditor

- **Objective Stance**: The service auditor must be independent in both fact and appearance. This means avoiding prohibited financial, employment, or managerial relationships and reducing other threats to an acceptable level.
- **Professional Standards**: Adherence to professional standards, such as those set by the American Institute of Certified Public Accountants (AICPA), is mandatory to maintain the auditor’s independence.

Independence Considerations in SOC Engagements (continued)

Service Organization

- **No Influence Over Audit**: The service organization should not have any influence over the auditor’s work. Management defines the system and agrees on the engagement scope, but it cannot direct the auditor's procedures, findings, or reporting.
- Independence: The service auditor must be independent of the service organization and may not perform management functions, such as designing or operating the controls being examined.

Subservice Organizations

- **Auditor's Independence from Subservice Organizations**: When the inclusive method is used, the auditor must also be independent of the subservice organization; under the carve-out method, independence from the subservice organization is not required.
- **Reporting Methods**: The choice of the inclusive or carve-out method for subservice organizations in the SOC report should not compromise the auditor’s independence.
- **Inclusive Method**: When management elects the inclusive method, the service auditor also tests the subservice organization's controls and obtains its written assertion.

Materiality, Risk and Written Representations

- Materiality in a SOC examination: an omission, misstatement or deficiency is material if it could reasonably influence users' decisions; consider qualitative as well as quantitative factors.
- Risk assessment: management identifies risks that threaten the control objectives (SOC 1) or service commitments and system requirements (SOC 2) and designs controls to address them. The service auditor separately assesses the risk that the description is not fairly presented or the controls are not suitably designed or operating effectively, and plans procedures in response.
- Written representations: the service auditor obtains a letter from management, dated as of the report date, confirming its assertion and responsibilities and that all relevant information was provided, and disclosing known deficiencies, fraud, noncompliance and subsequent events. Representations support other evidence; they do not replace it.

Impact of Subsequently Discovered Facts on SOC Engagements

In System and Organization Controls (SOC) engagements, including SOC 1® and SOC 2®, the discovery of new information or facts after the date of the report can have significant implications.

Nature of Subsequently Discovered Facts: These are facts that existed at the date of the service auditor's report and, had they been known at that date, may have caused the service auditor to alter the report. Such facts could include errors in the reported information, breaches or failures in controls, or non-compliance with the criteria outlined in the SOC report.

Impact on SOC Engagements

Reevaluation of the Report:

- The service auditor must reassess the conclusions drawn in the report in light of the new information.
- This could lead to a revision of the auditor’s opinion if the new facts significantly affect the report's accuracy or the conclusions drawn.

Communication with the Service Organization:

- The auditor is required to discuss the new information with the service organization to understand its implications fully.
- If the report needs revision, the service organization should inform users who are relying on it; if the service organization does not, the service auditor takes steps to prevent further reliance on the report (often after seeking legal advice).

SOC 1® vs SOC 2® Engagements: Effect of Subsequent Events

Types of Subsequent Events

Subsequent events generally fall into two categories:

Events about conditions that existed during the period covered:

- These events provide additional evidence about conditions that existed during the period covered by the description (or at its as-of date for a Type 1) and can affect the service auditor's understanding of the system or the operating effectiveness of controls.
- Example: Discovery of a control deficiency in the control system that existed during the period covered by the system description.

Events about conditions that arose after the period covered:

- These are events that provide evidence about conditions that arose after the period covered by the system description but before the date of the service auditor's report.
- Example: A new line of service introduced by the service organization after the period covered by the system description.

SOC 1® vs SOC 2® Engagements: Effect of Subsequent Events

Procedures for Evaluating Subsequent Events

Inquiry and Documentation

- The service auditor should inquire of management about any significant subsequent events.
- Review documentation or evidence that can support management's responses.

Analyzing the Impact:

- Evaluate how these events affect the service organization's system and controls.
- For SOC 1®, focus on the impact on user entities' internal control over financial reporting.
- For SOC 2®, assess the effect on the service commitments and system requirements and the applicable Trust Services Criteria.

Adjustments and Disclosure:

- Determine if the system description requires revision or if additional disclosures are needed in the service auditor’s report.
- For significant subsequent events that affect the service auditor's conclusion, the service auditor must address them in the report, e.g., by modifying the opinion or adding a paragraph.

Reporting Subsequent Events

- Management discloses significant subsequent events in its description.
- If an event must be disclosed to keep users from being misled and management has not disclosed it, the service auditor discloses it in the service auditor's report; if management refuses to disclose an event, the auditor may also modify the opinion or withdraw.

CUECs Effect on SOC Reports (SOC 1®, SOC 2®)

Complementary User Entity Controls (CUECs): These are controls that the service organization assumes, in the design of its service, will be implemented by the user entities (the organizations that use the service). They are necessary, together with the service organization's controls, to achieve the control objectives (SOC 1) or service commitments and system requirements (SOC 2).

Effect on SOC 1® Reports

- **SOC 1® Report**: This report focuses on the internal control over financial reporting. It's used primarily by auditors to assess the impact of a service organization's controls on a user entity's financial statements.
- **CUECs in SOC 1®**: In a SOC 1® report, CUECs are identified as controls that a user organization needs to implement so that the control objectives in the service organization's description can be achieved, as they relate to the user's financial reporting. The absence or failure of CUECs at the user entity could lead to misstatements in their financial reports.

Effect on SOC 2® Reports

- **SOC 2® Report**: This report is based on the Trust Services Criteria and focuses on controls other than those over financial reporting, related to security, availability, processing integrity, confidentiality, and privacy.
- **CUECs in SOC 2®**: For a SOC 2® report, CUECs are necessary, together with the service organization's controls, to achieve its service commitments and system requirements based on the applicable Trust Services Criteria. They ensure that the system's security, availability, processing integrity, confidentiality, and privacy are managed appropriately at the user end. If CUECs are not appropriately implemented, the service organization's efforts in these areas may be compromised.

Carve-Out vs. Inclusive Method for Subservice Organizations

In the context of System and Organization Controls (SOC) reports, specifically SOC 1 and SOC 2 reports, the treatment of subservice organizations (third-party service providers used by the primary service organization) can follow either the "carve-out" or "inclusive" method. These methods determine how the controls of subservice organizations are reported; carve-out reports list Complementary Subservice Organization Controls (CSOCs): controls the service organization assumes the subservice organization will implement.

Carve-Out Method

- **Definition**: In the carve-out method, the primary service organization excludes (or "carves out") the controls of the subservice organization from its own system description.
- **Reporting**: The service organization's SOC report describes the services the subservice organization performs, the CSOCs it assumes are in place, and its own controls for monitoring the subservice organization, but it does not include or test the subservice organization's controls.
- **Responsibility**: It's implied that the user entities will obtain assurance directly from the subservice organizations about the effectiveness of their controls, or assume additional risk.
- **Advantage**: Simplifies the primary service organization’s SOC report as it does not need to describe or evaluate the controls of the subservice organization.
- **Consideration**: The user entities need to understand that they must consider the controls at the subservice organization separately, either by performing their own due diligence or obtaining a separate SOC report for the subservice organization.

Carve-Out vs. Inclusive Method for Subservice Organizations (continued)

Inclusive Method

- **Definition**: In the inclusive method, the service organization includes the controls of the subservice organization within its own SOC report.
- **Reporting**: The SOC report provides information about the subservice organization's controls and how they are monitored by the primary service organization.
- **Assurance**: This method provides the user entities with assurance about the controls at both the service organization and its subservice organizations (design, and for a Type 2 report, operating effectiveness).
- **Advantage**: Gives a more comprehensive view of the control environment affecting the services used by the user entities.
- **Consideration**: This approach can be more complex and resource-intensive as it requires the primary service organization to understand, monitor, and report on the controls of its subservice organizations.

Carve-Out vs. Inclusive Method for Subservice Organizations (continued)

- A vendor is a subservice organization only if the service organization uses it to perform services likely to be relevant to user entities' internal control over financial reporting (SOC 1) or to its service commitments and system requirements (SOC 2), and its controls are needed, with the service organization's controls, to achieve the control objectives or commitments. Not every vendor qualifies.
- Under the inclusive method, subservice organization management also provides a written assertion and representations, the service auditor tests the included subservice controls, and the auditor must be independent of the subservice organization.
- CUECs and the report: the description lists the complementary user entity controls (CUECs), the opinion says the controls provide reasonable assurance the control objectives would be achieved (Type 2: were achieved) if user entities applied the CUECs assumed in the design, and the service auditor does not test CUECs; user entities and their auditors do.

Appropriate Form and Content of a SOC 1® and SOC 2® Report

System and Organization Controls (SOC) reports, specifically SOC 1® and SOC 2®, are structured to provide detailed information about the control environment of a service organization. Below are guidelines for determining the appropriate form and content of these reports:

Common Elements for Both SOC 1® and SOC 2® Reports

Title Page: Includes the title of the report, the name of the service organization, and the date (Type 1) or period (Type 2) covered.

Table of Contents: Lists the major sections of the report and page numbers for easy navigation.

Independent Service Auditor's Report:

- **Introduction**: States that the report is an examination of controls.
- **Management's Assertion**: Summary of the service organization's assertion about the fair presentation of the description and the suitability of design (and, for Type 2, the operating effectiveness) of its controls.
- **Auditor's Opinion**: The auditor's opinion on whether the description is fairly presented, the controls are suitably designed, and, for a Type 2 report, the controls operated effectively.

Appropriate Form and Content of a SOC 1® and SOC 2® Report (continued)

Service Organization's Assertion: A statement from management asserting the fairness of the presentation of the description, the suitability of the design of the controls and, for a Type 2 report, their operating effectiveness.

Description of the System: Detailed description of the service organization's system, including its services and controls (for SOC 2: infrastructure, software, people, procedures, and data).

Description of Tests of Controls and Results (Type 2 reports only): Detailed account of the tests conducted, including methodologies, samples, and results.

Other Information Provided by the Service Organization: Any additional information that the service organization wishes to include. It is not covered by the auditor's opinion.

Appropriate Form and Content of a SOC 1® and SOC 2® Report (continued)

SOC 1® Report Specifics

- **Focus**: Specifically addresses controls relevant to user entities' internal control over financial reporting.
- **Users**: Restricted to management of the service organization, user entities, and the auditors of the user entities' financial statements.

SOC 2® Report Specifics

- **Focus**: Addresses controls related to security, availability, processing integrity, confidentiality, and privacy based on the Trust Services Criteria.
- **Users**: Restricted to specified parties with sufficient knowledge of the service organization's system, such as management of the service organization and of user entities, business partners, and regulators.

Appropriate Form and Content of a SOC 1® and SOC 2® Report (continued)

Additional Elements for SOC Reports

- **Management's Description of the Service Organization's System**: A narrative description of the service organization's system based on criteria set forth in the AT-C 320 (SOC 1) or DC 200 (SOC 2).
- **Control Objectives and Control Activities**: For SOC 1®, this includes specific control objectives and activities that relate to internal control over financial reporting. For SOC 2®, it includes the Trust Services Criteria relevant to the scope of the examination.
- **Information About the Subservice Organizations**: Inclusion or exclusion of subservice organizations' controls (inclusive or carve-out method).
- **Other Information**: Additional information the service organization chooses to include; not covered by the opinion.

The form and content of SOC 1® and SOC 2® reports are designed to provide a comprehensive view of the control environment at a service organization. These reports are essential tools for user entities to assess the risks associated with outsourcing services and to make informed decisions. The precise structure of these reports is governed by the American Institute of Certified Public Accountants (AICPA) standards and guidelines.

Appropriate Form and Content of a SOC 1® and SOC 2® Report (continued)

- Opinion types: unmodified; qualified (a material but not pervasive deficiency or scope limitation); adverse (material and pervasive deficiencies); disclaimer (unable to obtain sufficient appropriate evidence, with possible effects that are material and pervasive). A modified report explains the matter in a basis-for-modification paragraph.
- Type 2 test results show, for each control, the test performed, its nature, timing and extent, and the results, including the number and nature of any deviations (exceptions), even when the auditor concludes the objective was still achieved.
- SOC 2 system description (DC1 to DC9): types of services; principal service commitments and system requirements; system components (infrastructure, software, people, procedures, data) and boundaries; significant system incidents; applicable criteria and related controls; complementary user entity controls (CUECs); subservice organizations and complementary subservice organization controls (CSOCs); criteria not relevant and why; and, for Type 2, significant changes during the period.
- The service organization tells users how to report system failures, incidents, concerns and complaints (Trust Services Criteria CC2.3, communication with external parties).

Health Insurance Portability & Accountability Act (HIPAA)

The Health Insurance Portability and Accountability Act (HIPAA) established standards for protecting sensitive patient health information. Its key rules are the Privacy Rule, the Security Rule and the Breach Notification Rule. Understanding the covered entities and the permitted uses and disclosures under these rules is critical for compliance.

HIPAA Privacy Rule

Covered Entities

- **Healthcare Providers**: Any provider of medical or health services that transmits any health information in electronic form in connection with transactions for which HHS has adopted standards.
- **Health Plans**: Entities that provide or pay the cost of medical care, including health insurance companies, HMOs, company health plans, and government programs like Medicare and Medicaid.
- **Healthcare Clearinghouses**: Entities that process nonstandard health information they receive from another entity into a standard format or vice versa.

HIPAA Privacy Rule

Permitted Uses and Disclosures

Treatment, Payment, and Healthcare Operations:

- **Treatment**: Sharing information for the purpose of providing, coordinating, or managing healthcare and related services.
- **Payment**: Use and disclosure of protected health information (PHI) for activities such as obtaining premiums or determining coverage.
- **Healthcare Operations**: Activities necessary for the covered entity to run its business and ensure quality of care (e.g., quality assessment, employee review, training medical students).

Health Insurance Portability & Accountability Act (HIPAA) (continued)

Opportunity to Agree or Object:

- Informal permission through opportunity to agree or object can be obtained in instances like sharing with friends, family, or others involved in the individual's care.

Incident To a Permitted Use:

- Uses and disclosures incidental to an otherwise permitted use or disclosure, provided reasonable safeguards and the minimum necessary standard are applied.

Public Interest and Benefit Activities:

- Including public health activities, reporting victims of abuse or neglect, law enforcement purposes, and other judicial and administrative proceedings.

Health Insurance Portability & Accountability Act (HIPAA) (continued)

HIPAA Security Rule

Covered Entities

- **Same as the Privacy Rule**: Healthcare Providers, Health Plans, and Healthcare Clearinghouses. The Security Rule covers electronic PHI only, applies to business associates as well as covered entities, and protects its confidentiality, integrity and availability.

Required Protections

- **Administrative Safeguards**: Policies and procedures designed to clearly show how the entity will comply with the act (e.g., training programs, incident response procedures).
- **Physical Safeguards**: Controlling physical access to protect against inappropriate access to protected data (e.g., facility access controls, workstation security).
- **Technical Safeguards**: Controlling access to computer systems and enabling covered entities to protect communications containing PHI transmitted electronically over open networks from being intercepted by anyone other than the intended recipient (e.g., access control, audit controls, integrity controls, transmission security).

Health Insurance Portability & Accountability Act (HIPAA) (continued)

- PHI is protected in any form (electronic, paper or oral). Business associates (vendors that create, receive, maintain or transmit PHI for a covered entity) are generally not covered entities (a covered entity can also act as a business associate of another) but must sign a business associate agreement and are directly liable under the Security Rule and parts of the Privacy Rule.

- Also permitted: disclosure to the individual, and a limited data set for research, public health or health care operations under a data use agreement. Any other use or disclosure needs the individual's written authorization. Most uses and disclosures are limited to the minimum necessary (not required for disclosures to a provider for treatment).

- Breach Notification Rule: after a breach of unsecured PHI, notify affected individuals without unreasonable delay (no later than 60 days after discovery) and the Department of Health and Human Services (HHS) (at the same time if 500 or more individuals are affected; otherwise in an annual log within 60 days after year end); a breach affecting more than 500 residents of a state or jurisdiction also requires notice to prominent media. Business associates notify the covered entity.

GDPR & PCI DSS

The General Data Protection Regulation (GDPR) is a comprehensive data protection law that applies across the European Union (EU) and the European Economic Area (EEA). It also impacts organizations outside these areas if they offer goods or services to, or monitor the behavior of, individuals in the EU/EEA. GDPR is designed to give individuals control over their personal data and to unify data privacy laws across Europe.

Scope of the GDPR

- **Geographical Reach**: Applies to organizations processing personal data in the context of an establishment in the EU/EEA, wherever the processing takes place, and to organizations outside these regions if they offer goods or services to, or monitor the behavior of, individuals in the EU/EEA.
- **Personal Data Definition**: Covers a broad spectrum of data that can be used to directly or indirectly identify an individual (e.g., name, photos, email addresses, bank details, social networking posts, medical information, IP addresses).

The Payment Card Industry Data Security Standard (PCI DSS) is a set of security standards designed to ensure that all companies that accept, process, store, or transmit payment card (credit and debit) account data maintain a secure environment. It applies to all entities involved in payment card processing, including merchants, processors, acquirers, issuers, and service providers. It also applies to any other entity that stores, processes, or transmits cardholder data or sensitive authentication data, or that could affect the security of the cardholder data environment.

GDPR & PCI DSS (continued)

- GDPR six principles: lawfulness, fairness and transparency; purpose limitation; data minimisation; accuracy; storage limitation; integrity and confidentiality. The controller must also be able to show compliance (accountability).
- Key concepts: a controller decides why and how personal data is processed; a processor processes it on the controller's behalf. Six lawful bases: consent, contract, legal obligation, vital interests, public task, legitimate interests. Data subject rights include access, rectification, erasure, restriction, portability and objection.
- PCI DSS v4.0.1, 12 principal requirements: (1) network security controls; (2) secure configurations; (3) protect stored account data; (4) strong cryptography over open, public networks; (5) anti-malware; (6) secure systems and software; (7) restrict access by business need to know; (8) identify users and authenticate access; (9) restrict physical access; (10) log and monitor all access; (11) test security regularly; (12) information security policies and programs.
- Breach notification: the controller notifies the supervisory authority without undue delay and, where feasible, within 72 hours of becoming aware, unless the breach is unlikely to risk individuals' rights; high-risk breaches are also communicated to affected individuals. Processors notify the controller.

NIST Cybersecurity Framework (CSF) & NIST SP 800-53

The National Institute of Standards and Technology (NIST) Cybersecurity Framework (CSF) is designed to help organizations manage and reduce cybersecurity risk. It is a voluntary framework designed for organizations of any size or sector; CSF 2.0 expanded its scope beyond critical infrastructure. The NIST CSF consists of three main components: the CSF Core, CSF Organizational Profiles, and CSF Tiers.

Purpose of NIST **SP 800-53:** NIST Special Publication 800-53 Revision 5, titled "Security and Privacy Controls for Information Systems and Organizations," provides a catalog of security and privacy controls for information systems and organizations. The primary purpose of this publication is to promote the development of secure and resilient information systems. NIST SP 800-53 helps in the selection of appropriate security controls to protect organizational operations, assets, individuals, other organizations, and the nation from a diverse set of threats including hostile cyber attacks, natural disasters, structural failures, and human errors.

Applicability

- **Federal Information Systems**: Mandatory for use by federal agencies (FISMA, FIPS 200) in developing security and privacy controls for federal information systems and organizations.
- **Beyond Federal Use**: While tailored for federal agencies, the guidance is also widely adopted by private sector organizations and non-federal information systems due to its comprehensive nature and relevance in addressing cybersecurity risks.

NIST Cybersecurity Framework (CSF) & NIST SP 800-53 (continued)

- Core: outcomes organized into Functions, Categories and Subcategories. Organizational Profiles: an organization's current and target outcomes. Tiers: how rigorous its cybersecurity risk governance and management are (Partial, Risk Informed, Repeatable, Adaptive).
- SP 800-53 target audience: people with system development, acquisition, management, oversight, security and privacy, and assessment and monitoring duties (e.g., authorizing officials, chief information officers, chief information security officers and senior privacy officials). Organizations select and tailor controls to risk (using the SP 800-53B baselines), implement them, and assess and monitor them. The controls are grouped into 20 families.
- NIST Privacy Framework: three parts, the Core (privacy outcomes in five functions: Identify-P, Govern-P, Control-P, Communicate-P, Protect-P), Profiles (current or target outcomes) and Implementation Tiers. Privacy risk can arise from authorized data processing, not only from cybersecurity incidents.

NIST CSF 2.0 Core Functions

The Framework Core organizes cybersecurity activities into six functions. Govern was added in CSF 2.0 and is cross-cutting: it informs how the other five functions are carried out.

- Govern (GV): Establish, communicate, and monitor cybersecurity risk management strategy, expectations, and policy. Includes organizational context, risk management strategy, roles & responsibilities, policy, oversight, and supply chain risk management. Govern is the foundation; it sets the tone for everything below.
- Identify (ID): Understand the organization's cybersecurity risk to systems, people, assets, and data. Includes asset management, risk assessment, and improvement (learning from prior incidents and assessments).
- Protect (PR): Use safeguards to manage the organization's cybersecurity risks. Includes identity management & access control, awareness & training, data security, platform security, and technology infrastructure resilience.
- Detect (DE): Find and analyze possible cybersecurity attacks and compromises. Includes continuous monitoring and adverse event analysis.
- Respond (RS): Take action regarding a detected cybersecurity incident. Includes incident management, incident analysis, incident response reporting & communication, and incident mitigation.
- Recover (RC): Restore assets and operations affected by a cybersecurity incident. Includes incident recovery plan execution and incident recovery communication.

Memory aid: "Good Ideas Prevent Dangerous Risks Rising" → Govern, Identify, Protect, Detect, Respond, Recover

CIS Controls & COBIT 2019

The CIS Controls, developed by the Center for Internet Security, are a set of best practices designed to help organizations protect against the most prevalent cyber threats. The CIS Controls are widely respected for their straightforward and prioritized approach to cybersecurity.

COBIT (Control Objectives for Information and Related Technologies) 2019 is a framework for developing, implementing, monitoring, and improving IT governance and management practices. It is widely used for aligning IT with business goals, managing risks, and ensuring effective resource utilization. The COBIT framework is built on six governance system principles and three governance framework principles, and defines the components of a governance system.

CIS Controls & COBIT 2019 (continued)

CIS Controls v8.1 (18):	
1 inventory of enterprise assets;	10 malware defenses;
2 inventory of software assets;	11 data recovery;
3 data protection;	12 network infrastructure management;
4 secure configuration;	13 network monitoring and defense;
5 account management;	14 security awareness and skills training;
6 access control management;	15 service provider management;
7 continuous vulnerability management;	16 application software security;
8 audit log management;	17 incident response management;
9 email and web browser protections;	18 penetration testing.

COBIT 2019 governance system principles (6):	provide stakeholder value; holistic approach; dynamic governance system; governance distinct from management; tailored to enterprise needs; end-to-end governance system.
Governance framework principles (3):	based on a conceptual model; open and flexible; aligned to major standards.
Components (7):	processes; organizational structures; principles, policies and procedures; information; culture, ethics and behavior; people, skills and competencies; services, infrastructure and applications.

Objectives of Mirroring and Replication in Data Management

In the context of data management and information technology, mirroring and replication are two strategies used to ensure data availability, reliability, and continuity. Although they share some similarities, each has specific objectives and is suited to particular scenarios. Here's an explanation of the objectives of both mirroring and replication:

Mirroring

Objective: Real-Time Data Redundancy

- **Description**: Mirroring involves creating an exact copy of a data set or database in real-time. Typically, this means that for every write operation that occurs on the primary data set, an identical write operation is simultaneously performed on the mirror.
- **Key Objectives**:
  - **High Availability**: Ensures continuous availability of data, even if the primary data source fails, by instantly switching to the mirrored data.
  - **Data Protection**: Protects against data loss due to hardware or site failures; it does not protect against corruption or accidental deletion, which are copied to the mirror, so backups are still needed.
  - **Performance Enhancement**: Can improve read performance by allowing read operations to be split across the primary and mirror sites.

Objectives of Mirroring and Replication in Data Management (continued)

Use Cases:

- **Critical Applications**: Ideal for critical applications where data availability and uptime are paramount.
- **Disaster Recovery**: Acts as a part of a disaster recovery plan by providing an up-to-date copy of data that can be used immediately.

Objectives of Mirroring and Replication in Data Management (continued)

Replication

Objective: Data Distribution and Consistency

- **Description**: Replication involves copying data from one location to another, but not necessarily in real-time. This can be done synchronously or asynchronously, and the data can be replicated to multiple locations.
- **Key Objectives**:
  - **Data Distribution**: Facilitates the distribution of data across different geographical locations or systems.
  - **Load Balancing**: Helps in load balancing by distributing queries among multiple copies of the data.
  - **Backup and Archiving**: Useful for creating backups and archives of data at regular intervals; a replica also copies errors and deletions, so it does not replace point-in-time backups.
  - **Data Consistency**: Ensures data consistency across multiple databases or systems, although there might be a time lag in asynchronous replication.

Use Cases:

- **Data Sharing**: Effective for scenarios where data needs to be shared across different departments or locations.
- **Reporting and Analysis**: Used for creating replicas for reporting and analysis purposes, without impacting the performance of the primary database.
- **Offline Processing**: Ideal for scenarios where data can be processed offline or where immediate consistency is not required.

Measures of System Availability in IT Management

System availability is a critical metric in IT management, indicating the degree to which an IT system is operational and accessible when required for use. It's typically expressed as a percentage and is a key component of service level agreements (SLAs). Understanding various measures of system availability, such as agreed service time and downtime, is essential for IT professionals.

Agreed Service Time:

- **Description**: This refers to the time period during which a system or service is expected to be operational and available to users. It is often specified in an SLA.
- **Example**: An SLA may specify that a system should be available from 8:00 AM to 6:00 PM on weekdays, totaling 50 hours per week. This is the agreed service time.

Downtime:

Planned Downtime:

- **Description**: This is the time period during which the system is intentionally taken offline for scheduled maintenance, updates, or other planned activities.
- **Impact on Availability**: Planned downtime is typically excluded from availability calculations, as it is a known and accepted part of system management.

Unplanned Downtime:

- **Description**: This refers to unexpected disruptions in service availability due to issues like system failures, software bugs, or external factors (e.g., power outages).
- **Impact on Availability**: Unplanned downtime is critical in availability calculations and negatively impacts the overall availability percentage.

Measures of System Availability in IT Management (continued)

Availability Percentage:

- **Calculation Method**: Availability is calculated using the formula:
  Availability (%) = [(Agreed Service Time - Unplanned Downtime) / Agreed Service Time] × 100
- **Example Calculation**: If the service level agreement (SLA) calls for 24/7 service, the agreed service time in a week is 168 hours (24 hours × 7 days), and there are 2 hours of unplanned downtime, the availability is calculated as follows:
  Availability (%) = [(168 hours - 2 hours) / 168 hours] × 100 = 98.81%

Mean Time Between Failures (MTBF) and Mean Time To Repair (MTTR):

- **MTBF**: A measure of the time elapsed between system failures. Higher MTBF indicates greater reliability.
- **MTTR**: The average time taken to repair a system and restore it to full functionality.
- Lower MTTR contributes to higher system availability.

Full Backup: Involves copying all data to the backup storage. This is the most comprehensive backup type.

- **Considerations**:
  - **Resource Intensive**: Requires more storage space and longer backup time.
  - **Recovery Advantage**: Simplifies the recovery process as only one backup set is needed.

Incremental Backup: Backs up only the data that has changed since the last backup (either full or incremental).

- **Considerations**:
  - **Efficiency**: Less storage space and quicker backup process.
  - **Recovery Complexity**: Requires the last full backup and all subsequent incremental backups for a complete recovery.

Evaluating the Appropriateness of Data Backup Types in an Organization (continued)

Differential Backup: Backs up data that has changed since the last full backup.

- **Considerations**:
  - **Storage and Time Balance**: More efficient than full backups but less so than incremental backups.
  - **Recovery Process**: Easier than incremental backups as it only requires the last full backup and the last differential backup.

Split Mirror Backup: Used when a firm has a large amount of data. One side of a mirrored disk set is split off (detached) to give a point-in-time copy, which is backed up, often to a remote or offline location, while the primary system stays online; the mirror is then resynchronized.

The **accounting system data backup** of year end information should be stored in some kind of secure off site location.

Evaluating the Appropriateness of Data Backup Types in an Organization (continued)

- Business impact analysis: identify critical business processes and the systems and resources they need, assess the impact of an outage over time, and set recovery priorities. RTO (recovery time objective): how quickly a process or system must be restored. RPO (recovery point objective): the maximum acceptable data loss, measured in time. The disaster recovery plan restores IT; the business continuity plan keeps the business running.

- Choose backup type and frequency from the RPO and RTO, keep copies protected and off site (or offline), and test restores regularly.

- Availability can also be expressed as MTBF / (MTBF + MTTR) (mean time between failures; mean time to repair).

In software development, different environments and types of tests are used to ensure that applications are developed, tested, and deployed efficiently and effectively. Each environment serves a specific purpose in the development lifecycle, and various tests are performed to validate different aspects of the software.

Different Environments:

Development Environment:

- **Purpose**: This is where developers write and initially test their code. It's a personal or shared space for developers to make changes without affecting other environments.
- **Characteristics**: Typically includes development tools and a local version of the application for individual use.

Understanding Different Environments and Types of Tests in Software Development (continued)

Staging Environment:

- **Purpose**: Serves as a pre-production environment, replicating the production environment as closely as possible. It's used for final testing and for identifying any bugs that might not have been caught in the development environment.
- **Characteristics**: Mirrors the production environment in terms of hardware, software, and configuration. This ensures that any tests conducted here yield results that closely match what will occur in production.

Production Environment:

- **Purpose**: This is the live environment where the application is available to end-users. It’s the final stage where the software runs after thorough testing in previous environments.
- **Characteristics**: Must be stable and secure, as it directly impacts the user experience and business operations.
- Control point: keep development, test and production separate; developers get no write access to production; changes reach production only through approved change requests, ideally migrated by someone other than the developer. Mask sensitive test data.

Understanding Different Environments and Types of Tests in Software Development (continued)

Types of Tests:

Unit Testing:

- **Purpose**: Tests individual components or units of the software to ensure that each part functions correctly in isolation.
- **How It's Done**: Developers typically write unit tests, and they are often automated.

Integration Testing:

- **Purpose**: Tests the interaction between integrated units/modules to detect interface defects.
- **How It's Done**: Conducted after unit testing, it ensures that different parts of the application work together as expected.

Understanding Different Environments and Types of Tests in Software Development (continued)

System Testing:

- **Purpose**: Tests the complete and integrated software to verify that it meets the specified requirements.
- **How It's Done**: Usually done by a team independent of the development team, it covers functional and non-functional testing aspects.

Acceptance Testing:

- **Purpose**: Validates if the software meets business needs and is ready for deployment.
- **Types**:
  - **User Acceptance Testing (UAT)**: Done by the end-users to ensure that software can handle required tasks in real-world scenarios.
  - **Beta Testing**: Conducted by a limited number of end-users to evaluate real-world usability and identify any remaining issues.
  - Note: Regression testing is not an acceptance test type; it re-runs existing tests at any level after a change or patch to confirm that previously working functions still work.

Approaches for Converting to a New Information System

1. Direct Conversion (Big Bang): In a direct conversion, the organization switches off the old system and starts using the new system all at once, across the entire organization.

- **Advantages**: Quick implementation and less costly in the short-term as it does not require running two systems simultaneously.
- **Risks**: High risk of disruption if the new system fails, as there is no fallback and can lead to significant operational challenges if issues are encountered.
- **Suitability**: Best for smaller or less complex systems where the risk of failure is low.

2. Parallel Conversion: This approach involves running both the old and new systems simultaneously for a certain period.

- **Advantages**: Reduces risk as the old system serves as a backup and allows for comparison and validation of data between the two systems.
- **Risks**: Resource-intensive, as it requires maintaining and operating two systems and can be confusing for staff to manage two sets of data and workflows.
- **Suitability**: Ideal for larger organizations or where the new system poses significant operational risks.

Approaches for Converting to a New Information System (continued)

3. Pilot Conversion: The new system is first deployed to a small group or a particular department for testing before rolling out to the entire organization.

- **Advantages**: Allows for testing and troubleshooting on a smaller scale before full implementation and reduces overall risk as results at the pilot site can be evaluated and adjusted.
- **Risks**: Can be time-consuming, as the full implementation is done in stages and initial pilot group may face challenges that need addressing before wider implementation.
- **Suitability**: Suitable for organizations that prefer a cautious approach and where the system's complexity or scale warrants proving the system at one site before a full rollout.

4. Phased Conversion: Similar to pilot conversion but involves implementing the new system in stages based on functionality rather than organizational units.

- **Advantages**: Allows the organization to gradually adapt to the new system and reduces the risk of total system failure at any one time.
- **Risks**: Longer overall implementation period and may face temporary integration challenges between old and new system components during the transition.
- **Suitability**: Useful for very large and complex systems or when different parts of the organization have varying readiness levels for the change.

Understanding Patch Management in IT

Patch management is a critical process in the field of information technology that involves the acquisition, testing, and installation of patches (updates) to software applications and systems. These patches are often released to address vulnerabilities, fix bugs, or add new features. Effective patch management is essential for maintaining the security and functionality of IT systems.

- Patch management steps: prioritize (by vulnerability severity and asset criticality), schedule, validate (the patch is authentic and unaltered), test (in a non-production environment, with a rollback plan), deploy, and monitor (confirm installation and that the vulnerability is closed).

- Change management tools: ticketing, version control, test libraries, build automation, monitoring and logging. Documentation: component inventory, baseline configuration, change requests, tickets and rollback plans. CI/CD (continuous integration and delivery) pipelines build in change control: automated tests, required approval before merge, and logged deployments. Configuration parameters: restrict who can change settings, compare them to the approved baseline, and log and review changes.

Data normalization

Normalization refers to the process of organizing data in a database to reduce redundancy and improve data integrity. This process involves structuring a database in a way that minimizes duplication of information across tables, ensures data dependencies are logical, and makes the database easier to maintain, although more tables can mean more joins when querying.

This concept is crucial in relational database design and is achieved through applying a series of rules, or "normal forms," each addressing a specific type of redundancy or anomaly. Starting from the first normal form (1NF), which eliminates repeating groups, to more advanced forms like the third normal form (3NF), which removes transitive dependencies, normalization helps in designing databases that are free from update anomalies, are consistent, and provide a high degree of data integrity.

Normalization typically involves dividing large tables into smaller, related tables and defining relationships between them. The aim is to isolate data so that additions, deletions, and modifications of a field can be made in just one table and other tables reference that single copy through primary key and foreign key relationships.

Data storage

Data storage and database schemas are crucial concepts in the realm of data management and analytics. Each type of data storage serves a different purpose, and the choice of database schema plays a significant role in how data is organized and accessed.

Types of Data Storage

Data Warehouse

- **Definition**: A data warehouse is a centralized repository that stores data from multiple sources. It's structured and often used for reporting and data analysis.
- **Characteristics**: Data in a warehouse is processed, cleaned, and structured. It's ideal for complex queries and data analysis.
- **Usage**: Primarily used by businesses for decision support systems, it enables the analysis of large volumes of historical data.

Data storage (continued)

Data Lake

- **Definition**: A data lake is a storage repository that holds a vast amount of raw data in its native format until it's needed.
- **Characteristics**: It stores unstructured, semi-structured, and structured data. Data lakes are more flexible than warehouses and can store all types of data.
- **Usage**: Useful for big data and machine learning projects where data in its raw form is required for diverse analytical processes.

Data Mart

- **Definition**: A data mart is usually a subset of a data warehouse. It is oriented to a specific business line or team.
- **Characteristics**: Since data marts hold only relevant data, they are smaller, more focused, and quicker to query than full data warehouses.
- **Usage**: Often used by specific departments within an organization (like sales, finance, or marketing) to focus on specific datasets relevant to their area.

Database Schemas

Star Schema

- **Definition**: In a star schema, a central fact table is surrounded by dimension tables.
- **Characteristics**:
  - **Fact Table**: Contains quantitative data for analysis.
  - **Dimension Tables**: Store descriptive attributes (dimensions) related to fact table data.
  - **Simplicity**: Simple design, easy to understand and navigate.
- **Usage**: Widely used in data warehouses because of its simplicity and effectiveness in handling simple queries.

Data storage (continued)

Snowflake Schema

- **Definition**: An extension of the star schema where dimension tables are normalized.
- **Characteristics**:
  - **Normalized Dimension Tables**: Sub-divided into additional tables. This reduces data redundancy.
  - **Complexity**: More complex than star schema, but can lead to efficiencies in some cases.
- **Usage**: Beneficial where the data model is complex and data redundancy needs to be minimized. However, it can be more difficult to navigate and understand.

Data storage (continued)

Comparison and Use Cases

- **Data Warehouse vs Data Lake**: A data warehouse is structured and used for specific analytical and reporting purposes, while a data lake is more about storing vast amounts of raw data for diverse use cases, including machine learning.
- **Star vs Snowflake Schema**: The star schema is simpler and generally faster for querying but can involve some data redundancy. The snowflake schema is more normalized (less redundancy) but can be more complex and slower in query performance.

Data life cycle

1. Data Creation: This is the initial phase where data is created or captured. This can happen in various ways, such as data entry, data acquisition from external sources, or automatic generation from sensors and machines. Ensuring the quality and accuracy of data at this stage is crucial as it sets the foundation for all subsequent stages.

2. Data Processing and Maintenance: Once data is created, it needs to be processed to be useful. This involves data validation, transformation, and integration. It is an ongoing process to ensure data remains accurate, complete, and relevant. It includes data cleaning, enrichment, and updating.

3. Data Storage: Data is then stored in a suitable storage solution which could be databases, data warehouses, or cloud storage systems. The choice of storage depends on the data's size, structure, and frequency of access. Security measures and data backup procedures are also crucial in this stage.

4. Data Usage: This stage involves the active use of data for various purposes such as analysis, decision-making, reporting, or operational support. Ensuring easy access, efficient processing, and accurate interpretation of data is vital. Data security is also a priority to protect sensitive information.

Data life cycle (continued)

5. Data Sharing and **Publication:** Involves the distribution of data within or across organizations. Publication means making data available to external parties, such as in open data initiatives. Requires compliance with data sharing policies and privacy laws, and maintaining data integrity during transfer.

6. Data Archiving: Inactive or less frequently accessed data is moved to an archive for long-term preservation. Data archiving helps in optimizing storage and improving the performance of active systems. It’s crucial to ensure data is archived in a way that it remains accessible and secure.

7. Data Disposition: The final stage involves the safe and secure disposal or deletion of data that is no longer needed. Compliance with legal and regulatory requirements is critical, along with ensuring that data is irrecoverably destroyed to prevent unauthorized access.

Follow a documented retention schedule, and suspend disposal of any data under a legal hold.

Relational databases

A relational database is a type of database that stores and provides access to data points that are related to one another.

Basic Explanation of Relational Databases

Analogy: Think of a relational database like a filing system in an office. In this system, information is stored in different folders (tables). Each folder contains many files (records), and each file has different sections (fields or columns) for various details. Just like how you can link information from different folders in an office, a relational database can link data from different tables.

Tables: A relational database is made up of tables. Each table is like a grid made up of rows and columns.

- **Rows**: Each row (also called a record) represents a single item, person, transaction, etc.
- **Columns**: Each column represents a specific attribute or type of information, like a name, date, amount, etc.

Relations: The term "relational" comes from the relational model, in which each table is a "relation". Tables are linked to each other through keys (like a customer ID).

Data Integrity: These databases maintain accuracy and consistency in data. For example, a customer's address is stored once in the Customers table, so changing it there changes it everywhere it is referenced.

Relational databases (continued)

Let's say you work in an accounting department and use a relational database to manage financial information.

- **Tables**: You might have several tables, such as:
  - Transactions table, which records each financial transaction (like sales or purchases).
  - Customers table, containing customer details.
  - Products table, listing all products or services with their prices.
- **Using the Database**: When a sale is made, you enter the transaction details in the Transactions table. This table has a link (like a customer ID) to the Customers table, so you know who made the purchase. It also links to the Products table to indicate what was bought.
- **Benefits**: This system makes it easy to:
  - Retrieve all transactions made by a particular customer.
  - Update a product's current price in one place (in the Products table) so that every new transaction uses it. Each recorded transaction keeps the price actually charged, so past sales do not change.
  - Generate financial reports, like total sales for a period, or a list of outstanding payments from customers.
- **Data Integrity**: If a customer's address changes, you only need to update it in the Customers table, and all associated transactions will automatically reference the updated address.

Relational databases (continued)

Examining a relational database's structure to evaluate its adherence to data integrity rules, use of a data dictionary, and implementation of data normalization involves a thorough understanding of several key components and principles.

1. Data Integrity Rules

Data integrity rules are crucial in maintaining the accuracy and reliability of data in a relational database. They help ensure that the data entered into the database is valid and consistent.

- **Entity Integrity**: Check if every table has a primary key that uniquely identifies each record. No two rows should have the same primary key value, and it must not be null.
- **Referential Integrity**: Look for foreign keys that link tables together. Ensure that foreign key values either match a primary key value in another table or are null. This maintains the consistency of relationships between tables.
- **Domain Integrity**: Examine if data fields in each table are restricted to specific ranges, types, or formats. This is often enforced through constraints, data types, or default values.
- **User-Defined Integrity**: Look for any additional rules specific to the business logic that the database is designed to serve.

2. Use of a Data Dictionary

A data dictionary is a centralized repository of information about data such as meaning, relationships to other data, origin, usage, and format.

- **Metadata Storage**: Identify if the database has a system catalog or a metadata (data that describes other data) repository that describes the structure of the database, including tables, columns, data types, and relationships.
- **Documentation**: A well-maintained data dictionary should also include information about constraints, indexes, stored procedures, and other relevant metadata.
- **Accessibility and Updates**: Check how the data dictionary is managed, whether it's automatically updated when database schema (blueprint of the database) changes, and how it's made accessible to users and database administrators.

3. Data Normalization

Data normalization is the process of organizing data to reduce redundancy and improve data integrity. It involves dividing large tables into smaller, interrelated tables and defining relationships between them.

- **Normalization Levels**: Determine the level of normalization (1NF, 2NF, 3NF, BCNF, etc.) the database adheres to. For instance, in the First Normal Form (1NF), tables should have primary keys and no repeating groups or arrays.
- **Table Design**: Look at the database tables to see if similar types of data are stored in separate tables and if relationships between these tables are properly defined.
- **Redundancy Check**: Check for data redundancy. If the same data is duplicated in multiple places, it's a sign that the database might not be fully normalized.
- 2NF: in 1NF, and every non-key field depends on the whole primary key (no partial dependency on part of a composite key). 3NF: in 2NF, and no non-key field depends on another non-key field (no transitive dependency).

SQL

What is SQL?

Definition: SQL stands for Structured Query Language. It's a standard programming language specifically designed to interact with databases.

Purpose: SQL is used to communicate with a database. It allows you to perform tasks such as retrieving specific data from large datasets, updating data, deleting data, and managing database structures.

Basics of Databases

Database: Think of a database like a digital filing cabinet where data is stored in an organized manner. In a database, this data is often stored in tables, which you can visualize as spreadsheets with rows and columns. Each row represents a record (like a single transaction or entry), and each column represents a different attribute of that record (like date, amount, description).

SQL (continued)

Relational Databases: Many databases are relational, which means they use relationships to connect different tables. This is particularly useful for organizing complex sets of data.

SQL in Action

- **Queries**: SQL works by executing "queries" – commands to the database to perform specific tasks. For example, a simple query might ask the database to give you all records of transactions that occurred in the past month.
- **Data Manipulation and Retrieval**: You can use SQL to add new data, update existing data, or delete unwanted data. But most importantly, you can retrieve data based on specific criteria – this is where SQL is extremely powerful for analysis.

SQL (continued)

SQL in the Accounting Profession

- **Financial Data Management**: Accountants deal with vast amounts of financial data. SQL helps in organizing, retrieving, and managing this data efficiently.
- **Reporting and Analysis**: Accountants often need to generate financial reports, analyze trends, or audit financial records. SQL can quickly pull specific data needed for these reports from large databases, saving time and reducing errors.
- **Budgeting and Forecasting**: SQL can assist in budgeting and financial forecasting by retrieving historical financial data, which can be used to predict future trends and prepare budgets.
- **Compliance and Auditing**: SQL can help in ensuring compliance with financial regulations by easily retrieving and organizing data required for audits or regulatory reviews.
- **Integration with Accounting Software**: Many accounting software applications use databases that can be queried using SQL. This allows for more customized data analysis and reporting than what might be available through standard software interfaces.

SQL (continued)

Example Use-Case

Imagine you're an accountant and need to prepare a report on all transactions above $10,000 for a specific client. Instead of sifting through thousands of entries manually, you can write an SQL query that directly retrieves this information from the database. This query will specify the conditions (transactions above $10,000) and the target (a specific client), and the database will return the relevant data almost instantly.

SQL in plain terms

For someone unfamiliar with SQL, think of it as a powerful tool that speaks the language of databases. In the accounting profession, it's like having a super-efficient assistant who can instantly find, organize, and calculate data from massive digital records, simplifying and enhancing the data management process. It's an essential skill in modern accounting, offering precision, efficiency, and a high degree of control over financial data.

SQL (continued)

Common Commands

SELECT: Used to select data from a database. The data returned is stored in a result table, sometimes called the result set.

- **Example**: SELECT name, age FROM employees;
- **Relevance**: Specifies the exact fields to be returned, ensuring only relevant data is included.
- **Completeness**: Excluding necessary fields can lead to incomplete data.

WHERE: Applies a condition to filter records.

- **Example**: WHERE age > 30;
- **Relevance**: Filters out irrelevant records.
- **Completeness**: Overly restrictive conditions can exclude relevant data.

FROM: Specifies the table from which to retrieve data.

- **Example**: FROM employees;
- **Relevance & Completeness**: Choosing the right table is crucial for both.

SQL (continued)

Common Commands

GROUP BY: Groups rows that have the same values in specified columns.

- **Example**: GROUP BY department;
- **Relevance**: Essential for aggregate calculations per group.
- **Completeness**: Misuse can lead to misleading aggregations.

HAVING: Applies a condition to groups.

- **Example**: HAVING COUNT(employee_id) > 5;
- **Relevance**: Filters groups based on a condition.
- **Completeness**: Can exclude relevant groups if not used properly.
- WHERE filters rows before grouping; HAVING filters groups after aggregation, so a condition on SUM or COUNT belongs in HAVING.

ORDER BY: Sorts the result set.

- **Example**: ORDER BY age DESC;
- **Relevance**: Sorting doesn't affect relevance.
- **Completeness**: Sorting doesn't affect completeness but affects presentation.

SQL (continued)

Clauses, Operators, Aggregate Functions, and String Functions

JOIN Clauses: Combines rows from two or more tables.

- **Example**: INNER JOIN department ON employees.department_id = department.id;
- **Relevance**: Essential for combining related data from different tables.
- **Completeness**: Missing or incorrect joins can lead to incomplete data.

Operators (e.g., AND, OR, NOT): Used to filter records.

- **Example**: WHERE age > 30 AND department = 'HR';
- **Relevance**: Fine-tunes the data retrieval.
- **Completeness**: Incorrect use can exclude relevant data.

Aggregate Functions (e.g., COUNT, SUM, AVG): Perform a calculation on a set of values and return a single value.

- **Example**: SELECT AVG(salary) FROM employees;
- **Relevance**: Useful for summarizing data.
- **Completeness**: Need to be used considering the grouping of data for meaningful results.

String Functions (e.g., CONCAT, SUBSTRING, UPPER): Perform operations on string data.

- **Example**: SELECT UPPER(name) FROM employees;
- **Relevance**: Transform data for specific needs.
- **Completeness**: Mainly affects the format, not the completeness of the data.

SQL (continued)

- Join types and completeness: INNER JOIN keeps only rows that match in both tables, so unmatched rows drop out. LEFT JOIN keeps every row of the left table (NULLs where no match); RIGHT JOIN does the same for the right table; FULL OUTER JOIN keeps all rows from both.
- NULLs: COUNT(*) counts all rows, but COUNT(column), SUM and AVG ignore NULLs. A WHERE condition such as amount > 0 also excludes rows where amount is NULL.

Ensuring Relevance and Completeness

- **Relevance**: Achieved by correctly using WHERE, SELECT, and JOINs to filter and retrieve the right data.
- **Completeness**: Ensured by not over-restricting data with WHERE and HAVING clauses, and by correctly joining tables to include all necessary data.

SQL (continued)

Example Scenario: Retrieve the total sales of each department from a database that has two tables: employees and sales.

```
SELECT department, SUM(sales_amount) as TotalSales
FROM employees
INNER JOIN sales ON employees.id = sales.employee_id
GROUP BY department
ORDER BY TotalSales DESC;
```

Analysis for Relevance and Completeness:

- **Relevance**: The query is relevant because it fetches sales data per department, which is the intended information.
- **Completeness**: It is not guaranteed. The INNER JOIN drops any sale whose employee_id has no matching employee (for example, a terminated employee who was deleted or a NULL employee_id), so total sales can be understated, and departments with no sales do not appear at all. Use a LEFT JOIN from sales to employees, or reconcile the query total to the sales ledger total, to confirm completeness; LEFT JOIN from employees to sales (and COALESCE the SUM to 0) to show zero-sale departments.

Threat agents

Internal vs. External Threat Agents

- **Internal Threat Agents**: These are individuals within an organization who may use their authorized access, intentionally or unintentionally, to harm the organization. This could be due to various reasons such as dissatisfaction, financial gain, or coercion. For example, a disgruntled employee might intentionally leak sensitive data to harm the company.
- **External Threat Agents**: External agents are those who are not part of the organization and typically include hackers, cybercriminal groups, or competitors. An example is a hacker group launching a cyber-attack to steal confidential data or disrupt services.

Nation-State vs. Non-Nation State-Sponsored Agents

- **Nation-State Sponsored**: These agents are supported or directed by a national government and are often involved in cyber espionage or cyber warfare. Their objectives can range from stealing sensitive government data to disrupting critical infrastructure. For instance, interference from one country into another country's affairs.
- **Non-Nation State-Sponsored**: These include independent groups or individuals such as hackers, hacktivists, and cybercriminals. They might be driven by financial gain, ideological beliefs, or personal challenges. For example, hacktivist groups like Anonymous often launch attacks for political or social causes.

Threat agents (continued)

Adversaries, Threat Actors, Attackers, and Hackers

- **Adversaries**: This is a broad term that encompasses any entity that poses a threat to the security of an organization. It includes anyone from a malicious insider to a rival corporation.
- **Threat Actors**: These are individuals or groups that carry out security threats. They can range from amateur hackers to highly sophisticated cybercriminal syndicates.
- **Attackers/Hackers**: Typically, these terms refer to individuals who actively exploit vulnerabilities in systems for various purposes. While ‘hackers’ can have a non-malicious connotation (as in 'white-hat hackers'), they are often perceived as entities seeking unauthorized access to systems.

Examples and Impact

- An internal agent could be an employee who inadvertently becomes a threat by falling prey to a phishing attack, compromising the company's network.
- An external threat actor could be a cybercriminal group orchestrating ransomware attacks that encrypt an organization's files and demand payment.
- A nation-state threat agent might engage in espionage to steal intellectual property or government secrets, similar to the activities attributed to China's PLA Unit 61398.
- Non-nation state actors like Anonymous have been known to carry out attacks for social or political reasons, such as attacking financial institutions for perceived injustices.

Types of attacks

Physical Attacks: Physical attacks target the tangible components of a network, such as servers, workstations, and network devices. This can include actions like theft, vandalism, or sabotage. **Example**: An intruder breaking into a data center and damaging servers, causing downtime and data loss.

Distributed Denial of Service (DDoS): DDoS attacks aim to overwhelm a system's resources, making it unavailable to its intended users. This is achieved by flooding the target with excessive requests from multiple distributed sources, often compromised devices (a botnet). **Example**: A high-profile website being inundated with traffic from thousands of botnet-infected devices, rendering it inaccessible to legitimate users.

Malware: Malware, or malicious software, includes various types of harmful software such as viruses, worms, trojan horses, and ransomware. These are designed to damage, disrupt, or gain unauthorized access to computer systems. **Example**: Ransomware encrypting an organization's files and demanding payment for their decryption.

Types of attacks (continued)

Social Engineering: Social engineering attacks manipulate individuals into divulging confidential information or performing actions that compromise security. These attacks often rely on psychological manipulation. **Example**: Phishing emails that trick employees into revealing their login credentials.

Web Application Attacks: These attacks target vulnerabilities in web applications to steal data, disrupt services, or take control of systems. Common methods include SQL injection, cross-site scripting, and cross-site request forgery. **Example**: An attacker using SQL injection to gain unauthorized access to a database through a website's input field.

Mobile Device Attacks: These attacks specifically target mobile devices like smartphones and tablets. They exploit vulnerabilities in mobile operating systems, apps, or communication networks. **Example**: A malicious app on a smartphone accessing and transmitting confidential corporate data.

Types of attacks (continued)

Buffer Overflow: A buffer overflow occurs when a program writes more data to a buffer than it can hold. This can overwrite adjacent memory, leading to erratic program behavior, crashes, and vulnerabilities. **Example**: An attacker might input excessively long strings of data to overflow a buffer, allowing them to inject malicious code into the system.

Mobile Code: Mobile code refers to programs that can be transmitted across a network and executed on a local system. While useful, they can also be exploited to perform malicious actions. **Example**: JavaScript, document macros, or older Java applets and ActiveX controls that are automatically downloaded and executed, potentially carrying harmful scripts.

Cross-Site Scripting (XSS): XSS attacks involve injecting malicious scripts into webpages viewed by other users. These scripts can hijack user sessions, deface websites, or redirect the user to malicious sites. **Example**: An attacker embeds a script in a comment on a website. When other users view the comment, the script executes, compromising their session.

SQL Injection: SQL injection involves inserting malicious SQL queries into input fields, exploiting vulnerabilities to manipulate databases. **Example**: An attacker inputs a SQL statement into a web form input box, giving them access to the database.

Key mitigations: input validation and bounds checking (buffer overflow); output encoding (XSS); parameterized queries with input validation (SQL injection).

Types of attacks (continued)

Race Conditions: A race condition occurs when a system attempts to perform two or more operations at the same time, but the sequence of events affects the outcome. **Example**: An attacker exploits a race condition to execute a privilege-escalation attack, where timing differences in processes allow unauthorized access.

Covert Channel: A covert channel is a method used to transfer information in a way that violates the security policy of the system, often exploiting a legitimate communication channel in unintended ways. **Example**: Data could be leaked through seemingly innocuous means, such as altering the timing of legitimate network traffic.

Replay Attack: In a replay attack, a valid data transmission is maliciously or fraudulently repeated or delayed. **Example**: An attacker intercepts and records authentication information, such as a password, and then retransmits it to gain access to a system.

Return-Oriented Attack: This attack exploits existing code within a program to execute a payload, rather than injecting new code. **Example**: An attacker uses a buffer overflow to manipulate the stack and execute a malicious payload using the program's own code.

Stages of a cyber attack

Reconnaissance: This is the preparatory phase where attackers gather information about the target. This can include identifying vulnerable systems, understanding network structures, or social engineering tactics.

- **Example**: An attacker might scan a company's website and social media profiles to gather information about network systems, employee details, and company operations.

Gaining Access: In this stage, attackers use the information gathered during reconnaissance to exploit vulnerabilities and gain unauthorized access. Methods can include hacking, phishing, or exploiting software vulnerabilities.

- **Example**: Using a phishing email to trick an employee into revealing their login credentials, which the attacker then uses to access the corporate network.

Escalation of Privileges: Once inside the system, attackers often seek to gain higher-level access. This involves escalating their user privileges to gain broader access to the system or network.

- **Example**: An attacker exploits a vulnerability to transform from a standard user to an administrator, gaining access to restricted areas of the network.

Stages of a cyber attack (continued)

Maintaining Access: Attackers aim to create a persistent presence within the network to extract data continuously or to launch further attacks. This can be achieved by installing backdoors or other malicious software.

- **Example**: Installing a rootkit or trojan horse to remain connected to the network without being detected.

Network Exploitation: This stage involves taking advantage of the access gained. Attackers can steal data, disrupt services, or use the network as a base for launching attacks on other targets.

- **Example**: Harvesting confidential data, deploying ransomware, or using the network to launch attacks on partnered organizations.

Covering Tracks: To avoid detection and ensure continued access, attackers often take steps to hide their activities. This includes erasing log files, disguising malicious activities, and using encryption.

- **Example**: Deleting log entries that show unauthorized access attempts or using encryption to hide data exfiltration activities.

Internet of Things (IoT)

The Internet of Things (IoT) refers to the interconnected network of physical devices, vehicles, home appliances, and other items embedded with electronics, software, sensors, and network connectivity, which enables these objects to collect and exchange data. The goal of IoT is to create devices that can communicate with each other and users in real-time without human intervention. Common IoT examples (1 to 7), IoT risks (8) and mobile and cloud risks (9):

1. **Smart home devices**: Thermostats, lighting systems, security cameras, door locks, and appliances that can be controlled remotely through a smartphone app or voice assistant.
2. **Wearable technology**: Smartwatches, fitness trackers, and health monitoring devices that collect data on physical activity, heart rate, and sleep patterns.
3. **Connected vehicles**: Cars with built-in sensors and internet connectivity that can provide real-time traffic updates, remote diagnostics, and autonomous driving capabilities.
4. **Smart cities**: IoT sensors and devices used in urban infrastructure to optimize energy consumption, traffic management, waste management, and public safety.
5. **Industrial IoT**: Connected sensors, equipment, and machinery used in manufacturing, agriculture, and energy sectors to improve efficiency, productivity, and predictive maintenance.
6. **Smart retail**: IoT devices used in stores to track inventory, analyze customer behavior, and enable automated checkout systems.
7. **Medical devices**: Connected devices such as insulin pumps, heart monitors, and asthma inhalers that can track patient data and provide real-time alerts to healthcare providers.

Internet of Things (IoT) (continued)

8 IoT cybersecurity risks: default or weak passwords, firmware that is hard or impossible to patch, weak or no encryption, a large and poorly inventoried attack surface, devices recruited into botnets for distributed denial-of-service (DDoS) attacks, vendor remote access, and physical-safety and privacy harm.

9 Mobile device risks: lost or stolen devices, malicious apps, insecure public Wi-Fi, smishing (text-message phishing) and unmanaged bring-your-own-device (BYOD) devices. Cloud risks: misconfiguration, weak identity and access management, insecure application programming interfaces (APIs) and unclear shared-responsibility boundaries.

Threat modeling and threat landscape

Introduction to Threat Modeling and Threat Landscape: In cybersecurity, being proactive is key. This means not only responding to threats as they occur but anticipating and preparing for them in advance. This is where threat modeling and understanding the threat landscape become essential.

Threat Modeling - A Proactive Approach to Security: Threat modeling is the process of identifying, assessing, and addressing potential threats to a system. It involves a systematic analysis of what controls or defenses are needed to prevent or mitigate the impact of these threats.

- **Identifying Potential Threats**: The first step is identifying what could potentially harm a system. This includes looking at potential attackers, their capabilities, and methods.
- **Determining and Prioritizing Potential Targets**: Analyzing what parts of a system are most likely to be targeted by attackers. This could be data, processes, or physical assets.
- **Assessing Vulnerabilities**: Determining where a system is most vulnerable.

  This includes software weaknesses, system misconfigurations, and human factors.
- **Mitigation Strategies**: Developing strategies to reduce risk. This includes implementing security controls, policies, and procedures.

Threat modeling and threat landscape (continued)

Threat Landscape - Understanding the External Environment: While threat modeling is an internal process, understanding the threat landscape involves looking externally at the broader environment. It's about understanding the current and emerging threats that exist in the world of cybersecurity.

- **Current Threats**: This includes the most common types of attacks happening today, such as ransomware, phishing, and DDoS attacks.
- **Emerging Threats**: Staying informed about new and evolving threats. This could include new strains of malware, novel cybercriminal tactics, or vulnerabilities in emerging technologies like IoT.
- **Industry-Specific Threats**: Different industries may face different types of threats. For instance, the financial sector is often targeted for financial gain, while government entities may face state-sponsored attacks.
- **Geopolitical Factors**: Understanding how geopolitical developments can impact cybersecurity. For example, tensions between nations can lead to an increase in state-sponsored cyber attacks.
- **Integrating Threat Modeling and Threat Landscape**
  - **Comprehensive Security Strategy**: Combining threat modeling with an understanding of the threat landscape provides a comprehensive approach to cybersecurity.
  - **Continuous Process**: Both threat modeling and understanding the threat landscape should be ongoing processes, adapting to changes in technology, business practices, and the external environment.

Strategies for Protecting Networks and Remote Access Devices

1. Isolation and Segmentation: To limit access within the network and contain potential breaches.

- **Methods**:
  - **Network Segmentation**: Dividing the network into smaller segments or subnetworks, each with controlled access.
  - **Isolation of Sensitive Data**: Ensuring critical data or systems are isolated from general network access.

Strategies for Protecting Networks and Remote Access Devices (continued)

2. Virtual Private Network (VPN): To provide secure remote access to the network.

- **Methods**:
  - **Encrypted Connections**: Using VPNs to encrypt data transmitted between remote devices and the network, thus protecting data integrity and confidentiality.
  - **Secure Authentication**: Implementing strong authentication methods, such as multi-factor authentication (MFA), for VPN access.

3. Wireless Network Security: To safeguard wireless connections against unauthorized access and eavesdropping.

- **Methods**:
  - **Strong Encryption**: Utilizing WPA3 or the latest secure encryption standards for Wi-Fi.
  - **Network Access Control**: Implementing access controls to restrict who can connect to the wireless network.

Strategies for Protecting Networks and Remote Access Devices (continued)

4. Endpoint Security: To protect individual devices that connect to the network remotely.

- **Methods**:
  - **Anti-malware Software**: Ensuring all remote devices have up-to-date anti-malware protection.
  - **Regular Updates**: Keeping all software and operating systems on the devices up to date to protect against known vulnerabilities.

5. System Hardening: To reduce the attack surface of systems and devices.

- **Methods**:
  - **Removing Unnecessary Services**: Disabling unused services and ports on devices and servers.
  - **Configuring Security Settings**: Applying the principle of least privilege and configuring security settings to enhance protection.

6. Intrusion Prevention and Detection Systems (IPS/IDS): To detect and prevent malicious activity within the network.

- **Methods**:
  - **Intrusion Detection Systems**: Monitoring network and system activities for malicious actions or policy violations.
  - **Intrusion Prevention Systems**: Actively blocking or preventing identified threats.

Understanding Layered Security and Defense-in-Depth

Layered security and defense-in-depth are foundational concepts in cybersecurity, emphasizing the need for multiple layers of defense to protect against a wide range of threats. Although they are often used interchangeably, each has distinct aspects.

Layered Security: Layered security, also known as "layered defense," refers to the practice of using multiple security measures to protect the various components of an organization's IT environment. The idea is that if one layer fails, others will continue to provide the necessary defense.

- **Key Aspects**:
  - **Diverse Protection Strategies**: Incorporating different types of security measures (like firewalls, antivirus software, intrusion detection systems) at different points in the system.
  - **Comprehensive Coverage**: Ensuring that all aspects of the IT environment (including hardware, software, network, and data) are protected.

Defense-in-Depth: Defense-in-depth is a strategy that implements multiple layers of defense (across the physical, technical, and administrative domains) to create a comprehensive, holistic security posture.

- **Key Aspects**:
  - **Holistic Approach**: Goes beyond technical solutions to include policies, procedures, and physical controls.
  - **Depth of Defense**: Focuses on creating depth by implementing security measures at various levels, from the perimeter to the core of the IT infrastructure.

Understanding Layered Security and Defense-in-Depth (continued)

Comparison and Interaction:

- **Layered Security in Practice**:
  - Implementing antivirus software on end-user devices.
  - Using firewalls to guard network perimeters.
  - Employing encryption for sensitive data.
- **Defense-in-Depth in Practice**:
  - Physical security controls (like access controls to buildings).
  - Administrative controls (such as user training and security policies).
  - Technical controls (like network segmentation and security monitoring).

Complementary Nature:

- **Layered Security as a Component of Defense-in-Depth**:
  - Layered security can be seen as a component of the broader defense-in-depth strategy. While layered security focuses more on technical measures, defense-in-depth encompasses a wider range of controls, including non-technical measures.

Key Concepts in Cybersecurity: Least Privilege, Zero Trust, Whitelisting, and Need-to-Know Principle

1. Least Privilege: The principle of least privilege involves granting users only the access and permissions necessary to perform their job functions. It restricts the access rights for users to the bare minimum necessary to complete their tasks.

Purpose: To minimize the potential damage from accidents, errors, or unauthorized use of resources. Reduces the attack surface by limiting the number of users with access to sensitive information or critical systems.

2. Zero **Trust:** Zero Trust is a security concept centered on the belief that organizations should not automatically trust anything inside or outside their perimeters and instead must verify everything trying to connect to their systems before granting access.

Purpose: To protect modern digital environments by enforcing strict user authentication and authorization. Aims to prevent data breaches and attacks by assuming that no user or device is trusted by default, regardless of their location or the network they are connected to.

3. Whitelisting **(also called allowlisting):** Whitelisting is a cybersecurity strategy that allows actions, applications, or IP addresses that are explicitly permitted and denies all others. It's the opposite of blacklisting (denylisting), which blocks specific entities and allows all others.

Purpose: To provide stringent control over which applications or processes are allowed to function on a system or network. Helps in preventing unauthorized or potentially harmful software from executing.

4. Need-to-Know Principle: The need-to-know principle states that sensitive information should be disclosed only to individuals who have a necessity for that information in order to perform their duties.

Purpose: To limit access to sensitive information to only those who require it to perform their job functions. Aims to reduce the risk of accidental or intentional data leaks or breaches.

Key Concepts in Cybersecurity: Least Privilege, Zero Trust, Whitelisting, and Need-to-Know Principle (continued)

- Zero trust grants access one resource and one session at a time with least privilege, checks both the user's identity and the device's security, and keeps re-checking while access continues.
- Vulnerability management: a continuous cycle to identify (asset inventory and scanning), prioritize (severity and exposure), remediate (patch, reconfigure or apply a compensating control), verify (rescan) and report vulnerabilities.
- Authentication: passwords (long, not reused, screened against breached lists), personal identification numbers (PINs), smart cards, biometrics (false acceptance vs false rejection), digital signatures, single sign-on, and multi-factor authentication (two or more different types: something you know, have or are). Authorization models: discretionary (the owner grants access), mandatory (the system enforces labels and clearances), role-based (access by job role); an access control list (ACL) states which users may do what to a resource.
- Acceptable use policy: rules for using the entity's systems and data. Mobile and bring-your-own-device (BYOD) policy: device enrollment (mobile device management), screen lock and encryption, remote wipe, app restrictions, and separation of work and personal data.

Fundamentals of encryption

Encryption is a crucial concept in the field of information security, used to protect data from unauthorized access.

Fundamentals of Encryption

Definition: Encryption is the process of converting data or information into a code, especially to prevent unauthorized access. It involves the use of algorithms to transform plaintext into ciphertext.

Key Concepts:

- **Plaintext**: The original, readable data.
- **Ciphertext**: The transformed, unreadable data.
- **Encryption Algorithm**: A method or formula used to convert plaintext into ciphertext.
- **Key**: A piece of information used by the algorithm to transform the data. In symmetric encryption, a single secret key is used for both encryption and decryption. In asymmetric encryption, a public key (shared openly) encrypts and a separate private key (kept secret) decrypts; for digital signatures the roles reverse (the private key signs, the public key verifies).

Purpose: The primary goal of encryption is to maintain data confidentiality. It ensures that only authorized parties can access the original data.

Techniques of encryption

Techniques of Encryption

Symmetric Encryption:

- Involves a single key for both encryption and decryption.
- **Examples**: AES (Advanced Encryption Standard), DES (Data Encryption Standard, now obsolete and insecure; replaced by AES).

Asymmetric Encryption:

- **Uses a pair of keys**: for confidentiality, the recipient's public key (known to everyone) encrypts and the recipient's private key (kept secret) decrypts; for digital signatures, the sender's private key signs and the sender's public key verifies.
- **Examples**: RSA (Rivest–Shamir–Adleman), ECC (Elliptic Curve Cryptography).

Hash Functions:

- Not encryption in the traditional sense but often discussed alongside encryption.
- Transforms data into a fixed-size hash or digest. This process is irreversible.
- Hash examples: SHA-256 and SHA-3 (MD5 and SHA-1 are no longer secure). Salt: a unique random value added to each password before hashing, so identical passwords give different hashes and precomputed (rainbow table) attacks fail. PKI (public key infrastructure): a certificate authority issues digital certificates that bind a public key to a verified identity; certificates can expire or be revoked.

Applications of encryption

Applications of Encryption

- **Data Security**: Protects data stored on computers and transmitted via the internet or other networks.
- **Secure Communications**: Used in applications like messaging services (e.g., WhatsApp, Signal) to ensure privacy.
- **E-Commerce**: Encryption is vital for securing online transactions.
- **Password Storage:** Hashing (a one-way function, distinct from encryption) is used for securely storing passwords. Because hashing is one-way, the password cannot be decrypted from the hash, but an attacker who steals the hashes can still guess weak passwords and compare the results, so passwords should be salted and hashed with a slow, purpose-built algorithm.
- **Digital Signatures**: Asymmetric cryptography is used for creating digital signatures: the sender signs with the sender's private key and anyone can verify with the sender's public key. Signatures authenticate the identity of the sender, ensure the integrity of the message and support non-repudiation, but they do not encrypt (hide) the message.
- **Cryptocurrencies**: Use cryptographic techniques to secure transactions and control the creation of new units.

Confidentiality vs privacy

Key Differences

- **Scope**: Confidentiality and privacy overlap, but neither is a subset of the other: confidentiality protects any information designated confidential (personal or not, such as trade secrets or pricing), while privacy applies only to personal information and covers its collection, use, retention, disclosure and disposal. Privacy includes the right to personal autonomy and control over personal information, whereas confidentiality is specifically about the unauthorized disclosure of information.

- **Legal and Ethical Considerations**: Privacy is often protected by laws and regulations (like GDPR or HIPAA in healthcare), which give individuals rights over their personal data. Confidentiality can be governed by legal agreements, ethical obligations, or specific organizational policies.

- **Application**: Confidentiality is typically more focused on organizational and professional contexts, such as client-attorney or doctor-patient relationships, whereas privacy is an individual's right applicable in a wider range of contexts.

Data protection

1. Data **Obfuscation:** Data obfuscation is the process of disguising original data to prevent unauthorized access while still allowing the data to be useful for development and testing.

- **Design & Development**: Use obfuscated data in development environments to protect sensitive information while allowing developers to work with realistic data sets.
- **Testing**: Implement obfuscated data in test environments to prevent exposure of sensitive data to unauthorized personnel or third parties.

2. Tokenization: Tokenization is the process of substituting sensitive data elements with non-sensitive equivalents, known as tokens, which have no extrinsic or exploitable value.

- **Development & Testing**: Use tokenization to ensure that actual sensitive data is not present in development and test environments, reducing the risk of data breaches.
- **Implementation**: Tokenize sensitive data before it enters the application, and de-tokenize only when necessary for processing.

3. Encryption: Encryption involves converting data into a coded format that can only be read with a key.

- **All Stages**: Encrypt sensitive data at rest and in transit across all stages of the application lifecycle.
- **Implementation**: Implement strong encryption standards like AES for data storage and TLS for data transmission.

Data protection (continued)

4. Access **Controls:** Access control involves restricting access to data based on user roles and permissions.

- **Design & Development**: Define strict user roles and permissions in the application design to control who can access sensitive data.
- **Implementation**: Implement role-based access controls to ensure that only authorized personnel can access sensitive data in the production environment.

5. Data **Masking:** Data masking involves hiding original data with modified content (characters or other data).

- **Development & Testing**: Use data masking in development and testing environments to protect sensitive data while maintaining functional integrity.

6. Use of Environment-Specific Configurations

- **Design & Development**: Ensure that development and testing environments do not use actual confidential data or have direct access to production databases.
- **Implementation**: Use environment-specific configurations to manage data differently across development, testing, and production environments.

Data protection (continued)

7. Regular Audits and Compliance Checks

- **All Stages**: Conduct regular audits and compliance checks to ensure that data protection measures are consistently applied and effective throughout the application lifecycle.

8. Pseudonymization: Pseudonymization is a data management and de-identification procedure by which identifying fields within a data record are replaced with artificial identifiers or pseudonyms. The data can be linked back to a person only with additional information that is kept separately and protected, so pseudonymized data is still personal data under GDPR (unlike anonymized data).

- **Development & Testing**: Use pseudonymized data to reduce the risk of breaching privacy laws and regulations.

9. Data Loss Prevention (DLP): tools and rules that find sensitive data and monitor or block its unauthorized movement (in use, in transit and at rest), for example stopping an email that contains customer credit card numbers.

Data breach implications: financial (forensics, notification, legal fines and penalties, lawsuits, lost revenue) and operational (downtime, recovery effort, and lost customer trust and reputation).

Understanding the Distinctions Between Security/Cybersecurity Events and Incidents

Security/Cybersecurity Events: A security event is any observable occurrence in a system or network. Events include a broad spectrum of activities, not all of which are indicative of a problem.

Examples: Successful logins, failed login attempts, firewall denials, system alerts, or any other recorded activity.

Characteristics: Many events occur in a system or network environment daily. Most events are routine and do not suggest a security issue.

Monitoring and Analysis: Security events are typically logged and monitored as part of normal network and system operations. Analysis of these events helps in identifying trends or unusual patterns that could indicate potential security issues.

Security/Cybersecurity Incidents: A cybersecurity incident is a specific event or a series of events that threaten the confidentiality, integrity, or availability of information system resources.

Examples: A successful malware attack, unauthorized access to sensitive data, a DDoS attack disrupting services, or a breach of secure information.

Understanding the Distinctions Between Security/Cybersecurity Events and Incidents (continued)

Characteristics: Incidents imply a violation of an organization's security policies, acceptable use policies, or standard security practices. Incidents often necessitate immediate investigation and response to mitigate risks and damages.

Response and Management: Incidents trigger a specific response protocol, which may include escalation procedures, forensic analysis, and incident reporting. Incident response teams are usually mobilized to handle the situation, conduct investigations, and implement recovery plans.

Incident response plan contents: roles and responsibilities (response team, management, legal, communications), contacts and escalation, methods and steps (prepare, detect and analyze, contain, eradicate, recover, post-incident review), timelines (including legal notification deadlines), and lessons learned.

Cyber insurance transfers part of the financial loss from an incident, subject to the policy's terms; it does not replace controls or remove operational and reputational harm.

To test whether the plan was followed, compare incident records (tickets, logs, approvals, timestamps, notices) with the plan's required steps and deadlines, and document any deviations.

Final recap of important topics

1. Process Control Information

- **Definition and Importance**: Process control involves the systematic monitoring and adjustment of processes to meet specific goals. It is part of internal control over financial reporting (ICFR) and relies on IT general controls (ITGCs).
- **Key Components**:
  - **Input Controls**: Ensure data integrity during data entry, e.g., validation checks.
  - **Processing Controls**: Monitor accuracy during processing, such as reconciliation procedures.
  - **Output Controls**: Validate output accuracy, like ensuring reports reconcile with underlying data.
- **Relevance to ISC Exam**:
  - Candidates may be tested on how process controls impact financial reporting and audit procedures.
  - Focus on identifying control weaknesses, understanding their financial impact, and recommending improvements.
- **Example Question**:
  - Evaluate a scenario where inadequate input controls lead to errors in financial reports. Identify compensating controls or recommend changes.

Final recap of important topics (continued)

2. SOC (System and Organization Controls) Reporting

- **Definition**:
  - SOC reports are assurance reports prepared in accordance with AICPA standards on an organization's controls (Type 1: design as of a date; Type 2: design and operating effectiveness over a period).
- **Types of SOC Reports**:
  - **SOC 1**: Focuses on controls relevant to financial reporting (e.g., payroll services).
  - **SOC 2**: Evaluates controls over security (always included) and any of availability, processing integrity, confidentiality, and privacy that are in scope.
  - **SOC 3**: Publicly available (general-use) report on the same Trust Services Criteria as a SOC 2 Type 2, without the detailed system description and the tests and results.
- **Components of SOC Reporting**:
  - SOC 2 common criteria CC1 to CC5 follow the COSO 2013 components: CC1 Control Environment; CC2 Communication and Information; CC3 Risk Assessment; CC4 Monitoring Activities; CC5 Control Activities.
  - Trust Services Criteria (TSC) for SOC 2.

Final recap of important topics (continued)

- **Relevance to ISC Exam**:
  - Candidates may need to differentiate between SOC reports and identify which is appropriate in a given scenario.
  - May involve analyzing report findings (control exceptions and deficiencies) and their effect on the service auditor's report.
- **Example Question**:
  - Interpret a SOC 1 report provided by a third-party payroll processor and assess its impact on the audit of the client’s financial statements.

Final recap of important topics (continued)

3. SQL Basics

- **Definition**:
  - Structured Query Language (SQL) is used to manage and query data in relational databases.
- **Common SQL Commands**:
  - **SELECT**: Retrieve data from a table.
  - **INSERT**: Add new data to a table.
  - **UPDATE**: Modify existing data.
  - **DELETE**: Remove data.
  - **JOIN (a clause used within SELECT):** Combine data from multiple tables based on a related column.
- **Relevance to ISC Exam**:
  - Candidates may encounter questions on using SQL to extract or manipulate data for financial reporting or audit purposes.
  - Understanding of database structures and their use in automated control testing.
- **Example Question**:
  - Examine an SQL query that retrieves transactions over $10,000 from a specific table and determine whether its results are relevant and complete (for example, a wrong comparison operator or an INNER JOIN that drops unmatched records).

Final recap of important topics (continued)

4. Relational Databases

- **Definition**:
  - Relational databases store data in tables that relate to each other through primary and foreign keys.
- **Key Concepts**:
  - **Tables**: Organized rows (records) and columns (fields).
  - **Primary Key**: A unique identifier for each record in a table.
  - **Foreign Key**: A field in one table that links to the primary key in another.
  - **Normalization**: Organizing data to minimize redundancy.
- **Relevance to ISC Exam**:
  - Candidates may need to understand how relational databases support data integrity and facilitate financial reporting.
  - Could involve analyzing database structures to identify control gaps.
- **Example Question**:
  - Evaluate a database schema to determine whether its design supports accurate financial reporting.

Final recap of important topics (continued)

5. Change Management

- **Definition**:
  - Change management refers to the processes ensuring that IT system changes (e.g., software updates, access control modifications) are authorized, tested, and documented.
- **Key Elements**:
  - **Change Request Process**: Who initiates and approves changes.
  - **Testing**: Validating changes in a non-production environment.
  - **Documentation**: Recording details of changes for audit trails.
  - **Monitoring**: Ensuring changes are implemented correctly and are not abused.
  - Also: separation of duties (developers do not move their own changes into production) and emergency changes that are approved and documented after the fact.
- **Relevance to ISC Exam**:
  - Frequently tested in the context of ITGC and how weak change management can affect system reliability and data integrity.
  - May involve analyzing scenarios where unauthorized changes lead to control failures.
- **Example Question**:
  - Identify weaknesses in a change management process and their impact on financial reporting.

Final recap of important topics (continued)

6. Tests of Controls

- **Definition**:
  - Tests of controls are audit procedures designed to evaluate the operating effectiveness of controls in achieving their objectives (in a SOC 2 Type 2, the service commitments and system requirements based on the applicable Trust Services Criteria). Deviations are reported as exceptions, and inquiry alone is not enough evidence that a control operated effectively.
- **Types of Tests**:
  - **Inquiry**: Asking personnel about control processes.
  - **Observation**: Watching a process in action.
  - **Inspection**: Reviewing documents for evidence of control operation.
  - **Reperformance**: Re-executing a control activity.
- **Relevance to ISC Exam**:
  - Questions may involve determining the appropriate test for a given control, interpreting results, or identifying weaknesses.
  - Often linked with substantive testing in audit scenarios.
- **Example Question**:
  - Describe the audit implications of a failed test of controls over payroll processing.

Final recap of important topics (continued)

7. Trust Services Criteria (TSC)

- **Definition**:
  - AICPA’s framework for evaluating an organization's controls related to security, availability, processing integrity, confidentiality, and privacy.
- **Key Categories:**
  - **Security**: Protecting systems and data.
  - **Availability**: Ensuring systems are operational.
  - **Processing Integrity**: System processing is complete, valid, accurate, timely, and authorized.
  - **Confidentiality**: Protecting sensitive information.
  - **Privacy**: Personal information is collected, used, retained, disclosed, and disposed of to meet the entity's objectives.
  - TSC organization: the common criteria (CC1 to CC9) apply in every SOC 2 and make up the security category; CC1 to CC5 follow COSO 2013's 17 principles, and CC6 to CC9 are supplemental criteria (access, system operations, change management, risk mitigation). Availability, processing integrity, confidentiality and privacy add their own criteria. Points of focus are guidance, not a required checklist.

Final recap of important topics (continued)

- **Relevance to ISC Exam**:
  - Often tested in the context of SOC 2 engagements or ITGC. Candidates may need to assess whether a control satisfies a specific TSC.
  - Could involve analyzing real-world scenarios, such as securing sensitive financial data.
- **Example Question**:
  - Evaluate whether a company’s data encryption policy aligns with the Trust Services Criteria for confidentiality.

But Your Journey's Just Beginning!

Congratulations on completing the memorization guide! This is a major milestone, but remember:

1. This guide covers the **minimum** you should have memorized before exam day.
2. Mastery comes through repetition. Don't just read it once and call it done!

Tips for Memorization Mastery:

- Alternate between this guide ⇒ flashcards ⇒ guide ⇒ flashcards for optimal retention.
- Our flashcards are built for repeated review to build long-term memory.
- Aim to memorize both the guide **and** flashcards for a comfortable pass.

Remember:

- Every formula, table, and concept in this guide represents potential "free points" on the exam, being the easiest questions you'll get. Aim to score 100% on the easy memorization-based questions.
- Consistent practice is key. Dedicate time each day to rewrite formulas and review concepts.

"Many before you have done it, and you can too. Use every tool at your disposal to achieve success!"

• Nick Palazzolo, CPA Instructor

Congratulations on finishing this memorization guide!

Just a reminder that this is the version of our memorization guide for CPA candidates who aren't yet actively studying on our platform. Our free account includes 100+ flashcards and 200+ practice questions, and is the perfect next step to solidify your memory of this material!

What makes Copernicus different?

- ✔ **Affordable:** Each course costs less than a single exam retake
- ✔ **ReadyRating Readiness Predictions:** a 5-level rating tells you when to sit or reschedule, to help you avoid retake fees
- ✔ **6,700+ Questions:** a deep practice bank across all six exams, including AICPA materials
- ✔ **200+ Task-Based Simulations:** realistic practice for the simulation testlets you will see on every section of the exam
- ✔ **Pass Guarantee:** Refund if you fail after reaching ReadyRating Level 5

Copernicus CPA Review is also super efficient. Our platform maximizes your readiness by actively discovering what you know, targeting gaps, reinforcing memory, and giving you a clear picture of what you need to do to pass. Our free account is free forever.

We hope you'll join us on your CPA journey! **Create your free account**

