CompTIA CySA+ CS0-003

Full Learning Guide

Welcome to your complete CySA+ CS0-003 learning guide.
*This manual is designed to **teach you every domain in depth**, not just summarize.*

Learning Objectives and Expectations

You'll master:

- Every CySA+ CS0-003 exam objective.
- All critical concepts a security analyst must know.
- How detection, response, vulnerability management, and reporting **fit together**.
- How to think like a **cybersecurity analyst**, not just memorize facts.

Each domain guide includes:

- Full topic breakdowns with real-world relevance.
- Tools, techniques, and frameworks used in actual SOCs.
- Practical examples, command-line insights, and analyst workflows.
- Memory aids, exam tips, and scenario-based thinking.

CySA+ CS0-003 Domains at a Glance

Each domain is weighted differently on the exam, with **Security Operations** being the largest.

- Domain 1: **Security Operations** (33%)
- Domain 2: **Vulnerability Management** (30%)
- Domain 3: **Incident Response and Management** (20%)
- Domain 4: **Reporting and Communication** (17%)

Quick Reminder: How the Exam Works

- **Number of Questions**: Up to 85
- **Format**: Multiple Choice + Performance-Based Questions (PBQs)
- **Time Limit**: 165 minutes
- **Passing Score**: 750 / 900 (approx. 83%)
- **Test Provider**: Pearson VUE (onsite or online)
- **Recommended Experience**: Network+, Security+, 3–4 years in cybersecurity or security operations

Top 10 CySA+ Exam Tips

1. **Review Core Tools Before the Exam**: Know what SIEM, EDR, Nessus, Wireshark, and SOAR tools do—even if only conceptually.
2. **Practice with PBQs**: Use labs or mock PBQs to practice interpreting logs, prioritizing vulnerabilities, and classifying incidents.
3. **Skip PBQs Strategically**: If a PBQ is taking too long, skip it and finish your multiple choice first—then return.
4. **Use Elimination for Tricky Scenarios**: Narrow answers based on logic, even if you're unsure—especially in detection and IR questions.
5. **Highlight Keywords**: Pay close attention to words like **FIRST**, **BEST**, **NOT**, **MOST LIKELY**.
6. **Expect Real-World Language**: You'll see log snippets, analyst workflows, and attacker scenarios. Think like you're on a SOC shift.
7. **Master IR and Kill Chain**: Know how to walk through PICERL, Cyber Kill Chain, and MITRE ATT&CK examples under pressure.
8. **Don't Panic If You Don't Know a Term**: Use reasoning. Many questions can be solved through context and understanding basic principles.
9. **Flag and Move If Unsure**: You can come back later—protect your time and mental energy.
10. **Stay Calm and Confident**: You've prepared to think like an analyst—don't second-guess your instincts during the exam.

Remember - You Don't Need Perfection to Pass!

To pass CySA+, you need about **83%**, meaning you can **miss ~14–15 questions** and still succeed.
The exam is scenario-driven—but manageable if you've studied context, tools, and analyst workflows.

Trust your preparation. Think like a security analyst. Keep moving forward.

Domain 1: Security Operations (33%)

Goal of Domain 1:

You must understand how to monitor, detect, and respond to security incidents using foundational knowledge in system architecture, logging, SIEM, threat intelligence, threat hunting, and detection tools. You must also master operational processes and learn how to optimize and integrate technologies in the Security Operations Center (SOC).

This domain is where you become a **cybersecurity analyst** — eyes on the glass, hunting threats, identifying attacks, analyzing indicators, and improving detection across endpoints, networks, and systems.

1.1 System and Network Architecture for Security Monitoring

Learn: What must be monitored and why?

Operating System (OS) Concepts for Security Monitoring

- **Processes and services**: Running programs that could be malicious.
- **Memory and disk usage**: Unusual spikes may indicate malware.
- **System logs**: Track login attempts, app crashes, config changes.
- **OS logging locations**:
  - **Windows**: Event Viewer (e.g., Security, Application, System logs).
  - **Linux**: /var/log/ directory, especially auth.log, syslog.

Network Infrastructure Basics

- Learn how data moves:
  - **Router**: Directs packets.
  - **Switch**: Connects endpoints in LANs.
  - **Firewall**: Controls inbound/outbound traffic.
  - **IDS/IPS**: Intrusion Detection/Prevention systems inspect traffic.
- Understand **normal** vs **anomalous** network flow.
- Know **DMZ** (demilitarized zone): Public-facing network segment, segregated for security.

Critical Systems to Monitor

- **Domain Controllers** (especially if using Active Directory)
- **Web servers**
- **Database servers**
- **VPN endpoints**
- **Cloud services (e.g., AWS, Azure logs)**

Key Concept to Learn: Monitoring is effective only if you understand baseline system behavior. Without a baseline, anomalies are invisible.

1.2 Log Ingestion and Log Management

What Are Logs?

- **Logs = records** of actions/events (auth attempts, file access, network connections).
- Sources:
  - **Firewalls**
  - **Servers**
  - **Applications**
  - **Endpoints**
  - **Cloud services (CloudTrail, Azure logs)**

Log Ingestion

- Aggregating logs from multiple sources into a central system (typically a **SIEM**).
- Often involves agents (e.g., Splunk Forwarders, Beats, Syslog).

Log Normalization

- Converting logs from different formats into a **standard structure** so they can be searched and analyzed easily.

Retention and Compliance

- Define how long logs are stored.
- Often guided by:
  - **Legal regulations (e.g., HIPAA, PCI DSS)**
  - **Organizational policies**

Concept to Learn: Logs provide the evidence trail in any investigation. If not collected properly, threats go undetected.

1.3 SIEM and Security Monitoring Tools

What is SIEM?

- **Security Information and Event Management**
- Example tools:
  - Splunk
  - IBM QRadar
  - LogRhythm
  - ELK (Elasticsearch, Logstash, Kibana)

SIEM Capabilities

- **Real-time alerting** on suspicious patterns.
- **Log search** and correlation across systems.
- **Dashboards** for SOC monitoring.
- **Use Cases** for:
  - Brute force attempts
  - Data exfiltration
  - Lateral movement

Log Analysis Queries

- Use **regex, queries**, and **filters** to find patterns.
- Example: Search for Event ID 4625 (Windows failed logon).

"Single Pane of Glass"

- Integration of multiple tools into **one dashboard** for complete situational awareness.

Concept to Learn: A well-tuned SIEM can detect threats in real-time and prevent costly breaches — only if log sources and use cases are set up properly.

1.4 Threat Intelligence

What is Threat Intelligence (TI)?

- TI is contextual information about threats — helps **anticipate, detect, and respond**.

Types of TI

1. **Strategic**

    - High-level, big-picture (e.g., geopolitical threats).
2. **Operational**
    - Campaigns, attack patterns (e.g., ransomware group activities).
3. **Tactical**
    - Techniques, tools, procedures (e.g., PowerShell abuse, phishing tactics).
4. **Technical**
    - Specific IOCs (IP addresses, file hashes, domains).

Sources of TI

- **Open-source**: VirusTotal, AbuseIPDB, AlienVault OTX
- **Commercial**: FireEye iSIGHT, Recorded Future
- **ISACs**: Sector-specific sharing groups (e.g., FS-ISAC for finance)

Threat Intelligence Platforms (TIPs)

- Used to **aggregate, enrich, and share** TI.
- Integrate with SIEM/SOAR.

Concept to Learn: TI turns reactive defense into proactive defense. Know how to use IOCs and correlate them with internal events.

1.5 Threat Hunting

What is Threat Hunting?

- **Proactive** search for threats that evaded detection tools.
- You form a **hypothesis** and investigate systems to confirm or deny it.

Types of Threat Hunting

- **Intel-based**: Using known IOCs (e.g., "Check if we've seen IP x.x.x.x").
- **TTP-based**: Looking for attacker behaviors (mapped to **MITRE ATT&CK**).
- **Anomaly-based**: Looking for deviations from normal behavior.

Tools for Hunting

- SIEM queries
- EDR logs
- Network traffic capture (e.g., Zeek/Bro, Wireshark)
- Endpoint process analysis

Outcome

- Finding IOCs missed by other tools.

- Creating **new detection rules or signatures**.

Concept to Learn: Threat hunting relies on creativity and deep understanding of environment baselines. It's not automatic — it's analytical investigation.

1.6 Recognizing Indicators of Malicious Activity

Indicator of Compromise (IOC)

- Clues that a system has been breached.
- Examples:
  - Known malicious IP
  - Unusual PowerShell command
  - Registry key changes
  - Base64-encoded payloads in logs

Types of Indicators

- **Network-based**: Suspicious traffic, beaconing, C2 communications.
- **Host-based**: Abnormal processes, modified files, unauthorized user accounts.
- **Application-based**: SQL injection attempts, repeated 500 errors.
- **Other**: Odd behavior, off-hours logins, geographic anomalies.

Analysis of Logs and Events

- Use tools like Splunk to correlate multiple events.
- Example:
  - Event ID 4625 (failed logon)
  - Followed by Event ID 4624 (success)
  - Then Event ID 4670 (file permissions changed)

Concept to Learn: Malware hides. You must connect the dots across logs, behaviors, and anomalies to uncover it.

1.7 Security Toolsets for Detection

Endpoint Detection and Response (EDR)

- Monitors endpoint activity.
- Captures telemetry (processes, file changes, network connections).
- Can **kill malicious processes** or isolate host.

Example Tools: CrowdStrike, SentinelOne, Carbon Black

Network-Based Detection

- IDS/IPS tools: Snort, Suricata.
- Detect malicious packets and signatures.

Packet Capture Tools

- **Wireshark** for packet analysis.
- **Zeek (Bro)** for protocol-level monitoring.

Sandboxing

- Isolate and run suspicious files to observe behavior.

Threat Lookup Tools

- VirusTotal, Hybrid Analysis, Any.run (sandbox), Shodan, GreyNoise.

Concept to Learn: Know when to use which tool — EDR for host behavior, IDS for traffic, sandbox for unknown files.

1.8 MITRE ATT&CK and Kill Chain Frameworks

MITRE ATT&CK

- Matrix of **tactics** (goals) and **techniques** (methods).
- Helps map attacker activity.
- Use for:
    - Gap analysis
    - Building detection rules
    - Threat hunting hypotheses

Cyber Kill Chain (Lockheed Martin)

1. Reconnaissance
2. Weaponization
3. Delivery
4. Exploitation
5. Installation
6. Command and Control
7. Actions on Objectives

Concept to Learn: Know the sequence of attacker actions. Detection earlier in the chain = better outcome.

1.9 Threat Actor Profiles and TTPs

Threat Actor Categories

- Nation-states
- Hacktivists
- Organized crime
- Insider threats

Tactics, Techniques, and Procedures (TTPs)

- Understand common attacker techniques:
  - Credential dumping
  - Lateral movement
  - Privilege escalation
  - Data exfiltration
- Use ATT&CK matrix for real-world mapping.

Concept to Learn: Know your enemy. Each threat actor group has preferred TTPs — analysts must be familiar with them.

1.10 Efficiency and Process Improvement

Standardization

- Use of playbooks, SOPs.
- Consistent response reduces errors.

Automation

- Use SOAR platforms (e.g., Palo Alto Cortex XSOAR) to:
  - Enrich alerts
  - Auto-quarantine
  - Notify teams

Integration

- SIEM + EDR + threat intelligence = faster decision making.

Single Pane of Glass

- Central console showing:
  - Alerts
  - Asset inventory
  - Remediation actions

Continuous Improvement

- Review:
  - Alert fidelity
  - Mean time to detect/respond (MTTD/MTTR)
- Adjust detection logic to reduce noise.

Concept to Learn: The best SOCs evolve. You must constantly tune, integrate, and optimize.

Summary of Domain 1: Security Operations

Master these:

- System & OS architecture for detection
- Log ingestion, normalization, and retention
- SIEM functionality and log analysis
- Threat intelligence types, feeds, and platforms
- Threat hunting process and hypotheses
- Indicators of compromise – network, host, app
- Detection toolsets – EDR, IDS/IPS, sandboxing, Wireshark
- MITRE ATT&CK and Cyber Kill Chain frameworks
- Threat actor types and TTPs
- SOC process improvements, SOAR, single-pane integration

Domain 2: Vulnerability Management (30%)

Goal of Domain 2:

You must learn how to discover, assess, prioritize, and remediate system and application vulnerabilities. This domain teaches you how to use scanning tools, interpret findings, calculate risk, and implement mitigation strategies while working within business and technical constraints.

This is where you become the **eyes that find weaknesses before attackers do** — scanning, analyzing, and driving security improvements through proactive vulnerability management.

2.1 Vulnerability Scanning Concepts and Methodologies

Learn: What is Vulnerability Management?

- Ongoing process to **identify, classify, prioritize, and mitigate vulnerabilities** in systems, networks, and applications.

Phases of Vulnerability Management:

1. **Discovery** – Inventory and identify assets.
2. **Scanning** – Search for vulnerabilities.
3. **Analysis** – Interpret scan results.
4. **Prioritization** – Rank based on risk.
5. **Remediation** – Apply patches, mitigations.
6. **Verification** – Rescan to ensure fix.
7. **Reporting** – Communicate progress.

Concept to Learn: Vulnerability management is not a one-time project — it is a continuous lifecycle.

Scan Types:

1. Internal Scanning

- Run **from inside** the network.
- Sees internal exposures an insider or malware might exploit.

2. External Scanning

- Run **from outside** the network.
- Identifies vulnerabilities in **public-facing systems** (e.g., websites, VPNs).

3. Credentialed Scanning

- Uses **valid system credentials** (e.g., admin account).
- Provides **deep insights**: missing patches, insecure configs.

4. Non-Credentialed Scanning

- Scans from the **outside-in** without credentials.
- Shows what an **external attacker** sees.

5. Active vs. Passive Scanning

- **Active**: Directly probes systems (Nmap, Nessus).
- **Passive**: Monitors network traffic to detect vulnerabilities.

6. Agent-Based Scanning

- Software installed on endpoints to report directly to scanner.
- Useful for **remote users** and mobile systems.

Concept to Learn: Know which scan method fits each scenario — credentialed = more accuracy, external = perimeter view.

2.2 Interpreting Vulnerability Scan Results

Vulnerability Scanner Tools to Know

- **Nessus** (Tenable)
- **OpenVAS**
- **Qualys**
- **Rapid7 InsightVM**
- **Microsoft Defender Vulnerability Management**

Key Report Elements

- **CVE ID** (Common Vulnerabilities and Exposures): Unique identifier (e.g., CVE-2023-12345)

- **CVSS Score**: 0.0–10.0 risk rating (Critical/High/Medium/Low)
- **Affected System**: Hostname or IP
- **Description**: Summary of vulnerability
- **Proof/Detection method**: Evidence from scan
- **Fix**: Patch or mitigation advice

Concept to Learn: Reports must be **analyzed**, not just read. False positives and prioritization matter more than volume.

2.3 Prioritizing Vulnerabilities and Assessing Risk

Learn: Not all vulnerabilities are equal.

Factors that Affect Prioritization:

- **CVSS score** (Severity)
- **Asset criticality**
- **Exposure** (Internet-facing vs internal)
- **Exploitability** (Known exploits, Metasploit modules)
- **Active exploitation** (from TI feeds)
- **Compliance implications**

Risk = Likelihood × Impact

- **Likelihood**: How easily can this be exploited?
- **Impact**: What's the damage if it is?

Example:

- SQL Injection on public website storing customer data = **High Risk**
- Outdated media player on HR intern's laptop = **Low Risk**

Concept to Learn: Risk-based prioritization ensures high-value fixes are done first. Use context, not just CVSS.

2.4 Validating and Confirming Vulnerabilities

False Positives

- Scanner flags a vuln that isn't truly exploitable or doesn't apply.
- Example: Nessus flags SSL issue, but that service is disabled.

Validation Methods

- **Manual check**: Log in and verify version or config.
- **Exploit attempt**: (in a test environment) use tools like Metasploit to confirm.
- **Corroborate with logs or EDR**

False Negatives

- A real vulnerability is **missed** by the scanner.
- Happens due to:
  - Misconfiguration
  - Limited scan permissions
  - Complex attack paths

Concept to Learn: Never assume scan output is perfect — confirm critical findings and scan with the right context.

2.5 Remediating and Mitigating Vulnerabilities

Remediation

- **Fixing the root problem** (e.g., applying a patch, upgrading software).

Mitigation

- **Reducing risk** without fully eliminating the vulnerability.
- Example: Apply WAF rules to block SQLi instead of fixing code immediately.

Workarounds

- Temporary fixes until full remediation is possible.
- Example: Disable a service rather than patch it immediately.

Compensating Controls

- Used when primary remediation isn't feasible.
- Example: Isolate unpatchable system behind strict firewall and add monitoring.

Patch Management Process

1. Scan and detect vulnerabilities
2. Test patches in staging
3. Deploy to production during **maintenance window**
4. Verify and document results

Concept to Learn: A patch is ideal, but real-world business constraints often require mitigation or compensation.

2.6 Types of Vulnerabilities

Categories of Vulnerabilities to Recognize

1. Software Bugs

- Examples:
  - Buffer overflow
  - Use-after-free
  - Race condition

2. Misconfigurations

- Default passwords
- Open S3 buckets
- Directory listing enabled
- Missing security headers

3. Weak Authentication

- No MFA
- Password reuse
- Shared admin accounts

4. Cryptographic Failures

- Weak encryption (e.g., MD5, DES)
- No HTTPS
- Missing certificate validation

5. Web App Vulns

- **SQL Injection**
- **XSS**
- **CSRF**
- **LFI/RFI**

6. End-of-Life Software

- No longer supported (e.g., Windows 7, PHP 5.x)
- No security patches available

Concept to Learn: Web applications, legacy systems, and misconfigurations are common entry points for attackers.

2.7 Vulnerability Databases and Scoring

CVEs and NVD

- **CVE**: A unique ID for a vulnerability (managed by MITRE).
- **NVD** (National Vulnerability Database): U.S. gov database, includes CVSS scores.

CVSS (Common Vulnerability Scoring System)

- Standardized method to rate vulnerability severity.
- Ranges from 0.0 to 10.0
- Scores are based on:
  - Attack vector
  - Complexity
  - Authentication required
  - Impact (Confidentiality/Integrity/Availability)

Severity Ratings:

- 0.0 = None
- 0.1–3.9 = Low
- 4.0–6.9 = Medium
- 7.0–8.9 = High
- 9.0–10.0 = Critical

Concept to Learn: CVE is the "name," CVSS is the "severity." Learn how to interpret both.

2.8 Special Scanning Considerations

Sensitive Assets

- SCADA/OT systems
- Medical equipment
- Legacy systems

Avoid active scanning on these — use:

- Passive analysis
- Manufacturer-approved tools
- Scheduled scans during maintenance windows

Asset Inventory

- Before scanning, know **what you own**:
  - IP ranges
  - Hostnames
  - Operating systems
  - Installed applications

Use discovery tools (Nmap, Netdisco) to map assets.

Scan Permissions and Timing

- Ensure credentials are up-to-date.
- Schedule scans during off-peak hours.
- Notify stakeholders to avoid panic from scan-related alerts.

Concept to Learn: Improper scanning can break systems — planning and approvals are key.

2.9 Secure Configuration and Compliance

Secure Configuration Management

- **Harden** systems:
  - Disable unused ports/services
  - Enforce password policies
  - Set permissions properly
- Use **CIS Benchmarks**, **STIGs** for baselines

Compliance Frameworks

- **PCI DSS**
- **HIPAA**
- **NIST 800-53**
- Require:
  - Regular scans
  - Proof of remediation
  - Reporting timelines

Concept to Learn: Compliance often drives vulnerability management schedules — deadlines and documentation matter.

2.10 Exception Handling and Risk Acceptance

Vulnerability Exception Process

- Some vulns can't be fixed immediately.
- Process must include:
  - Risk acceptance form
  - Justification (business or technical)
  - Expiration date for exception
  - Approval from management/security

Tracking Exceptions

- Use ticketing systems (e.g., Jira, ServiceNow).
- Review exceptions periodically.
- Document any compensating controls in place.

Concept to Learn: Accepting risk = formal, documented process — not just ignoring it.

2.11 Secure Software Development & Code Scanning

Secure SDLC

- Integrate security **from design to deployment**.
- Perform:
  - Threat modeling
  - Secure coding reviews
  - Security testing

Static Application Security Testing (SAST)

- Scans source code **without running it**.
- Identifies:
  - Insecure functions
  - Hardcoded passwords
  - Input validation issues

Dynamic Application Security Testing (DAST)

- Tests the application **while running**.
- Simulates real attacks (e.g., SQL injection).
- No access to source code needed.

Software Composition Analysis (SCA)

- Scans third-party libraries for known CVEs.
- Essential due to supply chain threats.

Concept to Learn: Fixing vulnerabilities starts in development. Use SAST, DAST, and SCA tools in CI/CD.

2.12 Attack Surface Management

What is the Attack Surface?

- All possible entry points for an attacker.
- Includes:
  - Open ports
  - Public web apps
  - Third-party tools
  - Exposed APIs

Reducing the Surface

- Uninstall unused software
- Close unnecessary ports
- Harden configurations
- Use cloud security posture tools

External Attack Surface Management (EASM)

- Tools that scan **your internet-facing assets** to find:
  - Forgotten websites
  - Misconfigured DNS
  - Expired TLS certificates

Concept to Learn: Reducing the attack surface = reducing risk. Inventory and minimize what's exposed.

Summary of Domain 2: Vulnerability Management

Master these:

- Vulnerability management lifecycle: discover → prioritize → fix
- Internal, external, credentialed, and passive scanning
- Scan result interpretation: CVEs, CVSS, impact
- Prioritization based on context and risk
- Remediation vs mitigation vs compensating control
- Types of vulnerabilities: web, misconfig, EOL, auth
- Validating and confirming findings (false positives/negatives)
- Patch management process and exceptions
- Secure coding: SAST, DAST, SCA
- Attack surface management concepts
- Compliance requirements for scans and remediation timelines

Domain 3: Incident Response and Management (20%)

Goal of Domain 3:

You must understand how to recognize, respond to, contain, and recover from cybersecurity incidents using formal procedures and frameworks. You'll need to know incident types, response phases, attacker methodologies, and how to coordinate roles, responsibilities, and communication across teams during an incident.

This is where you become **the responder during chaos** — analyzing, documenting, mitigating, and learning from cyber incidents to protect your organization.

3.1 Incident Response Lifecycle

Learn: What is an Incident?

- Any event that:
  - **Violates security policy**
  - **Disrupts operations**
  - **Threatens data confidentiality, integrity, or availability**
- Examples: malware infection, unauthorized access, DDoS attack, data breach

---

Phases of Incident Response (NIST SP 800-61 / SANS PICERL)

1. Preparation

- Pre-incident activities:
  - Develop IR policy and plan
  - Form incident response team (IRT/CSIRT)
  - Define incident severity levels
  - Create incident playbooks (e.g., phishing, ransomware)
  - Train staff and run tabletop exercises
  - Maintain forensic tools and contact lists (legal, PR, law enforcement)
  - Set up logging and alerting infrastructure (SIEM, EDR)

Concept to Learn: Strong preparation = faster response. Without it, everything falls apart.

2. Detection and Analysis

- Identify potential incidents via:
  - Alerts from IDS, SIEM, antivirus
  - User reports
  - Abnormal logs or behavior
- Confirm if it's a true incident
- Categorize type:
  - Malware, web attack, data theft, insider abuse, DoS, etc.
- Determine scope and impact

Concept to Learn: You can't respond to what you can't see — log monitoring, alert tuning, and user awareness are key.

3. Containment

- Stop the incident from spreading or worsening.
- **Short-term** containment:
  - Isolate affected systems
  - Block malicious IPs or domains
  - Disable compromised accounts
- **Long-term** containment:
  - Set up temporary firewall rules
  - Apply WAF protections
  - Segment network traffic

Concept to Learn: Contain first, investigate second — stop the bleeding before diagnosis.

4. Eradication

- Remove the root cause and artifacts:
  - Delete malware or malicious files
  - Remove attacker accounts or tools
  - Patch exploited vulnerabilities
- Validate that systems are clean

Concept to Learn: Eradication is deeper than containment — it ensures the attacker can't come back.

5. Recovery

- Restore operations and monitor.
- Actions:
  - Restore from clean backup
  - Rebuild systems if needed
  - Monitor systems post-recovery for signs of reinfection
- Communicate restoration to stakeholders

Concept to Learn: Don't just reboot — validate, test, and watch carefully before declaring victory.

6. Lessons Learned (Post-Incident Review)

- Hold a **postmortem meeting**:
  - What worked?
  - What failed?
  - Root cause?
  - Time to detect/contain/recover?
- Update IR playbooks
- Submit full **incident report** with timeline and recommendations
- Share IOCs and TTPs with TI platforms or peers (if safe/legal)

Concept to Learn: The value of an incident is in what you learn from it — don't waste that opportunity.

3.2 Incident Types and Severity

Common Incident Types

- **Malware infection**
- **Phishing or spear phishing**
- **Credential compromise**
- **Insider threat**
- **Web application attacks** (SQLi, XSS)
- **Denial-of-Service (DoS/DDoS)**
- **Data exfiltration**
- **Unauthorized access**
- **Misconfiguration or policy violation**
- **Supply chain attack**

Severity Classification

- Based on:
  - Scope of impact
  - Data involved (PII, financial, classified)
  - Systems affected
  - Urgency and business criticality

Example Scale:

- **Critical** – PII exfiltration from core database
- **High** – Malware affecting 10+ systems
- **Medium** – User reports phishing

- **Low** – Policy violation without malicious intent

Concept to Learn: Classifying incidents helps allocate resources, response level, and leadership involvement.

3.3 Roles and Responsibilities in Incident Response

Core Roles

- **Incident Commander** – Leads the response effort
- **SOC Analyst** – Monitors and triages alerts
- **Forensic Analyst** – Acquires and analyzes evidence
- **Malware Analyst** – Reverse-engineers binaries
- **Communications Lead** – Coordinates internal/external updates
- **Legal/Compliance** – Ensures regulatory reporting and risk control
- **Management/Executives** – Decide on business risk, PR

External Stakeholders

- **Law enforcement**
- **Customers/partners**
- **Cyber insurance providers**
- **Vendors or MSSPs**

Concept to Learn: Know who to involve and when — communication and coordination prevent chaos.

3.4 Attack Frameworks and Methodologies

Cyber Kill Chain (Lockheed Martin)

1. **Reconnaissance** – attacker gathers info (open ports, email addresses)
2. **Weaponization** – create exploit, malware
3. **Delivery** – send via phishing, USB, etc.
4. **Exploitation** – trigger vulnerability
5. **Installation** – malware installs
6. **C2 (Command & Control)** – attacker communicates with system
7. **Actions on Objectives** – exfiltrate data, destroy systems, etc.

Concept to Learn: Interrupting the chain early prevents full compromise.

MITRE ATT&CK Framework

- Tactics (goals) like:
  - Initial Access
  - Execution
  - Privilege Escalation
  - Defense Evasion
  - Credential Access
  - Discovery
  - Lateral Movement
  - Exfiltration
  - Impact
- Techniques:
  - Pass-the-Hash
  - PowerShell abuse
  - DLL sideloading
  - Living off the Land Binaries (LOLBins)

Concept to Learn: MITRE helps you **map attacker behavior** and anticipate next steps in a kill chain.

Diamond Model of Intrusion Analysis

- 4 core components:
  - **Adversary** (attacker)
  - **Infrastructure** (C2 servers, phishing domains)
  - **Capability** (tools, malware, exploits)
  - **Victim** (targeted entity)
- Used to pivot analysis between related elements.

Concept to Learn: Analyze attacks in a structured, repeatable way to build better detections.

3.5 Indicators and Evidence Collection

Types of Indicators of Compromise (IOCs):

- **File hashes (MD5, SHA256)**
- **Suspicious IPs/domains**
- **Registry key changes**
- **Unusual process names**
- **Encoded PowerShell commands**
- **Event log anomalies**

Sources of Evidence

- **Memory (RAM) dumps**
- **Disk images**
- **Network captures (pcap)**
- **SIEM logs**
- **Email headers**
- **Firewall logs**
- **Cloud audit logs**

Concept to Learn: Evidence must be preserved **with chain of custody** for potential legal use.

3.6 Forensics and Investigation Techniques

Volatile vs Non-Volatile Data

- **Volatile**: Lost after shutdown (RAM, running processes)
- **Non-Volatile**: Persistent (disk, logs)

Live Response

- Capturing volatile data from a live system.
- Tools: FTK Imager Lite, Sysinternals, Volatility

Disk Forensics

- Make forensic image (bit-by-bit clone)
- Analyze:
  - Deleted files
  - Browser cache
  - File timelines
  - Encryption artifacts

Memory Forensics

- Analyze memory dumps for:
  - Malware
  - Passwords
  - Command history
  - Injected code

Concept to Learn: Start with volatile data — once the system reboots, it's gone.

3.7 Incident Reporting and Documentation

Incident Reports Should Include:

- Timeline of events
- Affected systems
- IOCs and analysis findings
- Root cause
- Actions taken (containment, eradication, recovery)
- Lessons learned
- Future recommendations

Other Documentation

- Ticketing system logs
- Playbook version used
- Communications sent (e.g., customer notifications)

Concept to Learn: If it's not documented, it didn't happen — solid reports support improvements and legal protection.

3.8 Communication During and After an Incident

Internal Communication

- Set up secure war room (chat, call, or ticketing)
- Designate a single source of truth
- Update management at regular intervals

External Communication

- Customers, partners, regulators
- PR and legal teams must approve messaging
- Timing is critical — don't delay required notifications

Out-of-Band Communication

- If attacker might be watching internal email/chat
- Use alternate channels (external phone, secure portal)

Concept to Learn: Clear communication reduces panic and improves coordination — but must be controlled to avoid leaks or legal issues.

3.9 Incident Metrics and Improvement

Key Performance Indicators (KPIs)

- **MTTD (Mean Time to Detect)**
- **MTTC (Mean Time to Contain)**
- **MTTR (Mean Time to Respond/Recover)**
- **Number of repeat incidents**
- **Incidents detected internally vs externally**

Continuous Improvement

- Track metrics
- Adjust tools, training, or staffing
- Update playbooks with new lessons

Concept to Learn: Good IR teams get better over time — but only if they measure and reflect.

Summary of Domain 3: Incident Response and Management

Master these:

- IR lifecycle: Prepare → Detect → Contain → Eradicate → Recover → Learn
- Incident types and classification (malware, phishing, insider, data exfil)
- Roles: analyst, forensic, comms, legal, exec, PR, law enforcement
- Kill Chain, MITRE ATT&CK, Diamond Model
- IOCs and evidence sources
- Live and post-mortem forensics: memory, disk, log, network
- Communication protocols: secure, controlled, timely
- Post-incident reporting, root cause, and lessons learned
- Incident metrics (MTTD, MTTR, KPIs)
- Importance of documentation and legal coordination

Domain 4: Reporting and Communication (17%)

Goal of Domain 4:

You must understand how to translate technical findings into actionable reports, communicate clearly with different audiences (technical, executive, legal), and recommend appropriate mitigation strategies. This domain also covers metrics, remediation planning, stakeholder identification, and the ability to handle sensitive or regulated information properly.

This is where you become a **translator between the technical world and business leadership** — turning analysis into action, reports into results, and alerts into improvements.

4.1 Reporting in Vulnerability Management

Purpose of Vulnerability Reports

- Communicate scan results, risk levels, and action plans to stakeholders.
- Facilitate tracking and documentation for compliance or audits.

Types of Vulnerability Reports

1. Executive Report

- High-level summary
- Focus: business risk, trends, compliance
- Format: charts, KPIs, simple language
- Example:
    - “42 critical vulnerabilities detected in Q1. Down from 89 in Q4. Top risk: unpatched VPN gateway.”

2. Technical Report

- In-depth vulnerability list
- Includes:
    - CVEs, CVSS scores
    - Affected systems
    - Remediation steps
- Targeted at system admins, IT staff

3. Compliance Report

- Tailored to framework (e.g., PCI, HIPAA)
- Demonstrates whether required controls are in place
- Often submitted to auditors

Concept to Learn: Match report format to audience. Executives want trends, tech teams need details.

Common Elements in Reports

- Date of scan/report
- Asset inventory (what was scanned)
- Vulnerability list with:
  - **CVE ID**
  - **CVSS score**
  - **Exploitability**
  - **Affected systems**
- Risk summary (number of critical/high/medium/low)
- Remediation recommendations
- Action plan with owners and deadlines

4.2 Remediation Planning

Learn: How to Turn Findings into Action

Components of a Remediation Plan

1. **What** needs to be fixed (CVE, misconfig)
2. **Where** (affected system)
3. **How** (patch, config change, mitigation)
4. **Who** is responsible
5. **When** (timeline/deadline)
6. **Status** (Open/In Progress/Closed)

Tracking Tools

- Ticketing systems: Jira, ServiceNow, Zendesk
- Spreadsheets for small teams
- Dashboards in SIEM or vuln scanners

Concept to Learn: Action plans must include accountability and timelines — otherwise, nothing gets fixed.

4.3 Risk Acceptance and Exception Handling

Not All Vulns Can Be Fixed Immediately

- Example reasons:
  - System is mission-critical and cannot be rebooted
  - Patch breaks legacy app
  - Vendor patch not yet available

Risk Acceptance Process

1. Identify unremediated vulnerability
2. Justify exception (technical, operational)
3. Apply compensating controls (firewall, segmentation)
4. Document exception:
   - Risk level
   - Approval from management
   - Expiry/review date
5. Track exceptions over time

Compensating Controls Examples

- If no TLS upgrade: restrict access to VPN only
- If no patch: increase monitoring and alerting

Concept to Learn: Accepting risk ≠ ignoring it. It must be justified, documented, and monitored.

4.4 Inhibitors to Remediation

Why Are Some Vulnerabilities Not Fixed Quickly?

1. Operational Constraints

- Patch requires system downtime
- No patching during holidays or fiscal close

2. Technical Conflicts

- Patch breaks app or integration
- System is too old to support update

3. Lack of Resources

- Not enough staff or tooling
- No budget for patch management system

4. Communication Breakdown

- Admins not notified properly
- No clear ownership of asset

5. Business Decision

- Execs accept risk due to cost/benefit analysis

Concept to Learn: You'll need to navigate business realities — not just click "patch all."

4.5 Stakeholder Communication

Who Are the Stakeholders?

- **IT/Infrastructure Teams** – implement patches, configs
- **Developers** – fix application-level vulns
- **Executives** – make budget/risk decisions
- **Legal/Compliance** – ensure proper handling/reporting
- **Security Team** – owns detection and response
- **Business Units** – affected by downtime, risk

Tailoring Communication to Audience

- **Executives**: business risk, compliance status, trendlines
- **Technical Teams**: system names, vulnerability IDs, fix details
- **Legal**: data involved, breach reporting requirements
- **End-users**: brief, non-technical updates (e.g., maintenance notice)

Concept to Learn: Know your audience. Use the right language and detail level to get results.

4.6 Communicating During Incidents

Communication During an Active Incident

- Set up secure **war room** or chat (e.g., Slack, Teams, Zoom)

- Appoint **communication lead**
- Send **regular updates**:
  - What happened
  - What's affected
  - What's being done
  - When is the next update?

Out-of-Band Communication

- If internal email/chat is compromised
- Use:
  - Personal email
  - Phone calls
  - Encrypted external messaging

Escalation Paths

- Clear chain of command:
  - SOC → IR lead → CISO → Legal → Executives
- Know when to involve:
  - **Law enforcement**
  - **Cyber insurance**
  - **Regulators**
  - **Customers**

Concept to Learn: During chaos, structured communication prevents panic and enables smart response.

4.7 Incident Reports and Post-Incident Communication

Incident Report Content

- Summary and timeline
- Incident type and severity
- Affected systems and data
- IOCs and evidence
- Root cause
- Actions taken (containment, eradication, recovery)
- Communications issued
- Legal and compliance steps taken
- **Lessons learned**
- Future recommendations

Who Reads It?

- Executives
- Legal
- Compliance
- SOC analysts (for future reference)

Lessons Learned Session

- Identify gaps:
  - Was detection delayed?
  - Did tools fail?
  - Did communication break down?
- Update:
  - Playbooks
  - Detection rules
  - Training materials

Concept to Learn: Post-incident analysis is how you level up. One incident should prevent ten future ones.

4.8 Metrics and KPIs

Vulnerability Management KPIs

- Number of critical vulns over time
- % of systems patched within SLA
- Mean Time to Remediate (MTTR)
- Recurring vulns (bad patch practices)

Incident Response KPIs

- Mean Time to Detect (MTTD)
- Mean Time to Contain (MTTC)
- Mean Time to Recover (MTTR)
- % of incidents detected by internal vs external sources
- % of post-incident reviews completed

Use of Dashboards

- Display real-time status for management
- Track trends across quarters
- Color-code (e.g., red = noncompliance, green = completed remediation)

Concept to Learn: You can't improve what you don't measure. Metrics drive accountability and funding.

4.9 Communication of Risk and Recommendations

Risk-Based Language

- Use terminology like:
  - "Critical asset exposed"
  - "High probability of data theft"
  - "Compliance risk under PCI DSS"
- Avoid jargon:
  - Don't say: "Apache 2.4.49 has CVE-2021-41773 RCE"
  - Say: "Our public website has a known vulnerability that allows attackers to run code remotely."

Making Recommendations

- Be clear, specific, and actionable.
- Example:
  - "Apply patch KB123456 to all Windows Server 2016 systems in Group X by Friday."
  - "Enable MFA for all admin accounts in AWS."

Concept to Learn: The best security report is one that **gets acted on** — clarity beats cleverness.

4.10 Regulatory Reporting and Legal Communication

When Must You Report Externally?

- Breach of personal data (PII/PHI)
- Incident involving customer impact
- Requirements under:
  - **GDPR** (72-hour notification)
  - **HIPAA**
  - **SOX**
  - **PCI DSS**

Regulatory Communication

- Often handled by legal team
- Analysts may provide:
  - Timeline
  - Logs

    - List of affected data/users
- Should be accurate and reviewed
- Only authorized personnel (PR/legal) should speak publicly

Concept to Learn: External reporting is mandatory in some cases — know the law, and don't wing it.

Summary of Domain 4: Reporting and Communication

Master these:

- Vulnerability report formats (executive, technical, compliance)
- Turning findings into remediation plans with ownership and deadlines
- Risk acceptance and exception documentation
- Why some vulns go unpatched (inhibitors)
- Stakeholder communication: who needs what
- Secure, structured comms during incidents
- Post-incident reports and lessons learned
- Metrics and KPIs for vuln management and incident response
- Making recommendations that business understands
- Reporting to regulators and legal coordination

Terms and Definitions

A

AAA – Authentication, Authorization, Accounting. Core security framework for identity management.

ACL (Access Control List) – List of permissions attached to an object defining who can access what.

APT (Advanced Persistent Threat) – Long-term targeted attack by skilled threat actors, often nation-state sponsored.

Asset – Anything valuable to an organization (data, systems, hardware, etc.).

ATT&CK (Adversarial Tactics, Techniques, and Common Knowledge) – MITRE framework describing attacker behaviors.

B

Baseline – Standard configuration used as a reference for secure settings.

Blue Team – Defensive security team responsible for protecting systems and responding to incidents.

C

CIA Triad – Confidentiality, Integrity, Availability; the foundation of information security.

CIRT/CSIRT – Computer Incident Response Team; team that handles security incidents.

CISO – Chief Information Security Officer; executive overseeing security.

CVE (Common Vulnerabilities and Exposures) – Identifier for known vulnerabilities.

CVSS (Common Vulnerability Scoring System) – Framework for rating severity of vulnerabilities (0.0–10.0).

C2 (Command and Control) – Communication channel used by attackers to manage compromised systems.

D

DAST (Dynamic Application Security Testing) – Testing web applications during runtime for vulnerabilities.

Diamond Model – Intrusion analysis model focused on adversary, infrastructure, victim, and capability.

DoS/DDoS – Denial of Service / Distributed DoS; flood-based attack disrupting services.

DLP (Data Loss Prevention) – Technology that prevents unauthorized data exfiltration.

E

EDR (Endpoint Detection and Response) – Tool that monitors endpoint activity and supports threat response.

Enumeration – Process of gathering information about systems, services, or users.

Exploit – Code or method used to take advantage of a vulnerability.

F

False Positive – Alert triggered for non-malicious activity.

False Negative – A real threat that was not detected by tools.

Firewall – Security device or software controlling traffic based on rules.

G

Gap Analysis – Comparison of current security state vs desired state to find weaknesses.

Greylisting – Temporarily rejecting emails from unknown senders to reduce spam.

H

Hashing – One-way cryptographic function producing a fixed output; used for integrity.

HIDS/NIDS – Host-based/Network-based Intrusion Detection System.

I

IAM (Identity and Access Management) – Framework for managing digital identities and access rights.

IOC (Indicator of Compromise) – Artifact that signals a potential breach (e.g., IP, file hash).

IR (Incident Response) – Process for detecting, containing, and recovering from incidents.

ISO 27001 – International standard for information security management.

K

Kill Chain – Model outlining phases of a cyberattack (Recon to Actions on Objectives).

KPI (Key Performance Indicator) – Metric used to measure effectiveness of a process (e.g., MTTR).

L

Least Privilege – Principle that users should have the minimum access necessary to do their job.

Log Aggregation – Collecting logs from multiple sources for analysis.

M

Malware – Malicious software (e.g., ransomware, trojans, worms).

MFA (Multi-Factor Authentication) – Use of two or more authentication factors.

MITM (Man-in-the-Middle) – Attack where adversary intercepts communication between parties.

MTTD/MTTR – Mean Time to Detect / Respond; time-based metrics for incident response.

N

NIST – National Institute of Standards and Technology; publishes cybersecurity frameworks.

Nmap – Network mapper used to discover hosts and services.

NVD (National Vulnerability Database) – U.S. government repository of vulnerability information.

O

OSI Model – 7-layer model describing network communication (Physical to Application).

OWASP – Open Web Application Security Project; publishes Top 10 web security risks.

P

PBQ (Performance-Based Question) – Hands-on style question in the exam simulating real tasks.

Phishing – Social engineering attack via fraudulent emails.

PKI (Public Key Infrastructure) – Framework for managing digital certificates and public-key encryption.

R

Reconnaissance – The attacker's information-gathering phase.

Red Team – Offensive security professionals simulating attacks for testing defenses.

Risk – Combination of likelihood and impact of a threat exploiting a vulnerability.

RTO/RPO – Recovery Time Objective / Recovery Point Objective; business continuity goals.

S

SAST (Static Application Security Testing) – Code analysis without executing the application.

SIEM (Security Information and Event Management) – Platform aggregating logs and generating alerts.

SLR (Service Level Requirement) – Specific security or uptime commitment in a contract.

SOAR (Security Orchestration, Automation, and Response) – Automates security workflows.

T

TI (Threat Intelligence) – Information about threats to inform defense.

TTPs – Tactics, Techniques, and Procedures used by threat actors.

TLS/SSL – Protocol for encrypting web traffic (HTTPS).

Threat Actor – Entity responsible for a threat (e.g., APT, insider, script kiddie).

U

UDP (User Datagram Protocol) – Connectionless protocol; often used for streaming or DNS.

UAC (User Account Control) – Windows security feature that prompts before changes.

V

VPN (Virtual Private Network) – Encrypted tunnel between client and network.

Vulnerability – A weakness in a system that can be exploited.

Vulnerability Scanner – Tool that detects missing patches, misconfigurations (e.g., Nessus).

W

WAF (Web Application Firewall) – Filters HTTP traffic to protect web apps from exploits.

Whitelisting – Allow list; blocks everything except explicitly approved items.

Wireshark – Tool for analyzing packet captures.

X

XDR (Extended Detection and Response) – Unified threat detection across endpoints, network, and cloud.

Z

Zero-Day – Vulnerability not yet known or patched by vendor.

Zero Trust – Security model assuming no implicit trust — always authenticate and verify.

MITRE ATT&CK Tactics and Examples

Tactic	Description	Common Techniques
Initial Access	How attackers gain entry	Phishing, Exploit Public-Facing App, Drive-by
Execution	Run malicious code	PowerShell, Macros, Command-Line Interface
Persistence	Maintain access	Startup scripts, Registry Run Keys, New Services
Privilege Escalation	Gain higher-level access	Exploiting SUID, Bypassing UAC, Token Manipulation
Defense Evasion	Avoid detection	Obfuscation, Disabling AV, Masquerading
Credential Access	Steal user credentials	Keylogging, LSASS dump, Brute Force
Discovery	Understand the environment	Network scans, AD queries, File/System enumeration
Lateral Movement	Move between systems	Pass-the-Hash, RDP, SMB shares
Collection	Gather target data	Screen capture, Email scraping, Clipboard logging
Command & Control	Communicate with compromised systems	DNS Tunneling, Web Traffic, Custom Protocols
Exfiltration	Steal data	HTTPS uploads, Cloud sync, Removable media
Impact	Disrupt operations or destroy data	Ransomware, Data Wipe, DDoS