Here's a structured and detailed report outline for your SOC Operations Documentation project, including all required components. Let me know if you'd like help filling in specific sections, creating diagrams, or gathering screenshots.

SOC Operations Documentation Report

Table of Contents

1. Introduction

This document outlines essential components of a Security Operations Center (SOC), demonstrating proficiency in tools, workflows, and operational processes. It aims to serve as both a training reference and operational guide for SOC personnel.

2. Overview of SOC Tools

A. SIEM Systems (e.g., Splunk, QRadar)

Purpose:
Security Information and Event Management systems collect, correlate, and analyze log and event data from across the IT environment.

Functionality:

- Log aggregation
- Event correlation and alerting
- Threat detection
- Dashboards and reports

Screenshot: *Insert Splunk/QRadar dashboard screenshot here*

Explanation:
SIEM platforms are critical for identifying suspicious activity, centralizing visibility, and automating threat detection.

B. Ticketing Platforms (e.g., ServiceNow, Jira, RTIR)

Purpose:
Manage the lifecycle of security incidents and ensure proper tracking, resolution, and auditing.

Functionality:

- Create, assign, and track tickets
- Set SLAs and escalation rules
- Integration with SIEM and communication tools

Screenshot: *Insert ServiceNow incident ticket screen*

Explanation:
Ticketing platforms enforce accountability and ensure consistent incident resolution.

C. Monitoring Solutions (e.g., Zabbix, Nagios, SolarWinds)

Purpose:
Real-time visibility into infrastructure health and performance.

Functionality:

- Availability and performance monitoring
- Alerts for failures or threshold breaches
- Visualization dashboards

Screenshot: *Insert monitoring dashboard*

Explanation:
Monitoring tools complement SIEMs by providing system-level alerts and uptime visibility.

3. SOC Workflows

Alert Handling & Escalation Path

```
flowchart TD
    A[Alert Detected by SIEM] --> B{Is Alert Valid?}
    B -- No --> C[Close Alert - False Positive]
    B -- Yes --> D[Create Ticket in Ticketing System]
    D --> E[Initial Triage by Tier 1 Analyst]
    E --> F{Severity Assessment}
    F -- Low --> G[Monitor]
    F -- Medium --> H[Tier 2 Investigation]
    F -- High/Critical --> I[Escalate to Tier 3 / IR Team]
    I --> J[Incident Response Initiated]
    J --> K[Resolution & Recovery]
    K --> L[Close Ticket & Document]
    L --> M[Shift Handover & Knowledge Transfer]
```

4. Shift Transition Procedures

A. Handover Requirements

- **Ticket Summary:** Open incidents, pending actions

- **System Health Overview:** Current monitoring status
- **Recent Alerts:** Any high/critical alerts in the last shift
- **Ongoing Investigations:** Timeline and next steps
- **Notes:** Analyst observations or known issues

B. Communication Best Practices

- Use a shared handover document or platform
- Maintain logs for each transition
- Conduct verbal or virtual briefings when necessary

5. Incident Handling Steps

Template:

Step	Description
Detection	Alert received via SIEM or monitoring tools
Triage	Validate alert, determine scope and severity
Investigation	Gather logs, run queries, correlate data
Containment	Isolate affected systems if necessary
Eradication	Remove threat from environment
Recovery	Restore systems and services
Lessons Learned	Conduct post-incident review
Documentation	Update ticket with full timeline, actions, and indicators

6. Screenshots and Operational Concept Explanations

A. SIEM Alert Example:

- Screenshot: *Splunk alert panel*
- Explanation: Shows real-time correlation of login anomalies

B. Ticket Example:

- Screenshot: *ServiceNow open incident screen*
- Explanation: Contains metadata, history, and resolution fields

C. Monitoring Dashboard:

- Screenshot: *Zabbix system status*
- Explanation: Highlights failed hosts and high CPU usage alerts

7. Conclusion

This documentation illustrates a comprehensive understanding of SOC operations, including key tools, workflows, incident management, and handover protocols. It is designed for practical use by analysts and continuous improvement of SOC processes.

8. Appendix

Mermaid Diagram Code

```
flowchart TD
    A[Alert Detected by SIEM] --> B{Is Alert Valid?}
    B -- No --> C[Close Alert - False Positive]
    B -- Yes --> D[Create Ticket in Ticketing System]
    D --> E[Initial Triage by Tier 1 Analyst]
    E --> F{Severity Assessment}
    F -- Low --> G[Monitor]
    F -- Medium --> H[Tier 2 Investigation]
    F -- High/Critical --> I[Escalate to Tier 3 / IR Team]
    I --> J[Incident Response Initiated]
    J --> K[Resolution & Recovery]
    K --> L[Close Ticket & Document]
    L --> M[Shift Handover & Knowledge Transfer]
```

Would you like help filling in the screenshots or editing this into a PDF or Word format?

Here's a **detailed report for the "Document SIEM Implementation"** project that meets all required elements, formatted professionally and ready for expansion with screenshots and tool-specific examples (e.g., Splunk, QRadar, or Elastic Stack).

SIEM Implementation Documentation

Table of Contents

1. Introduction

This document details the implementation of a Security Information and Event Management (SIEM) system, explaining its architecture, rule creation, log source relevance, and notification settings. It demonstrates functional comprehension suitable for SOC analysts and system integrators.

2. SIEM Architecture Components

A typical SIEM system includes the following:

A. Data Sources

- **Function**: Devices and applications that generate log data (e.g., firewalls, servers, EDR agents).
- **Relationship**: Serve as input to the collection layer.

B. Collection & Parsing Layer

- **Function**: Gathers log data using agents or APIs and normalizes it into a common format.
- **Tools**: Universal Forwarders (Splunk), Winlogbeat (Elastic), syslog daemons.
- **Relationship**: Feeds clean, normalized logs into the correlation engine.

C. Correlation Engine

- **Function**: Applies rules to detect suspicious patterns and anomalies.
- **Features**: Rule-based, statistical, or machine learning logic.
- **Relationship**: Sits at the heart of threat detection.

D. Storage Layer

- **Function**: Long-term log retention and archival.
- **Technology**: Indexers (Splunk), ElasticSearch nodes.
- **Relationship**: Supports compliance, audit, and historic investigations.

E. Dashboard & Alerting

- **Function**: Presents visual insights and alerts to analysts.
- **Examples**: Splunk dashboards, QRadar offenses, Kibana visualizations.
- **Relationship**: Analyst-facing layer for real-time monitoring and response.

3. Sample Correlation Rule

A. Use Case

Detect multiple failed login attempts from a single IP within 5 minutes, potentially indicating a brute-force attack.

B. Rule Template (Example for Elastic SIEM / Splunk Pseudocode)

```
Rule Name: Multiple Failed Logins from Same IP
Condition:
  event_type = "login_failure" AND
  count(src_ip) > 5 within 5 minutes
Action:
  Trigger alert
  Notify SOC channel
```

C. Logic Explanation

- **event_type = "login_failure"** filters relevant logs.
- **count(src_ip) > 5** identifies repeated failures from a single source.
- **within 5 minutes** defines the time window to detect burst activity.

4. Key Log Sources and Their Significance

A. Windows Event Logs

- **Source**: Windows servers and endpoints.
- **Importance**: Logs include authentication events, system errors, user logins, privilege escalations.
- **Use Cases**: Detect RDP brute force, privilege misuse.

B. Firewall Logs

- **Source**: Network perimeter devices (e.g., Palo Alto, Cisco ASA).
- **Importance**: Logs blocked connections, port scans, allowed inbound traffic.
- **Use Cases**: Detect reconnaissance, lateral movement, unauthorized access.

C. Endpoint Detection & Response (EDR) Logs

- **Source**: CrowdStrike, SentinelOne, Microsoft Defender.
- **Importance**: Capture processes, file activity, network access at host level.
- **Use Cases**: Detect malware execution, process injection, data exfiltration attempts.

5. Notification Configuration

A. Alerting Channels

- **Email**: SOC distribution list
- **Slack/MS Teams**: Integration via webhooks
- **Ticketing**: Integration with ServiceNow or Jira

B. Escalation Settings

- **Criticality Mapping**: Severity 1-5 levels
- **Auto-assignment**: Tier 1 for low/medium, Tier 2+ for high severity

C. Sample Configuration (Generic)

```
{
  "alert_name": "Brute Force Login Alert",
  "severity": "High",
  "actions": [
    {
      "type": "email",
      "recipients": ["soc-alerts@example.com"]
    },
    {
      "type": "webhook",
      "url": "https://hooks.slack.com/services/ABC123"
    },
    {
      "type": "ticket",
      "system": "ServiceNow",
      "auto_assign": "Tier2"
    }
  ]
}
```

Screenshot Placeholder:
Insert screenshot of notification rule settings from your SIEM (e.g., Splunk alert action settings, Elastic detection rule UI, or QRadar offense notification setup)

6. Screenshots & Tool Interfaces

(Ensure all screenshots include time stamps, rule names, log examples, or alert conditions)

1. **Log Source Ingestion Interface**
2. **Correlation Rule Builder**
3. **Alert Trigger Example**
4. **Notification Channel Configuration**

7. Conclusion

This documentation demonstrates a working understanding of SIEM architecture, log source integration, detection rule creation, and alert notification. A functional SIEM enhances an organization's ability to proactively detect and respond to threats across its environment.

Would you like me to generate a real-world correlation rule for Splunk/Elastic or create example screenshots for each section?

Certainly! Below is a **detailed report for "Document Threat Detection Principles"**, fully structured to meet the specified requirements. You can use this as a comprehensive reference or submission, and it can be supplemented with screenshots and examples from your lab environment or SIEM platform.

Threat Detection Principles Documentation

Table of Contents

1. Introduction

This document explores the core principles of threat detection in a Security Operations Center (SOC). It focuses on rule mechanisms, detection scenarios, threat indicators, structured threat analysis, and a practical alert investigation exercise. The goal is to equip SOC analysts with both conceptual and operational knowledge for identifying and mitigating cyber threats.

2. Detection Rule Mechanisms

Detection rules are logic-based conditions used to identify suspicious activity across various data sources. These mechanisms can be categorized into:

A. Signature-Based Detection

- **Definition:** Matches known patterns (hashes, command lines).
- **Use Case:** Detecting known malware variants.
- **Pros:** Fast, low false positives.
- **Cons:** Ineffective against zero-day threats.

B. Behavior-Based Detection

- **Definition:** Identifies deviations from expected behavior.
- **Use Case:** Detecting lateral movement or data exfiltration.
- **Pros:** Detects unknown threats.
- **Cons:** Requires baselining and can generate false positives.

C. Heuristic/Anomaly-Based Detection

- **Definition:** Uses statistical models or machine learning to identify anomalies.
- **Use Case:** Login attempts at odd hours, geographic anomalies.
- **Pros:** Adaptive to evolving threats.
- **Cons:** Higher tuning effort required.

D. Correlation-Based Detection

- **Definition:** Links multiple low-level events across systems to create a meaningful detection.
- **Use Case:** Failed logins + PowerShell execution + file modification.
- **Pros:** Contextual and layered detection.
- **Cons:** Rule complexity and dependency on log completeness.

3. Detection Scenarios with Examples

Scenario 1: Brute Force Login Attempt

Rule Logic:

- More than 5 failed login attempts from the same IP in 3 minutes.

Detection Query (Elastic SIEM Example):

```
event.type: "authentication_failure" AND count(source.ip) >
5 within 3m
```

Response:

- Lock source IP temporarily.

- Alert SOC and create investigation ticket.

Scenario 2: Suspicious PowerShell Execution

Indicators:

- PowerShell used with obfuscated script or suspicious parameters.

Detection Query (Splunk Example):

```
index=windows sourcetype=WinEventLog*
(Image="*powershell.exe" AND CommandLine="*bypass*")
```

Response:

- Retrieve parent-child process chain.
- Investigate user and host context.

Scenario 3: Data Exfiltration via Cloud Storage

Indicators:

- Unusual upload volume to Dropbox/Google Drive.

Detection Query (Firewall/Proxy Logs):

```
(destination.domain: "dropbox.com" OR "drive.google.com")
AND bytes_out > 1GB
```

Response:

- Isolate host.
- Confirm if the activity was user-initiated or malicious.

4. Threat Indicator Categories

A. Indicators of Compromise (IOCs)

Definition: Artifacts known to be associated with malicious activity.

- **Examples:** IPs, domains, hashes, file paths.
- **Application:** Used in blacklists and detection rules.

B. Indicators of Attack (IOAs)

Definition: Behaviors or sequences suggesting malicious intent.

- **Examples:** Unusual registry modification, suspicious script execution.
- **Application:** Enhance behavioral detection and alert triage.

C. Tactics, Techniques, and Procedures (TTPs)

Definition: Attacker's methodology from frameworks like MITRE ATT&CK.

- **Examples:** T1059 – Command-line execution.
- **Application:** Used for threat modeling and coverage analysis.

5. Structured Threat Analysis Methodology

A consistent methodology helps ensure thorough investigation and triage. Based on guidelines, here's a structured approach:

Step 1: Initial Alert Review

- Check alert source, time, and rule triggered.

Step 2: Context Gathering

- Who is the user?
- What system is affected?
- Were there previous alerts?

Step 3: Event Correlation

- Look for related events before and after the alert.
- Check for other hosts or users involved.

Step 4: Hypothesis Building

- Example: "User credentials may be compromised."

Step 5: Validation

- Run threat intelligence enrichment, review logs, check EDR activity.

Step 6: Response & Documentation

- Contain threat if necessary.
- Document actions taken and lessons learned.

6. Alert Investigation Exercise

Scenario:
An alert triggered for suspicious PowerShell activity on a domain-joined host used by a finance user.

Investigation Steps:

Step	Action	Findings
Initial Alert	PowerShell with encoded command	Alert from Elastic SIEM
Host Info	Checked hostname `FIN-LAPTOP-03`	Domain-joined, used by `j.doe`
User	Login pattern review	Consistent usage history
Process Chain	Parent = `winword.exe`, child = `powershell.exe`	Abnormal execution path
Network	Outbound to unknown IP	Potential command-and-
EDR	Process flagged by SentinelOne	Malicious behavior detected

Conclusion:

The alert was **confirmed** as **malicious macro attack** leading to PowerShell-based execution and outbound connection. Host was isolated and user credentials reset.

7. Conclusion

This report demonstrated understanding of fundamental detection concepts including rule mechanisms, scenario-based detection, and threat indicator categorization. It outlined a structured approach to threat analysis and successfully documented a hands-on alert investigation aligned with SOC workflows and detection methodologies.

Would you like me to generate visual diagrams (e.g., MITRE ATT&CK mapping or alert triage flow) or sample screenshots for your SIEM platform?