Incident Response and Disaster Recovery Plan:

Introduction

Incident Response (IR) and Disaster Recovery (DR) are essential for minimizing security risks and ensuring business continuity. The following plan outlines automated logging, predefined response playbooks, recovery mechanisms, and communication strategies for effective security incident handling.

Different organizations have developed standardized frameworks for Incident Response (IR) to ensure a structured and effective approach to handling security incidents.

NIST (SP 800-61) outlines a four-phase approach:

(a) Preparation – Establish policies, tools, and teams for incident response.
(b) Detection & Analysis – Identify, analyze, and assess security incidents.
(c) Containment, Eradication & Recovery – Isolate threats, remove malicious elements, and restore operations.
(d) Post-Incident Activity – Conduct reviews, document lessons, and improve defenses.

SANS Incident Handler's Handbook follows a six-phase process:

(a) Preparation – Develop response plans, tools, and security controls.
(b) Identification – Detect and verify security incidents.
(c) Containment – Limit the damage by isolating affected systems.
(d) Eradication – Remove malware, compromised accounts, or vulnerabilities.
(e) Recovery – Restore systems to normal operations securely.
(f) Lessons Learned – Analyze the incident, document improvements, and refine security measures.

Both frameworks emphasize proactive preparation, structured response, and continuous improvement, ensuring organizations efficiently manage security incidents and strengthen resilience.

Implementing Automated Logging and Alerting for Security Events

Why is it important?

- Logs capture security events (e.g., unauthorized access, failed login attempts, malware detection).
- Alerts notify teams when anomalies are detected, allowing quick action.
- Helps in forensic analysis post-incident.

Tools for Different Scenarios

Scenario	Tool Used	Reason
System-wide security logging	SIEM (Splunk, ELK Stack, Wazuh, Graylog)	Centralized log collection and analysis
Network security monitoring	Zeek (Bro), Suricata, Wireshark	Detects network intrusions
Cloud Security Events	AWS CloudTrail, Azure Security Center	Monitors cloud-based security logs
Endpoint Detection and Response (EDR)	Microsoft Defender for Endpoint, CrowdStrike	Detects malware & unauthorized access

Implementation Steps

I. **Enable Logging Across All Systems**
   - Windows Event Logs
   - Linux Audit Logs (`auditd`)
   - Cloud security logs (AWS, Azure)

II. **Centralize Logs in a SIEM**

```
Example: Forward logs to Elastic Stack
filebeat modules enable system
filebeat setup -e
service filebeat start
```

III. **Configure Alerts for Critical Events**
   - Failed login attempts > 3 times → Send alert to SOC team.
   - New admin user added outside office hours → Trigger investigation.
   - Unusual data access (privilege escalation) → Log, block, and alert.

Result: The security team gets real-time alerts and logs for threat monitoring.

Define Incident Response Playbooks

Why are Playbooks Needed?

- They provide step-by-step procedures to handle security incidents.
- Ensures **c**onsistent response actions for all incidents.
- Reduces response time and business impact.

Playbooks for Different Security Incidents

Scenario 1: SQL Injection Attack

Detection Tools:

- Web Application Firewall (ModSecurity, Cloudflare WAF)
- Database Activity Monitoring (Imperva, SQLmap)
  Response Steps:

1. Alert Security Team (via SIEM).
2. Block malicious IPs using firewall.
3. Extract attack details (timestamp, affected database, attacker's IP).
4. Patch vulnerable SQL queries (`Prepared Statements` or `ORM`).
5. Perform a security review to prevent future injections.

Scenario 2: Unauthorized Access Detected

Detection Tools:

- EDR Solutions (CrowdStrike, Microsoft Defender)
- Log Analysis (Splunk, Elastic Stack)

Response Steps:

1. Revoke session tokens for the compromised user.
2. Lock affected user account in Active Directory.
3. Force a password reset for the user.
4. Check access logs to assess potential data theft.
5. Apply multi-factor authentication (MFA) for future access.

Scenario 3: Privilege Escalation

Detection Tools:

- EDR & SIEM (IBM QRadar, Wazuh)
- Behavioral Analysis (Darktrace, Exabeam)

Response Steps:

a) Kill unauthorized processes running with admin privileges.
b) Remove the attacker’s elevated privileges immediately.
c) Analyze logs for exploit techniques (kernel exploits, credential dumping).
d) Deploy patches for the exploited vulnerability.
e) Audit all user roles and permissions.

Result: Playbooks ensures rapid and effective response to security threats.

Set Up Automated Disaster Recovery for Critical Failures

Why is Disaster Recovery Important?

- Ensures business continuity in case of server crashes, ransomware, or data loss.
- Reduces downtime by restoring services quickly.

Disaster Recovery Tools & Methods

Scenario	Tool Used	Reason
Backup & Restore	Veeam, Acronis, AWS Backup, Azure Backup	Regular automated backups
Server Crash Recovery	VM Snapshots (VMware, Hyper-V), Terraform, Ansible	Quickly restore lost VMs
Cloud Failover	AWS Auto Scaling, Azure Site Recovery	Ensures high availability
Database Resilience	MySQL Replication, PostgreSQL Streaming	Avoids single points of failure

Disaster Recovery Plan

I. **Automated Backups**

- Frequency: Daily for databases, weekly for full system backups.
- Tool Example: AWS Backup (Cloud), Veeam (On-Prem).

Example: Windows Server Backup Automation

```
wbadmin start backup -backupTarget:D: -include:C: -quiet
```

II. **Failover Mechanism**

- If AWS EC2 instance crashes, trigger an auto-scale replacement.

```
aws ec2 create-instance --image-id ami-12345678 --count 1
```

III. **Automated Ransomware Protection**

- Immutable backups (cannot be modified by malware).
- Zero-trust network segmentation (blocks unauthorized lateral movement).

IV. **Disaster Recovery Testing**

- Conduct DR drills in every 6 months.
- Simulate data breaches, server crashes, and malware attacks.

Result: Minimized downtime, data integrity maintained, business continuity ensured.

Communication Strategies & Post-Incident Reviews

Why is Communication Important?

- Ensures stakeholders are informed in a security incident.
- Reduces panic & misinformation.
- Helps in regulatory compliance (GDPR, HIPAA).

Communication Tools & Methods

Scenario	**Tool Used**	**Reason**
Incident Alerts	PagerDuty, Microsoft Teams, Slack Webhooks	Real-time notifications
Post-Incident Reporting	Confluence, Jira, ServiceNow	Documentation & tracking
Executive-Level Updates	Email Templates, Zoom Briefings	Formal communication

Communication Plan

I. **Security Alert System**

- High-priority incidents → SOC team (PagerDuty Alert).
- Low-priority alerts → Email to IT Admins.

II. **Post-Incident Review Process**

- Root cause analysis (RCA).
- Lessons learned documentation.
- Update security controls to prevent future attacks.

Result: Stakeholders are informed, compliance is maintained, and processes are improved.

Conclusion: A well-defined Incident Response and Disaster Recovery (IR/DR) plan ensures quick threat detection, efficient response, and minimal downtime. Automated security logging, structured playbooks, and disaster recovery mechanisms enhance resilience against cyber threats. Furthermore, regular backups, failover strategies, and clear communication protocols strengthen business continuity. This proactive approach reduces risks, ensures compliance, and maintains operational stability.