# Incident Management Standard Operating Procedure (SOP)

Version: 1.1

Last Updated: October 2025

Prepared by: IT Operations Team

## 1. Objective

This document defines the standard operating procedure for identifying, triaging, resolving, and closing production incidents detected by monitoring tools or PagerDuty alerts.

## 2. Scope

Applicable to all production applications, servers, containers, and infrastructure components managed by the DevOps and IT Operations teams.

## 3. Common Incident Scenarios

- Database connection refused
- API latency spike
- CPU usage > 90%
- Server not reachable
- Disk space full

## 4. Roles & Responsibilities

- On-Call Engineer: First responder to PagerDuty incidents.
- Service Owner: Approves and validates fixes.
- Incident Manager: Coordinates communication and escalations.
- DevOps Engineer: Supports infrastructure-related fixes.

## 5. Incident Severity Levels

P1: Critical outage affecting multiple users (Response 15 min)

P2: Major component unavailable (Response 30 min)

P3: Minor degradation or error (Response 60 min)

## 6. General Response Workflow

1. Alert received from PagerDuty
2. Acknowledge incident within 5 minutes
3. Investigate issue and logs
4. Execute remediation steps
5. Update incident in PagerDuty and Slack
6. Verify recovery and close
7. RCA within 48 hours

## 7. Resolution Procedures by Scenario

### 7.1 Server Not Reachable

Alert Example: Ping check failed for host or SSH unreachable

Severity: P1 if production impacted

Resolution Steps:

1. Validate if issue is isolated: ping/traceroute server
2. Attempt SSH login to host
3. Check server status in cloud console
4. Restart VM if unresponsive
   - Azure: az vm restart --name app-server-01 --resource-group prod-rg
   - AWS: aws ec2 reboot-instances --instance-ids i-0abcd12345
5. Once reachable, verify service health: curl -s http://app-server-01:8080/health
6. If still unreachable, escalate to Network Operations (L2)
7. Update PagerDuty and Slack

### 7.2 Connection Refused

1. Check DB or API service: systemctl status <service>
2. Restart if down: systemctl restart <service>
3. Validate connectivity: telnet db-server 5432
4. Escalate to DBA team if needed.

### 7.3 High CPU Usage

1. Identify high CPU process: top -o %CPU

2. Restart process or scale out service.

## 7.4 Disk Space Full

1. Check disk usage: df -h
2. Clear temp/log files: rm -rf /var/log/*.gz
3. Archive large files.

## 8. Escalation Policy

L1: On-Call Engineer (15 min)

L2: Network/Infra (30 min)

L3: Service Owner (1 hr)

L4: Head of IT Ops (2 hr)

## 9. Post-Incident Review

Conduct RCA within 48 hours.

Document root cause, fix, and preventive action.

## 10. Reference Commands

- curl -s http://<service>/health
- sudo systemctl restart <service>
- journalctl -u <service> -n 100
- az vm restart --name <vm> --resource-group <rg>
- docker restart <container>

## 11. Quick Escalation Message

Incident: app-server-01 unreachable (P1)

Action: Server restarted via Azure CLI

Status: Recovered

Next Step: RCA pending