Security Procedures

Overview

This document outlines comprehensive security procedures for the HX-Infrastructure-Ansible automation platform, covering security scanning, compliance validation, incident response, and ongoing security maintenance.

Security Framework

Security Principles

- **Defense in Depth**: Multiple layers of security controls
- **Least Privilege**: Minimal access rights for users and systems
- **Zero Trust**: Verify everything, trust nothing
- **Continuous Monitoring**: Ongoing security assessment and monitoring
- **Incident Response**: Rapid detection and response to security events

Compliance Standards

- **CIS Benchmarks**: Center for Internet Security guidelines
- **NIST Cybersecurity Framework**: Risk management framework
- **SOC 2**: Security and availability controls
- **ISO 27001**: Information security management
- **PCI DSS**: Payment card industry standards (if applicable)

Security Scanning Procedures

Automated Security Scanning

Daily Security Scans

```
# Run comprehensive security scan
ansible-playbook playbooks/security/daily_scan.yml -i inventory.yml

# Check for critical vulnerabilities
./scripts/security/check_critical_vulns.sh

# Update security dashboards
ansible-playbook playbooks/monitoring/update_security_dashboard.yml -i inventory.yml
```

Weekly Security Assessment

```
# Full infrastructure scan
ansible-playbook playbooks/security/weekly_assessment.yml -i inventory.yml

# Compliance check
ansible-playbook playbooks/security/compliance_check.yml -i inventory.yml

# Generate security report
./scripts/security/generate_weekly_report.sh
```

Monthly Security Review

```
# Comprehensive security audit
ansible-playbook playbooks/security/monthly_audit.yml -i inventory.yml

# Review access controls
ansible-playbook playbooks/security/access_review.yml -i inventory.yml

# Update security policies
ansible-playbook playbooks/security/update_policies.yml -i inventory.yml
```

Manual Security Procedures

Vulnerability Assessment

```
# Scan for vulnerabilities
nmap -sV -sC target-hosts
nessus-scan --targets inventory.yml

# Check for misconfigurations
ansible-playbook playbooks/security/config_audit.yml -i inventory.yml

# Review security logs
ansible all -i inventory.yml -m shell -a "grep -i 'failed\|error\|denied' /var/log/
auth.log"
```

Penetration Testing

```
# External penetration test
# (Performed by authorized security team only)

# Internal security assessment
ansible-playbook playbooks/security/internal_pentest.yml -i inventory.yml

# Web application security test
ansible-playbook playbooks/security/webapp_security_test.yml -i inventory.yml
```

Access Control Management

User Access Management

User Provisioning

```
# Create new user account
ansible-playbook playbooks/security/create_user.yml -i inventory.yml \
  --extra-vars "username=newuser groups=developers"

# Set up SSH keys
ansible-playbook playbooks/security/setup_ssh_keys.yml -i inventory.yml \
  --extra-vars "username=newuser ssh_key_file=~/.ssh/newuser.pub"

# Configure sudo access
ansible-playbook playbooks/security/configure_sudo.yml -i inventory.yml \
  --extra-vars "username=newuser sudo_commands='ALL=(ALL) NOPASSWD: /usr/bin/sys-
temctl'"
```

User Deprovisioning

```
# Disable user account
ansible-playbook playbooks/security/disable_user.yml -i inventory.yml \
  --extra-vars "username=olduser"

# Remove SSH keys
ansible-playbook playbooks/security/remove_ssh_keys.yml -i inventory.yml \
  --extra-vars "username=olduser"

# Archive user data
ansible-playbook playbooks/security/archive_user_data.yml -i inventory.yml \
  --extra-vars "username=olduser"
```

Access Review

```
# Review user accounts
ansible all -i inventory.yml -m shell -a "cut -d: -f1 /etc/passwd"

# Check sudo access
ansible all -i inventory.yml -m shell -a "grep -v '^#' /etc/sudoers"

# Review SSH keys
ansible all -i inventory.yml -m shell -a "find /home -name authorized_keys -exec cat
{} \;"

# Check failed login attempts
ansible all -i inventory.yml -m shell -a "grep 'Failed password' /var/log/auth.log |
tail -20"
```

Service Account Management

Service Account Creation

```
# Create service account
ansible-playbook playbooks/security/create_service_account.yml -i inventory.yml \
  --extra-vars "service_name=monitoring service_user=monitor"

# Generate API keys
ansible-playbook playbooks/security/generate_api_keys.yml -i inventory.yml \
  --extra-vars "service_name=monitoring"

# Configure service permissions
ansible-playbook playbooks/security/configure_service_permissions.yml -i
inventory.yml \
  --extra-vars "service_name=monitoring"
```

Service Account Rotation

```
# Rotate service account credentials
ansible-playbook playbooks/security/rotate_service_credentials.yml -i inventory.yml

# Update API keys
ansible-playbook playbooks/security/update_api_keys.yml -i inventory.yml

# Verify service functionality
ansible-playbook playbooks/security/verify_service_access.yml -i inventory.yml
```

Secrets Management

Ansible Vault Management

Creating Encrypted Variables

```
# Create new vault file
ansible-vault create group_vars/production/vault.yml

# Edit existing vault file
ansible-vault edit group_vars/production/vault.yml

# Encrypt existing file
ansible-vault encrypt group_vars/production/secrets.yml

# Decrypt file for editing
ansible-vault decrypt group_vars/production/secrets.yml
```

Vault Key Rotation

```
# Change vault password
ansible-vault rekey group_vars/production/vault.yml

# Re-encrypt with new key
ansible-vault rekey group_vars/*/vault.yml

# Update vault password files
echo "new_password" > .vault_pass_production
chmod 600 .vault_pass_production
```

External Secrets Management

HashiCorp Vault Integration

```
# Configure Vault authentication
ansible-playbook playbooks/security/configure_vault_auth.yml -i inventory.yml

# Retrieve secrets from Vault
ansible-playbook playbooks/security/retrieve_vault_secrets.yml -i inventory.yml

# Rotate Vault tokens
ansible-playbook playbooks/security/rotate_vault_tokens.yml -i inventory.yml
```

AWS Secrets Manager

```
# Configure AWS credentials
ansible-playbook playbooks/security/configure_aws_secrets.yml -i inventory.yml

# Retrieve secrets from AWS
ansible-playbook playbooks/security/retrieve_aws_secrets.yml -i inventory.yml

# Update secrets in AWS
ansible-playbook playbooks/security/update_aws_secrets.yml -i inventory.yml
```

Network Security

Firewall Management

Firewall Configuration

```
# Configure host-based firewall
ansible-playbook playbooks/security/configure_firewall.yml -i inventory.yml

# Update firewall rules
ansible-playbook playbooks/security/update_firewall_rules.yml -i inventory.yml \
  --extra-vars "allow_ports=[22,80,443] deny_ports=[23,21]"

# Check firewall status
ansible all -i inventory.yml -m shell -a "ufw status verbose"
```

Network Segmentation

```
# Configure network zones
ansible-playbook playbooks/security/configure_network_zones.yml -i inventory.yml

# Set up VPN access
ansible-playbook playbooks/security/setup_vpn.yml -i inventory.yml

# Configure network monitoring
ansible-playbook playbooks/security/setup_network_monitoring.yml -i inventory.yml
```

SSL/TLS Management

Certificate Management

```
# Generate SSL certificates
ansible-playbook playbooks/security/generate_ssl_certs.yml -i inventory.yml \
  --extra-vars "domain=example.com"

# Install certificates
ansible-playbook playbooks/security/install_ssl_certs.yml -i inventory.yml

# Check certificate expiration
ansible all -i inventory.yml -m shell -a "openssl x509 -in /etc/ssl/certs/server.crt -
noout -dates"
```

Certificate Renewal

```
# Renew Let's Encrypt certificates
ansible-playbook playbooks/security/renew_letsencrypt.yml -i inventory.yml

# Update certificate stores
ansible-playbook playbooks/security/update_cert_stores.yml -i inventory.yml

# Restart services after renewal
ansible-playbook playbooks/security/restart_ssl_services.yml -i inventory.yml
```

Compliance and Auditing

Compliance Scanning

CIS Benchmark Compliance

```
# Run CIS benchmark scan
ansible-playbook playbooks/security/cis_benchmark.yml -i inventory.yml

# Generate compliance report
./scripts/security/generate_cis_report.sh

# Remediate CIS findings
ansible-playbook playbooks/security/remediate_cis.yml -i inventory.yml
```

NIST Framework Assessment

```
# NIST cybersecurity assessment
ansible-playbook playbooks/security/nist_assessment.yml -i inventory.yml

# Generate NIST report
./scripts/security/generate_nist_report.sh

# Track remediation progress
ansible-playbook playbooks/security/track_nist_remediation.yml -i inventory.yml
```

Audit Logging

Log Configuration

```
# Configure audit logging
ansible-playbook playbooks/security/configure_audit_logging.yml -i inventory.yml

# Set up log forwarding
ansible-playbook playbooks/security/setup_log_forwarding.yml -i inventory.yml

# Configure log retention
ansible-playbook playbooks/security/configure_log_retention.yml -i inventory.yml
```

Log Analysis

```
# Analyze security logs
ansible-playbook playbooks/security/analyze_security_logs.yml -i inventory.yml

# Generate audit reports
./scripts/security/generate_audit_report.sh

# Check for suspicious activity
ansible all -i inventory.yml -m shell -a "grep -i 'suspicious\|attack\|intrusion' /
var/log/syslog"
```

Incident Response

Incident Detection

Automated Detection

```
# Run security monitoring
ansible-playbook playbooks/security/security_monitoring.yml -i inventory.yml

# Check for indicators of compromise
ansible-playbook playbooks/security/check_ioc.yml -i inventory.yml

# Analyze network traffic
ansible-playbook playbooks/security/analyze_network_traffic.yml -i inventory.yml
```

Manual Investigation

```
# Collect forensic data
ansible-playbook playbooks/security/collect_forensics.yml -i inventory.yml \
  --extra-vars "incident_id=INC-2024-001"

# Analyze system artifacts
ansible all -i inventory.yml -m shell -a "find /tmp -type f -mtime -1"

# Check running processes
ansible all -i inventory.yml -m shell -a "ps aux | grep -v grep"
```

Incident Response Procedures

Immediate Response

1. **Isolate Affected Systems**
   ```
   bash
     # Isolate compromised host
     ansible-playbook playbooks/security/isolate_host.yml -i inventory.yml \
       --extra-vars "target_host=compromised-server"
   ```

2. **Preserve Evidence**
   ```
   bash
     # Create forensic image
     ansible-playbook playbooks/security/create_forensic_image.yml -i inventory.yml \
       --extra-vars "target_host=compromised-server"
   ```

3. **Notify Stakeholders**
   ```
   bash
     # Send incident notification
     ansible-playbook playbooks/security/notify_incident.yml -i inventory.yml \
       --extra-vars "incident_severity=high"
   ```

Investigation and Recovery

1. **Analyze Impact**
   ```
   bash
     # Assess damage
     ansible-playbook playbooks/security/assess_damage.yml -i inventory.yml
   ```

2. **Contain Threat**
   ```
   bash
     # Block malicious IPs
     ansible-playbook playbooks/security/block_malicious_ips.yml -i inventory.yml \
       --extra-vars "malicious_ips=['1.2.3.4','5.6.7.8']"
   ```

3. **Eradicate Threat**
   ```
   bash
     # Remove malware
     ansible-playbook playbooks/security/remove_malware.yml -i inventory.yml
   ```

4. **Recover Systems**
   ```
   bash
   ```

```
   # Restore from clean backup
   ansible-playbook playbooks/security/restore_clean_backup.yml -i inventory.yml
```

Post-Incident Activities

Lessons Learned

```
# Generate incident report
./scripts/security/generate_incident_report.sh --incident-id INC-2024-001

# Update security procedures
ansible-playbook playbooks/security/update_security_procedures.yml -i inventory.yml

# Conduct security training
ansible-playbook playbooks/security/security_training.yml -i inventory.yml
```

Security Hardening

System Hardening

Operating System Hardening

```
# Apply OS hardening
ansible-playbook playbooks/security/os_hardening.yml -i inventory.yml

# Configure secure boot
ansible-playbook playbooks/security/configure_secure_boot.yml -i inventory.yml

# Set up intrusion detection
ansible-playbook playbooks/security/setup_ids.yml -i inventory.yml
```

Application Hardening

```
# Harden web servers
ansible-playbook playbooks/security/harden_webserver.yml -i inventory.yml

# Secure database servers
ansible-playbook playbooks/security/secure_database.yml -i inventory.yml

# Configure application security
ansible-playbook playbooks/security/configure_app_security.yml -i inventory.yml
```

Security Monitoring

Continuous Monitoring

```
# Set up security monitoring
ansible-playbook playbooks/security/setup_security_monitoring.yml -i inventory.yml

# Configure alerting
ansible-playbook playbooks/security/configure_security_alerts.yml -i inventory.yml

# Deploy security agents
ansible-playbook playbooks/security/deploy_security_agents.yml -i inventory.yml
```

Threat Intelligence

```
# Update threat intelligence feeds
ansible-playbook playbooks/security/update_threat_intel.yml -i inventory.yml

# Check for known threats
ansible-playbook playbooks/security/check_threat_indicators.yml -i inventory.yml

# Generate threat report
./scripts/security/generate_threat_report.sh
```

Security Training and Awareness

Security Training

```
# Conduct security awareness training
ansible-playbook playbooks/security/security_awareness.yml -i inventory.yml

# Phishing simulation
ansible-playbook playbooks/security/phishing_simulation.yml -i inventory.yml

# Security skills assessment
ansible-playbook playbooks/security/security_assessment.yml -i inventory.yml
```

Documentation and Procedures

- Maintain up-to-date security procedures
- Regular review of security policies
- Document security incidents and lessons learned
- Keep security contact information current

Emergency Contacts

Security Team

- **Security Manager**: [Phone/Email]
- **Incident Response Lead**: [Phone/Slack]
- **Security Analyst**: [Phone/Email]
- **CISO**: [Phone/Email]

External Contacts

- **Law Enforcement**: [Phone]
- **Legal Counsel**: [Phone/Email]
- **Cyber Insurance**: [Phone/Policy Number]
- **Security Vendor**: [Phone/Support]

Communication Channels

- **Security Alerts**: #security-alerts
- **Incident Response**: #incident-response
- **Security Team**: security-team@company.com
- **Emergency Hotline**: [Phone]

Tools and Resources

Security Tools

- **Vulnerability Scanners**: Nessus, OpenVAS, Qualys
- **SIEM**: Splunk, ELK Stack, QRadar
- **IDS/IPS**: Suricata, Snort, OSSEC
- **Forensics**: Volatility, Autopsy, SANS SIFT

Security Resources

- **NIST Cybersecurity Framework**
- **CIS Controls**
- **OWASP Top 10**
- **SANS Security Policies**

Related Documentation

- Deployment Runbook (DEPLOYMENT_RUNBOOK.md)
- Troubleshooting Guide (TROUBLESHOOTING_GUIDE.md)
- Monitoring Guide (MONITORING_GUIDE.md)
- Backup and Recovery (BACKUP_RECOVERY.md)