SOC Incident Response Playbook 1: Ransomware Infection

Scenario

An endpoint or server exhibits signs of ransomware activity such as file encryption, ransom notes or alerts from EDR/XDR tools.

Incident Classification

Category	Details
Incident Type	Malware – Ransomware
Severity	High
Priority	Critical (due to potential business impact and data loss)
Detection Sources	EDR/XDR, SIEM, User Report, Antivirus, NDR

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Backup and recovery strategy	Periodic offline backups, test restoration
Endpoint protection	EDR with behavioural detection and rollback features
User awareness training	Email and USB media handling education
Logging coverage	Windows logs, Sysmon, file access logs, network flows
IOC and threat feed subscriptions	Include ransomware-specific indicators

2. Detection & Analysis

Step	Action
Confirm ransomware activity	EDR alert, presence of ransom note, encrypted file extensions
Isolate affected host	Disconnect from the network or use EDR containment
Identify ransomware strain	Based on ransom note, file hash or filename pattern
Analyse logs and behaviour	Track source of execution, lateral movement, suspicious scheduled tasks or services
MITRE ATT&CK mapping	T1486 (Data Encrypted for Impact), T1059 (Command Execution), T1021.002 (SMB Lateral Movement)

3. Containment

Step	Action
Isolate affected systems	Block at switch, firewall or via EDR
Disable infected accounts	Especially if used for lateral movement
Block external communication	Prevent C2 and key exchange over the internet
Snapshot impacted systems	For forensic analysis (if required)

4. Eradication

Step	Action
Remove malware artifacts	Delete ransomware files, scripts, scheduled tasks
Patch vulnerabilities	Address exploited attack vectors such as RDP, SMB, outdated software
Perform full antivirus/EDR scan	Across all hosts within affected VLAN/subnet
Validate removal	Ensure no persistence mechanisms remain (registry keys, startup items, services)

5. Recovery

Step	Action
Restore from clean backup	Confirm backups are unaffected before restoration
Rebuild systems if needed	For systems without clean backups
Monitor restored systems	Use SIEM and EDR to ensure no reinfection occurs
Reset passwords	Particularly for privileged and affected users

6. Lessons Learned & Reporting

Step	Action
Conduct post-incident review	Analyse root cause, initial access method and response efficiency
Update detection rules	Enhance SIEM and EDR correlation rules and triggers
Document findings	Include indicators, affected systems and timeline
Share IOCs	Internally and with threat intel communities if allowed

Tools Typically Involved

- SIEM (e.g., Splunk, QRadar, Sentinel)
- EDR/XDR (e.g., CrowdStrike, Cortex XDR, SentinelOne)
- Forensics tools (e.g., FTK, Velociraptor, KAPE)
- Network logs (e.g., Zeek, Suricata, NetFlow)
- Backup systems (e.g., Veeam, Rubrik, Commvault)

Success Metrics

Metric	Target
Detection Time	<10 minutes from encryption onset
Isolation Time	<15 minutes after detection
Recovery Time	Depends on backup availability, ideally <24 hours
Containment Scope	No lateral movement outside original VLAN

SOC Incident Response Playbook 2: Insider Data Exfiltration

Scenario

An internal employee, contractor or privileged user attempts to or successfully exfiltrates sensitive data through unauthorised channels such as personal email, cloud storage, removable media or file transfer tools.

Incident Classification

Category	Details
Incident Type	Insider Threat – Data Exfiltration
Severity	High (especially for regulated or confidential data)
Priority	High to Critical
Detection Sources	DLP, SIEM, Proxy logs, CASB, EDR, Email gateway

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Define sensitive data categories	Classify files: PII, financial data, trade secrets
DLP implementation	Set detection policies on endpoints, network, email
Activity monitoring	Log user access, file transfer and cloud app usage
Insider risk training	Educate employees about acceptable data handling
Access control enforcement	Role-based access, least privilege, segmentation

2. Detection & Analysis

Step	Action
Trigger detection alert	DLP violation, abnormal download, large email attachments, unusual file uploads
Analyse access logs	Look for file access, transfer times and destinations
Investigate user behaviour	Check for privilege escalation, login time anomalies, failed access attempts
Confirm intent or misconfiguration	Determine if action was malicious, accidental or a policy gap
MITRE ATT&CK mapping	T1020 (Automated Exfiltration), T1048 (Exfiltration over Alternative Protocol), T1537 (Transfer Data to Cloud Account)

3. Containment

Step	Action

Suspend user access	Temporarily disable account if risk is high
Block exfiltration channels	Revoke cloud sharing, block email to external domains, disable USB ports
Isolate endpoints	If malicious software is suspected on the user device
Preserve forensic evidence	Do not shut down systems unless necessary; capture volatile data if possible

4. Eradication

Step	Action
Remove unauthorised tools	E.g., personal file transfer apps, rogue extensions
Apply stricter policies	Adjust DLP rules or firewall rules to block repeat attempts
Correct misconfigured permissions	Reduce overexposed data shares, folder-level access

5. Recovery

Step	Action
Restore access (if justified)	After confirming no ongoing threat
Notify stakeholders	Legal, HR, compliance and management teams
Conduct impact assessment	Confirm if data was actually exfiltrated and its sensitivity

6. Lessons Learned & Reporting

Step	Action
Document the incident	Timeline, data types, actor intent, system used
Strengthen monitoring	Improve alerting on specific file types and transfer methods
Conduct user training or disciplinary action	If incident is confirmed malicious or negligent
Report to regulators	If required by law (e.g., PDPA, GDPR, HIPAA)
Update insider threat policy	Incorporate new insights into security procedures

Tools Typically Involved

- SIEM (e.g., Splunk, IBM QRadar, Microsoft Sentinel)
- DLP systems (e.g., Symantec, Forcepoint, Microsoft Purview)
- CASB (e.g., Netskope, Microsoft Defender for Cloud Apps)
- Endpoint agents (e.g., EDR with data transfer monitoring)
- Proxy & firewall logs
- Email Security Gateway (e.g., Proofpoint, Mimecast)

Success Metrics

Metric	Target
Detection Time	<10 minutes from data transfer
Investigation Time	<1 hour from alert
Containment Time	<30 minutes
Regulatory Response Time	Within required legal timeframe (e.g., 72 hours)

SOC Incident Response Playbook 3: Cloud Account Compromise

Scenario

An attacker gains unauthorized access to a user's cloud account, possibly through phishing, password spraying, token theft or OAuth abuse. The attacker may access email, storage, admin functions or cloud infrastructure.

Incident Classification

Category	Details
Incident Type	Identity Compromise – Cloud Account
Severity	High (especially if privileged account is involved)
Priority	Critical if lateral movement or data access is observed
Detection Sources	SIEM, CASB, Cloud-native logging (e.g., AWS CloudTrail, Azure AD), Email gateway, EDR

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enable cloud logging	Use AWS CloudTrail, Azure Sign-in logs, Google Workspace audit logs
Implement MFA	Enforce for all users, especially privileged accounts
Monitor user behaviour	Integrate cloud logs into SIEM, use anomaly detection
Set geo-restrictions and login alerts	Alert on impossible travel or first-time access from unknown IPs
Apply least privilege	Use RBAC policies and regular permission audits

2. Detection & Analysis

Step	Action
Detect login anomalies	Failed logins, impossible travel, MFA bypass alerts
Correlate with threat intel	Match IPs, user agents or domains with IOC feeds
Check access logs	Review mailbox, storage, IAM or API activity after compromise
Look for privilege escalation	Identify if the attacker attempted to gain more access or created backdoor accounts

MITRE ATT&CK mapping	T1078 (Valid Accounts), T1087.004 (Cloud Account Discovery), T1556.004 (Forge Web Credentials), T1531 (Account Access Removal)

3. Containment

Step	Action
Revoke sessions and tokens	Invalidate all active sessions, OAuth tokens and refresh tokens
Reset password	Enforce strong password and enable MFA if not already enabled
Suspend account	If compromise is confirmed and impact is high
Block IP addresses	If attacker used known bad IPs or TOR exit nodes

4. Eradication

Step	Action
Remove malicious inbox rules or automation	Clean auto-forward rules, inbox filters and calendar sharing changes
Disable rogue applications	Revoke consent for unauthorised third-party apps
Review admin roles	Revert unauthorized admin access or privilege changes
Restore modified data	If integrity issues occurred (e.g., mailbox deletion, S3 file replacement)

5. Recovery

Step	Action
Re-enable account access	After ensuring full control is restored and no persistence remains
Notify affected users or stakeholders	Especially if business email compromise (BEC) occurred
Monitor for post-recovery login anomalies	Use SIEM or CASB to detect reuse attempts or related attacks
Update access policies	Refine conditional access, session timeout and MFA enforcement rules

6. Lessons Learned & Reporting

Step	Action
Conduct root cause analysis	Phishing, weak password, token theft, misconfiguration
Update playbooks and detection rules	Add improved indicators and logic to SIEM or CASB rules

Educate users	Reinforce training on phishing and cloud security practices
Document incident report	Include timeline, method of access, affected resources and actions taken
Fulfill legal reporting obligations	If applicable (e.g., PDPA, GDPR, customer SLAs)

Tools Typically Involved

- SIEM (e.g., Microsoft Sentinel, Splunk, QRadar)
- Cloud-native logs (e.g., AWS CloudTrail, Azure Log Analytics, Google Workspace audit logs)
- CASB (e.g., Netskope, Microsoft Defender for Cloud Apps)
- Cloud Security Posture Management (e.g., Wiz, Prisma Cloud)
- EDR/XDR with identity correlation (e.g., CrowdStrike, Cortex XDR)

Success Metrics

Metric	Target
Detection Time	<15 minutes from suspicious login
Response Time	<1 hour to lock and reset credentials
Containment Time	<30 minutes after confirmation
Post-incident Monitoring Period	7–14 days minimum

SOC Incident Response Playbook 4: Web Application Exploitation

Scenario

An attacker exploits a vulnerability in a web application or server to gain unauthorised access, execute commands or extract sensitive data. The attack may be detected via WAF alerts, SIEM logs or anomalous behaviour.

Incident Classification

Category	Details
Incident Type	Application-layer Attack
Severity	High to Critical (depends on data exposure or lateral movement)
Priority	High
Detection Sources	Web Application Firewall (WAF), SIEM, IDS/IPS, Web server logs, Cloud monitoring tools

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Conduct regular vulnerability assessments	Use tools like Nexpose, Tenable or Burp Suite
Implement a WAF	Configure OWASP top 10 rule sets (e.g., ModSecurity, Cloudflare, AWS WAF)
Log HTTP traffic	Ensure proper logging from web servers, app servers and proxies
Patch management	Automate patch cycles for web frameworks, plugins and platforms
Code review & DevSecOps integration	Integrate SAST/DAST tools into CI/CD pipeline

2. Detection & Analysis

Step	Action
Alert from WAF or SIEM	SQL injection, RCE, XSS, LFI/RFI attempts
Review logs	Analyse HTTP requests, server responses, unusual error codes (e.g., 500, 403)
Validate input payloads	Confirm attack vector via payload (e.g., ' OR 1=1--, <?php system($_GET[cmd]) ?>)

Check for compromise	Look for shell uploads, privilege escalations, abnormal process execution
MITRE ATT&CK mapping	T1190 (Exploit Public-Facing Application), T1059 (Command Execution), T1505 (Server Software Component)

3. Containment

Step	Action
Block attacker IPs	Use WAF, firewall or reverse proxy to block source IP
Disable affected web functions	Temporarily shut down vulnerable modules or APIs
Isolate the application server	Disconnect from internal network if lateral movement is suspected
Revoke session tokens	If user sessions or cookies are believed to be hijacked

4. Eradication

Step	Action
Remove malicious scripts or shells	Search for web shells, reverse shell listeners or backdoors
Patch exploited vulnerability	Update code, platform, plugin or misconfiguration
Harden application	Implement input validation, sanitisation, parameterised queries
Scan entire application stack	Revalidate with updated vulnerability scanner to confirm fix

5. Recovery

Step	Action
Restore services	Bring application back online after confirming clean state
Monitor post-restoration	Closely observe logs for repeat attempts or backdoor access
Notify affected users or customers	If data breach occurred, comply with disclosure requirements
Conduct retest	Confirm no residual access or re-exploitation risk exists

6. Lessons Learned & Reporting

Step	Action
Perform root cause analysis	Identify coding flaw, misconfiguration or patch delay

Update SIEM and WAF rules	Add custom detections based on observed exploit vectors
Improve secure coding practices	Conduct refresher training for developers on OWASP Top 10
Document incident timeline	Include detection time, TTPs, impact and mitigation steps
Report as required	If personal data was affected, report to regulators or customers

Tools Typically Involved

- WAF (e.g., ModSecurity, AWS WAF, Cloudflare)
- SIEM (e.g., Splunk, Sentinel, QRadar)
- Web server logs (e.g., Apache, Nginx)
- Vulnerability scanners (e.g., Nessus, Qualys, Nikto)
- EDR/XDR (if lateral movement occurred)
- Forensics tools (if shell or system compromise suspected)

Success Metrics

Metric	Target
Detection Time	<5 minutes from WAF/SIEM alert
Containment Time	<30 minutes from confirmation
Vulnerability Patch Time	<24 hours (critical) or <7 days (high)
Post-Incident Retest Time	Within 48 hours after recovery

SOC Incident Response Playbook 5: Supply Chain Attack

Scenario

An organisation is compromised through a trusted third-party service, software update, library, plugin or IT service provider. The attacker uses the trusted relationship to move laterally, deploy malware or exfiltrate data.

Incident Classification

Category	Details
Incident Type	Supply Chain Compromise
Severity	Critical (due to indirect trust exploitation)
Priority	Critical
Detection Sources	Threat intelligence, SIEM, EDR, vulnerability reports, system anomalies

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Maintain third-party inventory	List all vendors, software providers and integrations
Conduct risk assessments	Evaluate criticality and access level of each vendor or dependency
Apply access restrictions	Use segmentation and least privilege for third-party services
Monitor software behaviour	Enable logging and behavioural analysis for all installed components
Validate software updates	Use secure channels and signed binaries for critical applications

2. Detection & Analysis

Step	Action
Identify abnormal behaviour	Outbound connections, registry changes, dropped files, execution from unexpected paths
Verify against threat intelligence	Check IoCs related to known supply chain breaches (e.g., SolarWinds, MOVEit, Kaseya)
Examine affected components	Determine if recent updates or third-party access triggered the behaviour

Review vendor communications	Check for public disclosures or breach notifications
MITRE ATT&CK mapping	T1195.002 (Compromise Software Dependencies), T1195.001 (Compromise Software Supply Chain), T1105 (Ingress Tool Transfer)

3. Containment

Step	Action
Disconnect affected systems	Prevent lateral movement and external communication
Suspend integrations or services	Disable connections to affected vendor software, APIs or modules
Block malicious binaries or signatures	Use EDR/XDR to prevent execution of known malicious components
Quarantine suspicious hosts	Isolate endpoints communicating with attacker infrastructure

4. Eradication

Step	Action
Remove malicious files or updates	Uninstall or roll back infected or trojanised software
Validate software integrity	Use hash comparison or vendor-signed binaries
Remove backdoors or persistence	Clean registry keys, scheduled tasks, rogue accounts or remote access tools
Update detection rules	Add new IoCs to SIEM and EDR platforms for early detection of reoccurrence

5. Recovery

Step	Action
Reinstall from clean source	Use validated installation media or updated software versions
Restore from backup	Only if backup is verified to be unaffected
Re-establish vendor connection	After patching or validation by third-party provider
Resume normal operations	After containment and eradication are fully verified

6. Lessons Learned & Reporting

Step	Action

Conduct a post-incident review	Determine timeline, attack path, vendor involvement
Update third-party risk program	Introduce more stringent onboarding, auditing and segmentation rules
Inform stakeholders	Notify management, legal and affected business units
Collaborate with the vendor	Share findings and request full disclosure on their mitigation status
Report if required	Regulatory and contractual obligations (e.g., PDPA, GDPR, customer SLAs)

Tools Typically Involved

- SIEM (e.g., Splunk, Sentinel, QRadar)
- EDR/XDR (e.g., CrowdStrike, Cortex XDR)
- Threat intelligence platforms (e.g., MISP, Recorded Future)
- Software integrity validation (e.g., sigcheck, file hashing tools)
- Configuration management tools (e.g., SCCM, Ansible, JAMF)

Success Metrics

Metric	Target
Vendor Notification Response Time	Within 24 hours of known vendor disclosure
Compromise Detection Time	<6 hours after initial signs
Isolation & Containment Time	<2 hours after confirmation
Remediation Completion Time	Within 48–72 hours for critical systems
Third-Party Reassessment Completion	Within 7 days of incident closure

SOC Incident Response Playbook 6: Malware via USB Device

Scenario

Malicious software is introduced into the environment through an infected USB storage device. This may include autorun malware, ransomware, keyloggers or tools used to establish persistence or exfiltrate data.

Incident Classification

Category	Details
Incident Type	Physical Media-Based Malware Infection
Severity	Medium to High (depending on malware type and spread)
Priority	High if lateral movement or sensitive data is involved
Detection Sources	EDR, antivirus/antimalware, SIEM, user report, USB monitoring tools

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Disable USB autorun	Group Policy settings or endpoint hardening
Implement USB control software	Allow only authorised devices; log USB insertions
Enforce endpoint protection	EDR with removable media protection and behavioural detection
Educate users	Train staff not to plug in unknown USB drives
Log USB usage	Enable audit policies for removable media activity

2. Detection & Analysis

Step	Action
Detect malware activity	Alert from antivirus, EDR or SIEM on process execution from USB
Identify USB event	Review logs for usb-storage, DevicePlugEvent or Removable Storage Device events
Analyse file origin	Determine whether execution started from USB drive (e.g., drive D:\ or E:)
Collect indicators	Hashes, filenames, execution chain, affected systems
MITRE ATT&CK mapping	T1200 (Hardware Additions), T1091 (Replication Through Removable Media), T1059 (Command and Scripting Interpreter)

3. Containment

Step	Action
Isolate infected system	Disconnect network and USB ports to prevent spread
Remove USB device	Preserve for forensic investigation if necessary
Block malicious file hashes	In EDR or AV systems across the organisation
Identify other exposed systems	Scan for similar infections or shared lateral movement paths

4. Eradication

Step	Action
Remove malware	Use EDR or AV tools to clean the infected files and processes
Delete suspicious files	From temporary folders, startup directories or root of USB drive
Remove persistence mechanisms	Check registry run keys, scheduled tasks, services
Perform full malware scan	On the infected host and nearby systems

5. Recovery

Step	Action
Restore system	From clean backup if necessary
Reinstate connectivity	After confirming the host is clean
Enable stricter USB policies	Allow only whitelisted devices or disable USB entirely in high-risk environments
Document the root cause	Device origin, user involved, type of malware, system impact

6. Lessons Learned & Reporting

Step	Action
Conduct awareness training	Reinforce security policy on device usage
Update USB policy	Improve endpoint controls and documentation procedures
Share findings with security team	Review detection gaps, response time and behavioural indicators
Document incident report	Include all actions taken, findings and recommendations

Tools Typically Involved

- Endpoint Detection and Response (e.g., CrowdStrike, Cortex XDR)

- USB control solutions (e.g., DeviceLock, Endpoint Protector, Microsoft Intune policies)
- Antivirus software (e.g., Windows Defender, Bitdefender, Kaspersky)
- SIEM for USB and file execution logging
- Windows Event Logs (Event ID 2003, 2102 for device insertion)

Success Metrics

Metric	Target
Detection Time	<10 minutes after USB malware execution
Isolation Time	<15 minutes after confirmation
Malware Removal Time	<1 hour (if no system rebuild required)
USB Policy Enforcement	100% of endpoints have policy applied
User Awareness Rate	≥ 90% of users aware of USB risks post-training

SOC Incident Response Playbook 7: DDoS Attack

Scenario

An external attacker launches a distributed denial-of-service (DDoS) attack targeting public-facing infrastructure such as websites, APIs, DNS servers or network gateways. The objective is to disrupt service availability, degrade performance or cause reputational and financial damage.

Incident Classification

Category	Details
Incident Type	Network/Application Layer Availability Attack
Severity	High (especially for customer-facing or critical systems)
Priority	Critical if sustained outage or service degradation occurs
Detection Sources	NOC alerts, SIEM, firewall logs, application monitoring tools, CDN/WAF, ISP notifications

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Implement DDoS protection	Use cloud-based mitigation services (e.g., Cloudflare, AWS Shield, Akamai)
Deploy WAF and rate limiting	Protect applications and APIs
Ensure scalable infrastructure	Use autoscaling groups or CDN caching to absorb surges
Establish communication with ISP	Predefine escalation process and mitigation support
Conduct DDoS drills	Simulate DDoS scenarios and validate response procedures

2. Detection & Analysis

Step	Action
Identify traffic surge	Monitor bandwidth, request rates or connection counts exceeding normal thresholds
Determine attack vector	Is it volumetric (UDP flood), protocol (SYN flood) or application-layer (HTTP GET flood)?
Correlate with logs	Identify source IPs, user agents, referrers, payloads

Confirm impact	Assess performance degradation, service outages or collateral damage
MITRE ATT&CK mapping	T1498 (Network Denial of Service), T1499 (Endpoint Denial of Service), T1498.001 (Direct Network Flood)

3. Containment

Step	Action
Engage cloud DDoS mitigation service	Route traffic through mitigation provider (e.g., Cloudflare Magic Transit)
Block malicious IPs	Using firewall, WAF or geo-blocking rules
Implement rate limiting and filters	Drop traffic by rate thresholds or specific patterns
Redirect or reroute traffic	Temporarily divert traffic to alternate IP or load balancer

4. Eradication

Step	Action
Drop traffic from confirmed malicious sources	Based on IP reputation or behavioural patterns
Adjust filtering rules	Fine-tune ACLs, WAF policies, IDS/IPS signatures
Remove temporary rules post-attack	Once attack subsides, restore normal access patterns
Investigate for blended threats	Confirm no malware or lateral movement occurred during the disruption

5. Recovery

Step	Action
Monitor for residual traffic	Use NOC dashboards and SIEM to track post-attack anomalies
Confirm service restoration	Perform user acceptance testing or API health checks
Notify affected customers or partners	If SLAs or public services were impacted
Resume normal routing	If temporary redirection or black-holing was used during attack

6. Lessons Learned & Reporting

Step	Action
Conduct incident review	Document timeline, impact, attacker strategy and response actions

Assess mitigation effectiveness	Determine what worked and what needs to be improved
Update response playbook	Refine thresholds, alerting rules and communication steps
Improve vendor coordination	Review performance of ISP and mitigation partners
Report as required	To regulators, leadership or clients if impact was severe or prolonged

Tools Typically Involved

- SIEM (e.g., Splunk, Sentinel, QRadar)
- Network traffic analysis tools (e.g., NetFlow, Zabbix, Ixia)
- DDoS protection services (e.g., Cloudflare, AWS Shield, Akamai Kona, Arbor)
- Firewall/WAF (e.g., Fortinet, Palo Alto, ModSecurity)
- CDN and DNS services (e.g., Cloudflare, Fastly, Akamai)
- ISP support and coordination channels

Success Metrics

Metric	Target
Time to Detect DDoS	<5 minutes from onset
Mitigation Engagement Time	<15 minutes from confirmation
Service Downtime	Zero or <30 minutes
Customer Notification Time	Within 1 hour if SLA is affected
Post-Mortem Completion	Within 48 hours

SOC Incident Response Playbook 8: Business Email Compromise (BEC)

Scenario

An attacker gains access to or spoofs a legitimate business email account to deceive internal staff, customers or partners into making unauthorised wire transfers, sharing credentials or altering financial records. This may involve phishing, credential theft or abuse of trusted relationships.

Incident Classification

Category	Details
Incident Type	Social Engineering / Identity Compromise
Severity	High to Critical (due to financial and reputational risk)
Priority	Critical
Detection Sources	Email gateway, SIEM, EDR, user report, cloud email audit logs

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enforce multi-factor authentication (MFA)	For all business email accounts, especially executives
Monitor mailbox activity	Enable cloud audit logs for Microsoft 365, Google Workspace
Configure email filtering	Block spoofed domains, implement SPF, DKIM, DMARC
Conduct anti-phishing training	Frequent phishing simulations and awareness sessions
Define financial control processes	Multi-person verification for wire transfers or invoice changes

2. Detection & Analysis

Step	Action
Identify suspicious activity	Unusual login location, new inbox rules, unexpected email content
Analyse email headers and metadata	Verify sending domain, IP reputation, reply-to address
Review mailbox rules and access logs	Look for auto-forwarding, deletion filters and unauthorised logins

Check for financial or HR engagement	Determine if attacker contacted internal or external finance/HR personnel
MITRE ATT&CK mapping	T1078 (Valid Accounts), T1114 (Email Collection), T1204 (User Execution), T1585.002 (Spoofing - Email Account)

3. Containment

Step	Action
Revoke access	Reset passwords and invalidate sessions/tokens for affected accounts
Disable inbox rules	Remove any malicious forwarding or deletion filters
Alert potentially impacted users	Notify those who received fake requests or were impersonated
Block attacker IPs	In email platform or at the firewall level if recurring

4. Eradication

Step	Action
Fully audit affected account	Review login history, email sent, calendar changes, contact manipulation
Remove malicious artefacts	Delete fake emails, remove rogue permissions or shared inbox access
Re-secure account	Enforce strong password policy and enable conditional access if supported
Conduct forensics (if needed)	Export logs and preserve evidence for legal or financial investigations

5. Recovery

Step	Action
Re-enable account access	After confirming no ongoing risk
Restore legitimate mail flow	Clear out auto-forwarding and ensure delivery settings are correct
Notify stakeholders	Inform internal teams, vendors or clients involved in the incident
Monitor for repeat activity	Set alerts for high-risk account behaviours for 14–30 days

6. Lessons Learned & Reporting

Step	Action
Perform root cause analysis	Identify how the compromise occurred (phishing, weak password, no MFA)

Report financial impact	Notify finance, risk and legal teams
Engage law enforcement or insurance	If fraud occurred or required by policy
Update playbook and alerts	Enhance detection rules for email forwarding, IP anomalies, login velocity
Train employees on social engineering tactics	Focus on finance, procurement and executive staff awareness

Tools Typically Involved

- SIEM (e.g., Sentinel, Splunk, QRadar)
- Email security gateways (e.g., Proofpoint, Mimecast, Microsoft Defender for Office 365)
- Cloud audit logs (Microsoft 365 Unified Audit Log, Google Workspace Admin Console)
- Identity platforms (e.g., Okta, Azure AD, Duo)
- Threat intel feeds for spoofed domain detection and email TTPs

Success Metrics

Metric	Target
Detection Time	<15 minutes from phishing or suspicious email activity
Containment Time	<1 hour after confirmation
Financial Fraud Prevention	Stop wire transfer or mitigate within 24 hours
Awareness Campaign Completion	100% of high-risk employees trained post-incident
Post-Incident Monitoring Period	Minimum of 30 days for affected accounts

SOC Incident Response Playbook 9: Unauthorised Privilege Escalation

Scenario

An attacker, either through a vulnerability, misconfiguration or stolen credentials, escalates privileges from a low-privilege user to an administrative or root-level account, potentially compromising critical systems or accessing sensitive data.

Incident Classification

Category	Details
Incident Type	Access Control Violation / Privilege Misuse
Severity	High to Critical
Priority	Critical
Detection Sources	SIEM, EDR, IAM logs, Sysmon, User Behaviour Analytics (UBA), Audit trails

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Implement RBAC and least privilege	Ensure users only have necessary access
Monitor privileged account activity	Set up alerting for group membership changes and privilege elevation
Log privilege escalation attempts	Enable audit logs in Windows (Event ID 4670, 4672, 4728) and Linux (sudo logs, auditd)
Conduct regular entitlement reviews	Periodic reviews of admin rights and group memberships
Harden endpoints	Patch privilege escalation vulnerabilities and monitor for exploit attempts

2. Detection & Analysis

Step	Action
Identify privilege escalation alerts	Unusual admin access, group changes or privilege tokens
Correlate with user behaviour	Check if the user normally has admin rights or elevated actions are expected
Analyse process tree	Look for unusual parent-child relationships (e.g., cmd.exe from Outlook)

Validate persistence techniques	Registry changes, scheduled tasks, services creation with elevated rights
MITRE ATT&CK mapping	T1068 (Exploitation for Privilege Escalation), T1548 (Abuse Elevation Control Mechanism), T1078 (Valid Accounts)

3. Containment

Step	Action
Disable affected user accounts	If elevation was unauthorised or compromised
Terminate elevated sessions or processes	Kill suspicious PowerShell, cmd or service processes
Block IP or device	If part of lateral movement or known attacker infrastructure
Notify IT or HR	If the user is internal and intent is unclear (malicious vs mistake)

4. Eradication

Step	Action
Revert permission changes	Remove elevated rights, group memberships or access tokens
Clean persistence mechanisms	Remove scheduled tasks, registry modifications, service entries created by the attacker
Patch exploited vulnerabilities	Apply fixes for kernel-level or OS-level flaws (e.g., CVE-2021-36934)
Review IAM policies and GPOs	Address any inherited misconfigurations or unintended permission inheritance

5. Recovery

Step	Action
Re-enable legitimate users	With correct access rights after review
Restore affected systems	If any configuration or data was altered during escalation
Resume operations	Once verified clean and secure
Conduct post-remediation scan	Confirm no backdoors or elevation paths remain

6. Lessons Learned & Reporting

Step	Action
Document full escalation path	How the privilege was gained and what was accessed or modified

Update SIEM detection rules	For abnormal privilege changes, sensitive command execution
Improve identity governance	Enforce stricter access request and approval workflows
Report if required	Especially if data was accessed or tampered with
Educate privileged users	On the importance of proper access hygiene and security controls

Tools Typically Involved

- SIEM (e.g., Splunk, Sentinel, QRadar)
- EDR (e.g., CrowdStrike, Cortex XDR, Microsoft Defender for Endpoint)
- IAM platforms (e.g., Azure AD, Okta, LDAP, Active Directory)
- Windows Event Logs (Security logs, Sysmon, GPO auditing)
- Linux audit tools (auditd, sudo logs)
- Threat Detection Rules (Sigma, KQL, YARA for suspicious privilege activity)

Success Metrics

Metric	Target
Detection Time	<5 minutes from escalation event
Containment Time	<30 minutes from confirmation
Reversion Time	<1 hour to remove elevated access
Audit & RCA Completion	Within 48 hours
Privileged Account Review Completion	100% within 7 days post-incident

SOC Incident Response Playbook 10: Cloud Storage Misconfiguration Exposure

Scenario

Sensitive or confidential data (e.g., logs, databases, personal information) is exposed to the public due to misconfigured permissions on cloud storage services, often discovered via threat intelligence feeds, automated scanners or internal audits.

Incident Classification

Category	Details
Incident Type	Data Exposure – Misconfiguration
Severity	High to Critical (depends on sensitivity of data)
Priority	High
Detection Sources	Cloud Security Posture Management (CSPM), SIEM, Threat Intel, External Notification (e.g., researcher, media), Audit Logs

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enforce default secure policies	Block public access at the organisation level for cloud storage services
Implement CSPM tools	Continuously monitor for misconfigurations (e.g., Wiz, Prisma Cloud, AWS Config)
Enable access logging	For cloud storage services (e.g., AWS S3 access logs, Azure diagnostics)
Tag and classify sensitive data	Use data classification tools to mark high-risk information
Perform regular cloud audits	Review access settings for storage buckets, blobs and containers

2. Detection & Analysis

Step	Action
Receive alert from CSPM or threat intel	Example: "Public read access detected on S3 bucket storing backup files"
Review object permissions	Determine which files are exposed and who can access them (public, anonymous, specific users)
Assess data sensitivity	Identify types of exposed data (e.g., PII, financial, passwords, API keys)

Check access logs	Identify if any unauthorised access has occurred (IP addresses, timestamps)
MITRE ATT&CK mapping	T1530 (Data from Cloud Storage Object), T1562.007 (Disable or Modify Cloud Storage Logging)

3. Containment

Step	Action
Restrict public access immediately	Remove 'public-read', 'allUsers' or 'anonymous' permissions from bucket or object
Disable sharing links	Revoke signed URLs or public object URLs
Notify affected teams	Alert data owners, compliance and security teams for risk assessment
Quarantine compromised credentials	If API keys or credentials were exposed, rotate immediately

4. Eradication

Step	Action
Review and fix IAM policies	Audit and adjust overly permissive roles or storage policies
Enable bucket/block-level protection	Enforce default encryption, versioning and public access blocking
Clean exposed data	Remove or archive unnecessary files, scrub exposed content
Reconfigure secure sharing mechanisms	Use identity-based access controls instead of public sharing links

5. Recovery

Step	Action
Validate proper access controls	Confirm access is restricted to intended users and services
Confirm data integrity	Ensure no tampering or unauthorised modifications occurred
Resume operations	Restore use of cloud storage once properly secured
Update inventory	Reflect current access control status in asset and data tracking systems

6. Lessons Learned & Reporting

Step	Action

Conduct root cause analysis	Identify whether exposure was due to human error, policy failure or automation
Update CSPM and SIEM detections	Tune alerts for permission drift and external access attempts
Train developers and DevOps teams	Reinforce secure configuration practices in CI/CD pipeline
Report if required	Notify regulators or customers if PII or confidential data was exposed
Document lessons learned	Update cloud governance policies and incident playbooks accordingly

Tools Typically Involved

- CSPM tools (e.g., Wiz, Prisma Cloud orca, AWS Config, Microsoft Defender for Cloud)
- SIEM (e.g., Sentinel, Splunk)
- Cloud audit logs (e.g., AWS CloudTrail, Azure Activity Logs, GCP Admin Activity)
- IAM systems (e.g., AWS IAM, Azure AD, Google IAM)
- DLP or classification systems (e.g., Microsoft Purview, Symantec DLP)

Success Metrics

Metric	Target
Detection Time	<1 hour from exposure
Access Removal Time	<30 minutes from alert
Public Exposure Duration	Ideally <1 hour
Impact Assessment Completion	Within 24–48 hours
Policy Remediation Time	Within 72 hours

SOC Incident Response Playbook 11: Credential Stuffing Attack

Scenario

An attacker uses automated tools and botnets to test large volumes of stolen credentials (typically from dark web breaches) against a login portal in hopes of reusing valid username-password combinations. This can lead to unauthorised access to user accounts and potential data theft or fraud.

Incident Classification

Category	Details
Incident Type	Account Takeover via Credential Abuse
Severity	High (especially in financial, SaaS or personal data services)
Priority	High
Detection Sources	SIEM, IAM logs, WAF, fraud detection systems, application logs, CDN security layers (e.g., Cloudflare, Akamai)

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enforce MFA	Strongest defence against credential reuse
Rate limit login attempts	Use WAF/CDN or application-level throttling
Monitor for credential stuffing patterns	Spike in failed logins, login attempts from multiple geographies
Use CAPTCHA or bot protection	Block automated tools
Subscribe to credential breach feeds	Integrate with HaveIBeenPwned, SpyCloud or similar sources

2. Detection & Analysis

Step	Action
Identify unusual login activity	Multiple failed attempts across many usernames from same IP
Review IAM and app logs	Track login frequency, IPs, device fingerprints, user agents
Check for bot behaviour	Impossible travel, excessive logins within a time window, sequential patterns
Validate account takeovers	Determine if login succeeded with breached credentials

MITRE ATT&CK mapping	T1110.001 (Brute Force - Password Guessing), T1078 (Valid Accounts), T1589.001 (Credentials: Usernames), T1589.002 (Passwords)

3. Containment

Step	Action
Block IPs or IP ranges	Use WAF, CDN or firewall to block offending sources
Trigger forced password resets	For impacted users whose credentials were reused
Throttle traffic	Apply tighter rate limits or geo-blocking rules temporarily
Suspend affected sessions	Invalidate active sessions and tokens for suspected accounts

4. Eradication

Step	Action
Remove test accounts or injected data	If attacker created new users or added persistent artefacts
Patch login abuse vectors	Harden login flow, disable username enumeration, limit error messaging
Enforce stronger passwords	Update password policies if weak credentials are in use
Enhance detection rules	Fine-tune alerting thresholds and response automation for credential stuffing attempts

5. Recovery

Step	Action
Notify users	Alert affected users about forced resets and possible compromise
Monitor for repeated attempts	Continue enhanced monitoring for 24–72 hours
Re-enable access	Once accounts are secured with MFA and/or new credentials
Review and test controls	Ensure rate limiting, MFA enforcement and logging mechanisms are effective

6. Lessons Learned & Reporting

Step	Action
Conduct root cause analysis	Was a specific API, endpoint or weak control abused?
Document affected users and accounts	Tally successful logins from malicious sources

Report to regulators	If account takeover results in breach of personal or financial data
Update security controls	Apply geo-fencing, browser fingerprinting, CAPTCHA and bot mitigation tools
Improve user communication	Provide guidance on password hygiene and breach alert follow-ups

Tools Typically Involved

- WAF/CDN (e.g., Cloudflare, Akamai, AWS WAF)
- IAM logs and systems (e.g., Azure AD, Okta, AWS Cognito)
- SIEM (e.g., Sentinel, Splunk)
- Bot detection services (e.g., reCAPTCHA, PerimeterX, Cloudflare Bot Management)
- Breach monitoring platforms (e.g., SpyCloud, HaveIBeenPwned)
- Threat intelligence platforms (e.g., Recorded Future, MISP)

Success Metrics

Metric	Target
Detection Time	<5 minutes from surge in login attempts
Containment Time	<30 minutes from attack confirmation
User Impact Mitigation Time	<2 hours for forced resets and notifications
Recurrence Rate	Zero re-use after controls applied
Post-Attack Monitoring Period	Minimum 7–14 days for affected systems or portals

SOC Incident Response Playbook 12: Unauthorised Internal Database Access

Scenario

An insider or compromised system accesses database resources in an unauthorised manner, such as bypassing access controls, querying sensitive tables or using privileged database accounts inappropriately. This may include data snooping, unauthorised exports or lateral movement toward database servers.

Incident Classification

Category	Details
Incident Type	Access Control Violation – Data Access Abuse
Severity	High (especially if PII, financial data or intellectual property is involved)
Priority	Critical
Detection Sources	SIEM, Database Activity Monitoring (DAM), User Behaviour Analytics (UBA), EDR, Application Logs

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Implement database activity monitoring	Use tools like Imperva DAM, IBM Guardium or native audit logs
Restrict access using least privilege	Use role-based access and limit direct DB access
Enable logging and alerts	Log all privileged queries, schema access and authentication events
Regularly review database roles and privileges	Audit permissions for all database users and service accounts
Encrypt sensitive data	Protect high-value fields (e.g., PII, passwords) at rest and in transit

2. Detection & Analysis

Step	Action
Receive alert from DAM or SIEM	Unusual query volumes, direct table scans or after-hours access
Correlate with user behaviour	Review user's historical database access patterns
Examine queries or transactions	Determine what data was accessed, modified or exported

Check for lateral movement	See if access followed endpoint or network compromise
MITRE ATT&CK mapping	T1071.001 (Exfiltration Over Web Protocol), T1213.003 (Access Sensitive Data in Databases), T1078 (Valid Accounts)

3. Containment

Step	Action
Revoke database access	Disable or suspend the offending account or connection
Isolate compromised endpoint	If access came from a breached host
Block outbound data transfer	Via DLP, firewall or proxy if exfiltration is suspected
Notify data owners and IT security	Involve stakeholders for immediate containment decisions

4. Eradication

Step	Action
Reset credentials or tokens	For database accounts that were abused
Remove rogue users or permissions	Audit database for hidden users, triggers or escalated privileges
Patch vulnerabilities	If a flaw in application or database was exploited
Clean up logs	Archive and secure logs for forensic investigation before removal or trimming

5. Recovery

Step	Action
Restore legitimate access	After proper revalidation of user roles
Monitor for repeat access	Apply enhanced monitoring for the same user or host
Perform integrity check	Validate that no data was altered or corrupted during the incident
Resume services	Resume application or database operations once secure and validated

6. Lessons Learned & Reporting

Step	Action
Conduct post-incident analysis	Understand whether this was malicious, accidental or systemic

Enhance database monitoring rules	Add new patterns and alert conditions
Train users and DB admins	Reinforce data access policies and logging expectations
Report data exposure	If required by law or policy (e.g., PDPA, GDPR, PCI DSS)
Update runbooks	Include playbook refinements and lessons learned in SOC documentation

Tools Typically Involved

- SIEM (e.g., Splunk, Sentinel, QRadar)
- Database Activity Monitoring (e.g., Imperva, IBM Guardium, AWS RDS Logs)
- User Behaviour Analytics (e.g., Exabeam, Securonix)
- EDR (if endpoint is involved)
- Application logs (e.g., from middleware or APIs calling the database)
- DLP and network proxy (to detect potential exfiltration)

Success Metrics

Metric	Target
Detection Time	<5 minutes from unauthorised access
Containment Time	<30 minutes from confirmation
Forensic Review Completion	Within 48 hours
Role/Permission Audit Completion	Within 7 days
Policy Update and Revalidation	Within 2 weeks

SOC Incident Response Playbook 13: Shadow IT Asset Discovery

Scenario

A previously unknown or unauthorised IT asset (e.g., cloud service, SaaS application, personal laptop, rogue Wi-Fi access point or unapproved web app) is discovered operating within or connected to the corporate environment, potentially bypassing security controls and increasing risk exposure.

Incident Classification

Category	Details
Incident Type	Asset Management / Policy Violation
Severity	Medium to High (based on data accessed or exposed)
Priority	High if linked to sensitive systems or users
Detection Sources	CASB, EDR, SIEM, Asset Discovery Tools, Proxy Logs, DNS Logs, Employee Tip-Offs

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Maintain up-to-date asset inventory	Use CMDB or automated asset discovery tools
Deploy CASB and endpoint telemetry	Detect unapproved SaaS use or external connections
Define acceptable use and app policies	Include clear guidance on what users are allowed to use
Monitor outbound DNS and proxy logs	Identify unusual domains or services in use
Educate staff on Shadow IT risks	Regular training and acceptable use policy (AUP) awareness

2. Detection & Analysis

Step	Action
Alert from CASB or network logs	Unapproved application or cloud service usage
Discover rogue device or access point	From network scans, NAC alerts or EDR telemetry
Correlate with user or department	Identify user or business unit responsible for use

Assess risk and scope	What data was accessed, where it's stored and who used it
MITRE ATT&CK mapping	T1584 (Compromise Infrastructure), T1087.001 (Account Discovery: Local Accounts), T1078 (Valid Accounts) — where Shadow IT may be part of attacker infrastructure or lateral movement

3. Containment

Step	Action
Block unauthorised service or domain	Via firewall, DNS or proxy controls
Disable rogue asset network access	Using NAC, switchport disablement or Wi-Fi controls
Revoke user access	To unauthorised apps or tools discovered in use
Notify responsible teams	Work with the team or user who introduced the asset to understand business intent

4. Eradication

Step	Action
Remove unapproved software	From endpoints, servers or internal systems
Decommission rogue infrastructure	Shutdown VMs, containers, cloud services or local hosts not in inventory
Clean credentials	If passwords or tokens were shared with unauthorised systems
Update asset discovery signatures	Add new detection rules for similar tools or configurations in future scans

5. Recovery

Step	Action
Onboard approved replacements	Help users move to authorised tools or services
Restore normal access	Only after all affected systems are validated and secured
Update asset inventory	Include newly discovered legitimate systems under official tracking
Revalidate user roles	Ensure no privilege creep or policy bypass remains active

6. Lessons Learned & Reporting

Step	Action
Document incident timeline	What was found, how, when and by whom
Improve user workflows	Provide secure and supported alternatives to Shadow IT solutions
Revise acceptable use policy	Clarify rules and include escalation for exceptions
Share incident report	With IT, security governance and department leads
Conduct follow-up audits	To verify similar assets or services are not in use elsewhere

Tools Typically Involved

- CASB (e.g., Netskope, Microsoft Defender for Cloud Apps, McAfee MVISION)
- SIEM (e.g., Splunk, Sentinel, QRadar)
- Endpoint tools (e.g., CrowdStrike, Cortex XDR)
- Network scanners (e.g., Nmap, Qualys, Nessus, Fing)
- DNS/Proxy logs and analytics (e.g., Cisco Umbrella, Squid, Zscaler)
- CMDB / IT asset management (e.g., ServiceNow, Lansweeper)

Success Metrics

Metric	Target
Detection Time	<1 day from introduction of asset
Containment Time	<4 hours from confirmation
Asset Inventory Update Time	Within 24 hours post-incident
User Re-education Completion	100% of involved users retrained within 7 days
Policy Compliance Enforcement	Confirmed for similar cases during next audit cycle

SOC Incident Response Playbook 14: RDP Brute-Force Attack

Scenario

An attacker launches a brute-force or password spraying attack against internet-exposed or internal RDP services to gain access using weak or reused credentials. Successful access may lead to lateral movement, malware deployment or data exfiltration.

Incident Classification

Category	Details
Incident Type	Credential Attack – RDP Login Abuse
Severity	High (especially for privileged or sensitive systems)
Priority	Critical if access is gained
Detection Sources	SIEM, Windows Security Event Logs, EDR, IDS/IPS, Firewall logs, Threat Intel

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Restrict RDP exposure	Use VPN, Zero Trust access or restrict via firewall
Enforce strong authentication	Use MFA and disable default admin accounts
Monitor RDP login events	Enable logging of Event ID 4625 (failed logins) and 4624 (success)
Apply account lockout policy	Limit the number of failed login attempts
Deploy honeypots or decoys	Detect brute-force attempts proactively on fake systems

2. Detection & Analysis

Step	Action
Alert from SIEM or EDR	Spike in failed RDP login attempts, password spraying behaviour
Review Event Logs	Filter by Event ID 4625 and identify common usernames and IPs
Correlate successful logins	Determine if brute-force succeeded and privilege was escalated
Analyse attacker IPs	Check geolocation, reputation and reoccurrence in other systems
MITRE ATT&CK mapping	T1110.001 (Brute Force - Password Guessing), T1078 (Valid Accounts), T1021.001 (Remote Services - RDP)

3. Containment

Step	Action
Block attacker IPs	At firewall, IDS or VPN gateway
Disable affected accounts	Lock or reset accounts that were targeted or compromised
Isolate affected hosts	If lateral movement or malware deployment is suspected
Throttle or disable RDP	Temporarily disable RDP on high-risk systems until secured

4. Eradication

Step	Action
Remove unauthorised access	Kill sessions, reset passwords and revoke tokens or certificates
Patch exposed systems	Update RDP services and OS to prevent exploits (e.g., BlueKeep)
Clean persistence mechanisms	Check for new scheduled tasks, services or registry keys added by attacker
Validate no lateral movement	Use EDR or log review to ensure attacker did not spread internally

5. Recovery

Step	Action
Reinstate secure RDP access	Only via VPN or bastion host with MFA
Notify users or IT teams	Alert those impacted by the attempted logins or credential resets
Monitor closely post-incident	Watch for continued brute-force activity or targeted retries
Conduct password audit	Prompt company-wide password hygiene checks if weak credentials were used

6. Lessons Learned & Reporting

Step	Action
Analyse timeline	Review when attack started, when it was detected and how quickly it was stopped
Update SIEM rules	Improve detection of brute-force indicators and high-failure thresholds
Revise access policies	Implement stricter controls over RDP use across the organisation
Share findings	With internal stakeholders and, if required, external authorities or vendors

Test security controls	Verify detection, prevention and response worked as expected or need tuning

Tools Typically Involved

- SIEM (e.g., Splunk, Sentinel, QRadar)
- EDR (e.g., CrowdStrike, Microsoft Defender for Endpoint, Cortex XDR)
- Firewall and VPN logs (e.g., Fortinet, Palo Alto, Cisco ASA)
- Windows Event Viewer (Security Logs: 4624, 4625, 4648, 4672)
- Threat intelligence platforms (for IP enrichment)
- Brute-force detection scripts or SOAR playbooks

Success Metrics

Metric	Target
Detection Time	<5 minutes from brute-force pattern onset
Containment Time	<30 minutes from confirmation
Credential Reset Time	<2 hours for compromised or targeted accounts
Exposure Time	No unauthorised RDP access exceeding 15 minutes
RDP Lockdown Coverage	100% of internet-facing RDP endpoints secured or removed

SOC Incident Response Playbook 15: Unauthorised Access to Development Environments

Scenario

An individual gains access to a development environment (e.g., Git repositories, staging servers, CI/CD platforms like Jenkins or GitLab CI or test databases) without authorisation. This may result in code theft, insertion of malicious code or exposure of credentials and secrets.

Incident Classification

Category	Details
Incident Type	Access Control Violation / Insider Threat
Severity	High (especially if source code or secrets are accessed or modified)
Priority	Critical
Detection Sources	SIEM, Git logs, IAM, DevOps tools (e.g., Jenkins, GitLab), Audit trails, Source code version control systems

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enforce role-based access control	Use IAM policies, SSO and least privilege for all dev tools
Enable auditing and logs	Track access to Git repositories, CI/CD systems and test environments
Integrate logging into SIEM	Stream logs from GitHub/GitLab, Jenkins, etc.
Apply secret scanning tools	Detect hardcoded credentials and tokens in code
Conduct DevSecOps training	Educate developers on secure coding and repository hygiene

2. Detection & Analysis

Step	Action
Alert from IAM/SIEM	Suspicious login, token use or repo access from unusual IP or user
Review Git/CI logs	Check recent commits, merges, pipeline executions and user access
Identify access method	Direct login, API token, SSH key or OAuth integration

Assess data touched	Determine whether source code, pipeline configs or secrets were accessed
MITRE ATT&CK mapping	T1087.001 (Account Discovery), T1059 (Command Execution via CI), T1606 (Forge Web Credentials), T1565.002 (Data Manipulation - Code Repositories)

3. Containment

Step	Action
Revoke user/API access	Disable user accounts or tokens used for access
Suspend pipeline execution	Pause CI/CD activities to prevent further compromise
Isolate affected systems	Temporarily block access to critical environments or servers
Notify DevOps and security teams	Coordinate containment and code review activities

4. Eradication

Step	Action
Remove unauthorised code or scripts	Revert malicious commits, rollback pipeline changes
Rotate secrets and credentials	Especially if found in code, environment variables or configuration files
Clean up compromised accounts	Remove old users, service accounts or tokens
Patch tool vulnerabilities	Apply updates to exposed or misconfigured DevOps platforms

5. Recovery

Step	Action
Restore secure state	Confirm code and pipelines are clean, access is limited to authorised users
Resume CI/CD operations	Only after validation of system integrity
Monitor codebase and build process	Set up enhanced logging and monitoring post-incident
Revalidate audit controls	Ensure access logs, versioning and change tracking are enabled and working

6. Lessons Learned & Reporting

Step	Action
Conduct a full review	Understand attack vector, user involved and data affected

Update access policies	Apply tighter controls to sensitive repos and build systems
Train DevOps personnel	Reinforce secure access management and code review policies
Report to stakeholders	Legal, compliance and clients if proprietary or regulated data is involved
Document playbook updates	Improve future detection and response processes in the SOC and DevOps teams

Tools Typically Involved

- SIEM (e.g., Sentinel, Splunk)
- Git platforms (e.g., GitHub, GitLab, Bitbucket)
- CI/CD tools (e.g., Jenkins, GitLab CI, CircleCI, Azure DevOps)
- IAM & SSO (e.g., Okta, Azure AD, Google Workspace)
- Secret scanners (e.g., TruffleHog, Gitleaks)
- Container security tools (e.g., Aqua, Prisma Cloud)

Success Metrics

Metric	Target
Detection Time	<10 minutes from unauthorised access
Access Revocation Time	<30 minutes from alert
Secret Rotation Time	<1 hour for high-value tokens or keys
Codebase Validation Time	<24 hours
Post-Incident Audit Completion	Within 3 business days

SOC Incident Response Playbook 16: Abuse of OAuth Integrations

Scenario

An attacker gains access to a user's cloud or application account by tricking them into authorising a malicious OAuth app (e.g., through phishing or social engineering). This gives persistent access without requiring login credentials, bypassing MFA in many cases.

Incident Classification

Category	Details
Incident Type	Third-Party App Abuse / Token-Based Account Compromise
Severity	High to Critical (especially if privileged or sensitive account access is granted)
Priority	Critical
Detection Sources	SIEM, OAuth audit logs, Cloud identity platforms (e.g., Google Workspace, Azure AD), User reports, Threat intelligence feeds

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Monitor OAuth authorisations	Enable logging for app consents and token grants
Limit third-party app permissions	Apply policies to restrict high-privilege access
Enforce admin consent workflows	Require security team approval for risky OAuth scopes
Educate users on phishing risks	Highlight risks of authorising unknown apps
Integrate identity logs with SIEM	Correlate token activity and app installations

2. Detection & Analysis

Step	Action
Detect suspicious app consent	OAuth app requesting abnormal permissions or used widely across accounts
Review audit logs	Check for tokens issued to unknown or recently created apps
Identify affected users	Map users who authorised the malicious app and assess data access scope
Investigate app behaviour	Determine if the app accessed email, files, cloud storage or contacts
MITRE ATT&CK mapping	T1525 (Implant Internal Image), T1556.004 (Forge Web Credentials), T1087 (Account Discovery)

3. Containment

Step	Action
Revoke app access	Remove the OAuth grant/token from affected accounts via admin portal or API
Block app at the tenant level	Ban the app's client ID in Google Workspace, Azure or GitHub settings
Disable impacted accounts (if needed)	If attacker used app access to escalate further
Notify users	Alert them about the revocation and potential data exposure

4. Eradication

Step	Action
Rotate access tokens and passwords	For users and service accounts if app accessed credentials or secrets
Clean up affected environments	Delete any backdoors, forwarding rules or uploaded files created via OAuth app
Strengthen tenant-wide policies	Restrict risky OAuth scopes (e.g., offline access, mail.readwrite, drive full access)
Update phishing protection	Block related phishing domains or links distributing the OAuth app

5. Recovery

Step	Action
Restore affected access	After confirming account is secure and OAuth app has been removed
Re-audit connected apps	Confirm no other high-risk apps are installed across users
Reinforce user MFA & session controls	Tighten identity policies (e.g., revoke sessions, require re-authentication)
Resume business operations	Once no further risk from the malicious integration remains

6. Lessons Learned & Reporting

Step	Action
Conduct impact analysis	What data was accessed or shared by the app? Over what timeframe?
Report if necessary	To legal, compliance, regulators (e.g., PDPA, GDPR) or customers

Update consent policies	Require tighter admin control for risky app scopes
Share incident details internally	Raise awareness and update security awareness training materials
Improve detection logic	Tune SIEM or SOAR playbooks to detect high-risk OAuth grants and anomalies

Tools Typically Involved

- Cloud admin consoles (e.g., Google Workspace Admin, Azure AD Portal, Microsoft 365 Defender)
- Identity Protection (e.g., Okta, Duo, Conditional Access)
- SIEM (e.g., Splunk, Sentinel, QRadar)
- Cloud security tools (e.g., Microsoft Defender for Cloud Apps, G Suite Alert Center)
- Threat intelligence for phishing and app reputation

Success Metrics

Metric	Target
Detection Time	<15 minutes from risky app grant
App Revocation Time	<30 minutes from identification
User Notification Time	<1 hour for affected users
OAuth Policy Enforcement	100% of users behind admin-consented model (for sensitive scopes)
Incident Resolution Time	Within 24–48 hours post-discovery

SOC Incident Response Playbook 17: Data Exfiltration via DNS Tunnelling

Scenario

An attacker uses DNS as a communication channel to exfiltrate data or maintain command and control. DNS tunnelling disguises malicious payloads or stolen data inside DNS queries, bypassing traditional detection since DNS traffic is usually allowed.

Incident Classification

Category	Details
Incident Type	Covert Channel – Data Exfiltration
Severity	High to Critical (especially if sensitive data is confirmed to be exfiltrated)
Priority	Critical
Detection Sources	DNS logs, SIEM, NDR, Threat intelligence, Endpoint alerts, Zeek, Suricata

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enable detailed DNS logging	From DNS resolvers and forwarders (e.g., BIND, Windows DNS, Unbound)
Implement DNS inspection	Use NDR (e.g., Corelight, Darktrace), firewall rules and pattern matching
Monitor for anomalous DNS activity	Large TXT queries, long domain names, unusual frequencies
Block known tunnelling tools	e.g., Iodine, DNScat2, DnsExfiltrator via threat intelligence
Enforce least privilege on outbound DNS	Allow only authorised DNS resolvers from internal assets

2. Detection & Analysis

Step	Action
Detect abnormal DNS patterns	Long subdomain lengths, frequent queries to rare domains, base64 encoding
Review DNS logs	Identify queried domains, query types (e.g., TXT, NULL) and endpoints involved
Correlate with asset behaviour	Determine if endpoints sending queries also show signs of compromise

Validate domain ownership	Check if the suspicious domains are attacker-controlled or registered recently
MITRE ATT&CK mapping	T1048.003 (Exfiltration Over Unencrypted/Obfuscated Non-C2 Protocol), T1071.004 (Application Layer Protocol: DNS), T1568.002 (Dynamic Resolution - DNS)

3. Containment

Step	Action
Block suspicious domains	At DNS resolver, firewall and proxy level
Isolate affected hosts	Disconnect from network to stop ongoing exfiltration
Redirect DNS traffic	Force all outbound DNS through monitored internal DNS servers
Alert internal stakeholders	IT, security and management should be informed immediately

4. Eradication

Step	Action
Remove tunnelling tools or malware	From endpoints using EDR or forensic analysis
Patch exploited vulnerabilities	If attacker gained access through known weaknesses
Clean persistence mechanisms	Check for scheduled tasks, registry changes or startup scripts
Review DNS configurations	Ensure no external DNS bypasses exist on endpoints or servers

5. Recovery

Step	Action
Restore network connectivity	Once system is verified to be clean and containment controls are in place
Resume DNS services	Enforce forwarding through secure DNS infrastructure with inspection
Revalidate affected systems	Perform full scan and traffic monitoring on previously infected hosts
Update threat detection rules	Enhance SIEM, NDR and firewall rules with new indicators and patterns

6. Lessons Learned & Reporting

Step	Action

Document full exfiltration path	Identify what data was targeted or lost, how and when
Update incident response plans	Add DNS-specific detection and response protocols
Improve DNS visibility	Enforce structured DNS logging and analytics across all environments
Report breach if applicable	Under PDPA, GDPR, HIPAA or industry-specific regulations
Share IOCs and findings	Internally and with external threat intelligence communities (e.g., ISACs)

Tools Typically Involved

- DNS Logging Platforms (e.g., Infoblox, Bind logs, Windows DNS logs)
- NDR (e.g., Corelight/Zeek, Darktrace, ExtraHop)
- SIEM (e.g., Splunk, Sentinel, QRadar)
- EDR (e.g., CrowdStrike, Cortex XDR)
- Threat Intelligence (e.g., Recorded Future, MISP, AbuseIPDB)
- Firewall and proxy logs

Success Metrics

Metric	Target
Detection Time	<15 minutes from abnormal DNS pattern
Containment Time	<30 minutes from confirmation
Data Loss Impact Report	Within 48 hours (or regulatory timeframe)
DNS Logging Coverage	100% of egress DNS activity logged and monitored
Incident Review Completion	Within 72 hours post-resolution

SOC Incident Response Playbook 18: Unauthorised JavaScript Injection on Public Websites

Scenario

An attacker injects malicious JavaScript code into a public-facing website (via compromised CMS, third-party scripts, misconfigured CDN or direct file replacement). This could lead to credential harvesting, skimming (e.g., Magecart), clickjacking, redirection to malicious sites or session hijacking.

Incident Classification

Category	Details
Incident Type	Web Application Compromise – Script Injection
Severity	High to Critical (especially if PII, card data or authentication data is captured)
Priority	Critical
Detection Sources	WAF, SIEM, Website Monitoring Tools, Bug Bounty Reports, Client Feedback, Threat Intel Feeds

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Implement CSP (Content Security Policy)	Restrict unauthorised scripts from loading
Monitor file integrity	Use tools to track changes in JS files on production
Use Subresource Integrity (SRI)	For third-party scripts to ensure they aren't tampered
Enable Web Application Firewall (WAF)	Block suspicious inputs or exploit attempts
Perform regular code audits	Especially on CMS plugins and third-party inclusions

2. Detection & Analysis

Step	Action
Alert from WAF or monitoring tool	Detection of injected or modified JavaScript
Validate file changes	Compare modified JavaScript to known good versions (git, backups)

Review injection source	Was it from a CMS plugin, third-party source or direct code edit?
Examine script behaviour	Analyse the payload: keylogging, exfiltration, redirection, data capture
MITRE ATT&CK mapping	T1059.007 (JavaScript), T1185 (Browser Session Hijacking), T1189 (Drive-by Compromise), T1557.002 (Input Capture via Web Script)

3. Containment

Step	Action
Remove or replace injected script	Immediately restore clean versions from backup or repository
Block malicious domain	If external scripts were involved, block via DNS, proxy or firewall
Disable affected parts of the site	Temporarily take down the compromised section or page if necessary
Alert customers/users	If data harvesting occurred, communicate the exposure risk quickly

4. Eradication

Step	Action
Identify root cause	Compromised admin credentials? Insecure plugin? Third-party breach?
Patch CMS or plugin	Apply updates and disable unnecessary or untrusted components
Replace compromised components	Reinstall from official sources with verified integrity
Clean residual access	Change admin credentials, revoke tokens, check server logs for persistence techniques

5. Recovery

Step	Action
Validate website integrity	Recheck all files and scripts for correctness and cleanliness
Resume normal operation	After confirming no malicious code remains
Perform vulnerability assessment	Especially on exposed CMS, JavaScript includes and APIs
Monitor for repeat attempts	Increase web traffic and behaviour monitoring temporarily

6. Lessons Learned & Reporting

Step	Action
Document the incident	Include injection vector, impact, attacker domain and mitigation steps
Update web app monitoring rules	Add new indicators of compromise for JavaScript integrity alerts
Train web developers and admins	On safe script practices, plugin security and change control
Report to regulators	If user data or payment information was compromised
Improve SDLC security	Integrate code scanning, dependency checks and CI/CD validation in development workflows

Tools Typically Involved

- Web Monitoring Tools (e.g., Detectify, JSWatcher, SilentPush, Snyk, Sucuri)
- SIEM (e.g., Splunk, Sentinel, QRadar)
- Web Application Firewall (e.g., Cloudflare WAF, AWS WAF, Imperva)
- CMS platforms and source repositories (e.g., WordPress, GitHub)
- File integrity monitoring (e.g., OSSEC, Tripwire)

Success Metrics

Metric	Target
Detection Time	<15 minutes from script modification or alert
Script Removal Time	<30 minutes after detection
Website Restoration Time	<2 hours if critical path is affected
Impact Notification Time	Within 24 hours (or regulatory SLA)
Repeat Attack Monitoring Duration	Minimum 7 days of enhanced surveillance

SOC Incident Response Playbook 19: Insecure API Endpoint Exploitation

Scenario

An attacker discovers and exploits insecure API endpoints—such as those lacking authentication, rate limiting or proper input validation—to perform unauthorised data access, modify business logic, escalate privileges or carry out denial-of-service (DoS) attacks.

Incident Classification

Category	Details
Incident Type	Application-Layer Exploit – API Abuse
Severity	High to Critical (depending on data sensitivity and access level gained)
Priority	High
Detection Sources	SIEM, API Gateway Logs, WAF, Runtime Application Security Protection (RASP), Application Logs, Threat Intelligence Feeds

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enable API gateway logging	Track request methods, source IPs, endpoints and parameters
Enforce input validation	Implement strict validation and sanitisation in backend logic
Deploy rate limiting & throttling	Prevent abuse through bulk or automated requests
Require authentication & authorisation	Use OAuth, JWT, API keys with role enforcement
Monitor API behaviour	Use anomaly detection to flag unexpected access patterns

2. Detection & Analysis

Step	Action
Receive alert	Excessive API calls, unauthorised data access, error spikes (e.g., 403s, 500s)
Identify affected endpoints	Analyse logs to determine what APIs were accessed and how
Review query patterns	Look for signs of enumeration, injection, mass scraping or business logic abuse

Correlate with user/IP	Determine whether the actor is internal, authenticated or abusing open APIs
MITRE ATT&CK mapping	T1190 (Exploit Public-Facing Application), T1499 (Endpoint DoS), T1001.003 (Data Obfuscation), T1539 (Steal Web Session Cookie)

3. Containment

Step	Action
Block offending IPs or tokens	At the API gateway, WAF or CDN level
Disable vulnerable endpoint	Temporarily disable or restrict access to the affected API
Throttle suspicious traffic	Enforce tighter rate limits for abusive patterns
Alert development and product teams	To assist with containment and business risk assessment

4. Eradication

Step	Action
Fix insecure API logic	Add authentication, access control and input validation
Patch or redeploy backend service	If vulnerability is rooted in code or library
Rotate affected credentials or API keys	Especially if token theft or privilege abuse occurred
Remove injected data	If attacker used the API to insert malicious or corrupt data

5. Recovery

Step	Action
Restore secure access	After verifying fix, monitor closely for any signs of bypass or regression
Inform affected users	If personal or sensitive data was accessed or altered
Retest affected APIs	Conduct regression and security testing before full reactivation
Resume full service	Once security and stability are verified in production environments

6. Lessons Learned & Reporting

Step	Action
Conduct a root cause analysis	Identify design, configuration or development oversight
Update SDLC policies	Include security testing for API endpoints (e.g., OWASP API Top 10)

Enhance monitoring and alerting	Add detection for enumeration, excessive calls or unusual inputs
Train developers	On secure API design, proper authentication and error handling
Document the incident	Include timeline, attacker behaviour, impact and mitigations applied

Tools Typically Involved

- API Gateways (e.g., Kong, AWS API Gateway, Apigee, Azure API Management)
- WAF (e.g., Cloudflare, AWS WAF, Imperva)
- SIEM (e.g., Sentinel, Splunk)
- RASP and Runtime Protection (e.g., Signal Sciences, Contrast Security)
- Application performance/logging tools (e.g., Datadog, New Relic)
- DAST tools (e.g., Burp Suite, OWASP ZAP)

Success Metrics

Metric	Target
Detection Time	<10 minutes from abnormal activity
Endpoint Restriction Time	<30 minutes after confirmation
Patch/Code Fix Deployment	<24–48 hours for critical API bugs
Retest & Recovery Time	Within 72 hours
Developer Training Coverage	100% of backend/API teams briefed within 7 days

SOC Incident Response Playbook 20: Insider Credential Theft and Misuse

Scenario

An insider (or an external actor using stolen insider credentials) uses valid accounts to access sensitive systems, extract data or perform unauthorised activities — often bypassing traditional security detection due to use of legitimate credentials.

Incident Classification

Category	Details
Incident Type	Insider Threat – Credential Abuse
Severity	High to Critical (especially if privileged accounts or sensitive data are involved)
Priority	Critical
Detection Sources	SIEM, UEBA (User and Entity Behaviour Analytics), IAM, EDR, DLP, HR tips or whistleblower reports

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Implement UEBA solutions	Detect anomalous user activity (e.g., Exabeam, Microsoft Defender, Securonix)
Enable logging for privileged accounts	Include session monitoring and command tracking
Enforce least privilege & RBAC	Ensure users only have access they truly need
Monitor for sensitive data access	DLP policies for file downloads, cloud storage, email forwarding
Set up alerting on atypical access patterns	Time of day, volume of activity, system accessed, location

2. Detection & Analysis

Step	Action
Alert from UEBA or SIEM	Abnormal data access, login behaviour or file movement
Correlate with job role	Determine if actions align with user's normal duties
Check access logs	Identify systems, files and data accessed

Review recent HR flags	Check if the user is under investigation, has resigned or shows behavioural risk indicators
MITRE ATT&CK mapping	T1078 (Valid Accounts), T1087 (Account Discovery), T1110.003 (Password Spraying), T1213.003 (Access Sensitive Data - Databases)

3. Containment

Step	Action
Suspend user account	Temporarily disable access during investigation
Revoke session tokens	Invalidate active sessions, VPN or API tokens
Quarantine endpoint	If file transfer, malware installation or persistence is suspected
Restrict further access	Implement just-in-time access or isolate the user's VLAN/subnet

4. Eradication

Step	Action
Investigate full activity scope	Review emails sent, files accessed/transferred, systems logged into
Revoke elevated access or credentials	Remove access from all privileged systems or services
Reset credentials and keys	For shared credentials or systems the user accessed
Clean up any changes	Roll back any script, configuration or data changes made by the user

5. Recovery

Step	Action
Restore access to legitimate users	If other accounts were suspended or disabled for investigation
Monitor systems touched	Use SIEM and EDR to monitor post-incident behaviour for a defined period
Notify stakeholders	Include HR, Legal and Compliance for coordination and investigation closure
Resume normal operations	Once it is verified that no lingering risk remains from insider activity

6. Lessons Learned & Reporting

Step	Action
Conduct a post-incident review	Determine how the misuse occurred and what failed to detect it earlier

Update detection rules	Add specific indicators of abuse for similar roles or behaviour
Review access policies	Tighten or adjust RBAC/least privilege settings and IAM processes
Enhance user monitoring policies	Periodic review of sensitive access by job role or department
Report if required	Internal governance bodies, regulators (e.g., PDPA, HIPAA) or affected clients

Tools Typically Involved

- SIEM (e.g., Splunk, Sentinel, QRadar)
- UEBA (e.g., Exabeam, Microsoft Defender for Identity, Securonix)
- EDR (e.g., CrowdStrike, Cortex XDR)
- IAM/SSO logs (e.g., Okta, Azure AD, Ping)
- DLP tools (e.g., Forcepoint, Symantec, Microsoft Purview)
- Endpoint and server logs
- HRIS integration (for real-time HR status feed)

Success Metrics

Metric	Target
Detection Time	<15 minutes from abnormal activity onset
Account Suspension Time	<30 minutes from alert confirmation
Root Cause Analysis Completion	Within 48 hours
Access Review Completion	100% of privileged access logs reviewed for the impacted user
Policy Review or Adjustment	Within 7 days of incident closure

SOC Incident Response Playbook 21: Cloud Identity Misconfiguration

Scenario

A misconfigured cloud identity or access control (e.g., overly permissive IAM role, wildcard access policy, unintended trust relationships) is exploited by an internal or external actor to gain elevated access, move laterally or access restricted resources.

Incident Classification

Category	Details
Incident Type	Misconfiguration – IAM / Access Policy
Severity	High to Critical (especially if privileged access or sensitive data is exposed)
Priority	Critical
Detection Sources	CSPM tools, Cloud audit logs, SIEM, IAM policy scans, Threat Intelligence, Red Team findings

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Use least privilege model	Enforce granular IAM roles and policy-based access controls
Deploy CSPM tools	Monitor for identity misconfigurations (e.g., Wiz, Prisma Cloud, Microsoft Defender for Cloud)
Enable detailed logging	Use AWS CloudTrail, Azure Activity Logs, GCP Audit Logs for tracking IAM events
Tag and classify identities	Differentiate human vs service identities and apply risk-based monitoring
Conduct access reviews	Regular audits of IAM roles, trust policies and access keys

2. Detection & Analysis

Step	Action
Alert triggered	CSPM or SIEM alert for excessive permissions, wildcard access ("*") or trust to Everyone
Validate misconfiguration	Review IAM policy, role assumptions, group memberships and any unusual inheritance
Review access logs	Determine if the misconfiguration has been exploited (API calls, resource access)

Assess affected assets	Identify what services or data were accessible using the misconfigured identity
MITRE ATT&CK mapping	T1078.004 (Cloud Accounts), T1098.001 (Additional Cloud Credentials), T1550.001 (Application Access Token Abuse)

3. Containment

Step	Action
Restrict or delete misconfigured role/policy	Immediately remove or correct the risky configuration
Revoke temporary credentials	Invalidate STS tokens, API keys, access tokens issued via the misconfigured identity
Quarantine affected resources	Isolate compromised services or data buckets if suspicious activity is confirmed
Notify cloud admin teams	Coordinate IAM changes and service validations across cloud accounts or regions

4. Eradication

Step	Action
Remediate IAM policy	Apply corrected policies with scoped permissions, conditions and role boundaries
Rotate affected credentials	Especially for users, service accounts or cloud-native secrets
Validate trust relationships	Reconfigure role assumptions and remove unintended cross-account trust
Remove unused roles/groups	Decommission identities that serve no operational need

5. Recovery

Step	Action
Restore proper access	Re-assign necessary permissions using least privilege principles
Monitor reconfiguration	Set temporary alerts on updated identities for post-fix behaviour validation
Re-enable services	After confirming configurations are secure and audit logs show no further misuse
Communicate status	Provide updates to security, DevOps and cloud platform owners

6. Lessons Learned & Reporting

Step	Action

Conduct impact analysis	Confirm whether data access or privilege abuse occurred and over what period
Improve IAM policies	Apply service control policies, permission boundaries and conditional logic
Update monitoring rules	Add detections for wildcard privileges, new identity creation and cross-account role use
Document the incident	Include the IAM resource affected, root cause, impacted assets and resolution steps
Report if needed	To internal stakeholders or regulatory bodies if sensitive data was accessed

Tools Typically Involved

- CSPM tools (e.g., Wiz, Prisma Cloud, Microsoft Defender for Cloud, AWS Config)
- SIEM (e.g., Splunk, Sentinel, QRadar)
- Cloud audit logs (e.g., AWS CloudTrail, Azure Activity Logs, GCP Audit Logs)
- IAM policy scanners (e.g., PMapper, CloudSploit, IAM Access Analyzer)
- SOAR for automated remediation
- Identity governance platforms (e.g., Saviynt, SailPoint, Okta)

Success Metrics

Metric	Target
Detection Time	<10 minutes for risky IAM change
Containment Time	<30 minutes from alert confirmation
Policy Fix Completion	<4 hours for critical misconfiguration
Credential Rotation Time	<2 hours for affected identities
Access Review Coverage	100% of affected identities audited post-incident

SOC Incident Response Playbook 22: CI/CD Pipeline Exploitation

Scenario

An attacker gains access to or exploits weaknesses in a CI/CD pipeline (e.g., Jenkins, GitLab CI, GitHub Actions) to manipulate build processes, inject malicious code or secrets or use the pipeline to pivot into broader infrastructure.

Incident Classification

Category	Details
Incident Type	Software Supply Chain / Pipeline Compromise
Severity	High to Critical (especially if deployment tampering or codebase access is confirmed)
Priority	Critical
Detection Sources	SIEM, Source Control Logs, CI/CD Logs, EDR, SAST/DAST Tools, Developer Reports, Threat Intel Feeds

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enforce RBAC on CI/CD tools	Restrict who can create or modify pipelines, runners or secrets
Enable detailed audit logging	For code pushes, pipeline changes, job execution and secret use
Use signed commits and artifacts	Validate authenticity of source and deployment packages
Monitor for pipeline abuse patterns	e.g., unexpected job triggers, privilege escalation via runners
Isolate build environments	Use ephemeral containers/VMs to limit lateral movement and access scope

2. Detection & Analysis

Step	Action
Alert from SIEM or DevSecOps tools	Unexpected job behaviour, credential use or build script changes
Review recent pipeline changes	Check job definitions, runner configurations and injected commands
Analyse source repo activity	Look for rogue commits, PRs or branch manipulations

Identify affected projects	Determine what builds/deployments may have been compromised
MITRE ATT&CK mapping	T1556 (Modify Authentication Process), T1587.002 (Malicious Code Signing), T1059.006 (CI/CD Job Command Execution), T1136.003 (Cloud Account Creation for Persistence)

3. Containment

Step	Action
Disable affected pipelines or runners	Stop further execution of compromised jobs
Revoke access to CI/CD tool	For compromised accounts or tokens
Block malicious artifacts	Prevent deployment of compromised containers, binaries or packages
Isolate affected environments	Temporarily remove impacted apps or services from the deployment path

4. Eradication

Step	Action
Clean malicious code or scripts	Revert to clean repo state; delete tampered build definitions
Rotate compromised secrets	Reissue API keys, cloud tokens, database credentials exposed in CI/CD logs
Patch vulnerabilities	Address misconfigurations in runners, plugins or access control
Audit third-party integrations	Remove or review access granted to external CI/CD plugins or services

5. Recovery

Step	Action
Restore trusted pipelines	After validating scripts, dependencies and configurations
Rebuild affected applications	Using known-good code and secured CI/CD process
Re-enable deployment	Once verified safe and complete validation is passed
Notify stakeholders	Inform developers, product owners and security teams of the recovery status

6. Lessons Learned & Reporting

Step	Action

Perform a root cause analysis	Determine whether the entry point was repo access, runner abuse or plugin compromise
Update pipeline security controls	Enforce code review, job approvals and secure secret management
Train DevOps and developers	On secure CI/CD practices and incident indicators
Report if necessary	If data was exposed or software shipped with malware (e.g., to customers, regulators)
Update playbooks	Include lessons learned and control enhancements for CI/CD monitoring

Tools Typically Involved

- CI/CD platforms (e.g., Jenkins, GitHub Actions, GitLab CI, Azure DevOps)
- SIEM (e.g., Splunk, Sentinel)
- EDR and Runtime protection (e.g., CrowdStrike, Aqua Security)
- Code and pipeline scanners (e.g., SonarQube, Checkov, TFSec)
- Source code management systems (e.g., GitHub, GitLab, Bitbucket)
- SOAR platforms (for auto-remediation of pipeline abuse)

Success Metrics

Metric	Target
Detection Time	<10 minutes from abnormal CI/CD activity
Job Disablement Time	<30 minutes from confirmation
Secret Rotation Time	<2 hours from exposure detection
Rebuild & Redeploy Time	Within 24–48 hours using verified code
CI/CD Access Review Completion	100% of user and integration access audited within 3 days

SOC Incident Response Playbook 23: Unauthorised Use of Generative AI Tools in Production

Scenario

An employee or system uses a generative AI tool in a production environment—either by pasting sensitive code, data or configuration into an AI prompt or by integrating an AI assistant into a live application—without formal approval or proper security evaluation.

Incident Classification

Category	Details
Incident Type	Policy Violation / Data Exposure Risk
Severity	Medium to Critical (depending on data sensitivity or automation impact)
Priority	High
Detection Sources	DLP, CASB, SIEM, Proxy Logs, Endpoint Telemetry, IT Governance Alerts, Security Awareness Reports

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Implement acceptable use policies	Clearly define boundaries for AI use across environments
Monitor AI platform access	Log and alert on interactions with generative AI URLs (e.g., openai.com, bard.google.com)
Use DLP and CASB tools	Monitor for sensitive data input into AI tools or external APIs
Enforce browser controls and blocking	Limit AI tool usage from high-sensitivity zones (e.g., finance, dev, prod)
Conduct user training	On generative AI risks and organisational compliance requirements

2. Detection & Analysis

Step	Action
Detect unauthorised use	DLP, CASB or proxy log alerts showing data pasted to AI tool or plugin usage in prod
Identify user or system	Correlate logs with source IP, user ID, browser agent or application logs
Review transmitted data	Determine if code, credentials, PII or intellectual property was included

Assess context of usage	Accidental misuse vs intentional automation or shadow AI integration
MITRE ATT&CK mapping	T1087.003 (Cloud Service Enumeration), T1567.002 (Exfiltration to Cloud Storage), T1203 (Exploit Public-Facing Application via AI Plugin/Extension)

3. Containment

Step	Action
Block further access	Disable user access to the AI tool or integration via firewall, proxy or CASB policy
Quarantine affected systems	If AI integration was in active code or service
Alert user and management	Notify stakeholders and freeze further use during investigation
Capture forensic snapshot	Of prompt history, browser activity and transferred data (where possible)

4. Eradication

Step	Action
Remove AI integration	From production services, scripts or pipelines if embedded
Revoke any API tokens used	In unauthorised AI integrations (e.g., OpenAI API keys)
Rotate exposed secrets	If credentials were pasted or stored by AI
Clean up policy violations	Update configurations to remove AI-related exceptions or allowlists if misused

5. Recovery

Step	Action
Restore access under policy	Only after users acknowledge acceptable use terms or AI plugins are audited and approved
Validate codebase and production changes	Ensure no unauthorised automation remains
Implement AI governance checks	Introduce review workflows for AI-related tool usage and integrations
Resume operations	Once security and compliance teams confirm risk is mitigated

6. Lessons Learned & Reporting

Step	Action

Conduct root cause analysis	Why and how the AI tool was accessed or integrated
Update monitoring controls	Add detections for new AI platforms or browser extensions
Improve internal education	Add AI-specific scenarios to cybersecurity awareness programs
Report if required	If IP or regulated data was exposed (e.g., GDPR, HIPAA, PDPA compliance)
Document incident	Include users involved, data accessed, remediation timeline and preventive steps

Tools Typically Involved

- DLP (e.g., Microsoft Purview, Symantec, Forcepoint)
- CASB (e.g., Netskope, Microsoft Defender for Cloud Apps)
- SIEM (e.g., Splunk, Sentinel, QRadar)
- Endpoint Detection and Response (e.g., CrowdStrike, Cortex XDR)
- Proxy/Firewall Logs (e.g., Zscaler, Palo Alto, Fortinet)
- Browser control tools (e.g., Chrome enterprise policies, Edge management)
- Generative AI access logs (if integrated with enterprise identity systems)

Success Metrics

Metric	Target
Detection Time	<10 minutes from data transfer or plugin use
Containment Time	<30 minutes for access revocation
Risk Assessment Completion	<24 hours from incident start
Policy Re-acknowledgment Rate	100% of involved users within 3 days
Compliance Review Timeframe	Within 7 days of incident resolution

SOC Incident Response Playbook 24: OAuth Token Replay Abuse

Scenario

An attacker obtains a valid OAuth access token (e.g., via phishing, token theft or insecure storage) and reuses it to access APIs, web applications or cloud services as the victim — bypassing MFA and other login protections since the token is already trusted.

Incident Classification

Category	Details
Incident Type	Identity Compromise – Token Abuse
Severity	High to Critical (depending on the scope and privilege of the token)
Priority	Critical
Detection Sources	SIEM, Cloud Audit Logs, API Gateway Logs, Identity Provider Logs (e.g., Okta, Azure AD), CASB, Threat Intel Feeds

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enforce token expiration and rotation	Set short token lifetimes and enforce refresh token limits
Monitor token usage patterns	Use identity protection or SIEM to alert on abnormal API access using tokens
Tie tokens to device/session	Bind issued tokens to IP/device fingerprints where possible
Implement Conditional Access	Check context (e.g., location, app, risk score) before allowing token-based access
Log all token issuance and usage	From identity providers and application gateways

2. Detection & Analysis

Step	Action
Alert triggered	Abnormal token usage such as reuse from new location or impossible travel behaviour
Investigate token use patterns	Check endpoints accessed, time of use, associated IP and user-agent metadata
Correlate with token issuance	Determine when and where the token was first created and if it aligns with the legitimate user

Assess exposure risk	Determine whether data access, privilege escalation or account actions occurred
MITRE ATT&CK mapping	T1528 (Steal Application Access Token), T1078.004 (Cloud Accounts – OAuth Abuse), T1550.003 (Token Impersonation)

3. Containment

Step	Action
Revoke active tokens	Invalidate both access and refresh tokens via IdP or app settings
Block source IPs	If replay originated from known malicious infrastructure or unusual regions
Suspend affected user accounts	Temporarily disable to prevent continued exploitation during investigation
Alert user and security team	Notify of the potential compromise and suspend external access if needed

4. Eradication

Step	Action
Rotate credentials and secrets	Especially for third-party applications or APIs tied to the same account
Audit and remove malicious app consents	Check for OAuth apps granted by the user that may be controlled by the attacker
Tighten app permission scopes	Restrict apps to only request minimum access necessary (principle of least privilege)
Apply security controls to apps	Require app verification or tenant-level consent approval for future apps

5. Recovery

Step	Action
Reinstate user account	After confirming user identity and account integrity
Monitor token activity post-recovery	Ensure new tokens are being used only from trusted locations and devices
Revalidate app and API access	Confirm legitimate session behaviour across critical services
Resume operations	After confirming full containment and credential hygiene

6. Lessons Learned & Reporting

Step	Action

Perform root cause analysis	Determine how the token was obtained (e.g., phishing, local storage exposure, browser extension)
Update token issuance policies	Reduce token lifetimes, enforce refresh limits and bind to context
Improve detections	Add token replay pattern signatures to SIEM and identity protection platforms
Educate users and dev teams	On secure storage and handling of OAuth tokens (especially in browser-based apps)
Report if required	Especially if sensitive data was accessed (e.g., under GDPR, HIPAA or PDPA)

Tools Typically Involved

- Identity Providers (e.g., Okta, Azure AD, Google Workspace)
- SIEM (e.g., Sentinel, Splunk, QRadar)
- CASB (e.g., Netskope, Microsoft Defender for Cloud Apps)
- API Security Tools (e.g., Salt Security, Noname, Imperva API Security)
- Cloud Audit Logs (e.g., AWS CloudTrail, Azure Sign-in logs, GCP Admin Activity)
- User Behaviour Analytics (e.g., Exabeam, Securonix)

Success Metrics

Metric	Target
Detection Time	<10 minutes from abnormal token use
Token Revocation Time	<15 minutes after confirmation
Account Risk Mitigation Time	<1 hour
OAuth App Audit Completion	100% of consents reviewed within 24 hours
Post-Incident Monitoring Period	Minimum 7 days with enhanced visibility

SOC Incident Response Playbook 25: Misconfigured Public Cloud Storage Access

Scenario

A cloud storage bucket, container or object is unintentionally made publicly accessible or exposed to unauthorised users (e.g., via public-read or authenticated users access settings). This may lead to data leakage, regulatory non-compliance or exploitation by threat actors.

Incident Classification

Category	Details
Incident Type	Cloud Misconfiguration – Public Exposure
Severity	High to Critical (depending on sensitivity of exposed data)
Priority	Critical
Detection Sources	CSPM Tools, Cloud Audit Logs, SIEM, Threat Intelligence, Manual Discovery, Bug Bounty Reports

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Deploy CSPM tools	Continuously scan for misconfigured storage (e.g., Wiz, Prisma Cloud orca Security)
Implement policy-as-code	Use tools like AWS Config, Azure Policies or GCP Org Policies to restrict public storage
Monitor access logs	Enable logging for bucket access and object-level events
Enable default encryption	Automatically encrypt all objects with KMS keys
Classify and tag sensitive data	Apply metadata for easier DLP and access control enforcement

2. Detection & Analysis

Step	Action
Alert from CSPM or cloud platform	Detection of public-read, public-write or wildcard access to storage
Verify access level	Confirm if the object, bucket or container is readable by anyone or broad IAM groups
Check access logs	Identify IPs, users or services that accessed the exposed resource

Determine data sensitivity	Was the data PII, financial, source code or internal documentation?
MITRE ATT&CK mapping	T1530 (Data from Cloud Storage Object), T1526 (Cloud Service Discovery), T1213.003 (Access Sensitive Data)

3. Containment

Step	Action
Remove public access	Immediately revoke public or everyone permissions on the storage resource
Apply least privilege policies	Lock down access to only required IAM identities or roles
Revoke temporary tokens (if abused)	Disable access keys or tokens used to exploit the exposure
Quarantine exposed data (optional)	Move sensitive files to a restricted bucket for analysis or remediation

4. Eradication

Step	Action
Correct IAM or ACL policies	Use policy templates or automation to enforce secure access controls
Rotate keys or tokens	If access keys, SAS tokens or signed URLs were exposed
Remove unauthorised files	Delete uploaded malware, backdoors or tampered content (if applicable)
Disable bucket listing	Prevent attackers from enumerating contents in the future

5. Recovery

Step	Action
Restore secure access	Only after verifying proper permissions are in place
Notify affected teams	Especially data owners, app teams, compliance and legal if sensitive data was involved
Resume usage	After confirming no remaining exposure or misconfiguration
Enable stronger logging	If not already in place, ensure CloudTrail/S3/Azure Blob/GCP audit logs are active

6. Lessons Learned & Reporting

Step	Action
Conduct full exposure analysis	Determine duration, access scope and data classification of exposed content

Update access control templates	Enforce deny-by-default posture for new storage resources
Train teams on cloud access policies	Educate on secure storage deployment and data handling best practices
Report if necessary	Under PDPA, GDPR, HIPAA, etc. if data breach involves personal or regulated data
Integrate detection into CI/CD	Catch public access settings before production using IaC scanning (e.g., Checkov, tfsec)

Tools Typically Involved

- CSPM (e.g., Wiz, Prisma Cloud orca)
- Cloud-native tools (e.g., AWS S3 Access Analyzer, Azure Defender, GCP Security Command Center)
- SIEM (e.g., Sentinel, Splunk)
- DLP tools (e.g., Microsoft Purview, Google DLP)
- IAM policy analyzers (e.g., PMapper, CloudSploit)
- Threat Intel feeds for leaked buckets/domains

Success Metrics

Metric	Target
Detection Time	<5 minutes from misconfiguration
Public Access Removal Time	<15 minutes from alert
Exposure Impact Report	Within 24 hours
IAM Policy Audit Completion	100% of affected projects or buckets within 48 hours
Compliance Review Completion	Within 72 hours (or regulatory deadline)

SOC Incident Response Playbook 26: Lateral Movement Across Cloud Workloads

Scenario

An attacker gains a foothold in one cloud workload (e.g., EC2, Azure VM, Kubernetes pod or container) and moves laterally by leveraging over-permissive roles, unsecured credentials, shared storage or misconfigured network rules to reach other workloads or services.

Incident Classification

Category	Details
Incident Type	Cloud Intrusion – Lateral Movement
Severity	High to Critical (depending on the systems accessed and data exposed)
Priority	Critical
Detection Sources	SIEM, CSPM, EDR on cloud workloads, Cloud Audit Logs, NDR, Threat Intel Feeds

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enforce network segmentation	Use security groups, NSGs or VPC firewalls to restrict east-west traffic
Enable workload logging	Activate OS logs, CloudTrail, Azure Activity Logs, GCP Audit Logs and flow logs
Deploy EDR/EDR for cloud	Install endpoint protection or runtime security tools (e.g., Falcon, XDR, Wiz Runtime)
Limit IAM role reuse	Ensure minimal sharing of roles/permissions across workloads
Harden images and infrastructure	Use secure images and enforce IaC best practices

2. Detection & Analysis

Step	Action
Alert triggered	Suspicious inter-instance communication, credential use or lateral command execution
Identify entry point	Locate the initial compromised workload or credential source
Trace lateral path	Review cloud flow logs, audit trails and EDR logs for signs of SSH, API calls, remote access

Analyse tools used	Was movement done via scripts, stolen tokens, RDP/SSH or cloud-native APIs?
MITRE ATT&CK mapping	T1021 (Remote Services), T1570 (Lateral Tool Transfer), T1086.001 (PowerShell on Cloud Host), T1534 (Internal Spearphishing or Role Impersonation)

3. Containment

Step	Action
Isolate affected workloads	Remove compromised instances/pods from the network or scale down affected services
Disable involved credentials or roles	Immediately revoke tokens, keys or IAM roles used in lateral movement
Block east-west traffic temporarily	Apply strict ACLs to prevent further movement while analysing scope
Alert platform and application owners	Notify relevant teams about affected environments

4. Eradication

Step	Action
Terminate compromised instances or containers	Rebuild using trusted images and validated IaC templates
Rotate affected credentials	Reissue cloud access keys, service principals and user passwords involved
Remove backdoors or persistence	Check cron jobs, startup scripts, IAM roles or installed malware
Fix network/security group rules	Prevent recurrence by enforcing least-access models

5. Recovery

Step	Action
Re-deploy clean workloads	From verified pipelines or hardened base images
Restore network trust zones	Gradually re-enable east-west communication with strict controls
Re-enable affected services	Only after thorough validation and logging is in place
Increase monitoring on recovery assets	Use SIEM and runtime tools to validate clean operation

6. Lessons Learned & Reporting

Step	Action

Document lateral movement path	Include accessed systems, tools used, timeline and exposed data
Update detection logic	Add indicators of compromise and new behavioural rules for lateral movement
Harden IAM and network design	Apply tighter segmentation, SSO constraints and token binding techniques
Conduct internal debrief	Review with security, cloud ops, DevOps and compliance teams
Report if necessary	To regulators or customers, especially if sensitive data or production systems were compromised

Tools Typically Involved

- SIEM (e.g., Sentinel, Splunk, QRadar)
- CSPM (e.g., Wiz, Prisma Cloud, Defender for Cloud)
- Cloud EDR (e.g., CrowdStrike, Cortex XDR, Falco for containers)
- Cloud audit logs (AWS CloudTrail, Azure Activity Logs, GCP Audit Logs)
- Network visibility (e.g., VPC Flow Logs, Azure NSG flow logs)
- SOAR tools for response orchestration

Success Metrics

Metric	Target
Detection Time	<15 minutes from lateral movement start
Isolation Time	<30 minutes from detection
Credential Rotation Time	<2 hours from confirmation
Affected Asset Recovery Time	Within 48 hours
Post-Mortem Report Completion	Within 3 business days

SOC Incident Response Playbook 27: Unauthorised Cloud Database Snapshot Exports

Scenario

A cloud database snapshot (e.g., AWS RDS snapshot, Azure SQL Database export, GCP Cloud SQL backup) is created or shared without approval. This may lead to sensitive data exfiltration if the snapshot is exposed to unauthorised users or shared publicly.

Incident Classification

Category	Details
Incident Type	Data Exposure – Snapshot Abuse
Severity	High to Critical (especially if PII, financial data or secrets are involved)
Priority	Critical
Detection Sources	Cloud Audit Logs, CSPM Alerts, SIEM, Storage Logs, Database Activity Monitoring Tools

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enforce snapshot encryption	Use customer-managed KMS keys and enforce encryption on all snapshots
Restrict snapshot sharing	Apply org-level policies to disallow public or cross-account snapshot sharing
Monitor snapshot creation	Alert on snapshot exports, shares and downloads using CSPM or SIEM
Tag sensitive databases	Classify resources for targeted monitoring and DLP enforcement
Enable Cloud Audit Logging	Ensure all snapshot-related actions (create, share, restore) are logged

2. Detection & Analysis

Step	Action
Alert triggered	Snapshot shared outside of organisation or created at unusual time/user
Investigate snapshot type and target	Determine which DB was snapped and whether the snapshot was shared publicly or to unknown accounts
Review access logs	Check if snapshot has been downloaded, restored or accessed
Correlate with user identity	Investigate the IAM identity or service account that performed the action

MITRE ATT&CK mapping	T1530 (Data from Cloud Storage), T1005 (Data from Local System), T1078.004 (Cloud Accounts), T1048 (Exfiltration Over Alternative Protocol)

3. Containment

Step	Action
Revoke access to shared snapshot	Remove sharing or make the snapshot private via console or CLI
Suspend offending account	Temporarily disable user or service account responsible
Disable download access	If snapshot was copied to an external S3 bucket, GCS or Azure blob, revoke access
Alert compliance and legal teams	Especially if data subject to regulatory protection was involved

4. Eradication

Step	Action
Delete unauthorised snapshots	Remove rogue or unapproved copies
Rotate affected credentials	If secrets were part of the database content or if service account was abused
Audit IAM permissions	Ensure snapshot creation and sharing are tightly scoped to trusted roles only
Review cross-account trust settings	Remove any risky or unmonitored permissions that allow sharing outside the organisation

5. Recovery

Step	Action
Restore trusted backup procedures	Reinstate verified, encrypted and access-controlled backups
Revalidate database and snapshot integrity	Ensure no tampering or backdoors were introduced via restore processes
Resume database operations	Once the environment and backups are secure and validated
Increase logging around critical databases	Apply heightened surveillance for a defined observation period

6. Lessons Learned & Reporting

Step	Action

Document incident scope	Timeline, data types exposed, accounts involved and resolution steps
Update detection rules	Add alerts for snapshot sharing or copying across boundaries
Train DB admins and developers	On secure snapshot procedures and IAM governance
Report to authorities	If breach involves personal, financial or government-regulated data
Improve cloud guardrails	Use Infrastructure-as-Code scanning and policy-as-code for future prevention

Tools Typically Involved

- Cloud-native logs (e.g., AWS CloudTrail, Azure Activity Logs, GCP Admin Audit Logs)
- CSPM (e.g., Wiz, Prisma Cloud, Microsoft Defender for Cloud)
- SIEM (e.g., Sentinel, Splunk)
- DLP tools (e.g., Microsoft Purview, Forcepoint DLP)
- SOAR platform for automated response
- Database Activity Monitoring (e.g., Imperva, Guardicore, native platform logging)

Success Metrics

Metric	Target
Detection Time	<10 minutes from snapshot creation or share
Public Access Removal Time	<30 minutes from confirmation
Snapshot Deletion Time	<1 hour for unauthorised snapshots
IAM Policy Audit Completion	100% of affected environments within 48 hours
Compliance Notification Deadline	Within 72 hours or as per regulatory requirements

SOC Incident Response Playbook 28: Container Breakout Attempt

Scenario

An attacker gains access to a container and attempts to escape the isolated environment to interact with the host operating system, escalate privileges or compromise other containers, pods or underlying infrastructure.

Incident Classification

Category	Details
Incident Type	Container Runtime Security – Escape Attempt
Severity	Critical (especially if host access or privilege escalation is achieved)
Priority	Critical
Detection Sources	Runtime Security Tools, SIEM, EDR, Kubernetes Audit Logs, Falco Rules, Container Logs

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Implement runtime security	Use tools like Falco, Aqua, Sysdig or Wiz Runtime for real-time detection
Enable Kubernetes and Docker audit logging	Capture container activity and host-level access attempts
Harden container images	Use minimal base images, scan for vulnerabilities and remove unnecessary tools (e.g., curl, bash)
Apply Pod Security Policies / OPA	Prevent privilege escalation, host mounts and container privilege mode
Monitor inter-container traffic	Enable east-west container traffic monitoring using NDR or eBPF-based tools

2. Detection & Analysis

Step	Action
Alert triggered	Attempt to access host filesystem (/proc, /root), escalate privileges or spawn unexpected binaries
Review container activity	Check for nsenter, chroot, mount, apt, wget or suspicious execs
Identify affected container and node	Determine pod name, namespace and underlying host VM/node

Analyse attacker behaviour	Was this a misconfiguration exploit (e.g., privileged: true) or remote code execution?
MITRE ATT&CK mapping	T1611 (Escape to Host), T1059 (Command and Scripting Interpreter), T1203 (Exploitation for Privilege Escalation)

3. Containment

Step	Action
Stop compromised pod or container	Forcefully delete or isolate the instance immediately
Isolate affected node	Remove the node from cluster scheduling and limit communication
Suspend service account access	Especially if the container had access to Kubernetes API or secrets
Snapshot affected container	If forensics is required, preserve memory and logs where possible

4. Eradication

Step	Action
Investigate the root cause	Vulnerable image, over-permissive configuration or exposed interface
Patch vulnerable workloads	Rebuild and redeploy affected pods with fixed configuration or image
Rotate secrets and credentials	Especially if stored in environment variables, configMaps or volumes
Remove backdoors or malicious tools	Search for rogue binaries, cron jobs or injected scripts in containers or host

5. Recovery

Step	Action
Rebuild workloads from trusted images	Use CI/CD pipelines with image signing and scanning
Reinstate node after sanitisation	Only after full forensic validation of the host system
Resume services	Reintroduce pods gradually and monitor closely for recurrence
Enhance monitoring for affected namespace or deployment	Apply anomaly detection for future deviations

6. Lessons Learned & Reporting

Step	Action
Conduct a post-mortem analysis	Document breakout vector, timeline and exposed assets
Improve container security policies	Enforce strict resource isolation and prevent reuse of affected patterns
Educate DevOps teams	On secure container configuration, minimal permissions and runtime risks
Report if required	If data exposure or host compromise occurred, notify regulatory bodies or clients
Update runbooks and response workflows	Add new rules and controls based on this attack scenario

Tools Typically Involved

- Runtime Security (e.g., Falco, Sysdig Secure, Aqua, Prisma Cloud Compute, Wiz)
- Kubernetes Audit Logs and RBAC logs
- SIEM (e.g., Sentinel, Splunk, QRadar)
- EDR (for host nodes, e.g., CrowdStrike, Cortex XDR)
- NDR (for container traffic visibility)
- Image Scanning (e.g., Trivy, Clair, Anchore)

Success Metrics

Metric	Target
Detection Time	<5 minutes from breakout attempt
Pod Termination Time	<10 minutes from alert
Root Cause Fix Time	<48 hours
Node Revalidation Completion	Within 24 hours post-removal
Image Hardening Review	100% of similar deployments audited within 3 business days

SOC Incident Response Playbook 29: Shadow IT SaaS Usage & Data Exposure

Scenario

An employee or team uses an unapproved SaaS application (e.g., personal Google Drive, Dropbox, Notion, ChatGPT) for work-related purposes, transferring corporate data without security oversight. This can result in unauthorised data exposure, regulatory breaches or insider misuse.

Incident Classification

Category	Details
Incident Type	Policy Violation – Unauthorised SaaS Usage
Severity	Medium to High (depending on the type and sensitivity of data involved)
Priority	High
Detection Sources	CASB, DLP, Proxy Logs, SIEM, Endpoint Agents, Shadow IT Discovery Tools, Employee Reports

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Implement CASB platform	Discover and monitor all SaaS usage beyond sanctioned apps
Set SaaS usage policies	Clearly define approved vs. unapproved services
Apply DLP on endpoints and cloud	Detect sensitive file uploads or clipboard transfers
Integrate proxy/firewall logs	Track SaaS usage by domain/IP and user
Train users on data handling and shadow IT risks	Promote awareness of compliance and approved tools

2. Detection & Analysis

Step	Action
Alert triggered	CASB or proxy detects unsanctioned SaaS usage or DLP flags sensitive file transfer
Identify the user and device	Match to IP, user ID and machine used for the transfer
Analyse the data shared	Determine if documents contained PII, PHI, customer info or internal IP

Investigate SaaS app risk profile	Evaluate whether the app has poor security practices or terms of service violations
MITRE ATT&CK mapping	T1087.003 (Cloud Service Enumeration), T1537 (Transfer Data to Cloud Account), T1213 (Data from Information Repositories)

3. Containment

Step	Action
Block access to the SaaS app	Use CASB, firewall or proxy rules to prevent further use
Suspend user's internet or cloud access	Temporarily if data exposure is severe or continued use is suspected
Notify user and manager	Conduct initial investigation interview if necessary
Prevent download/export of shared data	Remove permissions or delete from the third-party app if possible

4. Eradication

Step	Action
Remove company data from unapproved platforms	Where feasible, contact the vendor or request user deletion
Revoke SaaS OAuth permissions	From user or enterprise accounts integrated with unapproved services
Tighten app controls	Configure CASB to auto-block newly discovered unapproved apps in high-risk categories
Remove access to shared data from external parties	If data was shared via link or collaboration features

5. Recovery

Step	Action
Reinstate user access under monitoring	After risk is remediated and policy is acknowledged
Monitor future SaaS usage	Apply stricter controls and alerts on repeat violations
Validate that no further data copies exist	Search endpoints and cloud storage for duplicates
Implement formal app request workflows	Make it easier for users to request approval of new tools securely

6. Lessons Learned & Reporting

Step	Action

Perform user and data risk review	Understand business reasons for Shadow IT use and sensitivity of data involved
Update SaaS control policies	Include newly discovered apps in the unapproved/blocked list or formally review them
Educate users	Add targeted training or post-incident briefings
Report to compliance/management	If regulatory data was exposed or customer confidentiality was breached
Review and update DLP/CASB configurations	Based on gaps that allowed this usage to go undetected

Tools Typically Involved

- CASB (e.g., Microsoft Defender for Cloud Apps, Netskope, Skyhigh Security)
- DLP (e.g., Forcepoint, Microsoft Purview, Symantec DLP)
- SIEM (e.g., Splunk, Sentinel)
- Web Proxies and NGFW (e.g., Zscaler, Palo Alto, Fortinet)
- Endpoint Monitoring Tools (e.g., CrowdStrike, Tanium)
- SaaS Access Governance Tools (e.g., BetterCloud, DoControl)

Success Metrics

Metric	Target
Detection Time	<5 minutes from data upload or SaaS access
SaaS Access Block Time	<15 minutes from alert
Data Removal Completion	Within 24 hours for public or third-party exposure
User Education Completion	100% of involved users re-briefed within 3 business days
Policy Review Update	Within 7 days to incorporate lessons learned

SOC Incident Response Playbook 30: API Key Leakage via Public GitHub Repositories

Scenario

A developer accidentally commits and pushes API keys, cloud credentials or other secrets to a public GitHub repository. These secrets can be harvested by attackers (including bots that monitor GitHub) and used to access critical systems, cloud resources or third-party APIs.

Incident Classification

Category	Details
Incident Type	Credential Exposure – Source Code Leak
Severity	Critical (especially for cloud or production credentials)
Priority	Critical
Detection Sources	GitHub Secret Scanning Alerts, TruffleHog, Gitleaks, Cloud Provider Alerts, Bug Bounty Reports

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enable GitHub secret scanning	Activate GitHub Advanced Security or third-party scanners (e.g., Gitleaks, TruffleHog)
Use pre-commit hooks	Integrate secret detection tools to prevent commits with secrets
Rotate secrets regularly	Use vaults (e.g., HashiCorp Vault, AWS Secrets Manager) and enforce key expiry policies
Educate developers	On secure coding practices and the dangers of pushing secrets
Monitor public GitHub repos	Use threat intel and GitHub APIs to continuously scan for exposed org assets

2. Detection & Analysis

Step	Action
Secret detected in commit	Alert from GitHub, internal scan or external report
Identify secret type and scope	API key, cloud access key, DB password, token, etc.
Correlate with owning user/repo	Find the developer who committed it and determine if repo is public

Analyse exposure window	How long was it public? Any signs of usage (e.g., logs, rate limits breached)?
MITRE ATT&CK mapping	T1552.001 (Credentials in Files), T1087 (Account Discovery), T1528 (Steal Access Token)

3. Containment

Step	Action
Immediately revoke the exposed secret	Deactivate API key, token or credential from provider
Restrict affected services	If the secret granted broad access, disable dependent integrations or pipelines
Alert developer and security team	Notify for immediate validation and remediation
Remove sensitive commit from history	Use tools like git filter-branch, BFG or git rebase to scrub secrets

4. Eradication

Step	Action
Replace exposed keys with new ones	Generate and distribute new keys securely via vault or secret manager
Audit cloud/API logs	Look for signs of abuse using the leaked key during its exposure period
Validate GitHub repo hygiene	Review commit history and remove any other sensitive information
Block repo or mark private	If it still contains risks or needs re-evaluation

5. Recovery

Step	Action
Restore service access using new secrets	Confirm integrations and pipelines are working with rotated credentials
Re-enable affected users or systems	Once no unauthorised access is detected
Monitor for abuse	Set alerts on any suspicious use of revoked credentials across services
Document impact and confirm clean repo state	Ensure dev teams comply with updated policies

6. Lessons Learned & Reporting

Step	Action

Conduct RCA	Why and how was the secret exposed? Human error, misconfigured Git tools, no scanning?
Improve pre-commit pipelines	Integrate Git hooks or CI/CD scanners to block such mistakes earlier
Train development teams	On secure software development lifecycle (SSDLC) and version control hygiene
Update incident documentation	Include key timelines, revoked credentials and mitigation efforts
Report if necessary	For breaches involving regulated data or third-party systems (GDPR, PDPA, PCI DSS, etc.)

Tools Typically Involved

- GitHub Advanced Security (secret scanning)
- TruffleHog, Gitleaks, GitRob
- HashiCorp Vault, AWS Secrets Manager, Azure Key Vault
- SIEM (e.g., Sentinel, Splunk) and threat detection systems
- Version control auditing (e.g., Git log parsing, commit reviewers)

Success Metrics

Metric	Target
Revocation Time	<15 minutes from detection
Commit Cleanup Time	<1 hour for critical secrets
Secret Replacement & Reintegration	<4 hours for production use
Exposure Window Analysis Completion	Within 24 hours
Developer Acknowledgement of Policy	100% of involved devs within 2 business days

SOC Incident Response Playbook 31: Unauthorised Access to CI/CD Secrets

Scenario

Secrets (such as cloud credentials, API tokens, SSH keys or environment variables) stored in CI/CD tools (e.g., Jenkins, GitHub Actions, GitLab CI, Azure DevOps) are accessed by an unauthorised party—either through misconfiguration, leaked logs, compromised runners or malicious pull requests.

Incident Classification

Category	Details
Incident Type	Credential Exposure – CI/CD Security Breach
Severity	Critical (especially for production or cloud infrastructure access)
Priority	Critical
Detection Sources	SIEM, Secret Scanning Tools, CI/CD Audit Logs, CSPM, Threat Intelligence, Bug Bounty Reports

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Store secrets in vaults	Use Secret Managers (e.g., HashiCorp Vault, AWS Secrets Manager) instead of plaintext CI/CD variables
Apply least privilege	Limit CI jobs and service accounts to only required permissions
Monitor CI/CD audit logs	Enable logging on runners, workflows and secret access
Scan repositories and pipelines	Use tools like TruffleHog, Gitleaks or GitHub Secret Scanning to detect exposed secrets
Secure CI/CD runners	Isolate, update and protect runners from tampering or privilege escalation

2. Detection & Analysis

Step	Action
Alert triggered	Access or exfiltration of secrets from pipeline logs or vault
Identify accessed secrets	What secrets were exposed and what systems do they control?
Review CI job and trigger source	Determine if this was a malicious job, PR abuse or insider misuse
Analyse logs and runtime metadata	Inspect job logs, runner behaviour, environment variables and external callbacks

MITRE ATT&CK mapping	T1552.004 (Credentials in CI/CD), T1529 (System Shutdown/Reboot to Disrupt), T1078.004 (Cloud Credentials Abuse), T1059 (Script Execution)

3. Containment

Step	Action
Revoke exposed secrets	Immediately disable or rotate credentials, tokens and keys
Disable CI jobs or pipelines	Especially those that were abused or scheduled to rerun
Lock down affected repositories or runners	Prevent further job execution and isolate suspicious PRs or commits
Notify affected platform and security teams	Alert developers, DevOps and SecOps

4. Eradication

Step	Action
Delete or clean vulnerable jobs or workflows	Remove embedded secrets or log outputs containing them
Rebuild and secure runners	Apply security updates, audit for rootkits or persistence and redeploy
Tighten secret handling	Use environment-level injection via secure vaults instead of hardcoded secrets
Update access control lists	Remove over-permissive roles or default trust to external contributors

5. Recovery

Step	Action
Rotate secrets in affected systems	Cloud accounts, APIs, databases, etc.
Resume CI/CD operations	After full validation and hardening of build jobs, runners and configs
Apply monitoring to rebuilt environments	Include anomaly detection on secret use and build behaviour
Restore legitimate PRs and code commits	Once verified as safe and authorised

6. Lessons Learned & Reporting

Step	Action

Conduct RCA	Identify root cause—vault misconfig, insider threat, leaked logs or misused permissions
Update CI/CD security policies	Enforce PR approval workflows, job restrictions and vault-only secrets
Train DevSecOps teams	On safe secret management and pipeline hygiene
Report to external stakeholders	If exposed secrets impacted clients, customers or regulated data
Document findings in runbook	Include indicators of compromise, timelines and detection gaps

Tools Typically Involved

- CI/CD Platforms (e.g., GitHub Actions, GitLab CI, Jenkins, Azure DevOps)
- Secret Management Systems (e.g., AWS Secrets Manager, Vault)
- Secret Scanning Tools (e.g., TruffleHog, Gitleaks, GitGuardian)
- SIEM (e.g., Splunk, Sentinel)
- SOAR (for response automation)
- CSPM (for cloud environment hardening and secret detection)
- Endpoint Monitoring (if runners or developers were targeted)

Success Metrics

Metric	Target
Secret Revocation Time	<15 minutes from detection
CI Job Suspension Time	<30 minutes
Impacted Secrets Replacement Time	<4 hours
Secure Runner Redeployment Time	<24 hours
Developer Training Completion	100% of relevant team within 3 business days

SOC Incident Response Playbook 32: Zero-Day Exploitation in Third-Party Libraries

Scenario

A critical vulnerability is disclosed (or actively exploited in the wild) in a third-party library or framework (e.g., Log4j, OpenSSL, Apache Struts, glibc) used within your environment. Attackers may exploit this zero-day before a patch or mitigation is available, often through remote code execution (RCE), information disclosure or privilege escalation.

Incident Classification

Category	Details
Incident Type	Zero-Day Exploitation – Supply Chain / Library
Severity	Critical (depending on exposure and exploitability)
Priority	Critical
Detection Sources	Threat Intelligence Feeds, Vendor Advisories, SIEM, EDR/XDR, Network Detection, Bug Bounty Reports

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Maintain SBOM (Software Bill of Materials)	Use tools like CycloneDX, Syft, Anchore to track dependencies in use
Subscribe to threat intelligence & CVE feeds	Ensure security team gets early alerts (e.g., CISA KEV, NVD, GitHub Security Advisories)
Tag critical workloads using affected libraries	Enable targeted logging and monitoring when an alert is raised
Establish emergency patch & mitigation process	Prepare for out-of-cycle updates and dev/test rollout plans
Harden external attack surfaces	Block unnecessary exposure (e.g., admin panels, debugging endpoints)

2. Detection & Analysis

Step	Action
Alert triggered	Public disclosure of a critical zero-day with working PoC or active exploitation reports
Identify affected systems	Use SBOM or asset management to list systems using the vulnerable library
Assess exposure	Determine if services are externally accessible or internally reachable

Monitor for IOCs	Collect indicators such as process anomalies, network callbacks, abnormal log entries
MITRE ATT&CK mapping	T1190 (Exploit Public-Facing Application), T1210 (Exploitation of Remote Services), T1588.006 (Vulnerability Disclosure)

3. Containment

Step	Action
Isolate exposed services	Block internet access to vulnerable applications if no patch is available
Deploy WAF/IPS virtual patches	Block known exploit patterns using signatures or payload filtering
Remove or disable plugins/modules	Temporarily disable functionality if it reduces risk without affecting operations critically
Notify internal stakeholders	Coordinate between security, dev and infra teams to begin emergency mitigation

4. Eradication

Step	Action
Apply vendor patch or upgrade	As soon as it's available; validate in staging before production rollout
Replace affected libraries	If patching is not feasible, switch to safe versions or alternatives
Remove dropped payloads or backdoors	From compromised hosts if exploitation already occurred
Clean temporary mitigations	Once systems are patched and confirmed safe

5. Recovery

Step	Action
Resume full application operations	After validation of patched environments
Conduct full forensics	Determine if systems were exploited before patching and whether data was accessed
Increase logging temporarily	Maintain enhanced visibility around patched systems for 7–14 days
Verify third-party components	Ensure vendors and partners also patch or mitigate the zero-day risk

6. Lessons Learned & Reporting

Step	Action

Document timeline and impact	From disclosure to mitigation and any confirmed incidents
Update vulnerability management policies	Include response to emerging threats and zero-days
Train dev and security teams	On monitoring dependencies and using SBOMs effectively
Report to regulators/customers	If breach or risk to sensitive data occurred (e.g., under GDPR, PDPA, PCI DSS)
Conduct tabletop exercises post-incident	Simulate similar scenarios to test readiness

Tools Typically Involved

- SBOM & Dependency Scanners (e.g., Anchore, Snyk, OWASP Dependency-Check)
- Threat Intel Platforms (e.g., MISP, Recorded Future, CISA KEV)
- SIEM (e.g., Sentinel, Splunk)
- EDR/XDR (e.g., CrowdStrike, Cortex XDR)
- WAF/IPS (e.g., Cloudflare, AWS WAF, Palo Alto)
- SOAR for automated playbook execution

Success Metrics

Metric	Target
Initial Triage Time	<30 minutes from public disclosure
Exposure Mapping Time	<2 hours to identify affected systems
Mitigation Deployment	Within 12–24 hours
Patch Completion	<48 hours for critical systems
Post-Incident Report	Within 72 hours

SOC Incident Response Playbook 33: Abuse of Stolen Session Tokens in SaaS Platforms

Scenario

An attacker gains access to a valid session token (e.g., via XSS, phishing, malware or token theft from endpoints) and uses it to impersonate a legitimate user on a SaaS platform (e.g., Microsoft 365, Google Workspace, Salesforce, Slack). This allows access without triggering MFA or login anomaly alerts.

Incident Classification

Category	Details
Incident Type	Account Hijack – Session Token Abuse
Severity	High to Critical (based on data access and privilege level)
Priority	Critical
Detection Sources	CASB, SIEM, EDR, SaaS Audit Logs, User Reports, Threat Intelligence Feeds

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Enable session management logs	In all SaaS platforms to capture token reuse or suspicious IP logins
Deploy CASB and SaaS Security tools	To monitor user behaviour, token anomalies and session reuse
Use Conditional Access policies	Based on geolocation, device trust and user risk scores
Educate users on phishing and token theft	Including how session tokens can be abused
Integrate endpoint protection	To prevent token theft via malware

2. Detection & Analysis

Step	Action
Alert triggered	Unusual session activity (e.g., login from known IP but unusual behaviour or location)
Check for duplicate sessions	Same token reused from different IPs or geolocations
Review recent user activity	Look for data downloads, permission changes, new app integrations

Analyse endpoint logs	Identify malware or tools (e.g., RedLine, Vidar) that may have extracted tokens
MITRE ATT&CK mapping	T1539 (Steal Web Session Cookie), T1078 (Valid Accounts), T1185 (Browser Session Hijacking)

3. Containment

Step	Action
Revoke all active sessions	Force logout for the affected user across all devices and apps
Disable user account temporarily	If attacker activity is ongoing or damage is high
Block attacker IPs or devices	At the SaaS provider, CASB or firewall level
Notify user and support team	Inform user to reset passwords and validate MFA devices

4. Eradication

Step	Action
Scan endpoint for malware	Ensure no token stealer is still active on the user's device
Rotate any exposed credentials or tokens	For linked applications or integrations
Review session storage practices	Ensure session tokens are not stored in plaintext or improperly cached
Strengthen SaaS login policies	Enforce re-authentication for sensitive actions or high-risk logins

5. Recovery

Step	Action
Re-enable user access with strict monitoring	After ensuring the endpoint is clean and MFA is re-enforced
Monitor user activity closely	Apply alerts on behavioural deviation or abnormal downloads
Educate user on signs of session hijacking	Reinforce best practices for session security
Conduct internal checks	To ensure no lateral movement or privilege abuse occurred

6. Lessons Learned & Reporting

Step	Action

Complete RCA	Determine how token was stolen: malware, phishing, browser sync, etc.
Improve session hygiene policies	Reduce session duration, prevent reuse across devices or geos
Train employees	Regularly on SaaS risks and session awareness
Document findings	In IR logs and lessons learned report
Notify third parties or regulators	If sensitive data was accessed or shared externally

Tools Typically Involved

- CASB (e.g., Microsoft Defender for Cloud Apps, Netskope, Lookout)
- SaaS Security Posture Management (e.g., Obsidian, AppOmni)
- SIEM (e.g., Sentinel, Splunk)
- Endpoint Detection and Response (e.g., CrowdStrike, Cortex XDR)
- Identity Providers (e.g., Okta, Azure AD, Google Workspace)
- Browser Security Tools (e.g., LayerX, Seraphic)

Success Metrics

Metric	Target
Session Revocation Time	<10 minutes from detection
Endpoint Validation Time	<1 hour
Post-incident MFA Reinforcement	100% completion within 24 hours
SaaS Behaviour Monitoring Duration	≥ 14 days post-incident
RCA and Reporting Completion	Within 3 business days

SOC Incident Response Playbook 34: Cloud-Native Ransomware in Object Storage

Scenario

An attacker gains access to cloud object storage (e.g., Amazon S3, Azure Blob Storage, Google Cloud Storage) and performs malicious actions such as encrypting files, altering permissions, deleting backups or placing ransom notes — without deploying ransomware binaries, purely using APIs or SDKs.

Incident Classification

Category	Details
Incident Type	Ransomware – Object Storage (Cloud-native)
Severity	Critical
Priority	Critical
Detection Sources	CSPM Alerts, SIEM, Cloud Storage Logs, CASB, CloudTrail, Access Analyzer, User Reports

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Apply strict IAM policies	Use least privilege principles for storage access (read/write/delete)
Enable versioning and MFA delete	In S3, GCS or Azure Blob to retain file history and block unauthorised deletes
Enable logging and monitoring	CloudTrail, Storage Access Logs and alerting on anomalous activity
Set up anomaly detection for storage	Sudden write/delete bursts, non-human behaviour, large-scale file access
Test restoration processes	Ensure backups and snapshots can be restored quickly and reliably

2. Detection & Analysis

Step	Action
Alert triggered	Unusual storage activity: mass object overwrites/deletions, access from unknown IPs or ransom note files
Correlate with IAM activity	Identify user or service account responsible for the storage operations
Determine extent of impact	Number of buckets/containers, types of data affected and presence of backups

Search for IOCs	Files renamed/encrypted, ransom notes (e.g., README_TO_RESTORE.txt), strange file extensions
MITRE ATT&CK mapping	T1485 (Data Destruction), T1486 (Data Encrypted for Impact), T1531 (Account Access Removal)

3. Containment

Step	Action
Revoke affected IAM roles or keys	Immediately disable access for the user or application responsible
Block malicious IP addresses	Using CSP firewall rules or geofencing
Lock down storage buckets	Remove public access and apply restrictive ACLs and policies
Alert cloud security and incident response team	Trigger emergency remediation plan

4. Eradication

Step	Action
Audit all storage policies and access logs	Ensure no other backdoors or malicious users remain active
Rotate access credentials	For all cloud accounts and applications involved
Remove attacker implants or files	Delete ransom notes, trojaned files or API logs left behind
Patch external entry points	If exploitation came via web app or exposed access key

5. Recovery

Step	Action
Restore from backup or object versioning	Use last known good versions or automated snapshots
Verify data integrity	Check that restored files are complete and unaltered
Resume business services	After storage and applications are validated safe
Increase logging and detection thresholds	For affected buckets and linked identities

6. Lessons Learned & Reporting

Step	Action
Conduct RCA	Trace attack vector, methods used and timeline

Improve monitoring and access controls	Enforce stricter policies on object storage access and anomaly detection
Update runbooks and alert rules	Include new attack patterns and prevention guidance
Train DevOps and cloud admins	On secure storage configurations and rapid response techniques
Report incident	To regulatory bodies and customers if sensitive data was affected

Tools Typically Involved

- Cloud Audit Logs (e.g., AWS CloudTrail, Azure Activity Logs, GCP Admin Logs)
- CSPM Tools (e.g., Wiz, Prisma Cloud, Microsoft Defender for Cloud)
- SIEM (e.g., Sentinel, Splunk, Chronicle)
- CASB (e.g., Netskope, Defender for Cloud Apps)
- Backup and DR Tools (e.g., AWS Backup, Azure Site Recovery, GCP Snapshots)
- SOAR (for auto-remediation)

Success Metrics

Metric	Target
Detection Time	<5 minutes from mass storage modification
IAM Key Revocation Time	<10 minutes from detection
Data Restoration Time	<6 hours (for critical data)
Access Policy Review Time	100% of affected buckets reviewed within 24 hours
RCA and Remediation Report	Completed within 72 hours

SOC Incident Response Playbook 35: Malicious Insider Staging Data in the Cloud

Scenario

A trusted user within the organisation abuses their access to sensitive data (e.g., PII, source code, financials) and begins uploading it to unapproved cloud platforms (e.g., personal Google Drive, Dropbox, Mega, OneDrive) for exfiltration. This may precede resignation, whistleblowing or corporate espionage.

Incident Classification

Category	Details
Incident Type	Insider Threat – Data Staging / Exfiltration
Severity	High to Critical (based on data sensitivity and exposure level)
Priority	Critical
Detection Sources	DLP, CASB, SIEM, Proxy Logs, Endpoint Agents, User Reports, Insider Threat Programs

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Implement insider threat monitoring	Use UEBA, CASB and DLP with behavioural baselines
Enforce approved cloud storage policy	Block unsanctioned SaaS uploads using CASB or firewall rules
Monitor large file transfers and compressions	Use endpoint and proxy rules to detect mass zipping or uploads
Train employees on data handling policies	Reinforce disciplinary and legal consequences of data misuse
Use tagging for sensitive files	Classify documents to apply targeted monitoring

2. Detection & Analysis

Step	Action
Alert triggered	DLP or CASB detects file upload to unapproved cloud service
Investigate user behaviour	Look for signs of resignation, policy violations or abnormal working hours
Correlate data access and transfer	Identify which files were accessed, downloaded, zipped or uploaded
Determine target cloud storage	Personal Google Drive, Dropbox, Mega, iCloud, etc.

MITRE ATT&CK mapping	T1537 (Transfer Data to Cloud Account), T1081 (Credentials from Password Stores), T1567.002 (Exfiltration to Cloud Storage)

3. Containment

Step	Action
Block further access to external storage	Enforce cloud app control through CASB or proxies
Suspend or limit user account	Temporarily if behaviour is clearly malicious or data loss is ongoing
Isolate user device	If malware or credential theft is also suspected
Preserve session and file logs	For forensic analysis and legal use

4. Eradication

Step	Action
Remove access to sensitive systems	Revoke elevated privileges and remove from sensitive groups or shares
Retrieve or delete staged data	If stored on corporate device or retrievable from personal cloud (with legal support)
Reset credentials and tokens	Especially if user had API access or was using automation tools
Disable shadow cloud accounts	Prevent further access or data sync from corporate systems

5. Recovery

Step	Action
Reassign critical duties	If employee was in a privileged role or part of a handover
Monitor for follow-up exfiltration attempts	Use enhanced logging for user accounts or similar profiles
Review audit logs across systems	Ensure no lateral activity or additional data transfers occurred
Resume normal operations	Once incident scope and risk are under control

6. Lessons Learned & Reporting

Step	Action
Conduct full insider threat RCA	Understand motive, opportunity and control weaknesses
Improve insider threat models	Refine UEBA rules and escalation playbooks

Inform HR and Legal	For possible disciplinary action, legal follow-up or prosecution
Review and update DLP/CASB policies	Add newly abused platforms or tactics to detection scope
Notify regulators or clients	If regulated data was exposed or customer confidentiality breached

Tools Typically Involved

- DLP (e.g., Microsoft Purview, Forcepoint, Symantec DLP)
- CASB (e.g., Netskope, Microsoft Defender for Cloud Apps)
- UEBA/Insider Threat Platforms (e.g., Splunk UBA, Exabeam, Varonis)
- SIEM (e.g., Sentinel, Splunk)
- Endpoint Monitoring (e.g., CrowdStrike, Trellix, Tanium)
- Proxy and NGFW (e.g., Zscaler, Palo Alto)
- HR Systems and Legal Support Tools

Success Metrics

Metric	Target
Detection Time	<5 minutes from data upload
Account Restriction Time	<15 minutes from alert
Data Recovery / Containment Time	<24 hours
Forensic Analysis Completion	Within 48 hours
Insider Threat Playbook Update	Within 3 business days

SOC Incident Response Playbook 36: Unauthorised SaaS OAuth Application Integration

Scenario

An employee or attacker grants a third-party application access to a corporate SaaS account using OAuth scopes (e.g., read email, access calendar, read/write files). These applications may exfiltrate data, impersonate users, or maintain persistent access without triggering standard credential or MFA alerts.

Incident Classification

Category	Details
Incident Type	OAuth Abuse – Unauthorised Third-Party App
Severity	High (depending on the scopes granted and data accessed)
Priority	High to Critical
Detection Sources	CASB, SSPM, SaaS Admin Portals, SIEM, Threat Intelligence Feeds

Phases and Actions

1. Preparation (Pre-Incident Setup)

Task	Tool/Action
Restrict app consent policies	Only allow OAuth consent for pre-approved or verified apps
Monitor OAuth activity logs	Use SIEM or SaaS security tools to track app authorisations and scopes
Educate users	About risks of authorising personal or unknown apps in corporate environments
Integrate SSPM/CASP tools	For visibility into authorised applications and risk scoring
Apply conditional access policies	To limit app connections from unmanaged devices

2. Detection & Analysis

Step	Action
Alert triggered	Risky or unapproved OAuth app detected with elevated scopes (e.g., read mail, read drive, send messages)
Identify user and application	Who authorised it, what scopes were granted, and what app was used
Analyse access logs	Check if app accessed sensitive data or performed actions (e.g., sending emails, downloading files)

Review app metadata	Reputation, risk score, domain registration, previous threat reports
MITRE ATT&CK mapping	T1528 (Steal Access Token), T1550.001 (Application Access Token), T1098.003 (External Account)

3. Containment

Step	Action
Revoke app access immediately	Via admin portal (e.g., Azure AD, Google Workspace, Slack admin)
Suspend impacted user account	If malicious behaviour or data leakage is confirmed
Block app domain or API endpoints	Via firewall, CASB, or DNS filter to prevent callback connections
Notify user and security team	Initiate internal investigation and containment measures

4. Eradication

Step	Action
Remove residual access tokens	Revoke all tokens granted to the app and refresh user sessions
Rotate credentials and MFA	If impersonation or token theft is suspected
Conduct full data access review	Determine what the app had access to and if data was exfiltrated
Update OAuth policy	Add the app to a blocklist or blacklist category in SSPM or CASB

5. Recovery

Step	Action
Reinstate user access with monitoring	Ensure user awareness and endpoint clean-up if malware is linked
Apply stricter app review process	Require internal approval for all new app integrations
Monitor for recurrence	Create detections for similar app authorisation patterns or behaviours
Validate SaaS logs and alerts	Ensure full visibility of high-risk OAuth events

6. Lessons Learned & Reporting

Step	Action

Complete root cause analysis	Why was the app authorised? Was it phishing, ignorance, or bypassed control?
Improve user training	On secure SaaS usage and application access warnings
Strengthen OAuth governance	Integrate SSPM and automate risk-based app approval/revocation
Document the incident	For compliance, audit trails, and policy updates
Notify affected parties or regulators	If customer data or sensitive records were accessed

Tools Typically Involved

- SSPM/CASP (e.g., AppOmni, Obsidian, Microsoft Defender for Cloud Apps)
- SaaS Admin Portals (e.g., Azure AD, Google Admin Console, Slack Admin)
- SIEM (e.g., Sentinel, Splunk)
- Threat Intelligence (for app risk scoring and reputation)
- SOAR (to automate detection and revocation)
- Endpoint Security (to ensure token origin is clean)

Success Metrics

Metric	Target
App Revocation Time	<15 minutes from detection
User Notification & Session Reset	<30 minutes
Full App Audit & RCA Completion	Within 48 hours
OAuth Policy Update	Within 3 business days
User Awareness Training Completion	100% within 5 business days