Threat Intel Report on

MITRE ATT&CK Framework

Tactics, Techniques, and Procedures

by

Team Cyber4

Mrunmayee Shirodkar: 2070

Vrushali Walve: 2071

Manali Gawade: 2035

Divya Bhogale: 2043

Table of Contents

Summary:

MITRE ATT&CK is a framework that explains **how hackers attack a system step by step**. Instead of guessing, it shows the **common methods attackers really use** in real life.

Think of it like a **map of a cyber-attack**.

First, attackers **collect information** about the target (like emails, websites, or systems).
Then they **enter the system**, usually using phishing emails, weak passwords, or software bugs.
After getting inside, they **run harmful programs**, **stay hidden**, and **try to get more control**.
Finally, they **steal data, damage systems, or stop services**.

MITRE ATT&CK helps **security teams understand these steps** so they can:

- Detect attacks early
- Block hackers
- Protect systems better

➢ Key Framework Overview

The MITRE ATT&CK framework is organized in a clear and structured way to explain how cyber attackers' work.

- It includes **14 tactics**, which represent the **main goals** of an attacker during a cyber-attack.
- The framework explains **216 techniques**, which show the **different methods attackers use** to achieve their goals.
- These techniques are further divided into **475 sub-techniques**, giving **detailed information** about attacker behaviour.
- MITRE ATT&CK also documents **more than 140 Advanced Persistent Threat (APT) groups**, along with the attack methods they commonly use.
- The framework is divided into **three main matrices**:
    - **Enterprise** (for computers and networks)
    - **Mobile** (for smartphones and tablets)
    - **ICS** (for industrial and control systems)

➢ Purpose of This Guide

This guide is prepared to act as a practical reference for people working in cybersecurity. It helps different security roles understand and handle cyber-attacks more effectively.

- It helps Blue Teams to improve their ability to detect and monitor attacks
- It supports Red Teams in creating realistic attack simulations

- It assists Security Architects in checking how strong the security systems are
- It helps Incident Response Teams in analyzing and investigating security incidents
- It supports Risk Management Professionals in identifying and evaluating security risks

➢ Scope of the Document

This document focuses on all 14 tactics of the MITRE ATT&CK Enterprise Matrix. For each tactic, the guide provides:

- A clear explanation of the tactic and its purpose
- Two important attack techniques along with their identification numbers
- Examples of real-world APT groups that use these techniques
- Common methods used to detect such attacks
- Security measures that can be taken to reduce or prevent attack
- Simple visual diagrams showing how attack steps are connected

Overview of the MITRE ATT&CK Framework:

MITRE ATT&CK is a publicly available cybersecurity framework that records and organizes the techniques used by cyber attackers. It is developed by analyzing **actual cyber incidents** and attacker activities. The framework helps in systematically understanding the behavior of attackers when they gain unauthorized access to information systems.

Simply put, MITRE ATT&CK provides a **structured way to study attacker actions** during a cyber-attack.

➢ Main Elements of MITRE ATT&CK

A. Tactics – The Purpose Behind the Attack

Tactics describe the **intention or objective** of an attacker at each stage of a cyber-attack.

1. They explain **why a particular action is performed**
2. The Enterprise Matrix includes **14 different tactical goals**
3. These goals follow a **logical sequence**, matching the stages of an attack lifecycle

B. Techniques – The Methods Used

Techniques explain the **specific approaches** attackers use to carry out their objectives.

1. They represent **how attackers perform an action**
2. The Enterprise Matrix documents **216 attack techniques**
3. Each technique is identified using a **unique reference number**
   (Example: **T1059**)

C. Sub-techniques – Detailed Attack Variations

Sub-techniques provide **more precise descriptions** of each technique.

1. They show **different ways** a technique can be executed
2. The framework includes **475 sub-techniques**
3. Sub-techniques follow a standard numbering format (Example: **T1566.001**)

D. Procedures – The "REAL WORLD ACTIONS"

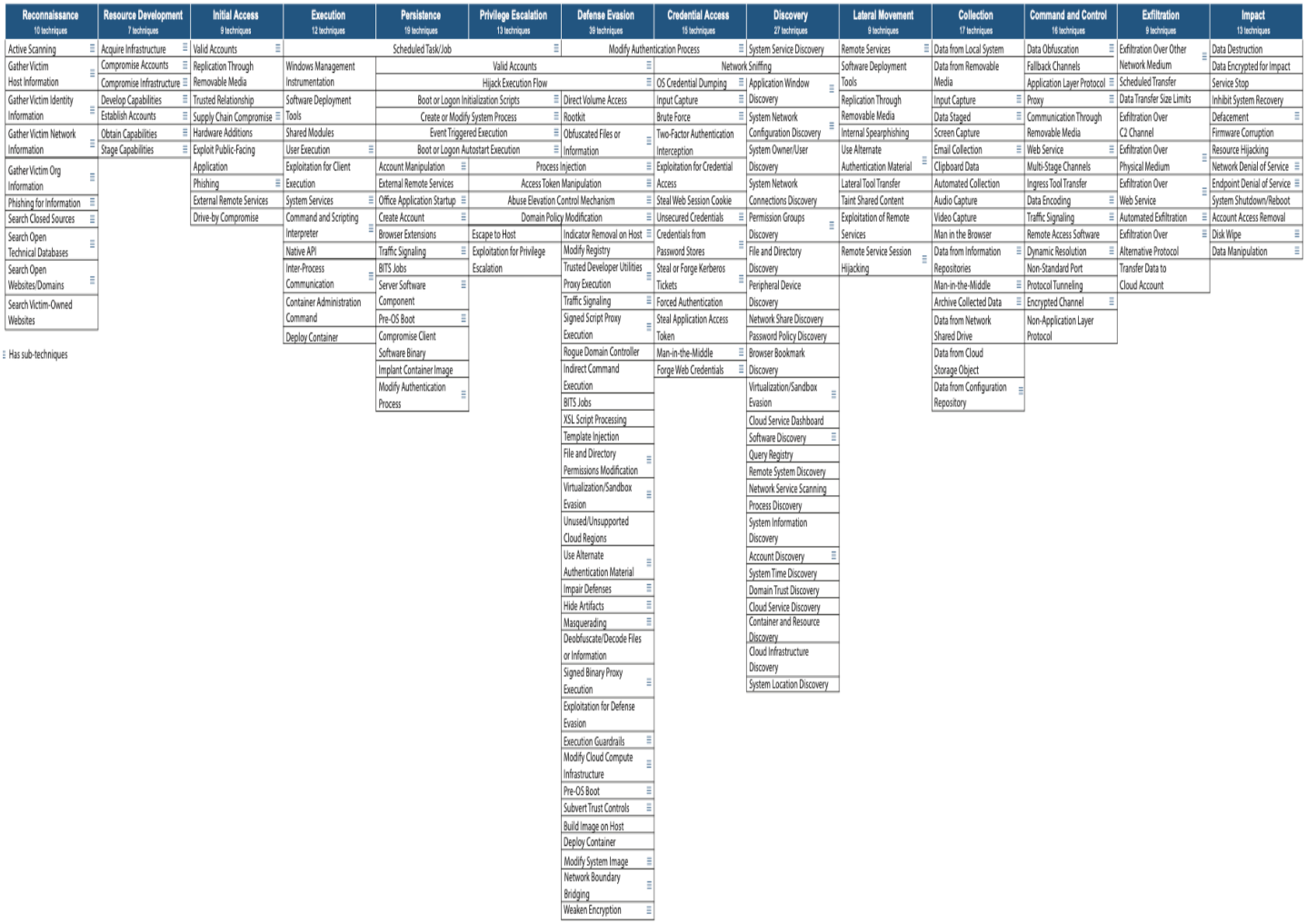
Procedures describe **how cyber attacker**s actually carry out **attacks in real situations.** They show the exact steps followed by specific threat actors while using **techniques** and **sub-techniques**

.

1. Procedures are based on **real attack** cases, not theory
2. They are connected to documented **APT campaigns**
3. They provide **proof-based threat information** gathered from investigations

Area of Use	How It Is Used	Advantage
Security Monitoring	Security alerts and logs are linked to known attacker behaviors	Improves accuracy of attack detection
Attack Simulation	Real attacker techniques are used to simulate cyber attacks	Tests how well security defenses work
Security Review	Existing security controls are compared with ATT&CK techniques	Helps identify gaps in protection
Threat Analysis	Attacker behavior is analyzed using a common framework	Improves understanding of threats
Incident Handling	Attacker actions are categorized during investigations	Reduces response and investigation time
Defense Testing	Detection and response systems are tested against known attacks	Measures overall security effectiveness

Attack Chain Visualization:



MITRE ATT&CK Enterprise Attack Chain Flow

Red vs. Blue Analysis:

Red and Blue Analysis is a method used to evaluate security, strategies, or systems by simulating both offensive (Red) and defensive (Blue) actions. It provides a dual perspective, helping organizations understand vulnerabilities and improve defenses.

Components:

1. **Red Team (Offensive Perspective):**
    - Role: Acts like an attacker or adversary.
    - Purpose: Identify weaknesses, test defenses, and simulate real-world attacks.
    - Activities:
        - Penetration testing
        - Social engineering simulations
        - Exploit development
        - Strategy probing
2. **Blue Team (Defensive Perspective):**
    - Role: Protects systems and responds to threats.
    - Purpose: Detect, prevent, and mitigate attacks.
    - Activities:
        - Monitoring systems for anomalies
        - Incident response
        - Security patching
        - Implementing defensive strategies

Benefits of Red and Blue Analysis:

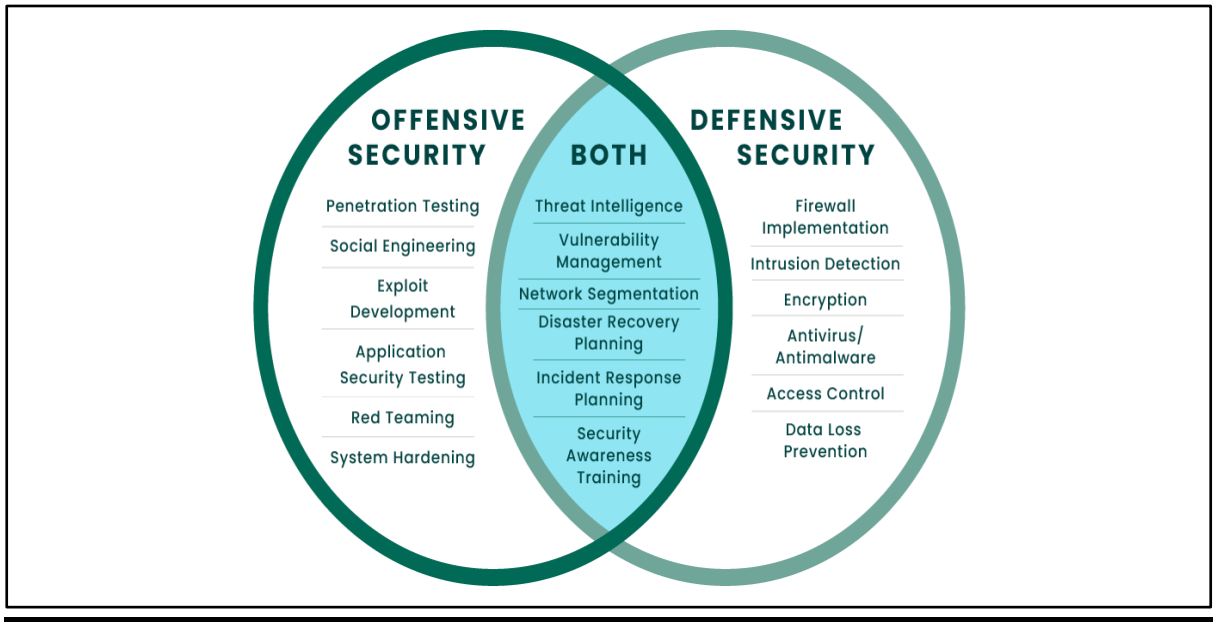
- Provides a realistic assessment of security or strategy.
- Helps identify vulnerabilities before attackers exploit them.
- Improves team readiness and response capabilities.
- Enhances collaboration between offensive and defensive units.

- Supports continuous improvement in security posture.

Applications:

- Cybersecurity
- Military or strategic simulations
- Physical security planning

Visual Representation Idea:



Red Team (Offensive Examples)

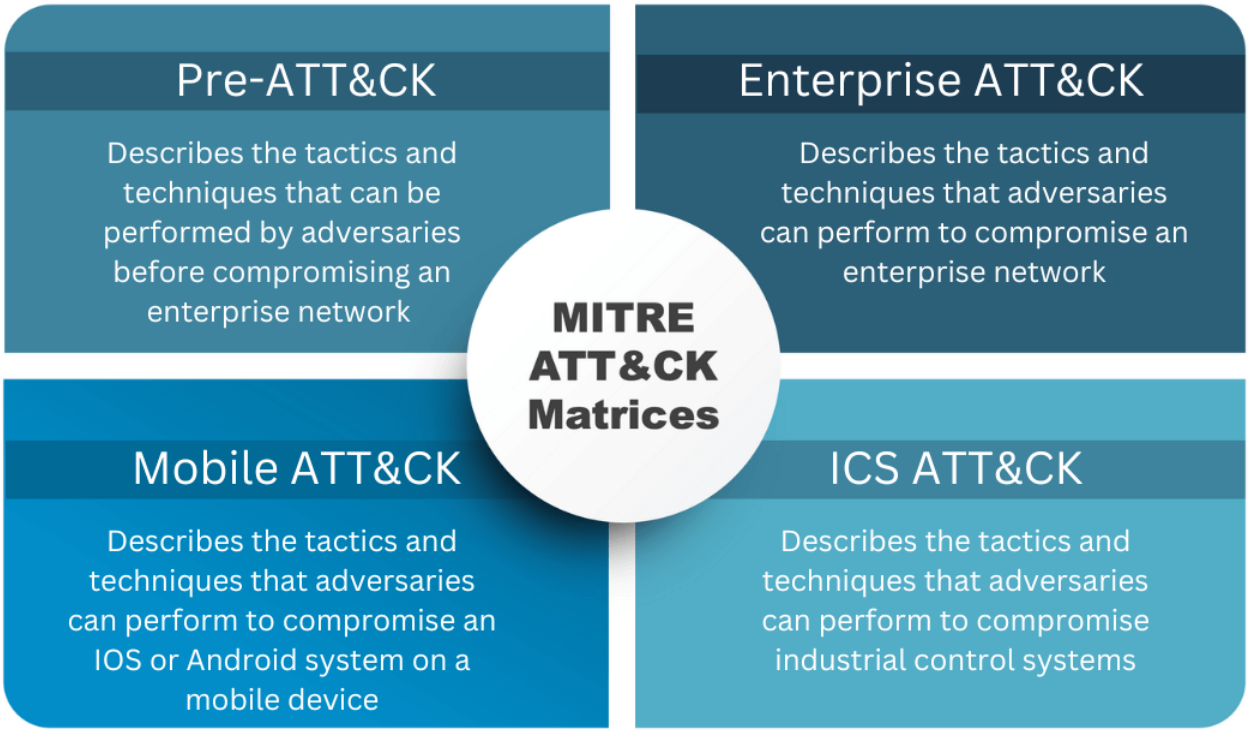
1. **Phishing Campaigns:** Simulating email attacks to trick employees into revealing credentials.
2. **Ransomware Simulation:** Deploying harmless ransomware in a controlled environment to test response.
3. **Penetration Testing:** Attempting to hack web applications or networks to find vulnerabilities.
4. **Social Engineering:** Calling employees to gain sensitive information.

5. **Physical Breach Test:** Trying to enter a secure facility without authorization.

Blue Team (Defensive Examples)

1. **Security Monitoring:** Using SIEM tools to detect unusual login patterns.
2. **Incident Response:** Containing and removing malware after a detected attack.
3. **Firewall & IDS/IPS Management:** Blocking suspicious traffic to prevent breaches.
4. **Patch Management:** Updating systems to close known vulnerabilities**.**
5. **Threat Intelligence Analysis:** Studying hacker methods to anticipate attacks.

Attack Lifecycle Stages:



Pre-ATT&CK focuses on activities adversaries perform *before* gaining access to a target network. This includes reconnaissance, information gathering, target selection, and preparation. Understanding Pre-ATT&CK helps organizations identify early warning signs and strengthen defenses before an actual breach occurs.

Enterprise ATT&CK is the most widely used matrix. It describes techniques attackers use to compromise enterprise environments such as Windows, Linux, macOS, cloud platforms, and networks. It covers the entire attack lifecycle, including initial access, execution, persistence, privilege escalation, lateral movement, and data exfiltration. Blue Teams use this matrix to improve detection rules, incident response, and threat hunting.

Mobile ATT&CK focuses on attacks targeting mobile devices running **Android or iOS**. It documents techniques like malicious apps, SMS phishing, network exploitation, and credential theft. This matrix is essential as mobile devices increasingly access sensitive enterprise data.

ICS ATT&CK is designed for **Industrial Control Systems**, such as power plants, manufacturing systems, and critical infrastructure. It highlights techniques that can disrupt physical processes, cause safety issues, or damage equipment, helping protect operational technology (OT) environments.

Overall, MITRE ATT&CK provides a common language for cybersecurity professionals and plays a vital role in modern Blue Team defense strategies.

The 12 Enterprise Tactics:

1) Initial Access (TA0001)

Tactic Objective: The objective of Initial Access is to gain entry into the target network. Adversaries aim to establish a foothold by exploiting vulnerabilities or tricking users.
This access enables further malicious activities within the network.

Tactic Description: Initial Access involves techniques used by attackers to break into a network for the first time. Common methods include spear phishing emails and exploiting weaknesses in public-facing systems. Attackers may also use stolen or valid credentials to log in remotely. Once access is gained, they try to maintain a foothold for continued operations.
This access may persist over time or be temporary if credentials are changed or access is revoked.

Tactic ID: TA0001
Total Techniques: 3 techniques
Typical Phase: Early Attack phase.
Attack Version: 17 October 2018

Technique 1: Wi-Fi Networks(T1669)

Description:

Attackers can enter a system by connecting to its Wi-Fi network. They may use open Wi-Fi that does not need a password or hack into secured Wi-Fi using stolen login details. To do this, the attacker must be physically nearby so they can find and connect to the wireless network. Once connected, they can try to access devices and data on the network.

Sub-techniques:

No sub-techniques.

Real Example:

Fake Wi-Fi hotspot:

An attacker sets up a fake Wi-Fi hotspot with a name similar to a legitimate network, such as “Office_WiFi_Free.” When users unknowingly connect to it, the attacker captures login credentials and network traffic, which can later be used to access the organization’s internal systems.

Detection Methods:

- Monitor for **unauthorized or rogue Wi-Fi access points** within or near the organization.
- Use **Wireless Intrusion Detection/Prevention Systems (WIDS/WIPS)** to detect suspicious activity.
- Watch for **unknown devices or MAC addresses** connecting to the network.
- Analyze network traffic for **unusual or insecure wireless connections**.
- Enable alerts for **failed or abnormal Wi-Fi authentication attempts**.

Mitigation Strategy:

Mitigation Strategy	Application	Priority
Use strong Wi-Fi encryption (WPA3/WPA2-Enterprise)	All organizational Wi-Fi networks	High
Strong and regularly changed Wi-Fi passwords	Employee and internal wireless networks	High
Disable or restrict open/guest Wi-Fi networks	Public and guest wireless access	Medium
Network segmentation	Wireless to internal network access	High
Deploy WIDS/WIPS	Corporate wireless infrastructure	High
Update and patch wireless devices	Routers, access points, controllers	Medium

Technique 2: Exploit Public-Facing Application(T1190)

Description:

Adversaries may exploit weaknesses in Internet-facing systems to gain initial access to a network. These weaknesses can include software bugs, misconfigurations, or unpatched services in web servers, databases, network devices, or cloud applications. By attacking exposed services such as websites, SSH, SMB, or management interfaces, attackers can execute malicious code or gain unauthorized access. In cloud or container environments, successful exploitation may also lead to access to cloud APIs or the underlying host. Edge devices and appliances are often targeted because they may lack strong security controls, making them easier entry points for attackers.

Sub-techniques:

No sub-techniques.

Real Example:

Equifax data breach (2017).

A well-known real-world example is the **Equifax data breach (2017)**. Attackers exploited a **public-facing Apache Struts web application vulnerability (CVE-2017-5638)** that had not been patched. By sending crafted requests to the exposed web server, the adversaries gained remote code execution, allowing them to access internal systems and steal sensitive data of millions of users.

Detection Methods:

- Monitor **web server logs** for suspicious requests or unusual URLs
- Detect **unexpected crashes or errors** in public-facing applications
- Alert on **unusual inbound traffic patterns** to web servers
- Use **IDS/IPS** to detect known exploit signatures
- Monitor for **new files or web shells** created on web servers
- Track **unauthorized command execution** on application servers

Mitigation Strategy:

Mitigation Strategy	Application	Priority
Regularly patch and update public-facing apps	Web servers, application servers	High
Deploy Web Application Firewall (WAF)	Internet-facing websites and APIs	High
Conduct regular vulnerability scanning	Public-facing systems	High
Restrict admin access and enforce least privilege	Web apps, databases	High
Disable unused services and open ports	Servers, network devices	Medium
Monitor and log all web application activity	SIEM, log management systems	Medium

2) Execution (TA0002)

Tactic Objective: The objective of the Execution tactic is to run malicious code or commands on a target system. Adversaries aim to execute their tools or scripts to achieve operational goals. Successful execution allows attackers to carry out further actions within the system.

Tactic Description: Execution consists of techniques that allow adversaries to run malicious code on a target system. This can include running scripts, exploiting vulnerabilities, or using legitimate applications to execute commands. Methods may involve phishing attachments, remote services, or user-initiated execution.
Successful execution often leads to malware installation, persistence, or further lateral movement. Execution can occur locally on the victim system or remotely through network-based services. Understanding execution methods helps defenders implement controls to prevent or detect malicious activity.

Tactic ID: TA0002
Total Techniques: 2 Techniques
Typical Phase: Compromise phase.
Attack Version: 17 October 2018

Technique 1: Command and Scripting Interpreter(T1059)

Description:

Adversaries may abuse command and script interpreters to execute malicious commands on compromised systems, using built-in tools like PowerShell, Command Prompt, Unix shells, or cross-platform languages such as Python and JavaScript to avoid deploying additional malware. Commands or scripts may be embedded in malicious documents or delivered as secondary payloads from a command-and-control server, allowing attackers to perform actions after initial access. Adversaries can also interact directly with systems through shells or terminals to manually run commands. In many cases, remote services such as SSH or RDP are used to achieve execution on target machines. Because these interpreters are commonly used for legitimate administration, their misuse can blend into normal activity and evade detection.

Sub-techniques:

T1059.001: PowerShell - Adversaries abuse PowerShell to execute malicious commands, perform system discovery, run local or remote code, download and execute payloads (often fileless), and leverage offensive tools or .NET automation interfaces to operate stealthily within Windows environments.

T1059.002: AppleScript - Adversaries may abuse the Windows command shell (cmd) to run malicious commands or batch scripts, either locally or remotely, including interactive control through command-and-control channels.

T1059.003: Windows Command Shell - Adversaries may abuse the Windows command shell (cmd) to execute malicious commands or payloads, either by running single commands, using batch files (.bat or .cmd), or interacting with the system remotely through command-and-control channels.

T1059.004: Unix Shell -Adversaries may abuse Unix shells to execute malicious commands or scripts on Linux, macOS, and ESXi systems, using interactive shells or shell scripts to run payloads, automate tasks, move laterally via services like SSH, and maintain persistence, even on lightweight systems that use stripped-down shells such as BusyBox.

T1059.005: Visual Basic -Adversaries may abuse Visual Basic and its derivatives (VBA and VBScript) to execute malicious commands, commonly by embedding macros in Office documents or using scripts to automate harmful actions on Windows systems.

T1059.006: Python -Adversaries may abuse Python to execute malicious commands or scripts, using its interpreter or compiled executables along with built-in libraries to interact with the system, download payloads, and perform harmful actions.

T1059.007: JavaScript -Adversaries may abuse JavaScript and its variants (JavaScript, JScript, and JXA) to execute malicious behavior across platforms, including running scripts in browsers or system scripting engines, delivering payloads via websites or secondary downloads, and using obfuscation to evade detection.

T1059.008: Network Device CLI -Adversaries may abuse command-line interfaces and scripting interpreters on network devices to execute malicious commands, alter device behavior, manipulate traffic, load malicious configurations, or disable security and logging features through local or remote access such as Telnet or SSH.

T1059.009: Cloud API -Adversaries may abuse cloud APIs to execute malicious actions by leveraging programmatic access through CLIs, cloud shells, PowerShell modules, or SDKs, using valid credentials to manipulate services such as compute, storage, IAM, networking, and security policies across cloud environments.

T1059.010: AutoHotKey & AutoIT -Adversaries abuse AutoIT and AutoHotKey scripts—sometimes compiled as executables—to automate Windows actions and stealthily execute malicious payloads such as keyloggers or malware via phishing.

T1059.011: Lua -Adversaries may abuse Lua scripts or interpreters to execute malicious commands by embedding or replacing Lua runtimes within applications for runtime code execution.

T1059.012: Hypervisor CLI -Adversaries may misuse hypervisor command-line tools to run malicious commands and manage virtual machines to enable further attacks like discovery or data encryption.

T1059.013: Container CLI/API -Adversaries may abuse Docker and Kubernetes CLIs, APIs, or SDKs to run commands, deploy containers, pull malicious or legitimate images, and discover resources within containerized and cloud environments.

Real Example:

WannaCry Ransomware Attack (2017)

After exploiting the EternalBlue SMB vulnerability, WannaCry used the **Windows Command Shell and PowerShell (T1059)** to execute malicious commands on infected systems, download additional payloads, encrypt files, and propagate itself across networks, causing widespread disruption worldwide.

Detection Methods:

- Monitor command-line and script execution (e.g., PowerShell, cmd.exe, bash, wscript) for suspicious or obfuscated commands.
- Enable detailed process and command-line logging (PowerShell Script Block Logging, Windows Event ID 4688).
- Detect abnormal parent–child process relationships (e.g., Office apps spawning PowerShell or cmd).

- Alert on execution with policy bypasses, encoded commands, or unusual execution flags.
- Monitor scripting engines accessing the network, downloading files, or modifying system settings.
- Correlate endpoint telemetry with user behavior to identify unusual or off-hours script execution.

Mitigation Strategy:

Mitigation Strategy	Application	Priority
Restrict scripting languages (PowerShell, cmd, Bash) to authorized users only	**Group Policy, Local Security Policy**	**High**
Enable PowerShell Constrained Language Mode	**Windows PowerShell, Group Policy**	**High**
Application allowlisting to block unauthorized scripts	**Windows Defender Application Control, AppLocker**	**High**
Disable unused scripting engines (wscript, cscript, AutoIT, AHK)	**Windows Features, Endpoint Security Tools**	**Medium**
Enforce script signing and execution policies	**PowerShell Execution Policy, Code Signing Certificates**	**High**
Monitor and log all command-line activity	**SIEM, Windows Event Logging, PowerShell Logging**	**High**
Limit macro execution in documents	**Microsoft Office Security Settings**	**Medium**
Use Endpoint Detection and Response (EDR) solutions	**Microsoft Defender for Endpoint, CrowdStrike**	**High**
Apply least-privilege access control	**IAM, Active Directory**	**High**

Technique 2: Software Deployment Tools

Description:

Adversaries may gain access to centralized configuration and software management tools used within an enterprise environment, which are designed to manage, deploy, and execute software across multiple systems from a single platform, and if compromised, these tools can provide attackers with large-scale remote command execution capabilities that enable lateral movement, system discovery, and execution of malicious actions across endpoints, while cloud-based management services may further allow control over both cloud and on-premises systems, some platforms permit scripts to run with high-level or SYSTEM privileges that amplify the attack impact, and such abuse typically requires elevated credentials or administrative access to the management system.

Sub-techniques:

No sub-techniques.

Real Example:

NotPetya Malware Attack (2017)

During the NotPetya attack, adversaries abused a legitimate **software update and deployment mechanism** of a Ukrainian accounting software (MeDoc) to distribute malicious code. Once inside enterprise networks, the malware leveraged administrative tools and deployment-like capabilities to execute commands, spread laterally, and deploy destructive payloads across multiple systems, causing widespread outages and data loss.

Detection Methods:

- Monitor usage of software deployment platforms (e.g., SCCM, Intune, AWS Systems Manager) for unusual or unauthorized script execution.
- Alert on deployment jobs or packages pushed outside normal maintenance windows.
- Track administrative account activity and detect anomalous logins to deployment consoles.
- Monitor mass execution of commands or software installations across multiple endpoints simultaneously.

- Enable detailed logging and audit trails for configuration and deployment actions.
- Detect communication between deployment tools and unknown or external infrastructure.
- Correlate endpoint execution events with deployment tool activity to identify misuse.

Mitigation Strategy:

Mitigation Strategy	Applicable Tools / Applications	Priority
Restrict administrative access to deployment tools	SCCM, Intune, AWS Systems Manager, Azure Arc, GCP Deployment Manager	High
Enforce multi-factor authentication for all admin accounts	All deployment and configuration tools	High
Monitor and alert on unusual deployment or script execution	SIEM, EDR, SCCM, Intune	High
Audit and log all actions performed via deployment tools	SCCM, Intune, AWS Systems Manager	High
Limit execution of scripts to signed or authorized scripts only	SCCM, Intune, PowerShell	Medium
Apply least-privilege access policies for service accounts	All enterprise deployment platforms	High
Regularly patch and update deployment tools and underlying systems	SCCM, Intune, cloud management platforms	Medium
Segment networks to limit the impact of compromised deployment tools	Network firewalls, VLANs, Cloud Security Groups	Medium

3) Persistence (TA0003)

Tactic Objective: Persistence tactic is to **maintain long-term access** to a compromised system. Adversaries aim to ensure they can return even after reboots or credential changes. This helps them continue malicious activities over time.

Tactic Description: Persistence includes techniques that allow attackers to keep access to a system after initial compromise. These techniques may involve creating startup programs, scheduled tasks, or malicious services. Attackers may also modify system configurations or use valid accounts to stay hidden. Persistence mechanisms often survive system restarts and user logouts. This ensures attackers can repeatedly access the system without re-exploiting it.

Tactic ID: TA0003
Total Techniques: 10 Techniques
Typical Phase: Post-Compromise phase
Attack Version: 17 October 2018

Technique 1: Power Settings(T1653)

Description:

Adversaries may prevent systems from hibernating, shutting down, or locking to maintain access and keep malware running. They can abuse utilities like **powercfg** or modify system settings such as lock screen and disk timeout, and in some cases, delete shutdown or reboot files to avoid termination of malicious activity.

Sub-techniques:

No sub-techniques.

Real Example:

APT29 (Cozy Bear) Campaigns

In some APT29 attacks, adversaries modified **Windows power settings** (T1653) to prevent infected machines from sleeping or entering hibernation.

This ensured that malware, including remote access tools, remained active on compromised systems, allowing continuous data collection and command execution without interruption.

Detection Methods:

- Watch for changes to power settings using tools like powercfg.
- Check registry keys for unusual sleep, hibernate, or lock screen changes.
- Alert if programs change power settings outside normal admin tasks.
- Look for processes that modify power settings while malware is running.
- Monitor computers that stay on for unusually long periods without activity.
- Use security tools (EDR) to detect suspicious persistence behaviors.

Mitigation Strategy:

Mitigation Strategy	Applications	Priority
Restrict changes to power settings to authorized users only	**Group Policy, Local Security Policy**	**High**
Monitor and audit power configuration changes	**SIEM, Windows Event Logging, EDR**	**High**
Enforce standard power policies across all endpoints	**Group Policy, SCCM, Intune**	**High**
Use EDR to detect abnormal persistence or malware activity	**Microsoft Defender for Endpoint, CrowdStrike**	**High**
Disable unnecessary scripts or tools that can modify power settings	**Endpoint Security Tools**	**Medium**
Educate users on risks of unauthorized scripts and malware	**Security Awareness Programs**	**Medium**
Regularly review and reset critical power settings	**SCCM, Intune, Local Security Policy**	**Medium**

Technique 2: Traffic Signaling(T1205)

Description:

Adversaries may use traffic signaling techniques, such as port knocking or magic packets, to hide malicious ports and activate command-and-control functions only when specific signals are received. These signals can trigger opening of closed ports, execution of malware tasks, or waking powered-off systems using Wake-on-LAN, helping attackers maintain stealthy persistence and enable lateral movement.

Sub-techniques:

T1205.001: Port Knocking - Adversaries may use **port knocking** to keep malicious ports hidden and only open or activate command-and-control connections after receiving a specific sequence of connection attempts, increasing stealth and evading detection.

T1205.002: Socket Filters - Adversaries may install stealthy network socket filters that listen for specially crafted packets to activate backdoors or command-and-control functions only when triggered, making detection difficult.

Real Example:

Cd00r Backdoor

The Cd00r backdoor used **traffic signaling (T1205)** by listening for a specific sequence of network packets (port knocking). Only after receiving the correct packet pattern would the hidden backdoor activate, allowing attackers to gain remote access while keeping ports closed and undetected.

Detection Methods:

- Monitor network traffic for unusual or repeated connection attempts to closed ports (port knocking patterns).
- Detect specially crafted packets with uncommon flags, sequences, or payloads.
- Analyze firewall logs for ports opening immediately after a specific packet sequence.
- Use network intrusion detection systems (IDS) to identify known port-knocking or magic-packet signatures.

- Monitor endpoints for raw socket or packet-capture library usage (e.g., libpcap, WinPcap).
- Correlate network events with sudden activation of new services or backdoor connections.

Mitigation Strategy:

Mitigation Strategy	Applicable Tools / Applications	Priority
Restrict and monitor access to network ports	Firewalls, Network ACLs	High
Implement intrusion detection and prevention systems	Snort, Suricata, Zeek	High
Monitor for unusual port sequences or repeated connection attempts	SIEM, Network Monitoring Tools	High
Alert on activation of services immediately after unusual traffic patterns	SIEM, EDR, Network Logs	High
Limit raw socket and packet-capture tool usage on endpoints	Endpoint Security Tools, Group Policy	Medium
Regularly update and patch network devices and firewalls	Network Management Tools, Patch Management Systems	Medium
Educate network admins on port-knocking and stealth techniques	Security Awareness Programs	Medium

4) Privilege Escalation (TA0004)

Tactic Objective: The objective of Privilege Escalation is to gain higher-level permissions within a compromised system.
Adversaries aim to move from limited user access to administrator or root access.
This allows them to fully control systems and perform restricted actions.

Tactic Description: Privilege Escalation includes techniques used by attackers to increase their access rights. This may involve exploiting software vulnerabilities or misconfigurations. Attackers can abuse weak permissions, insecure services, or credentials. Higher privileges enable attackers to disable security tools and access sensitive data. Successful privilege escalation often supports persistence and lateral movement.

Tactic ID: TA0004
Total Techniques: 4 Techniques
Typical Phase: Compromise phase
Attack Version: 17 October 2018

Technique 1: Valid Accounts(T1078)

Description:

Adversaries may steal and abuse legitimate user credentials, including inactive or high-privilege accounts, to gain unauthorized access, move laterally, escalate privileges, and evade detection by blending in with normal user activity across local, domain, and cloud systems.

Sub-techniques:

T1078.001: Default Accounts - Adversaries may exploit built-in or default accounts with unchanged or exposed credentials (e.g., Administrator, root, service accounts) to gain unauthorized access, escalate privileges, and evade defenses across systems and environments.

T1078.002: Domain Accounts - Adversaries may compromise and abuse Active Directory domain accounts, including high-privilege users or service accounts, to gain widespread access, escalate privileges, and move laterally across domain-managed systems.

T1078.003: Local Accounts - Adversaries may abuse compromised local account credentials to gain access, escalate privileges, dump additional credentials, and move laterally across systems through password reuse.

T1078.004: Cloud Accounts - Adversaries may compromise cloud accounts—whether cloud-only, federated, or synced—to gain access, maintain persistence, escalate privileges, bypass security controls, and move laterally across cloud and on-premises environments, exploiting misconfigurations, over-privileged roles, or managed identities to harvest sensitive data.

Real Example:

SolarWinds Supply Chain Attack (2020)

In the SolarWinds breach, attackers stole and abused **valid accounts (T1078)**, including legitimate user and service credentials, to access cloud and on-premises environments. By using these valid credentials, the adversaries blended in with normal user activity, moved laterally, escalated privileges, and maintained persistence while evading traditional security controls.

Detection Methods:

- Monitor authentication logs for anomalous logins such as unusual times, locations, or devices.
- Detect impossible travel and concurrent logins from different geographies.
- Alert on excessive or abnormal use of privileged or service accounts.
- Monitor for account usage after long periods of inactivity (dormant accounts).
- Track changes to account permissions, role assignments, or group memberships.
- Correlate login activity with suspicious actions like data access or remote services.
- Use UEBA and SIEM tools to identify deviations from normal user behavior.

Mitigation Strategy:

Mitigation Strategy	Applicable Tools / Applications	Priority
Enforce strong, unique passwords for all accounts	Active Directory, Azure AD, IAM tools	High
Enable multi-factor authentication (MFA) for all users	Azure AD, Okta, Duo Security	High
Disable or remove inactive or unused accounts	Active Directory, Azure AD, IAM tools	High
Monitor login activity for anomalous behavior	SIEM, UEBA, Microsoft Sentinel, Splunk	High
Restrict the use of high-privilege accounts	Group Policy, IAM, Role-Based Access Control	High
Regularly audit account permissions and roles	Active Directory, IAM tools, Cloud Admin Consoles	Medium
Implement Just-In-Time (JIT) access for administrative accounts	Azure AD Privileged Identity Management, AWS IAM	Medium
Educate users on phishing and credential protection	Security Awareness Programs	Medium

Technique 2: Boot or Logon Initialization Scripts(T1037)

Description:

Adversaries may use boot or logon initialization scripts to maintain persistence on a system by automatically executing malicious programs when the system starts or a user logs in. These scripts can perform administrative tasks, run additional programs, or send information to internal or external servers. Depending on the script's configuration, local or administrative credentials may be required to modify or execute it. Some initialization scripts run with elevated privileges, allowing adversaries to escalate their access and perform actions beyond their normal permission level. This technique enables attackers to

maintain long-term access and execute commands without triggering immediate detection.

Sub-techniques:

T1037.001: Logon Script (Windows) - Adversaries may achieve persistence by configuring Windows logon scripts in the registry to automatically run malicious scripts each time a user logs in.

T1037.002: Login Hook - Adversaries may establish persistence on macOS by modifying the `com.apple.loginwindow.plist` login or logout hooks to execute a malicious script with root privileges whenever a user logs in or out.

T1037.003: Network Logon Script - Adversaries may gain persistence by assigning malicious network logon scripts through Active Directory or Group Policy, causing them to run automatically for users across multiple systems at login.

T1037.004: RC Scripts - Adversaries may achieve persistence on Unix-like systems by modifying RC startup scripts (e.g., `rc.local`) to run malicious commands as root during system boot, ensuring execution across reboots.

T1037.005: Startup Items -- Adversaries can achieve persistence on macOS by placing malicious scripts or executables in the /Library/StartupItems directory, causing them to run automatically at boot with root-level privileges

Real Example:

Turla (Snake/Uroburos) APT Group

Turla has used **boot or logon initialization scripts (T1037)** to maintain persistence by modifying Windows logon scripts and startup configurations so that malicious code automatically executed whenever a user logged in, allowing long-term access while blending in with legitimate system behavior.

Detection Methods:

- Monitor changes to startup and logon script locations (e.g., Windows Registry keys, Startup folders, Group Policy logon scripts).
- Alert on modifications to Unix/Linux startup scripts such as rc.local, init scripts, or systemd service files.
- Track creation or modification of macOS login hooks, launch agents, or startup items.

- Enable file integrity monitoring on boot and logon script directories.
- Detect unusual parent–child process relationships triggered at boot or user logon.
- Review Group Policy and Active Directory logs for unauthorized script assignments.

Mitigation Strategy:

Mitigation Strategy	Applicable Tools / Applications	Priority
Restrict who can create or modify startup/logon scripts	Active Directory, Group Policy, Local Security Policy	High
Monitor and audit changes to boot and logon scripts	SIEM, File Integrity Monitoring (FIM), EDR	High
Enforce least-privilege access for users and admins	IAM, Active Directory, RBAC	High
Disable unused or legacy startup mechanisms	System Configuration Tools, OS Security Settings	Medium
Use allowlisting for startup scripts and executables	AppLocker, WDAC, Endpoint Security Tools	High
Regularly review Group Policy logon script assignments	Group Policy Management Console	Medium
Keep operating systems and startup services updated	Patch Management Tools, OS Updates	Medium

5) Defense Evasion (TA0005)

Tactic Objective: The objective of Defense Evasion is to avoid detection and bypass security controls.
Adversaries aim to remain hidden while carrying out malicious activities. This allows them to maintain access and complete their objectives without being discovered.

Tactic Description: Privilege Defense Evasion consists of techniques that attackers use to bypass or disable security mechanisms.
Adversaries may also manipulate system or application configurations to hide their presence. Techniques like file deletion, credential dumping evasion, and disabling security services are common. Successful defense evasion increases the attacker's chances of maintaining persistence and achieving their goals undetected.

Tactic ID: TA0005
Total Techniques: 11 Techniques
Typical Phase: Post-Compromise phase
Attack Version: 17 October 2018

Technique 1: Direct Volume Access(T1006)

Description:

Adversaries may access disk volumes directly to bypass normal file access controls and evade file system monitoring. By reading raw file system structures or using tools like NinjaCopy, vssadmin, or wbadmin, they can copy or manipulate data stealthily without triggering standard security controls.

Sub-techniques:

No sub-techniques.

Real Example:

Ryuk Ransomware (2018–2019)

Ryuk ransomware used techniques similar to **Direct Volume Access (T1006)** by leveraging Windows utilities and raw volume access to copy and encrypt files directly from system drives. This allowed the malware to bypass standard file

permissions and monitoring, rapidly encrypting data across compromised systems.

Detection Methods:

- Monitor for unusual use of volume-level utilities such as vssadmin, wbadmin, esentutl, or raw disk access tools.
- Alert on processes accessing disk volumes directly outside normal application behavior.
- Enable file integrity monitoring (FIM) to detect unexpected file copies or modifications.
- Track PowerShell or command-line executions performing raw volume operations.
- Correlate access to shadow copies, backups, or volume snapshots with non-administrative processes.
- Use EDR solutions to detect suspicious low-level disk operations that bypass normal file system APIs.

Mitigation Strategy:

Mitigation Strategy	Applications	Priority
Restrict use of volume-level utilities to authorized users	Group Policy, Local Security Policy, Endpoint Security Tools	High
Monitor and audit raw disk or volume access	SIEM, EDR, File Integrity Monitoring (FIM)	High
Disable unnecessary or legacy backup/shadow copy tools	System Configuration Tools, OS Security Settings	Medium
Enforce least-privilege access for users and service accounts	IAM, Active Directory, RBAC	High
Use allowlisting to block unauthorized disk utilities	AppLocker, WDAC, Endpoint Security Tools	High

Regularly backup and secure critical data	Backup Solutions, Cloud Storage, On-prem Backup Systems	Medium
Keep systems patched and updated to prevent exploitation of volume access vulnerabilities	Patch Management Tools, OS Updates	Medium

Technique 2: Network Boundary Bridging(T1599)

Description:

Adversaries may compromise routers, firewalls, or network segmentation devices to bypass security boundaries between trusted and untrusted networks. By reconfiguring these devices, they can allow normally blocked traffic, enabling command and control, data exfiltration, and lateral movement into other network segments or organizations.

Sub-techniques:

T1599.001: Network Address Translation Traversal - Adversaries may modify or implant malicious NAT configurations on network devices to bypass network boundaries, route otherwise blocked traffic, and obscure malicious activity between trusted and untrusted networks.

Real Example:

VPNFilter Malware (2018)

The VPNFilter malware compromised routers and network devices, allowing attackers to modify device configurations and **bridge network boundaries (T1599)** to intercept traffic, enable command-and-control communication, and access internal networks that were otherwise protected by perimeter defenses.

Detection Methods:

- Monitor network device configurations for unauthorized changes, including NAT rules and routing tables.
- Track unusual traffic patterns crossing network boundaries, such as unexpected internal-to-internal or external-to-internal flows.
- Alert on administrative logins to routers, firewalls, or segmentation devices from unusual sources or at odd times.

- Use network intrusion detection systems (IDS) to identify anomalous routing or packet modifications.
- Audit device firmware and software integrity to detect unauthorized patching or image modifications.
- Correlate logs from multiple network devices to detect unexpected bridging or multi-hop proxy activity.

Mitigation Strategy:

Mitigation Strategy	Applications	Priority
Restrict administrative access to network devices	AAA systems, RADIUS, TACACS+, Firewalls, Routers	High
Monitor and audit configuration changes on network devices	SIEM, Network Configuration Management Tools	High
Enable logging and alerting for unusual NAT or routing changes	SIEM, Network Monitoring Tools	High
Apply least-privilege access and role separation for network admins	IAM, RBAC, Network Device Access Controls	High
Keep device firmware and software up to date and verify integrity	Patch Management Tools, Firmware Integrity Checks	High
Segment networks and limit unnecessary inter-network traffic	VLANs, Firewalls, ACLs	Medium
Use IDS/IPS to detect abnormal traffic bridging	Snort, Suricata, Zeek	Medium
Regularly backup and validate network device configurations	Network Management Systems, Backup Tools	Medium

6) Credential Access (TA0006)

Tactic Objective: The objective of the Credential Access tactic is to obtain usernames, passwords, or other authentication information from a target system. By stealing valid credentials, adversaries can impersonate legitimate users and gain unauthorized access. This enables them to expand their presence and access sensitive resources.

Tactic Description: Credential Access refers to techniques used by adversaries to collect login credentials stored or used on a system. Attackers may capture passwords through malware such as keyloggers, memory scraping tools, or by extracting saved credentials from browsers and operating systems. In some cases, fake login prompts or phishing techniques are used to trick users into revealing their credentials. Once obtained, these credentials can be reused to access other systems, move laterally within the network, or maintain long-term access.

Tactic ID: TA0006
Total Techniques: 6 Techniques
Typical Phase: Post-Compromise phase
Attack Version: 17 October 2018

Technique 1: Network Sniffing(T1040)

Description:

Network sniffing allows adversaries to observe network traffic to gather information such as IP addresses, services, and system details. In cloud environments, traffic mirroring features can be abused to capture data from virtual machines. Captured traffic may include cleartext data, especially where encryption is terminated at load balancers. This information can then be exfiltrated and used to support further attacks such as lateral movement or credential theft.

Sub-techniques:

No sub-techniques.

Real Example:

Ettercap-based Man-in-the-Middle Attacks

Attackers have used tools like **Ettercap** to perform **network sniffing (T1040)** by poisoning ARP tables, allowing them to capture unencrypted network traffic, steal credentials, and gather network configuration details to enable lateral movement within compromised networks.

Detection Methods:

- Monitor for network interfaces placed into promiscuous mode on endpoints.
- Detect ARP spoofing or poisoning activity using network monitoring tools.
- Analyze network traffic for unusual packet capture behavior or traffic mirroring.
- Monitor cloud environments for unauthorized use of traffic mirroring services.
- Alert on execution of packet capture tools (e.g., tcpdump, Wireshark, Ettercap).
- Correlate network device logs for unexpected packet capture or monitor commands.

Mitigation Strategy:

Mitigation Strategy	Applicable Tools / Applications	Priority
Monitor for and prevent promiscuous mode on network interfaces	EDR, Endpoint Security Tools	High
Detect and alert on ARP spoofing or poisoning	IDS/IPS, Network Monitoring Tools (Snort, Zeek)	High
Restrict execution of packet capture tools	AppLocker, WDAC, Endpoint Security Tools	High

Monitor cloud traffic mirroring configurations for unauthorized use	AWS Traffic Mirroring, GCP Packet Mirroring, Azure vTap	High
Encrypt sensitive network traffic end-to-end	TLS, IPsec, VPNs	High
Audit and log network device capture or monitoring commands	Network Device Logging, SIEM	Medium

Technique 2: Brute Force(T1110)

Description:

Adversaries may use brute force attacks to repeatedly guess passwords or crack stolen password hashes in order to gain unauthorized access to accounts, often combining this technique with credential discovery, remote services, or evasion of location-based security controls.

Sub-techniques:

T1110.001: Password Guessing - Adversaries may attempt password guessing by repeatedly trying common or likely passwords to access accounts, risking detection through multiple failed logins or account lockouts.

T1110.002: Password Cracking - Adversaries may crack stolen password hashes using guessing techniques or rainbow tables to recover plaintext passwords and reuse them to access systems and services.

T1110.003: Password Spraying - Adversaries may perform **password spraying** by trying one or a few common passwords across many accounts to gain access while avoiding account lockouts.

T1110.004: Credentials Stuffing - Adversaries may conduct **credential stuffing** by reusing leaked username-password pairs from other breaches to exploit password reuse and access target accounts, often triggering multiple failed logins or account lockouts.

Real Example:

Microsoft Exchange Online Password Spraying Attacks

Attackers have conducted large-scale **brute force and password spraying attacks (T1110)** against Microsoft 365 and Exchange Online accounts by repeatedly attempting common or leaked passwords across many user accounts, successfully compromising valid accounts with weak credentials and gaining unauthorized access to email and cloud resources.

Detection Methods:

- Monitor authentication logs for **multiple failed login attempts** from a single IP or across many accounts.
- Detect **password spraying patterns**, where one password is tried against many usernames.
- Identify **abnormal login times or locations** inconsistent with normal user behavior.
- Alert on **repeated authentication failures followed by a success**.
- Use SIEM and identity protection tools to flag **high-volume or rapid login attempts**.

Mitigation Strategy:

Mitigation Strategy	Applications	Priority
Enforce strong, complex, and unique passwords	**Active Directory, IAM Tools**	**High**
Enable multi-factor authentication (MFA)	**Azure AD, Okta, Duo Security, Google Workspace**	**High**
Monitor and alert on repeated failed login attempts	**SIEM, Microsoft Sentinel, Splunk**	**High**
Implement account lockout or throttling policies	**Active Directory, IAM Systems**	**High**
Educate users on phishing and credential hygiene	**Security Awareness Training Programs**	**Medium**

Use password blacklists to block common passwords	**IAM Tools, AD Password Policies**	**Medium**
Regularly audit and disable inactive accounts	**Active Directory, IAM Tools**	**Medium**

7) Discovery (TA0007)

Tactic Objective: The **attacker is trying to understand your system and network**. After getting access, they look around to see what devices, users, and resources are available. This helps them know where they are and what they can control. They usually use **normal system tools** that already exist in the operating system. The goal is to plan the next steps of the attack more effectively.

Tactic Description: Discovery is the stage where an attacker tries to **understand the victim's environment** after getting access. They collect information about the system, users, network, and connected devices. This helps them see what resources are available and what they can control. Attackers often use **built-in operating system tools** to avoid suspicion. The main goal is to decide the next actions and move further inside the network.

Tactic ID: TA0007
Total Techniques: 11 Techniques
Typical Phase: Post-Compromise phase
Attack Version: 17 October 2018

Technique 1: File and Directory Discovery(T1083)

Description:

Adversaries search and list files or directories on a system or network share to identify valuable data, configurations, or targets. They use common command-line tools (like dir, ls, find) or custom scripts to map the file system. This information helps them decide next steps, such as data theft, privilege escalation, or further compromise.

Sub-techniques:

No sub-techniques.

Real Example:

WannaCry ransomware attack

During the WannaCry ransomware attack, the malware scanned infected Windows systems using commands similar to dir to locate important user files (such as documents and backups) across directories before encrypting them, helping the attackers identify valuable data to target.

Detection Methods:

- Monitor execution of file-listing commands such as dir, tree, ls, find, or locate, especially when run frequently or by non-admin users.
- Detect abnormal access to sensitive directories (e.g., system folders, user home directories, network shares).
- Use endpoint detection tools (EDR) to identify scripts or binaries performing large-scale file enumeration.
- Analyze command-line logging (Windows Event ID 4688) for suspicious discovery-related commands.
- Monitor network devices for unusual CLI commands like show flash or dir executed outside normal administrative activity.

Mitigation Strategy:

Mitigation Strategy	Applications	Priority
Restrict user permissions to sensitive directories	Active Directory, IAM Tools, File System ACLs	High
Monitor and alert on unusual file enumeration commands	SIEM, EDR, Microsoft Sentinel, Splunk	High
Enable logging of command-line activity and scripts	Windows Event Logging, Sysmon, Linux auditd	High
Use endpoint protection to detect suspicious file discovery scripts	EDR, Antivirus, Endpoint Security Tools	High
Segment network shares and sensitive file systems	VLANs, Firewalls, Access Control Lists	Medium
Regularly review and audit file access permissions	File Integrity Monitoring (FIM), IAM Tools	Medium

Technique 2: Password Policy Discovery(T1201)

Description:

Adversaries try to discover an organization's password policy (such as length, complexity, and lockout limits) so they can tailor brute-force or dictionary attacks that avoid account lockouts and increase the chances of guessing valid passwords.

Sub-techniques:

No sub-techniques.

Real Example:

2017 Equifax breach

During the **2017 Equifax breach**, attackers obtained detailed information about password policies for internal systems. This allowed them to craft targeted password guessing and brute-force attacks that complied with complexity requirements, helping them escalate privileges and move laterally within the network.

Detection Methods:

- Monitor queries to domain controllers or Active Directory for requests related to password policy attributes.
- Alert on abnormal use of account management tools or scripts that enumerate password requirements.
- Detect repeated access attempts to Group Policy Objects or security settings associated with password configurations.
- Use SIEM or EDR solutions to flag unexpected queries to cloud IAM services for password policy information.
- Correlate access logs with unusual administrative or service account activity targeting password policy endpoints.

Mitigation Strategy:

Mitigation Strategy	Applications	Priority
Restrict access to password policy information	Active Directory, IAM Tools, GPOs, Cloud IAM	High
Monitor and alert on enumeration of password policies	SIEM, Microsoft Sentinel, Splunk	High
Limit use of administrative or service accounts for querying policies	IAM Tools, RBAC, Active Directory	High
Enforce least privilege for all accounts	Active Directory, IAM, Cloud IAM	High
Audit and review access to policy-related configurations	SIEM, File Integrity Monitoring, AD Auditing	Medium
Educate administrators on secure handling of password policy info	Security Awareness Training Programs	Medium
Implement multi-factor authentication to reduce reliance on password guessing	Azure AD, Okta, Duo Security	High

8) Lateral Movement (TA0008)

Tactic Objective: The adversary aims to move from one compromised system to other systems in the network.
This helps them reach valuable assets like servers or critical data.
It also allows attackers to expand their control inside the environment.

Tactic Description: Lateral Movement involves using access on one system to gain access to another system within the same network. Attackers often reuse stolen credentials, tokens, or existing trust relationships. Common methods include remote services like RDP, SMB, SSH, or Windows Admin Shares. They may also exploit weak passwords or misconfigured permissions.
This tactic helps attackers avoid detection while gradually spreading across the network.

Tactic ID: TA0008
Total Techniques: 1 Techniques
Typical Phase: Post-Compromise phase
Attack Version: 17 October 2018

Technique 1: Software Deployment Tools(T1072)

Description:

Adversaries may misuse centralized management tools that organizations normally use to control and update computers across the network. If attackers gain access to these tools, they can execute commands on multiple machines at once and move sideways across the network. This access can help them spread malware, collect data, or even damage systems, such as wiping hard drives. Because these tools are trusted and commonly used, malicious actions through them can be hard to detect.

Sub-techniques:

No sub-techniques.

Real Example:

NotPetya (2017) attack:

A well-known real-world example occurred during the NotPetya (2017) attack. Attackers gained access to a company's network and abused software deployment and management tools to spread malware across multiple systems. Once they compromised administrative credentials, they used centralized management capabilities similar to tools like SCCM to push malicious updates to many endpoints at once. This allowed the attackers to move laterally very quickly and execute destructive payloads across the entire enterprise network.

Detection Methods:

- Monitor **unusual login activity** between internal systems
- Detect **remote execution tools** (PsExec, WMI, PowerShell remoting) used abnormally
- Watch for **credential reuse** across multiple machines
- Alert on **unexpected access to admin shares** (e.g., C$, ADMIN$) Monitor **sudden use of software deployment or management tools**
- Use **network traffic analysis** to spot abnormal internal connections
- Track **accounts accessing many systems in a short time**

Mitigation Strategy:

Mitigation Strategy	Application	Priority
Restrict administrative access to deployment tools	SCCM, Intune, AWS Systems Manager	High
Enforce multi-factor authentication (MFA)	Deployment platforms and admin accounts	High
Monitor and log all deployment actions	SCCM, Intune, cloud management tools	High
Apply least privilege principles	User and service accounts	High
Regularly patch and update deployment software	SCCM, Intune, AWS Systems Manager	Medium
Audit and alert on anomalous script execution	Endpoint systems and deployment servers	Medium

9) Collection (TA0009)

Tactic Objective: The adversary aims to move from one compromised system to other systems in the network.
This helps them reach valuable assets like servers or critical data.
It also allows attackers to expand their control inside the environment.

Tactic Description: Collection involves techniques used by attackers to capture data of interest after gaining access.
They may collect documents, screenshots, keystrokes, emails, or database records.
Data can be gathered from local systems, shared drives, or cloud services.
Attackers often use built-in tools to avoid detection while collecting information.
The collected data is usually prepared for exfiltration to external systems.

Tactic ID: TA0009
Total Techniques: 4 Techniques
Typical Phase: Post-Compromise phase
Attack Version: 17 October 2018

Technique 1: Input Capture(T1056)

Description:

Adversaries may capture user input to steal credentials or sensitive information entered during normal system use. This can be done covertly, such as hooking credential APIs or keylogging, or through deception by tricking users into entering details into fake login pages or portals. Such techniques allow attackers to collect usernames, passwords, and other confidential data without the user's awareness.

Sub-techniques:

T1056.001: Keylogging - Adversaries may use keylogging to secretly record keystrokes over time to capture credentials, sometimes forcing reauthentication (e.g., clearing cookies) to increase the chance of stealing login details.

T1056.002: GUI Input Capture - Adversaries can display fake but realistic system or application login prompts (e.g., UAC dialogs, installers,

browser/email logins) to trick users into entering credentials, which are then captured by malicious scripts or programs.

T1056.003: Web Portal Capture - Adversaries may compromise external login portals (such as VPN or web login pages) to secretly capture and exfiltrate user credentials entered by legitimate users while allowing normal login to continue.

T1056.004: Credential API Hooking - Adversaries may intercept Windows or Linux API/system function calls to capture authentication credentials directly from function parameters instead of recording keystrokes.

Real Example:

DarkHotel APT campaign

In the **DarkHotel APT campaign**, attackers used **keyloggers and fake login prompts** on compromised hotel business center computers to capture executives' usernames and passwords as they logged into email and VPN services, enabling further network access.

Detection Methods:

- Monitor for **unauthorized keylogging behavior**, such as processes accessing keyboard hooks or low-level input APIs (e.g., SetWindowsHookEx on Windows).
- Detect **suspicious API hooking** or DLL injection activity using EDR solutions.
- Watch for **unexpected credential prompts or fake GUI dialogs**, especially those not tied to legitimate system processes.
- Analyze **process behavior** for abnormal access to input devices (/dev/input/* on Linux).
- Monitor **outbound network traffic** for small, frequent data exfiltration patterns that may indicate stolen keystrokes or credentials being sent to a C2 server.

Mitigation Strategy:

Mitigation Strategy	Application / Where to Apply	Priority
Enable Multi-Factor Authentication (MFA)	**All user accounts, VPNs, cloud apps, admin logins**	**High**
Deploy Endpoint Detection & Response (EDR)	**User endpoints, servers, virtual machines**	**High**

Restrict API hooking and DLL injection	**Windows systems via EDR, AppLocker, Exploit Guard**	**High**
Use least privilege access	**User accounts, service accounts**	**High**
Application whitelisting	**Workstations and critical servers**	**Medium**
Monitor for unauthorized input device access	**Linux endpoints (/dev/input/*), macOS**	**Medium**
User awareness training	**All employees**	**Medium**
Encrypt network traffic (TLS/HTTPS)	**Web apps, internal services**	**Medium**
Regular patching and OS updates	**Operating systems and applications**	**Low**
Disable unnecessary admin credential prompts	**End-user systems**	**Low**

Technique 2: Data from Local System(T1005)

Description:

Adversaries may look for sensitive files and data stored on a local system, such as configuration files, databases, virtual machine files, or information in memory, before stealing it. They often use command-line tools, scripts, or network device commands to search the file system and automatically collect valuable data for later exfiltration.

Sub-techniques:

No sub-techniques.

Real Example:

Target data breach (2013)

In the **Target data breach (2013)**, attackers who had gained internal access searched local point-of-sale (POS) systems to collect sensitive data stored in system memory. Malware installed on the POS devices scraped credit card information locally before the data was later exfiltrated to attacker-controlled servers.

Detection Methods:

- Monitor **file access patterns** for unusual or bulk reads of sensitive files (e.g., documents, databases, configuration files).
- Detect **abnormal command-line activity** (cmd, PowerShell, bash) used to search or copy files.
- Watch for **unexpected access to process memory** or virtual machine files.
- Use **Endpoint Detection and Response (EDR)** to identify automated collection tools or scripts.
- Alert on **suspicious file staging** in temporary or user directories prior to exfiltration.

Mitigation Strategy:

Mitigation Strategy	Application / Tool	Priority
Least privilege access	**Role-Based Access Control (RBAC), IAM**	**High**
Endpoint Detection & Response	**Microsoft Defender for Endpoint, CrowdStrike, SentinelOne**	**High**
File access monitoring	**File Integrity Monitoring (FIM), OSSEC, Tripwire**	**High**
Disable unnecessary admin rights	**Group Policy, Local Security Policy**	**High**
Data Loss Prevention (DLP)	**Microsoft Purview DLP, Symantec DLP**	**Medium**

10) Command and Control (TA0011)

Tactic Objective
The adversary aims to maintain communication with compromised systems.
This communication allows them to send commands and receive stolen data.
It helps attackers control infected systems remotely over time.

Tactic Description
Command and Control involve techniques used by attackers to communicate with systems they have compromised. Through C2 channels, adversaries can issue commands, update malware, or move laterally within the network.
Communication may occur over common protocols such as HTTP, HTTPS, DNS, or cloud-based services to blend in with normal traffic. Attackers often encrypt or obfuscate C2 traffic to avoid detection.C2 channels can be temporary or long-term, depending on the attacker's goals. This tactic is critical for maintaining persistence and coordinating further malicious actions.

Tactic ID: TA0011
Total Techniques: 7 Techniques
Typical Phase: Post-compromise phase
Attack Version: 17 October 2018

Technique 1: Non-Application Layer Protocol(T1095)

Description:

Adversaries may hide command-and-control (C2) traffic by using low-level, non-application layer protocols such as ICMP, UDP, or VMCI instead of common protocols like HTTP/HTTPS. These protocols are often less monitored or invisible to traditional security tools, allowing attackers to communicate stealthily, bypass network controls, and maintain persistent access, especially in virtualized environments like ESXi.

Sub-techniques:

No sub-techniques.

Real Example:

PingTunnel / ICMP backdoor

The **PingTunnel / ICMP backdoor** used by malware such as **Backdoor.Ping** and **PLATINUM APT** is a real example, where attackers tunnel command-and-control (C2) traffic inside **ICMP echo request/reply packets** to bypass firewalls and monitoring systems that allow ping traffic but block application-layer protocols.

Detection Methods:

- Monitor **ICMP, UDP, and other non-application layer traffic** for abnormal frequency, size, or patterns.
- Detect **ICMP packets carrying payloads** or data larger than normal echo requests/replies.
- Use **network behavior analysis (NBA)** to identify covert or tunneled communications.
- Inspect **east–west traffic** within virtualized environments (e.g., ESXi VMCI activity).
- Alert on **unusual protocol usage** between hosts that normally do not communicate using those protocols.
- Correlate **endpoint logs and network telemetry** to detect hidden C2 channels.

Mitigation Strategy:

Mitigation Strategy	Application / Where Applied	Priority
Restrict ICMP, UDP, and non-essential protocols	Firewalls, Routers, Network Gateways	High
Deep Packet Inspection (DPI)	Next-Generation Firewalls (NGFW), IDS/IPS	High
Network Segmentation	Internal Network, Cloud VPC/VNETs	High
Monitor East-West Traffic	Data Centers, Virtualized & Cloud Environments	Medium

Disable Unused Protocols	Servers, Network Devices, ESXi Hosts	High
Enable Logging & Alerting for Protocol Abuse	SIEM, NDR Tools	High
VMCI Access Restriction	ESXi Hosts, Hypervisors	High
Anomaly-Based Traffic Detection	Network Behavior Analysis Tools	Medium
Regular Network Configuration Audits	Routers, Firewalls, Cloud Networking	Medium
Endpoint Security Controls	EDR/XDR on Hosts & VMs	Medium

Technique 2: Hide Infrastructure(T1665)

Description:

Adversaries may change or hide their network traffic to avoid being detected by security tools. They do this by masking malicious destinations, filtering out traffic from security scanners, or using trusted services so their activity looks normal. Attackers often use VPNs, proxies, or cloud infrastructure to hide their real IP address and blend in with legitimate user traffic. They may also block traffic coming from security researchers or sandbox environments to prevent analysis. By hiding their Command-and-Control (C2) infrastructure this way, attackers can stay active longer without being blocked or taken down.

Sub-techniques:

No sub-techniques.

Real Example:

APT29 (Cozy Bear):

The group hid its Command-and-Control (C2) infrastructure by using legitimate cloud services and trusted domains, such as compromised websites and popular hosting providers. They masked malicious traffic behind HTTPS encryption and proxy servers, making the C2 communication appear like normal web traffic. In some cases, they also used URL shorteners and benign-looking domains that initially served harmless content and later redirected victims to malicious

infrastructure, helping them evade detection and delay takedown by security teams.

Detection Methods:

- Monitor **network traffic to trusted cloud services** for unusual or suspicious behaviour
- Detect **frequent IP address changes** or traffic routed through VPNs and proxies
- Identify **shortened URLs or redirect chains** used in emails or web traffic
- Monitor **domain reputation changes**, especially domains that suddenly become malicious
- Use **TLS/HTTPS inspection and behavioural analysis** to detect hidden C2 traffic
- \Alert on **connections that bypass geolocation or conditional access controls**
- Correlate traffic patterns that resemble **C2 beaconing behaviour** even on trusted domains

Mitigation Strategy:

Mitigation Strategy	Application	Priority
Implement strict firewall and proxy rules	Network perimeter devices	High
Block known malicious domains and IPs	DNS filtering, firewall	High
Monitor and restrict use of unauthorized cloud or proxy services	Endpoints, network security	High
Use threat intelligence feeds to detect malicious infrastructure	SIEM, IDS/IPS	Medium
Apply network segmentation to limit C2 reach	Enterprise network	Medium
Enable logging and alerting for anomalous redirection or tunneling	Network monitoring systems	Medium
Keep all software and security tools updated	Endpoints and network devices	Medium

11) Exfiltration (TA0010)

Tactic Objective
The adversary aims to steal sensitive data from the compromised environment.
They try to move collected information outside the victim's network.
This data is usually sent to attacker-controlled systems for misuse or sale.

Tactic Description
Exfiltration occurs after attackers have collected valuable data such as credentials, files, or recordings.
They may transfer data over network channels, cloud services, or removable media.
Attackers often compress or encrypt data to avoid detection.
Legitimate services like email, cloud storage, or web traffic may be abused.
Exfiltration techniques are designed to blend with normal activity.
This helps attackers steal data without raising security alerts.

Tactic ID: TA00010
Total Techniques: 2 Techniques
Typical Phase: Post-Compromise phase
Attack Version: 17 October 2018

Technique 1: Exfiltration Over Alternative Protocol(T1048)

Description:

Adversaries may exfiltrate stolen data using **alternate network protocols** different from their main C2 channel to evade detection, such as FTP, HTTP/S, DNS, SMTP, or SMB. These protocols may be **encrypted or obfuscated** to blend with normal traffic. In cloud environments, attackers can also abuse **legitimate cloud services and APIs** (e.g., SharePoint, GitHub, AWS S3) to download sensitive data without triggering traditional network-based alerts.

Sub-techniques:

T1048.001: Exfiltration Over Symmetric Encrypted Non-C2 Protocol - Adversaries may exfiltrate data using symmetric encryption (e.g., AES, RC4)

over various protocols, manually sharing keys to hide the data from detection even in normally unencrypted or already encrypted channels.

T1048.002: Exfiltration Over Asymmetric Encrypted Non-C2 Protocol - Adversaries may exfiltrate data over asymmetrically encrypted protocols like HTTPS/TLS, using public-key cryptography to securely transmit data to alternate locations while hiding it from detection.

T1048.003: Exfiltration Over Unencrypted Non-C2 Protocol - Adversaries may exfiltrate data over unencrypted protocols like HTTP, FTP, or DNS, often using obfuscation techniques such as encoding or embedding data in headers to hide it from detection.

Real Example:

Target data breach (2013)

In the **Target data breach (2013)**, attackers exfiltrated stolen credit card data from internal systems to external servers using **FTP**, which was different from their primary command-and-control communication channel, helping them evade detection.

Detection Methods:

- Monitor outbound network traffic for **unusual or unauthorized protocols** (e.g., FTP, DNS, SMB) leaving the network
- Detect **abnormal data transfer volumes or timing**, especially outside business hours
- Use **DLP (Data Loss Prevention)** to identify sensitive data moving over non-standard channels
- Analyze **proxy, firewall, and IDS/IPS logs** for connections to unknown external destinations
- Look for **unexpected use of command-line tools** (e.g., curl, ftp, scp) by users or processes

Mitigation Strategy:

Mitigation Strategy	Application	Priority
Restrict use of non-essential protocols (FTP, SMTP, SMB, etc.)	Firewalls, Network Gateways	High
Implement Data Loss Prevention (DLP) policies	DLP Solutions, Email Gateways	High
Monitor and alert on unusual outbound traffic	SIEM, NDR, Firewall Logs	High
Encrypt sensitive data in transit	VPNs, TLS/SSL for permitted protocols	Medium
Use proxy or web gateways to control HTTP/S traffic	Web Proxies, NGFW	High
Enforce least-privilege access to systems and data	Endpoint & Identity Management	High
Regularly audit cloud and SaaS data access	Cloud Security Posture Management (CSPM)	Medium
Limit command-line tools that can exfiltrate data	Endpoint Protection, EDR/XDR	Medium
Network segmentation for sensitive environments	VLANs, Subnets, Cloud VPCs	Medium
Educate users on safe handling of sensitive data	Security Awareness Training	Medium

Technique 2: Automated Exfiltration(T1020)

Description:

Adversaries may automatically exfiltrate collected data without human interaction, allowing continuous or scheduled transfer of sensitive information out of the network. This reduces the need for manual actions and helps attackers remain stealthy. Automated exfiltration often works alongside other techniques,

such as sending data over command-and-control channels or alternative network protocols.

Sub-techniques:

T1020.001: Traffic Duplication - Adversaries may abuse traffic mirroring or duplication features, including in cloud environments, to automatically capture and exfiltrate network data through compromised or controlled infrastructure.

Real Example:

Capital One breach (2019)

In the **Capital One breach (2019)**, attackers leveraged misconfigured AWS resources and automated scripts to exfiltrate sensitive customer data from cloud storage, continuously transferring large volumes of information without manual intervention.

Detection Methods:

- Monitor **unusual or continuous outbound data transfers**, especially outside business hours
- Detect **abnormal traffic volume or frequency** from hosts, cloud workloads, or service accounts
- Alert on **automated scripts or scheduled tasks** initiating network connections
- Analyze **cloud audit logs (AWS CloudTrail, Azure Activity Logs)** for repeated data access and export actions
- Use **DLP tools** to identify sensitive data leaving the network
- Inspect **network flow logs** for consistent, scripted exfiltration patterns

Mitigation Strategy:

Mitigation Strategy	Application	Priority
Monitor and alert on unusual outbound data flows	SIEM, NDR, Cloud Monitoring	High
Implement Data Loss Prevention (DLP) controls	DLP Solutions, Email Gateways, Cloud DLP	High
Restrict automated scripts and scheduled tasks that transfer data	Endpoint Protection, EDR/XDR	High
Apply least-privilege access for cloud and network resources	IAM, Cloud Security Posture Management	High
Enable logging and auditing of data access and transfers	Cloud Audit Logs (AWS CloudTrail, Azure Activity Logs), Syslog	High
Segment sensitive systems and data	VLANs, Subnets, Cloud VPCs	Medium
Encrypt sensitive data at rest and in transit	TLS/SSL, VPNs, Cloud Encryption	Medium
Regularly review and patch network devices to prevent misuse of traffic mirroring	Network Devices, Cloud Infrastructure	Medium
Conduct periodic security awareness training	Security Awareness Programs	Medium
Implement anomaly detection for unusual data access patterns	UEBA, AI-based Threat Detection	High

12) Impact (TA0040)

Tactic Objective
The adversary aims to disrupt, damage, or destroy systems and data. They seek to degrade an organization's availability, integrity, or trust. The goal is often to cause financial loss, operational downtime, or fear.

Tactic Description
Impact techniques are used after attackers have achieved their main access goals.
Adversaries may delete data, encrypt systems, deface websites, or shut down services.
Common examples include ransomware, data destruction, and denial-of-service attacks. These actions can interrupt business operations and critical services.
Impact tactics are often the most visible phase of an attack. They are used to pressure victims, send messages, or maximize damage.

Tactic ID: TA00040
Total Techniques: 4 Techniques
Typical Phase: Final phase
Attack Version: 14 March 2019

Technique 1: Firmware Corruption(T1495)

Description:

Adversaries can damage a system by corrupting or overwriting its firmware, such as the BIOS, which controls how hardware starts and works. Because firmware is stored in non-volatile memory, this damage can make devices unable to boot or function properly, sometimes permanently. Such attacks can disable computers, disks, or network devices, causing denial of service and, in some cases, permanent data loss.

Sub-techniques:

No sub-techniques.

Real Example:

NotPetya (2017)

In **NotPetya (2017)**, the malware overwrote the Master Boot Record (MBR) and targeted firmware on infected systems, rendering computers unbootable and causing widespread denial-of-service across organizations worldwide, including critical infrastructure and multinational corporations.

Detection Methods:

- Monitor for **unexpected changes to firmware or BIOS versions** using hardware inventory and integrity tools
- Use **UEFI/BIOS integrity verification** at boot to detect tampering
- Detect **system boot failures or abnormal POST behavior**
- Monitor for **suspicious firmware update activity** or unsigned firmware installations
- Leverage **endpoint security solutions** that check firmware hashes against known-good values
- Inspect **system logs** for errors or failed firmware integrity checks
- Employ **hardware-based security features** (e.g., TPM, Secure Boot) to detect unauthorized firmware modifications

Mitigation Strategy:

Mitigation Strategy	Application	Priority
Enable Secure Boot and UEFI protections	**BIOS/UEFI settings**	**High**
Use firmware integrity monitoring and verification	**Endpoint Security, EDR, Hardware Inventory Tools**	**High**
Apply only signed and vendor-approved firmware updates	**Firmware Update Management, Vendor Tools**	**High**
Maintain regular firmware backups	**IT Asset Management, Backup Solutions**	**Medium**

Restrict administrative access to firmware update utilities	**IAM, Endpoint Security**	**High**
Monitor for abnormal system boot behavior or POST errors	**SIEM, EDR**	**High**
Implement hardware-based protections like TPM	**Hardware Security Modules, TPM**	**Medium**
Educate IT staff on safe firmware update procedures	**Security Awareness Programs**	**Medium**

Technique 2: System Shutdown/Reboot(T1529)

Description:

Adversaries may shut down or reboot systems to stop users from accessing them, cause disruption, or slow down security teams during an attack. They can do this using normal system commands, network device tools, cloud or virtual machine controls, or Windows system functions that force a shutdown or crash. Sometimes these actions make the problem look like a system failure instead of an attack. To perform this, attackers often need high-level permissions, which they may gain after breaking into the system. This technique is mainly used to interrupt services, damage systems, or make recovery and investigation harder.

Sub-techniques:

No sub-techniques.

Real Example:

During the **NotPetya ransomware attack (2017)**, the malware forced infected Windows systems into repeated shutdowns and reboots, causing widespread outages and preventing normal operation. This behavior disrupted business continuity and hindered incident response efforts across numerous organizations globally.

Detection Methods:

- Monitor Windows Event IDs (1074, 6006, 6008) for unexpected shutdowns or restarts.
- Detect execution of shutdown or reboot commands like shutdown.exe or Restart-Computer.
- Observe Windows API calls such as InitializeSystemShutdownExW, ExitWindowsEx, or NtRaiseHardError.
- Alert on unusual use of accounts with shutdown or administrative privileges.
- Analyze network device logs for abnormal reboot or reload commands via CLI or management interfaces.

Mitigation Strategy:

Application / Tool	Mitigation Strategy	Priority
Windows Group Policy	Restrict who can execute shutdown/reboot commands and enforce user permissions	High
Endpoint Detection & Response (EDR)	Monitor and alert on shutdown/reboot commands and API calls	High
SIEM (Security Information & Event Mgmt)	Correlate unexpected shutdown/reboot events with user accounts and processes	Medium
Network Device Management	Limit access to network devices and log all reload/reboot commands	High
Backup & Recovery Solutions	Ensure regular backups and disaster recovery plans are in place	High
Privileged Access Management (PAM)	Control and monitor accounts with administrative or shutdown privileges	High

MITRE ATT&CK Implementation Plan:

Phase 1: Initial Security Setup (0–30 Days)

1. Match current security tools with ATT&CK techniques
2. Find which attack techniques are not covered
3. Turn on basic system and security logs
4. Apply multi-factor authentication for remote access
5. Install endpoint protection on important systems

Phase 2: Security Improvement Stage (3–6 Months)

1. Improve security coverage for common ATT&CK techniques
2. Create alert rules based on ATT&CK attacks
3. Practice attack scenarios with the security team
4. Divide the network to reduce attack spread
5. Start basic threat hunting activities

Phase 3: Continuous Security Maturity (1–2 Years)

1. Cover almost all ATT&CK attack techniques
2. Use automation tools for faster response
3. Perform regular red team testing
4. Use threat intelligence linked to ATT&CK
5. Regularly check and improve security controls

Conclusion:

The MITRE ATT&CK Framework is a powerful tool that helps organizations understand the behavior of cyber attackers in a structured way.

Using ATT&CK, organizations can also improve their ability to detect attacks early. By knowing the signs of each technique, security teams can monitor their systems more effectively and identify suspicious activity before it causes serious harm. The framework also helps in creating better incident response plans. When an attack occurs, defenders can quickly understand what the attacker is trying to do and respond in a precise and organized way.

References and Resources

Official MITRE Resources

MITRE ATT&CK Website: https://attack.mitre.org/

ATT&CK Navigator: https://mitre-attack.github.io/attack-navigator/

ATT&CK for Enterprise: https://attack.mitre.org/matrices/enterprise/

MITRE D3FEND: D3FEND Matrix | MITRE D3FEND™

ATT&CK Evaluations: https://evals.mitre.org/

Document Information:

Title: MITRE ATT&CK Framework - Tactics, Techniques, and Procedures Guide

Version: 1.0

Last Updated: December 2025

Framework Version: ATT&CK v18

Classification: Unclassified / Public

Distribution: Unlimited

For update and latest information, visit: https://attack.mitre.org//

End of Document