Threat Intelligence – Proof of Concept (PoC)

NAME :- MISHTHI NILESH CHAVAN

INTERN ID :- 368

- **Introduction :-**

Threat intelligence (TI) is the practice of collecting, analyzing and using information about threat actors, their motivations, tools, techniques, and infrastructure to inform defensive actions. A PoC built on TI demonstrates how attacks progress (the attacker's kill chain / MITRE ATT&CK tactics), what telemetry you should collect, how to detect the activity, and which mitigations work best.

- **What is Threat Intelligence?**

Definition (simple): Threat intelligence is processed information about cyber threats that helps organizations make decisions to prevent, detect or respond to attacks.

- **Types of Threat Intelligence**

- **Strategic:** High-level, business-focused (executive briefings, trends, geopolitical risk).
- **Operational:** Information about ongoing campaigns and adversary capabilities.
- **Tactical:** TTPs (how attackers operate) — useful to blue/red teams.
- **Technical:** IOCs like IPs, domains, file hashes — good for blocking and detection rules.

- **Threat Intelligence Lifecycle**

1. **Requirements & Planning** — what intelligence do we need?

2. **Collection** — gather from logs, sensors, open sources, feeds.
3. **Processing** — normalize, enrich, dedupe data.
4. **Analysis** — interpret, pivot, identify relationships.
5. **Dissemination** — send to SOC, IR, executives with different formats.
6. **Feedback** — measure usefulness and revise requirements.

- **Objective :-**

  - **Objective :** Demonstrate how attackers move through the MITRE ATT&CK tactics to achieve their goals, show step-by-step examples, capture detection opportunities, and provide mitigations and recommendations to reduce organizational risk.

- **Methodology & Tools :-**

  - **Methodology**

1. Literature review (MITRE, vendor writeups, public reports).
2. Mapping known TTPs to logs and controls.
3. Creating step-by-step example procedures (non-destructive).
4. Developing detection idea rules and mitigation recommendations.
5. Building a mini lab simulation (isolated VMs) to show telemetry.

  - **Safety & Ethics**

- Never run real malware on production systems.
- Only simulate or explain steps, or execute benign equivalents in an isolated lab (air-gapped or NATed) with explicit permission.
- Provide proper disclaimers in the report.

- **Overview of MITRE ATT&CK**

MITRE ATT&CK is a knowledge base of observed adversary behaviors. It is organized by **tactics** (the goal) and **techniques/sub-techniques** (how the goal is achieved). This PoC applies the ATT&CK tactic taxonomy to create realistic, defensible detection and mitigation recommendations.

- ## Detailed Tactic-by-Tactic Analysis (All 14):-

  1. Reconnaissance (TA0043) Overview: Passive and active collection of publicly available or target-specific information used to plan an intrusion. Why attackers use this: To map targets, discover people to spear-phish, find exposed services, and identify vulnerabilities. Common techniques -

- **Search open sources (Google dorking)**
- **Subdomain enumeration (crt.sh, certificate transparency)**
- **Social media profiling (LinkedIn, Twitter)**
- **Network scanning (Nmap, Masscan)**

Example procedure (step-by-step)

1. **Use whois to check domain registration data.**
2. **Use crt.sh or certspotter to find subdomains from certificate transparency logs.**
3. **Run amass or subfinder to enumerate subdomains.**
4. **Use theHarvester to collect email addresses.**
5. **Scan discovered hosts with nmap -sS -p- -T4 target.com (lab only).**

Adversary tools

- **amass, subfinder, theHarvester, Nmap, Masscan, Google dork lists.**

Detection signals & logs

- **High volume of DNS queries for multiple subdomains within short windows.**
- **External IPs performing repeated requests to /.git or /.env etc.**
- **Web server logs showing many unique user-agents flagged as scanners.**

Mitigations

- **Limit public exposure of sensitive files; remove internal docs from public sites.**
- **Use rate limiting and WAF rules to block scanning patterns.**
- **Monitor for unusual DNS query spikes.**

Limitations

- **Passive recon is hard to detect (they use public sources). Detection mostly possible for active scanning.**

Example IOCs

- **Suspicious domains similar to company names.**
- **IPs from known scanning services performing many head requests.**

Sample detection rule idea

- **Alert when external IPs request more than X distinct subdomains or more than Y probing URIs in Z minutes.**

2 . Resource Development (TA0042)

Overview: Activities where attackers prepare infrastructure and capabilities — domains, hosting, email accounts, or malware.

Why used: Modern attacks often require long-term infrastructure: C2 domains, phishing domains, cloud storage, or build environments.

Common techniques

- **Register domains (typosquatting).**
- **Create accounts on social platforms.**
- **Acquire hosting, buy VPS.**
- **Develop or acquire malware.**

Example procedure

1. **Register target-portal[.]com via a registrar.**
2. **Point DNS to attacker VPS and obtain a Let's Encrypt cert to appear legitimate.**
3. **Create fake LinkedIn profiles to connect with staff.**
4. **Upload phishing kit and configure mail server for spear-phishing campaigns.**

Adversary tools

- **Domain registrars, VPS providers, phishing kits, fraud marketplaces.**

Detection signals & logs

- **New domain registrations similar to your brand.**
- **TLS certs for subdomains you don't own.**
- **New social accounts interacting with employees.**

Mitigations

- **Use brand monitoring and domain-watching services; register high-risk variants proactively.**
- **Educate employees to verify contact origins.**
- **Blacklist or block suspicious registrars/hosting IP ranges in your environment.**

Limitations

- **Domain registration is legal; detection relies on contextual signals (typosquatting, adult content, suspicious hosting).**

IOCs

- **Domain xyz-payments[.]com created recently pointing to suspicious IPs.**

3. Initial Access (TA0001)

Overview: The means by which an attacker gains a foothold in a network or system.

Why used: Without initial access the attack chain cannot progress.

Common techniques

- **Phishing (T1566), drive-by compromise, exploiting public-facing apps (T1190), valid accounts (T1078), supply chain (T1195).**

Example procedure — Spear-phishing with malicious document (safe explanation)

1. **Create a macro-enabled Word document that runs a benign script (in lab, use a script that writes a log file).**
2. **Send via a crafted email that appears to be an invoice.**

3. When a user opens and enables macros, the script runs and connects to a lab listener (only in controlled environment).

Adversary tools

- Phishing frameworks (GoPhish), exploit kits, compromised email accounts.

Detection signals & logs

- Email gateway logs showing attachments with macros.
- Endpoint telemetry showing powershell.exe launched by winword.exe.
- Unusual child processes spawned by Office executables.

Mitigations

- Block macros from the internet by policy, enable Protected View, disable legacy macros.
- Strong email filtering & DKIM/DMARC enforcement.
- User awareness training; phishing simulations.

Limitations

- Social engineering can still bypass technical controls when users are tricked.

IOCs

- Attachments: Invoice_2025.docm, *.docm with embedded macros from external senders.
- Process chain: winword.exe -> powershell.exe -ExecutionPolicy Bypass.

4. Execution (TA0002)

Overview: Running adversary-controlled code on a target system.

Why used: Execution enables payloads, lateral movement, persistence, and data access.

Common techniques

- Command interpreters (T1059 – PowerShell, cmd), scheduled tasks (T1053), malicious scripts, exploit for client execution (T1203).

Example procedure — PowerShell downloader (lab-safe)

1. Host a benign script on a local lab HTTP server that writes a file to C:\temp\ (do not host malware).
2. Trigger execution with: powershell.exe -NoProfile -ExecutionPolicy Bypass -File \\labserver\payload.ps1 (lab only).
3. Observe process creation logs and network logs.

Adversary tools

- PowerShell Empire, Metasploit, Cobalt Strike (note: only reference in PoC).

Detection signals & logs

- PowerShell ScriptBlock logging (Event ID 4104).
- Suspicious base64 or encoded PowerShell commands (look for -EncodedCommand).
- Parent-child process anomalies (e.g., explorer.exe -> powershell.exe).

Mitigations

- Enable PowerShell logging & Constrained Language Mode.
- Block -ExecutionPolicy Bypass in endpoint policies or detect it.
- Application allowlisting.

Limitations

- Fileless techniques reduce disk artifacts — need memory/behavioral detection.

IOCs

- Process command line tokens with -ExecutionPolicy Bypass or long base64 strings.

5 Persistence (TA0003)

Overview: Methods attackers use to survive restarts and maintain access.

Why used: So attacker access persists without repeated exploitation.

Common techniques

- **Registry Run keys (T1547.001), scheduled tasks (T1053.005), service creation, startup folder, hidden accounts (T1136).**

Example procedure — Registry Run key

1. **Attacker sets: HKCU\Software\Microsoft\Windows\CurrentVersion\Run\Updater = powershell -File C:\Users\Public\updater.ps1**
2. **On every login the script runs, ensuring persistent access.**

Adversary tools

- **Persistence scripts, scheduled task utilities, service installers.**

Detection signals & logs

- **New Run keys, new scheduled tasks creation events.**
- **EDR alerts for modifications of registry keys associated with auto-start.**

Mitigations

- **Monitor registry changes via Sysmon (Event ID 13) or EDR.**
- **Restrict ability to create scheduled tasks or modify registry to admin roles only.**
- **Harden group policy to prevent arbitrary autoruns.**

Limitations

- **Some persistence can be subtle (e.g., abusing signed binaries) and hard to detect.**

IOCs

- **Registry entries pointing to unexpected PowerShell scripts, unexpected scheduled task names.**

6 Privilege Escalation (TA0004)

Overview: Gaining higher rights (admin/SYSTEM) to perform more sensitive actions.

Why used: Higher privileges allow access to protected data, install drivers, disable protections.

Common techniques

- **Exploiting unpatched vulnerabilities (T1068), token impersonation (T1134), bypassing UAC (T1548.002), process injection.**

Example procedure — UAC bypass using fodhelper (safe explanation)

1. **Add registry keys under HKCU\Software\Classes\ms-settings\Shell\Open\command to point to a benign script.**
2. **Launch fodhelper.exe to auto-elevate and run the script (lab only, harmless command).**

Adversary tools

- **Mimikatz (for token theft), privilege escalation exploit scripts.**

Detection signals & logs

- **Unexpected use of known auto-elevation binaries (fodhelper.exe, rundll32.exe) followed by suspicious commands.**
- **Event logs with process execution under elevated accounts.**

Mitigations

- **Keep systems patched; enable UAC to Always Notify.**
- **Application allowlisting for high-privilege binaries.**
- **Monitor for known UAC bypass patterns.**

Limitations

- **Zero-day escalations exist and require rapid patching and defense-in-depth.**

IOCs

- **Registry keys under ms-settings\Shell\Open\command with unknown values; presence of Mimikatz shadow artifacts.**

7. Defense Evasion (TA0005)

Overview: Techniques to bypass security controls and avoid detection.

Why used: To execute longer without being caught and to blend with normal activity.

Common techniques

- **Obfuscation (T1027), abusing signed binaries / LOLBins (T1218), disabling defenses (T1562), process injection (T1055).**

Example procedure — signed binary proxy execution

1. **Drop payload DLL and execute via rundll32.exe payload.dll,EntryPoint to leverage a signed Windows binary.**
2. **Attacker hides real activity under a trusted executable.**

Adversary tools

- **Custom packers, Veil, living-off-the-land binaries (LOLBins), Cobalt Strike.**

Detection signals & logs

- **Signed binary executing code from nonstandard locations.**
- **Unusual parent/child relationships (e.g., rundll32.exe launching unknown network connections).**
- **Sudden disabling of Windows Defender (PowerShell Set-MpPreference calls).**

Mitigations

- **Application control, block execution from temp directories, monitor for signed binary misuse.**
- **Alert on changes to AV settings.**

Limitations

- **Attackers constantly find new LOLBins and obfuscation methods; behavioral detection is necessary.**

IOCs

- **Unexpected rundll32.exe loads pointing to non-OS DLLs; base64 encoded commands in command lines.**

8. Credential Access (TA0006)

Overview: Stealing credentials (passwords, hashes, tokens) to authenticate laterally.

Why used: Credentials are re-usable and often enable deeper access without re-exploitation.

Common techniques

- **LSASS memory dumping (T1003.001), credential dumping from browsers (T1555.003), brute force (T1110), keylogging (T1056).**

Example procedure — LSASS dump (lab explanation only)

1. **Use procdump (lab with permission) to dump LSASS memory for analysis in isolated environment.**
2. **Extract credentials using mimikatz (lab only, for educational demonstration).**

Adversary tools

- **Mimikatz, procdump, web browser password recovery tools, Hydra for brute force.**

Detection signals & logs

- **Creation of LSASS process dumps, unusual process reading LSASS memory.**
- **Abnormal logon events: many failed logins, followed by successful logins.**

Mitigations

- **Enable LSASS protection (Credential Guard), disable tools that can read memory from non-privileged accounts.**
- **Use robust MFA and rotate service account passwords.**

Limitations

- **Credential theft via phishing or physical access bypasses many automated controls.**

IOCs

- **Presence of procdump.exe or mimikatz.exe running on endpoints; abnormal authentication patterns.**

9 Discovery (TA0007)

Overview: Post-compromise exploration to learn about systems, accounts, and network topology.

Why used: To identify where high-value assets are and plan next steps.

Common techniques

- **System information, network config, account enumeration, process listing.**

Example procedure

1. **Run systeminfo, net user, ipconfig /all to gather host details.**
2. **Use Nmap internally to identify other hosts and open ports (lab only).**

Adversary tools

- **built-in Windows commands, PowerShell scripts, Nmap, BloodHound (for Active Directory mapping).**

Detection signals & logs

- **Commands like net user, query user, or whoami running remotely.**
- **Lateral scanning patterns (Nmap internal scans).**

Mitigations

- **Limit ability to run privileged discovery commands, monitor for unusual enumeration activity.**
- **Endpoint restrictions on installed tools.**

Limitations

- **Discovery using legitimate admin tools may produce false positives; correlation across telemetry helps.**

IOCs

- **Logs showing many net commands executed from non-admin times or accounts.**

10. Lateral Movement (TA0008)

Overview: Methods to move from the initial host to other systems in the environment.

Why used: Access to a single host is rarely enough — attackers move to servers, domain controllers, backups.

Common techniques

- Remote services (RDP, SMB), Pass-the-Hash, PsExec, remote code execution.

Example procedure — PsExec lateral move (lab explanation)

1. Use PsExec.exe \\target -u DOMAIN\user -p password cmd (lab with permission) to run remote commands.
2. Copy tools to remote host and execute.

Adversary tools

- PsExec, RDP, WMI, PowerShell Remoting, Pass-the-Hash tools.

Detection signals & logs

- Unexpected network connections to SMB/445 or RDP sessions from internal hosts.
- Event ID 4624 (logon) correlated with suspicious source hosts.

Mitigations

- Restrict admin credentials, use privileged access workstations, network segmentation, disable unnecessary remote services.

Limitations

- Legitimate admin activity may look similar; need context and baselining.

IOCs

- SMB connections to multiple hosts from a single workstation, elevated remote logons.

11. Collection (TA0009)

Overview: Gathering and preparing targeted data for exfiltration.

Why used: Attackers focus on high-value data to meet strategic goals (espionage, financial theft).

Common techniques

- Data from local systems, network shares, screen capture, keylogging.

Example procedure

1. Search for file types: Get-ChildItem -Recurse -Include *.docx,*.xls* -Path C:\Users\ and copy into temporary folder.
2. Archive the files into a ZIP (lab safe) for later exfiltration.

Adversary tools

- Custom scripts, PowerShell, RAR/zip utilities, screen grabbing malware.

Detection signals & logs

- Large numbers of file reads in short time, new archive files, unusual access to shared directories.

Mitigations

- DLP (Data Loss Prevention) policies, file access monitoring, restrict access to sensitive directories.

Limitations

- Encryption of files at rest won’t prevent exfiltration if attacker has proper access.

IOCs

- Unexpected ZIP files in temp folders, scheduled tasks running file collection scripts.

12. Command and Control (TA0011)

Overview: Remote control channels between compromised hosts and operator infrastructure.

Why used: Command & Control (C2) allows the attacker to run commands, update malware, and orchestrate actions.

Common techniques

- C2 over HTTP/S, custom TCP, DNS tunneling, legitimate remote access tools.

Example procedure — HTTPS C2 (explanation)

1. Malware periodically posts to https://commandserver.example/poll and receives commands; data looks like normal HTTPS traffic but to suspicious domains.

Adversary tools

- Cobalt Strike, custom RATs, DNS tunneling tools.

Detection signals & logs

- Unusual periodic outbound connections to uncommon domains, small encrypted beacons at regular intervals, DNS requests with long or encoded subdomain strings.

Mitigations

- Block or proxy connections to unknown or unusual domains.
- Use egress filtering, HTTPS inspection (where policy allows), and DNS logging.

Limitations

- Encrypted HTTPS channels and use of legitimate cloud services make detection harder.

IOCs

- Domains not in normal allowlists; beaconing patterns (regular intervals).

13 Exfiltration (TA0010)

Overview: Moving stolen data out of the target network.

Why used: This is the final goal in many theft cases — transfer copies of sensitive data to attacker control.

Common techniques

- Exfil over C2 (T1041), cloud storage uploads, FTP, email.

Example procedure — Exfil via cloud storage

1. **Archive C:\Temp\leak.zip and upload to Google Drive / Dropbox using API keys.**
2. **Attacker collects from cloud storage.**

Adversary tools

- **Scripts using cloud APIs, FTP clients, stealthy HTTP POSTs.**

Detection signals & logs

- **Large outbound uploads to cloud storage.**
- **Unusual API calls from internal accounts.**

Mitigations

- **DLP, restrict uploads from endpoints to unapproved cloud apps, monitor cloud API activity.**

Limitations

- **Use of legitimate cloud providers complicates blocking; need context-aware DLP.**

IOCs

- **Uploads to cloud storage from non-business accounts or during odd hours.**

14 Impact (TA0040)

Overview: Adversary actions that manipulate, interrupt or destroy systems and data (ransomware, DoS, data destruction).

Why used: To disrupt business operations, extort organizations, or sabotage targets.

Common techniques

- **Data encrypted for impact (ransomware), data destruction, service disruption.**

Example procedure — Simulated file encryption (lab)

1. **In lab, run a script that renames files to .encrypted after making safe copies. Drop a ransom note (text file) — always simulated and reversible.**

Adversary tools

- **Ransomware families (WannaCry, Ryuk) — in PoC reference only, never run on production.**

Detection signals & logs

- **Mass file rename events, spike in file write/modify operations, deletion of shadow copies, suspicious encryption-like writes.**

Mitigations

- **Offline backups, air-gapped copies, rapid detection & isolation, maintain immutable backups.**

Limitations

- **Fast-moving ransomware can encrypt backups if they are reachable; test backup isolation.**

IOCs

- **Creation of HOW_TO_DECRYPT.txt, mass file activity to many user folders.**

➔ Detection Rules & Log Sources (SIEM/EDR Ideas)

- Event sources to collect

- Windows Event Logs (Security, System, Application)
- Sysmon (ProcessCreate, NetworkConnect, Registry events)
- EDR telemetry (process injection, fileless executions)
- DNS logs, proxy logs, firewall logs, cloud provider API logs

- Example detection rules

- Alert: powershell.exe with "-ExecutionPolicy Bypass" or -EncodedCommand.
- Alert: winword.exe spawning powershell.exe.
- Alert: Multiple distinct DNS queries to unknown subdomains within 60 minutes.
- Alert: Outbound connections to newly created/rare TLS certificates.

- For each rule include: data sources, suggested thresholds, false positive notes, and recommended response playbook (isolate host, gather memory snapshot, block IP).

➔ Challenges, Limitations, Ethics & Safety :-

Challenges

- High false positives from legitimate admin activity.
- Encrypted traffic reduces visibility.
- Attackers using legitimate services (GitHub, Dropbox) for C2/exfiltration.

Limitations

- Detection and mitigation require investment in telemetry and human analysis.
- PoC cannot fully replicate real attackers' stealth — realistic threats change constantly.

Ethics & Safety

- Never run live malware — educational references only.
- All testing must have documented approval and use isolated test en **Conclusion & Recommendations**

➔ Conclusion:-

This PoC demonstrates the full ATT&CK tactic spectrum. Defense requires a layered approach: strong identity security (MFA, password hygiene), endpoint telemetry and EDR, network monitoring (DNS, proxy), hardened configurations, user training, and tested backup/recovery procedures..