ZeroMagenta AI Triage Tool

AI-Powered Security Detection Pipeline

Cybersecurity Lab Portfolio | Sanitized for Public Sharing

This project documents the design and development of an AI-powered security triage pipeline built from scratch. Using an isolated Windows 10 lab environment, MITRE ATT&CK techniques were simulated with Atomic Red Team and captured by Sysmon. The resulting telemetry was ingested by a Python-based tool that uses Google Gemini AI to automatically triage alerts, enrich findings with MITRE ATT&CK framework data, and map defensive countermeasures using the MITRE D3FEND framework. The tool was built iteratively across five versions, each one solving a real limitation of the previous. An open-source contribution was made to the attackcti Python library after identifying a data gap during development.

TOOLS USED

- Atomic Red Team (Red Canary) -- Attack simulation framework
- Sysmon v15.20 (Microsoft Sysinternals) -- Endpoint telemetry capture
- SwiftOnSecurity Sysmon Config -- Community detection configuration
- Google Gemini API (gemini-2.5-flash) -- AI-powered alert triage
- MITRE ATT&CK via attackcti Python library -- Technique enrichment
- MITRE D3FEND REST API -- Defensive countermeasure mapping
- Python 3.14 -- Pipeline development
- VMware Workstation Pro 17 -- Virtualization
- Windows 10 Pro (isolated VM) -- Lab endpoint
- Windows 11 Pro (host machine) -- Script execution environment

LAB ENVIRONMENT

All attack simulations were conducted in an isolated Windows 10 VM running on VMware Workstation Pro 17. The VM was configured with Host-Only networking, preventing any communication with the internet or the host machine network. Sysmon was installed using the SwiftOnSecurity community configuration to capture meaningful endpoint telemetry. Atomic Red Team was installed to simulate MITRE ATT&CK techniques. The Python triage scripts ran on the host machine, with Sysmon XML exports transferred via VMware Tools drag-and-drop.

All screenshots show simulated lab environment data. Sensitive details have been redacted.

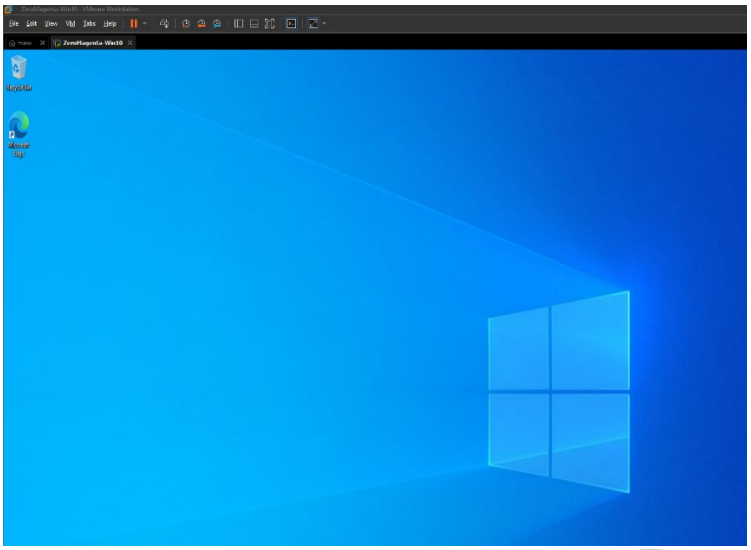


Figure 1 ZeroMagenta-Win10 isolated Windows 10 VM running in VMWare Workstation Pro 17. Host only networking enabled for full isolation during attack simulation.

```
PS C:\Windows\system32> C:\Users\zm_user\Downloads\Sysmon\Sysmon64.exe -accepteula -i C:\sysmonconfig.xml


System Monitor v15.20 - System activity monitor
By Mark Russinovich and Thomas Garnier
Copyright (C) 2014-2026 Microsoft Corporation
Using libxml2. libxml2 is Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.
Sysinternals - www.sysinternals.com

Loading configuration file with schema version 4.50
Sysmon schema version: 4.91
Configuration file validated.
Sysmon64 installed.
SysmonDrv installed.
Starting SysmonDrv.
SysmonDrv started.
Starting Sysmon64..
Sysmon64 started.
PS C:\Windows\system32>
```

Figure 2 Sysmon v15.20 installation using SwiftOnSecurity community configuration. Configuration validated and all Sysmon services started successfully.

METHODS

Attack Chain Simulation

A ten-technique attack chain was executed using Atomic Red Team, designed to simulate the sequence of actions an attacker takes after gaining initial access to an endpoint:

1. T1033 -- System Owner and User Discovery
2. T1082 -- System Information Discovery
3. T1016 -- System Network Configuration Discovery (including QakBot recon simulation)
4. T1049 -- System Network Connections Discovery
5. T1057 -- Process Discovery
6. T1012 -- Query Registry
7. T1053.005 -- Scheduled Task Persistence
8. T1547.001 -- Registry Run Key Persistence
9. T1562.001 -- Disable or Modify Security Tools
10. T1003.001 -- OS Credential Dumping via LSASS

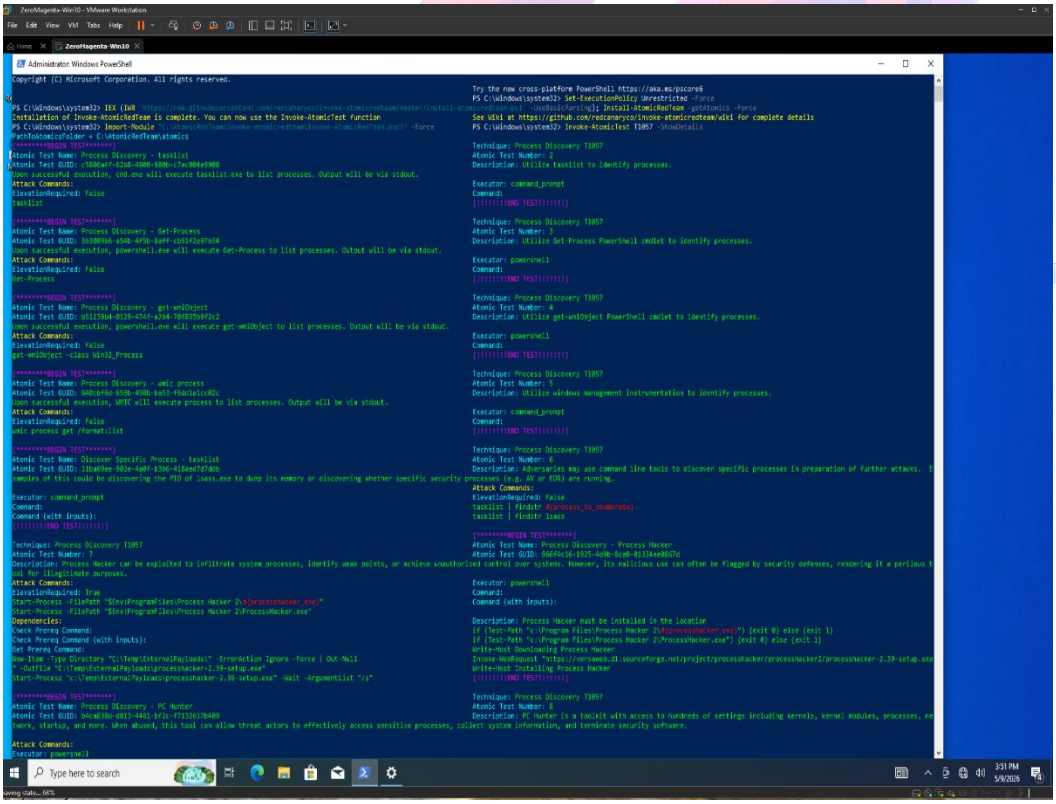


Figure 3 Atomic Red Team installed and operational. ShowDetails output for T1057 Process Discovery displaying all available sub-technique variations with attack commands and executors.

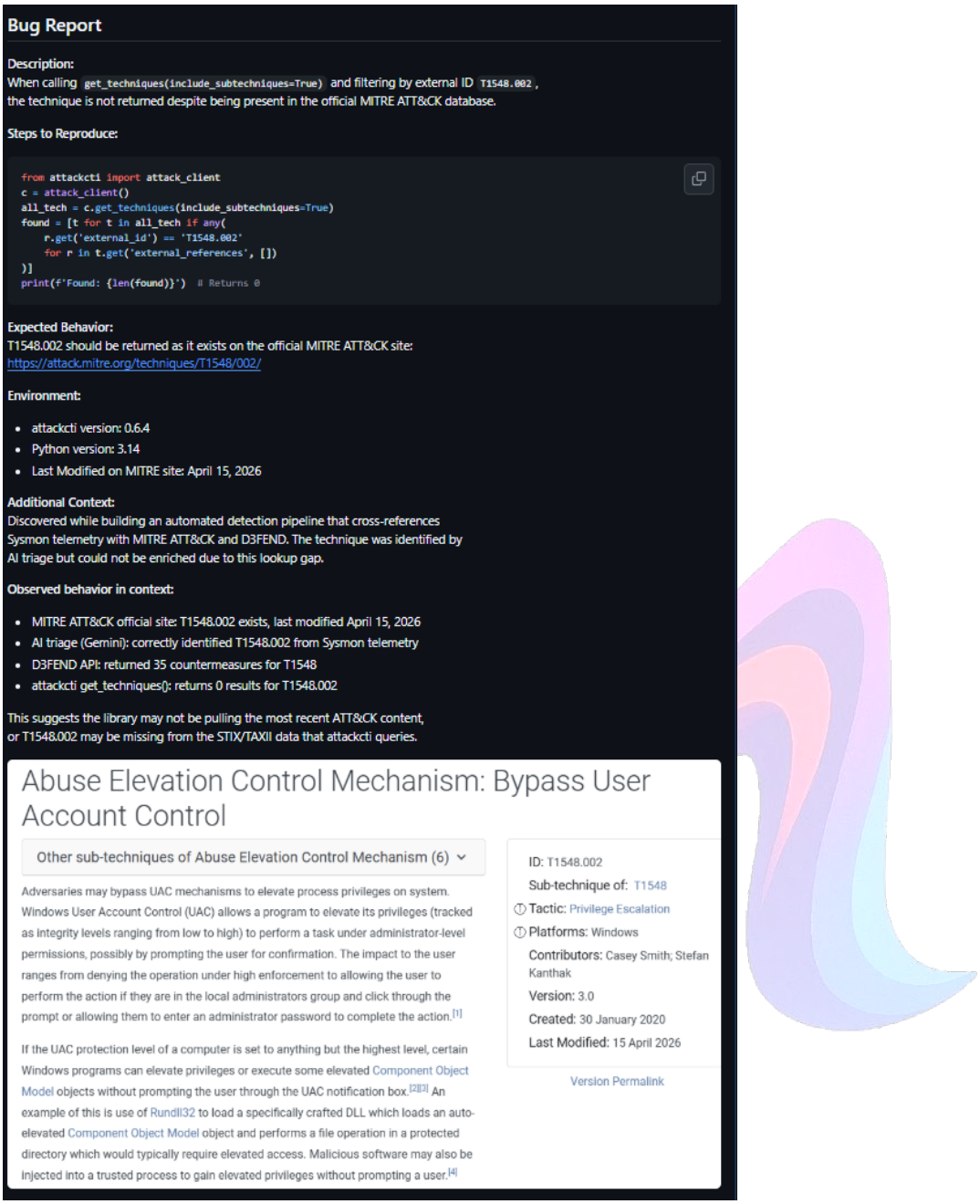
Pipeline Development

The triage tool was built across five versions, each addressing a specific limitation:

- V1 -- Hardcoded mock alert based on T1057 output sent to Gemini AI. Proved the triage concept end to end.
- V2 -- Dynamic input accepted from terminal. Added error handling and modular function structure.
- V3 -- Real Sysmon telemetry ingested automatically. XML export parsed, filtered for suspicious activity using keyword matching, and sent to AI for triage.
- V4 -- MITRE ATT&CK API enrichment added via attackcti library. Each identified technique automatically enriched with tactic, detection guidance, and reference URL.
- V5 -- MITRE D3FEND integration added. Each detected technique automatically mapped to defensive countermeasures via the D3FEND REST API. Full JSON report generated.

Open Source Contribution

During V4 development, T1548.002 (Bypass UAC) was identified in AI triage output and D3FEND results but returned zero results from the attackcti library. After verifying the technique exists on the official MITRE ATT&CK site (last modified April 15, 2026) and confirming the latest version of attackcti (0.6.4) was installed, a bug report was submitted to the OTRF/ATTACK-Python-Client repository on GitHub. No existing issue had been filed for this gap.



Bug Report

Description:
When calling `get_techniques(include_subtechniques=True)` and filtering by external ID `T1548.002`, the technique is not returned despite being present in the official MITRE ATT&CK database.

Steps to Reproduce:

```
from attackcti import attack_client
c = attack_client()
all_tech = c.get_techniques(include_subtechniques=True)
found = [t for t in all_tech if any(
    r.get('external_id') == 'T1548.002'
    for r in t.get('external_references', [])
)]
print(f'Found: {len(found)}')  # Returns 0
```

Expected Behavior:
T1548.002 should be returned as it exists on the official MITRE ATT&CK site:
https://attack.mitre.org/techniques/T1548/002/

Environment:

- attackcti version: 0.6.4
- Python version: 3.14
- Last Modified on MITRE site: April 15, 2026

Additional Context:
Discovered while building an automated detection pipeline that cross-references Sysmon telemetry with MITRE ATT&CK and D3FEND. The technique was identified by AI triage but could not be enriched due to this lookup gap.

Observed behavior in context:

- MITRE ATT&CK official site: T1548.002 exists, last modified April 15, 2026
- AI triage (Gemini): correctly identified T1548.002 from Sysmon telemetry
- D3FEND API: returned 35 countermeasures for T1548
- attackcti get_techniques(): returns 0 results for T1548.002

This suggests the library may not be pulling the most recent ATT&CK content, or T1548.002 may be missing from the STIX/TAXII data that attackcti queries.

Abuse Elevation Control Mechanism: Bypass User Account Control

Other sub-techniques of Abuse Elevation Control Mechanism (6)

Adversaries may bypass UAC mechanisms to elevate process privileges on system. Windows User Account Control (UAC) allows a program to elevate its privileges (tracked as integrity levels ranging from low to high) to perform a task under administrator-level permissions, possibly by prompting the user for confirmation. The impact to the user ranges from denying the operation under high enforcement to allowing the user to perform the action if they are in the local administrators group and click through the prompt or allowing them to enter an administrator password to complete the action.[1]

If the UAC protection level of a computer is set to anything but the highest level, certain Windows programs can elevate privileges or execute some elevated Component Object Model objects without prompting the user through the UAC notification box.[2][3] An example of this is use of Rundll32 to load a specifically crafted DLL which loads an auto-elevated Component Object Model object and performs a file operation in a protected directory which would typically require elevated access. Malicious software may also be injected into a trusted process to gain elevated privileges without prompting a user.[4]

ID: T1548.002
Sub-technique of: T1548
Tactic: Privilege Escalation
Platforms: Windows
Contributors: Casey Smith; Stefan Kanthak
Version: 3.0
Created: 30 January 2020
Last Modified: 15 April 2026

Version Permalink

Figure 4 Bug report submitted to OTRF/ATTACK-Python-Client GitHub repository. T1548.002 confirmed present on official MITRE ATT&CK site but returning zero results from attackcti v0.6.4.

KEY RESULTS

1. Living off the Land vs Signature Detection

Running the attack chain demonstrated the fundamental gap between signature-based and behavioral detection. Techniques using legitimate Windows tools -- tasklist, systeminfo, whoami, reg query, ipconfig -- passed through Windows Defender completely undetected. Mimikatz was blocked immediately upon execution. The attempt to disable Defender via T1562 failed silently due to Tamper Protection. This confirmed that Living off the Land techniques require behavioral detection, not signature matching, to identify.

Notable observation: T1057 Process Discovery launched Windows Task Manager as one of its sub-techniques. T1547 Boot or Logon Autostart Execution triggered the Windows Credential Manager popup, demonstrating real credential access behavior using a built-in Windows interface. Both actions used legitimate Windows tools and generated no Defender alerts.

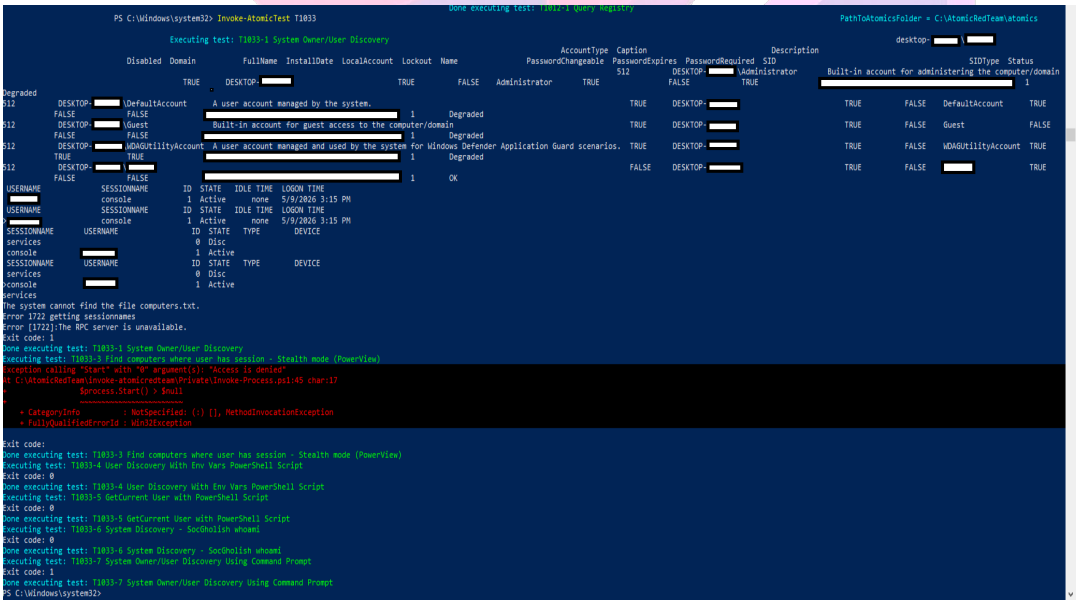


Figure 5 T1033 System Owner and User Discovery executing. Windows Defender PowerView stealth mode attempt (red text) while native discovery commands succeeded. Simulated lab environment, sensitive details redacted.

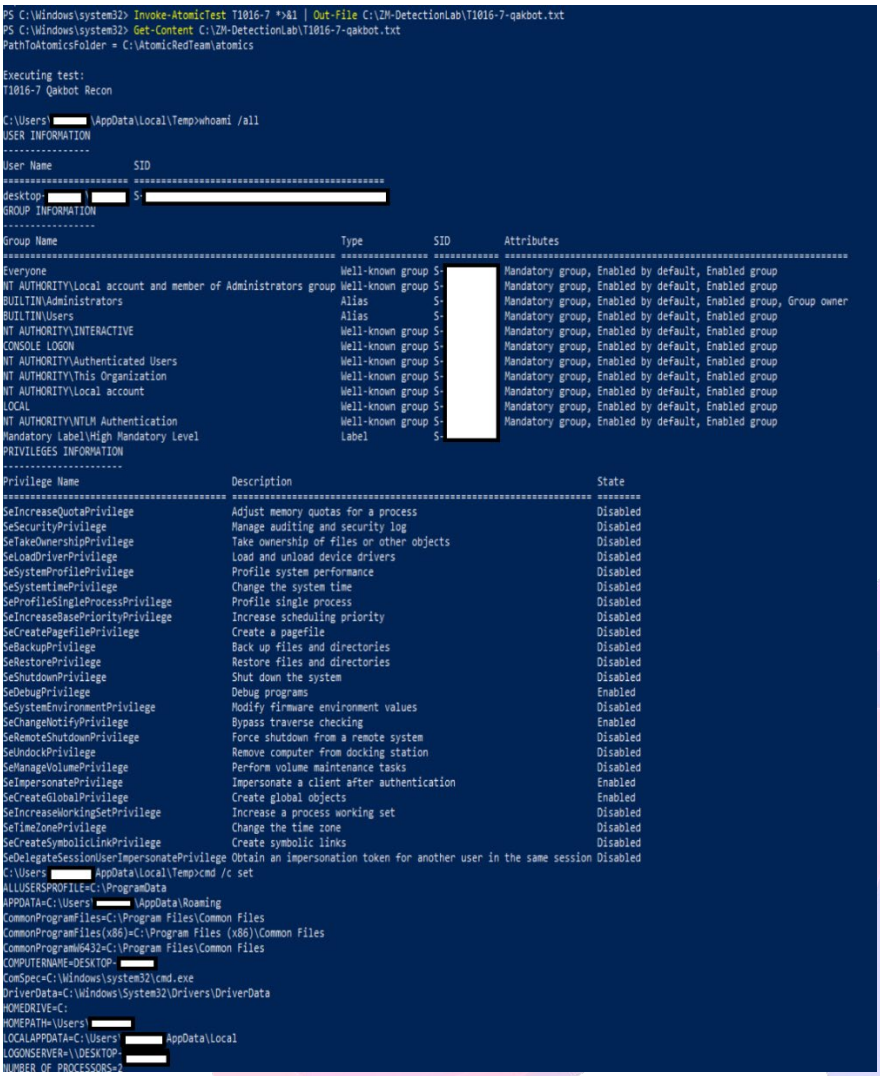


Figure 6 T1016-7 QakBot recon simulation executing whoami /all. Output shows full privilege enumeration including SeDebugPrivilege and SeImpersonatePrivilege enabled – high value targets for credential theft and privilege escalation. Simulated lab environment, sensitive details redacted.

2. V5 Pipeline Output

The final V5 pipeline ingested 2,077 Sysmon events from the attack chain. The AI triage output identified:

- Overall severity: High
- 10 MITRE ATT&CK techniques correctly identified including T1059.001 PowerShell, T1082 System Information Discovery, T1016 Network Configuration Discovery, T1057 Process Discovery, T1003 OS Credential Dumping, T1105 Ingress Tool Transfer, T1558.003 Kerberoasting
- QakBot recon chain correctly identified from C:\AtomicRedTeam\atomics\T1016\src\qakbot.bat
- Offensive tools detected: SharpView, Seatbelt, SharpUp, SharpWatson downloaded via PowerShell from GitHub
- Attack narrative generated describing coordinated reconnaissance and privilege escalation preparation
- MITRE ATT&CK enrichment: all 10 techniques enriched with tactic classification and reference URLs
- D3FEND countermeasures mapped: T1003 returned 48 countermeasures, T1033 returned 31, T1016 returned 19

Figure 7 V5 complete pipeline output. 2,077 Sysmon events ingested, High severity assigned, 10 MITRE ATT&CK techniques identified including QakBot recon chain, offensive tool ingress from GitHub, and Kerberoasting attempt. MITRE ATT&CK enrichment connecting. Simulated lab environment, sensitive details redacted.

```
============================================================
MITRE ATT&CK ENRICHMENT
============================================================

T1518.001: Security Software Discovery
  Tactic: discovery
  URL: https://attack.mitre.org/techniques/T1518/001/

T1016: System Network Configuration Discovery
  Tactic: discovery
  URL: https://attack.mitre.org/techniques/T1016/

T1082: System Information Discovery
  Tactic: discovery
  URL: https://attack.mitre.org/techniques/T1082/

T1057: Process Discovery
  Tactic: discovery
  URL: https://attack.mitre.org/techniques/T1057/

T1003: OS Credential Dumping
  Tactic: credential-access
  URL: https://attack.mitre.org/techniques/T1003/

T1105: Ingress Tool Transfer
  Tactic: command-and-control
  URL: https://attack.mitre.org/techniques/T1105/

T1059.003: Windows Command Shell
  Tactic: execution
  URL: https://attack.mitre.org/techniques/T1059/003/

T1059.001: PowerShell
  Tactic: execution
  URL: https://attack.mitre.org/techniques/T1059/001/

T1033: System Owner/User Discovery
  Tactic: discovery
  URL: https://attack.mitre.org/techniques/T1033/

T1204.002: Malicious File
  Tactic: execution
  URL: https://attack.mitre.org/techniques/T1204/002/

Querying MITRE D3FEND for defensive countermeasures...
  T1518.001: 5 countermeasures found
  T1016: 19 countermeasures found
  T1082: 7 countermeasures found
  T1057: 6 countermeasures found
  T1003: 48 countermeasures found
  T1105: 11 countermeasures found
  T1059.003: 15 countermeasures found
  T1059.001: 15 countermeasures found
  T1033: 31 countermeasures found
  T1204.002: 30 countermeasures found
```

Figure 8 V5 MITRE ATT&CK enrichment section. All 10 identified techniques automatically enriched with tactic classification and direct reference URLs via the attackcti library.

```
============================================================
D3FEND DEFENSIVE COUNTERMEASURES
============================================================

T1518.001 — 5 total countermeasures available
  [MODEL] Configuration Inventory
  [DETECT] System Call Analysis
  [RESTORE] Restore Configuration
  [ISOLATE] System Call Filtering
  [ISOLATE] Content Quarantine
  Full details: https://d3fend.mitre.org/offensive-technique/attack/T1518/

T1016 — 19 total countermeasures available
  [DETECT] File Analysis
  [DETECT] File Integrity Monitoring
  [DETECT] Dynamic Analysis
  [DETECT] Emulated File Analysis
  [DETECT] System Call Analysis
  Full details: https://d3fend.mitre.org/offensive-technique/attack/T1016/

T1082 — 7 total countermeasures available
  [DETECT] System Call Analysis
  [DETECT] Process Spawn Analysis
  [DECEIVE] Decoy Environment
  [ISOLATE] System Call Filtering
  [ISOLATE] Executable Allowlisting
  Full details: https://d3fend.mitre.org/offensive-technique/attack/T1082/

T1057 — 6 total countermeasures available
  [DETECT] System Call Analysis
  [DETECT] Process Spawn Analysis
  [ISOLATE] System Call Filtering
  [ISOLATE] Executable Allowlisting
  [ISOLATE] Executable Denylisting
  Full details: https://d3fend.mitre.org/offensive-technique/attack/T1057/

T1003 — 48 total countermeasures available
  [MODEL] Data Inventory
  [DETECT] Process Lineage Analysis
  [DETECT] Process Self-Modification Detection
  [DETECT] Process Spawn Analysis
  [DETECT] File Analysis
  Full details: https://d3fend.mitre.org/offensive-technique/attack/T1003/

T1105 — 11 total countermeasures available
  [DETECT] User Geolocation Logon Pattern Analysis
  [DETECT] Protocol Metadata Anomaly Detection
  [DETECT] Client-server Payload Profiling
  [DETECT] Per Host Download-Upload Ratio Analysis
  [DETECT] Network Traffic Signature Analysis
  Full details: https://d3fend.mitre.org/offensive-technique/attack/T1105/

T1059.003 — 15 total countermeasures available
  [DETECT] File Analysis
  [DETECT] File Integrity Monitoring
  [DETECT] Dynamic Analysis
  [DETECT] Emulated File Analysis
  [EVICT] File Eviction
  Full details: https://d3fend.mitre.org/offensive-technique/attack/T1059/

T1059.001 — 15 total countermeasures available
  [DETECT] File Analysis
  [DETECT] File Integrity Monitoring
  [DETECT] Dynamic Analysis
  [DETECT] Emulated File Analysis
  [EVICT] File Eviction
  Full details: https://d3fend.mitre.org/offensive-technique/attack/T1059/

T1033 — 31 total countermeasures available
  [MODEL] Data Inventory
  [DETECT] File Analysis
  [DETECT] File Integrity Monitoring
  [DETECT] System Call Analysis
  [DETECT] Process Lineage Analysis
  Full details: https://d3fend.mitre.org/offensive-technique/attack/T1033/

T1204.002 — 30 total countermeasures available
  [DETECT] Relay Pattern Analysis
  [DETECT] User Geolocation Logon Pattern Analysis
  [DETECT] Protocol Metadata Anomaly Detection
  [DETECT] Client-server Payload Profiling
  [DETECT] Per Host Download-Upload Ratio Analysis
  Full details: https://d3fend.mitre.org/offensive-technique/attack/T1204/
```

Figure 9 V5 MITRE D3FEND defensive countermeasure mapping. Each detected technique automatically queried against D3FEND API returning countermeasures -- T1003 OS Credential Dumping returned 48 countermeasures, T1033 returned 31.

3. Detection Engineering Limitation

The keyword-based filtering approach has a known gap. Sophisticated attackers obfuscate commands using Base64 encoding or PowerShell aliases that produce identical results without triggering keyword matches. The AI analyzes only what the filter passes through. This limitation is documented intentionally -- understanding where a detection tool fails is as valuable as understanding where it succeeds. Future versions will implement behavioral chain analysis to detect suspicious sequences regardless of individual command content

KEY TAKEAWAYS

- Demonstrated end-to-end purple team workflow: attack simulation, telemetry capture, AI-assisted triage, and defensive countermeasure mapping in one automated pipeline
- Confirmed Living off the Land behavior in practice -- legitimate Windows tools used for reconnaissance evaded Defender while known malware tools were caught immediately
- Integrated two MITRE frameworks (ATT&CK and D3FEND) into a single automated enrichment pipeline, moving from raw telemetry to structured security intelligence without manual analysis
- Identified and reported a data gap in the open source attackcti library where T1548.002 returns zero results despite existing in the official MITRE ATT&CK database
- Built iteratively across five versions with each version solving a real limitation -- from hardcoded proof of concept to automated telemetry ingestion with multi-framework enrichment
- Dynamic input architecture demonstrates the difference between a demo script and a reusable tool -- V2 accepting any alert versus V1 hardcoded to one specific scenario

Source Code: github.com/gracia-villarreal/zeromagenta-ai-triage