



Automated Threat Intelligence Pipeline: Wazuh + MISP + IRIS

Overview

I designed and implemented a real-time **Network Detection and Response (NDR)** workflow. This project automates the cross-referencing of endpoint file activity against global threat intelligence feeds to significantly reduce the "Mean Time to Detect" (MTTD).

The Technology Stack

- **Wazuh (XDR/SIEM):** Monitored endpoints via File Integrity Monitoring (FIM).
- **MISP (Threat Intel):** Served as the source of truth for malicious MD5/SHA256 hashes.
- **Python 3:** A custom bridge script connecting Wazuh alerts to the MISP REST API.
- **IRIS (Incident Response):** Automated ticket generation for high-severity matches.

Technical Implementation

1. Real-Time Detection (FIM)

Configured Wazuh to monitor specific directories. As seen in my logs, Rule **554** ("File added to the system") triggers immediately upon file creation.

Document Details		View surrounding documents	View single document
Table	JSON		
<code>r _index</code>	wazuh-alerts-4.x-2026.02.01		
<code>r agent.id</code>	004		
<code>r agent.ip</code>	192.168.110.158		
<code>r agent.name</code>	MISP		
<code>r data.integration</code>	misp_file_hashes		
<code>r data.misp.event_id</code>	130		
<code>r data.misp.found</code>	1		
<code>r data.misp.info</code>	44d88612fea8a8f36de82e1278abb02f		
<code>r data.misp.permalink</code>	https://192.168.110.158/events/view/130		
<code>r decoder.name</code>	json		
<code>r full_log</code>	{ "misp": { "found": 1, "event_id": "130", "info": "44d88612fea8a8f36de82e1278abb02f", "permalink": "https://192.168.110.158/events/view/130"}, "integration": "misp_file_hashes" }		
<code>r id</code>	1769928223.87408		
<code>r input.type</code>	log		
<code>r location</code>	misp		
<code>r manager.name</code>	wazuh		
<code>r rule.description</code>	MISP: file hash matched		
<code># rule.firedtimes</code>	1		
<code>r rule.groups</code>	misp, malware		
<code>r rule.id</code>	100802		
<code># rule.level</code>	12		
<code>r rule.mail</code>	true		
<code>r timestamp</code>	Feb 1, 2026 @ 12:43:43.193		

2. Automated Intelligence Querying

I developed a Python integration script that intercepts the FIM alert, extracts the MD5 hash, and queries MISP.

- **Challenge:** Standard integrations often fail due to SSL certificate verification in lab environments.
- **Solution:** Optimized the script to handle self-signed certificates and implemented a fallback logic to prevent "NoneType" crashes.

```
root@wazuh:~# grep "MISP" /var/ossec/logs/integrations.log
MISP_DEBUG: Querying MISP for hash: 44d88612fea8a8f36de82e1278abb02f
MISP_DEBUG: Match found and sent to Wazuh!
MISP_DEBUG: Querying MISP for hash: 44d88612fea8a8f36de82e1278abb02f
MISP_DEBUG: Match found and sent to Wazuh!
MISP_DEBUG: Querying MISP for hash: 44d88612fea8a8f36de82e1278abb02f
MISP_DEBUG: Match found and sent to Wazuh!
MISP_DEBUG: Querying MISP for hash: 44d88612fea8a8f36de82e1278abb02f
MISP_DEBUG: Match found and sent to Wazuh!
MISP_DEBUG: Querying MISP for hash: 44d88612fea8a8f36de82e1278abb02f
MISP_DEBUG: Match found and sent to Wazuh!
MISP_DEBUG: Querying MISP for hash: 44d88612fea8a8f36de82e1278abb02f
MISP_DEBUG: Match found and sent to Wazuh!
root@wazuh:~#
```

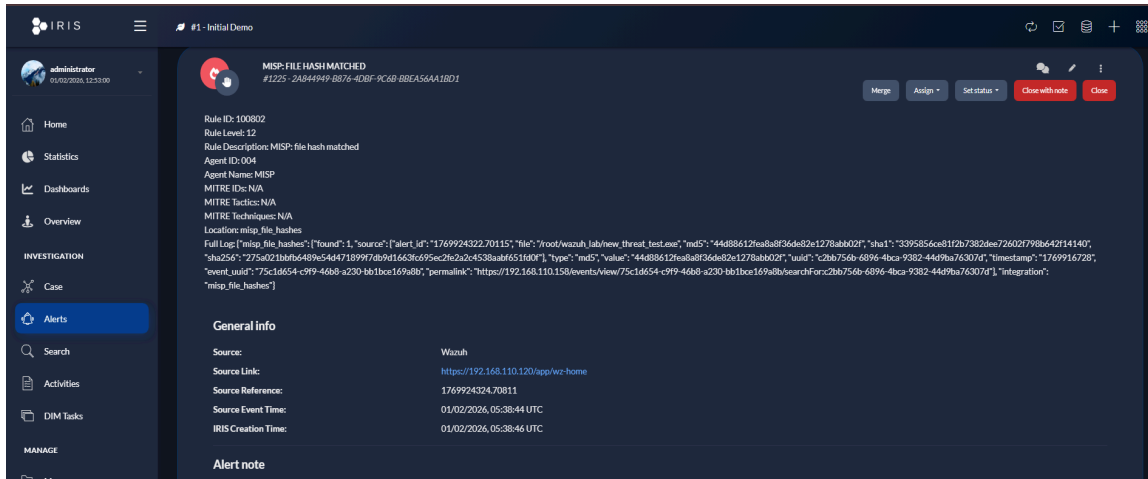
3. Custom Alert Engineering

I authored a custom ruleset ([Rule 100802](#)) that elevates any hash match to a **Level 12 (High Severity)** alert.

- **Rule Logic:** If `misp.found == 1`, generate a critical alert including the MISP Permalink and Event ID.

4. Automated Incident Response (IRIS Integration)

The final stage of the pipeline automatically pushes critical alerts to the **IRIS dashboard**. This allows security analysts to see the full "Forensic Story" (MD5, file path, and threat category) in one place.



Key Achievements

- **Automated Intelligence:** Successfully integrated feeds from **MalwareBazaar** and **CIRCL OSINT**.
- **SOC Optimization:** Level 0 "noise" is filtered out, ensuring analysts only see Level 10+ alerts in IRIS.
- **MITRE Mapping:** Mapped file execution alerts to **MITRE ATT&CK T1204.002** (Malicious File).