Windows Host IDS (HIDS)

Features

Feature	Description
File Integrity Monitoring (FIM)	Detects new/modified/deleted critical files using SHA256 hashes
Process Monitoring	Detects suspicious processes & unusual CPU/memory usage
Registry Monitoring	Watches persistence keys and critical system changes
Windows Event Log Analysis	Detects failed logins, service changes, firewall modifications
Network Monitoring	Tracks new outbound connections, suspicious IPs, and ports
Alerts	Logs, email notifications, and desktop pop-ups
Dashboard	Flask + Chart.js for visual monitoring in real-time
Docker Deployment	Runs HIDS in a container for portability & demo purposes

Tech Stack

- Python = core monitoring scripts
- Flask = real-time web dashboard
- SQLite = logs storage
- watchdog / psutil / winreg / pywin32 = monitoring & event parsing
- Docker = containerize the app
- HTML / Chart.js = dashboard charts

Architecture Overview

Purpose

- Shows Windows security expertise
- Demonstrates real-time monitoring & alerting
- Shows ability to build dashboards & deploy using Docker
- Ready to put on GitHub with README + screenshots
- Strong talking point for interviews in cybersecurity, DevSecOp

How to Use

- Install dependencies:
    - pip install -r requirements.txt
- Run locally:
    - python main.py or use run.bat
- Docker deployment:
    - docker-compose up -d
- Access dashboard:
    - http://localhost:5000

I pushed the project to a GitHub repository, the link can be found from the link below:
https://github.com/RafaelAngeloChristianto/host_intrusion_detection-system

Key Files Breakdown:

1. Core Orchestration

- main.py - Main application entry point
- config/settings.py - All configuration settings
- config/rules.py - Detection rules and thresholds

2. Monitoring Modules

- monitors/file_monitor.py - File integrity using watchdog + SHA256 hashing
- monitors/process_monitor.py - Process monitoring with psutil
- monitors/registry_monitor.py - Registry monitoring with winreg
- monitors/eventlog_monitor.py - Windows Event Log parsing
- monitors/network_monitor.py - Network connection monitoring

3. Alerting System

- alerts/alert_manager.py - Central alert coordinator
- alerts/email_notifier.py - SMTP email alerts
- alerts/desktop_notifier.py - Desktop notifications

4. Dashboard

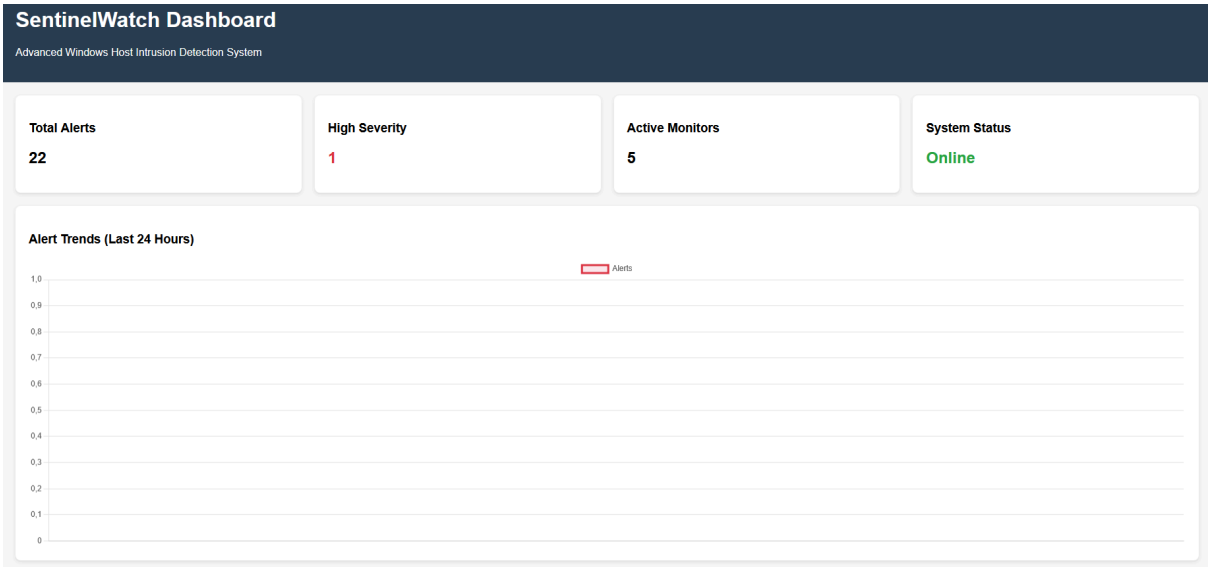
- dashboard/app.py - Flask app initialization
- dashboard/routes.py - API endpoints and views
- dashboard/utils.py - Data formatting for Chart.js

5. Supporting Files

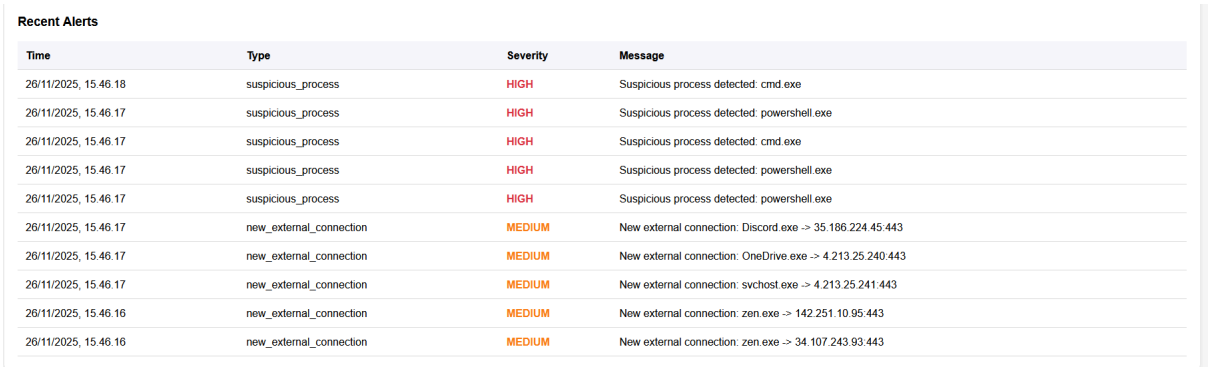
- database/models.py - Data models for alerts/logs
- utils/logger.py - Centralized logging
- utils/helpers.py - Common utility functions

Results

There is a dashboard made with Flask we can visit for more in detailed information about our activities



The number of alerts will continuously change, as long as the HIDS is still actively on.



Recent Alerts

Time	Type	Severity	Message
26/11/2025, 15.46.18	suspicious_process	HIGH	Suspicious process detected: cmd.exe
26/11/2025, 15.46.17	suspicious_process	HIGH	Suspicious process detected: powershell.exe
26/11/2025, 15.46.17	suspicious_process	HIGH	Suspicious process detected: cmd.exe
26/11/2025, 15.46.17	suspicious_process	HIGH	Suspicious process detected: powershell.exe
26/11/2025, 15.46.17	suspicious_process	HIGH	Suspicious process detected: powershell.exe
26/11/2025, 15.46.17	new_external_connection	MEDIUM	New external connection: Discord.exe -> 35.186.224.45:443
26/11/2025, 15.46.17	new_external_connection	MEDIUM	New external connection: OneDrive.exe -> 4.213.25.240:443
26/11/2025, 15.46.17	new_external_connection	MEDIUM	New external connection: svchost.exe -> 4.213.25.241:443
26/11/2025, 15.46.16	new_external_connection	MEDIUM	New external connection: zen.exe -> 142.251.10.95:443
26/11/2025, 15.46.16	new_external_connection	MEDIUM	New external connection: zen.exe -> 34.107.243.93:443

Reasonings for high and medium severity.

- cmd.exe and powershell.exe are marked suspicious because they are powerful built-in tools commonly used in attacks, so the HIDS treats any execution as high-risk.
- New external connections are flagged because any outbound traffic to public IP addresses is treated as potentially risky, even if from legitimate apps.