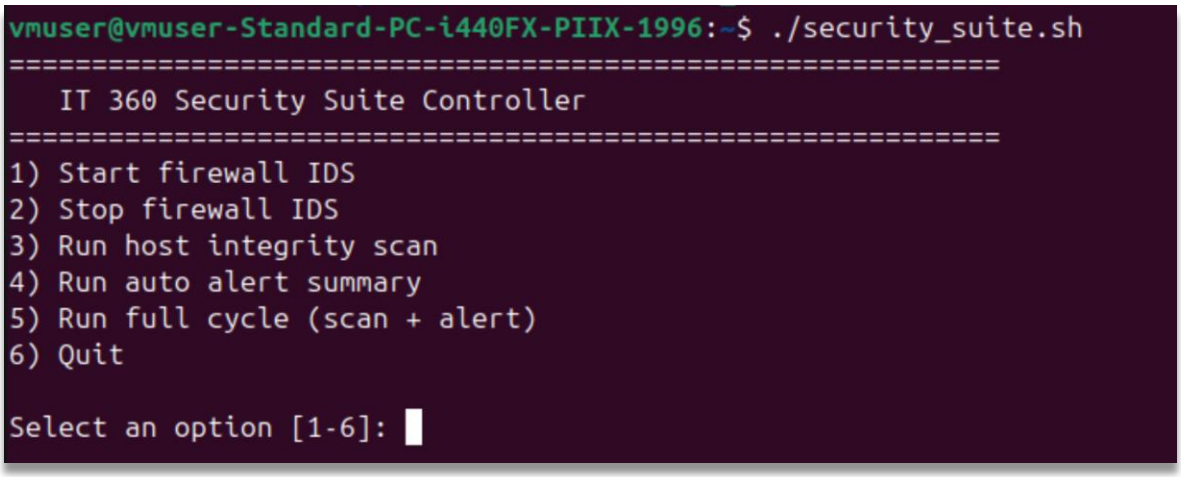
Stephano Valdivia

Dariel Nunez

Security Suite

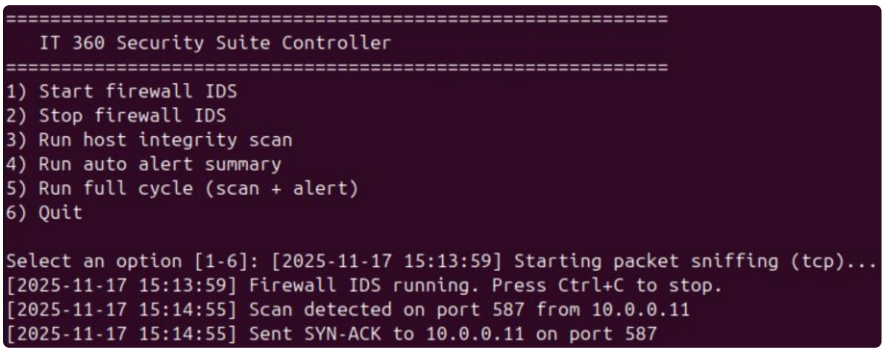
IT 360 001

The purpose of this project was to design and demonstrate a small security monitoring tool that supports basic digital forensics tasks. The tool focuses on identifying suspicious network activity, detecting important changes to the host, and producing summaries that investigators can review later. In digital forensics, it is often necessary to understand what happened on a system and when. This project addresses that need by providing real-time observation combined with evidence preservation. The tool brings together network intrusion detection, host integrity checking, and automated alerting into one simple, menu-driven system.

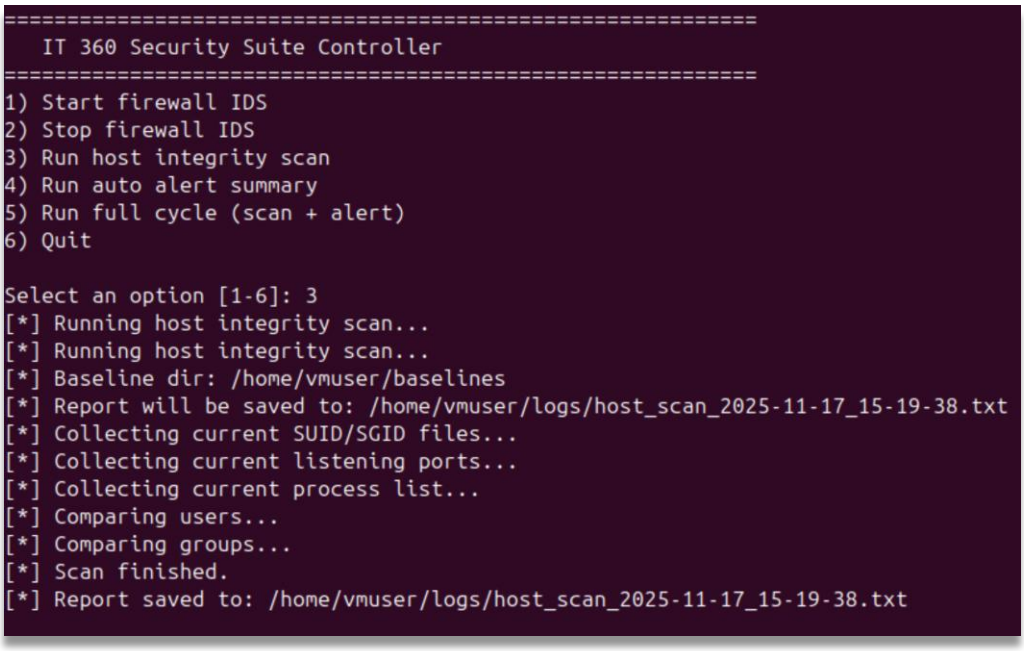


Technical Implementation

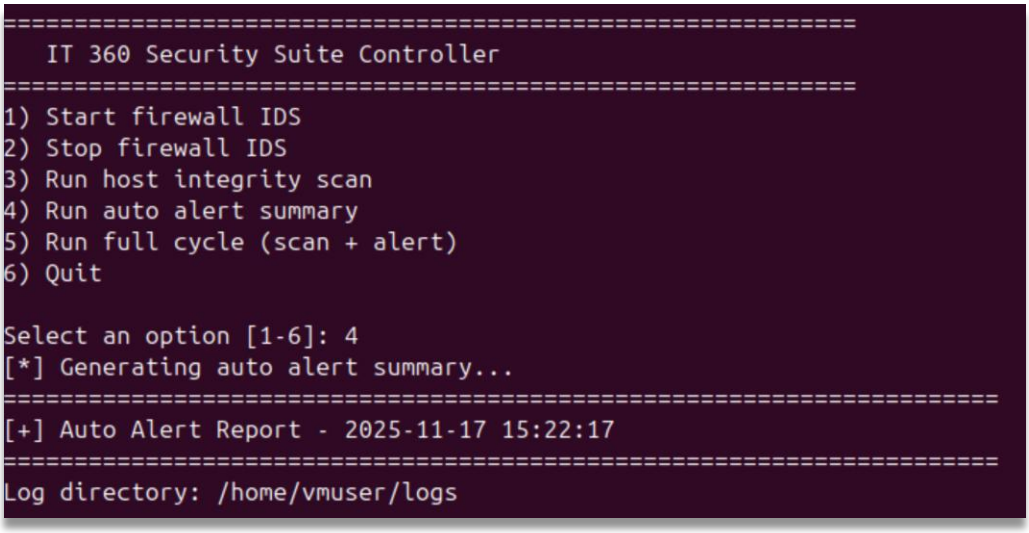
The system is made up of three components that work together but can also run independently. The first component is the network firewall IDS written in Python. It uses Scapy to monitor packets and detects TCP SYN scans, which are commonly used for reconnaissance. When a suspicious IP address is identified, the script automatically blocks it using iptables. Python was chosen here because Scapy provides a straightforward way to inspect packets without low-level coding.



The second component is the host integrity scanner written in Bash. It compares the current state of the machine to a previously recorded baseline. This includes checking for new SUID or SGID files, new listening ports, unexpected processes, and changes to system users or groups. Bash was used because these tasks rely heavily on built-in Linux utilities such as ss, find, and /etc/passwd.



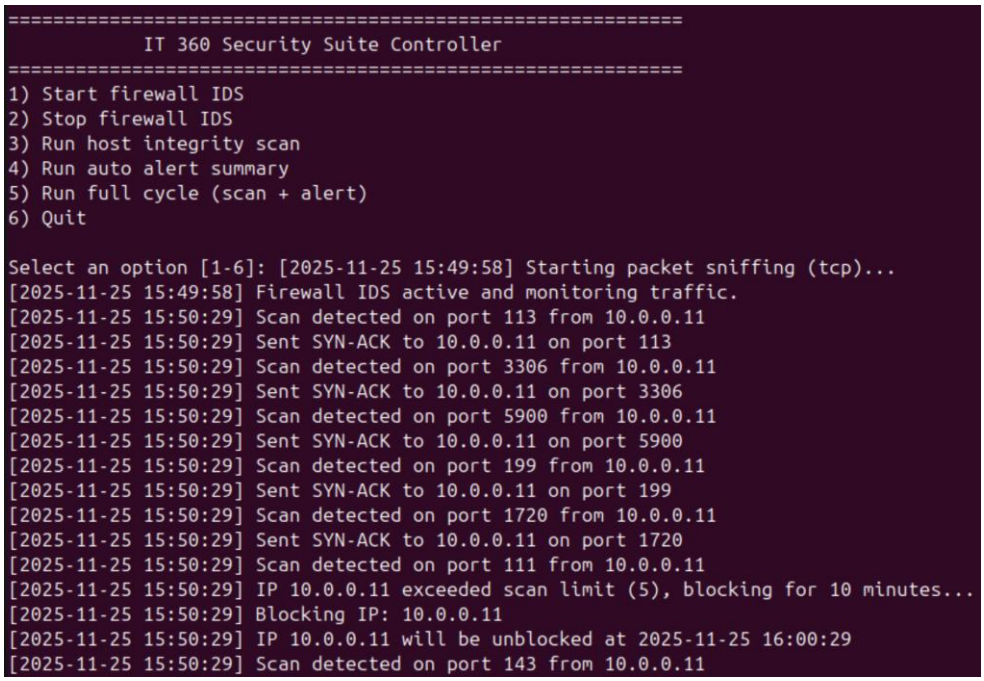
The third component is the auto-alert summary script in Python. It reads the log created by the firewall IDS and the most recent host scan report. It then combines those results into one summary so an investigator can quickly review recent activity. This avoids the need to read multiple files separately and provides a clear, time-stamped overview of events.



All components are controlled through a main Bash script that provides a numbered menu. This allows the user to start or stop the IDS, run a host scan, or generate an alert summary without dealing directly with each script.

Results

Testing showed that all three components functioned as expected. When the Ubuntu VM was scanned from another machine using Nmap, the IDS detected the SYN scan, identified the attacking IP, and added a firewall rule to block it. This demonstrated real-time detection and automated response.



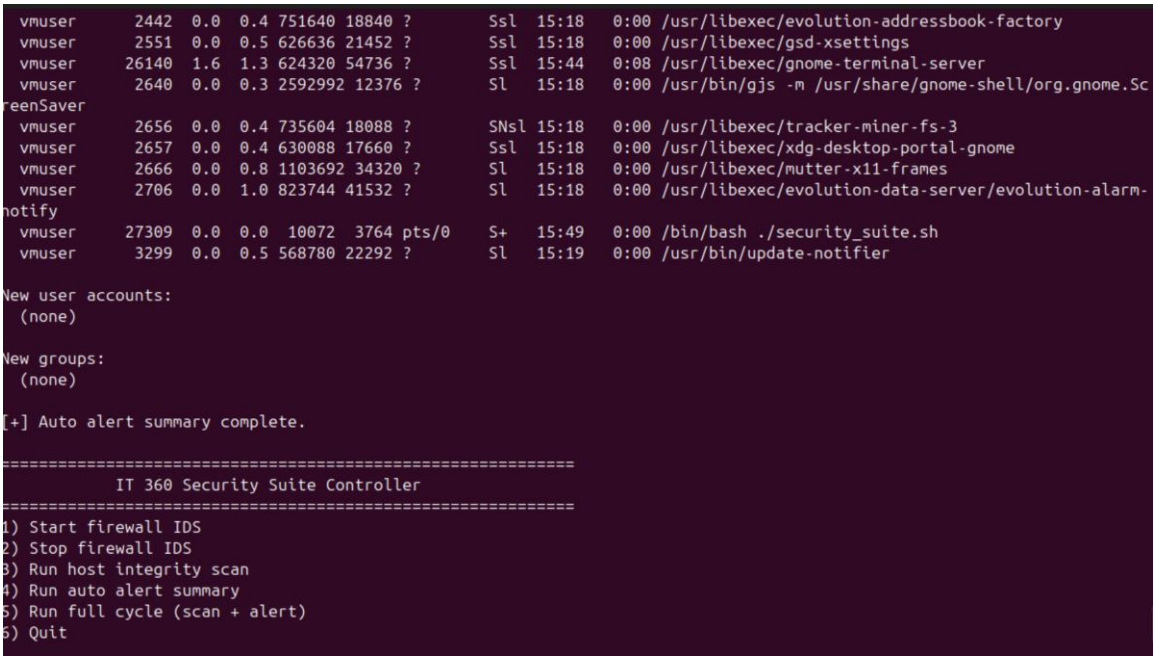
The host integrity scanner correctly flagged intentional changes made for testing, such as creating a new SUID file or opening a new port. Each scan produced a clear report showing what changed since the baseline.

```
============================================================
           IT 360 Security Suite Controller
============================================================
1) Start firewall IDS
2) Stop firewall IDS
3) Run host integrity scan
4) Run auto alert summary
5) Run full cycle (scan + alert)
6) Quit

Select an option [1-6]: 3

[*] Running host integrity scan...
[*] Running host integrity scan...
[*] Baseline dir: /home/vmuser/IT360_GroupProject_Fall2025/src/baselines
[*] Report will be saved to: /home/vmuser/IT360_GroupProject_Fall2025/src/logs/host_scan_2025-11-25_15-52-30.txt
[*] Collecting current SUID/SGID files...
[*] Collecting current listening ports...
[*] Collecting current process list...
[*] Comparing users...
[*] Comparing groups...
[*] Scan finished.
[*] Report saved to: /home/vmuser/IT360_GroupProject_Fall2025/src/logs/host_scan_2025-11-25_15-52-30.txt
```

```
=== Network Alerts (group_firewall.log) ===
Total scan events detected: 232
Blocked IPs:
  - 10.0.0.11 (blocked 1 time(s))
```

The auto-alert script successfully read both logs and created a combined summary. This gave an easy-to-read snapshot of network activity and host changes without requiring the user to search through multiple folders.

```
============================================================
           IT 360 Security Suite Controller
============================================================
1) Start firewall IDS
2) Stop firewall IDS
3) Run host integrity scan
4) Run auto alert summary
5) Run full cycle (scan + alert)
6) Quit

Select an option [1-6]: 4

[*] Running auto-alert summary...
========================================================================
[+] Auto Alert Report - 2025-11-25 15:53:01
========================================================================
Log directory: /home/vmuser/IT360_GroupProject_Fall2025/src/logs

=== Network Alerts (group_firewall.log) ===
Total scan events detected: 232
Blocked IPs:
  - 10.0.0.11 (blocked 1 time(s))

=== Host Integrity Alerts (latest host_scan report) ===
Using report: host_scan_2025-11-25_15-52-30.txt

New SUID/SGID files:
  (none)
```



Lessons Learned & Conclusion

One of the strongest aspects of the project was keeping the design modular. It made testing easier and allowed each script to be fixed or improved without affecting the others. The menu interface also made the tool feel more complete and easier to demonstrate.

There were challenges along the way. File-path issues appeared when running the tool on a fresh VM, especially when locating logs and the virtual environment. Installing Scapy correctly required the use of a Python virtual environment to avoid dependency conflicts. Another issue was ensuring each script had the correct permissions to execute. These problems helped reinforce the importance of environment setup and consistent directory structure in digital forensics tools.

If the project were extended in the future, improvements could include adding alert delivery through email, expanding detection for more attack types, adding log rotation, or designing a small GUI to make the tool more approachable.

Overall, the project met the assignment requirements and produced a working, demonstratable security tool that ties together network monitoring, host integrity checks, and forensic reporting.