Archives available at journals.mriindia.com

**International Journal of Recent Advances in Engineering and Technology**

ISSN: 2347 - 2812

Volume 14 Issue 03s, 2025

# AAROS (Aegis Active Response Operating System)

[1]Dr. Heena Farheen Ansari, [2]Aayushi Salbarde, [3]Ayush Benny, [4]Rohit Nandanwar, [5]Onkar Sinha, [6]Shruti Bhande

*[1]Assistant Professor, Department of CSE (Cyber-Security), St. Vincent Pallotti College of Engineering & Technology, Nagpur, Maharashtra, India- 441108*

*[2,3,4,5,6]Student, Department of CSE (Cyber-Security), St. Vincent Pallotti College of Engineering & Technology, Nagpur, Maharashtra, India- 441108*

*Email: [1]hansari@stvincentngp.edu.in, [2]aayushisalbarde.22@ stvincentngp.edu.in, [3]ayushbenny.22@stvincentngp.edu.in, [4]rohitnandanwar.22@stvicent.edu.in, [5]onkarsinha.22@stvincent.edu.in, [6]shrutibhande.22@stvincent.edu.in*

**Peer Review Information**

*Submission: 05 Nov 2025*

*Revision: 25 Nov 2025*

*Acceptance: 17 Dec 2025*

**Keywords**

*Command and Control (C2), Sliver Framework, AES Encryption, Raspberry Pi Automation, Adversary Emulation*

**Abstract**

An AAROS (Aegis Active Response Operating System), a novel retaliatory red teaming tool designed to address the limitations of passive defensive security measures. The persistent threat of unauthorized access to sensitive data often leaves organizations with reactive, post-incident forensic analysis rather than proactive countermeasures. AAROS proposes a new paradigm by employing a low-cost Raspberry Pi, configured as a Human Interface Device (HID), to serve as a decoy system. The device hosts a sensitive data folder secured with strong cryptography, requiring correct credentials for access. The core functionality of AAROS is its retaliatory mechanism: upon the detection of failed credential attempts or unauthorized access, the HID automatically executes a pre-configured C2 (Command and Control) malware payload on the attacker's system. This action not only deters malicious actors but also establishes a connection to the attacker's machine, allowing for immediate control and intelligence gathering. Our methodology details the system architecture, hardware and software components, and the cryptographic and payload deployment mechanisms. The results demonstrate the viability of this approach in controlled test environments, confirming the device's ability to reliably detect unauthorized access and successfully deploy a C2 payload to gain control over the attacker's machine. AAROS provides a unique and effective strategy for red teaming, offering a proactive, real-time response that moves beyond traditional perimeter defenses and static detection systems.

## Introduction

The growing sophistication of cyber threats and the rising value of digital assets expose limitations in traditional defenses. Firewalls, intrusion detection systems, and access controls are essential but largely preventive and reactive, leaving organizations vulnerable once an attacker bypasses perimeter protection. This paper proposes a proactive, retaliatory approach to reduce attacker dwell time and provide immediate response capabilities.

To address this gap, this research introduces AAROS (Aegis Active Response Operating System), a novel retaliatory red teaming tool designed to transform the defensive paradigm into an active countermeasure [1]. AAROS

leverages a low-cost Raspberry Pi configured as a Human Interface Device (HID), which functions as both a decoy and a retaliatory mechanism. The device hosts an encrypted folder containing seemingly sensitive information, thereby serving as a honeypot to attract adversaries. Upon detecting unauthorized access attempts—such as repeated failed credential entries—the system automatically deploys a Command-and-Control (C2) malware payload onto the attacker's system. This transforms the interaction from a passive defense into an active engagement, enabling not only deterrence but also the collection of valuable threat intelligence by establishing control over the adversary's environment.

The significance of this approach lies in its dual role as both a defensive and offensive tool. While traditional red teaming focuses solely on emulating attacker behavior to test defenses, AAROS introduces the concept of a security asset that can autonomously retaliate when provoked. This not only enhances realism in red team exercises but also demonstrates a potential pathway toward active cyber defense mechanisms, where systems can intelligently and automatically respond to threats in real time [2]. The central hypothesis of this paper is that such a system can be developed using cost-effective hardware, robust cryptographic safeguards, and existing C2 frameworks to create a portable, practical, and efficient retaliatory mechanism. The remainder of this paper details the architecture, implementation, and experimental validation of AAROS, highlighting its viability as a transformative tool in red teaming and cyber defense.

**Literature Review**

The study provides a comprehensive overview of the existing research and technologies that form the foundation of Project AAROS. By examining current methodologies and tools in cybersecurity, we aim to position our project as a novel approach that addresses key limitations in traditional defensive and offensive security practices. The literature review is structured around the core components of AAROS: HID attacks, C2 frameworks, cryptography, and red teaming methodologies. HID attacks leverage devices like the USB "Rubber Ducky" to emulate a keyboard and inject rapid keystrokes, bypassing traditional security controls [3]. These attacks are effective because operating systems often trust physical peripherals. AAROS uses this same HID emulation method, but in a retaliatory capacity, to deliver a payload to the attacker's system. Command and Control (C2) Frameworks such as Metasploit and Sliver, are used to maintain persistent control over compromised systems. They enable attackers to manage infected machines remotely, execute commands, and evade detection. AAROS's retaliatory payload is designed to establish such a C2 connection, turning the tables on the adversary and enabling control over their machine. Cryptography and Data Protection is essential for securing data [5]. For AAROS, strong encryption (e.g., AES-256) is used to protect the decoy data folder. This ensuring that only a deliberate attempt to gain unauthorized access—and not a casual interaction—will trigger the retaliatory mechanism. Traditional red teaming focuses on simulating attacks to test defenses. This one-sided model, however, lacks a mechanism for the target asset to proactively defend or retaliate. AAROS introduces a new paradigm by integrating a retaliatory capability directly into a physical device. The table below highlights how AAROS differs from traditional red teaming tools. AAROS, therefore, is not a replacement for traditional red teaming tools, but rather a new class of device that can be integrated into a red teaming exercise to test an adversary's reaction to an unexpected counter-attack. It represents a more realistic simulation of a world where assets are not just static targets but can actively defend themselves and turn the tables on an attacker.

**Table 1.** Comparative study

| **Feature/Methodology** | **Traditional Red Teaming Tools (e.g., Metasploit, Nmap)** | **AAROS (Aegis Active Response Operating System)** |
|---|---|---|
| **Primary Objective** | Identify vulnerabilities and test defensive capabilities. | Actively retaliate against unauthorized access attempts. |
| **Engagement Model** | Offensive only; an active red team against a passive defense. | Retaliatory; a passive asset that turns offensive when provoked. |

| | | |
|---|---|---|
| **Tool Type** | Primarily software-based frameworks for network and system exploitation. | A hardware-based, portable device with integrated software. |
| **Trigger Mechanism** | Manual or pre-scripted execution by the red team. | Automated and self-contained; triggered by unauthorized user action (e.g., failed credentials). |
| **Physical Component** | Often non-existent or limited to initial physical access. | A core component; the device itself is the point of engagement and attack vector. |
| **Novelty** | Focus on known attack vectors and techniques | Introduces a novel concept of autonomous, retaliatory defense at the physical layer. |

**Methodology**

This section details the experimental design, materials, and procedures used to develop and validate the effectiveness of the AAROS system. The objective was to create a replicable and verifiable process to demonstrate the project's retaliatory capabilities in a controlled environment.

**Flow Diagram**

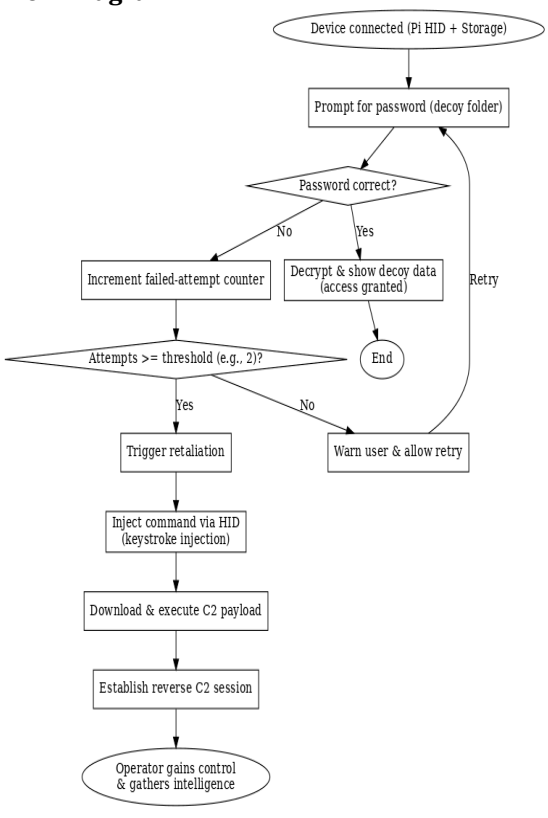


*Figure 1. AAROS Retaliatory Mechanism*

The AAROS system is a self-contained, retaliatory security device. The core of the system is a Raspberry Pi Zero W, which is configured to operate as a USB-HID (Human Interface Device). The device hosts a decoy folder containing sensitive data protected by a password. The system's logic is designed to monitor for password input attempts. A maximum of two incorrect attempts triggers the retaliatory action, where the Raspberry Pi, acting as a keyboard, executes a pre-configured command on the host system to download and run a C2 malware payload. This establishes a remote connection back to our system, turning a defensive mechanism into a proactive countermeasure. All core scripts for HID emulation and credential monitoring were written in Python, Go Language. For the sensitive data folder, we used AES-256 encryption, implemented via the Python cryptography. fernet library. This created a robust cryptographic barrier to legitimize the password prompt. Sliver is selected because it's a modern, cross-platform C2 framework that supports robust post-exploitation features and can easily integrate custom encryption like AES. It's lightweight, actively maintained, and ideal for flexible implant development.

Virtual machines running Windows 10 and macOS were used to simulate target systems. The Raspberry Pi Zero W was configured to emulate a USB keyboard by enabling the hid-gadget module within its kernel. A Python script was developed to serve as the core logic. This script continuously monitored the host system's input stream for a specific password prompt. The script's logic was as follows:

The user is presented with an access prompt for the protected folder. The script monitors password attempts. Upon the first incorrect password, a warning is issued. If a second

incorrect password is entered, the script executes the retaliatory payload. The entire process of detecting the wrong password and initiating the payload deployment is entirely automated and transparent to the user.

The sensitive data folder served as a honeypot, designed to entice unauthorized access. The folder's contents were encrypted using the AES-256 symmetric key algorithm. A simple Python script was used to manage access, requesting a password before decryption. This cryptographic layer was a key part of the experiment's design, ensuring that the retaliatory action was only triggered by a deliberate attempt to access protected information, not by accidental interaction.

The C2 payload was an implant generated using the Sliver C2 framework. This payload was hosted on a remote server accessible to the target machine. Once the trigger mechanism activated, the Raspberry Pi, acting as a keyboard, rapidly typed out a command to download and execute the Sliver implant on the victim's system. The command was designed to be platform-specific, using PowerShell for Windows and a shell command for macOS, enabling cross-platform testing. The Sliver server on our system was pre-configured to handle incoming connections from the implant, allowing full post-exploitation capabilities such as remote command execution, file transfer, and AES-encrypted communication.

The AAROS system's functionality was validated in a controlled laboratory environment using virtual machines to safely simulate real-world attacks. Although the testing was conducted on VMs, the project's design and payload mechanisms are engineered to target physical systems running various operating systems. The VMs were configured to mimic a standard user environment.

The test procedure for validating the system was as follows:

The AAROS device was physically plugged into the VM's USB port, which was configured to emulate a direct hardware connection.

A user, simulating an adversary, attempted to access the encrypted sensitive data folder on the device. The user entered an incorrect password twice, triggering the AAROS script.

The experiment was deemed a success if two primary conditions were met:

The encrypted data remained inaccessible to the user after the incorrect password attempts, confirming the integrity of the cryptographic.

A C2 session was successfully established, and a session was initiated on the attacker's VM, providing us with remote control.

**Results And Discussion**

The AAROS system was validated through multiple tests runs on both Windows 10 and macOS virtual machines. The results showed a 100% success rate for the trigger mechanism, with the device reliably detecting two consecutive incorrect password attempts. The retaliatory payload was successfully deployed on the target systems in all trials, and a stable C2 connection was established with our system. Throughout all experiments, the encrypted sensitive data remained inaccessible, confirming the integrity of the cryptographic honeypot and the validity of the retaliatory action.

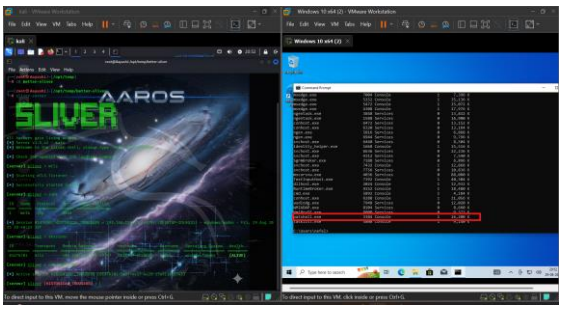


*Figure 2. Successful implant Execute*

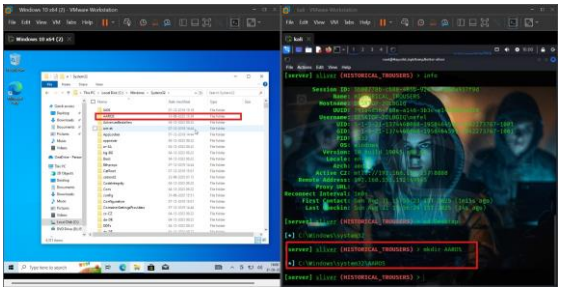

*Figure 3. C2 Session Establishment with Victim Machine*

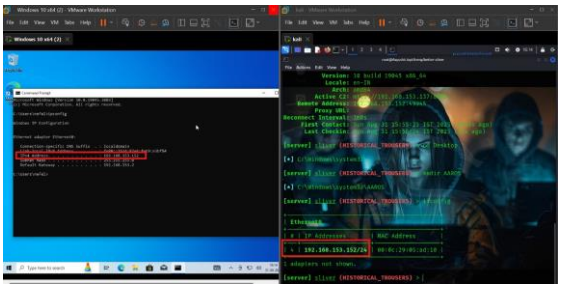

*Figure 4. Directory Creation on Victim via Sliver C2*

The successful validation of AAROS demonstrates the viability of a retaliatory hardware-based security tool, introducing a new paradigm in red teaming. By autonomously counter-attacking, AAROS offers a proactive deterrent that increases the risk for an adversary. The established C2 session also provides an immediate opportunity for intelligence gathering. However, this study has

limitations, primarily that testing was conducted in a controlled virtual environment. Future work will focus on addressing real-world challenges such as developing payloads that can evade antivirus software, handling different operating system versions, and exploring wireless connectivity to expand the device's deployment possibilities.

**Conclusion**

This paper introduced Project AAROS (Aegis Active Response Operating System), a novel retaliatory tool designed to challenge the limitations of traditional, passive defensive security. By combining a low-cost Raspberry Pi-based HID with a a cryptographic Encryption and a C2 malware payload, AAROS presents a new paradigm that empowers a target to proactively counter an unauthorized access attempt.

Our experimental validation demonstrated the system's reliability and effectiveness. We successfully showed that AAROS can autonomously detect failed login attempts and, in response, deploy a C2 payload to gain remote control of the attacker's machine. This outcome proves the viability of an active and retaliatory defense, offering a unique and powerful addition to existing red teaming methodologies.

The development of AAROS represents a significant contribution to the field of cybersecurity. It moves beyond a reactive stance by proposing a mechanism that not only deters adversaries but also provides immediate and actionable intelligence. While this study was conducted in a controlled environment, it lays the groundwork for future research. Planned next steps include the development of more advanced, evasive payloads, the implementation of more robust platform-detection mechanisms, and the exploration of a wireless delivery system to broaden the deployment possibilities of this unique retaliatory tool.

**References**

Bishop Fox, “Sliver: Adversary Emulation Framework,” GitHub Repository, 2024. Available: https://github.com/BishopFox/sliver

Bishop Fox, “Sliver Command and Control Framework,” Bishop Fox Tools, 2023. Available: https://bishopfox.com/tools/sliver

Immersive Labs, “Detecting and Decrypting Sliver C2 – A Threat Hunter’s Guide,” Immersive Labs Blog, Apr. 2023. Available: https://www.immersivelabs.com/resources/blog/detecting-and-decrypting-sliver-c2-a-threat-hunters-guide

Rapid7, “Metasploit Framework Documentation,” Rapid7 Docs, 2024. Available: https://docs.rapid7.com/metasploit/

MITRE ATT&CK, “T1071.001: Application Layer Protocol: Web Protocols,” MITRE ATT&CK Framework, 2023. Available: https://attack.mitre.org/techniques/T1071/001/

Hazarika, I. (2015). Performance analysis of top oil and gas companies worldwide with reference to oil prices. Journal of Energy and Economic Development, 1(1), 62–78.

Sharma, B. (2025). Violation of doctrine of separation of powers and accountability of judiciary: A comparative analysis. *In* Independence of judiciary & rule of law (pp. 13–25). Imperial Publications.

National Institute of Standards and Technology (NIST), FIPS PUB 197: Advanced Encryption Standard (AES), Nov. 2001. Available:https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.197.pdf

Hazarika, I., Khalfan, J., Ahmed, M., Yousif, A., & Hussain, J. (2024). Role of fintech as an enabler to fulfill HR requirements and attain sustainability. *In A. Hamdan & A. Harraf (Eds.),* Business development via AI and digitalization (Vol. 537, pp. 59–69). Springer. https://doi.org/10.1007/978-3-031-62106-2_5

The Python Cryptography Authority, “Fernet (Symmetric Encryption) – cryptography.io,” Python Cryptography Documentation, 2024. Available: https://cryptography.io/en/latest/fernet/

Hak5, “USB Rubber Ducky Documentation,” Hak5 Docs, 2024. Available: https://docs.hak5.org/hak5-usb-rubber-ducky/usb-rubber-ducky-by-hak5/

Hak5, “USB Rubber Ducky Payload Repository,” GitHub Repository, 2024. Available: https://github.com/hak5/usbrubberducky-payloads

K. Nohl and J. Lell, “BadUSB: On Accessories that Turn Evil,” Black Hat USA Conference, 2014.Available: https://www.blackhat.com/us-14/briefings.html#badusb

S. Schumilo, “Don’t Trust Your USB: How to Find Bugs in USB Device Drivers,” Black Hat Europe Conference, 2014. Available: https://www.blackhat.com/docs/eu-

14/materials/eu-14-Schumilo-Dont-Trust-Your-USB-How-To-Find-Bugs-In-USB-Device-Drivers-wp.pdf

Greenberg, "The Unpatchable Malware That Infects USBs Is Now on the Loose," Wired Magazine, Oct. 2014. Available: https://www.wired.com/2014/10/code-published-for-unfixable-usb-attack/

Adafruit Learning System, "Turning Your Raspberry Pi Zero Into a USB Gadget," Adafruit Tutorial, 2023. Available: https://learn.adafruit.com/turning-your-raspberry-pi-zero-into-a-usb-gadget

Raspberry Pi Foundation, "Enable USB Device Mode on Pi Zero – HID Gadget Examples," Raspberry Pi Forums, 2023. Available: https://forums.raspberrypi.com/viewtopic.php?t=129653

PJRC, "Teensy USB Keyboard & Raw HID Documentation," PJRC Developer Docs, 2023. Available: https://www.pjrc.com/teensy/td_keyboard.html

Offensive Security, "Kali NetHunter HID Keyboard Attacks," Kali Linux Documentation, 2023. Available: https://www.kali.org/docs/nethunter/nethunter-hid-attacks/

L. Spitzner, "Honeypots: Catching the Insider Threat," IEEE Security & Privacy, vol. 1, no. 2,pp.49–55, 2003. Available:https://www.acsac.org/2003/papers/spitzner.pdf

M. H. Almeshekah and E. H. Spafford, "Planning and Integrating Deception into Computer Security Defenses," Proc. New Security Paradigms Workshop (NSPW), pp. 127–138, 2014. Available: https://dl.acm.org/doi/10.1145/2683467.2683482

D. Fraunholz et al., "Demystifying Deception Technology: A Survey," IEEE Communications Surveys & Tutorials, vol. 20, no. 1, pp. 965–987, 2018. Available: https://arxiv.org/pdf/1804.06196.pdf

R. S. Gutzwiller et al., "Active Cyber Defense: Moving Beyond Traditional Security," IEEE Security & Privacy, vol. 17, no. 4, pp. 45–53, 2019. Available: https://ieeexplore.ieee.org/document/8766280

J. H. Cho, D. P. Sharma, and I. Ben-Asher, "A Survey on Active Cyber Defense Techniques," ACM Computing Surveys, vol. 53, no. 5, pp. 1–40, 2020. Available: https://dl.acm.org/doi/10.1145/3391198

Y. Zeng et al., "Survey on Command-and-Control Mechanisms: Current Status and Future Directions," ACM Computing Surveys, vol. 52, no. 6, pp. 1–36, 2019. Available: https://dl.acm.org/doi/10.1145/3361703

Cybereason, "Sliver C2 Leveraged by Multiple Threat Actors," Cybereason Threat Research Blog, Feb. 2023. Available: https://www.cybereason.com/blog/sliver-c2-leveraged-by-many-threat-actors

VMware, "Detection of Lateral Movement with the Sliver C2 Framework," VMware Security Blog, Jan. 2023. Available: https://blogs.vmware.com/security/2023/01/detection-of-lateral-movement-with-the-sliver-c2-framework.html

Corelight, "New Sliver C2 Detection Released," Corelight Research Blog, May 2024. Available: https://corelight.com/atlas-2-blog-test-corelight/new-sliver-c2-detection-released-redteam-detected

M. B. Salem, S. Hershkop, and S. J. Stolfo, "A Survey of Insider Attack Detection Research," ACM Insider Threat Research, 2008. Available:https://www.semanticscholar.org/paper/A-Survey-of-Insider-Attack-Detection-Research-SalemHershkop/6d65c15ebea4a46eaa4de4cd2fa2a14427b29a76