1. SYSTEM IDENTIFICATION

1.1. System Name/Title: HONEYPI NIDS

1.1.1. System Unique Identifier: HPI

1.2. Responsible Organization:

Name:	Zachary Weaver
Phone:	704-621-8447

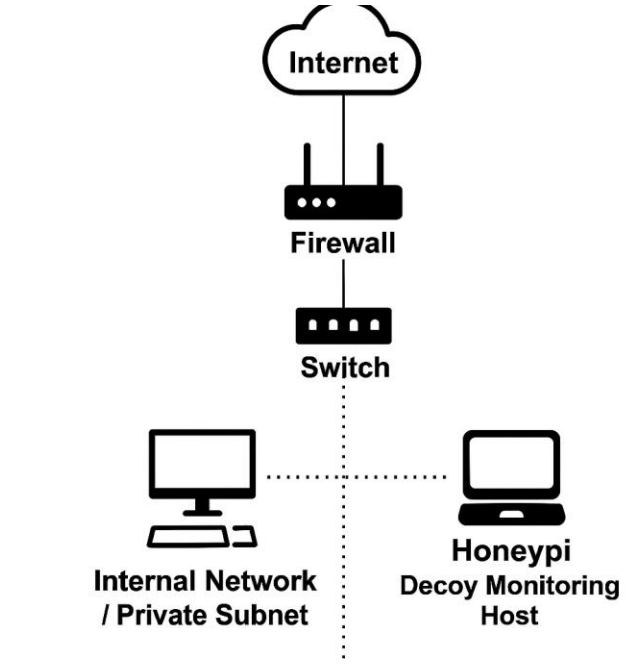
1.3. General Description/Purpose of System: HoneyPi is a portable Raspberry Pi-based security device designed to function as a Network Based Intrusion Detection System (NIDS) within a controlled environment. Its main purpose is to emulate decoy applications and services to attract, log, and analyze suspicious access attempts.

1.3.1. Number of end users and privileged users:

Roles of Users and Number of Each Type:

Number of Users	Number of Administrators/ Privileged Users
TBD	1

2. SYSTEM ENVIRONMENT



2.1. Hardware and Software Inventory

A full copy of all the hardware and software is maintained in the project repository and can be found at: https://github.com/zach-weav/HoneyPi/blob/main/docs/build/HoneyPi_HWSW_Inventory.csv

3. SECURITY REQUIREMENTS

(Note: The following security requirements have been created based on system design requirements and unit testing results)

To ensure the secure deployment and operation of the HoneyPi NIDS, the following security requirements have been defined. These requirements are based on industry best practice including the NIST framework. They will be integrated into the overall design, configuration, and ongoing maintenance of the device.

3.1. Access Control (AC)

3.1.1. Access to the HoneyPi host device must be restricted to authenticated users via secure credentials or keys.

3.1.2. Docker containers must run with the lease privilege required, and unnecessary ports must be closed.

3.1.3. Grafana must require authentication to access the dashboard.

3.2. Awareness and Training (AT)

3.2.1. All operators and testers of the device must receive guidance on ethical use and network policy compliance

3.2.2. A README file and ethical notice are to be included with the deployment package to reinforce intended educational and defensive usage

3.3. Configuration Management

3.3.1. All Docker images used should be built from minimal, verified base images.

3.3.2. Configuration files (Grafana, Promtail, Loki, Prometheus) must be controlled and stored in the project repository.

3.3.3. All Docker containers must mount the required configuration and log files explicitly using volume bindings to limit direct host filesystem access.

3.3.4. The system must auto-start only predefined containers via a startup script ran through the crontab.

3.4. Identification and Authentication (IA)

3.4.1. Grafana access must require login credentials.

3.4.2. The Raspberry Pi Host must require strong passwords and disable root SSH access.

3.5. Incident Response (IR)

3.5.1. Email alerts should be configured in Grafana for events containing high-risk keywords showing attacker connection attempts.

3.5.2. The system administrator must receive alerts in real-time upon suspicious activity.

3.5.3. All alerts must be tagged and stored in Loki for further analysis.

3.6. Risk Assessment and Security Review (RA)

3.6.1. The HoneyPi must undergo periodic security testing, including peer-led tests and controlled attacks.

3.6.2. All components of the containerized applications should be reviewed after any update to verify a secure configuration.

3.6.3. All feedback from the testers should be analyzed and applied as part of the security improvement cycle.

4. RECORD OF SECURITY REQUIREMENT CHANGES

Date	Description	Made By: