#### **Title: Development of "Cyberguard.py" - A Portable Cybersecurity Pre-Investigation Tool**

By

SPINES ONYANGO ODHIAMBO

**Abstract**

In today's digital landscape, safeguarding computing devices against cyber threats is crucial. The "Cyberguard.py" project addresses this need by developing a Python-based, cross-platform security tool designed to operate on both Windows and Ubuntu systems. This tool, intended to be executed automatically from a flash drive, performs essential security checks to identify, resist, and shield against malware attacks. The primary objectives include creating a comprehensive health check script that evaluates malware presence, firewall status, and critical security features. Through meticulous research, planning, and testing, the project aims to provide an educational resource that demonstrates fundamental cybersecurity practices and Python programming techniques. The expected outcomes are a functional cybersecurity tool, enhanced user awareness of regular security checks, and a foundational framework for future cybersecurity projects. The "Cyberguard.py" project offers a practical solution to improving device security while serving as a valuable learning platform for aspiring cybersecurity professionals.

#### **Introduction**

In the modern digital age, ensuring the security of computing devices is paramount. With the increasing prevalence of malware and cyber-attacks, there is a growing need for tools that can quickly assess the health and security of a system. The "Cyberguard.py" project was conceived as an educational initiative aimed at developing a portable, executable program that can be run from a removable flash drive to identify, resist, and shield against malware attacks when plugged into any device. This proposal outlines the objectives, methodology, and expected outcomes of the "Cyberguard.py" project.

#### **Objectives**

The primary objectives of the "Cyberguard.py" project are:

1. **Develop a Cross-Platform Tool**: Create a Python-based script that can operate on both Windows and Ubuntu systems to perform preliminary security checks.
2. **Automated Execution**: Ensure the program executes automatically when the flash drive is inserted, initiating a security check on the host device.
3. **Comprehensive Health Check**: Assess the security status of the device, including malware detection, firewall status, and essential security features.
4. **Educational Purpose**: Provide an educational tool that demonstrates fundamental cybersecurity practices and Python programming techniques.

#### **Methodology**

The development process for "Cyberguard.py" involves the following steps:

1. **Research and Planning**:
   * Identify key security checks required for assessing device health.
   * Research Python libraries and system commands necessary for implementation on both Windows and Ubuntu.
2. **Script Development**:
   * **Malware Scanning**: Implement a function to scan for malware using built-in tools like Windows Defender on Windows and ClamAV on Ubuntu.
   * **Firewall Status Check**: Develop a function to check and report the status of the system firewall.
   * **System Security Check**: Create a function to verify the presence of critical security processes and features, such as Windows Defender on Windows and AppArmor/SELinux on Ubuntu.
3. **Cross-Platform Compatibility**:
   * Ensure the script functions correctly on both Windows and Ubuntu by handling system-specific commands and outputs.
4. **Testing and Validation**:
   * Conduct thorough testing on various Windows and Ubuntu systems to validate the script’s functionality and reliability.
   * Make necessary adjustments based on test results to ensure robust performance.
5. **Documentation**:
   * Provide comprehensive documentation, including user instructions, code explanations, and troubleshooting tips.

#### **Expected Outcomes**

The successful completion of the "Cyberguard.py" project will result in:

1. **A Functional Cybersecurity Tool**: A Python script that can be run from a flash drive to perform essential security checks on any Windows or Ubuntu device.
2. **Enhanced Security Awareness**: An educational tool that helps users understand the importance of regular security checks and how to perform them using Python.
3. **Foundation for Future Projects**: A robust framework that can be expanded upon for more advanced cybersecurity tools and educational purposes.

#### **Conclusion**

The "Cyberguard.py" project represents a practical and educational approach to enhancing device security. By leveraging Python programming and system commands, the project aims to create a versatile tool that can help users quickly assess the security status of their devices. The project not only contributes to improved cybersecurity practices but also serves as an excellent learning opportunity for those interested in the intersection of programming and cybersecurity.