PROJECT

SYNOPSIS

OF

Automated Keystroke Injection Device

ALIGARH MUSLIM UNIVERSITY

Submitted by:-

Syed Mohammad Hamza Husain
23CSMSA113
GM5680
MSc. Cybersecurity & Digital Forensics, III

Submitted to: -

Dr. Faisal Anwar
Dr. Mohd Sajid

Abstract

This project aims to design and implement a DIY version of a Rubber Ducky penetration testing tool with MicroSD card support and USB interaction capabilities. Using an Arduino Pro Micro microcontroller, the tool will act as a Human Interface Device (HID), allowing it to inject pre-configured keystroke payloads into a target system. The inclusion of MicroSD storage provides flexibility in storing and executing a variety of payloads, while the Micro USB interface allows for direct computer interaction. This project demonstrates the vulnerabilities in systems that accept unauthorized input via USB devices, raising awareness of potential security flaws. The tool will be tested to evaluate its effectiveness in executing keystroke injection attacks and offer insights into security mitigations.

1. Introduction

1.1 Background:

The USB Rubber Ducky is a popular penetration testing tool that emulates a USB keyboard, capable of injecting keystroke payloads into computers to automate security attacks. This project aims to design a DIY version of a Rubber Ducky, which includes MicroSD support for storing payloads and a Micro USB connector for direct interaction with computers. By simulating keyboard input, this tool can demonstrate vulnerabilities in systems that allow unauthorized access via keystroke injection.

1.2 Objective:

The objective of this project is to build a custom rubber ducky pentest tool using an Arduino Pro Micro, a MicroSD card module, and a Micro USB connector for computer interaction. The tool will be capable of reading payloads from the MicroSD card and executing them as keystroke injections when connected to a computer.

2. Problem Statement

2.1 Problem Description:

Many systems are vulnerable to USB HID (Human Interface Device) attacks, where malicious actors exploit the trust that computers place in keyboard inputs. A USB device masquerading as a keyboard can inject a series of harmful commands into a system. This project seeks to implement a rubber ducky-style device to illustrate this security flaw and explore mitigation techniques.

2.2 Importance of the Problem:

The problem is important because USB-based attacks are often overlooked, despite their simplicity and effectiveness. By designing and demonstrating a DIY rubber ducky, we aim to raise awareness of this vulnerability and offer solutions to improve system security.

3. Study of Existing Systems

3.1 Overview:

The commercial USB Rubber Ducky by Hak5 is a well-known device in the cybersecurity community, allowing users to deploy pre-programmed scripts to perform automated keystroke attacks. However, commercial devices are costly, and DIY alternatives using microcontrollers like the Arduino Pro Micro offer a more accessible solution for educational and research purposes.

3.2 Limitations:

DIY rubber ducky tools can be limited by processing power, storage capacity, and the complexity of payloads compared to commercial solutions. The absence of wireless control also restricts certain remote attack scenarios.

3.3 Comparative Analysis:

Compared to commercial USB Rubber Ducky devices, this project will focus on building a cost-effective version using an Arduino Pro Micro. While the DIY version lacks certain advanced features (e.g., wireless control), it retains core functionality like script execution via USB and offers flexibility through MicroSD-based payload storage.

4. Proposed Solution

4.1 Overview:

The project will use an Arduino Pro Micro as the core microcontroller to emulate a keyboard. A MicroSD card module will be included for payload storage, allowing flexibility in the types of payloads that can be executed. The Micro USB connector on the Arduino Pro Micro will serve as the interface for interacting with a computer, allowing the device to inject keystrokes upon connection.

4.2 Key Features:

• USB HID Emulation: The tool will act as a keyboard, injecting keystrokes via pre-stored payloads.

• MicroSD Card Support: Payloads will be stored on a MicroSD card and read by the Arduino during execution.

• Compact Design: Using a Micro USB connector and a small footprint, the tool will be portable and easy to deploy.

• Cost Efficiency: The tool will be built with inexpensive components, making it accessible for educational and research purposes.

5. Scope of the Project

5.1 Inclusions:

• Building the hardware for the rubber ducky tool using an Arduino Pro Micro and MicroSD card module.
• Developing software for reading payloads from the MicroSD card and simulating keyboard inputs.
• Testing the tool on different systems to evaluate its effectiveness in automated keystroke injection.

5.2 Exclusions:

• Wireless capabilities and advanced attack scenarios (e.g., remote command execution) are outside the scope of this project.

6. Sample Design

Replace the image of 'Data Preprocessing Image' with a block diagram illustrating the hardware connections:
• Arduino Pro Micro connected to the MicroSD module.
• Micro USB connector for interaction with the computer.

7. Feasibility Analysis

7.1 Technical Feasibility:

The project is technically feasible with the available hardware (Arduino Pro Micro, MicroSD card module) and software tools (Arduino IDE, Keyboard.h library).

7.2 Economic Feasibility:

The project is cost-effective, as it uses low-cost components and open-source software.

7.3 Operational Feasibility:

The rubber ducky tool can be integrated into various pentesting scenarios and offers a straightforward workflow for execution.

8. Tools and Technologies

8.1 Programming Languages:

• C++ for programming the Arduino Pro Micro in the Arduino IDE.

8.2 Development Tools:

• Arduino IDE for writing and uploading the payload execution code.

8.3 Hardware Components:

• Arduino Pro Micro, MicroSD card module, and Micro USB connector.

9. Expected Outcomes

9.1 Deliverables:

• A functional rubber ducky pentest tool with MicroSD-based payload execution.
• Documentation detailing the hardware setup, payload development, and testing results.

9.2 Success Criteria:

• The tool should successfully execute payloads from the MicroSD card and demonstrate the feasibility of keystroke injection attacks on target systems.