

IoT Penetration testing

Gagandeep Sur



March 4, 2022

Sam houston state university

Huntsville

Contents

[1 Project Overview: 1](#_Toc97291046)

[2 Introduction 1](#_Toc97291047)

[2.1 IoT architecture 2](#_Toc97291048)

[2.2 Exploits 2](#_Toc97291049)

[3 Experiment: 2](#_Toc97291050)

[3.1 Hardware setup 2](#_Toc97291051)

[3.2 Application 2](#_Toc97291052)

[3.3 Attack workflow 2](#_Toc97291053)

[3.4 Setting up the Metasploit Framework Database 3](#_Toc97291054)

[4 Work In progress 9](#_Toc97291055)

[5 Uncompleted work 9](#_Toc97291056)

[6 Summary 9](#_Toc97291057)

[7 Add the link to your project here: 10](#_Toc97291058)

Capstone Project (Midterm Progress report)

IoT based Penetration Testing

# Project Overview:

Penetration testing is field of ethical hacking concentrating on exploiting the weakness of a system. Iot penetration testing is similar but at a smaller scale where the exploitation requires customized attack and efficient use of resources, algorithms, and payloads to target vulnerable assets. The project discusses customized way to use penetration testing on IoT devices.

# Introduction

A penetration test (pen test) is an authorized simulated attack performed on a computer system to evaluate its security. Penetration testers find the weaknesses in a system using the same tools, techniques, and processes as attackers to find and demonstrate the business impacts. Penetration tests simulate attacks that could threaten an IT infrastructure and its various constituents, which may be exposed as a result and lead to leakage of confidential or proprietary data. They can examine whether a system is robust enough to withstand attacks from authenticated and unauthenticated positions, as well as a range of system roles. With the right scope, a pen test can dive into any aspect of a system[1]

## IoT architecture

IoT based penetration testing includes engaging to test application and physical layers.

### Sensing layer

This consists of the hardware sensors and sensor networks.

### Communication Layer

This consists of the communication mechanism that allows thesensing layer to communicatewith the Management layer for example – Wifi, 3G, LTE, Ethernet etc.

### Management Layer

This is the topmost layer and is responsible for making sense out of the raw data and provide a presentable and fancy view to the users. It includes the cloud, storage, apps etc.

## Exploits

### Application layer attack

Socket and port can be exposed due to poor utilization of the practices

e.g. user authentication, weak password, lack of firewall, open ports on application layer.

### Hardware layer attack

Easy access to hardware e.g. sensors (camera, temperature) without any authentication or user privileges control.

# Experiment:

## Hardware setup

IoT using raspberry pi 3B+ model (1 GB RAM, 32GB storage). Camera Ardufruit, IR temperature sensor

## Application

Build a python application to run a server to accept commands to control the camera and IR sensor. The camera and IR sensor works in tandem to capture images and temperature and run analysis if a subject is outside the acceptable temperature range.

## Attack workflow

Kali Linux as host machine on dedicated desktop; target Raspberry PI’s open ports running custom build web server.

Use Nmap to get information on open ports, using other tools (nessus) to gather information on possible exploits and attack it using those payloads.

#### Issues

Initially, set up a lab with VMware workstation and kali Linux VM, and used Metasploit framework for penetration testing which is inbuilt feature in Kali Linux. Tried to use nmap to gather information about all the open ports on raspberry pi (IoT), but somehow it was not working using Kali Linux as a virtual machine. So had to forgo using VM and switch to real machine with Kali Linux installed on it.

## Setting up the Metasploit Framework Database

### Connect to Database (PostgreSQL)

The Metasploit Framework is an amazing tool, made even better by the fact that we can configure it to connect to a database and save the hosts, services, and other things we've discovered. You don't have to do connect MSF to a database, but if you're going to use it for more than just a one-off task you may as well. Having everything stored in a database also allows us to export the database and move it to another Kali installation or use it to help write those all-important reports. After the database starts, you can use any of the following commands to manage the database:

* msfdb reinit - Deletes and reinitializes the database.
* msfdb delete - Deletes the database.
* msfdb start - Starts the database.
* msfdb stop - Stops the database.
* msfdb status - Shows the database status.

Launch the Metasploit Framework console, then use the db\_status command to verify that Metasploit Framework Console is connected to the PostgreSQL database:

A computer screen capture

Description automatically generated with medium confidence

### Penetration

MSF console is ready to dive in penetration testing on device. These are the steps being taken for pen testing:

### Nmap reconnaissance

Nmap stands for Network Mapper. This is the first step in penetration testing. It's a Linux command-line utility that scans a network for IP addresses and ports, as well as detects installed software. Nmap allows network administrators to discover which devices are connected to their network, find open ports and services, and scan for vulnerabilities. Nmap tool is being used in this project to see all open ports on IoT operating system. I am using d\_nmap here, where d denotes as a database where all the searches from nmap will be saved for future use.

Graphical user interface, text

Description automatically generated

Nmap is very good at detecting services and versions of the services running on the target system. But this service detection features does not come by default there are some *Nmap* flags or options that you can pass fo*r* *Nmap* to be able to detect service running on the target machine.

Text

Description automatically generated

### 

Nessus tool for vulnerability Nessus is an open-source network vulnerability scanner that uses the Common Vulnerabilities and Exposures architecture for easy cross-linking between compliant security tools. In fact, Nessus is one of the many vulnerability scanners used during vulnerability assessments and penetration testing engagements, including malicious attacks*and*checks computers to find vulnerabilities that hackers COULD exploit.Downloaded the Nessus tool from <https://www.tenable.com/products/nessus>. After downloading it on the system, I get it start from msfconsole using following commands:

Text

Description automatically generated

Nessus tool is used to scan vulnerabilities in Iot device. Created some policy for scan. After creating policy, I launched the scan name IoTpentest1 scan.

A screenshot of a computer

Description automatically generated with medium confidence

*This scan shows some vulnerabilities in IoT device and shows vulnerability severity level* ***CVSS 3.0.*** The Common Vulnerability Scoring System (CVSS) a public framework for rating the severity of security vulnerabilities in software.*Here in this scan its shows severity level CVSS 3.0 which means vulnerabilities are very low in ranking.*

Graphical user interface

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

Text

Description automatically generated

**Text

Description automatically generated**

I will be iteratively running these steps as project progresses.

I import the Nessus vulnerabilities report into msfconsole using db\_import command

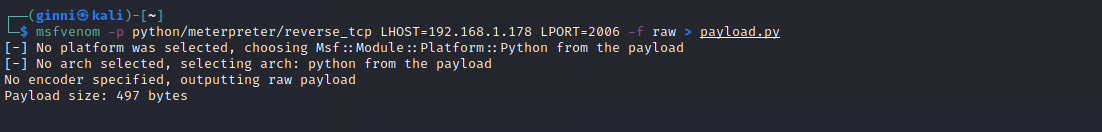
Text

Description automatically generated

### Exploitation:

The exploitation phase of a penetration test focuses solely on establishing access to a system or resource by bypassing security restrictions. The main focus is to identify the main entry point into the device and to identify high value target assets.

Now I have vulnerability report from Nessus that can be used to exploit IoT device. First I need to write a payload which can get entry in to the device remotely and can be further exploited against its vulnerabilities. I wrote a payload using msfvenom to get remote entry as shown in following figure



I tried to exploit target system using msfconsole exploits i.e., use exploit/multi/handler

Text

Description automatically generated

### Post Exploitation:

# Work In progress

Application layer programming to Passover commands to the camera and thermal image using IR temperature-profile sensor.

Post exploitation to evaluate information obtained from payload.

Development of automating the workflow of running nmap, checking exploits and vulnerability, choosing payloads.

Results hasn’t been ascertained and evaluated yet cause the experiment is still in progress.

# Uncompleted work

Development of plugin to use in Metasploit to be completed by end of 31st March 2022

# Summary

The project is made good progress so far in terms of setting up the hardware and applications. Making iterative progress by making use of tools such as nmap, nessus, metasploit to generate and attack target machine using different payloads.

# Add the link to your project here:

The project’s documentation and application layer source code is maintained on github [gagansur/IoT-Pentesting (github.com)](https://github.com/gagansur/IoT-Pentesting)