University of Roehampton London

MSc Project -CMP060L050H

Project proposal

Project Title : Securing IoT Smart Home Automation with ESP8266 and TLS Encryption

STUDENT NAME : MOHAMED ABISHEIK MOHAMED ALI

STUDENT ID : A00037677

MSc. CYBERSECURITY

Title

Contents

Introduction

I'm working on my MSc project, where I'm designing a secure Internet of Things (IoT) system that controls four relays using an ESP8266 microcontroller. I'm programming it in C/C++ through the Arduino IDE and integrating it with Sinric Pro for voice control via Amazon Alexa and Google Home. This smart home automation system allows for both manual and remote relay control, with a strong focus on cybersecurity to tackle IoT vulnerabilities. My motivation comes from the rapid growth of IoT devices expected to hit 75 billion by 2025 and the increasing cyber threats, like the Mirai botnet, which took advantage of weak passwords and unencrypted communications. I chose this topic to gain practical experience in securing IoT systems, which aligns perfectly with my goal of becoming an IoT security engineer.

The project tackles significant security issues, such as hardcoded Wi-Fi credentials and insecure HTTP communication, by implementing solutions like WiFi-Manager, Transport Layer Security (TLS) encryption, and rate limiting. The need in the industry is clear 90% of IoT vulnerabilities arise from coding mistakes, according to the U.S. Department of Homeland Security, which puts user privacy and critical infrastructure at risk. My research follows the OWASP IoT Top 10 guidelines, contributing to the development of secure IoT solutions for smart homes. By analyzing and securing this system, I hope to offer valuable insights for both industry and academia, while also enhancing my skills in secure coding, threat assessment, and IoT protocol security. This project is a stepping stone for my career, helping me build the essential skills needed in the ever-evolving field of IoT cybersecurity, and ensuring safer connected environments.

Problem Statement

My MSc project dives into some serious cybersecurity issues plaguing Internet of Things (IoT) systems, particularly focusing on a smart home automation setup that uses an ESP8266 microcontroller to manage four relays. I programmed it through the Arduino IDE and linked it with Sinric Pro, Amazon Alexa, and Google Home. The main challenge here is the widespread security vulnerabilities found in IoT devices, like hardcoded passwords, unencrypted HTTP traffic, and the absence of rate limiting. These flaws can lead to unauthorized access, data interception, and even denial-of-service (DoS) attacks. For instance, the Arduino sketch I provided hardcodes Wi-Fi credentials and Sinric Pro keys, which could put the network at risk if someone extracts the firmware. Plus, the lack of encryption in communication with Sinric Pro opens the door for man-in-the-middle (MITM) attacks, jeopardizing both relay control and user data. These vulnerabilities reflect real-world IoT threats, like the Mirai botnet, which took advantage of weak credentials to take over devices.

The stakeholders affected by these issues include smart home users, IoT device manufacturers, and providers of critical infrastructure. Smart home users, who depend on systems like mine to manage their lights or appliances, could face privacy violations and unauthorized control of their devices, which might lead to safety hazards (think about appliances being turned on maliciously). Manufacturers, especially those making ESP8266-based devices, risk damaging their reputation and finances due to insecure products, as 66% of consumers steer clear of brands with known vulnerabilities. Critical infrastructure, which is becoming more intertwined with IoT, faces a domino effect of risks from compromised devices, as evidenced by attacks on power grids. The OWASP IoT Top 10 points out weak credentials and insecure network services as major vulnerabilities, highlighting the far-reaching consequences.

This issue is really important for a few key reasons. For starters, the IoT market is expected to balloon to 27 billion devices by 2025, which means cybercriminals will have a much larger playground to exploit. Additionally, insecure IoT devices can seriously undermine user trust and safety, with a staggering 90% of vulnerabilities arising from coding errors, according to the Cybersecurity and Infrastructure Security Agency (CISA). On top of that, regulations like GDPR demand secure data management, making it vital to have strong IoT security in place to steer clear of legal troubles. My project aims to tackle these challenges by using solutions like WiFi-Manager to get rid of hardcoded credentials, implementing TLS encryption for safe communication, and applying rate limiting to fend off DoS attacks. By addressing these weaknesses, my research meets the industry's growing need for secure IoT development, paving the way for safer smart homes and helping me achieve my goal of becoming an IoT security engineer. Solving this problem not only safeguards users but also pushes forward both academic and industry initiatives to standardize secure IoT practices, ultimately reducing risks in our increasingly connected world.

Aims and Objectives

My MSc project is all about creating a secure Internet of Things (IoT) system that can control four relays using an ESP8266 microcontroller. I'm programming it in C/C++ through the Arduino IDE and integrating it with Sinric Pro for voice control via Amazon Alexa and Google Home. A big focus of mine is tackling critical cybersecurity vulnerabilities. The main issue I'm addressing is the insecurity of IoT devices, which often have problems like hardcoded credentials, unencrypted HTTP communication, and are vulnerable to denial-of-service (DoS) attacks, as highlighted in the Arduino sketch. These issues can lead to unauthorized access, data interception, and misuse of devices. By fixing these weaknesses, I hope to build a secure smart home automation system that not only enhances the safety of IoT deployments but also helps me on my journey to becoming an IoT security engineer.

Objectives:

- Implement a secure way to manage credentials, so we can get rid of hardcoded Wi-Fi and Sinric Pro keys.
- Set up encrypted communication using TLS to ensure that data exchanged with Sinric Pro is safe and sound.
- Add rate limiting and intrusion detection measures to guard against DoS attacks and unauthorized access.
- Assess the system's security by conducting penetration testing focused on the OWASP IoT Top 10 vulnerabilities.
- Document our findings to offer valuable insights for developing secure IoT solutions in smart homes.

Approach:

I plan to tweak the Arduino sketch to include some secure practices. For the first goal, I'll utilize the WiFi-Manager library to set up Wi-Fi credentials on the fly, while storing Sinric Pro keys in EEPROM with a bit of basic obfuscation to keep them safe from firmware extraction risks. Moving on to the second goal, I'll implement WiFiClientSecure for secure TLS communication with Sinric Pro, which will help ensure that our data remains confidential and intact. For the third goal, I'll introduce rate limiting in the onPowerState function and keep an eye on any suspicious activities, like rapid relay toggles, to help with intrusion detection. Lastly, I'll carry out penetration testing using tools such as Wireshark to simulate man-in-the-middle attacks and Kali Linux to try and extract credentials, all while assessing how effective our mitigation strategies are. The tech I'll be using includes the ESP8266 (NodeMCU), Arduino IDE (C/C++), Sinric Pro, and libraries like ESP8266WiFi and WiFiClientSecure.

When it comes to research strategies and methods, a mixed-methods approach seems to be the best fit, blending experimental development with empirical testing. I'll develop and test the secure IoT system in a controlled lab setting, using quantitative metrics (like how well we prevent attacks) alongside qualitative analysis (such as how user-friendly the secure features are). The penetration testing will adhere to OWASP guidelines, and I'll conduct a literature review to frame my findings within the broader context of existing IoT security research. This comprehensive approach aims to create a robust and secure system, contributing to both industry and academic advancements in IoT cybersecurity.

Legal, Social, Ethical and Professional Considerations

I'm working on controlling four relays using an ESP8266 microcontroller through the Arduino IDE, and I've integrated it with Sinric Pro, Alexa, and Google Home. This setup brings up a bunch of important legal, social, ethical, and professional issues. On the legal side, I need to make sure my system complies with GDPR since it processes user data, like voice commands and relay states, which get sent to cloud services such as Sinric Pro and Alexa. If I don't secure this data properly, for instance, by not using TLS, I could face fines that could reach up to €20 million. To avoid that, I plan to implement encryption to keep user privacy intact. From a social perspective, if IoT devices aren't secure, they can really damage user trust. In fact, about 70% of consumers steer clear of brands that have known vulnerabilities. If things like hardcoded credentials or unencrypted HTTP communication are exploited, users could end up facing serious risks, like unauthorized control over their home devices, which could compromise their safety—imagine someone maliciously toggling your appliances! To counter this, my project will use WiFi-Manager and TLS, which should help build trust in smart home technology.

The collection and transmission of user data to cloud services raises some significant privacy concerns. I need to ensure that users give informed consent for their data to be processed, and I have to be transparent about how Alexa and Google Home manage voice data. Additionally, I'll conduct penetration testing to simulate attacks, like man-in-the-middle (MITM) attacks, while following ethical hacking guidelines to prevent any unintended harm. On the professional front, as a cybersecurity practitioner, I have to stick to standards like the OWASP IoT Top 10 to ensure I'm following solid security practices. If I fail to secure the system, it could really hurt my professional reputation. By focusing on secure coding, thorough testing, and proper documentation, I aim to maintain my ethical and professional integrity, ultimately contributing to safer IoT ecosystems.

Background

My MSc project is all about creating a secure Internet of Things (IoT) system that can control four relays using an ESP8266 microcontroller. I programmed it in C/C++ through the Arduino IDE and integrated it with Sinric Pro, allowing for voice control via Amazon Alexa and Google Home. This project tackles some serious cybersecurity issues that IoT devices face, like hardcoded passwords, unencrypted communication, and vulnerability to denial-of-service (DoS) attacks. The goal is to enhance the security of smart home automation. In the background section, I dive into the key literature on IoT security, place my project within the smart home landscape, evaluate how it stacks up against the latest practices, and look at the maturity, novelty, and relevance of the research area.

Literature Review: IoT Security:

The rapid rise of IoT devices, expected to hit 27 billion by 2025, has transformed smart homes, allowing us to control appliances like lights and thermostats from afar. However, this growth has also increased cybersecurity threats. The Mirai botnet, which took over thousands of IoT devices in 2016 by exploiting weak passwords, showcased the disastrous potential of insecure systems, creating botnets that disrupted internet services. Following that, research by Alrawi et al. (2019) pinpointed common vulnerabilities in IoT, such as weak authentication, unencrypted communication, and insecure firmware updates, with a staggering 60% of devices failing basic security checks. The OWASP IoT Top 10 framework highlights these risks, identifying weak credentials, insecure network services, and the absence of secure update mechanisms as major concerns. My project is designed to tackle these vulnerabilities head-on, with a focus on securing an ESP8266-based system.

Smart home IoT systems frequently depend on budget-friendly microcontrollers like the ESP8266, which are popular for their WiFi capabilities and low cost. A study by Fernandes et al. (2016) highlighted some security vulnerabilities in these devices, such as hardcoded credentials and unencrypted HTTP communication, which I noticed in the initial setup of my Arduino sketch (for example, the hardcoded WiFi SSID for Redmi 8 and Sinric Pro keys). These weaknesses can lead to man-in-the-middle (MITM) attacks and unauthorized access, putting user privacy and safety at risk. Recent research suggests several ways to address these issues: secure credential storage (like using EEPROM), TLS encryption, and rate limiting to fend off DoS attacks. For example, Kolias et al. (2018) recommend using dynamic authentication and encrypted protocols to safeguard IoT communication, which aligns perfectly with the goals of my project.

Cloud-based IoT platforms, such as Sinric Pro, make it easier to connect with voice assistants like Alexa and Google Home, but they also bring new risks. Research by Apthorpe et al. (2017) shows that cloud services often gather sensitive user information (like voice commands), making it essential to implement strong encryption and data minimization practices to meet GDPR requirements. The GDPR enforces secure data processing, with potential fines reaching €20 million for non-compliance, highlighting the importance of TLS in my system. Additionally, penetration testing, as recommended by OWASP, is a common approach to assess IoT security. Alaba et al. (2017) demonstrated this by using tools like Wireshark to identify MITM vulnerabilities.

Project Context:

This project is all about the smart home IoT ecosystem, where devices like the ESP8266 make automation possible, but they also come with some serious security issues. The backdrop here is the growing popularity of smart home tech, with a whopping 41% of U.S. households expected to use IoT devices by 2024, according to Statista. People want to control their devices effortlessly with voice assistants, but security vulnerabilities can really shake their trust—70% of consumers steer clear of brands that have known security problems. My system, which manages relays for various appliances, is a classic example of smart home applications, but it puts a strong emphasis on cybersecurity to tackle these worries. The project is being carried out in a controlled lab setting, utilizing the ESP8266 (NodeMCU), Arduino IDE, and Sinric Pro, along with penetration testing to mimic real-world attacks.

Relationship to State of the Art:

My project is in line with the latest IoT security practices while also tailoring them to a specific scenario. Current research highlights the importance of secure communication (like TLS), dynamic credential management (think WiFi-Manager), and intrusion detection, as noted in studies by Sicari et al. (2015). I've implemented WiFiClientSecure for TLS and WiFiManager to avoid hardcoded credentials, reflecting these advancements. However, the challenge of integrating these solutions into a budget-friendly ESP8266 system with Sinric Pro hasn't been widely explored, as most research tends to focus on high-end devices or broad frameworks. My project fills this gap by applying cutting-edge techniques to a practical smart home system, making it more accessible for those on a budget.

Penetration testing, guided by the OWASP IoT Top 10, represents a cutting-edge approach, but applying it to ESP8266-based systems with cloud integration (like Sinric Pro) is quite innovative. While Fernandes et al. (2016) explored similar devices, their focus was on proprietary platforms rather than open-source solutions such as Sinric Pro. My project takes this a step further by assessing cloud-to-device security, filling a significant research gap. Additionally, the incorporation of rate limiting and intrusion detection logs, inspired by Kolias et al. (2018), aligns perfectly with the latest trends in IoT resilience.

Maturity of the Research Area:

IoT security has become a well-established field, with a wealth of literature emerging since the early 2010s, largely due to incidents like Mirai. Standards such as the OWASP IoT Top 10 and NISTIR 8259 offer robust frameworks for securing devices. However, specific applications like securing low-cost microcontrollers with cloud-based voice control are still relatively uncharted territory, especially for open-source platforms like Sinric Pro. While the field is mature in terms of general principles (like encryption and authentication), it is still evolving in practical implementations for resource-constrained devices, making my project both timely and relevant.

This project builds on previous research rather than starting from scratch. While studies like those by Alrawi et al. (2019) and Fernandes et al. (2016) have looked into IoT vulnerabilities, there's a noticeable gap when it comes to securing systems based on the

ESP8266 with Sinric Pro integration. My work takes it a step further by implementing and testing specific solutions like WiFi-Manager, TLS, and rate limiting within a real-world smart home setting. The blend of penetration testing, cloud integration, and affordable hardware is what makes this approach unique, contributing something fresh to the field of IoT security research.

The techniques and theories I'm using are tried and true. Secure coding practices in C/C++ with the Arduino IDE, TLS encryption through WiFiClientSecure, and WiFi-Manager for managing credentials are all standard methods that you can find in the literature. Penetration testing, guided by OWASP and utilizing tools like Wireshark, is a well-established methodology. The foundational theories of IoT security—confidentiality, integrity, and availability (the CIA triad)—form the backbone of my approach, as highlighted by Sicari et al. (2015). These established techniques not only ensure that the project is feasible but also allow for practical application to a specific system.

The results of this project are bound to catch the attention of both industry professionals and academics. Companies that make smart home devices, like TP-Link and Philips, are on the lookout for budget-friendly security solutions that work well with low-cost microcontrollers such as the ESP8266, since having insecure devices can really hurt their brand reputation. My proposed solutions, including WiFi-Manager and TLS, provide practical insights for creating secure and affordable products. Cybersecurity firms like Fortinet and Palo Alto might find the penetration testing results particularly useful, as they can help shape their vulnerability assessment practices. From an academic standpoint, this project adds valuable knowledge to the field of IoT security research, especially for open-source platforms, which could pique the interest of conferences like BSides. Additionally, regulatory bodies focused on enforcing GDPR may find the compliance strategies I've developed to be quite relevant. By tackling real-world IoT security issues, my project has a wide-ranging appeal that extends well beyond the university setting.

References

- What is the Mirai Botnet? - https://www.cloudflare.com/learning/ddos/glossary/mirai-botnet/
- Alaba, F. A., Othman, M., Hashem, I. A. T. and Alotaibi, F. (2017) 'Internet of Things security: A survey', Journal of Network and Computer Applications, 88, pp. 10–28. Available at: https://www.sciencedirect.com/science/article/abs/pii/S1084804517301455
- IoT Analytics (2024) State of IoT 2024. Available at: https://iot-analytics.com/number-connected-iot-devices/
- National Institute of Standards and Technology (NIST) (2020) NISTIR 8259: Foundational cybersecurity activities for IoT device manufacturers. Available at: https://www.nist.gov/itl/applied-cybersecurity/nist-cybersecurity-iot-program