Smart Home Climate Automation

EEL5739C IoT Security and Trust
Group: 15

Dieter Steinhauser
Department of Electrical and Computer Engineering
University of Florida
Gainesville, Florida, USA
dsteinhauser@ufl.edu

Elizavetta Stetsenko
Department of Electrical and Computer Engineering
University of Florida
Gainesville, Florida, USA
e.stetsenko@ufl.edu

Lucas Mueller
Department of Electrical and Computer Engineering
University of Florida
Gainesville, Florida, USA
lucas.mueller@ufl.edu

Nikodem Gazda
Department of Electrical and Computer Engineering
University of Florida
Gainesville, Florida, USA
ngazda@ufl.edu

***Abstract*— The Smart Home Climate System is a proposed IoT system that implements IoT Security policies reviewed in course materials. Communication security in the design is assured using RSA encrypted MQTT with AWS brokerage. Node devices publish temperature and humidity data which are interpreted by a thermostat subscriber. Individual nodes implement security practices to obfuscate communication keys and sensor data.**

I. Introduction

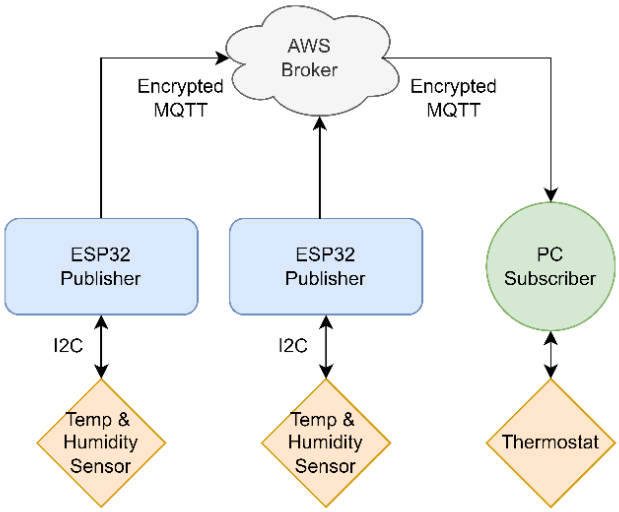

Figure 1: Network Diagram.

The proposed IoT application is a long house AC control system that aims to maintain optimal temperature and humidity levels throughout the house. It consists of two or more IoT devices: one or more devices monitor the temperature and humidity from multiple ends of the house, while the receiving device receives that data and controls the AC and dehumidifier based on predefined thresholds. The AC and dehumidifier activate when any of the devices report a temperature or humidity outside of a desired range. This system ensures that the entire house stays comfortable, even in areas that are distant from the AC unit. This system is particularly beneficial for long houses or homes with a sprawling layout, where the temperature and humidity levels can vary significantly between different areas. It can be deployed in residential buildings and commercial spaces such as offices or retail stores with long corridors or multiple rooms. It would allow the user to dictate climate thresholds for each room individually and the system will maintain them.

II. Methods

A. System Design

Implementation of the home climate system, we are using two ESP32 devices acting as nodes in our system. They interact with temperature and humidity sensors via I2C communication. These nodes are placed throughout the home to collect data and publish information via MQTT to the broker. Nodes are secured in difficult to notice spaces and are placed within tamper proof housings. I2C lines would also be obfuscated so that data would not be easily sniffed. MQTT is secured using RSA encryption and communicating using the port 8883 for MQTT. The broker of our system is AWS as it provides both physical and networked security using their remote cloud infrastructure. A smart thermostat would also use RSA encrypted MQTT as a subscriber to listen to node data. Temperature and humidity data that goes beyond thermostat thresholds interacts with the Air handler fan and heat exchanger to manage the climate.

B. Security Challenges

Some of the system's challenges include securing the ESP 32 devices such that they are not easily accessible. Without a password or physical safeguards, the device can be accessed through the debugging ports. Through this, a malicious third party could forge temperature readings from the publishing device, alter the temperature thresholds set by the subscribing devices, or disable the devices altogether. This could result in the threat of an attacker overloading the AC resulting in a raised AC bill or even fire.

C. Risk Assessment

To compile a thorough risk assessment, we first looked at the NIST risk rating scale. In the document SP 800-30, NIST provides guidance on risk assessment methodologies. This prompted us to categorize our risks in terms of their threat and vulnerability levels. In this case, threat describes the impact this risk poses to the system, and vulnerability describes the likelihood of the risk being exploited. The magnitude for both these metrics is in our risk matrix, seen in Figure 2.

Some of the physical risks to our system include damage to the board and gaining access to the ports. Damage to the board would be very bad for the system so its impact rating is high, however, considering that this device will likely be located inside, the likelihood of this happening is rather low, and that is why it has a low vulnerability score. When this risk is put into the CVSS calculator, it gets a 4.6/10 score. It is noteworthy that the board's physical security is an example of risk transference, ultimately the homeowner's responsibility. We also considered Network risks, for example if an adversary were to be sniffing packets on the network, they could gain information about the network they shouldn't have access to. If data concerning the AC is being sent over the network, it could be used to determine when the homeowner is not home and could therefore lead to a burglary threat. This vulnerability receives a CVSS score of 6.6/10. Finally, we considered some software risks as well. One very common threat to any system is when the software goes out of date. Especially when dealing with software that handles encrypted information, it is important to ensure that said software is kept up to date to maintain high levels of security. Out of date software can compromise an entire system, this vulnerability gets a CVSS score of 5.3/10.

TABLE I-2: ASSESSMENT SCALE – LEVEL OF RISK (COMBINATION OF LIKELIHOOD AND IMPACT)

Likelihood (Threat Event Occurs and Results in Adverse Impact)	Level of Impact				
	Very Low	Low	Moderate	High	Very High
Very High	Very Low	Low	Moderate	High	Very High
High	Very Low	Low	Moderate	High	Very High
Moderate	Very Low	Low	Moderate	Moderate	High
Low	Very Low	Low	Low	Low	Moderate
Very Low	Very Low	Very Low	Very Low	Low	Low

Figure 2: Risk Matrix.

D. Proposed Safeguards

Safeguards were implemented to address the risks found in the risk assessment. To address unauthorized data access and endpoint security, the design would include using the cryptographic core of the ESP32 to store network configuration data and communication keys to avoid access of sensitive data from node device. I2C communication lines are obfuscated in system design to limit sensor data sniffing. Data is transferred using encrypted protocols to secure data from Wireshark sniffing and man-in-the-middle attacks (MitM). AWS, being a remote broker, provides physical and network security for the system, and transfers some of the risk to the AWS secure data centers. The thermostat device incorporates similar hardware and software security practices. VLANs can also be configured to minimize network exposure and contact with general traffic.

III. Results

A. Design Implementation

We effectively implemented an AWS brokered MQTT system using TLS/SSL encryption between publisher and subscriber nodes using the ESP32 lab kits and a Mosquitto enabled PC. The two lab kits effectively sampled their sensors for data and published the information to the single MQTT topic. A PC with a python script effectively subscribed to the topic with the same encryption standards and performed calculations to interact with an HVAC system.

IV. Discussion

A. Presentation Questions

Please discuss how you assigned each risk its risk score. E.g., how did you determine causing damage to the board has "significant" consequences vs "catastrophic" or "moderate". Can you produce a more objective way to determine risk?

NIST provides a Common Vulnerability Scoring System Calculator in which you fill in a set of parameters according to the specifications of your system and receive a resulting vulnerability score. This score also includes an assessment of the impact metrics. For example, when considering the vulnerability of physical access/damage to our device, we filled in the calculator as described by Figure 3.

This risk requires an attacker to physically touch or manipulate the vulnerable components, rendering a physical attack vector (AV:P), low complexity (hammer) (AC:L), high privilege because a physical attack on the system would require entry to the home (PR:H), no user interaction (UI:N), unchanged scope (S:U), no confidentiality impact since damage to the board doesn't provide the attacker with any new information (C:N), low integrity impact since the attacker can't modify files but can alter sensor readings (I:L), and a high availability impact, since the attacker can deny full access to the system by destroying the component (A:H). Given these selections, the physical access/damage risk received an overall CVSS score of 4.6. We were able to leverage the descriptions that pop up when you hover over any parameter to accurately assess the vulnerability of each risk in our system.

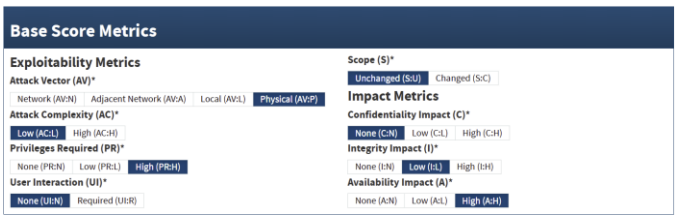

Figure 3: Example of the filled in Common Vulnerability Scoring System Calculator for the risk of physical damage to the board.

We asked in the Q&A what is the difference between the risk of port access vs the risk of RSA private key access and the answer you gave was along the lines of the RSA private key

being recovered via software rather than from the port. Could you try and describe how this might happen? Describe how you might recover the RSA private key from a device without physical access to the board.

RSA private key access being compromised poses a more severe risk than port access since it directly affects encrypted communications, since access to RSA private keys means that a malicious party can decrypt secure communication. Port access does not directly compromise encrypted communication, since unauthorized port access requires additional steps to access secure data, which might not even occur if it is a well secured system. Port access threats can be easily mitigated through firewalls to restrict access, or regular scanning of ports to check for abnormal changes. Unauthorized RSA private key access requires more demanding mitigation strategies, such as using secure encryption and storing the private keys in secure hardware. Recovering an RSA private key via software without physical access to the board is reliant on exploiting the weaknesses in the security of the system, such as not properly configured file transfer services, faulty implementation of encryption algorithms, and man in the middle attacks if the exchange process is not secured.

V. Conclusion

Overall, the IoT Smart Climate System worked as expected with ample safeguards for data security and integrity in a deployment environment. While there are hardware, software, and network considerations for improving the design security, threat of the system design is low, and vulnerability is mitigated through many safeguards to ensure a low-risk deployment.