Name : K. Lakshiita
College : SRM Institute Of Science and Technology
Register ID: RA2211026010216
BTECH CSE - AI & ML

Internship : Atsuya Technologies
Date : June 3,2024 to July 7,2024

Digital Lock Project Report Using STSAFE-A110 for Key Generation, Encryption, and Decryption

Introduction :

In this project, I developed a digital lock system utilizing the STSAFE-A110 secure element for key generation, encryption, and decryption, with ESP32 microcontrollers serving as the core hardware for both the lock and key. The primary focus of this project was to create a secure and reliable digital locking system with a subscription-based model for generating and managing keys. The development environment used was Arduino IDE.

System Overview :

The digital lock system consists of two main components:

1. Lock Unit: An ESP32 microcontroller controls the locking mechanism, communicates with the secure element (STSAFE-A110), and processes the received keys for validation.
2. Key Unit: Another ESP32 microcontroller acts as the digital key, which generates and transmits the necessary cryptographic keys for unlocking the lock unit.

Both the lock and key units communicate wirelessly, utilizing the ESP32's Wi-Fi capabilities for secure transmission.

STSAFE-A110: Secure Key Generation

The STSAFE-A110 is a highly secure hardware element designed to handle cryptographic functions. In this project, the STSAFE-A110 was employed for generating, storing, and managing cryptographic keys.
Public and Private Key Pair Generation: The STSAFE-A110 generates a pair of public and private keys securely within the hardware. The private key is kept within the STSAFE-A110, ensuring that it is never exposed to potential threats. The public key is used for communication with the ESP32 microcontroller and for sharing with the key unit.

Encryption and Decryption: The STSAFE-A110 handles all encryption and decryption tasks using the generated keys. This ensures that sensitive data, such as the keys themselves and any related communication, is protected against unauthorized access.

ESP32 Microcontrollers: Lock and Key Units

Two ESP32 microcontrollers were used, one for the lock unit and the other for the key unit.

Lock Unit:The ESP32 in the lock unit is responsible for interfacing with the STSAFE-A110, controlling the locking mechanism, and processing the encrypted data received from the key unit. Upon receiving the encrypted key from the key unit, the ESP32 passes it to the STSAFE-A110 for decryption and validation. If the decrypted key matches the expected value, the lock is disengaged.

Key Unit:The key unit's ESP32 generates a request for a new key from the STSAFE-A110, which creates a subscription-based key. The key is then encrypted and transmitted to the lock unit. The key unit can also be configured to periodically request new keys, ensuring that the system remains secure over time.

Subscription-Based Key Management

A unique feature of this project is the implementation of a subscription-based key generation system. The STSAFE-A110 is configured to generate new keys periodically or upon request, ensuring that keys are rotated regularly. This approach enhances security by minimizing the risk of key exposure or reuse.

Each new key generated by the STSAFE-A110 is associated with a specific time or usage period, after which it becomes invalid. This subscription-based model is particularly useful for applications where access needs to be granted for limited periods, such as in rental properties or shared workspaces.

Development Environment

The entire project was developed using the Arduino IDE, a versatile and user-friendly environment for programming microcontrollers like the ESP32. The Arduino libraries for both the ESP32 and the STSAFE-A110 were utilized to streamline development and ensure compatibility between the hardware components.

Conclusion

This digital lock system project successfully demonstrated the integration of secure hardware (STSAFE-A110) with microcontroller-based systems (ESP32) for developing a robust, subscription-based key management system. The use of public and private key encryption, combined with the periodic generation of new keys, ensured that the system was both secure and flexible, making it suitable for a wide range of applications. The project also highlighted the capabilities of the Arduino IDE in managing complex hardware interactions and cryptographic processes.