Proximity Power: Enhancing IoT Security through Proximity Security

Benjamin George Rutkowski *brutkows@uci.edu*
Tylor Langford *tylorl@uci.edu*
Ryan Ngoon *rngoon@uci.edu*
Hongzhuo Chen *hz.chen@uci.edu*

Abstract

1 Introduction

The proliferation of Internet of Things (IoT) devices has ushered in an era where ubiquitous connectivity and intelligent automation have become integral to daily life and industrial operations. These devices, ranging from smart home appliances to complex industrial systems, communicate and interact within expansive networks, often relying on proximity-based solutions to facilitate seamless connectivity and functionality. However, the rapid adoption and integration of IoT technology have amplified security concerns, particularly regarding the vulnerability of proximity-based communication protocols to adversarial attacks, necessitating rigorous analysis of existing solutions to fortify IoT ecosystems against emerging threats.

Proximity-based solutions in the IoT realm are pivotal for ensuring efficient and effective communication between devices that are geographically close. Technologies such as Bluetooth Low Energy (BLE), Near Field Communication (NFC), and Zigbee have been instrumental in enabling these interactions. Each of these technologies offers unique strengths, such as low power consumption, ease of deployment, and robust connectivity, which make them suitable for various IoT applications. However, these advantages come with inherent weaknesses that can be exploited, posing significant risks to the security and integrity of IoT networks.

This paper presents a comprehensive analysis of existing proximity-based solutions for IoT, scrutinizing their architectural frameworks, communication protocols, and security mechanisms. By evaluating these aspects, we aim to identify the strengths that can be leveraged and the weaknesses that need to be addressed to enhance overall IoT security. The analysis includes a comparative study of BLE, NFC, and Zigbee, examining their respective vulnerabilities and the potential impact of these vulnerabilities on IoT systems.

Through this investigation, we contribute to the body of knowledge in IoT security by highlighting critical areas that require improvement and proposing strategies to mitigate identified risks. Our findings are expected to inform the development of more secure proximity-based communication protocols and systems, ultimately strengthening the resilience of IoT networks against adversarial attacks. By addressing these security challenges, we pave the way for the safer deployment and operation of IoT devices, ensuring that the benefits of this transformative technology can be fully realized without compromising on security.

2 Context / Related Work

Existing research on IoT security primarily focuses on data encryption and network security, but less attention has been given to proximity-based authentication mechanisms, which are increasingly crucial as IoT devices become more prevalent in everyday life. IoT devices such as Smart Home Devices, Wearables, Automotive, Mobile devices, Retail and Payment, as well as Industrial robotics. The use cases for these devices vary greatly but generally are in the business of collecting and processing data around us and communicating that data to distributed or central repositories. We performed a review of several of these sectors to understand the current landscape of what components these devices generally have, the security methodology and communication protocols they use, and if adding a proximity based security layer would be both feasible and worthwhile.

According to IoT Analytics [1] over 88% of registered IoT connections (>14 billion devices) are split between three major communication protocols: Wifi at 31%, Bluetooth at 27%, Cellular at 20%,, with the remaining split between various other methods such as Zigbee, and LoRaWAN. Within these protocols, the main security architecture is reliant on an encryption scheme based on a set of shared keys. The key insight into proximity based security is not to create an entirely new method for encrypting the communication between these devices, but in allowing a way to quickly and effectively share these keys with limited to no user interaction or user interface and with as small a compute and memory footprint as possible.

Within the realm of proximity-based security, several technologies and protocols have been proposed and implemented, each with its own set of advantages and limitations. One prevalent approach involves the use of Bluetooth Low Energy (BLE) technology, which enables devices to establish short-range wireless connections for data exchange. BLE-based proximity solutions often rely on signal strength and device proximity estimation to determine trust levels and authenticate communication partners. While BLE offers low power consumption and widespread compatibility, it may be susceptible to relay attacks and signal interference in densely populated environments.

In addition to BLE, Near Field Communication (NFC) and Radio Frequency Identification (RFID) technologies have also been explored for proximity-based authentication in IoT settings. NFC enables devices to communicate over short distances (typically a few centimeters), making it suitable for secure transactions and access control applications. Similarly, RFID systems utilize radio waves to identify and track ob-

jects within proximity, offering scalability and robustness for IoT deployments. However, both NFC and RFID may face challenges related to range limitations, interoperability, and susceptibility to cloning and spoofing attacks.

Despite the growing interest and investment in proximity-based security for IoT, there remains a lack of comprehensive understanding regarding the performance, reliability, and vulnerabilities of existing systems. Moreover, the dynamic nature of IoT deployments introduces unique challenges, such as device heterogeneity, network congestion, and physical obstructions, which may impact the effectiveness of proximity-based authentication mechanisms. Therefore, it is essential to conduct a survey of related work and past approaches to proximity-based security in IoT environments to inform our research and identify areas for improvement.

In addition to short range communication protocols (that can be taken advantage of for proximity-based security), the prevalence of environment sensing capabilities in IoT devices has led to research into gesture based or physical stimuli to further ensure a trusted relationship between communicating devices. The research into this avenue of proximity-based security is an expanding field and utilizes several different methods to try and capture and distill gestures and movement into unique identifiers. The challenge here, as with the different short range communication technologies, is the true durability of these approaches against concerted adversaries. As the IoT sector grows and these different technologies get adopted into a larger family of devices, it will be import to analyze these approaches to provide insight into the different angles of defense that need to be shored up, along with novel new implementations of these concepts into hardened systems that are still user-centric.

3 Approach

This project will adopt a comprehensive approach to analyze proximity-based security solutions in IoT environments. We will begin by conducting a thorough literature review to identify existing proximity-based systems and their underlying technologies. Through this process, we aim to establish a comprehensive understanding of the various technologies, protocols, and deployment architectures employed in proximity-based authentication systems. Subsequently, we will evaluate these systems through a combination of attempting simulated attacks, vulnerability assessments, and experimental measurements. These attacks may include relay attacks, eavesdropping, man-in-the-middle attacks, and spoofing attempts. By systematically probing the systems' defenses and analyzing their responses to adversarial stimuli, we can uncover vulnerabilities and assess the robustness of the authentication mechanisms. Additionally, we will explore novel approaches to enhance the security of proximity-based authentication in order to offer a more holistic view of the landscape.

In order to accomplish these tasks we will be attempting to implement three tools on an actual microcontroller (ESP32) to get a full suite of measurements and attack vectors. Our plan was to evaluate the following proximity tools as a representation of the types of proximity based security mechanisms in use today.

1. Move2Auth [2]: This is a tool that uses RSS field strength and an accompanied gesture by a pairing device to create a trust relationship. The general idea is that the device requesting to join will only accept communication and updated keys from a pairing device that establishes it is a secure device by being close enough to induce a signal strength change and follow a physical gesture that is generated at the time of pairing. Despite this local communication, the key is still shared over a network as cleartext since a cipher can't be established until a key is provided.

2. T2Pair [3]: This idea uses a very similar approach to Move2Auth but relies on a physical button or actuator to provide a unique physical signal. The idea is that this button press can be measured and the timing of the action can be used as method to see a cipher or key to enable secure communication between a device with minimal or no UI and a secure network. This system utilizes some random delays in order to prevent spoofing or mimicking by a local attacker, and may be a possible key approach to improving the Move2Auth baseline system.

3. Secure Beacon [4]: This is more utilized for performing secure transactions at something like a business or corporation where employees may need to make regular transactions that need to be secure but should not be able to complete those actions if they are not onsite. There a several different possible uses for this, and potentially could be used to augment the gesture methods above by creating a mesh of secure devices that you would need to interact with to prove identity before being accepted into the local network.

Each of these tools utilizes different but similar methods for implementing proximity based enhanced security to IoT devices. By the end of our experimentation we plan to be able to provide a comprehensive overview of proximity's effectiveness as a tool for secure pairing with IoT devices, as well as the effectiveness of the different styles of proximity checking and where these tools should be implemented.

We were only able to implement Move2Auth on our local devices due to time constraints, and leave further comparison for future work.

4 Threat Model

We describe the threat model that is assumed throughout the context of this paper. There are three types of agents a

Target IoT device, a *Personal IoT* device and *Malicious* devices. Target IoT devices are public, that is anyone has access to the public interface. Personal IoT devices are private and unique to a user and are mobile, *e.g.*, smartphone, smartwatch, key fob, *etc.*. The goal of the Target device is to accurately identify when a *valid* Personal device is within physical proximity of the Target device itself. Note, the Target devices publicity allows any arbitrary Personal device to *attempt* to connect/pair/unlock *etc.* but it will only allow valid Personal devices to do so. Furthermore, the valid set of Personal devices is arbitrary so it may also include *any* Personal device that is within proximity of the Target device.

Malicious devices are a set of one or more devices controlled by a Malicious agent with the goal to trick Target devices to receive false information about the physical proximity of certain Personal devices. For example, a Malicious device may want to trick the target device that an existing Personal device is within proximity of the Target when in reality it is not; this could be used, for example, to unlock a car when the key-owner is away. The Malicious agent may also want to disguise itself as a novel Personal device that the Target device falsely accepts as a valid Personal device.

Malicious devices can listen to any signal transmitted by any device at any frequency or with any protocol. They also have the ability to transmit any signal or send any message from any protocol. They do not have access to any secret information on any non-malicious device, Target or Personal, *e.g.*, private keys. They do have the freedom to roam anywhere physically and multiple Malicious devices may be deployed together to initiate an attack. That is, one device may be located within proximity of a Target device and another may be located near by a Personal device that the attackers may want to exploit.

5 Rubric

Effectiveness Against Attacks The measure of an approaches resistance to the following attacks: Brute force, Man-In-The-Middle (MITM), and Relay. These were evaluated based on research and simulated attacks/implementations using the following scoring:

Ease of Implementation Measure of the complexity of the integrating the solution with existing hardware and software, as well as the need for either specialized equipment of difficult to reproduce or tuneable algorithms.

User Experience Measure of the intuitiveness and learning curve associated with each method. Intended to capture the probability of an method being accepted and used as intended by a user.

Evaluated based on similar approaches and general team consensus.

Cost Analysis Measure of implementation cost based on direct and supporting hardware as well as possible added processes during manufacturing.

Score	Description
3	Excellent: Demonstrates robust resistance to brute force, MITM and Relay attacks; comprehensive security audits and simulated attack scenarios show no vulnerabilities.
2	Satisfactory: Moderate resistance; security audits and simulated attacks reveal some vulnerabilities, but they are manageable and have mitigation strategies.
1	Poor: Limited resistance; numerous and critical vulnerabilities found, showing a lack of effective security measures

Table 1: Attack Resilience

Score	Description
3	Integration is straightforward with existing hardware and software; no specialized equipment or algorithms needed
2	Satisfactory: Integration is feasible but requires some modifications and specialized components or algorithms.
1	Poor: Integration is complex; significant modifications and specialized equipment or algorithms are required.

Table 2: Ease of Implementation

Costs are based on 1000 unit purchases and could possibly be lower at scale. The given costs included parts sources from digikey and were based on components mentioned in the papers or used during testing. The price cut-offs were based on cost being less than a percentage of the list price of the lower end devices we reviewed on the marked.

System Compatibility Measure of compatibility with current and future IoT devices, along with scalability of the method.

6 Methodology Grading

We will now evaluate the present defenses move2auth, secure beacon, and t2pair using our established rubric to establish a quantitative assessment of the security.

6.1 Effectiveness Against Brute Force Attacks

Move2Auth. This defense leverages physical movement patterns for authentication. The dynamic nature of human movements creates a high entropy environment, making brute force attacks impractical. The variability in movement patterns ensures that attackers cannot easily replicate the authentication

Score	Description
3	Excellent: Highly intuitive with minimal learning curve; positive user feedback and high usability scores
2	Satisfactory: Moderate learning curve; generally positive user feedback with some usability concerns.
1	Poor: Difficult to use with steep learning curve; negative user feedback and significant usability issues.

Table 3: User Experience

Score	Description
3	Excellent: Total implementation cost < $0.40 (< 1%)
2	Satisfactory: Total implementation cost < $2.00 (<5%)
1	Poor: Total cost > $2.00

Table 4: Cost Analysis

sequence.

Secure Beacon. This defense uses Bluetooth Low Energy (BLE) signals with rolling codes for authentication. The rolling code mechanism significantly enhances security against brute force attacks, as the authentication code changes frequently and unpredictably.

T2Pair. This defense employs touch-to-pair mechanisms, where devices must be physically tapped together. This direct physical interaction significantly mitigates the risk of brute force attacks by requiring the attacker to have physical access to the devices.

6.2 Effectiveness Against MITM Attacks

Move2Auth. The continuous and dynamic nature of movement-based authentication in Move2Auth complicates the execution of MITM attacks. Attackers would need to replicate precise movements in real-time, which is extremely challenging. Given an overall evaluation of 3: Excellent.

Secure Beacon. This defense's rolling code system and encrypted communication channels provide strong protection against MITM attacks. The frequent change of authentication codes makes it difficult for attackers to intercept and use the same code. However, with the addition of a relay attack, the becomes easily exploitable, and is therefore given a total score of 1: poor.

T2Pair. The physical proximity requirement of this defense inherently protects against MITM attacks. The necessity of direct physical contact means that intercepting communication without detection is virtually impossible. Because of the heterogeneous nature of this approach along with the natural entropy of a humans interaction with a device that are hard to monitor remotely, T2Pair is rated with a 3: Excellent.

Score	Description
3	Excellent: Highly compatible with a wide range of current and future IoT devices; excellent scalability and adaptability.
2	Satisfactory: Compatible with most IoT devices with some limitations; moderate scalability and adaptability.
1	Poor: Limited compatibility with IoT devices; poor scalability and adaptability.

Table 5: System Compatibility

6.3 Ease of Implementation

Move2Auth. Implementing Move2Auth requires integrating motion sensors and developing algorithms to accurately interpret movement patterns. While not overly complex, it does necessitate specialized hardware and software development. Because there is a certain amount of tuning and component matching required, this was rated as 2: Satisfactory.

Secure Beacon. This defense is relatively easy to implement with existing BLE infrastructure. The primary requirement is the development of secure rolling code algorithms, which can be integrated into current BLE-based systems with moderate effort. With the added challenge of needing to maintain secure locations, rolling codes, and manual distribution of PK's and requires an additional piece of hardware to tie two unrelated devices together, this was rated as 2: Satisfactory.

T2Pair. This is straightforward to implement, relying on existing NFC or similar proximity-based technologies. The primary implementation effort lies in ensuring reliable and secure communication during the touch interaction. Because this approach allows for such a large amount of flexibility without needing to create a specific measurement implementatoin per component, this was rated as 3: Excellent.

6.4 User Experience

Move2Auth. Users must perform specific movements to authenticate, which can be intuitive but may require some learning. The method's ease of use depends on the intuitiveness of the required movements and the accuracy of movement detection. The length of the gesture (>3 seconds) as well as the repeated nature of the process slightly detracts from the expected adoption and thus was rated 2: Satisfactory.

Secure Beacon. This defense operates seamlessly in the background, requiring minimal user interaction. This makes

it highly user-friendly, as authentication occurs automatically when the user is within the required proximity range. This approach requires almost no user interaction and provides seamless access via geofencing which makes it desirable from a user perspective and was therefore rated 3: Excellent.

T2Pair. This requires users to physically tap devices together, a simple and intuitive action. This method is user-friendly and easy to understand, ensuring broad usability across different user demographics. The ability to seamlessly use this approach across any device family is a large advantage, but due to some of the shared actions needed to be performed between devices we expect a certain amount of resistance to user adoption. Although only a slight drawback, compared to the seamless nature of SecureBeacon, T2Pair was rated 2: Satisfactory.

6.5 Cost

Move2Auth. The cost of implementing Move2Auth includes the expense of motion sensors and the development of movement recognition algorithms. These costs can be moderate to high, depending on the required accuracy and complexity. Based on our implementation, assuming a device already has some way to read signal strength, the additional hardware could be implemented for $2.00 and thus was rated 2: Satisfactory.

Secure Beacon. This relies on BLE infrastructure, which is relatively inexpensive and widely available. The main cost factor is the development of secure rolling code mechanisms, which is generally low to moderate. However, the general implementation for this requires a dedicated device to allow access to other devices thus incurring the cost of a device vs components in our evaluation. This lead to Secure Beacon being rated 1: poor.

T2Pair. This requires NFC or similar proximity-based hardware, which is inexpensive and widely integrated into many devices. The overall cost is low, making it a cost-effective solution for proximity-based authentication. This approach is able to additionally use almost any peripheral sensing device as along as some measurable action can be taken. Often this is accomplished with a button or similar sensor already incorporated into a device (since IoT is expected to have some external sensing), and thus even with a button integration the cost was expected to be <$0.40 and therefore is rated 3: Excellent.

6.6 System Compatibility

Move2Auth. This is compatible with systems that can integrate motion sensors, such as smartphones and wearable devices. However, its application might be limited in environments lacking these sensors. With the requirement of specific sensor sets that are not necessarily ubiquitous, but are very common in most smartphones today this was rated a 2: Satisfactory.

Secure Beacon. This is compatible with a wide range of BLE-enabled devices, including smartphones, tablets, and IoT gadgets. Its widespread compatibility makes it a versatile solution for various IoT applications. However, because it requires a fixed device and manual initial sharing it was rated 2: Satisfactory.

T2Pair. This defense is compatible with NFC-enabled devices, which are increasingly common in smartphones and other IoT devices. This broad compatibility ensures its applicability across diverse IoT ecosystems. The ability to use any peripheral sensing component across devices led this approach to be rated 3: Excellent.

6.7 Summary of Results

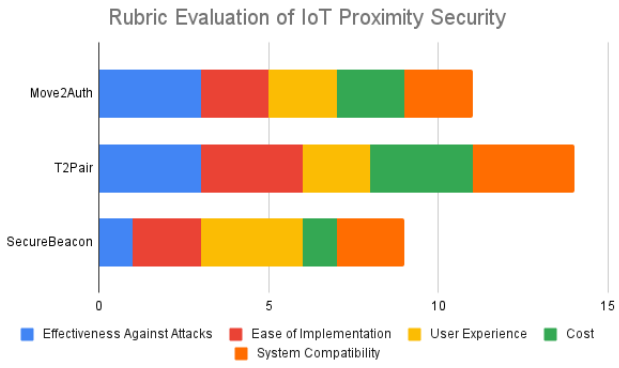

Figure 1: Comparison of different proximity based IoT security methods in standardized rubrick

We considered any method with a total rubric score greater than 11 to be a viable direction for further and deeper study, as well as ready to be implemented into the current IoT ecosystem. Move2Auth at 11 and T2Pair at 14 show great promise, and with some additional work on making them more user friendly and faster, there is a real opportunity to make IoT security both seamless and secure.

7 Qualitative Assessment

In this section, we offer qualitative analysis based on the cost, implementation and rubric scores of each defense from the previous section.

The comparative analysis of Move2Auth, Secure Beacon, and T2Pair reveals a nuanced landscape of proximity-based IoT security solutions, each with its distinct strengths and challenges. Move2Auth leverages physical movement patterns for authentication, providing a high level of security due

to the dynamic and unpredictable nature of human movements. This characteristic significantly enhances its effectiveness against brute force attacks, as the complexity and variability of movement patterns are challenging to replicate. Moreover, the continuous authentication process in Move2Auth mitigates the risk of MITM attacks, as attackers would need to perform real-time replication of movements, a feat that is technically demanding. However, the requirement for motion sensors and sophisticated movement recognition algorithms introduces a higher implementation cost and complexity, potentially limiting its adoption in environments lacking the necessary hardware infrastructure. Secure Beacon, on the other hand, employs BLE signals with rolling codes for authentication, striking a balance between robust security and ease of use. The rolling code mechanism is particularly effective in thwarting brute force and MITM attacks, as the frequent change of authentication codes makes it difficult for attackers to intercept and reuse codes. Secure Beacon's seamless operation in the background enhances its user-friendliness, as minimal user interaction is required, making it an attractive option for scenarios where ease of application is paramount. Additionally, the cost of implementing Secure Beacon is relatively low, given the widespread availability and affordability of BLE infrastructure. This makes it a versatile solution compatible with a broad range of IoT devices, from smartphones to industrial sensors. T2Pair stands out for its simplicity and cost-effectiveness. By requiring users to physically tap devices together for authentication, T2Pair provides a straightforward and intuitive user experience. This physical proximity requirement inherently protects against brute force and MITM attacks, as the need for direct contact between devices ensures that interception or replication of the authentication process is nearly impossible without physical access. The low cost of implementing T2Pair, due to the inexpensive and commonly available NFC hardware, further enhances its appeal. Its compatibility with NFC-enabled devices, which are prevalent in consumer electronics, ensures that T2Pair can be readily integrated into a wide array of IoT ecosystems. However, the necessity of physical contact may be seen as a limitation in scenarios where touchless interaction is preferred or required.

7.1 Move2Auth

Security Level and Mechanisms. Move2Auth provides a high level of security through the use of unique human movement patterns for authentication. It incorporates localized cryptography methods to enhance security, translating movement into cryptographic keys that are difficult to replicate, which adds an extra layer of protection against potential attacks.

Signal Required and Adoption. Move2Auth requires motion sensors such as accelerometers and gyroscopes, which are commonly found in modern smartphones and wearable devices. The widespread adoption of these sensors in consumer electronics facilitates the implementation of Move2Auth, making it a viable option for a broad range of applications. Minimum Hardware Capability: The minimum hardware capability required for Move2Auth includes a processor capable of real-time signal processing and movement recognition. Modern mobile processors and microcontrollers with integrated sensor hubs are sufficient to handle these tasks. Devices should also have sufficient memory to store movement patterns and cryptographic keys securely.

Computational/Power Loads. Move2Auth involves moderate computational and power loads due to the need for continuous movement monitoring and real-time cryptographic processing. The peak load occurs during the authentication process, while the average load depends on the frequency of authentication attempts. Modern mobile and wearable devices, designed to handle continuous sensor data processing, can efficiently manage these demands without significant impact on battery life.

Network and Communication Methods. Move2Auth primarily uses local device communication without relying heavily on network infrastructure. Bluetooth and Wi-Fi can be employed for transmitting authentication data between devices. While Bluetooth Low Energy (BLE) is a suitable option due to its low power consumption, alternatives like Ultra-Wideband (UWB) could offer better accuracy and security at the cost of higher power consumption.

Viability of Use. Move2Auth is highly viable for use in environments where security is paramount and the necessary hardware infrastructure is available. Its application spans from personal devices to secure access in smart homes and offices, leveraging the widespread presence of motion sensors in consumer products.

Major Vulnerabilities. A potential vulnerability of Move2Auth lies in the accuracy and sensitivity of the motion sensors. Inaccurate sensors or interference from external movements could lead to false rejections or acceptances. Additionally, sophisticated attackers with the ability to closely mimic the user's movements could pose a security risk, although this remains a challenging and low-probability threat.

Actual Implementation This was the only tool we were able to physically implement using 2 ESP32S3-DevkitC-1's [5] to simulate and IoT device and potential pairing device. This sandbox implementation was meant to help quantify several aspects of the approach including: development time and challenges, capability of RSSI matching at different power levels, ability to utilize existing hardware on different devices, and overall power consumption. Our experiments helped put the different components of proximity based security for IoT in perspective and had the following results. First, in order to match the RSSI signals, getting the sampling rate high enough with basic NimBLE libraries on the ESP32 was challenging and we were only able to sample at 10 Hz, but that was fast enough to make a matched signal with an MSE < 2.9 when

comparing the matched signals. However, to run this sampling the MCU's had an average power draw of 437 mW per device over the course of the 3 second gesture measurement. The IMU (LSM6DSOXTR) power was neglible compared to the bluetooth signal measurements at <3 mW. Less than 700 additional bytes of memory were required for all computations. We disregarded the memory needed to implement the Bluetooth layer as we assumed that would already be available on the devices under test. Under these conditions we were able to match the RSSI and IMU readings somewhat closely, but even with time matching, getting the IMU signal to match with the RSSI left room for improvement in our particular implementation.

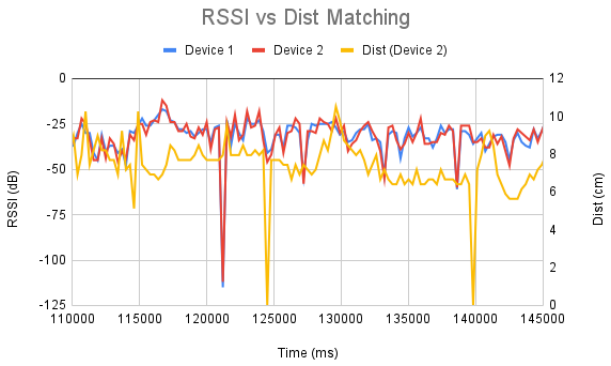

Figure 2: Matching 2 devices RSSI signals and an IMU to corroborate proximity/identity

7.2 Secure Beacon

Security Level and Mechanisms. Secure Beacon offers robust security by utilizing BLE signals combined with rolling codes for authentication. The rolling code mechanism ensures that the authentication credentials are constantly changing, making it difficult for attackers to reuse intercepted codes. This approach significantly strengthens security against both brute force and MITM attacks, as the codes are time-sensitive and encrypted.

Signals Required and Adoption. Secure Beacon requires BLE signals, which are extensively adopted in modern consumer electronics, including smartphones, tablets, and a wide array of IoT devices. BLE's widespread presence in consumer products enhances the feasibility of implementing Secure Beacon across various platforms.

Minimum Hardware Capability. The minimum hardware requirement for Secure Beacon includes a BLE module and a processor capable of handling rolling code generation and encryption. Most contemporary mobile devices and IoT hardware are equipped with the necessary BLE capabilities and processing power to support these functions.

Computational/Power Loads. The computational load for Secure Beacon is relatively low, focused primarily on generating and validating rolling codes. The power consumption is also minimal, particularly when using BLE, which is designed for low-energy communication. Peak computational loads occur during the generation and transmission of authentication codes, but these processes are optimized to minimize power usage.

Network and Communication Methods. Secure Beacon relies on BLE for communication, offering a good balance between power efficiency and range. Alternatives like Wi-Fi or UWB could provide extended range or higher accuracy but at the expense of increased power consumption. BLE remains the most practical choice due to its low energy profile and sufficient range for most IoT applications.

Viability of Use. Secure Beacon is highly viable for use in various IoT contexts, especially where seamless and automatic authentication is desirable. Its ease of integration with existing BLE infrastructure and low power requirements make it suitable for consumer electronics, smart homes, and industrial applications.

Major Vulnerabilities. The main vulnerability of Secure Beacon is the reliance on BLE, which can be susceptible to signal jamming and relay attacks. Additionally, if the rolling code generation algorithm is not sufficiently secure, it could be exploited. Ensuring robust encryption and secure code management practices is critical to mitigating these risks.

7.3 T2Pair

Security Level and Mechanisms. T2Pair provides robust security through its touch-to-pair mechanism, which requires physical proximity for authentication. This method inherently prevents remote attacks, as the devices must be in direct contact. While primarily based on physical separation, T2Pair can incorporate cryptographic methods to encrypt the exchanged data during the pairing process, adding an additional security layer.

Signals Required and Adoption. T2Pair relies on NFC signals, which are increasingly common in modern smartphones and other consumer devices. The broad adoption of NFC technology in contactless payments and data transfer supports the feasibility of T2Pair as a secure authentication method.

Minimum Hardware Capability. The minimum hardware requirements for T2Pair include an NFC module and a basic processor capable of handling the data exchange and any associated cryptographic operations. Most contemporary mobile devices and many IoT products come equipped with the necessary NFC capabilities.

Computational/Power Loads. The computational load for T2Pair is minimal, as the touch-to-pair process involves straightforward data exchanges and basic cryptographic operations. The power consumption is also very low, with NFC technology designed for short-range, low-energy communica-

tion. The peak load occurs during the brief pairing process, with negligible average power consumption.

Network and Communication Methods. T2Pair utilizes NFC for communication, offering excellent power efficiency and security within a very short range. While alternatives like BLE or UWB could provide different trade-offs in terms of range and power, NFC remains optimal for touch-based interactions due to its inherent security and low power profile.

Viability of Use. T2Pair is highly viable for secure authentication in scenarios where physical contact is feasible and desirable. Its ease of use, low cost, and compatibility with NFC-enabled devices make it suitable for applications ranging from secure device pairing to contactless payments and access control systems.

Major Vulnerabilities. The primary vulnerability of T2Pair lies in the physical requirement for device contact, which, while secure, may not be practical in all scenarios. Additionally, NFC can be susceptible to eavesdropping and relay attacks if not properly secured. Ensuring robust encryption and secure handling of the NFC communication is essential to mitigating these risks.

References

[1] Satyajit Sinha. State of iot 2023: Number of connected iot devices growing 16% to 16.7 billion globally.

[2] Jiansong Zhang, Zeyu Wang, Zhice Yang, and Qian Zhang. Proximity based iot device authentication. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9, 2017.

[3] Wayne A. Jansen, Serban I. Gavrila, and Vlad Korolev. Proximity-based authentication for mobile devices. In Hamid R. Arabnia, editor, *Proceedings of The 2005 International Conference on Security and Management, SAM 2005, Las Vegas, Nevada, USA, June 20-23, 2005*, pages 398–404. CSREA Press, 2005.

[4] Xiaopeng Li, Qiang Zeng, Lannan Luo, and Tongbo Luo. T2pair: Secure and usable pairing for heterogeneous iot devices. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, CCS '20, page 309323, New York, NY, USA, 2020. Association for Computing Machinery.

[5] Esp32-s3-devkitc-1 datasheet.