Employing Keyed Hash Algorithm, Sequential Probability Ratio Test, and Temperature Comparison Test as Security Against Node Capture Attacks of IoT-Based WSNs

Jhon Aron F. Varca, Earl Nestor T. Velasquez, and Joseph Bryan G. Ibarra
Mapúa University, Manila, Philippines
Email: aronacrav@gmail.com; earlvelasquez007@gmail.com; jbgibarra@mapua.edu.ph

Abstract —The emergence of IoT opened new opportunities for development in various fields. With all the information that it gathers, it became an interesting target for multiple attackers. Thus, this study will enforce security solutions to IoT-based devices specifically in the perception layer by incorporating a Temperature Comparison Test, Keyed Hash Algorithm and evaluating it using SPRT especially in the defense against malicious activities detected in the nodes of a network namely for Mobile and Immobile attacks. For immobile attacks, using the keyed hash algorithm and the SPRT, the hash key of the passcodes was compared to determine the safety of the nodes. Hence, from the functionality test that was conducted, and evaluating the data gathered using SPRT and Bernoulli's equation, the reliability of the protocol to detect Immobile attacks is concluded to have a 100% detection rate. For mobile node attacks, the study assumes the environment to be under normal, warm, and cold room temperatures. where both mobile and without mobile attack is simulated, the result shows that there is only an overall 3% difference from the temperature measure by the sensor to the ambient temperature. Hence, combining these protocols that are applied in the study eliminates the single points of failure in the nodes that are either applicable only to a distributed scheme or mobility support, the study also compared the tested protocol to the other existing protocols.

Index Terms—Internet of Things (IoT), Sequential Probability Ratio Test (SPRT), Keyed Hash Algorithm, Temperature Comparison Test

I. Introduction

The emergence of the internet has become an integral part in the lives of many people. It continues to expand its horizons at a considerably fast pace. One of the concepts born from the internet, is the Internet of Things (IoT) or some call it the Internet of Objects. Technically speaking, IoT is referred to as the interconnection of an object to a network to other connected devices [1]. Nowadays, billions of devices around the world are connected to the internet and are simultaneously exchanging, collecting, and sharing data. Due to the rapid advances especially in the field of communication and technology, the continuous development of the IoT opens tremendous opportunities for a wider range of applications. This is one of the most popular high-tech technologies that are in-demand right now. According to Lee and Fumagalli, it is expected that by the end of 2020, the number of IoT connected devices will be approximately 20 billion [2]. Even so, despite its continuous evolution, especially with it being applied in the industry like healthcare, business, agriculture, and companies continuously introducing countless IoT-based devices and products [3], these devices are known to have many vulnerabilities which makes them very susceptible to cyber threats. In fact, these vulnerabilities include the lack of privacy and protection of the personal data that are being collected by the IoT systems [4]. With that said, along with its explosive growth, the number of security issues, possible threats and attacks against the device or the individual is increasing drastically as well.

Thus, with all the information that it gathers, it became an interesting target for multiple attackers like the hackers, cybercriminals, and a lot more. Device tampering can be done by hacking the IoT devices, it has negative impacts such as unexpected changes in the functionality, security breaches, safety risk to humans, or ruined devices. Without a trusted IoT ecosystem, the growth of IoT applications may not reach a high demand and lose all its potential. In addition to this, the sources of security threats are found in the four different layers of an IoT application which are the sensing or perception layer, network layer, middleware, and application layer [5]. These layers utilize different types of technologies which are subject to their own security issues and threats. Perception layers are known to be more prone to attacks since these are used in IoT applications such as GPS, WSNs, RFID where they collect data from the sensors and use it to control the physical component of the device. Potential attackers can try to compromise this IoT layer by attacking the nodes to launch attacks against a third-party entity [6]. This type of attack is known as Node Capturing. Basically, when a node is captured, it removes the sensor node to compromise the network and redeploys them to perform various attacks [7]. By doing so, the sensor in the device can be tampered and may result in false and inaccurate readings that would lose its functionality [8].

Manuscript received April 17, 2021; revised October 15, 2021.
Corresponding author email: aronacrav@gmail.com
doi:10.12720/jcm.16.11.500-507

With that said, the recommendations of the previous related research led to the consideration of employing security solutions to protect the device. Although a lot of research addresses the various security challenges that an IoT device experiences, the security needs for the attacks against security or privacy of the perception layer in IoT devices are not yet well-recognized. Furthermore, most of the existing mechanisms and solutions are not equipped to detect the node capture attack immediately. Thus, to have a stronger system resiliency in the sensing layer of the system, further research needs to be conducted to develop a protocol to immediately detect a node capture attack.

Hence, in this paper, the deployment and enforcement of security solutions to IoT-based devices specifically in the perception layer will be thoroughly investigated. A protocol will be designed to implement an enhanced security by incorporating a Temperature Comparison Test, Keyed Hash Algorithm and evaluating it using the Sequential Probability Ratio Test especially in the defense against malicious activities detected in the nodes of a network namely for Mobile and Immobile attacks. This study specifically seeks: (a) to eliminate the single points of failure in the immobile nodes by providing a protocol that incorporates a high availability system, which will prevent it from various attacks; (b) to design a security circuit by using sensors and actuators wherein the parameters will measure and process the collected data if any changes or interruption occurs in the system; (c) and to evaluate and test the effectiveness of the security circuit by comparing the protocol to other existing protocols and by developing a graphical user interface to provide accurate data against malicious attacks to the sensor nodes.

The realization of the importance of enforcing security mechanisms in IoT-based systems will help in mitigating IoT risks, which are the most challenging problems that the communicating systems are experiencing today. Also, by improving the security in the nodes, it will help in easing up the delivery, accessing, exchanging and authorizing of data which will protect the privacy of both the individual who owns the device and the system itself. Moreover, by doing so, it can open a lot of opportunities for engineers and researchers to build an architectural circuit or conduct further research which can overcome other security issues in the different layers that the IoT experiences.

The focus of this study is to deploy and enforce security solutions to IoT-based devices, specifically a wireless sensor network. Moreover, the design will implement enhanced security in IoT systems especially in the defense against malicious activities detected in the nodes of a network. The study will only focus on the security issue experience at the sensing layer, a major security threat encountered namely Node Capturing. The study will also cover mitigating attacks on mobile and immobile nodes. The study's testing will only be done in a controlled temperature environment and will not be deployed outside to minimize the extraneous variables experienced. The study will not cover the other types of security threats in IoT applications such as the security issues at network layer, middleware layer, gateways, and application layer.

II. Review of Related Literature

A. *Internet of Things: Security, Node Capture Attack*

The internet has grown to be the backbone of virtual communication worldwide. It has become a platform for the people to communicate with others by establishing a connection between their computer and any other computer that is connected globally. One of its technologies that is gaining a huge popularity because of its rapid advancement is the Internet-of-Things. Hence, to have effective communication and to avoid security breaches in IoT systems, a lightweight, secure, and reliable IoT protocol should be used so that it will not compromise the computational ability and efficacy of the device. Furthermore, the IoT communication protocols should protect and ensure the optimum security of the data being exchanged between the connected devices [9].

Security issues are still a huge problem for IoT devices. This information can be easily accessed through major data breaches which the consumers are wary of as their personal data can be known by the hackers [10]. There are different types of attacks affecting the IoT system. In a study by [11], four main categories of attack were presented namely physical, software, network, and encryption.

As stated, one of the vulnerabilities in the perception layer is the Node Capture Attacks. Node capture attacks occur to devices that are in hostile environments. Basically, what happens is that the attacker attacks a legitimate sensor node and physically captures, reprograms, and redeploys the compromised node back into the network [12]. The compromised node then steals private information such as cryptographic keys, unique ID information and a lot more, which ultimately results in the entire network being overtaken [13]. A node capture attack can be classified into a mobile node capture attack and immobile nod capture attacks. In a mobile capture attack, the attacker is continuously moving or in a moving vehicle while compromising the network and in an immobile attack, the attacker gets the node. Thus, further research is needed to be conducted to develop a protocol to immediately detect a node capture attack for a stronger system resiliency in the sensing layer of the system. To make the hardware implementation of the protocol, sensors and actuators will be utilized.

B. *Standard Temperature of Room and Water*

The water and room temperature plays a crucial part for the determination of change by the sensor. Hence, the standard water temperature for hot water is 54.4°C and above. For lukewarm water, it is 32.3 degrees to 43.3°C and for cold water it is 15°C and below [14]. On the other hand, the room temperature is almost equivalent to the water temperature and is as follows. For a normal room temperature, it is 20°C-30°C, for a warm room temperature it is 31°C-40°Cand 8°C-15°C for a cold room temperature [15].

C. Evaluation of Existing Solutions in Node Capture Attacks

Since node capturing is one of the most challenging security threats that researchers have encountered. Several schemes have already been proposed to provide security against node capture attacks. Different schemes have different goals to achieve in the resilience for node capture. One of these proposed solutions is the Sensor Node Capture Attack Detection and Defense (SCADD) protocol that provides both detection and defense against node capture attacks by creating security strategies. In their simulation results, the SCADD carefully misdirects the attack by replacing the important information inside the memory with fake data. Moreover, in case the SCADD misjudges the attack, it can also restore the nodes [16]. In another study, Trusted Platform Module enabled Program Integrity Verification (TPIV) protocol was introduced to detect node capture attacks in a wireless sensor network. In the proposed protocol, it relies on the strength of the cryptographic hash function and can secure efficiently by detecting the node capture attack. In case of an adversary, it can put additional memory in the node when captured. The TPIV also ensures that only the authorized verifier can execute the commands [17]. In addition to this, Secure Decentralized Data Transfer was also proposed to improve resilience against node captures attacks in a wireless sensor network. In this proposal, the Secret Sharing Scheme (SSS) was utilized to disperse confidential information without needing a secret key. Common security methods are based on public key cryptosystems, but it could cause a major problem with the encryption of data. After the simulation of the proposal, it was confirmed that using SSS was more effective than the common security method, TinySec [18]. A Lightweight White-box Symmetric Encryption Algorithm (SMS4) can also be applied for the node capture attack security in wireless sensor networks. The main idea of this proposal is to merge several steps of round function of the SMS4 into table lookups that are blended randomly by generated mixing bijections. It is a good countermeasure against key compromise in a node capture attack [19]. Another study was made to improve resilience in node capture attack by using ICmetrics. In ICmetric technology, it computes the metrics based on the hardware and software of the properties of the sensor node; in this method, it does not require a stored private key for it to operate. Any change that occurs in the hardware and software of the node generates a different ICmetric associated with that node to stop the attack entering the network [20]. A study by Mishra and Turuk [21], proposed a key renewal model in a wireless sensor network using age replacement policy to determine the expected time interval of two successive key renewals to mitigate the node capture attack in the network. Common attacks steal mostly all the key information from the captured node. Renewing the keys can be done through an authenticated channel in a periodic manner. With that said, all the studies only did simulations and have not included hardware implementation for the different protocol which is needed to highlight the strengths and weaknesses of the protocol in the physical environment. Furthermore, the existing problem with the proposed mechanisms and solutions is that it does not detect the node capture attack immediately. Thus, further research is needed to be conducted to develop a protocol to immediately detect a node capture attack for a stronger system resiliency in the sensing layer of the system.

D. Other IoT Related Research

Presently, IoT-related research in the technology and engineering fields is also being intensively investigated. A research by [22] developed a phasor measurement unit and incorporate it into IoT technology. The researchers used the Proteus software simulation tool to design an electronic circuit, then built a prototype and integrated it with IoT. The architecture functions by calculating the phase difference between two sinusoidal waves. Another research that incorporated the concept of IoT technology is by [23] whose concept of detection is similar to this paper except that they designed a device which not only monitors the various aspects of indoor air quality (IAQ), but also sends an alert to the user about potential threats in the given area to notify the user, the control system employs wireless data transmission and a mobile application. The sensor system also used machine learning with Support Vector Machines, which assisted the system in attempting to forecast relevant data as accurately as possible.

III. Methodology

A. Conceptual Framework

The input of the system will be the temperature measured from each sensor as well as the passcode assigned and the generated keyed hash function of it. The system process is divided into two where the first process is for the mobile node attack which is the comparison of the measured temperature of the sensor to the set temperature. While the second process is by using the Sequential Probability Ratio Test which verifies the integrity of the node by comparing the first hash function generated to the second hash function. This will increase the resilience and protection of the node from various malicious attacks such as mobile and immobile node attacks. Lastly, the sensor's red LED will turn on if the measured temperature is out of range from the set temperature and otherwise if it does fall within the range. For the SPRT, the null hypothesis is that the two functions are Asymmetric or not the same and the Alternative Hypothesis is that the two hash keys are

Symmetric. The output for both these processes are then displayed on the graphical user interface (see Fig. 1).

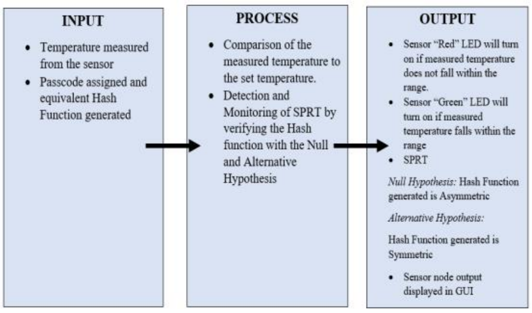

INPUT
Temperature measured from the sensor
Passcode assigned and equivalent Hash Function generated
PROCESS
Comparison of the measured temperature to the set temperature.
Detection and Monitoring of SPRT by verifying the Hash function with the Null and Alternative Hypothesis
OUTPUT
Sensor "Red" LED will turn on if measured temperature does not fall within the range.
Sensor "Green" LED will turn on if measured temperature falls within the range
SPRT
Null Hypothesis: Hash Function generated is Asymmetric
Alternative Hypothesis:
Hash Function generated is Symmetric
Sensor node output displayed in GUI

Fig. 1. Conceptual framework

B. *Overall System Process Flow*

The system will start with the setting of the temperature range for each sensor accordingly. This will serve as the basis for the conditions of whether the green or red LED will turn on. The sensor will then measure the temperature of the room and compare it to the set temperature. If the measured temperature is within the range of the set temperature, the green LED will turn on, otherwise the red LED turns on (see Fig. 2 and Fig. 3).

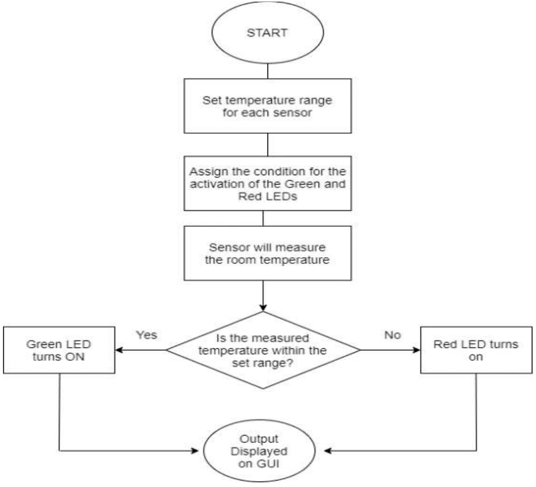

START
Set temperature range for each sensor
Assign the condition for the activation of the Green and Red LEDs
Sensor will measure the room temperature
Is the measured temperature within the set range?
Yes
No
Green LED turns ON
Red LED turns on
Output Displayed on GUI

Fig. 2. Mobile node system process flow diagram

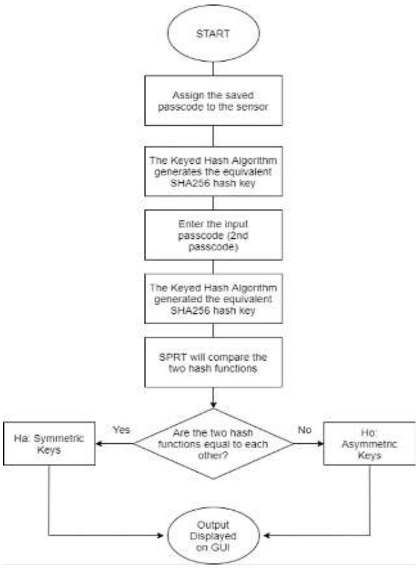

START
Assign the saved passcode to the sensor
The Keyed Hash Algorithm generates the equivalent SHA256 hash key
Enter the input passcode (2nd passcode)
The Keyed Hash Algorithm generated the equivalent SHA256 hash key
SPRT will compare the two hash functions
Are the two hash functions equal to each other?
Yes
No
Ha: Symmetric Keys
Ho: Asymmetric Keys
Output Displayed on GUI

Fig. 3. Immobile Node System Process Flow Diagram

In addition to this, the system will also enter the assignment of the saved password for the sensor. This will serve as the private passcode and is not allowed to be changed by the user. After which, it will enter the generation of its equivalent SHA256 hash function by the Keyed Hash Algorithm. Then, the user will enter the second passcode which will determine the SPRT of the sensor. The Sequential Probability Ratio Test is done by setting a condition for the Null and Alternative hypothesis. In this case, the null hypothesis is that the first and second hash keys are not equal or "Asymmetric" and the Alternative Hypothesis is that the first and second hash keys are equivalent to each other or "Symmetric". Both the output for temperature and keyed hash algorithm are displayed on the graphical user interface.

C. *Security Protocol for Immobile Attacks*

For the testing of the security circuit of Immobile nodes, to assume that an Immobile node attack took place, the group altered the Second or "Input Passcode" to determine if the system will discover the change. It is important to note that in the code *pin2* is the only one that can be altered since *pin1* is fixed and is the saved password for the sensor. Hence, it is expected that the serial monitor will show that the two hash keys are Asymmetric.

1) *Data gathering for immobile attack*

The first attack, immobile node capture attack, was a kind of attack where an attacker extracts the cryptographic keying material and modifies the code for the node to behave maliciously. In the simulation, the hacker input different passcodes to the sensor. Using the hash key algorithm and the SPRT, the hash key of the passcodes was compared and determined the safety of the nodes. The SPRT will be used to verify the accuracy of the system to detect immobile attacks. There was a total of 30 trials done separated into 3 sessions where different combinations of password are done to test the reliability, accuracy, and repeatability of the system. For this testing, we assumed two instances for the hypothesis testing as shown below.

Ho: Hash Function generated is Asymmetric
Ha: Hash Function generated is Symmetric

Table I shows the output when there is no immobile attack or with immobile attack to all the nodes. In the first five trial of each session, trial 1-5, using the passcode REALCODE as the shared and input, the hash key algorithm generated the same hash key and when it was tested with the SPRT, the output was symmetric thus the remark for the node was safe which follows our alternative hypothesis. The average detection speed for the SPRT was 5.9 seconds.

For the second five trial of each session, namely trial 5-10, the input passcode is different with the shared passcode for each node. Hash key algorithm was used to generate the hash key of the input and when the SPRT test started, the output shown was asymmetric meaning that the node was unsafe following our null hypothesis. As mentioned above, the testing was done consecutively for 30 trials for the immobile attack set-up, this will serve as the *i* of the system for the equation. The success probability of the system in detecting change in the hash key of the node is represented as **λ.** While the registered passcode is represented as λ′, then denote *Si* as,

$$Si = \begin{cases} 0, if\ no\ alterations \\ 1, if\ with\ alterations \end{cases}$$

From this, the equation presented is:

$$\Pr(Si = 1) = 1 - \Pr(Si = 0) = \lambda \qquad (1)$$

If λ = λ′, it is assumed that the node is not attacked. However, if λ ≠ λ′, it is most likely that the node is attacked. From there the null and alternative hypothesis will be reformulated as: *Ho*: λ ≠ λ′ and *Ha:* λ = λ′. With that said, out of the 30 trials conducted, when there is no attack done to the system, the null hypothesis is rejected. On the other hand, when there were various instances of attack assumed, the detection rate of the system in recognizing alterations in the input hash key is concluded to be 100%, proving the reliability of the protocol.

TABLE I: NODES WITH AND WITHOUT IMMOBILE ATTACK

Trial	**Passcodes (Shared, Input)**	**HASHKEY GENERATED**	**SPRT**	**Remark**	**Speed of Detection**
1	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9	SYMMETRIC	SAFE	5.91s
	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9			
2	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9	SYMMETRIC	SAFE	5.92s
	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9			
3	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9	SYMMETRIC	SAFE	5.88s
	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9			
4	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9	SYMMETRIC	SAFE	5.89s
	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9			
5	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9	SYMMETRIC	SAFE	5.89s
	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9			
6	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9	ASYMMETRIC	UNSAFE	5.89s
	REALCOD	88a0b464c33b5e9af8de8423f1f044cd7eb48baa5778969f8af6d74bef273e06			
7	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9	ASYMMETRIC	UNSAFE	5.9s
	REALCOD	88a0b464c33b5e9af8de8423f1f044cd7eb48baa5778969f8af6d74bef273e06			
8	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9	ASYMMETRIC	UNSAFE	5.92s
	REALCO	cdaa5274bc19a4e343322ad26fac3e4eee7d1a84a4d14740c059a72b58e2c7c7			
9	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9	ASYMMETRIC	UNSAFE	5.9s
	REALCO	cdaa5274bc19a4e343322ad26fac3e4eee7d1a84a4d14740c059a72b58e2c7c7			
10	REALCODE	1909b47040a10ba2a507bf932416b6bd99473854849b0262e442cdc193b326d9	ASYMMETRIC	UNSAFE	5.9s
	realcode	1fd5cad001538da54700d2cb7c16ff8d5c29a794066ab14d926fa71bcfa97bd2			

D. Security Circuit for Mobile Nodes

1) Experimental setup

The experimental setup shown in Fig. 4 will be the main topology where the research and testing will be conducted. The topology will be mainly composed of the user's laptop, the server (internet), temperature sensor (DS18B20), Arduino Uno (represented as MCU-PT), and the LED which will serve as the indicator of the output. The setup will begin by the user accessing the router in which the sensors are connected. Next, the data gathered from the sensors will be used in conducting the Sequential Probability Ratio Test. This scheme will allow the system to detect the node capture attack more efficiently. The different setups in which the topology will be conducted includes the testing of immobile node capture attacks and mobile node capture attacks. Each of the nodes work independently and are all connected through one router. Monitoring can be done through the GUI in the mobile device and can be monitored in another device as long as that device is connected to the internet, the nodes can be monitored in real time. It can be also monitored through the server where each sensor will log the data to Google Spreadsheet.

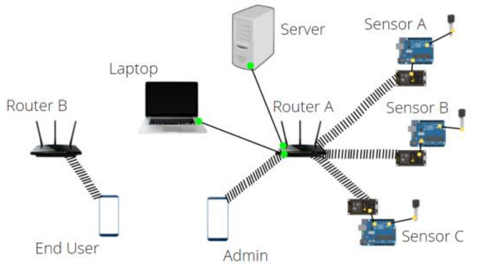
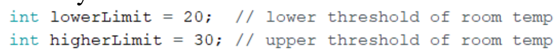


Fig. 4. Experimental Set Up

2) Testing of security circuit for mobile nodes

The testing of the security circuit against the node capture attacks will be done by analyzing the change in the LED output of the sensors without node capture attack and with a node capture attack. The threshold set for the hot, cold, and normal room setup is based on [15]. The upper and lower threshold can be changed by the user accordingly to fit the environment where the sensor will be deployed. This threshold will determine the LED output. Fig. 5 shows the part of the code where the threshold can be changed. It is advisable for the user to first measure the ambient temperature of the environment where the sensor will be deployed so as to maintain the accuracy of the data.



```
int lowerLimit = 20;  // lower threshold of room temp
int higherLimit = 30; // upper threshold of room temp
```

Fig. 5. Temperature threshold code

When the LED is green, it means that the sensor is still safe and no attack was done and when the LED is red, it means that a node capture attack was detected. Individual testing for each sensor will be conducted to improve the accuracy of data. To minimize the extraneous variables experienced during the testing, the group decided to deploy the circuit in a controlled environment (see Fig. 6).

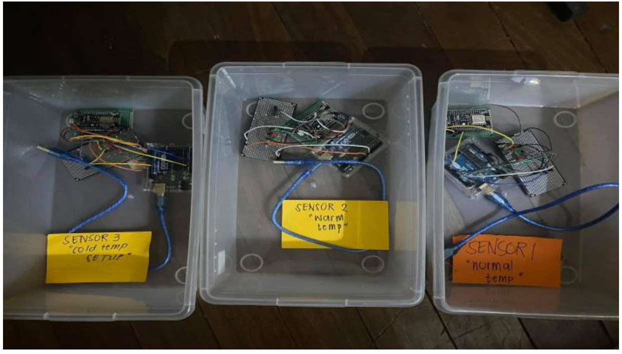


Fig. 6. Three sensors in a controlled environment

For the mobile node attacks, three (3) scenarios are considered which are: (1) hot room temperature, (2) normal room temperature, and (3) cold room temperature.

3) Data gathering - temperature comparison test

For the mobile node capture attack, it was a kind of attack where with a modest manpower, the attacker could periodically visit each node, pick it up, and move it. It was simulated that the attacker placed the node in another location. The testing was done for a week to observe if the sensor's measured temperature will still remain accurate. The percentage difference between the ambient temperature and sensor temperature were also calculated to test the accuracy of the sensor reading using percentage difference.

An SPRT for the probability of success of the system in detecting the temperature change were used where *i* is the number of trials, then *Si* is denoted as,

$$Si = \begin{cases} 0, if\ led\ is\ green \\ 1, if\ led\ is\ red \end{cases}$$

Then, the equation for the SPRT can be presented again,

$$\Pr(Si = 1) = 1 - \Pr(Si = 0) = \lambda \qquad (2)$$

λ is represented as the change of the temperature in the sensor reading and λ' as the set temperature. When $\lambda = \lambda'$, it is assumed that the node is not attacked with the green LED lighting up, but if $\lambda \neq \lambda'$, then the node is likely to be under mobile attack with the red LED that lights up. The null and alternative hypothesis formulated with the given condition: *Ho*: $\lambda \neq \lambda'$ and *Ha:* $\lambda = \lambda'$.

Sensor A, B, and C all have different set temperature in each sensor. The sensor temperature reading was compared to the ambient temperature to test the accuracy of the sensor reading in different settings. There is only 3% difference in the temperature between the ambient and sensor proving the accuracy of the sensor to detect an attack. In the testing, all nodes were able to detect the mobile attack attempts in 30 trials proving the effectiveness and reliability of the protocol with 100% detection rate.

E. Evaluation

In Table II the security protocols applied in the study were compared with the protocols in the related studies. Keyed hash algorithm has a distributed scheme as used in the immobile attack security but cannot be used for the mobile attack. With this, the Temperature Comparison Test was used for security against mobile attacks in the study. SPRT was also used to support both of the protocols used in the system. The Sensor Node Capture Attack Detection and Defense only communicates between the sensor nodes. The Trusted Platform Module enabled Program Integrity Verification relies on the strength of the cryptographic hash function and can secure efficiently by detecting the node capture attack. The Secure Decentralized Data Transfer was used for mobile wireless sensor networks but no application when there is an immobile attack. Compared to the Lightweight White-box Symmetric Encryption Algorithm, it's used for security against immobile attacks by communicating to each router but has no application against mobile attacks. Another example from previous protocols are ICmetrics and Key Renewal Model but the two protocols is only applicable against immobile attacks but has no defense against mobile node attacks.

TABLE II: NODE A WITH AND WITHOUT MOBILE ATTACK

Trial	Set Temperature (°C)	Ambient Temperature (°C)	Sensor Temperature (°C)	Percentage Difference (%)	LED	Remark
1	20-30	25.1	24.31	3.20	GREEN	SAFE
2	20-30	25.1	24.35	3.03	GREEN	SAFE

3	20-30	25.1	24.44	2.66	GREEN	SAFE
4	20-30	25.1	24.5	2.42	GREEN	SAFE
5	20-30	25.1	24.56	2.17	GREEN	SAFE
6	20-30	30.8	30.62	0.59	RED	UNSAFE
7	20-30	31.5	31.12	1.21	RED	UNSAFE
8	20-30	32.7	32.06	1.98	RED	UNSAFE
9	20-30	32.7	32.13	1.76	RED	UNSAFE
10	20-30	33.3	32.19	3.39	RED	UNSAFE

TABLE III: COMPARISON OF THE APPLIED PROTOCOLS AND PREVIOUS PROTOCOLS

	Security Methods	**Immobile Attack**	**Mobile Attack**
Applied Protocols	Keyed Hash Algorithm	Yes	No
	Temperature Comparison Test	No	Yes
	Sequenced Probability Ratio Test	Yes	Yes
Previous Protocols	Sensor Node Capture Attack Detection and Defense	No	No
	Trusted Platform Module enabled Program Integrity Verification	No	Yes
	Secure Decentralized Data Transfer	No	Yes
	Lightweight White-box Symmetric Encryption Algorithm	Yes	No
	ICmetrics	Yes	No
	Key Renewal Model	Yes	No

IV. CONCLUSION

IoT-based Wireless Sensor Networks are prone to node capture attacks due to the information it gathers. Node capture attacks are a threat to WSNs with different security issues, possible threats, and attacks against the device. The Keyed Hash Algorithm and SPRT covers the security for the immobile attacks. The generated pair-wise key is compared with the generated input hash using the SPRT. The key safeguard the node from compromising the WSNs. Hence, from the functionality test that was conducted, and the data gathered in Table III shows the reliability of the protocol to detect Immobile attacks and is concluded to have 100% detection rate. The Temperature Comparison Test protects the node from mobile attacks using sensors and actuators that measure and process the collected data if any changes or interruption occurs in the system. Mobile attacks are immediately detected when the node is moved from its controlled environment and the measured temperature is not within the set temperature in the node. With that said as shown in Table III, where both mobile and without mobile attack is simulated, the result shows that there is only an overall 3% difference from the temperature measure by the sensor to the ambient temperature. Hence, combining these protocols that are applied in the study eliminate the single points of failure in the nodes that are either applicable only to a distributed scheme or mobility support, the researchers compared the tested protocol to the other existing protocols. This makes a difference when it comes to ensuring the safety of the network since it detects both mobile and immobile attacks. In addition to this, the state of the nodes can be monitored anywhere with the GUI application. Therefore, using the applied protocols, the resilience against node capture attack is strengthened and the overall security of the network is enhanced.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

All the authors contributed in the research; Jhon Aron F. Varca created the codes for the key hash algorithm and wrote the paper; Earl Nestor T. Velasquez build the prototype and made the codes for the temperature comparison test and also wrote the paper; Joseph Bryan G. Ibarra checked and analyzed the data and provided guidance until the completion of the research; all authors had approved the final version.

REFERENCES

[1] F. Xia, L. T. Yang, L. Wang, and A. Vinel, "Internet of things," *Int. J. Commun. Syst.*, vol. 25, no. 1101–1102, 2012.

[2] C. Lee and A. Fumagalli, "Internet of things security-multilayered method for end to end data communications over cellular networks," in *Proc. IEEE 5th World Forum Internet Things, WF-IoT 2019 - Conf. Proc.*, 2019, pp. 24–28.

[3] P. Yadav and S. Vishwakarma, "Application of internet of things and big data towards a Smart City," in *Proc. - 2018 3rd Int. Conf. Internet Things Smart Innov. Usages, IoT-SIU 2018*, 2018, pp. 1–5.

[4] M. Frustaci, P. Pace, G. Aloi, and G. Fortino, "Evaluating critical security issues of the IoT world: Present and future challenges," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2483–2495, 2018.

[5] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A survey on IoT security: Application areas, security threats, and solution architectures," *IEEE Access*, vol. 7, pp. 82721–82743, 2019.

[6] A. Mosenia and N. K. Jha, "A comprehensive study of security of internet-of-things," *IEEE Trans. Emerg. Top. Comput.*, vol. 5, no. 4, pp. 586–602, 2017.

[7] M. V. Bharathi, R. C. Tanguturi, C. Jayakumar, and K. Selvamani, "Node capture attack in wireless sensor network: A survey," in *Proc. IEEE Int. Conf. Comput. Intell. Comput. Res. ICCIC 2012*, no. i, pp. 20–22, 2012.

[8] K. S. Kumar, S. Sahoo, A. Mahapatra, A. K. Swain, and K. K. Mahapatra, "Security enhancements to system on chip devices for IoT perception layer," in *Proc.- 2017 IEEE Int. Symp. Nanoelectron. Inf. Syst. iNIS 2017*, vol. 2018-Febru, 2018, pp. 151–156.

[9] C. Sharma and N. K. Gondhi, "Communication protocol stack for constrained IoT systems," in *Proc. 3rd International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)*, 2018, pp. 1–6.

[10] Y. Yang, L. Wu, G. Yin, L. Li, and H. Zhao, "A survey on security and privacy issues in internet-of-things," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1250–1258, 2017.

[11] I. Andrea, C. Chrysostomou, and G. Hadjichristofi, "Internet of things: Security vulnerabilities and challenges," in *Proc. IEEE Symposium on Computers and Communications*, vol. 2016-Febru, 2016, pp. 180–187.

[12] S. Agrawal, M. L. Das, and J. Lopez, "Detection of node capture attack in wireless sensor networks," *IEEE Syst. J.*, vol. 13, no. 1, pp. 238–247, 2019.

[13] B. Butani, P. Kumar Shukla, and S. Silakari, "An exhaustive survey on physical node capture attack in WSN," *Int. J. Comput. Appl.*, vol. 95, no. 3, pp. 32–39, 2014.

[14] Check Washing Machine Water Temperatures for Better. HANDYMAN. [Online]. Available: https://www.familyhandyman.com/project/check-washing-machine-water-temperatures-for-better-performance/

[15] What are the regulatory Definitions for 'Ambient', 'Room'.. [Online]. Available: https://www.gmp-compliance.org/gmp-news/what-are-the-regulatory-definitions-for-ambient-room-temperature-and-cold-chain

[16] S. H. Jokhio, I. A. Jokhio, and A. H. Kemp, "Node capture attack detection and defence in wireless sensor networks," IET Wirel. Sens. Syst., vol. 2, no. 3, pp. 161–169, 2012.

[17] S. Agrawal, M. L. Das, and J. Lopez, "Detection of node capture attack in wireless sensor networks," *IEEE Syst. J.*, vol. 13, no. 1, pp. 238–247, 2019.

[18] E. Kohno, T. Ohta, and Y. Kakuda, "Secure decentralized data transfer against node capture attacks for wireless sensor networks," in *Proc. International Symposium on Autonomous Decentralized Systems*, 2009, pp. 1–6.

[19] Y. Shi, W. Wei, and Z. He, "A lightweight white-box symmetric encryption algorithm against node capture for WSNs," *Sensors (Switzerland)*, vol. 15, no. 5, pp. 11928–11952, 2015.

[20] R. Tahir and K. McDonald-Maier, "Improving resilience against node capture attacks in wireless sensor networks using ICmetrics," in *Proc. Third International Conference on Emerging Security Technologies,* 2012, pp. 127–130.

[21] A. K. Mishra and A. K. Turuk, "A key renewal model for wireless sensor network under node capture attack," 2011.

[22] J. C. D. C. Gallano, V. J. D. Malvas, J. L. F. Quirona, R. C. S. Soriano, M. C. Pacis, and F. R. G. Cruz, "Design and implementation of phasor measurement unit with IoT technology," in *Proc. IEEE 12th Int. Conf. Humanoid, Nanotechnology, Inf. Technol. Commun. Control. Environ. Manag*. 2020.

[23] P. R. Meris, *et al.*, "IOT Based - automated indoor air quality and LPG leak detection control system using support vector machine," in *Proc. 11th IEEE Control Syst. Grad. Res. Colloquium*, 2020, pp. 231–235.

Jhon Aron F. Varca was born in Manila, Philippines, in 2000. He is currently studying B.S. degree Electronics Engineer in Mapúa University. He specialized in Advanced Internet Protocol Networking. His research interests include wireless networks, network security, smart city, and Internet of Things.

Earl Nestor T. Velasquez was born in Antipolo City, Philippines in 1999. He is taking the B.S Electronics Engineering from Mapúa University. His research interests include network security, smart city, and Internet of Things.

Joseph Bryan G. Ibarra is currently a professor in the School of Electrical, Electronics and Computer Engineering in Mapúa University. He received his B.S. degree and M.S. degree both in Electronics Engineering from Mapúa University. His research interests include electronics, communication systems, wireless networks, and Internet of Things.