IoT-based Base Transceiver Station (BTS) Security System on Telkomsel Tower (Case Study of Telkomsel Site in Malang Area)

Mochammad Junus[1], **Hadiwiyatno Hadiwiyatno**[2], **Farel Rizky Oktavianto**[3]

[1,3] Digital Telecommunication Network Study Program, Department of Electrical Engineering, State Polytechnic of Malang, 65141, Indonesia.

[2] Telecommunication Engineering Study Program, Department of Electrical Engineering, State Polytechnic of Malang, 65141, Indonesia.

[1]mochammad.junus@polinema.ac.id, [2]hadiwiyatno@polinema.ac.id, [3]rizkyfarel131@gmail.com

Abstract— **At BTS sites, a lot of expensive telecommunications equipment is stored without adequate security. One of the telecommunication rooms that require a high level of security is the BTS (Base Station Transceiver) shelter. Nowadays crime has become part of everyday life and is difficult to separate, criminals have the opportunity to commit crimes anywhere. This research resulted in the design and manufacture of a tool that operates in the BTS battery compartment which is used to check the battery condition, how the system works, how the components work, test and analyze the system to draw a conclusion. Based on the results and testing of the IoT-based Base Transceiver Station (BTS) Security System above, several conclusions can be drawn, among others: PIR sensor test results were obtained with a delay of 5.76 seconds when sending notifications to Telegram Bot. Then in testing with RFID cards can be read properly at a distance of 1 to 5 cm by the RFID reader and when the registered card is affixed then the door lock solenoid will open, the maximum distance from the RFID reader is 5 cm. From the test results of the Infrared (PIR) sensor, the average delay is 5.76 seconds for daytime conditions with an average lux meter value of 16.2 and an average delay of 6.8 seconds at night with an average lux meter value of 0.6, this is a pretty good number because the Infrared (PIR) sensor is functioning properly.**

***Keywords*— BTS Security System, RFID Access Control, Battery Monitoring System, ESP32, Internet of Things.**

I. INTRODUCTION

Base Transceiver Stations (BTS) are critical nodes in cellular networks, facilitating wireless communication between user equipment and the core network infrastructure [1]. These sites house expensive and vital equipment, including transceivers and, crucially, backup battery banks essential for maintaining operations during power outages [2], [3]. However, the often remote and dispersed nature of BTS sites makes them vulnerable to theft and vandalism. Incidents such as the theft of 24 Telkomsel BTS battery units, resulting in a financial loss of approximately IDR 96 million, underscore the severity of this persistent security challenge [4]. Such breaches not only cause significant economic damage but also risk degrading network quality and service reliability for end-users, directly impacting Quality of Service (QoS) parameters like service availability [5].

Existing security measures often rely on conventional physical locks and periodic manual checks, which are reactive and insufficient for real-time threat response [6]. Prior research has explored technological solutions, including security systems that send SMS alerts using microcontrollers [7] and sensor networks for general BTS condition monitoring [8]. More recent developments involve using single-board computers like the Raspberry Pi with infrared sensors for intrusion detection, notifying personnel via messaging applications [9]. While progressive, many systems lack a tightly integrated approach that combines reliable intrusion detection with robust, credential-based access control, leaving a critical gap in securing the most vulnerable assets like battery compartments [10], [11].

The Internet of Things (IoT) paradigm offers a powerful framework for developing intelligent, connected, and automated security systems [12]. IoT architectures enable seamless data communication between sensors, controllers, and actuators over the internet, allowing for real-time remote monitoring and control [13], [14]. Key enabling technologies include the ESP32 microcontroller, renowned for its integrated Wi-Fi/Bluetooth capabilities, low power consumption, and sufficient processing power for embedded security applications [15], [16]. For detection, Passive Infrared (PIR) sensors provide an efficient and low-power method for identifying unauthorized human movement within a defined area by detecting changes in infrared radiation [17]. For access control, Radio Frequency Identification (RFID) technology offers a more secure and manageable alternative to traditional keys, allowing for digital authentication, audit trails, and integration with solenoid door locks [18], [19]. Notifications can be effectively pushed to security personnel in real-time using popular platforms like Telegram Bot, which offers a reliable API for IoT applications [20].

To address the identified security gap, this research proposes the design and implementation of an integrated IoT-based security system for BTS battery compartments. The proposed system synthesizes multiple technologies: PIR sensors for motion-based intrusion detection and an RFID-based locking mechanism for enforcing strict access control, with the ESP32 serving as the central processing unit [15], [18]. The system is designed to provide a proactive security response by instantly sending alert notifications via a Telegram Bot upon detecting any unauthorized access attempt or intrusion [9], [20]. By integrating preventive access control with real-time intrusion detection and remote alerts, this system aims to enhance the physical security of critical telecommunications infrastructure, thereby reducing financial losses and ensuring service continuity [5].

II. METHOD

To design an IoT-based Base Transceiver Station (BTS) Security System for the Telkomsel Tower, the tools and materials have been determined after reading various references. So, the design of this tool was carried out to complement several features that did not exist in innovations that had existed before.

A. Block Diagram

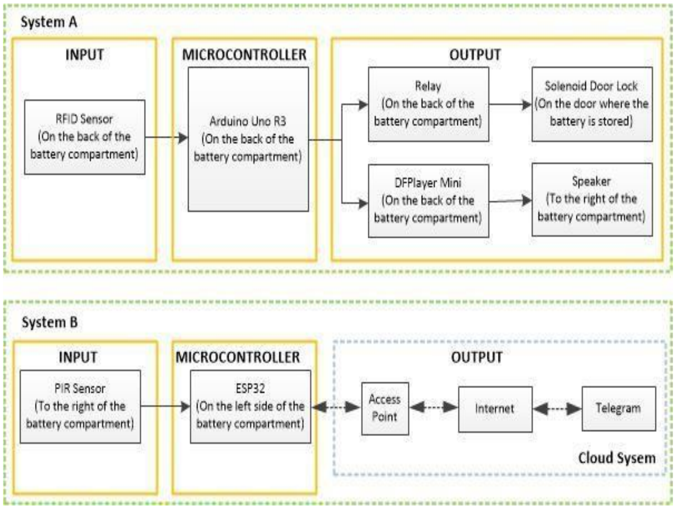

System A
INPUT
RFID Sensor (On the back of the battery compartment)
MICROCONTROLLER
Arduino Uno R3 (On the back of the battery compartment)
OUTPUT
Relay (On the back of the battery compartment)
Solenoid Door Lock (On the door where the battery is stored)
DFPlayer Mini (On the back of the battery compartment)
Speaker (To the right of the battery compartment)
System B
INPUT
PIR Sensor (To the right of the battery compartment)
MICROCONTROLLER
ESP32 (On the left side of the battery compartment)
OUTPUT
Access Point
Internet
Telegram
Cloud Sysem

Figure 1. Block diagram

It can be explained that in system A there is an RFID sensor input that works by using a registered RFID card to open the door lock solenoid and the speaker will sound indicating the correct RFID card is used. When the RFID card that is attached has not been registered or the wrong card is pasted, the speaker will sound and when the card that is attached is not registered and is wrong up to three times, the siren will sound. Then in system B there is a PIR sensor input which works by detecting movement on the BTS when the sensor is active it will send input to ESP32 which is sent to BOT Telegram and can be accessed by the Technician Team.

B. Flowchart System

The design of the system described above will be depicted in Fig. 2.

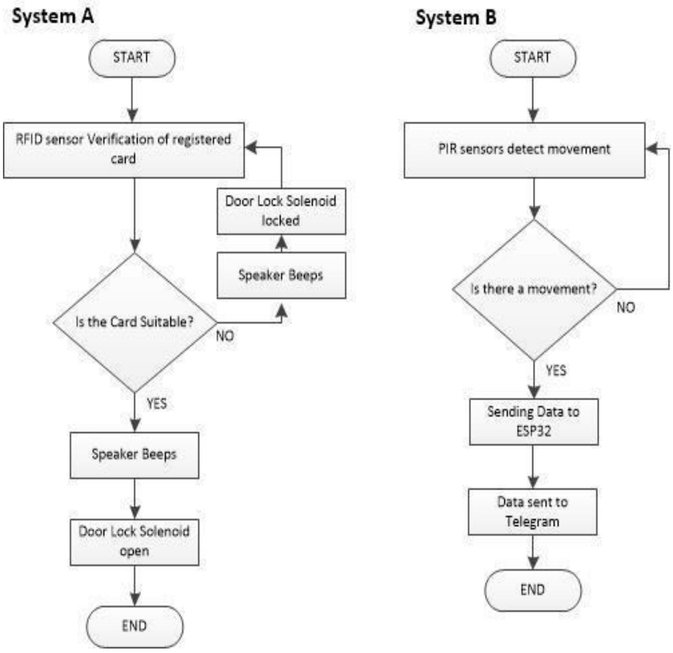


Figure 2. Design of the system

It can be explained that in system A when the RFID sensor will check whether the registered card is correct or not, if it does then the speaker will sound and the door lock solenoid will open but if it's not then the solenoid will remain locked. Then when an unregistered RFID card is attached and wrong up to three times, the siren will sound. Then on system B when the PIR sensor detects movement, if so, then ESP32 will send data to BOT Telegram which can be accessed by the technician team.

C. Schematic Diagram of System A

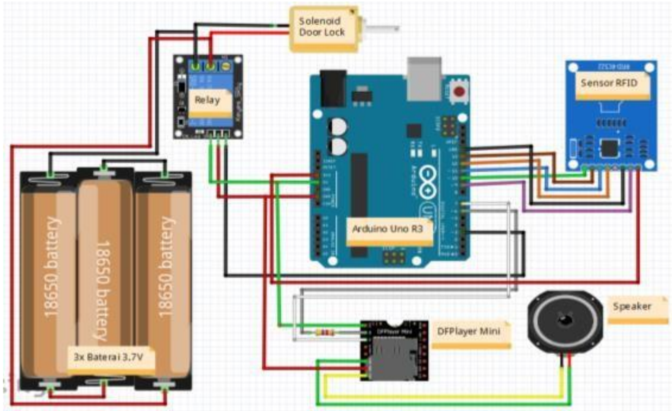


Figure 3. Schematic diagram of system A

In the Schematic Diagram of System A, it is explained that to connect the Arduino Uno R3 with the mini DFPlayer module using a 4k7 resistor and the voltage input using a 5V charger adapter cable to run the Arduino Uno R3 and the Radio Frequency Identification (RFID) sensor.

In Table I, it can be seen for the wiring circuit between the Android Uno R3 and the RFID Sensor used for input from system tool.

TABLE I
ARDUINO PIN CABLE TO RFID SENSOR

PIN ARDUINO	PIN RFID-RC 522
3.3 V	3.3 V
GND	GND
-9	RST
-10	SDA
-11	MOSI
12	MISO
13	SCK

In Table II, pin SPK_1 is connected to the positive cable, and pin SPK_2 is connected to the negative cable from the speaker.

TABLE II
ARDUINO TO DFPLAYER CABLE

PIN ARDUINO	DFPLayer mini
5V	VCC
-6	RX
7	TX
	SPK_1
GND	GND
	SPK_2

TABLE III
ARDUINO TO RELAY CABLE

PIN ARDUINO	DFPLayer mini
5V	VCC
-6	RX
7	TX
	SPK_1
GND	GND
	SPK_2

Then in Table III, the Common Contact (CC) pin is connected to the positive cable, and the Normally Closed (NC) pin is connected to the negative cable of the door lock solenoid and 12V battery. Door lock solenoid and speaker are the output of system B which is placed on the BTS door.

D. *Schematic Diagram of System B*

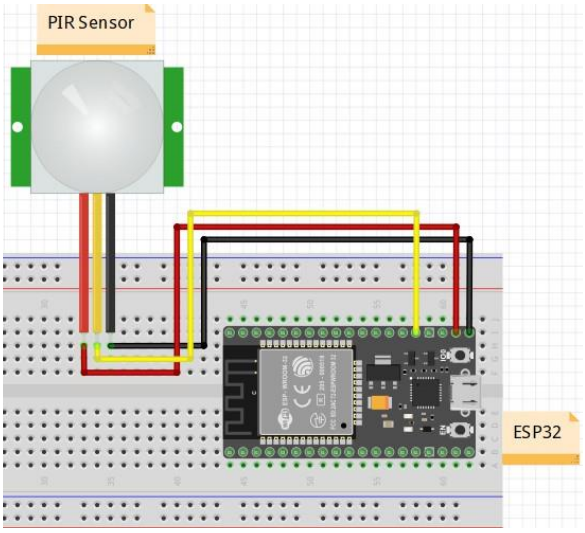


Figure 4. Schematic diagram of system A

The Schematic Diagram of System B explains that for voltage input using a microUSB cable and a 5V charger adapter to run the ESP32 microcontroller and infrared sensor (PIR). Then the PIR sensor VCC is connected to pin 3V3, the OUT pin on the PIR is connected to pin D4, and GND is connected to GND on ESP32. In System B, ESP32 is useful for receiving input results from the PIR sensor and then sending them to a telegram that can be accessed by the technician team.

E. *Tool Design*

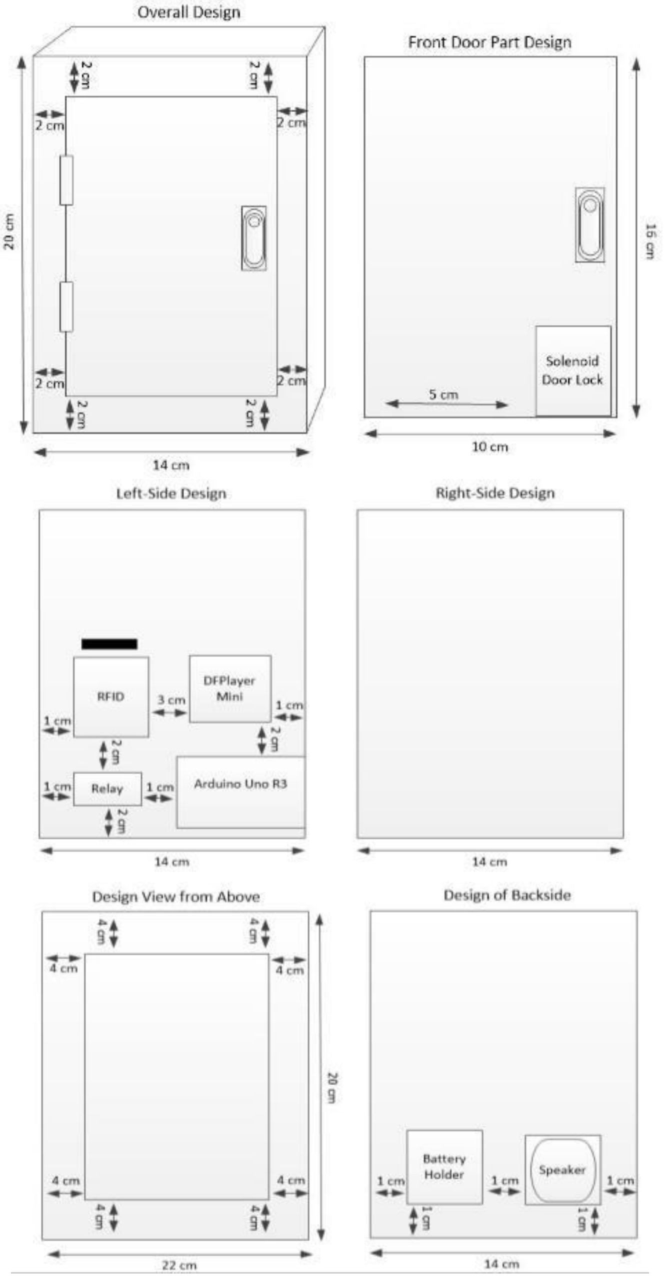


Figure 5. Tool design

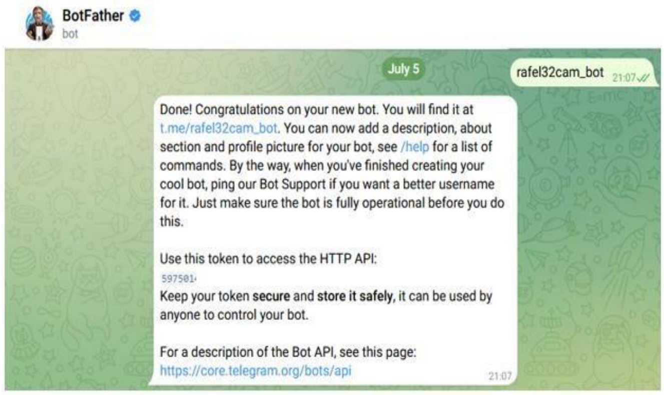
The specifications of the miniature design results can be explained in the following explanation:

- On the basis of the miniature using acrylic with a size of 4 mm with a size of 20 cm x 22 cm.

- On the right, left and back miniature walls, use acrylic with a thickness of 3 mm with a size of 10 cm x 16 cm.
- On a miniature door frame with a thickness of 3 mm with a size of 14 cm x 20 cm.
- At the top of the miniature with a thickness of 3mm with a size of 14 cm x 14 cm.

F. *Telegram Bot Software Design*

1) Designing for Token Bot: Search BotFather and send a message to BotFather:/start, to start a conversation with the bot then send a message /newbot to create a telegram bot and send a message with the bot name to be used with the conditions (Bot Name) bot. Then after that send a message with the username of the bot that was created if so, there will be a message from BotFather with Token access to access the bot that was created.



BotFather
bot

July 5

rafel32cam_bot 21:07

Done! Congratulations on your new bot. You will find it at t.me/rafel32cam_bot. You can now add a description, about section and profile picture for your bot, see /help for a list of commands. By the way, when you've finished creating your cool bot, ping our Bot Support if you want a better username for it. Just make sure the bot is fully operational before you do this.

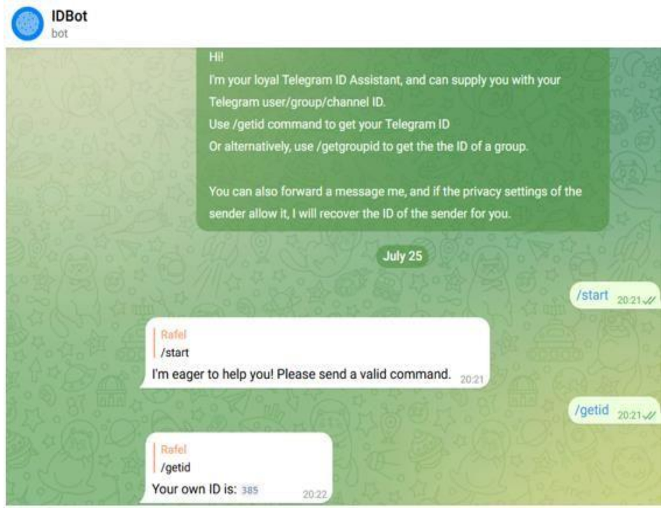
Use this token to access the HTTP API:
597501

Keep your token **secure** and **store it safely**, it can be used by anyone to control your bot.

For a description of the Bot API, see this page:
https://core.telegram.org/bots/api 21:07

Figure 6. Creating a bot in BotFather

2) Design for Chat ID: Search IDBot and send a message to IDBot: /start, to start a conversation with the bot and then send the message /getid to get the Chat ID from the IDBot bot, if so then IDBot will send the message "Your own ID is" through the message.



IDBot
bot

Hi!
I'm your loyal Telegram ID Assistant, and can supply you with your Telegram user/group/channel ID.
Use /getid command to get your Telegram ID
Or alternatively, use /getgroupid to get the the ID of a group.

You can also forward a message me, and if the privacy settings of the sender allow it, I will recover the ID of the sender for you.

July 25

/start 20:21

Rafel
/start
I'm eager to help you! Please send a valid command. 20:21

/getid 20:21

Rafel
/getid
Your own ID is: 385 20:22

Figure 7. Creating a ChatID in IDBot

G. *Miniature (Front View)*

Miniatures are made with acrylic, for system A and system B tools using a voltage of 5V.

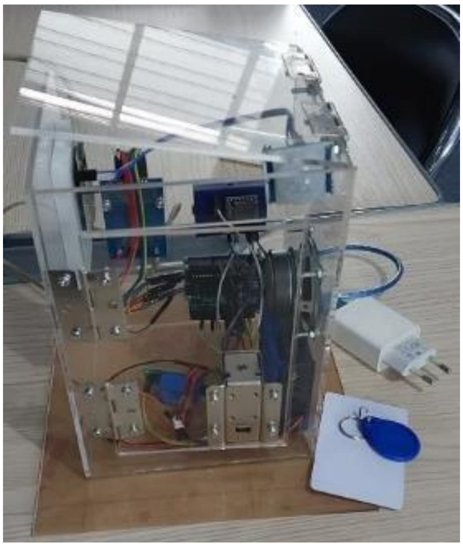


Figure 8. Miniature view (front view)

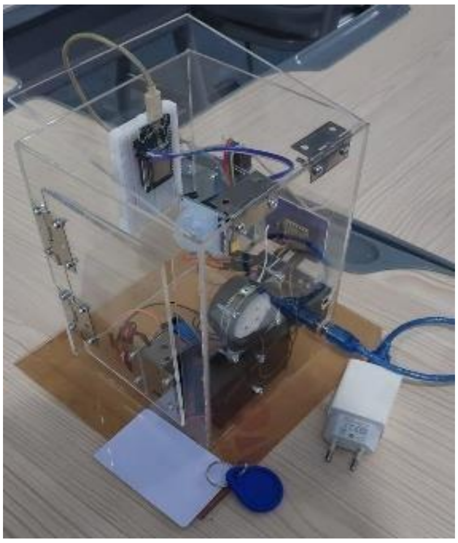


Figure 9. Miniature view (oblique view)

Miniature Display (View A It can be seen from the picture above that the design of system A is placed on the left and system B is placed on the right of the miniature place, the miniature design is made in such a way as the real conditions of the Base Transceiver Station (BTS) both systems use a 5V charger adapter as a power input.

H. *Quality of Services*

Data discussion involves the analysis and interpretation of the results obtained during the testing phase.

1) Delay: to calculate the Delay on the internet connection used on the microcontroller using the formula (1).

$$Delav = \frac{total\ delay}{total\ packages\ received} \quad (1)$$

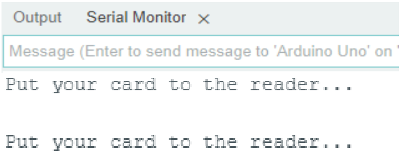
2) Packet Loss: to calculate Packet Loss on an internet connection used on a microcontroller using the formula (2).

$$Packet\ Loss = \frac{(Datapackets\ transmitted - Datapackets\ received)}{Data\ packets\ transmitted} \times 100\% \quad (2)$$

III. RESULTS AND DISCUSSION

A. Tool Testing System A

Testing the system A tool is done by trying with one RFID card and one RFIS tag which is tested by using 3 possibilities including, when using an unregistered RFID tag and incorrectly using the card used up to 3x which causes an alarm to sound. Fig. 11 is a serial monitor display when the tool is turned on.



```
Put your card to the reader...

Put your card to the reader...
```

Figure 11. Serial Monitor when the device is powered on

When the registered RFID card is attached to the RFID Reader, the speaker will sound and the door lock solenoid will open as shown in Fig. 12 and the display on the Serial Monitor in Fig. 13.

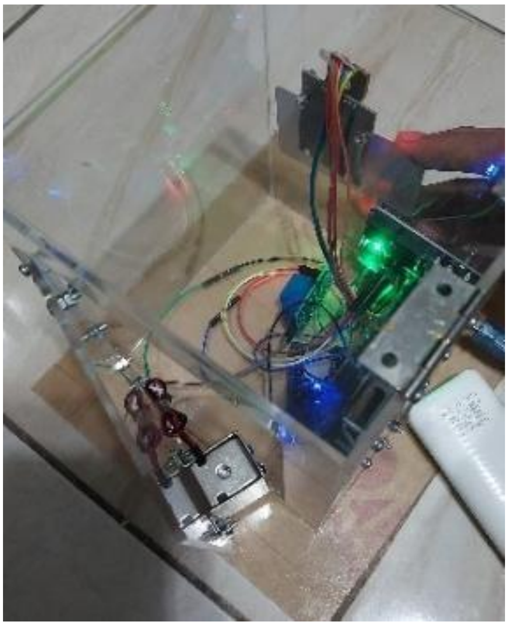
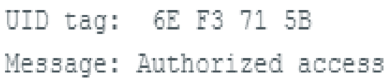


Figure 12. Display when the registered RFID card is affixed



```
UID tag:  6E F3 71 5B
Message: Authorized access
```

Figure 13. Serial Monitor when an RFID card is registered

Then when an unregistered RFID tag is attached to the RFID Reader, the speaker will sound and the door lock solenoid will be closed as shown in Fig. 14 and the display on the Serial Monitor can be seen in Fig. 15.

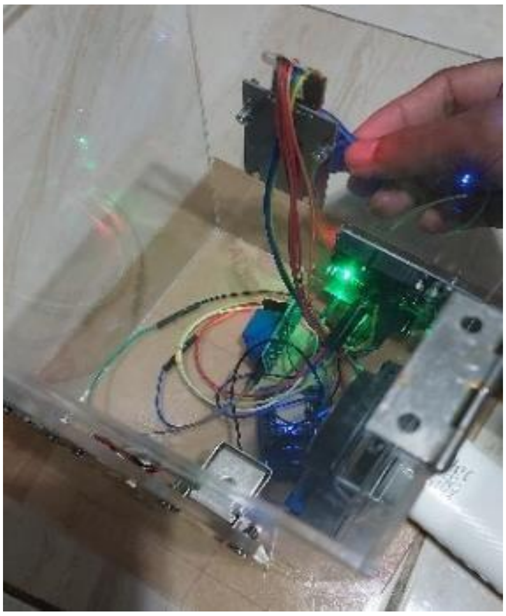
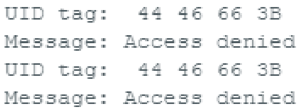


Figure 14. Display when an unregistered RFID tag is affixed



```
UID tag:  44 46 66 3B
Message: Access denied
UID tag:  44 46 66 3B
Message: Access denied
```

Figure 15. Display when unregistered RFID tag

Then when an unregistered RFID tag is attached to the RFID Reader and is wrong up to 2 times, the speaker will sound and the door lock solenoid will be closed as shown in Fig. 16 and the display on the Serial Monitor can be seen in Fig. 17.

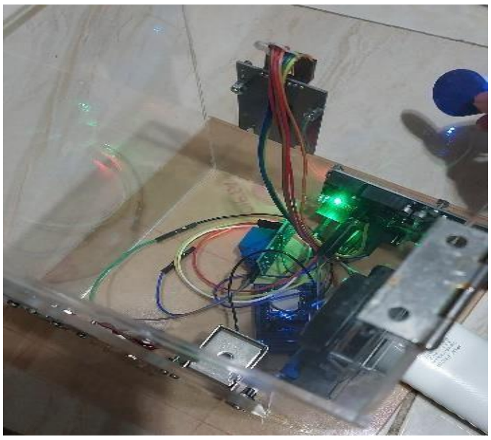
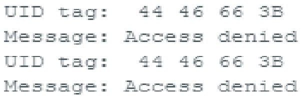


Figure 16. Display when the registered RFID card is affixed



```
UID tag:  44 46 66 3B
Message: Access denied
UID tag:  44 46 66 3B
Message: Access denied
```

Figure 17. Display when an RFID tag is not registered

The following Table IV shows the testing of three different conditions of testing system tool A.

TABLE IV
TOOL TESTING SYSTEM A

Testing	RFID Tag	Solenoid Door Lock	Speaker	DFPlayer Mini
Scan RFID Tag correctly	6EF3715B	Open	Sound 0001.mp3 (correct)	On
Incorrect RFID Tag Scan	4446663B	Not Open	Sound 0002.mp3 (wrong)	On
Scan RFID Tag incorrectly 3x	4446663B	Not Open	Sound 0003.mp3 (sirene)	On
Closing the door	-	Lock	-	Off

In Table IV are the results of testing the dual security system designed and made on system tool A, which has three different conditions such as when using a registered RFID card, when unregistered card up to approximately 3 times.

The following Table V shows the testing of different distance of testing system tool B.

TABLE V
RFID READER READING DISTANCE TESTING

TAG DISTANCE (CM)	READING TEST
1 CM	Detected
2 CM	Detected
3 CM	Detected
4 CM	Detected
5 CM	Detected
6 CM	Not Detected
7 CM	Not Detected
8 CM	Not Detected

In Table V are the results of testing the double lock security system made in system A, which is tested by attaching the RFID card and RFID tag to the RFID reader using several different distances.

B. Tool Testing System B

The following Table VI shows the testing of different distance of testing system tool B.

TABLE VI
PIR SENSOR SENSITIVITY TEST RESULTS

Object Distance (m)	Telegram Bots	Days		Night	
		Lux Meter	Delay	Lux Meter	Delay
1 m	Motion Detection	21	5,29 sec	1	6,90 sec
2 m	Motion Detection	17	6,20 sec	0	6,15 sec
3 m	Motion Detection	19	5,24 sec	0	7,39 sec
4 m	Motion Detection	15	5.94 sec	1	6,85 sec
5 m	Motion Detection	9	6,13 sec	1	7,18 sec
6 m	-	-	-	-	-
7 m	-	-	-	-	-
8 m	-	-	-	-	-

From the test results in Table VI which are carried out when in the daytime and at night, an average delay of 5.75 seconds is obtained for daytime conditions with an average at night with an average lux meter value of 0.6. This is a pretty good number because the infrared (PIR) sensor works well by showing a high level of sensitivity.

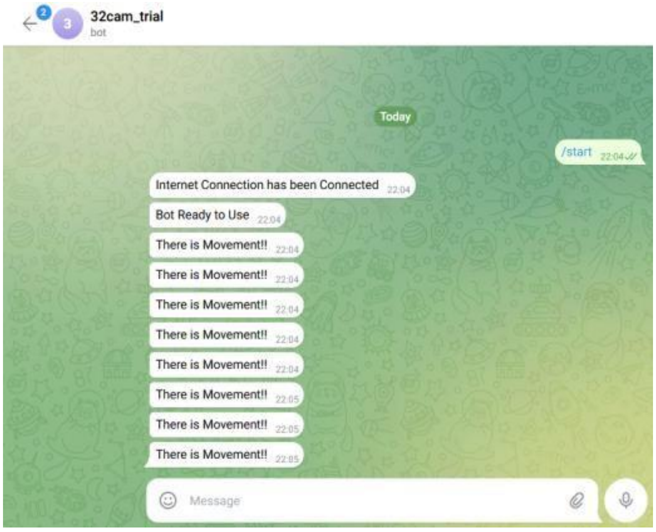

32cam_trial
bot
Today
/start 22:04
Internet Connection has been Connected 22:04
Bot Ready to Use 22:04
There is Movement!! 22:04
There is Movement!! 22:04
There is Movement!! 22:04
There is Movement!! 22:04
There is Movement!! 22:04
There is Movement!! 22:05
There is Movement!! 22:05
There is Movement!! 22:05

Figure 18. Telegram bot during montion detection

In Fig. 18, when the device is turned on, the telegram bot will receive a notification "Internet Connection has been connected" which indicates that the ESP32 device is successfully connected to the internet network. Then there is a notification "Bot ready to use" which indicates that the system B is ready to use and the notification "the is movement" indicates that the PIR sensor detects movement and sends input to the telegram bot.

TABLE VII
INTERNET SIGNAL QUALITY TESTING

Provider connection x		Provider connection y	
Delay	Packet Loss	Delay	Packet Loss
0.110ms	0.2%	0.108ms	0%
0.004ms	0.1%	31.861ms	0%
0.004ms	0%	225.99ms	0%
0.255ms	1.2%	0.148ms	0%
0.003ms	0.2%	0.192ms	0%
0.002ms	0.1%	0.122ms	0%
1.006ms	2.0%	0.004ms	0%
0.504ms	0%	0.035ms	0%
0.025ms	0%	0.131ms	0%
0.012ms	0%	0.001ms	0%

For a comprehensive comparison, the raw data from Table VII is summarized with key statistical indicators below.

TABLE VIII
COMPREHENSIVE COMPARISON

Metric	Provider	Min	Max	Average	Std. Deviation	Notes
Delay (ms)	X	0.002ms	1.006ms	0.1925ms	~0.316ms	Consistent, low latency.
	Y	0.001ms	225.99ms	25.86ms	~71.3ms	Unstable; two severe spikes.
Packet Loss (%)	X	0%	2.0%	0.38%	~0.65%	Minor, occasional losses.
	Y	0%	0%	0%	0%	

Provider X Analysis: Exhibits excellent latency performance with an average delay of only 0.1925 ms and no extreme spikes. This indicates a highly stable and responsive connection, which is ideal for the instant transmission of intrusion alerts. However, it shows occasional packet loss (avg 0.38%), meaning a very small fraction of data packets failed to reach their destination. While low, this could, in rare instances, lead to a missed notification.

Provider Y Analysis: Shows a critical flaw in latency stability. While most delay values are low (even better than Provider X in some iterations), two severe spikes (31.861ms and 225.99ms) drastically increase the average. A 225.99ms delay is significant for a real-time system and would cause a noticeable lag in alert delivery. Conversely, its 0% packet loss record demonstrates perfect reliability in data delivery when the connection is established.

The test reveals a clear trade-off: Provider X offers superior and consistent speed (low average delay) at the cost of minor reliability (low packet loss). Provider Y offers perfect reliability (zero packet loss) but suffers from unacceptable and unpredictable latency spikes (high delay variance).

For an IoT security system where timely alert delivery is paramount, Provider X is the more suitable choice. The consistent low latency ensures near-instantaneous notifications. The observed minimal packet loss rate (0.38%) poses a much lower operational risk compared to the potential multi-second alert delays inflicted by Provider Y's latency spikes. It is recommended to proceed with Provider X as the primary network for the system, with the note that its signal quality at the specific installation site is excellent for the intended application.

IV. CONCLUSION

Based on the results and testing of the IoT-based BTS Security System, it can be concluded that the system operates effectively, where the PIR sensor has an average delay of 5.76 seconds when sending notifications to Telegram, and the RFID reader can effectively read registered cards at a distance of 1 to 5 cm to open the door solenoid, whereas cards beyond 6 cm will not be detected. The PIR sensor performance is also considered stable in both daytime (16.2 lux) and nighttime (0.6 lux) conditions. For future system development, it is suggested that the Arduino Uno R3 microcontroller be replaced with a device capable of direct internet connection so that input monitoring can be done via smartphone, and a sensor should be added to monitor the battery compartment in dark conditions. Additionally, using a power supply as the main power source is recommended for more stable device performance, and adding a tracker module to the battery is highly recommended to track the battery's location in the event of theft.

REFERENCES

[1] A. A. Alimi, K. O. Odeyemi, and O. E. Oyebo, "An overview of cellular network infrastructure and management," in *Proc. World Congr. Eng.*, London, U.K., 2019, vol. 2, pp. 876–881.

[2] M. H. Husnoo, A. R. Namayand, and R. K. R. D. D. Singh, "Power backup solutions for remote telecommunication base stations: A review," *Int. J. Renew. Energy Res.*, vol. 10, no. 3, pp. 1123–1134, Sep. 2020.

[3] A. Farrih, *Free Networking and SSH Every Day*, Sep. 21st, 2013. [Online]. Available: http://privacyyou.blogspot.com/2013/09/bts-device.html. [Accessed: Sep. 7, 2023].

[4] "Maling Baterai BTS Telkomsel, AKP Sunaryo: Kerugian Sebesar 96 Juta," *Transsumatera.id*, Dec. 31, 2021. [Online]. Available: https://transsumatera.id/2021/12/31/ [Accessed: Aug. 27, 2023].

[5] A. A. Sukmandhani, "QoS (Quality of Services)," Bina Nusantara University, 2020. [Online]. Available: https://onlinelearning.binus.ac.id/computer-science/post/qos-quality-of-services [Accessed: Sep. 7, 2023].

[6] R. R. P. S. Fernando and S. C. Gupta, "Security vulnerabilities in telecommunications infrastructure: A systematic review," *IEEE Access*, vol. 9, pp. 123456-123470, 2021.

[7] M. I. S. Siregar, "Design of electronic security system equipment in BTS shelter in real time via SMS based on ATMega16 microcontroller and GSM module," *J. Phys.: Conf. Ser.*, vol. 978, no. 1, p. 012045, 2018.

[8] Z. Y. Hutapea and L., "Sensor network implementation for Base Transceiver Station (BTS) monitoring," *Komputika: J. Comput. Syst.*, vol. 8, no. 1, pp. 13–20, Jun. 2019.

[9] R. Ardiansyah, Ferdiansyah, and I. Susanti, "Implementation of infrared sensors and cameras for site security system BTS via Telegram based on Raspberry PI 3," in *Proc. 4th Int. Conf. Skanika*, 2021, pp. 44–49.

[10] E. C. Nugroho, P. Paryanta, and S. Fikri, "Door Locker Via Wifi Based on Arduino Uno R3 Microcontroller," *Go Infotech: STMIK AUB's Sci. J.*, vol. 26, no. 1, pp. 76–84, 2020.

[11] P. A. Topan, D. F. S. A. Rohman, S. B. J. J. and Y. F., "Utilization of Arduino and DFPlayer Mini Technology for Audio Player Device at Raudhatul Jannah Mosque, Gontar Village, Sumbawa Regency, West Southeast Nusa Tenggara," *J. Abdi Insani*, vol. 9, no. 4, pp. 1797–1807, 2022.

[12] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[13] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013.

[14] I. Stojmenovic, "Machine-to-machine communications with in-network data aggregation, processing, and actuation for large-scale cyber-physical systems," *IEEE Internet Things J.*, vol. 1, no. 2, pp. 122–128, Apr. 2014.

[15] Espressif Systems, *ESP32 Series Datasheet*, Version 3.9, 2022. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/ [Accessed: Aug. 29, 2023].

[16] A. S. Sahidin and S. Alam, "Automatic Hand Washing Machine Using Proximity Sensor and DFPlayer Mini Based on Arduino Uno," *J. Mosfet*, vol. 1, no. 1, pp. 1–7, 2021.

[17] A. I. Purnamasari and A. S. , "Development of Passive Infrared Sensor (PIR) HC-SR501 with ESP32-CAM Microcontrollers based on Internet of Things (IoT) and Smart Home as Motion Detection for Residential Security," *Prosiding SISFOTEK*, vol. 3, no. 1, pp. 148–154, 2019.

[18] A. Candra and F. Nurlaila, "Design of a Locker Security System Using Arduino Uno Based RFID on Employee Lockers of SMK Yadika 2 Jakarta," *BULLET: J. Multidiscip. Sci.*, vol. 1, no. 04, pp. 712–720, 2022.

[19] R. G. Pratama, "Design of Arduino Uno R3 based house locking system with Radio Frequency Identification (RFID) and selenoid door lock," *Ubiquitous: Comput. Appl. J.*, vol. 2, no. 1, pp. 45–50, Jan. 2019.

[20] F. A. R. B. Styanto and E. Firmansyah, "Designing a safe security tool using RFID and buzzer alarm based on internet of things with Telegram notification," *J. Inf. Technol.*, vol. 8, no. 2, pp. 132–139, Oct. 2022.