One Standard to Rule Them All? Assessing the Disruptive Potential of Jamming Attacks on Matter Networks

Felix Klement, Emily Vorderwülbeke and Stefan Katzenbeisser
Chair of Computer Engineering, University of Passau, Germany
{felix.klement, stefan.katzenbeisser}@uni-passau.de, vorderwuel@fim.uni-passau.de

***Abstract*—This study addresses the vulnerability of Matter networks against reactive jamming attacks in Internet of Things (IoT) solutions. Through an analysis of various scenarios, we provide empirical evidence demonstrating the substantial threat posed by these attacks, as indicated by their exceptionally high success rate of 91%. Additionally, we identify significant weaknesses in the jamming detection mechanism within OpenThread, a widely used implementation of the Thread protocol. To address this issue, we propose a novel passive jamming detection approach fot the transport and networking layer in Matter with an accuracy of 96%. Our findings emphasize the need for robust security measures in Matter networks, such as integrating passive detection nodes or directly incorporating detection mechanisms into the Matter standard. Future work involves expanding scenario coverage and exploring efficient integration options within the standard.**

***Index Terms*—Security, IoT, Matter, Jamming, Jam Detection**

I. Introduction

There is a problem that should probably be known to many involved in dealing with consumer Internet of Things (IoT) solutions, e.g. for a smart home. Namely, how to facilitate connectivity between different devices that cannot be connected via a common bridge or anything equivalent. Until today, there have been some efforts to fix this problem. There are now a number of different IoT standards and open source implementations of the approaches: oneM2M (*OpenMTC*) [1], OPC-UA (*open62541*) [2], DDS (*OpenDDS*) [3], OCF (*IoTivity*) [4]. However, none of them has yet gained widespread acceptance in the field of consumer home automation. One of the reasons for this is that many of the above-mentioned standards are mostly geared towards Machine-to-Machine (M2M) and industrial IoT environments. As a result, few or none of the leading manufacturers for home automation in the consumer sector are incorporating these standards into their products. On the one hand, the alliance developing Matter consists of the largest manufacturers of consumer products such as Apple, Google, Amazon, Comcast, Zigbee Alliance, IKEA, Huawei, Schneider and many more. Secondly, this standard is based on the network protocol Thread. Thread is an already well known and established IPv6-based mesh network protocol that uses 6LoWPAN and utilises IEEE 802.15.4 internally. It enables reliable and secure communication between smart home devices with fast response times, increased range and long battery life. The most widely recognized and extensively employed protocol implementation is OpenThread, developed by Google. There are already many devices that are Thread compatible out of the box. These include Amazons eero Wi-Fi router, some Belkin WeMo smart switches and Googles Nest products [5].

To the best of our knowledge, this is the first investigation of the Matter standard from a security perspective, examining the underlying security assumptions and implementations with regard to their susceptibility to jamming attacks. In this context, the work at hand addresses the robustness of the aforementioned standard in practice. Specifically, it raises the question of whether the standard is susceptible to interference from jamming attacks. Of particular interest is the effectiveness of the integrated Thread protocol, which incorporates jamming detection functionality, in identifying and countering such attacks. Moreover, it necessitates a systematic exploration of viable methods to improve the prevailing detection mechanisms or augment their accuracy.

In our study, we outline a range of scenarios and assess their efficacy. We demonstrate that our jamming attacks yield succeful outcomes in around 91% of cases. Additionally, our investigation reveals substantial vulnerabilities in the Jam Detection (JD) mechanism within OpenThread when confronted with our specific attack methodology. Lastly, we introduce a novel passive jamming detection approach utilizing a gradient boosting classifier, which exhibits an impressive 96% accuracy in detecting such attacks.

The subsequent sections of this paper is structured as follows: The following section presents an overview of relevant literature. Section III provides a short explanation of the Matter standard. Section IV outlines the adversary model, while Section V elaborates the methodology employed for our jamming attack. In Section VI, we showcase the passive JD technique. Finally, Section VII summarizes our findings, presents concluding remarks, and offers prospects for future research.

This work has been partially funded by the German Federal Ministry of Education and Research – Bundesministerium für Bildung und Forschung (BMBF), as part of the Project "6G-RIC: The 6G Research and Innovation Cluster" (project number 825026), as well as, by the Bavarian Ministry of Science within the framework of the research cluster "ForDaySec: Security in everyday digitalisation".

II. Related Work

In the subsequent section, we provide a concise overview of relevant research concerning jamming in wireless networks. Additionally, we analyze select publications related to OpenThread, which is of particular significance due to its utilization of Thread as the foundational network and transport layer in Matter. It is important to note that attacks targeting the network protocol can have a significant impact, and any discovered insights or tangible outcomes can be further developed and expanded as required.

A. Jamming Attacks in Wireless Networks

Research on jamming attacks in wireless networks has been a subject of investigation for a considerable duration. Xu et al. [6] made notable contributions by pioneering the categorization of various jamming attack models. Their work encompassed the identification of constant, deceptive, random, and reactive jammers, laying the foundation for the study of jamming attacks on wireless networks. Additionally, they devised two distinct detection algorithms capable of accurately classifying the aforementioned jamming models.

Wilhelm et al. [7] conduct an extensive investigation into the entanglements of jamming attacks in wireless networks, with particular emphasis on reactive jamming attacks. Their findings indicate that the advent of software-defined radios amplify the vulnerability to reactive jamming attacks. Notably, the study highlights the concerning fact that the construction of a reactive jammer is achievable even with inexpensive hardware, devoid of high-end technological requirements, while still retaining the capability to effectively disrupt wireless networks.

Aras et al. [8] focus on the examination of selective jamming attacks within LoRaWAN networks, representing a specific variant of reactive jamming attacks. The study encompasses the evaluation of both a standalone selective jamming attack and a combination of both jamming and wormhole attacks, wherein the jammer is divided into two devices linked via an alternative network. Notably, both attacks proved successful in their endeavors, leading to adverse consequences for the targeted network. It is worth noting that this experimentation employed readily available, inexpensive hardware, thereby underscoring the fact that disrupting a network does not necessitate the use of costly, high-end technology.

B. Pertinent OpenThread Security Publications

Akestoridis et al. [9] show how Thread behaves in terms of network security. To do this, they use hardware and software tools they have already developed for security analysis of Zigbee networks. This is possible because Zigbee and Thread are both based on the IEEE 802.15.4 standard. Among other things, they demonstrated how to successfully implement an energy-deprivation attack.

In [10] the authors show a taxonomy for security assessment for Building Automation Systems (BAS) protocols. Using this, they show that Thread is vulnerable to a few novel attack possibilities. Furthermore, they offer some improvements that could enhance the performance of the network as well as the security of Thread.

Dinu et al. [11] perform a side-channel analysis of the Thread networking stack. They show how network-specific mechanisms and differential electromagnetic analysis can allow an attacker to manipulate the security material or access network credentials. In general, however, the result is that Thread's sophisticated security mechanisms prevent a side-channel attack from being easy.

The absence of any existing published research regarding the behavioral implications in the context of jamming attacks for Matter networks emphasizes the need of our research effort. Specifically, we aim to investigate the potential realization of such attacks and ascertain the efficacy of relevant technologies in detecting such adversarial actions through appropriate methodological approaches.

III. Matter Standard

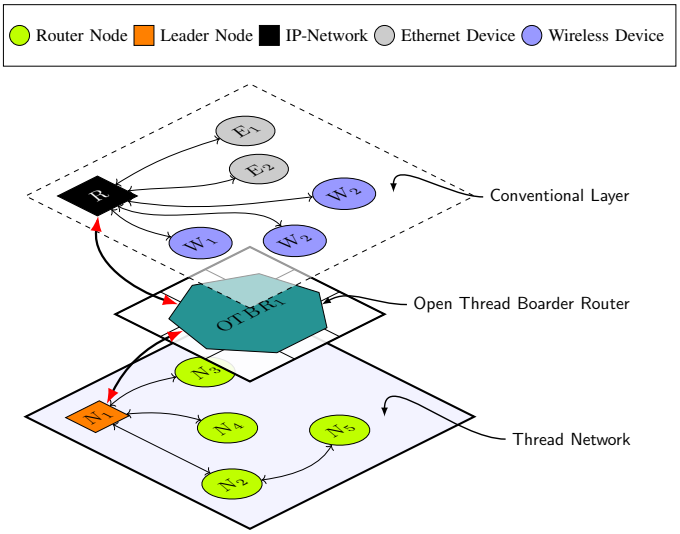

Fig. 1. Rough overview of the Matter network layers and the interconnection between them

Matter is a standard based on the Internet Protocol (IP). It uses Wi-Fi [12] and Thread [13] as network layers and uses Bluetooth low energy for the onboarding process of new devices into the network. A rough overview of the individual layers and their interaction can be taken from Figure 1. Here the upper layer is the conventional layer with various ethernet and wireless (Wi-Fi, Bluetooth) devices connected via a router. The router is in turn connected to the Open Thread Boarder Router (OTBR) and thus enables access to the Thread network. Within this Thread network one leader node and several router nodes are connected to each other. There are a number of other network components within Thread, which we will not discuss in detail in this paper. With Matter, the consortium is developing an application layer standard that builds on the previously mentioned technologies. It is important to note that it is not a standalone network

protocol. The main goal is to simplify the development of IoT devices for manufacturers and at the same time streamline the interoperability of devices for end users. The project adheres to an open-source methodology, augmenting transparency and consequently strengthening credibility in regards to security measures. With this approach, the attack surface in terms of security evaluations is much clearer than with proprietary implementations. The Connectivity Standards Alliances (CSA) ventures are supported by many industry giants such as Amazon, Apple and Google. From our point of view, this is promising, because it increases the chance enormously that this standard will really be applied across the board. In the following subsections, we will briefly highlight and concisely explain the functionalities to provide the reader with a holistic overview.

A. Thread Network/Transport Layer

Matter is based on the network and transport layer called Thread. The specification [14] is an IP based mesh network protocol specifically for IoT. It is based on IEEE 802.15.4 at the PHY/MAC layers. Generally speaking, it can be considered a secure wireless mesh network protocol based on existing IEEE and IETF standards. However, it is fundamentally different from other protocols in this area such as Bluetooth Low Energy (BLE), Mesh or Zigbee. Within the network 6LoWPAN is used to transmit IPv6 packets via Low-Rate Wireless Personal Area Network (LR-WPAN), using both IP routing and User Datagram Protocol (UDP). An open-source implementation of the standard has been developed by Google since 2017 under the name OpenThread [15].

1) Jam Detection: OpenThread provides a feature for detecting possible jamming attacks within the network. Which we review in the following sections under the impact of our attack on functionality. The detection mechanism utilizes the monitoring of the Received Signal Strength Indicator (RSSI), which quantifies the strength of Radio-Frequency (RF) signals received, over a predefined time frame. In detail, first the JD state is set to false, a RSSI threshold value, a detection window in which the RSSI is measured, and an occupancy time that must be smaller than the detection window are defined. Then one defines the number of seconds in which the RSSI value must exceed the RSSI threshold value. In one second intervals, the RSSI is sampled multiple times and if every reading of the RSSI in this time frame is above the specified threshold, this one second interval is regarded as jammed. If an aggregated number of one second intervals is more than or equal to the busy period seconds, the JD state is set to true, else it is set to false. In Figure 2, we present a segment of the recorded RSSI values, expressed in decibels milliwatts (dBm), alongside the operational state of the JD mechanism within an operational Matter network. In this example the busy period is set to 8, the detection window is set to 16 and the threshold is set to -45 dBm. This results in the bitmap of the last 64 seconds as shown in Figure 3. The bit at position 2^6 is the eighth in a row within sixteen steps. Thus the state of the JD is switched from false to true from then on.

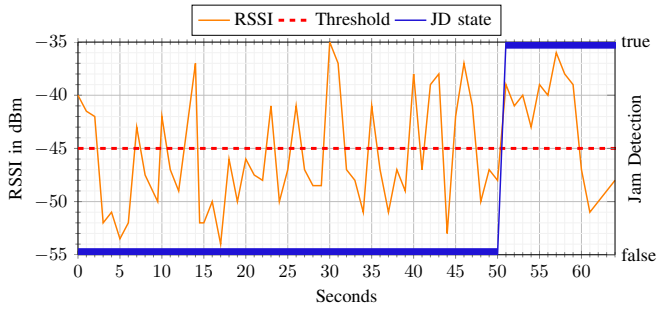

Fig. 2. Correlation of the RSSI samples and the JD state over a period of 64 seconds.

B. Platform Security

Matter provides different variable levels of security depending on the needed requirements. Basically, each individual Matter-enabled device implements an application layer built upon a Matter software stack. This implementation in turn runs on a hardware platform. The hardware can then provide various security primitives. For example, cryptographic functions, generation of random numbers, secure storage of cryptographic keys and much more.

The standard was developed according to the security by design concept. The principles are divided into the areas of privacy and security. To achieve important cryptographic goals, Matter is based on Public Key Infrastructure (PKI). Together with some protocols for session establishment, the goals for identification, data privacy and integrity are ensured. In order to verify whether the device really is what it claims to be, Matter uses a so-called root of trust (see Figure 4). Both X.509 certificates in the persistent device memory are validated together with the respective manufacturer or individual product information during the so-called commissioning process [16].

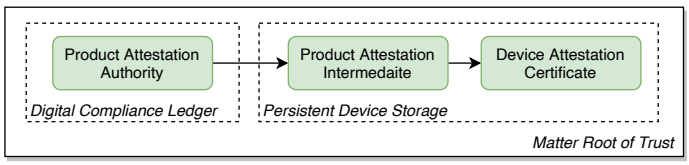

Fig. 4. Overview of the involved certificate nodes

IV. Adversary Model

In the following section we explain our proposed adversary model. We assume that the adversary possesses no prior knowledge of the targeted network and lacks cryptographic keys necessary for legitimate access to that network. The security objective at risk in our scenario with respect to Matter networks is that of availability. The attacker aims to breach this target as quickly and efficiently as possible, with its main goal being to render the targeted victim inoperable for an extended period of time without raising suspicion. Further we anticipate that the attacker possesses the essential hardware and software tools, which fall within the communication range of the targeted Matter network, enabling them to capture, transmit, jam, and analyze packets from the respective Matter network. Our assumption is that the end user does not deliberately

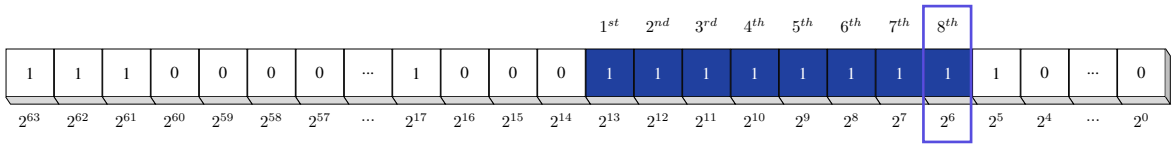

Fig. 3. Extracted Jam Detection bit map from the RSSI example in Fig.2

undermine the security of the network, as our focus lies solely within the realm of non-physical attacks.

V. Reactive Jamming Attack

In the subsequent section, we illustrate the utilization of our reactive jamming methodology to carry out an attack on the network. Within this particular attack paradigm, the attacker initially identifies a genuine packet in transmission and subsequently deploys jamming techniques to disrupt the packet. Once the jammer detects the initiation of a packet, it immediately emits a random signal that interferes with the actual signal present on the channel. If executed proficiently and timed accurately, this interference leads to corruption of the original data [17], thereby rendering the intended recipient incapable of receiving the legitimate packet. This method of jamming is regarded as an effective and energy-efficient attack strategy [18], as the jammer only consumes energy during data transmission within the network. Moreover, the jammer remains active solely during instances of legitimate network traffic, thereby rendering detection challenging, particularly through the utilization of parameters such as Received Signal Strength (RSS) values [19] as used by the JD of the Thread network.

A. Attack Scenarios

In total, our experiment is divided into five different scenarios. In each of the individual scenarios, the constellation between the transmitting node, receiving node and jammer is different. By varying the positions, we investigate what effect the position of the jammer has on the successful execution of the attack. Each scenario is divided into four sub-experiments, in each of which the jammer is active, but the jamming detection settings differ from each other. The respective configurations for jammer detection, as indicated in the columns of Table I, are as follows:

- no active JD
- active JD with a RSSI threshold of 0
- active JD with a RSSI threshold of -20
- active JD with a RSSI threshold of -45

For the JD, we set the detection window as well as the busy period to 64 seconds. Both legitimate nodes are part of the same Matter network that was previously established. In each of the individual scenarios, the router node sends a simple UDP message to the leader node of the network. At the same time, the jammer attempts to jam this message. On the receiving side, if the message transmission is successful, the RSS value of the received message is registered, otherwise the error code is stored. If the jamming detection is active, the history of the bitmap is also logged. In the following we denote $\mathcal{S}_\mathcal{N}$ as the sending node, $\mathcal{R}_\mathcal{N}$ as the receiving node, $\mathcal{J}_\mathcal{N}$ as the jammer node and $\mathcal{D}$ as the distance in centimeter between two nodes. We hereby establish the following five scenarios in which all components are linearly aligned, each of which is systematically depicted by the rows of TableI:

1) $\mathcal{D}_{\mathcal{SR}} = 100$ between $\mathcal{S}_\mathcal{N}$ and $\mathcal{R}_\mathcal{N}$, with the $\mathcal{J}_\mathcal{N}$ located exactly in the middle of both.
2) $\mathcal{D}_{\mathcal{SR}} = 50$ between $\mathcal{S}_\mathcal{N}$ and $\mathcal{R}_\mathcal{N}$, with the $\mathcal{J}_\mathcal{N}$ located to the left of the $\mathcal{R}_\mathcal{N}$ with $\mathcal{D}_{\mathcal{JR}} = 50$.
3) $\mathcal{D}_{\mathcal{SR}} = 50$ between $\mathcal{S}_\mathcal{N}$ and $\mathcal{R}_\mathcal{N}$, with the $\mathcal{J}_\mathcal{N}$ located to the left of the $\mathcal{S}_\mathcal{N}$ with $\mathcal{D}_{\mathcal{JS}} = 50$.
4) $\mathcal{D}_{\mathcal{SR}} = 100$ between $\mathcal{S}_\mathcal{N}$ and $\mathcal{R}_\mathcal{N}$, with $\mathcal{J}_\mathcal{N}$ located directly next to $\mathcal{S}_\mathcal{N}$.
5) $\mathcal{D}_{\mathcal{SR}} = 100$ between $\mathcal{S}_\mathcal{N}$ and $\mathcal{R}_\mathcal{N}$, with $\mathcal{J}_\mathcal{N}$ located directly next to $\mathcal{R}_\mathcal{N}$.

If a attacker were to attempt the disruption of matter nodes, it is probable that, in most instances, they would be positioned at relatively grater distances. However, for our particular application, the spatial separation of these nodes is not of primary significance. Our foremost objective is to empirically demonstrate the feasibility of interference with matter nodes, irrespective of their spatial proximity or distance, and more crucially, to illustrate that existing jamming detection mechanisms are inadequate in detecting reactive jamming attacks. Additionally, we contend that the introduction of our five attacker scenarios encompasses a comprehensive range of possible circumstances. Reactive jamming attacks occurring within the same constellation but at a greater spatial separation are anticipated to result in comparable detrimental consequences for the nodes.

B. Implementation

Our reactive jammer utilizes the ATUSB board developed by Qi-Hardware as part of the Ben Wireless Personal Area Network (WPAN) project. This board is capable of establishing communication with any device that adheres to the IEEE 802.15.4 standard for the 2.4 GHz band, specifically at the physical and link layers. It operates within a maximum indoor range of 10 meters and maintains a standard data rate of 25 kB/s. The board incorporates an Atmega32U2 microcontroller and the Atmel AT86RF231 transceiver, which facilitates the interface between USB and Serial Peripheral Interface (SPI) protocols. Furthermore, the ATUSB board has high precision receive time stamping, which is achieved by issuing an `RX_START` interrupt when the device synchronizes to a frame. This interrupt can be received via an interrupt mechanism between the microcontroller and the RF chip.

The operational process is as follows: The microcontroller enables the interrupt by utilizing the interrupt mask and signals its occurrence by toggling the Interrupt Request (IRQ) pin.

Upon detecting the interrupt, the microcontroller accesses the interrupt register via the SPI to retrieve relevant information. The interrupt line is capable of signaling various events, including the reception of an Acknowledgement (ACK), the initiation and completion of frame transmission. This functionality is crucial for ensuring prompt generation of an interrupt packet by the interrupt sender upon receiving a packet on a specific channel.

In our implementation of the jammer, we enhance the functionality of the Interrupt Service Routine (ISR). By default, the AT86RF231 transceiver operates in the `RX_ON` state, which corresponds to its receiving mode. In this state, the Phase-Locked Loop (PLL) frequency synthesizer is active, allowing us to intercept incoming frames. To rapidly transmit a jamming signal upon detecting an incoming signal, we utilize the `REG_IRQ_STATUS` register. When this register indicates an `IRQ_RX_START` event, we promptly initiate the jamming procedure. To accomplish this, we activate the `RX_ON` state of the transceiver by setting the `TRX_CMD` register bit within the `TRX_STATE` register. In our specific scenario, the frame payload is irrelevant; we solely define the frame size, and the transceiver transmits a frame containing the current contents of its buffer. Following the transmission, the power amplifier is automatically deactivated, and a `TRX_END` interrupt is generated. Consequently, the transceiver transitions back to the `PLL_ON` state, enabling us to resume listening for incoming frames.

C. Effectiveness Evaluation

The evaluation of the execution involves the utilization of various methodologies. We employ a combination of techniques to assess the performance. Our analysis encompasses not only the RSSI at the receiving node but also the reception of both valid and invalid packets using a Universal Software Radio Peripheral (USRP) and Wireshark. Additionally, we leverage RFtap to capture and store the signal quality of each packet. Moreover, in scenarios where the JD is activated, the devices log the history bitmap at the receiving node, providing a comprehensive view of the jamming events.

For every configuration and scenario, a total series of five hundred UDP frames executed in five cycles are generated. A single F in the column means that frame transmission failed in each of the five rounds for all one hundred frames. Table I displays the aggregated mean values derived from these individual iterations. Upon closer examination of the outcomes obtained from each distinct scenario, it becomes evident that our jamming method achieves a high level of effectiveness across the majority of cases. Out of a total of 100 execution cycles, 91 cycles had more packets successfully disrupted by the jammer than successfully transmitted. This empirical evidence substantiates the efficacy of our methodology, which exhibits an approximate success rate of around 91%. However, when only looking at scenario 3 in the table, the success rate declined to 55% . It is important to acknowledge that the observed decrease in success rate cannot be attributed to the functioning of the JD mechanism. In OpenThread, JD operates in a passive manner and does not actively undertake any preventive measures. Our assumption is that the arrangement of nodes in this particular scenario may not be optimal for achieving satisfactory jamming performance. Of particular interest is the consistent observation that the JD bitmap remains constant at `0x000000000000` across all iterations. This signifies that the employed detection methodology used in the transport and network layers of Matter networks fails to effectively identify and respond to our specific type of attack. This demonstrates the limited efficacy of JD when employing reactive jamming, thereby necessitating the development of enhanced detection methodologies. Therefore, our detection model is introduced in the subsequent section.

TABLE I
MEASUREMENTS RESULTS OF A RECEIVED FRAME.
ABBREVIATIONS: J: JAMMER; JD: JAM DETECTION; T: RSSI THRESHOLD; F: FRAME RX FAILED (RSS VALUES IN DBM)

№	**Baseline RSSI Values**	**J ✓ JD ✗**	**J ✓ JD ✓ *T: 0***	**J ✓ JD ✓ *T: -20***	**J ✓ JD ✓ *T: -45***
1)	-59, -58, -58, -56, -56	F	F	F	F
2)	-55, -53, -54, -53, -52	F	F	F	F
3)	-44, -49, -39, -43, -40	F, -42, F, F,-41	-46, -45, F, F,F	F, F, -38, -40, F	-44, -45, F, - 46, F
4)	-59, -58, -58, -56, -56	F	F	F	F
5)	-59, -58, -58, -56, -56	F	F	F	F

VI. PASSIVE JAM DETECTION

The approach for identifying attacks adheres to the passive concept of the Matter JD. The primary objective here is to achieve reliable and prompt identification of an attack. To accomplish this, we position a Linux-based USRP node within proximity of the Matter network. It should be noted that this node does not need to be an authorized component of the target network; rather, detection relies on the analysis of intercepted encrypted radio transmissions. By utilizing our trained model, we can determine in nearly real-time, the presence or absence of a reactive jamming attack within the tapped network.

Our detection model utilizes Packet Capture (PCAP) files obtained during the experimental procedures. Additionally, we retained logging files from the jammer node containing global timestamps and the current mode (attack or idle). This enables us to generate an extensive labeled dataset for training and testing purposes. After evaluating various classification techniques, the random forest and gradient boosting classifier emerged as the most suitable options for our classification problem. Given their comparable performance in our tests, we ultimately opted for gradient boosting as the preferred approach. The classification report presented in Table II demonstrates the performance metrics of our present model, revealing a weighted average f1-score of 0.96. Additionally, we have computed the Receiver Operating Characteristic (ROC) curve, as depicted in Figure 6. The analysis reveals an impressive area

under the ROC Curve (AUC-ROC) value of 0.96. Moreover, in consideration of the categorical nature of our classification task, we constructed the confusion matrix to further assess the model's performance (see Figure 5). The obtained results further evidence of the efficacy and precision of our model. Notably, the model exhibits exceptional predictive capabilities, demonstrating remarkable consistency with the actual ground truth labels within our designated test dataset.

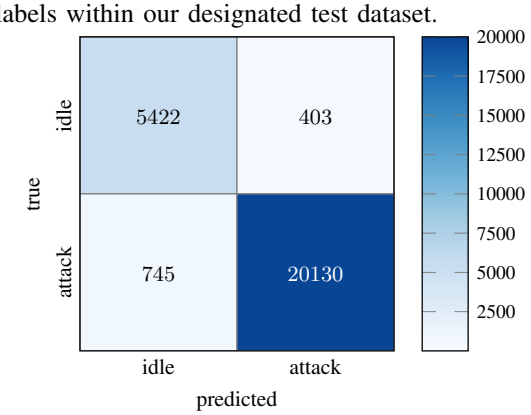

Fig. 5. Confusion matrix for the predictive accuracy of our model by comparing the predicted labels with the true labels of the dataset

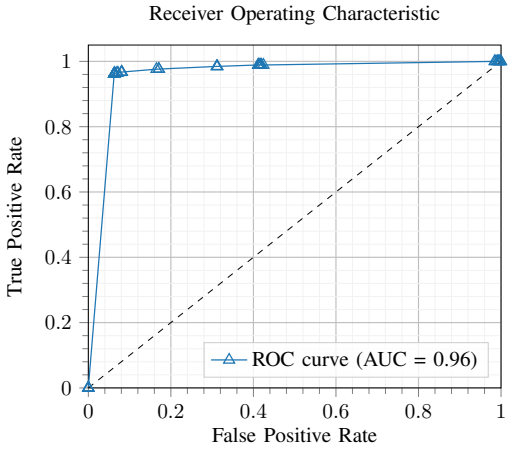

Fig. 6. ROC curve with the calculated area under the curve

TABLE II
CLASSIFICATION REPORT OF THE TRAINED MODEL WHEN USING GRADIENTBOOSTINGCLASSIFIER

	precision	recall	f1-score	support
Idle	0.88	0.93	0.90	5825
Attack	0.98	0.96	0.97	20875
accuracy			0.96	26700
macro avg	0.93	0.95	0.94	26700
weighted avg	0.96	0.96	0.96	26700

VII. CONCLUSION

Matter networks are very vulnerable to reactive jamming attacks. In this paper, we have presented 5 realistic scenarios and investigated their vulnerability to potential attacks. It can be seen through our measurements that on average, the jammer can successfully jam packets at 91%. Furthermore, we found that the JD feature in the transport and network layers has a devastatingly poor accuracy. In none of the successful attacks, the method was able to indicate an attack. In contrast, our passive detection of reactive jamming attacks, shows a very good accuracy of 96%. By using such passive detection nodes in a Matter network, the security against jamming attacks would improve immensely. The research at hand notably examines the influence of reactive jamming on both the availability of network nodes and the efficacy of attack detection mechanisms. In future work we want to extend the set of scenarios to have an even larger set of training data for our detection model. Furthermore, we are looking for solutions how to efficiently integrate this detection directly into the Matter standard, so that the additional passive detection node becomes obsolete.

REFERENCES

[1] oneM2M. (2023) The iot standard. oneM2M. [Online]. Available: https://www.onem2m.org

[2] O. Foundation. (2023) The industrial interoperability standard. OPC Foundation. [Online]. Available: https://opcfoundation.org/about/opc-technologies/opc-ua/

[3] D. Foundation. (2023) Data distribution service. DDS Foundation. [Online]. Available: https://www.dds-foundation.org

[4] O. Foundation. (2023) Open connectivity foundation. OFC Foundation. [Online]. Available: https://openconnectivity.org

[5] A. M. by Global Inventures. (2023) Thread certified products. Threat Group. [Online]. Available: https://www.threadgroup.org/What-is-Thread/Thread-Benefits#certifiedproducts

[6] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The feasibility of launching and detecting jamming attacks in wireless networks," in *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, 2005, pp. 46–57.

[7] M. Wilhelm, I. Martinovic, J. B. Schmitt, and V. Lenders, "Short paper: Reactive jamming in wireless networks: How realistic is the threat?" in *Proceedings of the fourth ACM conference on Wireless network security*, 2011, pp. 47–52.

[8] E. Aras, N. Small, G. S. Ramachandran, S. Delbruel, W. Joosen, and D. Hughes, "Selective jamming of lorawan using commodity hardware," in *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, 2017, pp. 363–372.

[9] D.-G. Akestoridis, V. Sekar, and P. Tague, "On the security of thread networks: Experimentation with openthread-enabled devices," in *Proceedings of the 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, ser. WiSec '22. Association for Computing Machinery, 2022, p. 233–244.

[10] Y. Liu, Z. Pang, G. Dán, D. Lan, and S. Gong, "A taxonomy for the security assessment of ip-based building automation systems: The case of thread," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 4113–4123, 2018.

[11] D. Dinu and I. Kizhvatov, "Em analysis in the iot context: Lessons learned from an attack on thread," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2018, no. 1, p. 73–97, Feb. 2018.

[12] W. Alliance. (2023) The worldwide network of companies that brings you wi-fi. WiFi Alliance. [Online]. Available: https://www.wi-fi.org

[13] A. M. by Global Inventures. (2023) Thread - home automation. Threat Group. [Online]. Available: https://www.threadgroup.org

[14] T. Group. (2023) Thread 1.1 specification request. Thread Group. [Online]. Available: https://www.threadgroup.org/ThreadSpec

[15] Google. (2023) An open foundation for the connected home. Google. [Online]. Available: https://openthread.io

[16] G. Jiacheng, "Matter security model," Mar. 2022. [Online]. Available: https://blog.espressif.com/matter-security-model-37f806d3b0b2

[17] A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos, and G. Pantziou, "A survey on jamming attacks and countermeasures in wsns," *IEEE communications surveys & tutorials*, vol. 11, no. 4, pp. 42–56, 2009.

[18] H. Pirayesh and H. Zeng, "Jamming attacks and anti-jamming strategies in wireless networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, 2022.

[19] M. Strasser, B. Danev, and S. Čapkun, "Detection of reactive jamming in sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 7, no. 2, pp. 1–29, 2010.