Machine Learning Based IoT Intrusion Detection System: An MQTT Case Study (MQTT-IoT-IDS2020 Dataset)

Hanan Hindy[1][0000−0002−5195−8193], Ethan Bayne[1][0000−0003−1853−2921], Miroslav Bures[2][0000−0002−2994−7826], Robert Atkinson[3][0000−0002−6206−2229], Christos Tachtatzis[3][0000−0001−9150−6805], and Xavier Bellekens[3][0000−0003−1849−5788]

[1] Division of Cyber Security, Abertay University, Dundee, Scotland, UK
{1704847,e.bayne}@abertay.ac.uk
[2] Department of Computer Science, FEE, Czech Technical University in Prague, Czechia
miroslav.bures@fel.cvut.cz
[3] EEE Department, University of Strathclyde, Glasgow, Scotland, UK
{robert.atkinson,christos.tachtatzis,xavier.bellekens}@strath.ac.uk

Abstract. The Internet of Things (IoT) is one of the main research fields in the Cybersecurity domain. This is due to (a) the increased dependency on automated device, and (b) the inadequacy of general-purpose Intrusion Detection Systems (IDS) to be deployed for special purpose networks usage. Numerous lightweight protocols are being proposed for IoT devices communication usage. One of the distinguishable IoT machine-to-machine communication protocols is Message Queuing Telemetry Transport (MQTT) protocol. However, as per the authors best knowledge, there are no available IDS datasets that include MQTT benign or attack instances and thus, no IDS experimental results available.

In this paper, the effectiveness of six Machine Learning (ML) techniques to detect MQTT-based attacks is evaluated. Three abstraction levels of features are assessed, namely, packet-based, unidirectional flow, and bidirectional flow features. An MQTT simulated dataset is generated and used for the training and evaluation processes. The dataset is released with an open access licence to help the research community further analyse the accompanied challenges. The experimental results demonstrated the adequacy of the proposed ML models to suit MQTT-based networks IDS requirements. Moreover, the results emphasise on the importance of using flow-based features to discriminate MQTT-based attacks from benign traffic, while packet-based features are sufficient for traditional networking attacks.

Keywords: IoT · Machine Learning · MQTT · Intrusion Detection

1 Introduction

A large number of Internet of Things (IoT) devices and networks have been utilised over the past years for different usage scenarios [13]. These use-cases include healthcare [4], smart cities [5], supply chain [1] and farming [3]. With this extended use of IoT, new protocols are being deployed [17]. One of the new prominent protocols used for machine-to-machine communication is MQTT [19].

Harsha *et al.* [8] discuss the different protocols used in various IoT networks, which include MQTT. The authors analyse the security risks associated with using MQTT. The authors results show that there are 53396 publicly available and accessible MQTT devices [8]. Their work, alongside the work by Dinculeană and Cheng [7], emphasises on the need for robust detection techniques for MQTT attacks to overcome the security vulnerabilities.

As discussed in [10], IoT Intrusion Detection Systems (IDS) have different requirements due to the uniqueness of the usage scenarios involved. IoT IDSs are required to be flexible, extendable, and built using real or simulated traffic suited for the intended usage [9]. However, publicly available IoT datasets are limited, thus limiting IoT IDS development [9].

In this manuscript, we aim at proposing and evaluating different Machine Learning (ML) based MQTT IDS. The contributions of this paper are as follows:

- Generating a novel IoT -MQTT dataset and releasing it for public consumption.
- Analysing a novel MQTT dataset which includes both benign and attack scenarios.
- Evaluating the significance of using high-level (flow-based) features to build the IDS.
- Assessing the proposed model using six different ML techniques.
- Examining the different needs of MQTT-based versus generic attacks detection, which emphasise the special setup and, thus the needs of MQTT (IoT) networks.

The remainder of this paper is organised as follows; Section 2 discusses the setup used for the dataset generation and provides an overview of the dataset and the extracted features. Section 3 presents the results obtained by applying different ML techniques to detect attacks. Finally, the paper is concluded in Section 4.

2 Dataset

This section provides a description of the dataset gathered by the MQTT sensors simulation. The dataset is published[1] in [12]. The dataset consists of five recorded scenarios; normal operation and four attack scenarios. The attacker performs four attack and each is recorded independently.

[1] `https://ieee-dataport.org/open-access/mqtt-internet-things-intrusion-detection-dataset`

The attack types are:

- Aggressive scan (Scan_A)
- User Datagram Protocol (UDP) scan (Scan_sU)
- Sparta SSH brute-force (Sparta)
- MQTT brute-force attack (MQTT_BF)

The data is acquired using tcpdump. The packets are collected by recording Ethernet traffic and then exporting to pcap files. The following tools were used as follows:

- Virtual machines are used to simulate the network devices.
- Nmap is used for the scanning attacks.
- VLC is used to simulate the camera feed stream.
- MQTT-PWN [2] is used for the MQTT brute-force attack.

Figure 1 visualises the network components. The network consists of 12 MQTT sensors, a broker, a machine to simulate camera feed, and an attacker. During normal operation, all 12 sensors send randomised messages using the "Publish" MQTT command. The length of the messages is different between sensors to simulate different usage scenarios. The messages content is randomly generated. The camera feed is simulated using VLC media player which uses UDP stream. To further simulate a realistic scenario each of the network emulators drop packets with 0.2%, 1%, and 0.13%. During the four attack scenarios recording, the background normal operation was left in action. The operating systems of the different devices are as follows; Tiny Core Linux for the sensors, Ubuntu for the camera & camera feed server, and finally, Kali Linux for the hacker.

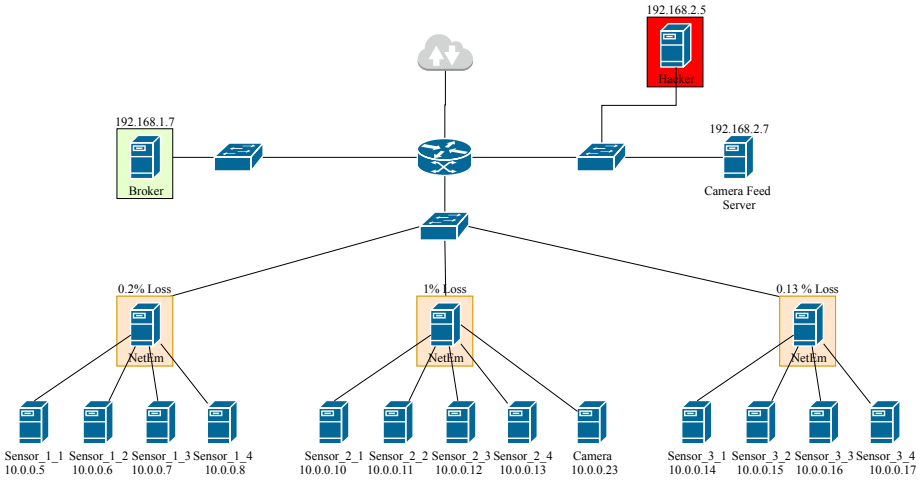


Fig. 1: MQTT Network Architecture [12]

The importance of this dataset is fourfold:

- The dataset simulates a realistic MQTT IoT network in a normal operation scenario.
- The dataset includes both generic networking scanning attacks, as well as, MQTT brute-force attack.
- Researchers can use this dataset to build and evaluate IoT Intrusion Detection Systems.
- The dataset is the first to include MQTT scenarios and attacks data.

The dataset is provided in its raw capture format (.pcap files), as well as processed features [12]. The features represent: (a) packet-based features, (b) Unidirectional-based features, and (c) bidirectional-based features [18]. Each feature set is used exclusively, as discussed in Section 3. The basic packet extracted features are listed in Table 1, fourth column. The feature list for unidirectional and bidirectional is listed in Table 1, columns five and six, respectively. It is important to note that for the bidirectional flows, some features (pointed as *) have two values—one for the forward flow and one for the backward flow. The two features are recorded and distinguished by a prefix "fwd_" for forward and "bwd_" for backward [12]. Furthermore, the distribution of instances is listed in Table 2.

In order to avoid specific features influence, the following features are dropped. These features are source and destination IP addresses, protocol, and MQTT flags. The data is split into 75% and 25% for training and testing, respectively.

Table 1: Features Description

Feature	Data Type	Description	Packet	Uni-flow	Bi-flow
ip_src	Text	Source IP Address	✓	✓	✓
ip_dest	Text	Destination IP Address	✓	✓	✓
protocol	Text	Last layer protocol	✓		
ttl	Integer	Time to live	✓		
ip_len	Integer	Packet Length	✓		
ip_flag_df	Binary	Don't fragment IP flag	✓		
ip_flag_mf	Binary	More fragments IP flag	✓		
ip_flag_rb	Binary	Reserved IP flag	✓		
prt_src	Integer	Source Port	✓	✓	✓
prt_dst	Integer	Destination Port	✓	✓	✓
proto	Integer	Transport Layer protocol (TCP/UDP)		✓	✓
tcp_flag_res	Binary	Reserved TCP flag	✓		
tcp_flag_ns	Binary	Nonce sum TCP flag	✓		
tcp_flag_cwr	Binary	Congestion Window Reduced TCP flag	✓		
tcp_flag_ecn	Binary	ECN Echo TCP flag	✓		

Table 1 continued

Feature	Data Type	Description	Packet	Uni-flow	Bi-flow
tcp_flag_urg	Binary	Urgent TCP flag	✓		
tcp_flag_ack	Binary	Acknowledgement TCP flag	✓		
tcp_flag_push	Binary	Push TCP flag	✓		
tcp_flag_reset	Binary	Reset TCP flag	✓		
tcp_flag_syn	Binary	Synchronization TCP flag	✓		
tcp_flag_fin	Binary	Finish TCP flag	✓		
num_pkts	Integer	Number of Packets in the flow		✓	*
mean_iat	Decimal	Average inter arrival time		✓	*
std_iat	Decimal	Standard deviation of inter arrival time		✓	*
min_iat	Decimal	Minimum inter arrival time		✓	*
max_iat	Decimal	Maximum inter arrival time		✓	*
num_bytes	Integer	Number of bytes		✓	*
num_psh_flags	Integer	Number of push flag		✓	*
num_rst_flags	Integer	Number of reset flag		✓	*
num_urg_flags	Integer	Number of urgent flag		✓	*
mean_pkt_len	Decimal	Average packet length		✓	*
std_pkt_len	Decimal	Standard deviation packet length		✓	*
min_pkt_len	Decimal	Minimum packet length		✓	*
max_pkt_len	Decimal	Maximum packet length		✓	*
mqtt_messagetype	Integer	MQTT message type	✓		
mqtt_messagelength	Binary	MQTT message length	✓		
mqtt_flag_uname	Binary	User Name MQTT Flag	✓		
mqtt_flag_passwd	Binary	Password MQTT flag	✓		
mqtt_flag_retain	Binary	Will retain MQTT flag	✓		
mqtt_flag_qos	Integer	Will QoS MQTT flag	✓		
mqtt_flag_willflag	Binary	Will flag MQTT flag	✓		
mqtt_flag_clean	Binary	Clean MQTT flag	✓		
mqtt_flag_reserved	Binary	Reserved MQTT flag	✓		
is_attack	Binary	1 if the instance represents an attack, 0 otherwise.	x	x	x

* represented as two features in the biflow features file (forward fwd and backward bwd)

Table 2: Dataset Instances Distribution

File Name	pcap file size	Number of Packets		Number of Uni-flow Instances		Number of Uni-flow Instances	
		Benign	Attack	Benign	Attack	Benign	Attack
normal	192.5 MB	1056230 (3.42%)	0	171836 (59.01%)	0	86008 (54.78%)	0
scan_A (aggressive)	16.2 MB	70768	40624 (0.13%)	11560	39797 (13.67%)	5786	19907 (12.68%)
scan_sU (UDP)	41.3 MB	210819	22436 (0.07%)	34409	22436 (7.71%)	17230	22434 (14.29%)
sparta	3.4 GB	947177	19728943 (63.93%)	154175	28232 (9.7%)	77202	14116 (8.99%)

3 Experiments and Results

This section discusses the conducted experiments. Note that the code is available on a GitHub repository [1].

Five-fold cross validation is used to evaluate each experiment. The metrics used for evaluation are as follows [9]: (a) Overall accuracy, as defined in equation 1, such that True Positive (TP) represents the attack instances correctly classified, True Negative (TN) represents the benign instances correctly classified, Positive (P) represents the number of attack instances and Negative (N) represents the total number of benign instance.

$$OverallAccuracy = \frac{TP + TN}{P + N} \tag{1}$$

For each class, Precision, Recall, and F1 Score are computed as shown in Equation 2, Equation 3, and Equation 4, respectively [9]. False Positive (FP) represents benign instances falsely classified as attack and False Negative (FN) represents the attack instances falsely classified as benign.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = \frac{2TP}{2TP + FP + FN} \tag{4}$$

Finally, the weighted average for precision, recall, and F1 score is calculated to demonstrate the overall performance.

Six ML techniques are employed for the classification purpose. The ML techniques are: Logistic Regression (LR), Gaussian Naïve Bayes (NB) , k-Nearest

[1] `https://github.com/AbertayMachineLearningGroup/MQTT_ML`

Neighbours (k-NN) , Support Vector Machine (SVM) , Decision Trees (DT) and Random Forests (RF) [21] [14] [16] [20] [15] [6] [11].

Table 3 details the overall accuracy of each of the ML techniques with packet, unidirectional and bidirectional features. It can be observed the performance rise accompanying flow-based features, both unidirectional and bidirectional. This rise could further be visualised in Figure 2.

Table 3: Overall detection accuracy

	Features		
	Packet	**Unidirectional**	**Bidirectional**
LR	78.87%	98.23%	99.44%
k-NN	69.13%	99.68%	99.9%
DT	88.55%	99.96%	99.95%
RF	65.39%	99.98%	99.97%
SVM (RBF Kernel)	77.4%	97.96%	96.61%
NB	81.15%	78%	97.55%
SVM (Linear Kernel)	66.69%	82.6%	98.5%

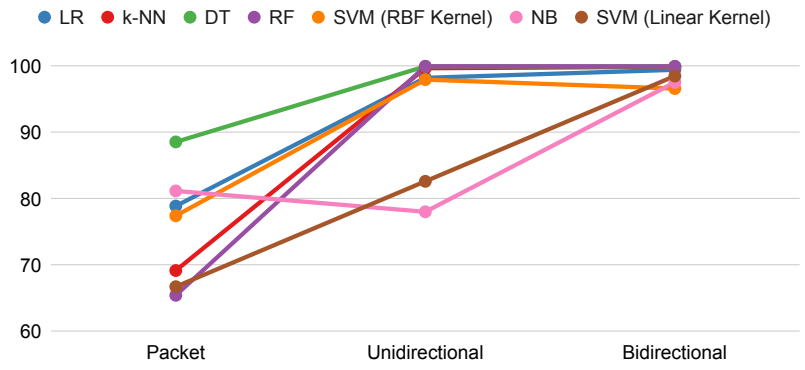


Fig. 2: Overall detection accuracy trend using different ML techniques

To further analyse the results, Table 4, Table 5 and, Table 6 show the detailed precision, recall, and F1-score for each of the classifiers. Each classifier is represented as a sub-table. Similar to Table 3, it is observed that the flow-based features are strongly enhance the results.

Furthermore, it is recognised that the Benign and the MQTT-BF attack are the two classes benefiting from flow features. This is reasoned by the fact that in IoT networks normal/benign operations are usually uncomplicated, due to their usage, requirements, and the nature of data of interest. Therefore, generic

attacks are quite distinctive. However, MQTT-based attacks have similar characteristics to benign MQTT communication. Since MQTT-based attacks rely on the available MQTT communication commands (i.e., publish, subscribe, etc), it is challenging to discriminate attacks from normal operations where the same commands are used. As a result, packet-based features in all the ML techniques were not suitable for benign and MQTT-BF classification. This observation could further be observed in the trends charts for benign class, MQTT_BF class, and weighted average metrics in Figure 3, Figure 4 and, Figure 5.

LR									
	Recall			**Precision**			**F1-score**		
	Packet	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**
Benign	0%	100%	99.02%	0%	93.33%	98.95%	0%	96.55%	98.99%
Scan_A	86.45%	70.87%	97.25%	98.39%	98.39%	97.21%	92.03%	82.39%	97.2%
Scan_sU	98.21%	98.03%	98.48%	99.34%	95.76%	100%	98.77%	96.88%	99.23%
Sparta	100%	100%	100%	98.22%	100%	100%	99.1%	100%	100%
MQTT_BF	100%	99.25%	99.58%	51.75%	99.82%	99.41%	68.2%	99.53%	99.5%
Weighted Average	78.87%	98.23%	99.44%	70.4%	98.32%	99.44%	72.97%	98.14%	99.44%

k-NN									
	Recall			**Precision**			**F1-score**		
	Packet	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**
Benign	17.43%	99.69%	99.95%	17.42%	98.85%	99.59%	17.43%	99.27%	99.77%
Scan_A	99.99%	99.97%	100%	99.99%	99.85%	99.9%	99.99%	99.91%	99.95%
Scan_sU	99.99%	99.96%	100%	99.99%	99.96%	100%	99.99%	99.96%	100%
Sparta	100%	100%	100%	100%	100%	100%	100%	100%	100%
MQTT_BF	25.84%	99.3%	99.75%	25.85%	99.82%	99.97%	25.84%	99.56%	99.86%
Weighted Average	69.13%	99.68%	99.9%	69.13%	99.68%	99.9%	69.13%	99.68%	99.9%

DT									
	Recall			**Precision**			**F1-score**		
	Packet	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**
Benign	69.29%	99.92%	99.88%	69.39%	99.92%	99.91%	69.34%	99.92%	99.9%
Scan_A	100%	100%	100%	99.98%	99.95%	99.9%	99.99%	99.97%	99.95%
Scan_sU	99.98%	99.91%	100%	100%	100%	100%	99.99%	99.96%	100%
Sparta	100%	100%	100%	100%	100%	100%	100%	100%	100%
MQTT_BF	72.56%	99.95%	99.93%	72.47%	99.95%	99.93%	72.51%	99.95%	99.93%
Weighted Average	88.55%	99.96%	99.95%	88.55%	99.96%	99.95%	88.54%	99.96%	99.95%

Table 4: 5-fold cross validation

RF									
	Recall			**Precision**			**F1-score**		
	Packet	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**
Benign	9.34%	99.96%	99.93%	8.99%	99.94%	99.95%	9.16%	99.95%	99.94%
Scan_A	100%	100%	100%	99.98%	99.95%	99.95%	99.99%	99.97%	99.98%
Scan_sU	99.98%	99.91%	99.96%	99.99%	100%	100%	99.99%	99.96%	99.98%
Sparta	100%	100%	100%	100%	100%	100%	100%	100%	100%
MQTT_BF	15.15%	99.96%	99.97%	15.69%	99.98%	99.96%	15.42%	99.97%	99.97%
Weighted Average	65.39%	99.98%	99.97%	65.44%	99.98%	99.97%	65.41%	99.98%	99.97%

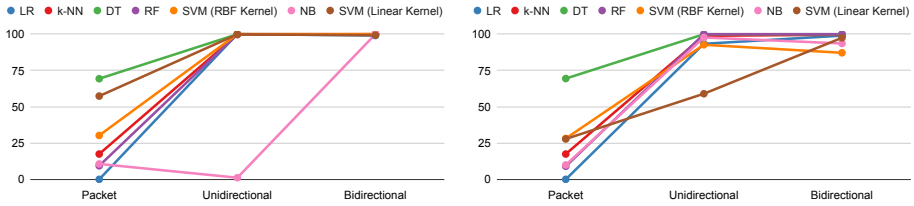
SVM (RBF Kernel)									
	Recall			**Precision**			**F1-score**		
	Packet	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**
Benign	30.23%	100%	100%	28.13%	92.67%	87.13%	28.8%	96.19%	93.12%
Scan_A	83.8%	70.16%	42.13%	99.99%	96.18%	99.88%	91.18%	81.13%	59.22%
Scan_sU	92.33%	99.96%	100%	99.74%	93.01%	94.34%	95.89%	96.36%	97.09%
Sparta	100%	100%	100%	91.17%	100%	100%	95.38%	100%	100%
MQTT_BF	72.42%	98.44%	98.3%	53.56%	100%	100%	59.53%	99.22%	99.14%
Weighted Average	77.4%	97.96%	96.61%	74.35%	98.05%	97.02%	74.89%	97.87%	96.15%

NB									
	Recall			**Precision**			**F1-score**		
	Packet	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**	**Packet**	**Uni**	**Bi**
Benign	10.62%	1.13%	99.96%	9.9%	97.68%	93.56%	10.25%	2.24%	96.65%
Scan_A	100%	99.25%	66.41%	99.23%	18.28%	100%	99.61%	30.88%	79.81%
Scan_sU	99.52%	97.76%	100%	100%	98.79%	98.52%	99.76%	98.27%	99.25%
Sparta	99.84%	100%	100%	100%	100%	100%	99.92%	100%	100%
MQTT_BF	90.27%	97.78%	100%	53.15%	100%	97.05%	65.84%	98.88%	98.5%
Weighted Average	81.15%	78%	97.55%	73.29%	95.43%	98.37%	75.99%	75.26%	97.77%

Table 5: 5-fold cross validation

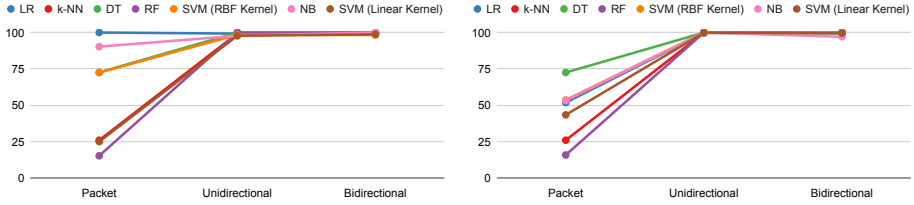
SVM (Linear Kernel)									
	Recall			Precision			F1-score		
	Packet	Uni	Bi	Packet	Uni	Bi	Packet	Uni	Bi
Benign	57.34%	99.84%	99.26%	27.8%	58.95%	97.45%	37.38%	73.82%	98.32%
Scan_A	83.28%	68.23%	84.1%	70.42%	70.35%	93.44%	69.7%	67.5%	87.01%
Scan_sU	78.13%	60.31%	97.76%	75.8%	70.71%	93.77%	76.92%	61.91%	95.27%
Sparta	87.64%	60.37%	99.99%	97.62%	99.94%	100%	89.89%	74.61%	99.99%
MQTT_BF	24.89%	97.79%	98.71%	43.3%	99.89%	99.55%	20.84%	98.83%	99.13%
Weighted Average	66.69%	82.6%	98.5%	65.42%	88.9%	98.66%	60.4%	82.42%	98.46%

Table 6: 5-fold cross validation



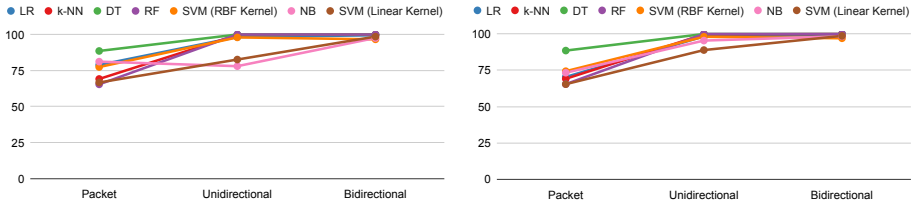
(a) Recall (b) Precision

Fig. 3: Benign Class Trends



(a) Recall (b) Precision

Fig. 4: MQTT_BF Class Trends



(a) Recall (b) Precision

Fig. 5: Weighted Average Trends

4 Conclusion and Future Work

This work aims at exploring the different challenges and requirements for building IDS for IoT models, using an MQTT network as a case study. This paper evaluates six different ML techniques as attack classifiers. A simulated MQTT network was used for data collection to simulate a real-life setup. Using the

dataset raw pcap files, three features levels were extracted; packet, unidirectional, and bidirectional features. Each feature level is used independently in the experiments. The experiments highlighted that generic networking attacks are easily discriminated from normal operation due to their distinguished behaviour and patterns compared to the IoT setup. However, MQTT-based attacks are more complicated and can easily mimic benign operation.

The experimental results further demonstrated that the flow-based features are better suited to discriminate between benign and MQTT-based attacks due to their similar characteristics. The weighted average recall rose from $\sim 75.31\%$ for packet-based features to $\sim 93.77\%$ and $\sim 98.85\%$ for unidirectional and bidirectional flow features, respectively. While the weighted average precision rose from $\sim 72.37\%$ for packet-based features to $\sim 97.19\%$ and $\sim 99.04\%$ for unidirectional and bidirectional flow features. Therefore, the experiments emphasised on the special challenges faced by IoT IDS, based on their custom communication patterns. The challenges were demonstrated through the difficulties to differentiate MQTT-based attacks from normal operations.

References

1. Abdel-Basset, M., Manogaran, G., Mohamed, M.: Internet of things (IoT) and its impact on supply chain: A framework for building smart, secure and efficient systems. Future Generation Computer Systems **86**, 614–628 (2018)
2. Abeles, Daniel; Zioni, M.: MQTT-PWN, IoT exploitation & recon framework. `https://mqtt-pwn.readthedocs.io/en/latest/index.html` (2018), (Accessed on 02/2020)
3. Ahmed, N., De, D., Hussain, I.: Internet of things (IoT) for smart precision agriculture and farming in rural areas. IEEE Internet of Things Journal **5**(6), 4890–4899 (2018)
4. Alansari, Z., Soomro, S., Belgaum, M.R., Shamshirband, S.: The rise of internet of things (IoT) in big healthcare data: Review and open research issues. In: Saeed, K., Chaki, N., Pati, B., Bakshi, S., Mohapatra, D.P. (eds.) Progress in Advanced Computing and Intelligent Engineering. pp. 675–685. Springer Singapore, Singapore (2018)
5. Arasteh, H., Hosseinnezhad, V., Loia, V., Tommasetti, A., Troisi, O., Shafie-khah, M., Siano, P.: Iot-based smart cities: a survey. In: 2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC). pp. 1–6. IEEE (2016)
6. Barber, D.: Bayesian reasoning and machine learning. Cambridge University Press (2012)
7. Dinculeană, D., Cheng, X.: Vulnerabilities and limitations of MQTT protocol used between IoT devices. Applied Sciences **9**(5), 848 (2019)
8. Harsha, M.S., Bhavani, B.M., Kundhavai, K.R.: Analysis of vulnerabilities in MQTT security using shodan API and implementation of its countermeasures via authentication and ACLs. In: 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI). pp. 2244–2250 (2018)
9. Hindy, H., Brosset, D., Bayne, E., Seeam, A.K., Tachtatzis, C., Atkinson, R., Bellekens, X.: A taxonomy of network threats and the effect of current datasets on intrusion detection systems. IEEE Access **8**, 104650–104675 (2020)

10. Hindy, H., Hodo, E., Bayne, E., Seeam, A., Atkinson, R., Bellekens, X.: A taxonomy of malicious traffic for intrusion detection systems. In: 2018 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (Cyber SA). pp. 1–4 (2018)
11. Hindy, H., Brosset, D., Bayne, E., Seeam, A., Bellekens, X.: Improving SIEM for critical SCADA water infrastructures using machine learning. In: Katsikas, S.K., Cuppens, F., Cuppens, N., Lambrinoudakis, C., Antón, A., Gritzalis, S., Mylopoulos, J., Kalloniatis, C. (eds.) Computer Security. pp. 3–19. Springer International Publishing, Cham (2019)
12. Hindy, H., Tachtatzis, C., Atkinson, R., Bayne, E., Bellekens, X.: MQTT-IOT-IDS2020: MQTT internet of things intrusion detection dataset. IEEE Dataport (2020). https://doi.org/10.21227/bhxy-ep04
13. Hodo, E., Bellekens, X., Hamilton, A., Dubouilh, P.L., Iorkyase, E., Tachtatzis, C., Atkinson, R.: Threat analysis of IoT networks using artificial neural network intrusion detection system. In: 2016 International Symposium on Networks, Computers and Communications (ISNCC). pp. 1–6. IEEE (2016)
14. Jr, D.W.H., Lemeshow, S., Sturdivant, R.X.: Applied logistic regression, vol. 398. John Wiley & Sons (2013)
15. Larose, D.T., Larose, C.D.: Discovering knowledge in data: an introduction to data mining. John Wiley & Sons (2014)
16. Lior, R.: Data mining with decision trees: theory and applications, vol. 81. World scientific (2014)
17. Nogues, M., Brosset, D., Hindy, H., Bellekens, X., Kermarrec, Y.: Labelled network capture generation for anomaly detection. In: International Symposium on Foundations and Practice of Security. pp. 98–113. Springer (2019)
18. Ring, M., Wunderlich, S., Grüdl, D., Landes, D., Hotho, A.: A Toolset for Intrusion and Insider Threat Detection, pp. 3–31. Springer International Publishing, Cham (2017). https://doi.org/10.1007/978-3-319-59439-2_1, https://doi.org/10.1007/978-3-319-59439-2_1
19. Stanford-Clark, A., Truong, H.L.: Mqtt for sensor networks (MQTT-SN) protocol specification. International business machines (IBM) Corporation version **1**, 2 (2013)
20. Steinwart, I., Christmann, A.: Support vector machines. Springer Science & Business Media (2008)
21. VanderPlas, J.: Python data science handbook: Essential tools for working with data. "O'Reilly Media, Inc." (2016)