# Industrial IoT Security Monitoring and Test on Fed4Fire+ Platforms

Diego Rivera[1], Edgardo Montes de Oca[1], Wissam Mallouli[1], Ana Cavalli[2], Brecht Vermeulen[3], and Matevz Vucnik[4]

[1] Montimage, 39 rue Bobillot, 75013 Paris, France
{first.last}@montimage.com
[2] Telecom SudParis, 9 rue Charles Fourier, 91011 Evry, France
ana.cavalli@telecom-sudparis.eu
[3] IMEC, Remisebosweg 1, 3001 Leuven, Belgium
brecht.vermeulen@ugent.be
[4] Jozef Stefan Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia
matevz.Vucnik@ijs.si

**Abstract.** This paper presents the main results of the experiments conducted using the MMT-IoT security analysis solution run on a IoT Fed4Fire+ platform (Virtual Wall - w.iLab proposed by IMEC, Belgium). MMT is a monitoring framework developed by Montimage, and MMT-IoT is the tool that allows monitoring and analysing the security and performance of IoT networks. The results obtained concern two principal advancements. First, the adaptations made to deploy MMT-IoT on the IoT platform in order to run the tool on the platform's IoT devices. Second, the deployment of the software allowed us to run preliminary tests on the selected platform for performing initial validation and scalability tests on this real environment. To this end, Montimage defined and implemented three test scenarios related to security and scalability with 1 or more clients. These results will be used to prepare a new experimentation phase involving also another Fed4Fire+ platform (LOG-a-TEC proposed by IJS, Slovenia).

**Keywords:** Monitoring · Testing · IoT · Industrial Applications · Fed4Fire · MMT

## 1 Introduction

Internet of things (IoT) is a concept that describes a network of interconnected devices capable of interacting with other devices, human beings and its surrounding physical world to perform a variety of tasks [1].

Modern IoT devices make use of sensors (e.g., accelerometer, gyroscope, microphone, light sensor, etc.) [6] to detect any changes in their surrounding and take necessary actions to improve any ongoing task efficiently [15]. The increasing popularity and utility of IoT devices in divergent application domains made the IoT industry to grow at a tremendous rate. According to a report by Business Insider [3], 30 billion devices will be connected to the Internet by 2020.

These devices can provide new functionality in different domains, but can also be used as vehicles to launch attacks (examples can be found for instance in [13, 9, 4, 7, 10, 11]).

The challenge of security monitoring on IoT network arises when trying to detect these attacks on devices that have strict resource limitations. Furthermore, existing centralised monitoring techniques (Intrusion Detection and Prevention Systems) cannot handle the large amounts of data that needs to be analysed, and have been designed to work on the edge of the networks and cannot cope with IoT networks that lack clear boundaries. In this paper, we present MMT-IoT, a security tool intended for addressing the requirements for security monitoring on IoT networks, and its application in industrial settings. MMT-IoT allows capturing IoT network traffic near the IoT devices and analyses them to detect potential attacks. In this work we take advantage of the industrial Fed4Fire testbed to deploy MMT-IoT on a near-to-real-life scenario, validate its security detection properties and perform initial scalability tests.

The rest of the paper is organized as follows. Section 2 presents the general architecture of the MMT-IoT tool. Section 3 presents the description of the Fed4Fire platforms that were used for the experimentation. Section 4 presents the methodology followed for the deployment and experiments on the Fed4Fire testbed, as well as the results obtained. Finally, Section 5 presents the conclusions and future work.

## 2 Montimage Monitoring Tool (MMT) for IoT Networks

The Montimage Monitoring Tool (MMT) [8] is a modular monitoring framework that allows detecting behavior, security and performance incidents based on a set of formal properties (written in XML) and embedded functions (written in C or any script or interpreted language). MMT allows real-time data capture, metadata extraction, correlation of data from different sources (i.e. network, applications traces and logs, operating systems), complex event processing, and distributed analysis. It uses temporal logic to detect given security properties (expected or anomalous), and statistical and machine learning-based analysis for detecting more sophisticated activities and behaviour. It is relatively easy to extend by adding new: i) properties and embedded functions; ii) plugins for parsing any structured message; iii) new dashboards for visualising data, statistics and alarms; and, iv) instructions to trigger reactions (e.g. mitigation or blocking of attacks).

In order to correctly adapt this approach – designed initially for traditional Ethernet networks – to IoT networks, it was required to split the network extractor (sniffer) in two parts: *the MMT-IoT Sniffer* (a Contiki-based IoT device), and the *MMT-IoT Bridge* (a Linux-based tool). The former is the IoT endpoint that is in charge of sniffing the packets and forwarding them – using a USB line – to a more powerful machine. The latter recovers the transferred packets from the USB line and injects them (encapsulated using the ZEP protocol) in the loopback interface of the machine, making the packets ready for analysis by the

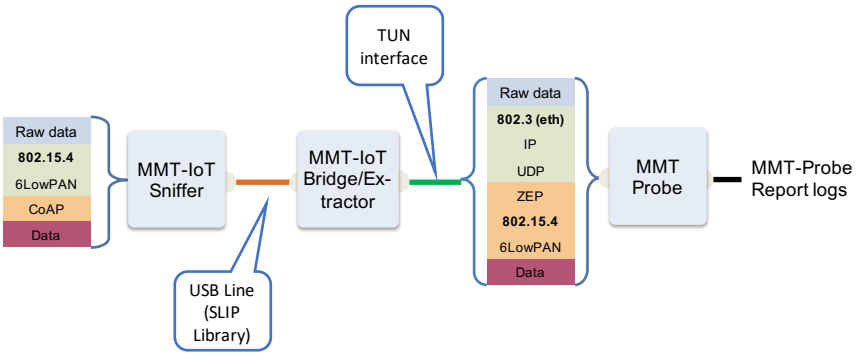


**Fig. 1.** General Architecture of the MMT-IoT solution.

MMT-Probe and MMT-Security. Figure 1 summarises the general architecture of the solution.

Concerning the MMT-IoT Sniffer, the implementation of this architecture was achieved by introducing modifications in the network drivers to make the sniffing feature work. Such modifications were focused in three main axes:

- *Radio driver in promiscuous mode*: This modification was done to avoid dropping of packets by the Contiki kernel.
- *Avoid dropping packets with bad checksum*: By default, the radio driver reads the packet and checks the CRC to detect potential transmission failures. If this check fails, the packer is discarded to avoid processing a mis-formatted packet and save energy. This behaviour was changed, since a sniffing solution must extract all the packets on the medium whether they are correct or not.
- *Insertion of callbacks to redirect the received packet*: A sniffer is a passive network element, therefore, once the packet is received on the radio driver layer, it is transferred via callbacks directly to the application layer. This behaviour bypasses the Contiki network processing and redirects the packets immediately using the USB line, saving energy in the sniffer device. The structure of the inserted callbacks is depicted in Figure 2.

Finally, the MMT-IoT Bridge is responsible for capturing the packets sent through the USB line and making them available for the security analysis performed by the MMT-Probe and MMT-Security; both part of the MMT software. This security analysis is performed by a set of security rules – previously assessed by a network security engineer – which codify the set of network events that need to be correlated for detecting security issues.

It is important to notice that computaiton complexity of detecting an attack is given by the rule itself; complex attacks require more complex rules which correlate a higher number of network events. Considering this, the computation complexity will be taken by MMT-Probe, and not MMT-IoT, which only redirects the traffic to MMT-Probe. This is why neither MMT-IoT Sniffer nor

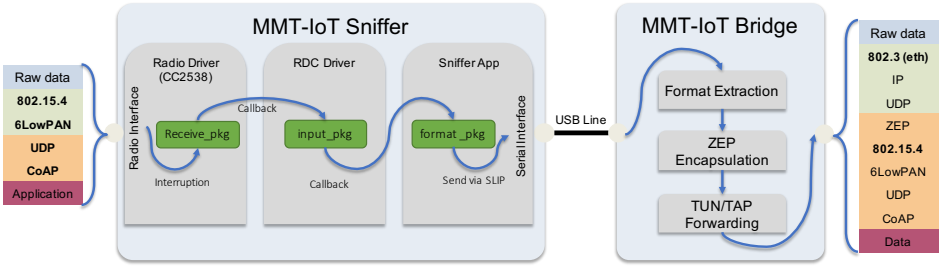


**Fig. 2.** Internal details of the MMT-IoT solution.

MMT-IoT Bridge components contain any complex logic, since, as mentioned before, the security analysis is performed by MMT-Probe.

## 3 Fed4Fire+ Testbeds

Experimentally driven research is considered to be a key factor for growing the European Internet industry. In order to enable this type of RTD activities, a number of projects for building a European facility for Future Internet Research and Experimentation (FIRE) have been launched, each project targeting a specific community within the Future Internet ecosystem. Through the federation of these infrastructures, innovative experiments become possible that break the boundaries of these domains. Besides, infrastructure developers can utilize common tools of the federation, allowing them to focus on their core testbed activities.

In this sense, Fed4FIRE+ is a project under the European Union Programme Horizon 2020, offering the largest worldwide federation of Next Generation Internet (NGI) testbeds. These provide open and reliable facilities supporting a wide variety of different research and innovation communities and initiatives in Europe, including the 5G PPP projects.

The following platforms, LOG-a-TEC and Virtual Wall – w.iLab that are part of Fed4FIRE+ where considered. It must be noted that only Virtual Wall – w.iLab was used to perform the experiments described in this paper. in the case of LOG-a-TEC, only a feasibility study was made and the experiments on this platform will be performed at a later stage.

### 3.1 LOG-a-TEC

LOG-a-TEC is proposed by IJS, Slovenia [14]. It is composed of several different radio technologies that enable dense and heterogeneous IoT, MTC and 5G experimentation. Specially developed embedded wireless sensor nodes can host four different wireless technologies and seven types of wireless transceivers. In order to enable different experiments in combined indoor/outdoor environments

using heterogeneous wireless technologies, the testbed is deployed within JSI's premises and outside in the surrounding park and on the walls of the buildings.

The feasibility of using this platform to carry out experiments has been validated and a new experimentation phase will allow performing the scenarios described and demonstrate the genericity of the monitoring solution.

### 3.2 Virtual Wall – w.iLab

The w.iLab platform [5] is an IoT and 5G emulation testbed that allows running experiments on nodes on real IoT deployments. This platform was designed by the IMEC, Belgium. It provides "bare metal" access to its nodes, i.e., it gives root access to physical machines that will be used to run the experiment. This allows the experimenter to have full control of the nodes on the testbed. The deployment of the MMT-IoT and MMT-Probe software and the execution of the tests are performed remotely without requiring major interventions from the operators. For this, we created credentials on the iMinds platform and performed a reservation of the Intel NUC nodes from the "Datacenter" floor of the platform. The jFED-Experimenter tool was required to design an experiment to access these nodes.

## 4 Experimental Evaluation

### 4.1 Methodology

Considering these testbeds, we used the w.iLab platform to deploy the MMT-IoT Sniffer and the MMT-Probe solutions. In this way, we were able to use the w.iLab t.1 platform to evaluate the scalability of these by overloading them. By performing the extraction of the packets from an IoT network, this experimentation pursues two principal sub-objectives: (1) perform an initial DPI-based security analysis on an IoT network traffic; and (2) determine the maximum throughput a single instance of MMT-IoT Sniffer can handle.

To achieve these objectives, we deployed a set of IoT devices as shown in Figure 3. In this deployment we used 3 types of devices:

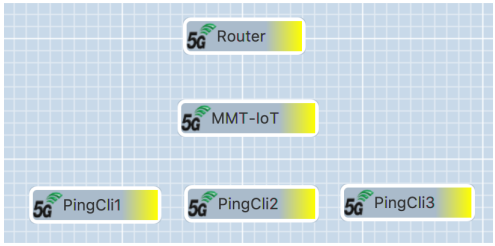


**Fig. 3.** Deployment of the MMT-IoT Solution of the w.iLab platform.

- Ping Client: An emulated IoT sensor programmed to attack the server. For the emulation purposes we used a client that performs a "ping" to the IoT router, however in real life a client can be any device generating some type of traffic.
- IoT Router: A gateway running a routing protocol to allow communications within the IoT network.
- MMT-IoT: A node running the Montimage software under test.

We used the deployment described above to perform initial validation and scalability tests in scenarios that contain respectively 1, 2, 3 malicious clients. We used these configurations to pursue both objectives previously mentioned: (1) the security analysis validation, by means of determining the number of detected attacks; and (2) the scalability of the MMT-IoT solution, by means of analyzing the number of extracted packets on each scenario. This latter aims to be a first test of the scalability of the MMT-IoT software, aiming to determine the amount of information an IoT sniffer is capable to handle.

To deploy the testing scenarios we used the nodes provided by the w.iLab platform, each one composed of a Linux machine with two Zolertia Re-Mote IoT nodes. On each node we used the Zolertia Re-mote nodes to install the corresponding device type (in form of an IoT firmware) and generate the test traffic. Additionally, we installed the MMT-IoT Bridge, MMT-Probe and MMT-Security software on the MMT-IoT Linux machine. This was done in order to read the packets extracted by the IoT sniffer and perform the security analysis on the same node.

The Ping Client IoT sensors were configured to trigger the attack every 10 seconds. At each triggering, the client sent a burst of 10 ICMP ping packets equally spaced within a second. Additionally, an RPL router image was deployed in the "IoT-Router" machine in order to allow packets to flow through the network. All the MMT software was deployed in the MMT-IoT machine, including the MMT-IoT sniffer (in the Zolertia remote connected to that node), the MMT-IoT Bridge (running on the same NUC machine) and the MMT-Probe (also running on the NUC machine). This latter was the component in charge of analyzing the extracted packets and performing attack detection according to a rule previously defined: "we should not allow more than 2 ICMP ping packets per second on an IoT network". This rule comes to the fact that in IPv6 network (and particularly in 6LowPAN networks) the ICMP traffic (and specifically the ping packets) is important to keep the network running. In this sense, the rule allows a fair amount of ICMP packets run through the network without raising an attack alert. This is done to reduce the number of false positives detected by MMT. Using this rule, MMT-Probe was capable of detecting the occurrence of three or more ICMP packets as an attack, generating a report in the MMT-Probe logs.

Each scenario was executed continuously during 5 minutes, in order to generate enough traffic for later analysis. The packets extracted with MMT-IoT Sniffer (using the tcpdump tool) and the MMT-Probe logs are used to check the number of detected attacks in the scenario.

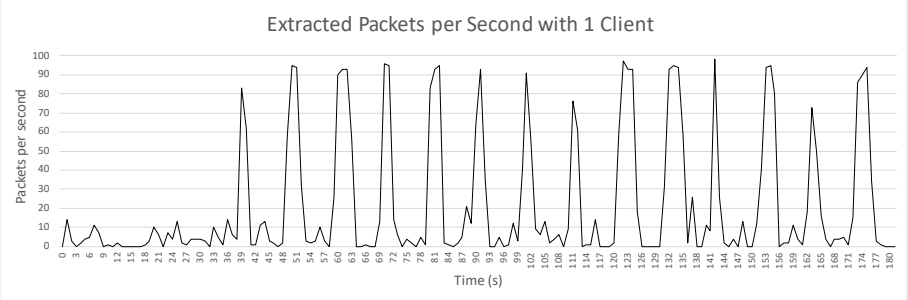


**Fig. 4.** Throughput extracted using MMT-IoT and 1 client.

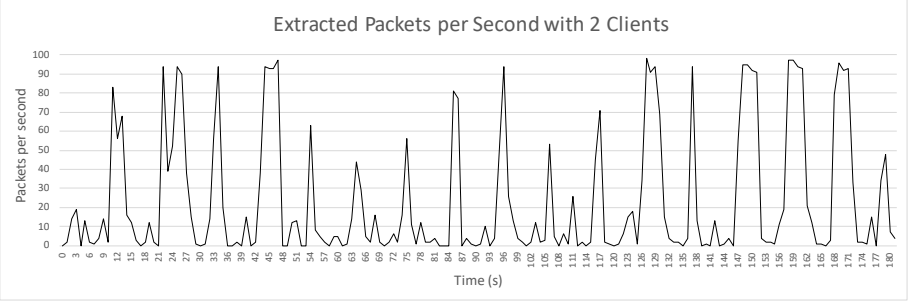


**Fig. 5.** Throughput extracted using MMT-IoT and 2 clients.

### 4.2 Results and Discussion

Figures 4, 5 and 6 show the results of the execution of the three scenarios. In this figure one can observe peaks each 10 seconds. These peaks correspond to the automatic triggering of the attacks, i.e. they show the moment when the clients started to send the ICMP ping packets. In these particular instances we observe a raise in the extracted traffic since there was more data available to be processed. In the 3-clients scenario we see that after 3 minutes of execution the peaks appear more often. We conjecture that this behaviour is due to some type of "desynchronization" between the three clients, and the different attacks appear more frequently.

An interesting observation is the limit of the extracted packets per second. Despite the fact that in the scenario we add more and more clients, and thus more traffic, the maximum number of packets extracted remained practically the same: around 95 packets per second. This opens the possibility of performing experiments to answer the following questions: "How does the packet size impact the number of packets extracted by MMT-IoT?" and "given the MTU of the IoT network, what is the upper limit of the throughput extracted by MMT-IoT?"

Finally, by analysing the logs of the MMT-Probe it was possible to count the number of attack detected. In the scenario with 1 attacking client, MMT-Probe

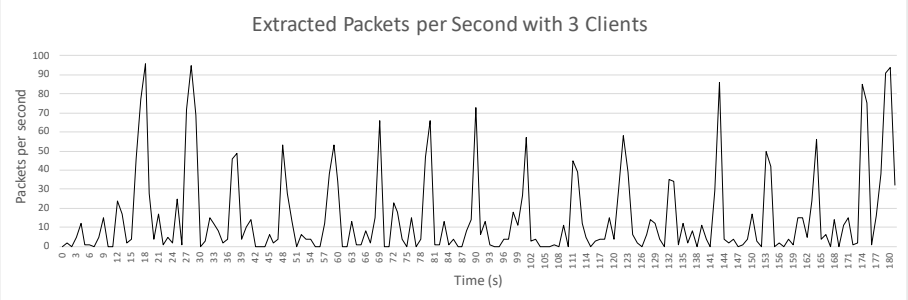


**Fig. 6.** Throughput extracted using MMT-IoT and 3 clients.

detected 183 attacks; with 2 clients, 1046; and with 3 clients, 968. These numbers allow us to validate the applicability of the MMT solution in the IoT networks. In the case of a single attacker, MMT-Probe was capable of analysing the packets extracted by the MMT-IoT Sniffer and detect a simple security threat inside an IoT network.

## 5 Conclusions and Future Work

This paper presented the MMT-IoT tool and its deployment on the Fed4Fire+ testbed. It also presented the results of the feasibility and preliminary tests performed on the Virtual Wall–w.iLab platform. These tests allowed us to validate a proof-of-concept version of MMT-IoT on a real IoT environment. In particular, they allowed increasing the Technology Readiness Level of the tool and the added value of a future product.

It is important to note that even though this paper aimed performing initial feasibility analysis of the scalability issues, the preliminary results allowed us to draw promising conclusions about the future of the tool. In particular, Montimage will aim extending this study in order to clarify how the size of the IoT packets influences the extracted throughput and experiment other more sophisticated attacks. Our first analysis point out that these experiments would allow us to identify potential optimisations in the MMT-IoT sniffer and improve the detection algorithms, aiming to increase the value of the tool and gaining competitive advantage over other similar products such as Bastille's Enterprise IoT Security [2] that uses Bayesian statistics to identify anomalies, and Pwnie Express' Pulse IoT Security Platform [12] that performs device discovery to detect rogue devices, vulnerability scans and policy-infringing connections.

As a future work, we prepare a new experimentation phase that will involve two Fed4Fire platforms: LOG-a-TEC and w.iLab.

## References


1. Bari, N., Mani, G., Berkovich, S.: Internet of things as a methodological concept. IEEE Fourth International Conference on Computing for Geospatial Research and Application (COM. Geo) pp. 48 – 55 (2013)
2. Basille: Enterprise IoT Security. https://www.bastille.net/product (2019), [Online; accessed on 12/07/2019]
3. Greenough, J.: How the internet of things will impact consumers, businesses, and governments in 2016 and beyond. IEEE 4th International Conference on Distance Learning and Education (ICDLE) (2015), http://www.businessinsider.com/how-the-internet-of-things-market-will-grow-2014-10
4. Hasan, R., Saxena, N., Haleviz, T., et al.: Sensing-enabled channels for hard-to-detect command and control of mobile devices. 8th ACM SIGSAC symposium on Information, computer and communications security pp. 469 – 480 (2013)
5. IMEC: w.ilab. https://doc.ilabt.imec.be/ilabt/wilab/index.html (2018), [Online; accessed on 12/07/2019]
6. Lane, N.D., Miluzzo, E., H. Lu, D.P., Choudhury, T., Campbell, A.T.: A survey of mobile phone sensing. IEEE Communications magazine **48**(9) (2010)
7. Maiti, A., Jadliwala, M., He, J., Bilogrevic, I.: (smart) watch your taps: side-channel keystroke inference attacks using smartwatches. ACM International Symposium on Wearable Computers pp. 27 – 30 (2015)
8. Montimage: MMT (Montimage Monitoring Tool). https://montimage.com/products/MMT_DPI.html (2019), [Online; accessed on 12/07/2019]
9. Nahapetian, A.: Side-channel attacks on mobile and wearable systems. 13th IEEE Consumer Communications & Networking Conference (CCNC) pp. 243 – 247 (2016)
10. Petracca, G., Reineh, A.A., Sun, Y., Grossklags, J., Jaeger, T.: Aware: Preventing abuse of privacy-sensitive sensors via operation bindings. 26th USENIX Security Symposium (2017)
11. Petracca, G., Sun, Y., Jaeger, T., Atamli, A.: Audroid: Preventing attacks on audio channels in mobile devices. 31st ACM Annual Computer Security Applications Conference pp. 181 – 190 (2015)
12. Pwnie: Pulse IoT Security Platform. https://www.pwnieexpress.com/pulse (2019), [Online; accessed on 12/07/2019]
13. Sikder, A.K., Aksu, H., Uluagac, A.S.: 6thsense: A contextaware sensor-based attack detector for smart devices. 26th USENIX Security Symposium pp. 397 – 414 (2017)
14. Vucnik, M., Fortuna, C., Solc, T., Mohorcic, M.: Integrating research testbeds into social coding platforms. European Conference on Networks and Communications (EuCNC) (2018). https://doi.org/https://doi.org/10.1109/EuCNC.2018.8443242
15. Yu, Y., Wang, J., Zhou, G.: The exploration in the education of professionals in applied internet of things engineering. IEEE 4th International Conference on Distance Learning and Education (ICDLE) pp. 74 – 77 (2010)