Analysis of IoT Security Risks based on the exposure of the MQTT Protocol

Daniel Kant [1], *Andreas Johannsen* [1], *Reiner Creutzburg* [2,3]

[1] Technische Hochschule Brandenburg, Department of Business and Management, Magdeburger Str. 50. D-14770 Brandenburg, Germany

[2] Technische Hochschule Brandenburg, Department of Informatics and Media, IT- and Media Forensics Lab, Magdeburger Str. 50, D-14770 Brandenburg, Germany

[3] SRH Berlin University of Applied Sciences, Berlin School of Technology, Ernst-Reuter-Platz 10, D-10587 Berlin, Germany

Email: kantd@th-brandenburg.de, johannse@th-brandenburg.de, creutzburg@th-brandenburg.de, reiner.creutzburg@srh.de

Abstract

Due to the tremendous growth of Internet of Things (IoT) applications - e.g. smart homes, smart grids, smart factories – and the emerging integration into industrial systems, the cyber threat landscape for IoT and IIoT applications is rapidly evolving. Security by Design principles are still widely neglected in the design of IoT devices and protocols. For consumer IoT, the privacy of the applicant can be compromised when devices are inappropriately secured. With regard to Industrial IoT, the usage of insecure IIoT protocols such as MQTT can have a severe impact on the industrial environment such as failure or impairment of production systems. We evaluate the prevalence of exposed IoT and IIoT devices related to the protocol MQTT by means of the search engine Shodan. The approach, design and results of our analysis are summarized in this paper.

Keywords

Internet of Things, Industrial Internet of Things, IoT, IIoT, MQTT, IoT protocol, IIoT protocol, IoT security, IIoT security, MQTT vulnerability

Introduction

The Internet of Things (*abbrev.:* IoT) is the general term for a network of systems or devices connected to each other via the Internet mostly for the purpose of connecting and exchanging data with other systems or devices. Studies assume that between 40 and 75 billion networked IoT devices will be available in 2025 [21] [33]. These can be sensors, refrigerators, washing machines, webcams, but also any application in the field of smart home or medical technology. The term IoT in particular describes the connection of systems or devices for the consumer sector. In an industrial context, connected IoT devices are called Industrial Internet of Things (IIoT). This subsumes terms such as smart factories or industrial robots and enables a clear distinction to be drawn from the IoT world of end consumers (e.g. smart home). IoT devices are used in the manufacturing industry, in agriculture, in hospitals and in the field of health care or energy and resource production. With the help of IIoT, value chains can be digitized, virtualized, analyzed and controlled in real time. The connection of embedded systems into business management processes offers considerable potential for process, product and service optimization across all sectors of industry allowing vast cost reductions [6] [24]. Moreover, the *Industry 4.0* paradigm shift requires the integration of Cyber-Physical Systems (CPS) into industrial production, logistics and services. In the past, the media has reported about various significant IT security incidents in connection with IoT and IIoT, such as the botnet Mirai in 2016 [18]. For the communication within an IoT environment, so-called IoT protocols such as MQTT are widely used. Build to operate under real-time conditions, these protocols have a lightweight header [35]. Security concerns were not considered within the design phase. This paper aims to give an overview about major security flaws related to the MQTT protocol as well as to outline and evaluate the exposure on the Internet by means of the search engine Shodan. Furthermore, we want to address additional required research in the field of MQTT.

IoT and IIoT Security

The originally isolated *operational technology* (OT) and *information technology* (IT) have converged into each other, which represents a vast security challenge [26]. With the advent of IoT, data exchange is supported through all layers of the automation pyramid, not just allowing adjacent layers to communicate [19]. This integration tendency reduces costs and allows business analytics of classic field data for the purpose of optimization. But on the other hand, the increased connectivity can represent a serious attack vector, especially for companies. A study states that at the average enterprise today, over 30% of all network-connected end points are IoT devices - excluding mobile devices [30]. IoT devices are produced for the “mass market” and are generally delivered in an insecure state, mostly with no or insufficient update mechanism. In particular, security aspects are still not sufficiently taken into account in the design and architecture of IoT devices (*Security by Design*). Cyber security incidents can have serious impacts when affecting industrial applications, ranging from data leakage to physical damage to the industrial systems [16]. In severe cases, cyber attacks can inflict physical damage to workers [7], since industrial control components directly handle physical processes in the real world. The primary objective is to avoid the failure or impairment of production and business processes. Availability must be guaranteed permanently: A downtime is unacceptable in the industrial context and can lead to significant financial losses [31]. IoT sensors are constantly sending data. The systematic analysis of (even) encrypted consumer IoT traffic e.g. webcams, smart watches or smart homes - mostly by means of *Big Data* and *Artificial Intelligence (AI)* - can result in compromising the privacy of IoT applicants. [3]. The botnet *Mirai* in 2016 abused IoT devices to perform Distributed Denial of Service (DDoS) attacks. One of the challenges with regard to IoT security are the limited hardware resources of IoT devices which hinders the implementation of common encryption and security standards [10]. IoT security issues arise in the collection, submission, storage as well as processing of IoT data [1]. In 2018, the OWASP (*Open Web Application Security Project*) - a non-profit organization dedicated to improve the security of web applications - released a ranking about the ten most critical vulnerabilities related to IoT [29]. The Top three vulnerabilities associated to IoT are *weak, guessable or hardcoded passwords*, *insecure network services* and *insecure ecosystem interfaces*; all three do have a lack in authentication and authorization in common. Concrete attack scenarios can be applied corresponding to the OWASP Top 10 (see [15]). With the approach in this paper, we want to illustrate how simple MQTT-related IoT devices can be found on the Internet. In particular, we want to determine, how many of the MQTT-related hits found by Shodan are representing a security threat. The MQTT protocol in general as well as in connection to security flaws will be introduced in the next section.

MQTT Protocol

IoT protocols are used to exchange data within an IoT ecosystem. The most commonly used IoT protocols are HTTP, XMPP, AMQP, CoAP and MQTT [27]. When used in an IoT environment, stateless protocols like HTTP require a permanent request and reply mechanism, which represents a challenge for the network traffic as well as for the constrained resources of IoT devices. Due to its lightweight header (2-Byte), which is particularly suitable for real-time applications and low-resource hardware, the Message Queuing Telemetry Transport Protocol (*abbrev.:* MQTT) has become the most common IoT protocol (widely used in version MQTT 3.1.1) [4]. The protocol works at the application layer to handle message-based communication for IoT use cases and utilizes the TCP protocol for transmission [4]. It is a stateful and bidirectional protocol taking advantage of the *report by exception* paradigm, meaning a communication to the server is only conducted when values have changed (to spare bandwidth). MQTT is a client server-based publish-subscribe protocol that is standardized according to *ISO 20922* [14]. Generally, there are two main roles: MQTT server - the so-called MQTT broker - and the MQTT client. The entire communication is managed by the MQTT broker (depicted in Figure 1) establishing a point-to-point connection to every communication partner. The MQTT clients cannot communicate directly. Sending (*publish*) and receiving messages (*subscribe*) are conducted via so-called "*topics*". A *topic* is a string containing the subject alike an URL or API call pattern. These topics are highly hierarchical so that MQTT clients can subscribe to topics or subtopics if desired. A widely used open source MQTT broker is *mosquitto*[1] by the Apache Foundation. Commonly used MQTT clients are *MQTT Lens*[2], *MQTT.fx*[3] and the *MQTT Explorer*[4] (Figure 5).

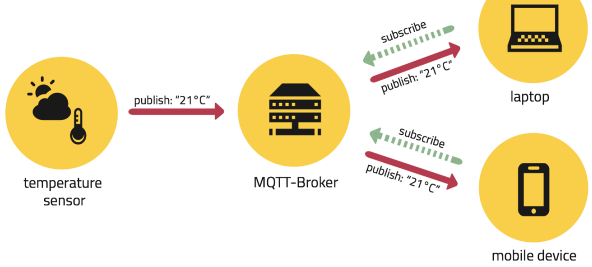


Figure 1. *MQTT communication principle [28]*

MQTT Security Flaws

The light-weighted header of MQTT is one main source for abuse. Even the official manual of MQTT is saying "*MQTT solutions are often deployed in hostile communication environments*" [4]. Further, the manual clarifies, that it is the *"implementer's responsibility to provide appropriate security features"* ending up with the recommendation to use TLS on TCP 8883 [4]. Consequent mechanisms for the authentication of users and devices, the authorization of access to server resources as well as the integrity and privacy of MQTT control packets and application data contained therein needs to be provided by implementations [4]. The official manual names the following security threats with regard to MQTT [4]:

- Devices could be compromised
- Data at rest in Clients and Servers might be accessible
- Protocol behaviors could have side effects (e.g. "timing attacks")
- Denial of Service (DoS) attacks
- Communications could be intercepted, altered, re-routed or disclosed
- Injection of spoofed Control Packets

Shodan search engine

Launched in 2009, Shodan is a search engine that specifically lists those devices that are directly connected to the Internet. This can be a server, a router, or any IP-enabled device. The search engine received broader media attention for the first time in 2013 by an online article of the information platform *CNN Money*, which reported that it succeeded in gaining mass access to traffic control systems as well as other control and service systems in the United States through Shodan [11] [12]. The search engine works similar to *Google*, with the difference that the so-called "*Shodan-Crawlers*" do not only search web pages but also record all ac-

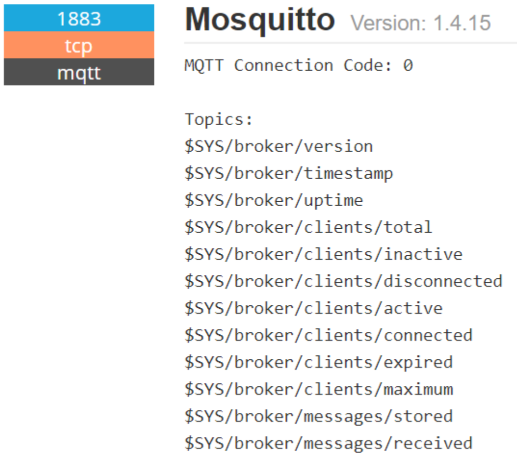
[1] `https://mosquitto.org/` - retrieved: December 2020

[2] `https://chrome.google.com/webstore/detail/mqttlens/hemojaaeigabkbcookmlgmdigohjobjm?hl=de` - retrieved: December 2020

[3] `https://mqttfx.jensd.de/` - retrieved: December 2020

[4] `http://mqtt-explorer.com/` - retrieved: December 2020

cessible servers and their services in order to index them. Shodan uses scanners distributed around the world [32] and searches all IP addresses on the Internet for “*well-known-ports*” used by popular services, and then automatically connects to these services or ports. Servers automatically send data to the users when a connection to the server is initialized. This connection data (“*Banner*”) can contain valuable information and is automatically stored by Shodan in a database. The search engine Shodan can be used to perform cyber security audits on IoT systems. One major advantage: IT Security researchers do not need to scan the related devices directly - the search engine has done this already. With regard to MQTT, Shodan can find a vast variety of MQTT-related ports as well as detailed information concerning the MQTT server (e.g. server version or MQTT connection codes). In Figure 2, an open MQTT port using the Open Source MQTT broker *mosquitto* was found via Shodan disclosing the supported server methods ($SYS-Topic-tree).



1883
tcp
mqtt

Mosquitto Version: 1.4.15

```
MQTT Connection Code: 0

Topics:
$SYS/broker/version
$SYS/broker/timestamp
$SYS/broker/uptime
$SYS/broker/clients/total
$SYS/broker/clients/inactive
$SYS/broker/clients/disconnected
$SYS/broker/clients/active
$SYS/broker/clients/connected
$SYS/broker/clients/expired
$SYS/broker/clients/maximum
$SYS/broker/messages/stored
$SYS/broker/messages/received
```

Figure 2. MQTT broker with accessible $SYS topic tree found in Shodan (request from 12-16-2020)

Objectives, Approach and Methodology

Research question

The MQTT protocol could be used to abuse MQTT-activated devices for a botnet [20]. Approaches by other researchers have shown that serious botnet attacks like Mirai can be adapted and improved [15]. The main goal of our approach is to find MQTT-related hits on the search engine Shodan, especially unprotected or insufficient protected IoT devices with flawed security. Additionally, we want to get a picture about the overall exposure of MQTT on the Internet by means of the search engine. MQTT has been the scope of various research papers. Unlike Al-Alami et al. (2017) [1] or Lundgren (2016) [20], our approach is not to actively scan devices (e.g. by means of vulnerability scanner or pentesting tools), we explicitly only use the functionalities and outputs of the search engine Shodan. In Figure 3, our methodology is shown. In the following section we describe how we proceed.

Research design

The search engine Shodan is the tool we use for our data research. As for the research method, we survey the indexed database Shodan is providing. Depending on how accurate the “*Shodan-Crawlers*” work to find MQTT hits, the search engine can give a picture and tendency about the worldwide exposure. Conveniently, Shodan offers the functionality to download queries and to acquire a report with statistical analysis and diagrams.

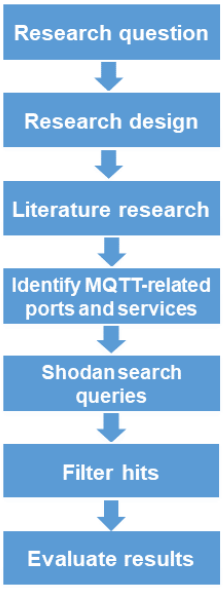


Figure 3. Approach and Methodology of the research paper

Literature research

As for the literature research, our primary aim is to get a picture about known IoT and IIoT device vulnerabilities. At the same time, we search about published scientific papers concerning the MQTT protocol and the search engine Shodan. We reviewed a wide range of various online catalogs (e.g IEEE Explore[5]) with regard to academic literature. Additionally, we also surveyed non-academic literature including articles or technical blogs. For the literature research the following search string was used:

- (“IoT” OR “IoT vulnerabilities” OR “IIoT” OR “IIoT vulnerabilities”) AND (“Shodan”) AND (“MQTT” OR “MQTT Protocol”)

Identify MQTT related ports and services

Based on our literature research, the following important TCP ports and services have been identified in connection with MQTT, which may affect the security of IoT devices. We describe the identified ports and services in the following.

MQTT unencrypt. We figured out the TCP port 1883 to be the most important with regard to the MQTT protocol, since it is the official default port used by the protocol [4]. The port is the reserved port for “non TLS communication” [4].

MQTT over SSL Another TCP port that is directly named in the official MQTT manual is TCP port 8883. It is the encrypted version of MQTT using TLS. The Client certificate version for the open source MQTT broker *mosquitto* is port 8884 [8].

MQTT Websocket There is another possibility to transport the MQTT protocol payloads besides the “conventional” transport protocol TCP and that is via *websockets*. The default port

[5] `https://ieeexplore.ieee.org/` - retrieved: December 2020

is 9001. The concrete port used for MQTT via websocket may vary depending particularly on the used vendor and product [8]. The vendor *HiveMQ*[6] for instance is using port 8000 for MQTT over websockets. *Mosquitto* is using port 8080, 8081 and 80 in connection to websockets [8].

MQTT Websocket over SSL There is a secure version based on TLS when using MQTT based on websockets. The port 9883 is the default SSL websocket port. The mosquitto MQTT broker is using the SSL standard port 443 [8].

Shodan search queries

Based on the identified MQTT-related ports and services we defined concrete statements that had to be converted in order to be applicable for the Shodan filter by means of the official *Shodan Guide* [22]. We conducted the following search queries.

Identified port or service	Shodan search queries
MQTT unencrypt.	mqtt port:“1883”
MQTT over SSL	mqtt port:“8883” mqtt port:“8884”
MQTT Websocket	mqtt port:“9001” mqtt port:“8000” mqtt port:“8080” mqtt port:“8081” mqtt port:“80”
MQTT Websocket over SSL	mqtt port:“9883” mqtt port:”443”
MQTT (no auth)	“MQTT Connection Code: 0”

Table 1. **Concrete Shodan search queries for the identified ports or services in connection to MQTT**

Filter hits

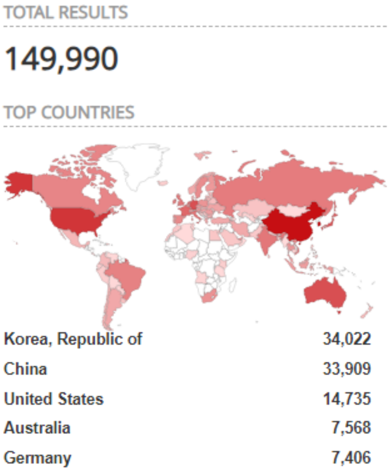
The hits found still need to be filtered, since among them are false positive and false negatives. Though there are standardized *well known ports*, principally everyone can bind a port to any service or protocol. So, we have to ensure (i.e. to filter), that an MQTT protocol version is explicitly recognized. Certainly it had do be considered that there are honeypots active, that willingly open a vast variety of ports. False positives can be matches that may have an open port TCP 1883 but do not deliver the MQTT protocol on this certain port. That is why be explicitly searched the related ports in connection with a “MQTT” string. The hits are downloaded as JSON and CSV file for further filtering and analyzing the attributes.

Evaluate results

In the final phase, we summarize and evaluate our results. The extent of exposure is aggregated and evaluated. With regard to the research question, a decision has to be made, what MQTT hits are considered to be insecure. A MQTT match found in Shodan is considered as insecure, if the hit is vulnerable with regard to the stated security threats of the MQTT manual [4] (see section MQTT Protocol). This concrete subset is defined as *Total insecure MQTT hits*. As for the summarization of the data, we choose a table for the presentation (see Table 2 on page 6). Finally, we compare our results with other researches and publications.

Results

We explicitly only used the functionalities and outputs of the search engine Shodan for investing and evaluating the MQTT-related devices. In particular we figured out that worldwide 149,990 hits were found in connection to the MQTT protocol on the Internet using the unencrypted TCP port 1883 (Figure 4). Most of the devices - precisely 34,022 - are located in South-Korea, followed by China with 33,909 MQTT devices. As for the United States, Shodan does find 14,735 MQTT matches. Furthermore Australia makes it to 7,568 hits. The Top 5 country is Germany, counting 7,406 matches in connection to MQTT port 1883. In particular, those MQTT brokers are critically insecure, since the *$SYS topic tree* can be accessed without authentication (MQTT connection code 0). Worldwide these are 102,437 devices (*MQTT no auth*). Consequently this means, it could be connected to the MQTT broker unauthorized with an open TCP Port 1883, what we have have tested under laboratory conditions (Figure 5). The $SYS-Topic-Tree is issuing extended system information about the MQTT broker as well as about the connected clients. Furthermore, issuing connection code 0 means, publishing and subscribing any topic is allowed i.e. the manipulation of the concrete IoT environment is possible. Finally, summarizing our results, we evaluate a violation against at least one enumerated security threat stated in the MQTT manual, as an insecure MQTT device.



TOTAL RESULTS

149,990

TOP COUNTRIES

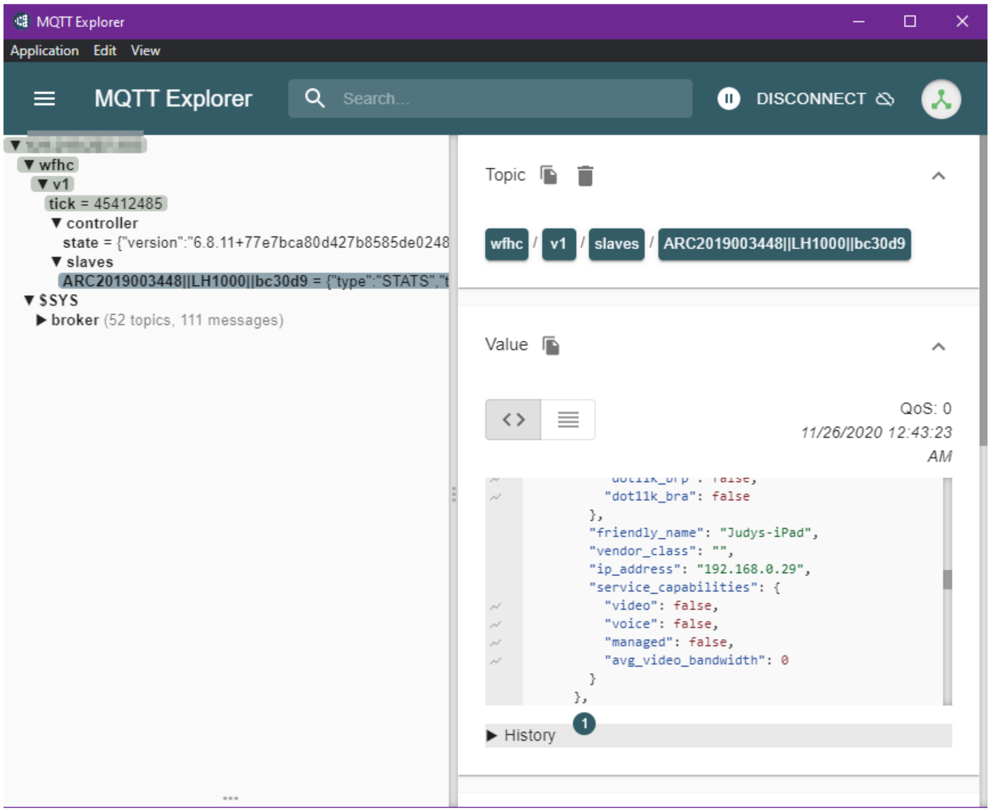
Korea, Republic of	34,022
China	33,909
United States	14,735
Australia	7,568
Germany	7,406

Figure 4. *MQTT matches found in Shodan related to TCP Port 1883 (request from 12-23-2020)*

Evaluation and Discussion

Our goal was to evaluate the prevalence of exposed MQTT devices on the Internet using the meta data issued by the search engine Shodan. In general, with regard to our research, we are significantly beyond all values of related work with 150,180 MQTT hits in Shodan. Andy et al. (2017) found about 24,998 matches [2]. The research of Hron (2018) revealed even more 49,197 MQTT hits - the countries with most MQTT devices were China, the United States and Germany. Hron has shown that 32,888 de-

[6] `https://www.hivemq.com/` - retrieved: December 2020



```
▼ [redacted]
  ▼ wfhc
    ▼ v1
      tick = 45412485
      ▼ controller
        state = {"version":"6.8.11+77e7bca80d427b8585de0248
      ▼ slaves
        ARC2019003448||LH1000||bc30d9 = {"type":"STATS","t
  ▼ $SYS
    ▶ broker (52 topics, 111 messages)
```

wfhc / v1 / slaves / ARC2019003448||LH1000||bc30d9

QoS: 0
11/26/2020 12:43:23 AM

```
          "dot11k_brp": false,
          "dot11k_bra": false
        },
        "friendly_name": "Judys-iPad",
        "vendor_class": "",
        "ip_address": "192.168.0.29",
        "service_capabilities": {
          "video": false,
          "voice": false,
          "managed": false,
          "avg_video_bandwidth": 0
        }
      },
```

Figure 5. *Missing authentication of an accessible MQTT broker using TCP Port 1883 (laboratory environment)*

vices could be accessed without authentication [13]. This corresponds to our subset *MQTT (no auth)*, we have found a higher number of 102,437 MQTT matches had no authentication enabled. With regard to the countries, the most MQTT hits we found in our research via Shodan are located in South-Korea and in China (both are having comparable values). In conclusion, in two years compared to Hron [13], the values for MQTT hits of the Republic of Korea have significantly increased. Concerning MQTT matches in the United States, and in comparison to Hron, the amount of hits have increased, but in the (negative) ranking the USA has fallen to the third rank. The 4th rank is dedicated to Australia, that was not even mentioned in Hron's research. Germany (former 3rd place) is now on the 5th place. Australia and Germany are having similar values. Even though we have not scanned the entire Internet on our own, previous research (about two to four years ago) did not even find close to that quantity. We assume that scanning the entire Internet manually would have resulted in even higher findings. In 2016, Lundgren scanned the Internet and found about 59,000 MQTT hits (with the aid of Shodan he only discovered 17,711) of them [20]. The results are surely getting more accurate, when scanning every single device and not only relying on data from Shodan. Despite the vast address range of IPv4, another issue here is the that the legitimacy of the scans is questionable. It is obvious that the number of IoT devices have heavily increased over the last years, so do the open MQTT ports. But since studies estimated a double-digit amount in Billions of IoT devices in 2020 [33], our hits can only be seen as a subset. With regard to devices that could not be found, it can be assumed that many IoT devices are secured in a cloud infrastructure (VPN) or are using other IoT protocols. We also searched for the websocket version of MQTT, but we have found only insignificant values for those. Although MQTT does support encryption based on SSL/TLS (TCP port 8883), we did not find suitable hits here either.

Summary and Conclusion

Due to the constrained resources - with regard to CPU, memory, network bandwidth - IoT devices take advantage of the MQTT protocol, which is optimized for a low-bandwith environment. In particular, MQTT broker are a lucrative target for cyber attackers. Especially cyber attacks within an IIoT environment can lead to failure, impairment or downtime of production systems. Protection against attacks particularly in the IIoT environment must be a trade-off between security and availability due to real-time demands. IoT and IIoT must ensure that the IT security objectives of confidentiality, availability and integrity are guaranteed. MQTT brokers should never be accessible unsecured over the Internet. Apparently, a vast amount of devices is not operating within a secure cloud (as originally intended). Over 150.000 insecure hits related to MQTT worldwide were found via Shodan. When using the default protocol port (TCP Port 1883), the traffic is completely unencrypted [2], often due to the resource-constrained IoT hardware [17]. Consequently, if no further security mechanisms are deployed, the transferred data from MQTT broker to MQTT client and *vice versa* are vulnerable for sniffing or data manipulation (*replay attacks*) i.e. the integrity

Indentified set	worldwide	KOR	CHN	USA	AUS	GER	Description
MQTT unencrypt.	149,990	34,022	33,909	14,735	7,568	7,406	Set of hits that are related to MQTT AND have an open TCP Port 1883.
MQTT over SSL	-	-	-	-	-	-	Set of hits that are related to MQTT AND have an open TCP Port 8883 OR 8884.
MQTT Websocket	190	2	35	120	-	4	Set of hits that are related to MQTT AND have an open TCP Port 9001 OR 8000 OR 8080 OR 8081 OR 80.
MQTT Websocket over SSL	133	3	4	101	-	-	Set of hits that are related to MQTT AND have an open TCP Port 9883 OR 443.
MQTT (no auth)	102,437	32,262	23,219	7,486	6,794	3,173	Subset of hits that are related to MQTT AND have an open TCP Port 1883 OR 8883 OR 8884 OR 9001 OR 8000 OR 8080 OR 8081 OR 80 OR 9983 OR 443 AND have an accessible *$SYS topic tree* with no authentication mechanism enabled (MQTT Connection Code 0).
Total insecure MQTT hits	**150,180**	**34,024**	**33,944**	**14,855**	**7,568**	**7,410**	Set of all found vulnerable MQTT-related hits

Table 2. **TOP 5 country IoT hits found in SHODAN related to the MQTT protocol (request from 12-23-2020)**

and confidentiality of data packets is not guaranteed. This aspect is shown by Andy 2017 et al. who easily captured MQTT data packets via the application Wireshark [2]. The privacy of IoT consumer products is endangered, since the devices are often sending sensor data permanently. A further major security issue is the flawed authentication. Though MQTT does support authentication when verifying a MQTT client via username and password [4], it is not mandatory [17]. An amount of over 100,000 devices had no authentication activated, thus unauthorized publisher or subscriber can simply connect to the MQTT broker. This discloses deficiencies in the configuration of the devices and represents a significant threat possibly abusing IoT devices for botnets. But even when authentication is activated, it is possible to intercept the credentials sent by the clients (*Man-in-the-Middle*), since the username and password are sent in plaintext [23] and all devices connected to the broker are using the same set of credentials [27]. Another security related issue is the whole topic of authorization for IoT devices. The original MQTT protocol itself does not provide authorization for IoT devices [4] i.e. no fine-grained access control [27]. This means even when IoT devices are authenticated to a MQTT broker, there is no control what actions the clients are allowed to do (e.g. *publish* or *subscribe* to a certain topic). This can result in an unauthorized manipulation of the entire IoT ecosystem. Just as the MQTT manual has stated, the assurance of authentication, integrity and privacy is left to the user [4]. This can be criticized, for both the consumer as well as the industrial usage scenario. Especially laymen have to take precautions and actions when relying purely on the protocol. Regrettably, standard security mechanisms and hardware protections such as TLS are not feasible, supported or activated within the IoT environment [9]. There are approaches how MQTT can be used more securely [34]. IoT appliances are inherently connected to the Internet. This is the major reason that the devices are easily found and indexed in search engines such as Shodan. The trend towards digital transformation and Industry 4.0 will continue. IoT and IIoT devices will proceed to emerge in the next years. Studies assume up to 75 billion connected IoT devices in 2025 [33]. Unfortunately, they are primarily produced for the mass market. Security is neglected in favor of functionality. The broad usage and the lack of security mechanisms make IoT devices a lucrative target for cyber attacks [17]. Therefore, IoT security should be conceived holistically and not be disregarded once a device is delivered. Service providers, developers and vendors are asked to adequately include security aspects already in the development phase (*Security by Design*).

Future Work

As for future work, we outline the following research for the scientific community:

- development of sophisticated and appropriate *Security by Design* methods to ensure security over the entire life cycle of IoT devices
- development and implementation of further applicable security solutions assuring confidentiality and integrity of MQTT data packages (similar to Dinculeana et al. [9] and Niruntasukrat et. al. [27])
- Further research on identity management and fine-grained device access control for IoT (as outlined in [27])
- alternative cryptographic mechanisms to TLS with less overhead operating within a resource-constrained IoT environment
- approach to clearly distinguish between IoT and IIoT devices to get a better picture of exposed IIoT devices and vulnerabilities taking into account the special industrial requirements (e.g. availability and realtime demand)
- Establish holistic IIoT security concepts to securely apply IoT and IIoT devices within corporate networks

- Set up a data-based ranking of vulnerabilities such as OWASP IoT Top 10 [29] especially tailored for IIoT appliances
- Automatic detection of CVE vulnerabilities of an IoT or IIoT device (outlined in [5])
- Research on further open ports and services (besides MQTT) of IoT devices as an additional attack
- Analysis of security aspects based on the MQTT protocol Version 5 [25]
- development of Secure MQTT protocols (outlined in [34])

References

[1] H. Al-Alami; H. Al-Bahadil: Vulnerability Scanning of IoT Devices in Jordan using Shodan, URL: https://www.researchgate.net/publication/321588682_Vulnerability_Scanning_of_IoT_Devices_in_Jordan_using_Shodan (retrieved: 29 Dec 2020), 2017

[2] S. Andy; B. Rahardjo; B. Hanindhito: Attack Scenarios and Security Analysis of MQTT Communication Protocol in IoT System, Proc. EECSI 2017, Yogyakarta, Indonesia, 19-21, URL https://ieeexplore.ieee.org/document/8239179 (retrieved: 29 Dec 2020), 2017

[3] N. Apthorpe; D Reisman; N. Feamster: A Smart Home is No Castle: Privacy Vulnerabilities of Encrypted IoT Traffic, URL: https://arxiv.org/abs/1705.06805 (retrieved: 15 Dec 2020), 2017

[4] A. Banks and R. Gupta: MQTT Version 3.1.1, OASIS, URL: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html (retrieved: 29 Dec 2020), published: 29 Oct 2014

[5] G. Blinowski and P. Piotrowski, URL: https://www.researchgate.net/publication/342588114_CVE_based_classification_of_vulnerable_IoT_systems (retrieved: 28 Dec 2020), 2020

[6] H. Boyes; B. Hallaq; J. Cunningham; T. Watson: The industrial internet of things (IIoT): An analysis framework, URL https://www.sciencedirect.com/science/article/pii/S0166361517307285 (retrieved: 13 Dec 2020), 2018

[7] S. Chhetri; N. Rashid; S. Faezi; M. A. Al Faruque: Security trends and advances in manufacturing systems in the era of industry 4.0. In: Proc. of IEEE/ACM International Conference on Computer-Aided Design (ICCAD), URL: https://ieeexplore.ieee.org/document/8203896 (retrieved: 29 Dec 2020), 2017

[8] S. Cope: MQTT Brokers and Cloud Hosting Guide, Steve Cope, updated: 28 Nov 2020, http://www.steves-internet-guide.com/mqtt-hosting-brokers-and-servers/#list (retrieved: 27 Dec 2020)

[9] D. Dinculeana and X. Cheng: Vulnerabilities and Limitations of MQTT Protocol Used between IoT Devices, URL: https://www.mdpi.com/2076-3417/9/5/848, (retrieved: 29 Dec 2020), 2019

[10] T. Fernandez-Carames and P. Fraga-Lamas: Teaching and Learning IoT Cybersecurity and Vulnerability Assessment with Shodan through Practical Use Cases, URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7309102/ (retrieved: 15 Dec 2020), 2020

[11] D. Goldman: Shodan: The scariest search engine on the Internet, CNN Money, New York, published: 8. April 2013, URL: http://money.cnn.com/2013/04/08/technology/security/shodan/ (retrieved: December 2020), 2013

[12] D. Goldman: The Internet's most dangerous sites, CNN Money, New York, URL: http://money.cnn.com/gallery/technology/security/2013/05/01/shodan-most-dangerous-internet-searches/, (retrieved: December 2020), 2013

[13] M. Hron: Are smart homes vulnerable to hacking?, https://blog.avast.com/mqtt-vulnerabilities-hacking-smart-homes (retrieved: 29 Dec 2020), 2018

[14] "ISO/IEC 20922:2016", Information technology - Message Queuing Telemetry Transport (MQTT) v3.1.1, URL: https://www.iso.org/standard/69466.html (retrieved: 29 Dec 2020), 2016

[15] X. Jiang; M. Lora; S. Chattopadhyay: An Experimental Analysis of Security Vulnerabilities in Industrial IoT Devices, URL: https://dl.acm.org/doi/10.1145/3379542 (retrieved: 29 Dec 2020), 2020

[16] D. Kant; R. Creutzburg; A. Johannsen: Investigation of risks for critical infrastructures due to the exposure of SCADA systems and industrial controls on the Internet based on the search engine Shodan, in: IS&T International Symposium on Electronic Imaging 2020 Mobile Devices and Multimedia: Enabling Technologies, Algorithms, and Applications 2020, Society for Imaging Science and Technology. URL: https://doi.org/10.2352/ISSN.2470-1173.2020.3.MOBMU-253 (retrieved: 5 Jan 2021). 2020

[17] M.Kofler; K. Gebeshuber; T. Hackner; P. Kloep; A. Zingsheim; M. Widl; F. Neugebauer; R. Aigner; S. Kania: Hacking & Security: Das umfassende Hacking-Handbuch, Rheinwerk Computing, 1st Edition, 2018

[18] C. Kolias; G. Kambourakis; A. Stavrou; J. Voas: DDoS in the IoT: Mirai and Other Botnets. Computer 2017,50, 80-84., URL https://ieeexplore.ieee.org/document/7971869 (retrieved: 29 Dec 2020), 2017

[19] A. Linneweber and M. Luckey: IIoT-Lösungen auslegen und integrieren - Auf die Interoperabilität kommt es an, in: IT&Production Internet of Things WK 2019, URL: https://www.i-need.de/?Artikel=163229&page=all (retrieved: December 2020), 2019

[20] L. Lundgren: Light Weight Protocol! Serious Equipment! Critical Implications! Lightweight Protocol! Serious Equipment! Critical Implications!, DEF CON 24, URL: https://media.defcon.org/DEF%20CON%2024/DEF%20CON%2024%20presentations/DEF%20CON%2024%20-%20Lucas-Lundgren-Light-Weight%20Protocol-Critical-Implications.pdf (retrieved: 12 Dec 2020), 2016

[21] C. MacGillivray and D. Reinsel: Worldwide Global DataSphere IoT Device and Data Forecast (2019–2023), URL: https://www.idc.com/getdoc.jsp?containerId=US46718220 (retrieved: 16 Dec 2020), 2019

[22] J. Matherly: Complete Guide to Shodan, version: June 2016, https://leanpub.com/shodan, (retrieved: Aug 2016).

[23] S. Metzler: Mosquitto Swatter:Demonstrating

MQTT Vulnerabilities, Bachelor Thesis, URL: https://se2.informatik.uni-wuerzburg.de/publications/download/paper/2080.pdf (retrieved: 29 Dec 2020), 2019

[24] B. Mohanta; P. Nanda; S. Patnaik: Management of V.U.C.A. (Volatility, Uncertainty, Complexity and Ambiguity) Using Machine Learning Techniques in Industry 4.0 Paradigm, B, in: New Paradigm of Industry 4.0 (Springer), URL: https://www.springer.com/gp/book/9783030257774 (retrieved: 13 Dec 2020), 2020

[25] MQTT Version 5.0, published: 07 March 2019, OASIS, URL: https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html (retrieved: 29 Dec 2020), published: 07 Mar 2019

[26] Guide to Industrial Control Systems (ICS) Security - Supervisory Control and Data Acquisition (SCADA) systems, Distributed Control Systems (DCS), and other control system configurations such as Programmable Logic Controllers (PLC), National Institute of Standards and Technology 2015, URL: http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-82.pdf(retrieved: 15 Dec 2020), 2015

[27] A. Niruntasukrat; C. Issariyapat; P. Pongpaibool; K. Meesublak; P.Aiumsupucgul; A. Panya: Authorization Mechanism for MQTT-based Internet of Things, URL: https://ieeexplore.ieee.org/document/7503802 (retrieved: 15 Dec 2020), 2016

[28] D. Obermaier, heise Developer, URL: https://www.heise.de/developer/artikel/Evolution-der-IoT-Kommunikation-MQTT-5-3941656.html (retrieved: 29 Dec 2020), 2018

[29] Open Web Application Security Project, Internet of Things (IoT) Top 10 2018, URL https://owasp.org/www-pdf-archive/OWASP-IoT-Top-10-2018-final.pdf (retrieved: 12 Nov 2020)

[30] Paloalto: State of Enterprise IoT Security in 2020, https://www.paloaltonetworks.com/resources/infographics/state-of-enterprise-iot-security-in-2020 (retrieved: 29 Dec 2020), 2020

[31] A. Sadeghi; C. Wachsmann; M. Waidner: Security and Privacy Challenges in Industrial Internet of Things. In: Proc. of the 52nd IEEE/ACM Design Automation Conference (DAC), page 54. ACM, URL: https://ieeexplore.ieee.org/abstract/document/7167238 (retrieved: 29 Dec 2020), 2015

[32] Getting the Most Out of Shodan Searches, SANS Penetration Testing, URL: http://pen-testing.sans.org/blog/2015/12/08/effective-shodan-searches?utm_medium=Social&utm_source=Twitter&utm_content=SANSPenTest+Blog+Shodan+Searches&utm_campaign=SANS+Pen+Test, (retrieved: April 2017), 2015

[33] Statista, Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 (in billions), URL: https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/ (retrieved: 29 Dec 2020), 2016

[34] M. Singh; M. A. Rajan; V. L. Shivraj; P. Balamuralidhar: Secure MQTT for Internet of Things (IoT), URL: https://ieeexplore.ieee.org/document/7280018 (retrieved: 13 Dec 2020), 2015

[35] D. Thangavel; X. Ma; A. Valera; H. Tan; C. Tan: Performance evaluation of MQTT and CoAP via a common middleware. In: Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Ninth International Conference on IEEE, URL: https://ieeexplore.ieee.org/document/6827678 (retrieved: 16 Dec 2020), 2014

This Paper was created within the German program „Mittelstand (SME) 4.0 - Competence Center for IT Economy“ of the Ministry for Economic Affairs and Energy that supports small and medium sized enterprises with regard to digital transformation. Further information at: www.itwirtschaft.de