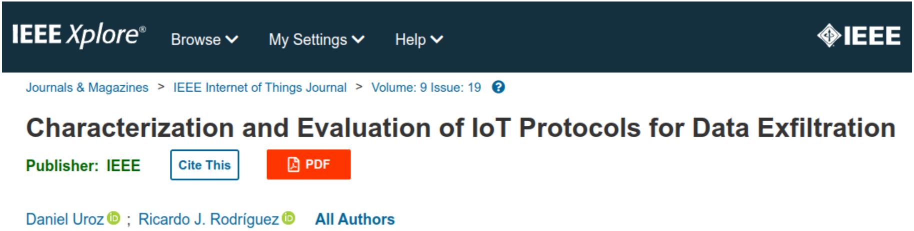
Characterization and Evaluation of IoT Protocols for Data Exfiltration

Daniel Uroz, Ricardo J. Rodríguez

Dept. of Computer Science and Systems Engineering
Universidad de Zaragoza, Spain

November 26, 2022

No cON Name 2022
Barcelona, Spain



Journals & Magazines > IEEE Internet of Things Journal > Volume: 9 Issue: 19

Characterization and Evaluation of IoT Protocols for Data Exfiltration

Publisher: IEEE

Daniel Uroz ; Ricardo J. Rodríguez **All Authors**

doi: 10.1109/JIOT.2022.3163469

$whoami

Daniel Uroz

- **PhD Student at Universidad de Zaragoza**
- **Research interests**:
  - Malware Analysis
  - Reverse Engineering
  - Network Security
  - Computer Forensics

Ricardo J. Rodríguez

- **Associate Professor at Universidad de Zaragoza**
- **Research lines**:
  - Program Binary Analysis
  - Digital Forensics
  - Offensive Security
  - Survivability Analysis with Formal Models

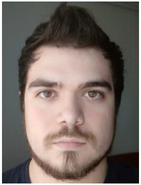
Research Team

We make really good stuff! ☺

- `https://reversea.me`
- `https://twitter.com/reverseame`
- `https://t.me/reverseame`

Ricardo J. Rodríguez
Associate Professor

Daniel Uroz
PhD Student



Razvan Raducu
PhD Student

Daniel Huici
MSc Student

Agenda

1. Introduction

2. Comparative Analysis

3. Chiton Tool

4. Experiments and Discussion

5. Conclusions and Future Work

Agenda

1 Introduction

2 Comparative Analysis

3 Chiton Tool

4 Experiments and Discussion

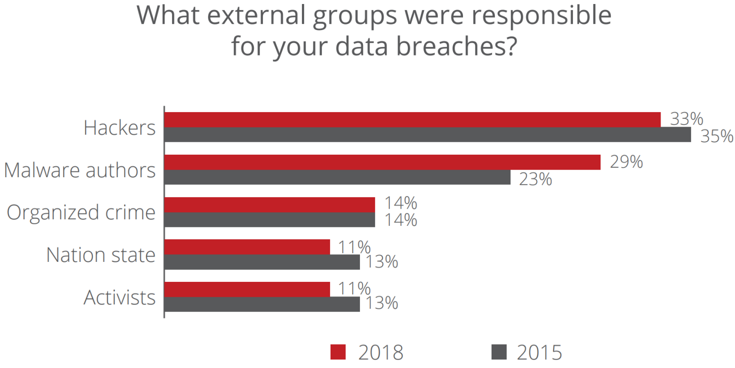
5 Conclusions and Future Work

Universidad Zaragoza

Introduction

Cybercriminals are interested in collecting information

- 61 % of data theft breaches in 2018 were perpetrated by external actors (McAfee)



https://www.mcafee.com/enterprise/en-us/forms/gated-form.html?docID=b5e4babd-d8f1-4155-a242-5c578ef4c6c8

Introduction

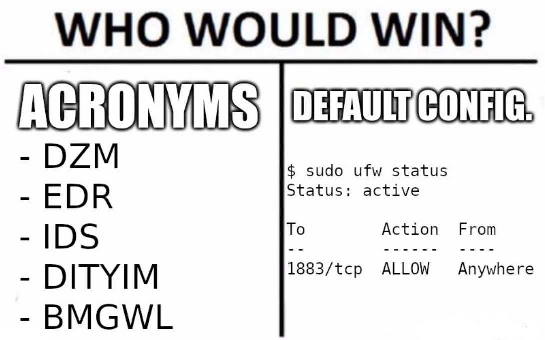
Your usual defenses

- Firewalls
- Demilitarized Zone Networks (DZM)
- Intrusion Detection Systems (IDS)
- Endpoint Detection and Response (EDR)

Introduction

Your usual defenses

- Firewalls
- Demilitarized Zone Networks (DZM)
- Intrusion Detection Systems (IDS)
- Endpoint Detection and Response (EDR)



```
$ sudo ufw status
Status: active

To         Action   From
--         ------   ----
1883/tcp   ALLOW    Anywhere
```

Introduction

Data Exfiltration

Exfiltration – *it comes from the military term*

- Unauthorized transfer of information from an information system (NIST)

Introduction

Data Exfiltration

Exfiltration – *it comes from the military term*

- Unauthorized transfer of information from an information system (NIST)

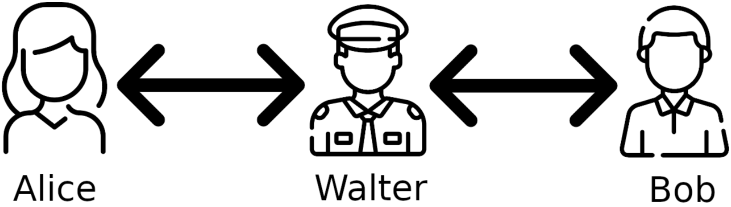
Countermeasures

- Data Loss Prevention (DLP) systems:
  - Endpoint
  - Network
  - Web/Mail Gateway

Introduction

Covert Channels

- Any communication channel that can be exploited by a process to transfer information in a way that violates the system security policy (U.S. DoD)
    1. Storage channels
    2. Timing channels

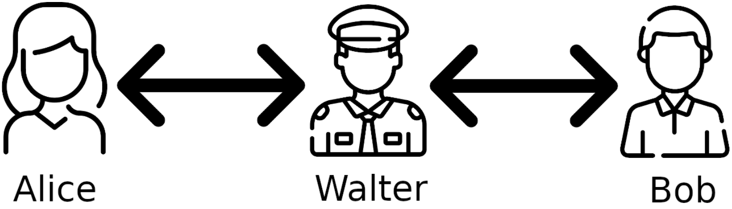


Icons from flaticon.com

Introduction

Covert Channels

- Any communication channel that can be exploited by a process to transfer information in a way that violates the system security policy (U.S. DoD)
  1. **Storage channels → Data exfiltration**
  2. Timing channels



Icons from flaticon.com

Covert channels are not new...

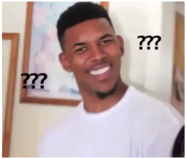
- TCP/IP (Rowland, 1997) and IPv6 (Graf, 2003) protocol suite
- HTTP(S), WLAN, VoIP, SSH, FTP, NTP (Mazurczyk et al., 2016)

Introduction

Covert Channel

Covert channels are not new...

- TCP/IP (Rowland, 1997) and IPv6 (Graf, 2003) protocol suite
- HTTP(S), WLAN, VoIP, SSH, FTP, NTP (Mazurczyk et al., 2016)
- IEEE 802.3 10 Gigabit Ethernet physical layer (Lee et al., 2014) and between virtualized systems in the cloud (Wu et al., 2015)

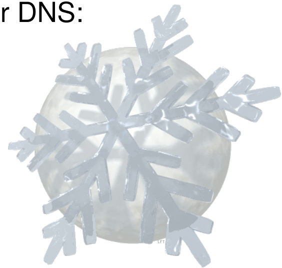


Introduction

Covert Channel

Tunneling protocols

- A specific type of storage covert channel where one protocol is embedded within the payload of another protocol
- Suitable for data exfiltration:
  - ↑ Throughput
  - ↓ Low-profile communication
- Example of IPv4 over DNS:



https://code.kryo.se/iodine

Internet of Things (IoT)

- Integration of various sensors, objects, and smart nodes capable of communicating with each other without human intervention

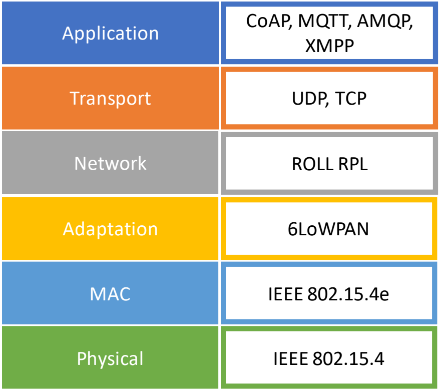
Introduction

IoT Protocols

Internet of Things (IoT)

- Integration of various sensors, objects, and smart nodes capable of communicating with each other without human intervention

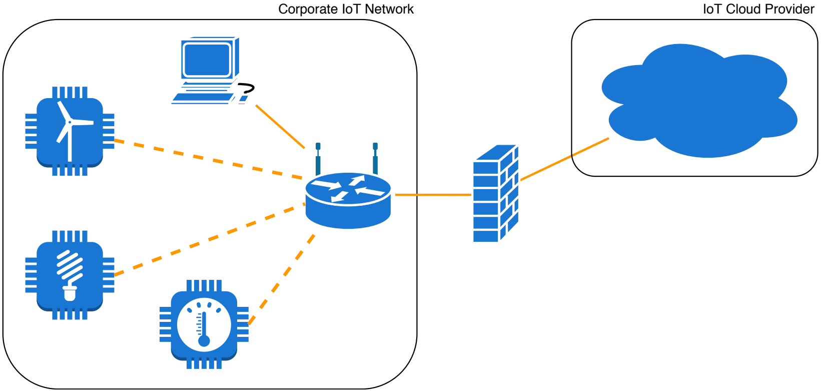
IoT Protocol Stack (Palattella et al., 2013)



Application	CoAP, MQTT, AMQP, XMPP
Transport	UDP, TCP
Network	ROLL RPL
Adaptation	6LoWPAN
MAC	IEEE 802.15.4e
Physical	IEEE 802.15.4

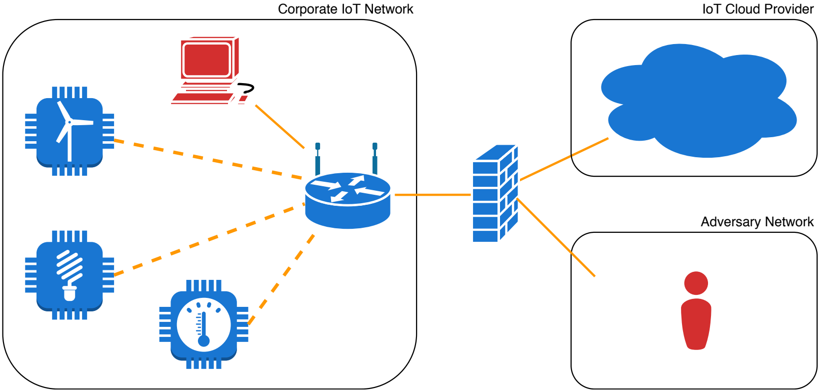
Introduction

Data Exfiltration in IoT Networks



Introduction

Data Exfiltration in IoT Networks



Which IoT protocol is best suited for data exfiltration?

Agenda

1 Introduction

2 Comparative Analysis

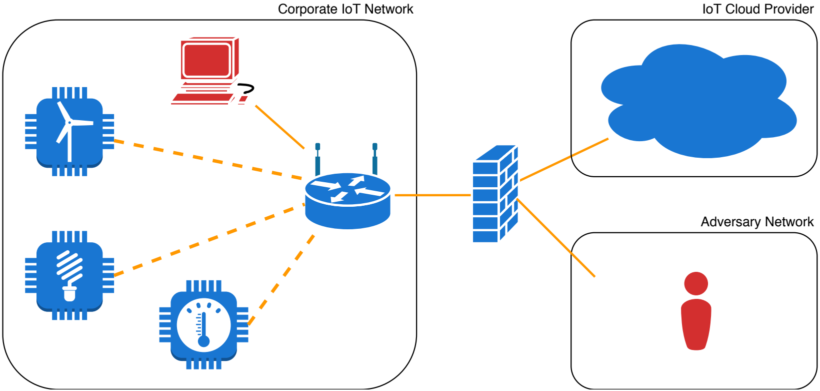
3 CHITON Tool

4 Experiments and Discussion

5 Conclusions and Future Work

Comparative Analysis

IoT Protocols



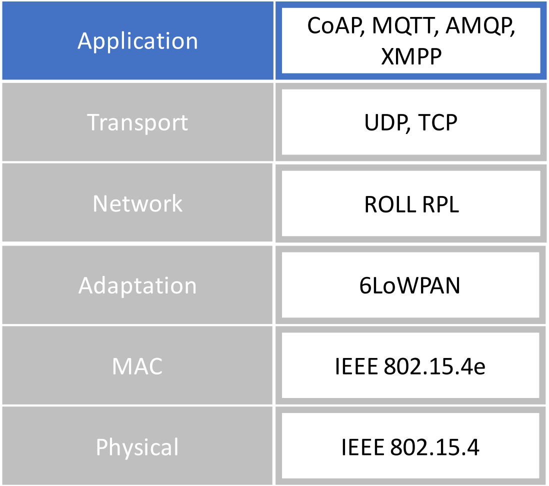
Comparative Analysis

IoT Protocols

Application	CoAP, MQTT, AMQP, XMPP
Transport	UDP, TCP
Network	ROLL RPL
Adaptation	6LoWPAN
MAC	IEEE 802.15.4e
Physical	IEEE 802.15.4

Comparative Analysis

IoT Protocols



Application	CoAP, MQTT, AMQP, XMPP
Transport	UDP, TCP
Network	ROLL RPL
Adaptation	6LoWPAN
MAC	IEEE 802.15.4e
Physical	IEEE 802.15.4

Comparative Analysis

IoT Protocols

aws

MQTT v3.1.1

MQTT v3.1.1
AMQP v1.0

MQTT v3.1.1

Comparative Analysis

IoT Protocols

MQTT v3.1.1

MQTT v3.1.1
AMQP v1.0

MQTT v3.1.1

For the sake of completeness:

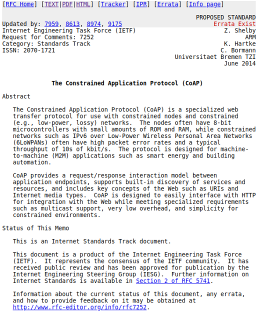
- MQTT v5.0
- CoAP v1.0

Comparative Analysis

Mmmmmm..., standards

Comparative Analysis

Mmmmmm..., standards



```
[RFC Home] [TEXT|PDF|HTML] [Tracker] [IPR] [Errata] [Info page]

                                                       PROPOSED STANDARD
Updated by: 7959, 8613, 8974, 9175                          Errata Exist
Internet Engineering Task Force (IETF)                         Z. Shelby
Request for Comments: 7252                                           ARM
Category: Standards Track                                      K. Hartke
ISSN: 2070-1721                                               C. Bormann
                                                 Universitaet Bremen TZI
                                                               June 2014
```

The Constrained Application Protocol (CoAP)

Abstract

The Constrained Application Protocol (CoAP) is a specialized web transfer protocol for use with constrained nodes and constrained (e.g., low-power, lossy) networks. The nodes often have 8-bit microcontrollers with small amounts of ROM and RAM, while constrained networks such as IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs) often have high packet error rates and a typical throughput of 10s of kbit/s. The protocol is designed for machine-to-machine (M2M) applications such as smart energy and building automation.

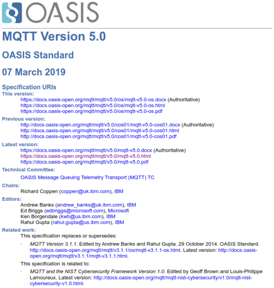
CoAP provides a request/response interaction model between application endpoints, supports built-in discovery of services and resources, and includes key concepts of the Web such as URIs and Internet media types. CoAP is designed to easily interface with HTTP for integration with the Web while meeting specialized requirements such as multicast support, very low overhead, and simplicity for constrained environments.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7252.



OASIS

MQTT Version 5.0

OASIS Standard

07 March 2019

Specification URIs

This version:
https://docs.oasis-open.org/mqtt/mqtt/v5.0/os/mqtt-v5.0-os.docx (Authoritative)
https://docs.oasis-open.org/mqtt/mqtt/v5.0/os/mqtt-v5.0-os.html
https://docs.oasis-open.org/mqtt/mqtt/v5.0/os/mqtt-v5.0-os.pdf

Previous version:
http://docs.oasis-open.org/mqtt/mqtt/v5.0/cos01/mqtt-v5.0-cos01.docx (Authoritative)
http://docs.oasis-open.org/mqtt/mqtt/v5.0/cos01/mqtt-v5.0-cos01.html
http://docs.oasis-open.org/mqtt/mqtt/v5.0/cos01/mqtt-v5.0-cos01.pdf

Latest version:
https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.docx (Authoritative)
https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html
https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.pdf

Technical Committee:
OASIS Message Queuing Telemetry Transport (MQTT) TC

Chairs:
Richard Coppen (coppen@uk.ibm.com), IBM

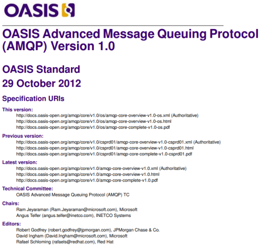
Editors:
Andrew Banks (andrew_banks@uk.ibm.com), IBM
Ed Briggs (edbriggs@microsoft.com), Microsoft
Ken Borgendale (kwb@us.ibm.com), IBM
Rahul Gupta (rahul.gupta@us.ibm.com), IBM

Related work:
This specification replaces or supersedes:

- *MQTT Version 3.1.1*. Edited by Andrew Banks and Rahul Gupta. 29 October 2014. OASIS Standard. http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html. Latest version: http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html.

This specification is related to:

- *MQTT and the NIST Cybersecurity Framework Version 1.0*. Edited by Geoff Brown and Louis-Philippe Lamoureux. Latest version: http://docs.oasis-open.org/mqtt/mqtt-nist-cybersecurity/v1.0/mqtt-nist-cybersecurity-v1.0.html.



OASIS

OASIS Advanced Message Queuing Protocol (AMQP) Version 1.0

OASIS Standard
29 October 2012

Specification URIs

This version:
http://docs.oasis-open.org/amqp/core/v1.0/os/amqp-core-overview-v1.0-os.xml (Authoritative)
http://docs.oasis-open.org/amqp/core/v1.0/os/amqp-core-overview-v1.0-os.html
http://docs.oasis-open.org/amqp/core/v1.0/os/amqp-core-complete-v1.0-os.pdf

Previous version:
http://docs.oasis-open.org/amqp/core/v1.0/csprd01/amqp-core-overview-v1.0-csprd01.xml (Authoritative)
http://docs.oasis-open.org/amqp/core/v1.0/csprd01/amqp-core-overview-v1.0-csprd01.html
http://docs.oasis-open.org/amqp/core/v1.0/csprd01/amqp-core-complete-v1.0-csprd01.pdf

Latest version:
http://docs.oasis-open.org/amqp/core/v1.0/amqp-core-overview-v1.0.xml (Authoritative)
http://docs.oasis-open.org/amqp/core/v1.0/amqp-core-overview-v1.0.html
http://docs.oasis-open.org/amqp/core/v1.0/amqp-core-complete-v1.0.pdf

Technical Committee:
OASIS Advanced Message Queuing Protocol (AMQP) TC

Chairs:
Ram Jeyaraman (Ram.Jeyaraman@microsoft.com), Microsoft
Angus Telfer (angus.telfer@inetco.com), INETCO Systems

Editors:
Robert Godfrey (robert.godfrey@jpmorgan.com), JPMorgan Chase & Co.
David Ingham (David.Ingham@microsoft.com), Microsoft
Rafael Schloming (rafaels@redhat.com), Red Hat

CoAP	MQTT	AMQP
112 pp.	137 pp.	125 pp.

ad

Comparative Analysis

Qualitatively Analysis

- **Message type**
  - CoAP *methods*
  - MQTT *control packets*
  - AMQP *performatives*
- **Transport**
  - UDP: CoAP
  - TCP: MQTT, AMQP
- **Error detection**
  - All protocols rely exclusively on the error detection mechanisms provided by the transport layers (UDP and TCP checksum field)

Each message type is divided in

- **Payload**: how much data a protocol can carry in a single message
- **Overhead**: each byte sent that does not represent exfiltrated data

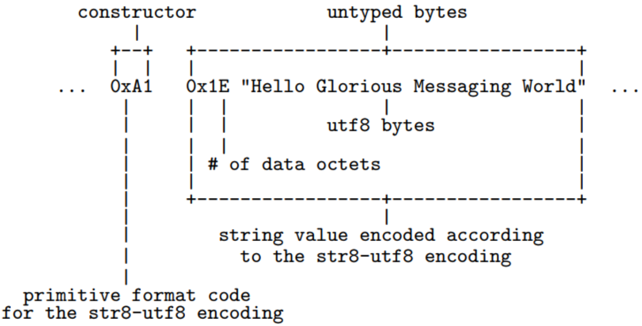
Comparative Analysis

Quantitative Analysis

Each message type is divided in

- **Payload**: how much data a protocol can carry in a single message
- **Overhead**: each byte sent that does not represent exfiltrated data

Overhead is not only headers

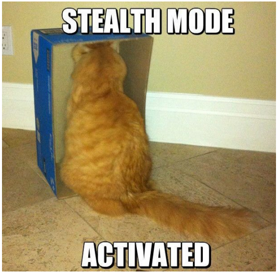


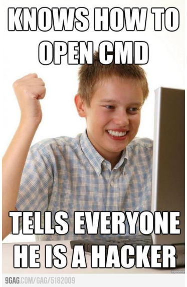
```
                 constructor                 untyped bytes
                      |                            |
                    +--+     +-----------------+-----------------+
                    |  |     |                                   |
               ...  0xA1     0x1E "Hello Glorious Messaging World"  ...
                      |      |  |              |                 |
                      |      |  |          utf8 bytes            |
                      |      |  |                                |
                      |      | # of data octets                  |
                      |      |                                   |
                      |      +-----------------+-----------------+
                      |                        |
                      |        string value encoded according
                      |          to the str8-utf8 encoding
                      |
             primitive format code
           for the str8-utf8 encoding
```

AMQP primitive type

Comparative Analysis

Quantitative Analysis





Adversary types

- **Stealthy adversary**: adapts messages to commonly used sizes
- **Rough adversary**: maximizes the possible payload for each message

Comparative Analysis

Quantitative Analysis

	Stealthy Adversary			Rough Adversary		
	Message Size	Overhead	# Messages	Message Size	Overhead	# Messages
CoAP						
`GET/DELETE`	1280	0.94 %	827	65,507	0.74 %	17
`POST/PUT`	1280	0.55 %	820	65,507	0.01 %	17
MQTT	(***version 3.1.1***)			(***version 5.0***)		
`CONNECT`	37	37.84 %	45,591	1,048,635	0.01 %	1
`CONNACK`	-	-	-	1,048,628	< 0.01 %	1
`PUBLISH`	65,495	0.01 %	17	1,048,583	< 0.01 %	1
`PUBACK/PUBREC/PUBREL/PUBCOMP`	5	60 %	524,288	1,048,627	< 0.01 %	1
`SUBSCRIBE`	263	2.28 %	4081	1,048,626	< 0.01 %	1
`UNSUBSCRIBE`	262	1.91 %	4081	1,048,625	< 0.01 %	1
`SUBACK/UNSUBACK`	5	60 %	524,288	1,048,627	< 0.01 %	1
`PINGREQ/PINGRESP`	-	-	-	-	-	-
`DISCONNECT`	-	-	-	1,048,627	< 0.01 %	1
`AUTH`	-	-	-	1,048,627	< 0.01 %	1
AMQP						
`Open`	4121	0.61 %	256	1,048,601	< 0.01 %	1
`Begin`	30	60 %	87,382	1,048,606	< 0.01 %	1
`Attach`	4126	0.63 %	256	1,048,602	< 0.01 %	1
`Flow`	30	60 %	87,382	1,048,625	< 0.01 %	1
`Transfer`	65,495	0.03 %	17	1,048,596	< 0.01 %	1
`Disposition`	20	80.00 %	262,144	1,048,605	< 0.01 %	1
`Detach`	19	78.95 %	262,144	1,048,615	< 0.01 %	1
`End`	-	-	-	1,048,613	< 0.01 %	1
`Close`	-	-	-	1,048,613	< 0.01 %	1

Exfiltration of 1,048,576 bytes (1 MiB) by IoT protocol. Message size is expressed in bytes

Agenda

1 Introduction

2 Comparative Analysis

3 Chiton Tool

4 Experiments and Discussion

5 Conclusions and Future Work

Chiton Tool

Chitón. Capricho nº 28, Francisco de Goya (Museo Nacional del Prado)

python + scapy

`https://github.com/reverseame/chiton`

GPLv3 Free Software

Free as in Freedom

Agenda

1 Introduction

2 Comparative Analysis

3 CHITON Tool

4 Experiments and Discussion

5 Conclusions and Future Work

Experiments and Discussion

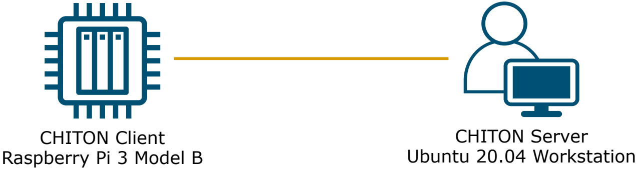
Threat Modeling

About our adversary...

- **Maximizes the payload exfiltrated** in each message
- Chooses **IoT traffic** to avoid heavily monitored networks
- **Does not have privileged access**
  - Only ports greater than 1024 are available to communicate

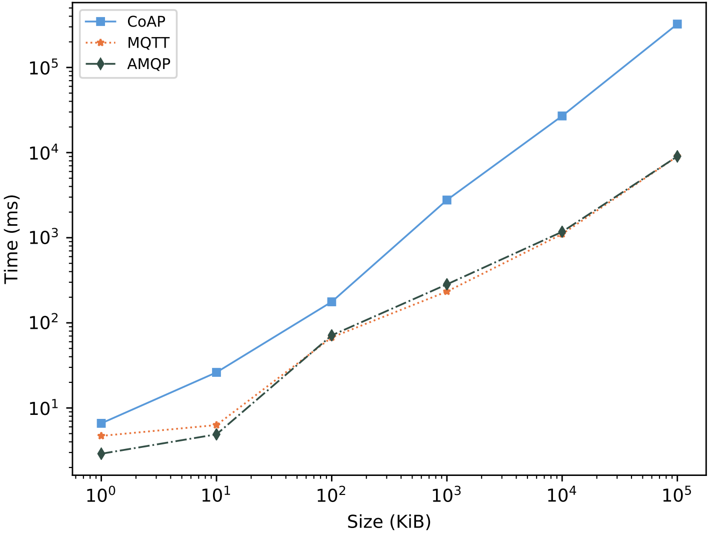
Experiments and Discussion
Description of Experiments

- Exfiltrate data from 1 KiB to 100 MiB using Chiton (x10)
- Message best suited for exfiltrating data (↑↑ payload/overhead ratio):
  - **CoAP**: `POST` and `PUT` *methods*
  - **MQTT**: `PUBLISH` *control packet*
  - **AMQP**: `Transfer` *performative*



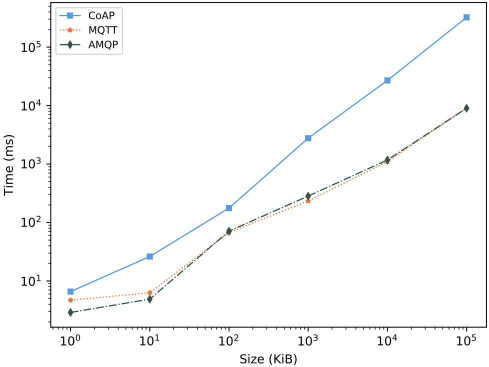
Experiments and Discussion

Results



Experiments and Discussion

Discussion



What happens with CoAP?

1. More messages are needed to send the same amount of data
2. It runs on top of UDP (unlike the MQTT and AMQP protocols)
3. **Its message size could be suboptimal**

Agenda

1 Introduction

2 Comparative Analysis

3 Chiton Tool

4 Experiments and Discussion

5 Conclusions and Future Work

Conclusions and Future Work

- *How to exfiltrate data over IoT protocols?*
  - **Use MQTT**
- `Chiton` **tool** (it's free software, use and expand it! ♥)
- *How to protect systems against data exfiltration over IoT protocols?*
  - Put DLP systems in place
  - Make sure your defense systems are stateful

Conclusions and Future Work

- *How to exfiltrate data over IoT protocols?*
  - **Use MQTT**
- `Chiton` **tool** (it's free software, use and expand it! ♥)
- *How to protect systems against data exfiltration over IoT protocols?*
  - Put DLP systems in place
  - Make sure your defense systems are stateful

Future work

- **Empirical tests with real firewalls or IDPS**
- **Emulate the "happy flow" of the protocol** (in case of detection)

References

McAfee, 2019 McAfee, “Grand Theft Data II: The Drivers and Shifting State of Data Breaches,” MSI-ACI Europe, 2019.

NIST, 2020 NIST, “Glossary: Exfiltration.” 2020. [Online]. `https://csrc.nist.gov/glossary/term/exfiltration`

U.S. DoD, 1985 U.S. Department of Defense, “Trusted Computer System Evaluation Criteria,” in The ‘Orange Book’ Series, London: Palgrave Macmillan UK, 1985, pp. 1–129.

Palattella et al., 2013 M. R. Palattella, N. Accettura, X. Vilajosana, T. Watteyne, L. A. Grieco, G. Boggia, and M. Dohler, “Standardized Protocol Stack for the Internet of (Important) Things,” IEEE Communications Surveys Tutorials, vol. 15, no. 3, Art. no. 3, 2013.

Rowland, 1997 C. H. Rowland, “Covert channels in the TCP/IP protocol suite,” First Monday, vol. 2, no. 5, Art. no. 5, 1997.

Graf, 2003 T. Graf, “Messaging over IPv6 destination options.” 2003.

Mazurczyk et al., 2016 W. Mazurczyk, S. Wendzel, S. Zander, A. Houmansadr, and K. Szczypiorski, Information Hiding in Communication Networks: Fundamentals, Mechanisms, Applications, and Countermeasures, 1st ed. Wiley-IEEE Press, 2016.

Wu et al., 2015 Z. Wu, Z. Xu, and H. Wang, “Whispers in the Hyper-Space: High-Bandwidth and Reliable Covert Channel Attacks Inside the Cloud,” IEEE/ACM Transactions on Networking, vol. 23, no. 2, Art. no. 2, 2015.

Lee et al., 2014 K. S. Lee, H. Wang, and H. Weatherspoon, “PHY Covert Channels: Can you see the Idles?,” in 11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14), Seattle, WA, Apr. 2014, pp. 173–185.

Characterization and Evaluation of IoT Protocols for Data Exfiltration

Daniel Uroz, Ricardo J. Rodríguez

Universidad Zaragoza
1542

Dept. of Computer Science and Systems Engineering
Universidad de Zaragoza, Spain

November 26, 2022

No cON Name 2022
Barcelona, Spain