Improving the resilience of the constrained Internet of Things : a moving target defense approach

Renzo Efraín Navas

▶ **To cite this version:**

Renzo Efraín Navas. Improving the resilience of the constrained Internet of Things : a moving target defense approach. Cryptography and Security [cs.CR]. Ecole nationale supérieure Mines-Télécom Atlantique, 2020. English. NNT : 2020IMTA0217 . tel-03123143

HAL Id: tel-03123143
https://theses.hal.science/tel-03123143

Submitted on 27 Jan 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

DOCTORAT BRETAGNE LOIRE / MATHSTIC

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

THÈSE DE DOCTORAT DE

L'ÉCOLE NATIONALE SUPÉRIEURE MINES-TÉLÉCOM ATLANTIQUE
BRETAGNE PAYS DE LA LOIRE - IMT ATLANTIQUE

ÉCOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Informatique*

Par

Renzo Efraín NAVAS

Improving the Resilience of the Constrained Internet of Things

A Moving Target Defense approach

Thèse présentée et soutenue à IMT Atlantique campus Rennes, le 9 Décembre 2020
Unité de recherche : Lab-STICC
Thèse N° : 2020IMTA0217

Rapporteurs avant soutenance :

Cristel PELSSER	Professeure, Université de Strasbourg, France.
Fabrice VALOIS	Professeur, INSA Lyon, France.

Composition du Jury :

Président :	Ludovic MÉ	Professeur, CentraleSupélec, France.
Examinateurs :	Frédéric CUPPENS	Professeur, Polytechnique Montréal, Canada.
	Cristel PELSSER	Professeure, Université de Strasbourg, France.
	Franck ROUSSEAU	Maître de conférences, Grenoble INP-Ensimag, France.
	Marco TILOCA	Ph.D. Senior Researcher, RISE, Suède.
	Fabrice VALOIS	Professeur, INSA Lyon, France.
Dir. de thèse :	Laurent TOUTAIN	Professeur, IMT Atlantique, France.
Encadr. de thèse :	Georgios Z. PAPADOPOULOS	Maître de conférences, IMT Atlantique, France.

Invité(s) :

Nora BOULAHIA CUPPENS	Professeure, Polytechnique Montréal, Canada.

IMPROVING THE RESILIENCE OF THE CONSTRAINED INTERNET OF THINGS

RENZO E. NAVAS

—Janus.

A Moving Target Defense approach

December 2020 – Version Archivage

Renzo E. Navas: *Improving the Resilience of the Constrained Internet of Things,* A Moving Target Defense approach, © December 2020

Dedicated to the loving memory of my grandfather,
Alberto A. Navas, “Ito”.

1929 – 2019

ABSTRACT

Internet of Things (IoT) systems are increasingly deployed in the real world, but their security lags behind the state of the art of non-IoT systems. Moving Target Defense (MTD) is a cyberdefense paradigm that proposes to perpetually randomize systems' components, with the intention of thwarting cyber attackers that previously relied on the static nature of them. Leaked system information is now ephemeral, and attackers are constrained by time. MTD has been successfully implemented in conventional systems, but its use to improve IoT security is still lacking in the literature. Throughout this thesis, we establish MTD as a cyber defense technique for the resource-constrained IoT.

First, we validated MTD as a suitable technique for IoT systems. We identified and synthesized existing MTD techniques for IoT using a systematic literature review method. Real-world usability evidence was leveraged from the state of the art; besides, we defined and used four novel entropy-related metrics to measure qualitative aspects of the existing techniques.

Second, we proposed a generic and modular distributed MTD framework that allows the instantiation of concrete MTD strategies suitable for the constrained IoT. Then, we designed an authenticated time synchronization protocol, proven secure using a computer-aided formal method. This protocol allows instantiating one of the fundamental components of our MTD framework.

Finally, we instantiated three concrete MTD techniques. Two at the upper network layers (dealing with port-hopping and application RESTful interfaces) and the third one at the physical layer using direct-sequence spread spectrum anti-jamming techniques. The physical layer technique also provided a fundamental study about the cross-correlation of pseudo-random sequences for wireless communication systems previously missing in the literature.

RÉSUMÉ

Les systèmes de l'Internet des Objets, ou «Internet of Things» (IoT), sont de plus en plus déployés dans le monde réel, mais leur sécurité est à la traîne par rapport à l'état de l'art des systèmes non IoT. Le «Moving Target Defense» (MTD), ou Défense par Cible Mouvante, est un paradigme de cyberdéfense qui propose de randomiser perpétuellement des composants de systèmes, dans l'intention de faire échec aux cyberattaquants qui s'appuyaient auparavant sur la nature statique de ceux-ci. Les informations du système qui fuient sont maintenant éphémères et les attaquants sont limités par le temps. Le MTD a été mis en œuvre avec succès dans des systèmes conventionnels, mais son utilisation pour améliorer la sécurité de l'IoT fait manque encore dans la littérature. Tout au long de cette thèse, nous établissons la MTD comme une technique de cyberdéfense pour l'IoT contraint.

Tout d'abord, nous avons validé la MTD comme une technique appropriée pour les systèmes IoT. Nous avons identifié et synthétisé les techniques MTD existantes pour l'IoT en utilisant une méthode systématique d'examen de la littérature. Nous avons également défini et utilisé quatre nouveaux paramètres liés à l'entropie pour mesurer les aspects qualitatifs des techniques existantes.

En deuxième lieu, nous avons proposé un framework générique et modulaire de MTD distribué qui permet l'instanciation de stratégies MTD concrètes adaptées à l'IoT contraint. Troisièmement, nous avons conçu un protocole de synchronisation temporelle authentifié, dont la sécurité a été prouvée par une méthode formelle assistée par ordinateur. Ce protocole permet d'instancier l'une des composantes fondamentales de notre framework de MTD.

Enfin, nous avons instancié trois techniques MTD concrètes. Deux au niveau des couches supérieures du réseau (portant sur le saut de port et sur des interfaces RESTful d'applications) et la troisième au niveau de la couche physique en utilisant des techniques anti-bourrage à étalement de spectre à séquence directe (direct-sequence spread spectrum). La technique de la couche physique a également fourni une étude fondamentale sur la corrélation croisée des séquences pseudo-aléatoires pour les systèmes de communication sans fil, qui était jusqu'alors absente de la littérature.

PUBLICATIONS

The following are the publications during the course of my thesis:

[1] Renzo E. Navas, Frédéric Cuppens, Nora Boulahia Cuppens, Laurent Toutain, and Georgios Z. Papadopoulos. "MTD, Where Art Thou? A Systematic Review of Moving Target Defense Techniques for IoT." In: *IEEE Internet of Things Journal* (2020), pp. 1–15. DOI: 10.1109/JIOT.2020.3040358.

[2] Renzo E. Navas, Frédéric Cuppens, Nora Boulahia Cuppens, Laurent Toutain, and Georgios Z. Papadopoulos. "Physical resilience to insider attacks in IoT networks: Independent cryptographically secure sequences for DSSS anti-jamming." In: *Computer Networks* (2020), pp. 1–16. DOI: 10.1016/j.comnet.2020.107751.

[3] Renzo E. Navas, Hélène Le Bouder, Nora Cuppens, Frédéric Cuppens, and Georgios Z. Papadopoulos. "Demo: Do not trust your neighbors! A small IoT platform illustrating a man-in-the-middle attack." In: *The 17th International Conference on Ad-Hoc Networks and Wireless – AdHoc-Now 2018* (Saint-Malo, France). Springer, Sept. 2018, pp. 120–125. DOI: 10.1007/978-3-030-00247-3_11.

[4] Renzo E. Navas, Håkon Sandaker, Frédéric Cuppens, Nora Cuppens, Laurent Toutain, and Georgios Z. Papadopoulos. "IANVS: A Moving Target Defense Framework for a Resilient Internet of Things." In: *2020 IEEE Symposium on Computers and Communications (ISCC)* (Rennes, France). IEEE, July 2020, pp. 1–6. DOI: 10.1109/ISCC50000.2020.9219728.

[5] Renzo E. Navas and Laurent Toutain. "LATe: A Lightweight Authenticated Time Synchronization Protocol for IoT." In: *2018 Global Internet of Things Summit (GIoTS)* (Bilbao, Spain). IEEE, June 2018, pp. 1–6. DOI: 10.1109/GIOTS.2018.8534565.

[6] Renzo E. Navas, Laurent Toutain, and Georgios Z. Papadopoulos. "Techniques de Moving Target Defense pour l'IoT (à paraitre)." In: *La gestion et le contrôle intelligents des performances et de la sécurité dans l'IoT*. ISTE Editions, 2021. Chap. 11, pp. 1–28.

I also had the pleasure to contribute with Dr. Routa Moussaileb to the following journal article, in a field not treated in this memoir:

[1] Routa Moussaileb, Renzo E. Navas, and Nora Cuppens. "Watch Out! Doxware on The Way..." In: *Journal of Information Security and Applications* 55 (2020), pp. 2214–2126. DOI: 10.1016/j.jisa.2020.102668.

AUDIOVISUAL PRODUCTIONS

I created audiovisual content related to the subject of this memoir:

[1] Renzo E. Navas. *Demo Video: IoT Man-in-the-Middle Attack*. Alternative url: https://www.youtube.com/watch?v=Zhrk5-IGKKE. Apr. 2018. URL: http://www.industry-of-the-future.org/asset/demo/.

[2] Renzo E. Navas. *Securing the IoT through Moving Target Defense (short film)*. Ed. by Festival de sciences en cour[t]s (scientific vulgarisation). Alternative url: https://www.youtube.com/watch?v=Kz8w_vXTBuQ. Oct. 2019. DOI: 10.5281/zenodo.4431808. URL: https://doi.org/10.5281/zenodo.4431808.

Finally, the raw video footage of this thesis defense is also accessible:

[1] Renzo E. Navas. *Public Defense of Doctoral Thesis: "Improving the Resilience of the Constrained Internet of Things" (raw video)*. Alternative url: https://www.youtube.com/watch?v=Lt-28_jOSuE. Dec. 2020. DOI: 10.5281/zenodo.4431731. URL: https://doi.org/10.5281/zenodo.4431731.

A book is a physical object in a world of physical objects.
It is a set of dead symbols. And then the right reader comes along,
and the words—or rather the poetry behind the words,
for the words themselves are merely symbols—spring to life
and we have the resurrection of the word. — Jorge Luis Borges

ACKNOWLEDGMENTS

Words are not enough to express my gratitude towards so many people and institutions.

I start with family and the person who gave me birth: Silvia Dora Maneiro (I love you, mom!). To my father Edgardo Alberto Navas and my siblings: René, Leda, and Carla. I miss you; it was hard to leave Saavedra/Nuñez. To the reason I could do it for so long: Tatiana, my love, my partner in life, I love you. To Ali and Ito, my grandparents –my grandfather game me so much, every time San Lorenzo plays I am with you–. My beloved godmother "Luli", Claudia, and Marito (Eze and Agus). To my family in Mercedes–Uruguay (Maneiro) –you are too many to enumerate! But I love you all, especially Maticolis ;)–, in Gonnet–La Plata (Peña), in Azul (Abonjo), in Malmö (Maneiro). Thank you.

Fortunately, it is a hard task to enumerate all the people I love in this life and I do not want to hurt any feelings by omitting them: the list of names is not exhaustive!

Thank you, to Friends and Teachers from all my –Argentinean– life: Escuela Nro. 4 D.E. 10 "Coronel Brandsen", Escuela Superior de Comercio "Carlos Pellegrini", and Universidad de Buenos Aires (Facultad de Ingeniería mostly, but also Filosofía, and Ciudad Universitaria), and Life. With some, I am still in contact, with others not as much, but all have a place in me. Gracias a todos, espero nos veamos pronto. Thanks to the public education system of the República Argentina: otherwise I will not be writing these words.

To Alejandro Lampropulos, who was the cause I came to France: "mi buen amigo". To the beautiful people I met and shared my life in Rennes since 2011 (you know who you are, and many are not in Rennes anymore): we had a blast! To the beautiful people at Télécom Bretagne/IMT Atlantique: thank you for making me feel at home (it's been almost ten years!). I will always remember my period there fondly: Cesson-Sévigné, yout people, the campus, the kfet, the MAISEL, all is just great –almost like a fairy tale–!

To German Castagnini, Nicolas Montavont, and Alberto Blanc, who mentored me as a young stagiaire. To Alberto Dams my Engineering thesis' tutor. To Laurent Toutain –my mentor–, to whom I owe much, thank you for giving me so many interesting projects, freedom, and opportunities to learn. To Frédéric Cuppens and Nora Boulahia Cuppens, thank you for believing in me and for giving me the opportunity to do this doctoral thesis about such a wonderful subject! My eternal gratitude towards you. To Georgios Z. Papadopoulos, co-supervisor and friend, thank you for being there Geo! Thanks to TUM and Prof. Georg Sigl, to the German-French Academy for the Industry of the Future, Contrat Plan Etat-Région, and the other support that allowed this thesis to be possible. Thanks to Cristel Pelsser and Fabrice Valois –the reviewers of this thesis– for your time and carefully reading a 200-page document. Thanks to the rest of the jury: Ludovic Mé, Franck Rousseau, and Marco Tiloca. To all of you, including my friends who were there on the frozen-video YouTube stream (epic FAIL, but you did not care): Thank you, I will never forget the Wednesday 9th of December 2020. Thanks to IMT Atlantique as an institution (RSM/SRCD department –especially OCIF and IRIS teams–), and the République Française: Liberté, Égalité, Fraternité.

To my wonderful group of friends at IMT Atlantique over the years, the ADER association –ADER is us! thank you for the pizza, laughs, sports, table-video games, chess, laser tag, bowling, BBQs, football, movie nights, etc etc etc :')–, my friends: you are so many that is a fact that I will forget somebody, and then you will be offended! I do not want that. If you are reading these words: I remember you, call me and let's have a virtual drink (Romain, Lucien, Vova/Moiz, Marina, Samy, Mae, Jairo, Juan, Rodrigo, Fede, Edwin?)! I will do an exception with the people with whom I shared office: Maru, Mauro, (María and Pablo), Nesrine, Lampro, Tanguychapituli, BB, Sarah, Tanguy Boludo, Manolo (y un abrazo para Robert), Mathieu, Aris, Tomás, Mauricio, Laudin, Ali, Kun, Mónica, Indra, Juan Carlos, and Tania. The people next door will forgive me, especially François, Saad, Hristina, and Xavier. Finally –another exception–, to the "PhD Babies", my friends whom we started theses at the same time: Routa and Farah (thank you for everything girls), Tommy, Guillaume, and Edu.

Last but not least, to the beloved pets in my life. My doggies: Titanitus and Uri (Valentina the turtle and Negro). My cats: Luna, Charlotte, and –especially– Tisha. You definitely make life better.

To all, including the ones I did not mention explicitly –or forget–: I am grateful that we have met in this life.

In my life I've loved you all
In My Life — The Beatles

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LISTINGS

ACRONYMS

6LoWPAN	IPv6 over Low-Power Wireless Personal Area Networks
6TiSCH	IPv6 over the TSCH mode of IEEE 802.15.4e
ACE	Authentication and Authorization for Constrained Environments
AES	Advanced Encryption Standard
AJ	Anti-Jamming
AKE	Authenticated Key Exchange
ASLR	Address Space Layout Randomization
AWGN	Additive White Gaussian Noise
BBN	Broadband Noise
BER	Bit Error Rate
BPSK	Binary Phase-Shift Keying
CBC	Cipher Block Chaining
CBOR	Concise Binary Object Representation
CC	Cross-Correlation
CCM	Counter with CBC-MAC
CDF	Cumulative Distribution Function
CDMA	Code-Division Multiple Access
CoAP	Constrained Application Protocol
COSE	CBOR Object Signing and Encryption
CSPR	Cryptographically Secure Pseudo-Random
CSPRNG	Cryptographically Secure Pseudo-Random Number Generator
CTR	Counter Mode
DoS	Denial-of-Service
DSSS	Direct-Sequence Spread Spectrum
DTLS	Datagram Transport Layer Security
ECDF	Empirical Cumulative Distribution Function
ECDH	Elliptic Curve Diffie-Hellman
EDHOC	Ephemeral Diffie-Hellman Over COSE
FHSS	Frequency-Hopping Spread Spectrum
FSR	Feedback Shift Register
HKDF	HMAC-based Key Derivation Function
HMAC	Hashed Message Authentication Code
IETF	Internet Engineering Task Force
IoT	Internet of Things
JSR	Jamming-to-Signal Ratio
LATe	Lightweight Authenticated Time
LFSR	Linear-Feedback Shift Register
LWIG	Light-Weight Implementation Guidance

MAC	Message Authentication Code
MP	Moving Parameter
MTD	Moving Target Defense
NCC	Normalized circular Cross-Correlation
NIST	National Institute of Standards and Technology
NLFSR	Non-Linear-Feedback Shift Register
NTP	Network Time Protocol
NTS	Network Time Security
OSCORE	Object Security for Constrained RESTful Environments
PMF	Probability Mass Function
PR	Pseudo-Random
PRF	Pseudo-Random Function
PSK	Pre-Shared Key
REST	REpresentational State Transfer
RPK	Raw Public Key
RPL	IPv6 Routing Protocol for Low-Power and Lossy Networks
RQ	Research Question
RTC	Real-Time Clock
RV	Random Variable
RX	Receiver
SDN	Software-Defined Networking
SDR	Software-Defined Radio
SHA	Secure Hash Algorithm
SLR	Systematic Literature Review
SNR	Signal-to-Noise Ratio
SS	Spreading Sequence
TC	Time Client
TLS	Transport Layer Security
TS	Time Server
TSCH	Time Slotted Channel Hopping
TX	Transmitter
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
WCS	Wireless Communication System
WSN	Wireless Sensor Network

INTRODUCTION

1 INTRODUCTION

1.1 CONTEXT AND MOTIVATION

In 2020, the Internet of Things (IoT) is a reality composed of billions of computing devices that interact with the physical world and the Internet. It lies at the frontier of the digital and physical realms, materialized in everyday objects and novel services that use them, like a light-bulb and a smart-home voice assistant. "IoT" is a broad term coined in 1999 [94] and overlaps with other concepts and technologies like *embedded* devices, Wireless Sensor Networks (WSNs), and *cyber-physical* systems. It signifies a reality made of heterogeneous devices, networks, and applications. In five years, IoT devices will double the human population [152] and this increase will also be qualitative: the relevance of the IoT in our lives and societies will keep growing[1].

The *networking* component is consistently present in IoT systems, despite their heterogeneous nature [52]. The concept of *synergy* is appropriate to describe them and justify this *fact*. A single IoT device may be constrained in terms of resources and capabilities. For example, it may perform a single sensing activity but lack persistent storage or a user interface. The possible in-node uses of the raw sensed data are limited. However, individual node limitations are less relevant when nodes are connected. In the former example, data collected at one connected node can be processed, stored, or shown at different nodes, specialized for those tasks. A distributed-computing service can use a potentially unlimited number of connected IoT devices to achieve a goal otherwise unachievable by any device. Without the nodes' *network* capabilities, there would be no synergy and no IoT.

A second *fact* is that cyber attackers are increasingly targeting IoT systems and devices [124]. Early deployments of IoT devices neglected

1 With how many IoT devices do *you* interact on a daily basis?

security aspects, and today many in-use household IoT devices still do not have proper security [7]. This led to the use of the alternate term of the "Internet of *Broken* Things". A well-known real attack example is the Mirai botnet that used more than 500k infected embedded Linux devices to launch a distributed Denial-of-Service (DoS) attack in 2016 [92]. As of today, a query for the string "`open ip camera`" in a web search engine will retrieve several indexing sites that list Internet-connected webcams and other devices that use default passwords and can be publicly accessed, a behavior probably unintended by their owners.

Once more, the *networking* component is a common denominator of most IoT attacks [9, 42, 116]. Understandably, as it enables distributed opportunities both for trusted parties and attackers.

The research community first highlighted the IoT security issue in the year 2010 [115, 192]. In the *constrained* IoT [24], the security challenges are exacerbated because the IoT devices have limited energy, processing power, or memory resources and security solutions must be specially tailored for their constraints. The Internet Engineering Task Force (IETF) open standardization body has been working on *constrained* IoT network security since the year 2011 [54]; it has already published security standards for the constrained IoT [153, 158], and has ongoing active research [73]. As of 2020, research and industrial communities better understand IoT attacks and its countermeasures, but many issues remain open[2] [159].

Meanwhile, in 2009 a disrupting cyber defense paradigm was explicitly proposed for the first time, the Moving Target Defense (MTD) [34]. MTD states that there is an *information* asymmetry between static systems and attackers. Attackers have unlimited time to study a system, find an exploit, develop and launch an attack. The information gathered does not expire; thus, defeat of a static system facing a persistent attacker is ineluctable. MTD's goal is to equilibrate this information asymmetry by limiting the time validity of -possibly leaked- system information. It proposes to achieve this goal by *proactively* modifying *components* of a system. Like the Heraclitean Fire [194], an MTD system is one and the same (i.e., provides its functional goals), yet it is in constant change. An attack relying on a particular instance of a component will be limited and mitigated by *time*. Fairly, the inherent -by design- capacity of a system to withstand or mitigate *unknown* attacks is increased by MTD—a desirable property in cyber *resilient* systems [146].

In 11 years, more than one hundred MTD-based techniques for computer systems have been proposed [28, 33, 100, 128, 160, 190, 203] and

2 As stated before, many deployed IoT nodes remain openly vulnerable, too.

some are used in most modern desktop and mobile operating systems [22]. However, MTDs targeted at IoT systems are not as widespread. Recent MTD surveys [33, 150, 203], identify around ten IoT-specific MTD techniques. MTD seems to be a promising technique for IoT systems, but there is this quantitative gap between MTD development on classical systems as compared to the IoT [202]. Reasonable doubts about the inadequacy of MTD techniques for the constrained IoT arise by this fact: *Why are there not more MTD for IoT proposals? Are existing MTD for IoT techniques even implementable in constrained IoT environments?*

The main goal of this dissertation is to improve the resilience of constrained IoT systems through the use of MTD techniques. We will focus on the *constrained* IoT and *network* components. In the next part of this manuscript, we go more in depth on the fundamental concepts presented in this introduction, establish the state of the art of MTD techniques for the constrained IoT, and validate that our dissertation goal is achievable.

1.2 RESEARCH GOAL AND QUESTIONS

We find clarifying to state the main research goal of this dissertation and the Research Questions (RQs) that arose while pursuing it.

Research Goal: To improve the resilience of constrained IoT systems through the use of MTD techniques.

Research Questions: The following **RQs** guide our dissertation, and need to be answered in order to achieve the stated goal,

RQ-1 : Is MTD for the constrained IoT possible?

RQ-2 : What is the status of MTD techniques for IoT?

RQ-3 : How to create *usable* and *secure* MTD techniques for the constrained IoT?

RQ-3.D1 : What are suitable Moving Parameters (MPs) in IoT systems?

RQ-3.D2 : How to move distributed MPs in IoT systems?

RQ-3.D3 : When to move the MP?

RQ-3.I1 : How to instantiate MTDs in concrete IoT use cases?

1.3 CONTRIBUTIONS

We summarize the contributions of this dissertation:

1. The first broad survey about MTD techniques for IoT. Besides, using an Systematic Literature Review (SLR) approach that can be scrutinized and replicated.

2. The definition of four novel entropy-related MTD *metrics* and their empirical application. Moreover, we provide the first empirical application of the Shannon Entropy MTD metric.

3. IANVS, a generic and modular MTD framework useful to instantiate concrete MTD techniques in constrained IoT systems.

4. The LATe protocol, a secure time synchronization protocol for IoT. We provided the *design*, and a formal-method security *evaluation*.

5. Two high-layer Network MTD techniques. First, an application-layer proposal for which we provide the *design*. Second, a transport-layer proposal for which we provide the *design*, a hardware *implementation*, and a theoretical and a hardware *evaluation*.

6. A physical-layer Network MTD Anti-Jamming (AJ) technique that mitigates insider-node jamming using Direct-Sequence Spread Spectrum (DSSS) radio modulation with independently- and Cryptographically Secure Pseudo-Random (CSPR)ly- generated Spreading Sequences (SSs). We provided the *design*, a simulation *implementation*, and an *evaluation* by simulation.

7. The first thorough study of the Cross-Correlation (CC) properties of uniformly random and CSPR sequence sets for Wireless Communication Systems (WCSs).

1.4 OUTLINE

The reminder of this manuscript is organized into three parts and a conclusive chapter.

Part i: *Exploration* is about discovering the field of MTD techniques for the constrained IoT. In Chapter 2, we present the MTD cyber defense paradigm, the context of the constrained IoT, and IoT-suitable IETF network security protocols. In Chapter 3, we establish the state of the art of MTD techniques for the constrained IoT following an SLR survey method. The results validate the MTD as a suitable technique for the

IoT and identify strengths and shortcomings in the existing techniques that our contributions will try to replicate and address, respectively.

Part ii: *Abstraction* is about the *design* of usable and secure MTD techniques for the constrained IoT. It is only composed of Chapter 4, where we provide guidelines for the design of MTD techniques by scrutinizing the topics of *what* IoT systems' components, *how*, and *when* to "move" them to instantiate MTDs. Notably, we present IANVS, a generic and modular framework that can be used to define concrete MTD techniques suitable for IoT.

Part iii: *Construction* is about the *instantiation* of usable and secure MTD techniques for the constrained IoT. In Chapter 5, we design the LATe synchronization protocol, a secure coarse-grained time synchronization protocol suitable for the constrained IoT; we provide a computer-aided formal method proof of its security claims, and discuss real-world issues and attacks not captured by the formal model. In Chapter 6, we instantiate two IANVS-based concrete Network MTD techniques motivated by a threat use case of a remote DoS attack: an application-layer MTD that targets Constrained Application Protocol (CoAP) resource's, and a transport-layer MTD that targets UDP port numbers; we implement and evaluate the UDP-ports technique in real IoT hardware, and we share the source code and raw-data results. In Chapter 7, we instantiate a IANVS-based Network physical layer MTD technique motivated by a threat use case of an insider node attacker (i.e., jammer). We use a DSSS AJ technique with the novelty of using independently- and CSPRly- generated Spreading Sequences. We implement and evaluate the proposed system and attacker-jammer in simulation using MATLAB. We explain the results by the CC properties of CSPR sequence sets, for which we provided an in-depth statistical and probabilistic study that was missing in the literature.

Finally, in Chapter 8, we provide some final conclusions and discuss future axes of research.

Part I

EXPLORATION

The first part of this memoir is about discovering the field of MTD techniques for the constrained IoT.

We present background and context, and establish the state of the art of MTD techniques for the constrained IoT.

2

BACKGROUND

2.1 INTRODUCTION

In this chapter, we introduce the general fields from which this thesis draws its motivation and challenges. First, we present the MTD cyber defense paradigm. Then, we present the constrained IoT and its security state of the art with a focus on IETF network standards.

2.2 THE MOVING TARGET DEFENSE PARADIGM

Moving Target Defense (MTD) is a cybersecurity paradigm whose goal is to mitigate unknown attacks by making components of a system inherently dynamic. This section reviews its rationale, history, and fundamental concepts focusing on concrete MTD proposals.

2.2.1 *Rationale and Resilience*

MTD RATIONALE MTD acknowledges the fact that vulnerabilities are present in any system and that cyber attackers with enough *time* will, eventually, find and exploit them. MTD's goal is to make the task of finding and exploiting a vulnerability more resource-consuming for the attackers, as compared to a non-MTD version of the system. MTD proposes to achieve this by constantly changing some of the system's components that, in turn, will also imply changing the system's *attack surface*.

The Attack Surface

"A system's attack surface is the subset of the system's resources that an attacker can use to attack the system." [75]

The attack surface 'movement' makes that the information an attacker gathered about the system is now limited in *time*. Thus, a discovered and then crafted attack at a given time t_0, might not work when the attacker launches it later at $t_0 + \Delta t$; because the target system is no longer the same: the attack surface changed, and the vector of attack may no longer be valid. MTD contrasts with systems' security measures that try to keep the attack surface small (i.e., attack surface reduction). MTD acknowledges that defenders do not entirely know the attack surface. Thus, those unknown vectors of attack can not be reduced in number (nor quality), but they can be 'moved'.

RESILIENCE MTD's founding document [34] states that the exploit of a vulnerability shall not prevent the MTD system to provide its service, and, even if exploited once, it will not be able to be exploited again. This is a desirable characteristic for *cyber resilient* systems.

Cyber Resilience

We adopt the definition given in NIST SP 800-160 Vol. 2 [146]:

> *cyber resilience*: "the ability to anticipate, withstand, recover from, and adapt to adverse conditions, stresses, attacks, or compromises on systems that include cyber resources."

This concept can be applied to a variety of entities including a mechanism, component, or system element, a system, a system-of-systems in a critical infrastructure sector, an organization, or a Nation. Whatsmore, cyber resilient systems are characterized by having security measures 'built in' as a foundational part of the architecture and design.

MTD is identified [146] as a suitable mechanism that can provide cyber resilience to systems. In this thesis, we focus in *constrained* IoT systems (See 2.3.1), and the implementation of MTD-based techniques to improve their cyber resilience. In the following, we provide a chronological-oriented presentation of MTD covering its sources of inspiration and milestones, and illustrate that is an active field of research with growing interest.

2.2.2 *A Brief History of MTD*

PRE-HISTORY The concept of changing system components to prevent unintended parties to disrupt its purpose is not new. Applications of this concept can be tracked in modern science at least to more than one hundred years ago in a patent of N. Tesla [174][1]. Particularly, the World War II (WWII) produced several advances in these kind of applications for communication systems (e.g., the Enigma machine, SIGSALY). This WWII setting also layed the foundations of modern cryptography-cryptology enabled by the Colossus computers [148]. Applications of the concept of defense through constant change in the Internet era can also be found at least since 2001 [84]. However, is not until the last decade that the term "Moving Target Defense" was coined and emerged as a cyberdefense paradigm.

MILESTONES AND PUBLICATIONS TREND The MTD paradigm was proposed in 2009 in the context of a U.S. National Cyber-Defense Summit [34] with support of the Federal Networking and Information

1 This is a precursor idea of the principles of Frequency-Hopping Spread-Spectrum wireless communication systems.

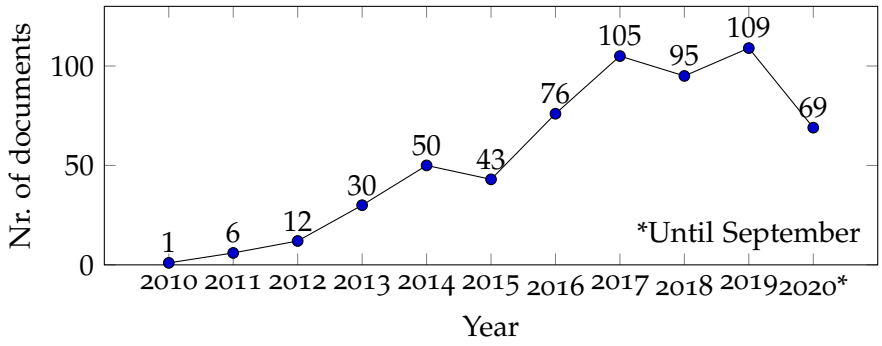


Figure 2.1: Number of MTD documents per year (Source: Scopus).

Technology Research and Development (NITRD) Program. Since then, MTD has been an important topic in cyberdefense with more than 500 scientific publications as of September 2020. Fig. 2.1 illustrates the number of scientific publications about MTD per year[2]. We can observe an upper-trend in the number of publications per year. We present a non-exhaustive chronology of MTD milestones in Table 2.1.

Table 2.1: MTD Highlighted Research Timeline

2009	MTD was proposed [34]
2011	MTD Book [75]
2012	MTD Book II [76]
2013	Survey from MIT Lincoln Lab 1st Ed. [127]
2014	1st ACM Workshop on MTD [77]
2018	Survey from MIT Lincoln Lab 2nd Ed. [190]
2020	7th ACM Workshop on MTD

In the following, we review fundamental concepts of the MTD paradigm with the focus on particular MTD techniques.

2 We used the Scopus meta-searcher and looked for the term in the title or abstract

2.2.3 *Fundamentals, Techniques, and Taxonomies*

The MTD literature can be divided into three fields [28]: *Theory* [67, 204–206], *Evaluation* [68, 132, 173, 198], and *Strategy* [28, 33, 100, 128, 160, 190, 203]. The *Strategy* field covers concrete MTD *techniques* that can be implemented in real systems. This thesis will largely focus on this field, which we review after the two others.

THEORY AND EVALUATION MTD *Theory* deals with mathematical-analytical theory, systems and attacker models, and theoretical tools to formally discuss about MTD. R. Zuang et al. defined three components that constitute the foundations of MTD *Theory*: MTD Systems Theory [206], a Cyber Attack Theory [205], and their interaction [204].

An MTD system

Some definitions from R. Zuang et al. [205, 206], in order to define the concept of an MTD system[a]:

- A *configuration parameter* is a unit of configuration information that can take on a *value* based on its *type*.
- A *configuration parameter type*, is a label identifiable with the *domain* of possible *values* that the *configuration parameter* can assume.
- A *configuration state*, is a unique assignment of *value(s)* from the *domain* of the configuration parameter type to a *configuration parameter*.
- A *configurable system*, is a system that is in a given *configuration state*, but this state can change based on a set of *configuration actions*. However, not all *configuration states* might be *valid*, because the system has functional *goals* and *policies* to comply to.

Finally, an *MTD system* is a *configurable system* that can adapt its configuration during execution, and it is always in a *valid configuration state* that can achieve the overall *goals* of the system.

a We prioritize narrative definitions, and point the reader to R. Zuang et al. [205, 206] for the formal definitions.

The MTD *Evaluaton* field deals with methods to evaluate and quantify the effectiveness of MTD systems. This field of research includes mostly the definition of *metrics* that allow not only the assessment of the effectiveness of a particular MTD system's *technique* but also allow the

comparison among different ones. The field has practical importance, because *metrics* can be used to guide the design and implementation of novel MTD systems, or modify existing ones to increase their efficiency against specific attack types.

MTD TECHNIQUES AND TAXONOMIES Finally, the MTD *Strategy* field covers concrete MTD *techniques* that can be implemented in systems. We use the terms *strategy* and *technique* interchangeably throughout this work, but prioritize the latter. There are more than 100 distinct MTD techniques, and several survey publications [28, 33, 100, 128, 160, 190, 203]. In this subsection, we focus on general design principles shared by them, and in a widely used taxonomy of techniques based on the MP element.

The Moving Parameter

Using MTD theory's nomenclature, we define the MP(s) as the *configuration parameter(s)* of an MTD system.

An MTD technique needs to define three fundamental design questions: WHAT, HOW, and WHEN to *move*. These principles were first proposed by Cai et al. [28], and can be defined as follows:

- WHAT to move determines the component(s) of the system to which the technique will be applied. In other words, the MP(s).
- HOW to move is about the methods for (i) define valid states of the MP, and (ii) chose one valid state for the system. MTD techniques use three types of methods: Shuffling (randomization), Diversification, and Redundancy-based.
- WHEN to move is about applying the state change, i.e., the decision process that triggers the MP value change. The literature identifies three types of decision processes: Time, Event, and Hybrid-based

To conclude this subsection, we present a taxonomy for MTD techniques based on the system layer to which the MP pertains. It was first proposed by Okhravi et. al [127, 128]. The taxonomy is the following:

1. *Network*. Techniques that change the network properties, e.g., protocols, addresses.
2. *Platform*. Techniques that change the computing platform properties, e.g., CPU architecture, OS, virtual machine instance.
3. *Runtime Environment*. Techniques that change the execution environment dynamically, e.g., RAM addresses, instruction set.

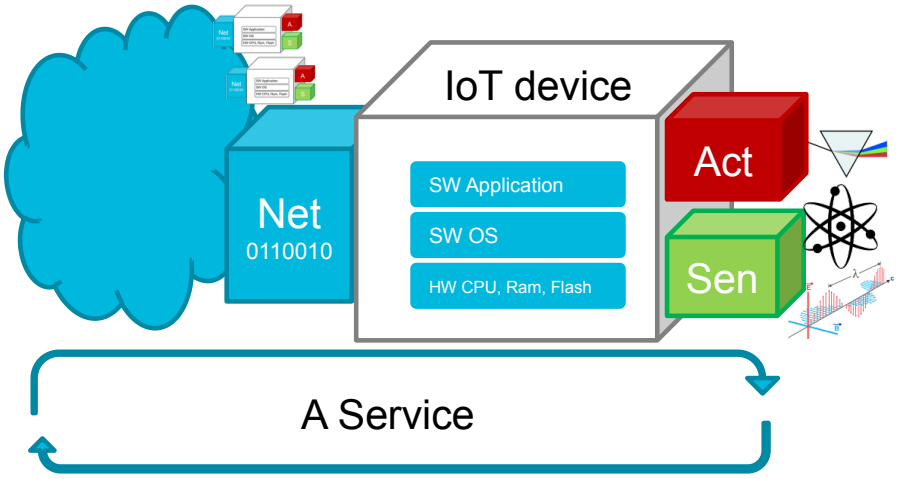


Figure 2.2: An IoT system.

4. *Software*. Techniques that change an application's binary code, e.g., binary objects shuffling, application diversification.

5. *Data*. Techniques that changes the format, encoding or representation of application data, i.e., same semantics with different syntax.

This taxonomy, based on the MP's system layer, is widely used in MTD literature. We will use it to categorize existing MTD techniques for the constrained IoT, and to reflect on which domains to propose novel ones.

2.3 THE CONSTRAINED INTERNET OF THINGS

2.3.1 *Definitions and Context*

The term "Internet of Things" was coined in 1999 [94], and is used to include a wide variety of objects that have internet connectivity. These objects interact with our physical world by means of *sensors* and *actuators*. The *networking* capability allows for remote supervision and control of these objects, and the realization of distributed services that may require many of them. These objects, or *devices*, act as interfaces between the physical and digital worlds. Fig. 2.2 illustrates our vision of an IoT *system* as many *devices* providing a *service*.

However, not all *things* are made equal and, particularly, resourceful-enough IoT devices that can run off-the-shelf software are out of scope of this work. In this thesis, we will focus on the *constrained* IoT.

The constrained Internet of Things

A *constrained IoT device* is a computing hardware device with network capabilities and limited energy, processing power, or memory resources. *Devices* that form a netwok are called *nodes*.

A *constrained IoT network* is a telecommunications network constituted of IoT nodes that exhibits constraints in terms of bandwidth, reliability, or topology stability. The nodes are not necessarily constrained.

The constrained IoT imposes novel challenges at hardware and software levels. For example, well-established network protocols, like HTTPS, simply can not run in a constrained device. Adaptations or novel protocols are needed for the constrained world.

The Moore's law in the constrained IoT context is "used" not to increase the computing power of the devices, but to lower their cost.

THE IOT ENABLES NOVEL SERVICES These challenges and constraints come with a desirable trade-off. The limitations of a constrained IoT device make the monetary cost per unit low. This fact makes the creation of networks with a large number of these types of nodes economically viable. Which, in turn, enables the realization of novel distributed applications that were otherwise not possible. For example, a system with hundreds of soil moist sensing IoT devices that helps farmers in the management of the irrigation process of their fields. *Smart* agriculture, industry, transportation, and cities are some examples of the novel fields of applications enabled by the IoT.

TERMINOLOGY FOR CONSTRAINED-NODE NETWORKS The IETF defines in RFC7228 [24] a terminology for constrained-node networks. A relevant output of the document is the distinction of three different *classes of constrained devices*, as shown in Table 2.2.

Table 2.2: Classes of devices according to RFC7228 [24] (KiB = 1024 bytes).

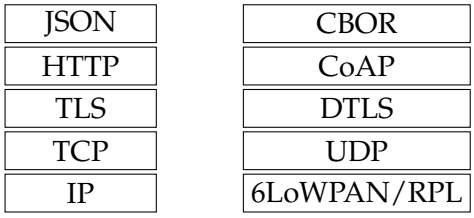
Device Class	RAM (KiB)	Flash (KiB)
Class 0	$<<$ 10	$<<$ 100
Class 1	$\approx$ 10	$\approx$100
Class 2	100	250

These classes depend on measurable quantities that allow us to determine unequivocally to which class a node pertains. In regards to what each class entails in terms of networking-protocols capabilities, Class 2 devices have not significant difficulties implementing standard protocols, while Class 0 devices are assumed not to have the resources required to communicate directly with the Internet in a *secure* manner.

Finally, Class 1 devices are the ones assumed on new IoT-tailored network protocols, and are the ones that present a challenge. In other words, current IETF IoT proposals target devices with around 10 KiB of RAM and 100 KiB of Flash. In this thesis, we aim for our resiliency-improving proposals to be usable in Class 1 constrained devices. The network aspect of the IoT plays a significant role in this thesis, which is why we devote special attention to the work done by the IETF. In the following subsection, we review the network protocol stack of the IoT.

2.3.2 *Connectivity and Interoperability: Standards*

In the IoT, heterogeneous devices and networks are the norm. Two properties fundamental to make distributed and heterogeneous systems to work together are *connectivity* and *interoperability*. Open network standards allow devices from different hardware manufacturers or software developers to achieve those properties. In a nutshell, connectivity is provided by the IPv6 protocol [40], and interoperability is provided by application-layer protocols like the CoAP [163]. The IETF is a standardization body responsible for most of the network open standards that enable the Internet and the IoT. In Fig. 2.3, we show side by side the IETF's standard Internet and the IoT network protocol stacks.



(a) standard **(b) IoT**

Figure 2.3: IETF protocols stack for: (a) standard Internet, and (b) IoT.

2.3.2.1 *Connectivity*

The first IoT-related work at IETF dates back to 2005 with the creation of the IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) Working Group (WG). The 6LoWPAN WG provided the first adaptation of IPv6 for constrained-node networks in RFC4944: *Transmission of IPv6 Packets over IEEE 802.15.4 Networks* [119], published in 2007. This document specifies a frame format, local-link address forming, and a compression scheme to deliver IPv6 packets in IEEE 802.15.4 networks. Other RFCs from the WG update and expand this document, and together they constitute the 6LoWPAN suite of protocols.

Mesh networks are common in 6LoWPANs and ad-hoc network-layer route formation is a non trivial problem that needed solutions adapted for the IoT. The ROLL (Routing Over Low power and Lossy networks) WG addressed the routing problem on 6LoWPANs, with the design of The IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [5] , published in 2012. RPL generates routes optimized for traffic to or from *root* nodes, or *sinks*. It creates a logical topology in the form of one or more Destination Oriented Directed Acyclic Graphs (DODAGs). RPL allows a high degree of customization. Among other parameters, novel metrics that affect topology formation can be specified in the form of RPL's Objective Functions. The 6LoWPAN and RPL documents provided the foundations of the constrained IoT connectivity by allowing IPv6 on Low-Power and Lossy Networks.

On top of the IP stack, the UDP [136] is widely used as the transport-layer protocol. Even if not designed for the IoT, UDP is lightweight enough to be used as is. The security-related Datagram Transport Layer Security (DTLS) [145] is the UDP-enabled equivalent of TLS that is used to secure most of standard Internet traffic. DTLS and other IoT security-related protocols will be reviewed in the next subsection.

Data and Information

Data are raw bits in the digital world, and *connectivity* allows *data* from node *A* to reach a node *B*, potentially located in the other side of the physical world. But, *data* has no use unless *B* can meaningfully interpret it. *Information* is *data* with a context that gives it a meaning, a semantic interpretation of the raw bits. The exchange of *information* between nodes is what we understand by *interoperability* and, in the context of the TCP/IP network layers model, it can be achieved at the application layer.

2.3.2.2 *Interoperability*

Application-layer network protocols are about the exchange of *information* in the context of a specific domain or application. Is at this upper network layer where *interoperability* will be achieved. In particular, the REpresentational State Transfer (REST) [49] architectural style, widely used in the World Wide Web, plays an important role in achieving interoperability for the IoT.

The RESTful Paradigm

RESTful services consist of *resources* hosted by a network node (a *server*). A *resource* has a given *state*, and a defined set of *resource operations* can generate *state transitions*. These *operations* can be triggered by a message from a remote node (a *client*), thanks to the use of Uniform Resource Identifiers (URIs). An URI uniquely identifies a *resource* over the network, e.g., `coap://[NodeIPv6Address]:5683/myresouce`. Some common RESTful operations are:

- GET: Retrieve a representation of the current state of the resource in a response message.
- PUT: Store the representation contained in the current message as the new state of the resoruce.
- DELETE: Delete the state of the resource.

Other operations are POST and PATCH.

The IETF's Constrained RESTful Environments (CoRE) WG, created in 2010, deals with application-level goals. It provides solutions for resource-oriented applications intended to run on constrained nodes and networks. One of his most valuable outputs has been The Constrained Application Protocol (CoAP) [163] (RFC 7252), published in 2014.

CoAP is a RESTful protocol inspired by HTTP, but information is encoded in binary form as opposed to human-readable text. CoAP defines *messages* that contain a RESTful *operation*, *options* (e.g. the URI of a resource), and a *payload*. Also, unlike HTTP, CoAP can be transported over datagram-oriented channels, like UDP over IP, where data can be lost or arrive out-of-order, because it provides application-layer ACKs and a request-response matching mechanism. On top of CoAP, the message *payload* will contain specific application-data, for example a sensor temperature value, but how to represent that kind of information is out-of-scope of CoAP.

The CBOR [25] is a binary data format inspired by JSON and provides a compact representation of most common data types used at Internet standards, like an unsigned integer or a text string. It also supports map structures (key-value pairs) and arrays. CBOR is the building block of other application-layer data schemes like CBOR Object Signing and Encryption (COSE) [153], or Sensor Measurement Lists (SenML) [78].

2.3.2.3 *What about Security?*

The IPv6 adaptation 6LoWPAN, the RPL routing protocol, the CoAP application protocol, and the COSE encoding scheme have become the pillars of *connectivity* and *interoperability* in the IoT. However, excepting the brief mentions of DTLS and COSE, we avoided references to security-related protocols. This has a narrative and also a chronological reason. As with the evolution of the standard Internet, the IoT was first developed and deployed without security in mind. It was not until IoT devices were already in the wild, that weak or complete lack of security started to be an issue [9, 42]. The first peer-reviewed academic publications about IoT security appear in the year 2010 [115, 192], and the first IETF drafts in 2011 [54][3]. In the next subsection, we present a panorama of the IETF's IoT security-related network protocols.

2.4 IETF NETWORK SECURITY FOR THE IOT

In this section, we review the most prominent IoT-friendly security proposals designed and published in the context of the IETF. These include protocols, data formats, and frameworks. We use a bottom-up layered approach. Some of these proposals serve as building blocks for more complex security protocols.

2.4.1 *Network-layer*

This is the lowest layer at which the IETF is competent. Below is the data link layer, which is the domain of other standardization bodies that define physical standards and link layer frame formats, such as IEEE or Bluetooth. Notwithstanding that, IETF has some cross-layer work, particularly within the 6TiSCH WG that deals with the Time-Slotted Channel Hopping (TSCH) mode of the IEEE 802.15.4 wireless standard. They define a work-in-progress framework that allows for secure node joining mechanisms in those networks [186]. This framework re-uses an application-layer solution to secure CoAP message exchanges. Another example is the Extensible Authentication Protocol (EAP) [185], an authentication framework that runs on top of the link layer and supports multiple authentication methods. IoT-friendly CoAP-based EAP exchanges have been proposed [104], but the current status of standardization efforts is uncertain. Both are examples of how to use basic security solutions to solve more complex and specific security issues.

3 Became RFC8576 in 2019 [55]

LIGHTWEIGHT IPSEC Network-layer security is based on adaptations of the IPsec protocol suite: the IKEv2 [48] key exchange mechanism, and the Encapsulating Security Payload (ESP) [83] and Authentication Header (AH) [82] formats. The Light-Weight Implementation Guidance (LWIG) WG is the home for this efforts. The Minimal Internet Key Exchange Version 2 (IKEv2) Initiator Implementation (RFC7815) [87] presents a lightweight version of the IKEv2 protocol. This protocol is used for performing mutual authentication between two nodes, and establishing and maintaining security associations (i.e., fresh cryptographic keys). Once both end nodes have a security association, they can encrypt and authenticate data using ESP (both security services), or AH (only authentication) headers. Both minimal adaptations of ESP and AH have standardization efforts [117, 141], but they did not reach standard status and are currently inactive. The future of IPsec encryption for IoT is not clear. Current security standardization efforts are focused on higher-layers.

2.4.2 *Transport-layer*

The classic Internet has one uncontested protocol that secures most of its traffic: The Transport Layer Security (TLS) [142, 144]. It provides the security services of data confidentiality, integrity, and authentication, and has mechanisms to detect replay attacks, among other features. It runs on top of TCP and, for example, is used to secure HTTP (i.e., HTTPS). Because TLS runs over TCP, it assumes that data packets are not lost and that arrive in order. However, these guarantees are not assured by the UDP protocol, or other non-reliable channels, used in most IoT systems. Thus, TLS can not run on top of UDP. The Datagram Transport Layer Security (DTLS) [145] protocol was designed to provide the same security guarantees as TLS, while overcoming the packet loss and reordering problems of datagram-oriented transports. DTLS is a suitable candidate to secure IoT traffic.

HTTPS stands for most of the traffic over the Internet: Netflix, Youtube, Google, Facebook, and Amazon use it.

DTLS IN A NUTSHELL DTLS version 1.2 [145] is a client-server protocol and has a layered design. It is composed of a base protocol called the *record layer*, and other four sub-protocols that run on top of it. The *record layer* specifies a data format, and is in charge of taking a *message* from the higher-layer sub-protocols, fragmenting the data into manageable datagrams, optionally compressing it, applying a MAC and encrypting the data, and transmitting it. Received data is, decrypted, verified, decompressed, reassembled, and then delivered to higher-layer sub-protocols. The four sub-protocols that run on top of the *record layer* are: the *Handshake* protocol, the *Change Cipher Spec* protocol, the *Alert* protocol, and the *application data* protocol. The *Alert*

protocol is used to report error conditions among DTLS client-server pairs, and the *Change Cipher Spec* protocol consist of a single message that signals transitions in ciphering strategies. The DTLS *Handshake* Protocol is responsible for negotiating a *session* that, notably, consists of a *session identifier*, a *cipher spec*, and a *master secret*. It also includes a stateless cookie exchange to prevent DoS attacks, and handles message loss, retransmission, and fragmentation.

The DTLS Handshake Protocol

The DTLS *Handshake* Protocol involves the following steps (text quoted from the standards[144, 145]):

- Exchange hello messages to agree on algorithms, exchange random values, the stateless cookie, and check for session resumption.
- Exchange the necessary cryptographic parameters to allow the client and server to agree on a premaster secret.
- Exchange certificates and cryptographic information to allow the client and server to authenticate themselves.
- Generate a master secret from the premaster secret and exchanged random values.
- Provide security parameters to the record layer.
- Allow the client and server to verify that their peer has calculated the same security parameters and that the handshake occurred without tampering by an attacker.

  The process also includes retransmission timers to handle message loss.

The items agreed during the *Handshake* protocol are then used to create security parameters, i.e., the *session*, used by the *record layer* when protecting *application data*.

DTLS, IOT-FLAVORED The DICE (DTLS In Constrained Environments) IETF WG was created in 2013. It was the first IoT security-oriented WG, and its sole purpose was profiling DTLS for IoT. It produced the standard *TLS/DTLS Profiles for the Internet of Things* [180] (RFC7925). This document presents two TLS/DTLS profiles, one for constrained clients, and another for constrained servers. The TLS/DTLS standards have many configuration options and protocol extensions, and this *profile* document explicit what choices to make to best support an IoT environment. The documents specifies mandatory-to-implement

functionality, and recommends what *cipher suites* are suitable for the IoT.

(D)TLS Cipher suites and Crypto-agility

DEFINITION The DTLS *record layer* protocol requires the agreement over a *suite of algorithms* to protect message exchanges. This *cipher suite* is selected by the server during the *Handshake* protocol, and it defines the following information[180]:

- Authentication and key exchange algorithm (e.g., PSK)
- Cipher and key length (e.g., AES with 128-bit keys)
- Mode of operation (e.g, CCM mode for AES)
- Hash algorithm for integrity protection (e.g., SHA-based HMACs)
- Hash algorithm for use with pseudorandom functions (e.g., SHA-based HMACs)
- Misc information (e.g., length of authentication tags)

CRYPTO-AGILITY DTLS and TLS support many *cipher suites*, which are publicly registered[a], and specified in their own standards. New cipher suites can be defined, and broken-cryptography ones deprecated, to reflect state-of-the-art cryptography.

a https://www.iana.org/assignments/tls-parameters/tls-parameters.xhtml#tls-parameters-4

Other DTLS parameters are configurable, and are also discussed. For example, *compression* is not needed in IoT (because application protocols are already size-optimized), suggestions about when to implement or not *keep-alive* messages, and determining the specific values to set in the *timeouts* mechanisms.

COAP, DTLS, AND CIPHER SUITES The CoAP standard defines a binding for DTLS. It specifies four modes of CoAP secure operation that depends on the pre-provisioned cyptographic material in the node: none, Pre-Shared Keys (PSKs), Raw Public Keys (RPKs), and certificates. The most constrained nodes that support DTLS security will be provisioned with PSK. They need to support the *cipher suite* `TLS_PSK_WITH_AES_128_CCM_8` [113]. This *cipher suite* makes use of the default TLS 1.2 Pseudo-Random Function (PRF), which uses an Hashed Message Authentication Code (HMAC) with the Secure Hash Algorithm (SHA)-256 hash function. For RPK, the *cipher suite* is `TLS_ECDHE_ECDSA_WITH_AES_`-

128_CCM_8 [114], which uses elliptic-curve cryptography. Finally, for certificates, the mandatory *cipher suite* is the same as RPK, but the key will be wrapped in a X.509 v3 [23] certificate. After the DTLS authentication credentials have been used on the *Handshake* protocol, all *cipher suites* encrypt with the same symmetric algorithm: Advanced Encryption Standard (AES) in Counter with CBC-MAC (CCM) mode, with a 128-bit key length, and an 8-byte authentication tag. Once encrypted, the added DTLS per-datagram overhead is of 13 bytes, without counting the tag.

CTLS: COMPACT TLS A work-in-progress document [143] defines a compact version of TLS v1.3 (cTLS) within the TLS WG. It is aimed at IoT systems that support reliable transport, like TCP. It is not interoperable with TLS v1.3. It is designed to use minimal bandwidth over the network. In order to do so, they use four techniques: omitting unnecessary values inherited from previous versions of TLS, omitting fields and handshake messages required for preserving backwards-compatibility, more compact encodings (e.g., using variable-length integers), and a template-based mechanism that allows for the creation of application-specific versions of TLS that omit unnecessary values.

THE LIMITS OF TRANSPORT-LAYER SECURITY One of the drawbacks of DTLS, or any transport-or-lower-layer security solution, is that end-to-end security cannot be achieved in the presence of proxies (e.g., CoAP proxies for forwarding, or caching), because the secure channel will be terminated at the proxy. This motivated the definition of application-layer IoT security solutions at the IETF that we study in the following subsection.

2.4.3 *Application-layer*

The application layer is the highest layer of abstraction in the TCP/IP and OSI network models. The heterogeneous IoT ecosystem has risen the interest in security solutions at this layer, because security services can be maintained end-to-end, independently of the heterogeneity of lower-layers[4], and even application-layer limitations (e.g., proxies). The lower-layer protocols we studied are *connection* oriented, meaning that first a secure channel establishment is made (e.g., DTLS *Handshake*, or IPSec IKEv2), and then application data is sent over the secure channel. Application-layer solutions are not necessarily connection oriented, and the security properties can be set on a per-message basis.

4 This statement is always true for any layer, but an advantage of app-layer security is that it can also be stacked directly on top of a layer-2 technology

Application-layer security is also referred as *object security*, because the security services of the data are self-contained. An independent unity of secured data is very desirable for many IoT applications: this data can be cached, proxied over different lower-layers, and it will still maintain its end-to-end security properties.

COSE: OBJECT SECURITY The CBOR Object Signing and Encryption (COSE) [153] (RFC 8152) is the pillar of *compact* object security. COSE defines how to create and process encryption, signatures, and MACs, using CBOR for data serialization. It also defines how to transport cryptographic keys. COSE was designed taking in account the constraints of nodes (e.g., a compact implementation stack that uses few node resources), and networks (e.g., compact messages). A COSE secure object can contain other binary objects, and thus its possible to compose complex objects. COSE has more flexible security properties than DTLS, but at the cost of more overhead per datagram (excluding the DTLS Handshake). COSE is not a network protocol, is secure message format and flexibility, makes it suitable to be used by other applications or cryptographic protocols that need to guarantee basic security services to the exchange of messages. For example, in this thesis (Ch. 5), we propose a secure network time synchronization protocol and use CBOR Object Signing and Encryption (COSE) to provide authentication to the messages exchanged.

COSE is inspired by Javascript Object Signing and Encryption (JOSE), that uses JSON for data serialization and its not compact

OSCORE: OBJECT SECURITY FOR COAP COSE *object security* is used to construct other security solutions. The most important example at the IETF is Object Security for Constrained RESTful Environments (OSCORE) [158] (RFC8613). OSCORE provides end-to-end encryption, integrity, and replay protection of CoAP messages. A CoAP message field can be encrypted and integrity protected (Class E field), integrity protected only (Class I), or unprotected (Class U). The CoAP payload is always encrypted and integrity protected. Class E message fields are transported in the ciphertext of the COSE object in the OSCORE message, these message fields are not visible to proxies and are called *Inner*. The Class I message fields are part of the additional authenticated data of the COSE object and Class U are unprotected. The Class I and Class U message fields are transferred in the header or options part of the OSCORE message, which is visible to proxies, and are called *Outer*. OSCORE also provides a secure binding between CoAP request and response messages, and freshness of requests and responses (i.e., anti replay-attack mechanisms). OSCORE requires a shared *security context* established between the CoAP client and server in order to process the COSE objects. The parameters of the security context are derived from a set of input parameters that are assumed to be pre-established, those parameters are: a *Master Secret*, a Sender ID, and a Recipient ID. It is

out-of-scope of OSCORE how to derive a Master Secret, but it requires that, in addition to being secret, it has a good amount of randomness as specified in [1]. Authenticated Key Exchange (AKE) mechanisms can be used to comply with these requirements.

EDHOC: AN AUTHENTICATED KEY ESTABLISHMENT OVER COSE Ephemeral Diffie-Hellman Over COSE (EDHOC) Ephemeral Diffie-Hellman Over COSE (EDHOC) [157] is a work-in-progress that provides a Elliptic Curve Diffie-Hellman (ECDH) key exchange with ephemeral keys suitable for the constrained IoT. The ECDH exchange and key derivation are based on the National Institute of Standards and Technology (NIST) SP-800-56A specification [15], the SIGMA protocol [96], and the HMAC-based Key Derivation Function (HKDF) specified in RFC5869 [95]. EDHOC provides mutual entity authentication, perfect forward secrecy, and identity protection. Its main use case is to establish an OSCORE *security context*. *Authentication* is based on cryptographic material established out of band, e.g. from a trusted third party.

Authentication and Authorization for IoT

The Authentication and Authorization for Constrained Environments (ACE) WG aims to produce a standardized solution for *authentication* and *authorization*, and enable authorized RESTful access to *resources* identified by a URI, and hosted on a resource server in constrained environments. The most relevant output is the ACE using the OAuth 2.0 Framework (ACE-OAuth) [155]. The ACE-OAuth framework defines profiles to be used over many network-layers, and thus it can be defined as a vertical/-transversal proposal, and not exclusively application-layer.

2.4.4 *A distributed-security toolkit*

In this subsection, we presented the IETF publications suitable for the *constrained* IoT that offer the basic security services of data confidentiality, integrity, and authentication. As we explored, these services could be provided at different network layers, and the choice in a real IoT system will depend on its particularities.

GOING HIGH, TO GO ANYWHERE Physical an link layer solutions were not explored, not because their are not relevant for IoT systems, but because finding convergence in lower layers is more difficult. In lower layers, closer to the physical world, there is less abstraction in-

volved. Thus, the solutions are dependent on the particular technology. In this section, we highlighted the higher network layers, because the starting point for the particular contributions of this thesis is an abstract framework (Ch. 4.3) that requires distributed security solutions. Thus, in order to be as generic as possible, the higher the network abstraction the better, because only same-or-upper-layer issues can disrupt the end-to-end *information security* properties of our system[5]. We prioritize application-layer solutions to achieve information security goals. However, our proposed framework can be used to improve an IoT system's resilience at *any* network layer[6]. In one of the last chapters of this work (Ch. 7), we secure the physical layer of a wireless IoT system.

STANDING ON THE SHOULDERS OF GIANTS We also reviewed less generic cryptographic protocols, like OSCORE and EDHOC. These are examples of how simpler components, like COSE, are used to incrementally construct more complex or tailored security solutions suitable for the *constrained* IoT. Another of the design principles of the IETF is *not reinventing the wheel*, which was exemplified by the TLS-adapted DTLS, and DTLS for IoT profile. Particularly in the security domain, is preferable to use cryptographically-proven security solutions instead of custom-made ones. Over the course of this thesis, we advocated for those principles. We re-use existing security protocols as much as possible; and, if we define custom protocols, re-use existing building blocks such as COSE object security.

In this thesis, we put the IETF's standardized distributed security toolkit at the service of novel MTD techniques in order to improve the resilience of *constrained* IoT systems.

2.5 CONCLUSION

This chapter introduced three essential domains that are transversal to the rest of the thesis: constrained IoT systems, network security protocols suitable for IoT, and the MTD paradigm. The relationship between the first two domains is explicit, but linking them with the MTD domain is not as straightforward, and this task will account for most of this thesis.

IOT RESILIENCE THROUGH MTD Improving the resilience of constrained IoT systems is the primary motivation of this thesis. The MTD

5 Lower-layer attacks, e.g., DoS, can disrupt the *flow* of information of the system, but countermeasures have to be taken in the same layer as the attack.

6 Or even at not distributed components of the system.

paradigm is acknowledged as a suitable cyber defense tool to design and instantiate resilient systems [146]. Several MTD techniques have been proposed in the literature, and some have improved the overall security of current computer systems. As an example, Address Space Layout Randomization (ASLR) techniques [22] are used in all modern general-purpose OSs. Thus, improving constrained IoT systems' resilience through the applicability of MTDs is a reasonable working hypothesis.

EXPLORING MTD FOR IOT However, MTD applied to IoT systems is a subject not extensively explored yet. Previous MTD surveys identify more than 100 different techniques, but only six that are suitable for the IoT. Only two of the most recent surveys [33, 203] explicitly mention the IoT. This situation, i.e., on the one hand a good potential of the MTD paradigm to improve system's security, but on the other hand a concrete lack of techniques for the IoT, leads to the formulation of the first RQ:

> **RQ-1**: Is MTD for the constrained IoT possible?

In the following chapter, we will present an answer to this and other RQs and derived sub-questions. We base our assertions on the evidence provided by existing MTD for IoT techniques.

3

STATE OF THE ART OF MTD TECHNIQUES FOR IOT

πάντα χωρεῖ καὶ οὐδὲν μένει
"everything changes and nothing stands still"

Heraclitus, quoted by Socrates
(Plato, *Cratylus* 402a, c. 360 B.C.E)

3.1 INTRODUCTION

In the previous chapter, we introduced the MTD cyberdefense paradigm, but the question of whether it is suitable for the constrained IoT (RQ-1) remained open. This chapter's primary purpose is to validate the feasibility of MTD as a cybersecurity technique for constrained IoT systems.

In order to answer RQ-1, we take an evidence-based approach. Existing MTD techniques for the constrained IoT (i.e., primary studies) provide the evidence. We use a Systematic Literature Review (SLR) method, we enforce the guidelines by Kitchenham et al. [86], to search, select, analyze, and synthesize existing techniques. Using the evidence the primary studies provide, we evaluate them in terms of their *real-world deployability* and *security foundations*. In addition, we define and use entropy-related metrics to categorize them. To the best of our knowledge, this is the first MTD survey to empirically use Shannon's entropy metric for the studied MTD techniques.

The results allow us to answer RQ-1 in the affirmative, and also provide an insight of the state of the art of the field, tackling RQ-2:

RQ-2: What is the status of MTD techniques for IoT?

Notably, we identified a predominance of *network* techniques, and a lack of sound security foundations in most of the techniques.

Contributions of this Chapter

- A Systematic Literature Review [86] of MTD techniques for the constrained IoT. This is the first MTD survey focused on IoT. Furthermore, two-thirds of the techniques were not previously identified by MTD literature.
- An evidence-based assessment of the security status of the techniques and validation of the feasibility of MTD for IoT. To the best of our knowledge, this is the first MTD review to focus on the cryptographic primitives of the techniques.
- The definition of four new entropy-related metrics and their application. Moreover, this is the first MTD review to make practical use of Shannon's entropy as a metric. The metrics have applications beyond the scope of this review.

3.2 MOTIVATIONS

IN-DEPTH IOT SURVEY LACKING In 11 years since the inception of MTD, more than 100 distinct techniques have been proposed [190]. Also, several MTD techniques survey articles have been published [28, 33, 100, 128, 160, 190, 203]. However, limited work has been published about MTD targeted at IoT systems. The most recent peer-reviewed MTD surveys [33, 203], identify less than five IoT-specific MTD techniques. A recent book chapter [150] focuses on MTD network techniques for IoT and effectively identifies around a dozen techniques. Even if MTD for IoT is an acknowledged promising field of study [150, 203], there is still a lack of an in-deep survey of its state of the art.

A SYSTEMATIC APPROACH In this chapter, we present a survey of MTD for IoT as thorough and transparent as possible. We also intend to provide evidence-based justification for MTD as a suitable cyber-defense paradigm for the IoT and not a mere *promising* or *future work* technique. Hence, this survey uses an SLR approach, widely employed in the Medical science fields, but adapted for the Software Engineering fields by P. Brereton, B.A. Kitchenham et al. [26]. This method focuses on defining and documenting the survey process (e.g., the search databases and strings, inclusion-exclusion criteria, data extraction methods), making it as transparent as possible, and reproducible by independent researchers. The methodology aims at producing evidence-based answers to clearly defined RQs.

3.3 RELATED WORK

The first survey of MTD techniques appeared in 2013 [128]. Since then, many have been published [28, 33, 100, 128, 160, 190, 203] identifying around 100 distinct techniques. However, IoT applicability is not considered in most of them. Indeed, only two peer-reviewed recent surveys consider MTD for IoT.

IOT IN MTD SURVEYS Zheng et al. [203] has a sub-subsection of *lightweight MTD*; it identified two techniques and mentioned that more MTD techniques for resource-constrained devices are required. Cho et al. [33] has a sub-subsection of *Internet-of-Things* within a discussion of application domains for MTD; it identified four techniques and acknowledged that MTD seems promising for IoT systems but with some limitations when compared with conventional MTD. A recent book chapter by Saputro et al. [150] focuses on the applicability of MTD for IoT applications. It extensively discusses general concepts of MTD, IoT, and Software-Defined Networking (SDN). It dedicates an entire section to MTD for IoT techniques. They focus on network-category techniques and propose a subdivision of the network taxonomy. They identify around a dozen MTD for IoT techniques from the Network category. They highlight the potential of SDN-based solutions and discuss that the military and industrial IoT applications may benefit from it.

SYSTEMATIC LITERATURE REVIEWS In respect to the SLR approach, none of the aforementioned surveys used it. However, Torquato et. al [177] conducted a systematic mapping study[1] of *MTD in cloud computing*. Hosseinzadeh et al. [71] performed a SLR of *Diversification and obfuscation techniques for software security*, a broader topic than MTD.

3.4 METRICS: DEFINITION

In this section, we define the metrics that will be employed in the SLR. Several metrics for MTD have been proposed [68, 132, 198]. However, in general, they are of difficult applicability to concrete and heterogeneous strategies. In this SLR, we use metrics related to the entropy of the MP. These metrics have the property of being applicable to the surveyed MTD techniques with a reasonable effort.

1 There are differences between a SLR and a *systematic mapping study*. A mapping study consists of broad research questions, and its main output is to classify literature in some way. A SLR has a narrower subject, and fewer studies will be included. Sometimes, the term *systematic review* is used for what is technically a *mapping study*.

3.4.1 *Shannon Entropy of the Moving Parameter*

This metric is based on the *maximum* Shannon's entropy of the MP of an MTD technique. Works by Zhuang et al. [206], and Hobson et al. [67] already used Shannon's entropy concept for MTD systems in a theoretical way. In this section, we present our own approach but refer the reader to those works for more information.

DEFINITION Let X be a MP of a system, x be a valid state for X, and E be the set of all valid states $\{x_1, x_2, ..., x_n\}$. We can use Shannon's information entropy concepts if we define X as a discrete Random Variable (RV) with possible values $\{x_1, x_2, ..., x_n\}$ and a probability mass function $P(X)$. The Shannon Entropy in bits of the MP X is defined as:

$$H(X) = -\sum_{i=1}^{n} P(X = x_i) \log_2 P(X = x_i) \tag{3.1}$$

Large values of $H(X)$ are desirable for MTD sytems. This assumption is defined as the *MTD Entropy Hypotesyis* [206], also Hobson et al. [67] defines that an MTD technique is *unpredictable* iff $H(X) \gg 0$.

EVALUATION In this SLR, we are interested in the maximum value $H(X)$ for a given technique. For a practical application of this metric to the MTD techniques in the literature, we use two results from information theory. First, $H(X)$ is maximized if $P(X)$ follows a discrete uniform distribution, i.e., every value x is equiprobable. For a RV X with n possible values $\{x_1, ..., x_n\}$, this maximal value is $\log_2(n)$ bits. Second, MTD techniques will take inputs and deterministically produce an output, i.e., the MP value. It is well known that a theoretical limit exists for the output entropy of a process [206]: the entropy of the output RV can not be greater than the sum of the entropy of the input RVs. For a single RV input Y, $0 \leq H(X) \leq H(Y)$.

EXAMPLES We present three examples of the use of $H(X)$ as a metric for MTD systems in which the MP X is:

- The OS firmware, and there are 2 possible states: $H(X) \leq \log_2(2) = 1$ bit.
- The Encryption Algorithm used, and there are 16 possible states: $H(X) \leq \log_2(16) = 4$ bits.

Table 3.1: Definition of MTD Entropy-related Metrics.

Metric	Description	Possible values
$H(X)$	Shannon Entropy of a Moving Paramter X	$\mathbb{R}_{\geq 0}$
GEN	Cost of generating a valid state x_i	{Low, Med., High}
STO	Cost of storing a valid state x_i	{Low, Med., High}
MOV	Cost of a state change $x_i \rightarrow x_j$	{Low, Med., High}
ATT	Cost of an attack, assuming a state x_i	{Low, Med., High}

- The IPv6 Address of 128-bits, but the secret key to calculate it is a value of 32-bits (input RV Y): $H(X) \leq H(Y) \leq \log_2(2^{32}) = 32$ bits.

To the best of our knowledge, this is the first MTD survey that evaluates the Shannon entropy of the studied techniques.

3.4.2 *Qualitative Entropy-related Metrics*

MOTIVATION Many qualitative factors of the entropy are not captured by the Shannon entropy $H(X)$. For example, attacking 16 different OS firmwares is arguably harder than attacking 16 different IP addresses. In addition, switching an OS firmware may consume more resources for the system than switching an IP address, and this will impact a real-world implementation of the technique. Thus, 1 bit of entropy of the OS firmware as the MP is not qualitatively equivalent to 1 bit of entropy of the IP address as the MP.

DEFINITIONS In order to capture some of these qualitative differences, we define four novel metrics: GEN, STO, MOV, and ATT. They are based on the MP X modeled as a discrete RV and are related to a valid value x_i. Their definition is in Table 3.1.

TWO COST-RELATED PERSPECTIVES GEN, STO, and MOV measure cost from a system's perspective. ATT measures cost from an attacker's perspective. The *cost* is estimated in terms of the entity's use of limited resources (i.e., time, computing power, hardware). A priori, the lower the cost of GEN, STO, and MOV, the higher $H(X)$ that will be attainable with fixed resources. The ATT metric aims at capturing the entropy exploration cost from an attacker's perspective. Because time is a limited resource for an attacker facing an MTD system, ATT gives a measurement of the entropy (*apparent* attack surface)

exploration speed. From a system's perspective, high ATT values are desirable. It will translate in an attack surface that will be *difficult* (i.e., costly, slow) to explore.

TERNARY-VALUED AND VALUATION All these metrics are of ternary value: Low, Med., and High. This choice is justified because of the inherent uncertainty and difficulty of measuring them. Binary values were discarded as too coarse-grained. The estimation of the metrics for an MTD technique was done using the evidence provided in the same publication and my own expertise. The assigned costs-values are relative to the other values present within the examined set of techniques. When hypothesis-assumptions were required for one technique, they were propagated to all of the techniques.

3.5 METHODOLOGY

The methodology used in this SLR is based on the SLR guidelines by Kitchenam et al. [86]. There are three main phases in a systematic review process:

1. *Planning*: Involves specifying the RQs and developing a protocol to follow.
2. *Conducting*: Involves study search, study selection, data extraction, and data synthesis.
3. *Documenting*: Involves reporting the systematic review process (e.g., protocol, outcomes), i.e., this chapter.

The conducting phase, with detail on the search and selection processes, is illustrated in Fig.3.1. In the following, we explicit the SLR RQs, we call them them SLR-RQs to avoid collision with the RQs guiding this thesis memoir, and detail the protocol and execution of the conducting phase.

3.5.1 *SLR Research Questions*

This systematic review aims to provide an overview of existing MTD techniques for IoT and insights about their maturity in terms of security and usability. To achieve this goal, we defined four RQs that guide this SLR:

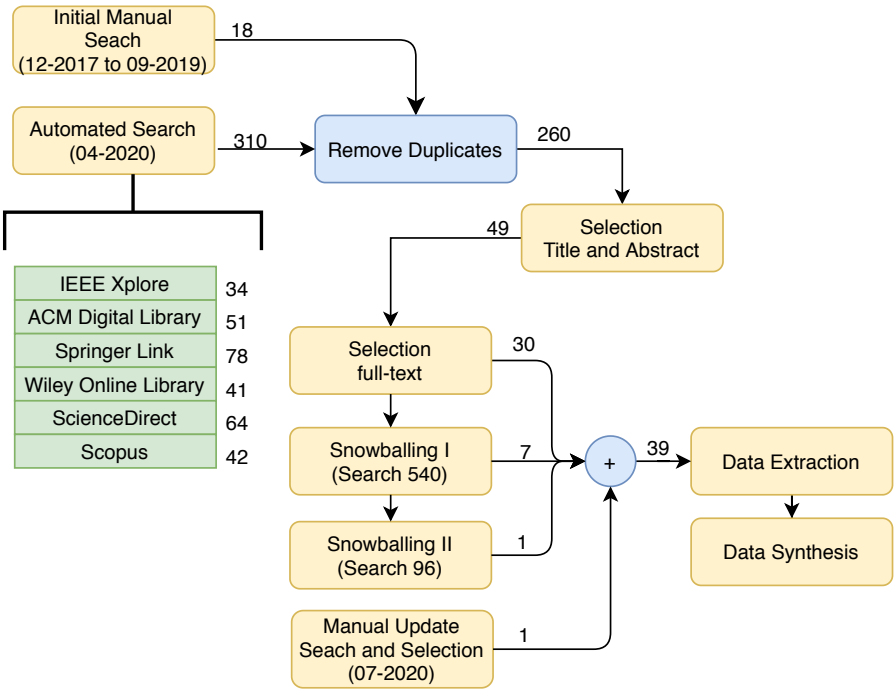


Figure 3.1: Conducting the SLR: detail on the search and selection processes. The number of articles after an activity is represented in labels at the exit edges.

SLR-RQ-1: How many proposals of MTD techniques for IoT exist?

SLR-RQ-2: What characteristics can be observed in the proposals?

SLR-RQ-3: How sound are the security foundations of the proposals?

SLR-RQ-4: To what extent the proposals can be used in a real deployment?

The RQs are ordered from the more generic to the more particular. The first two are broad RQs. We separated them because SLR-RQ-1 focuses on quantitative facts while SLR-RQ-2 on qualitative ones. The last two, SLR-RQ-3 and SLR-RQ-4, inquire into technical qualitative properties of the proposals. They are useful to give an assessment of the *maturity* of the MTD for IoT field.

3.5.2 *Search Process*

The search process involved three complementary methods: manual search, automated search, and snowballing.

3.5.2.1 *Initial Manual Search and Update*

The initial set of articles was obtained non methodically from Dec. 2017 to Sep. 2019. It included articles suggested by colleagues, manually found using references in other articles, and non-systematic searches in Google Scholar. In addition, we manually searched among all editions of the *ACM Workshop on Moving Target Defense*. Initially, the set was not large enough to justify a SLR. In Sep. 2019 the set consisted of 18 articles, and we estimated a SLR. In July 2020, we did a manual update to include recently published work.

REPRODUCIBILITY Manually including article [2], the automated search, and the snowballing method will yield the same results without the need of this manual set.

3.5.2.2 *Automated Search*

For the automated search, we defined a *search string* and used six well-known digital databases.

DATABASES The databases used were the following:

- IEEE Xplore
- ACM Digital Library
- Springer Link
- Wiley Online Library
- ScienceDirect
- Scopus (meta-searcher)

SEARCH STRING The search string was the following:

```
("mtd" OR "moving target defense") AND ("iot" OR "internet
of things")
```

Note that the term `mtd` yield many false positives (e.g., machine-type devices, minimum-traces-to-disclosure). We suggest researchers not using acronyms in the search string. The title and abstract of the articles will include the unabridged term of the acronym. This will reduce false positives and make the search and selection process less time-consuming. The automated search was conducted during the month of April 2020.

DUPLICATES REMOVAL We used the JabRef reference manager to combine and remove duplicates from the raw search results. We prioritized exporting-importing in BibTeX format.

3.5.2.3 *Snowballing*

DEFINITIONS Snowballing is a search technique that identifies potential additional articles to include in a systematic literature review [195]. Snowballing can be applied iteratively over the selected articles. Backward snowballing captures (past) articles that are on the reference list of an included study. Forward snowballing captures (future) articles that refer to an included study.

APPLICATION We applied both backward and forward snowballing using the Scopus meta-searcher. The first iteration was applied to the studies selected by the initial manual and automated search process. We performed two iterations. Snowballing was performed during April-May 2020.

3.5.3 *Selection Process*

The selection process is applied to search results and determines which studies are included in our review. We explicit the inclusion and exclusion criteria used to filter the results.

INCLUSION The inclusion criteria are:

- I1: Studies that propose MTD-based techniques that can be used in constrained IoT devices.
  - The level of detail of the technique is not excluding.
  - Not mentioning MTD nor IoT is not excluding.
- I2: Studies in the English language.
- I3: Peer-reviewed studies or books.

EXCLUSION The exclusion criteria are:

- E1: Studies that despite mentioning MTD and IoT:
  - Propose techniques for non-constrained devices, e.g., smart vehicles (broad use of the term IoT).

  - Propose techniques not applicable to IoT devices directly, but to other non-constrained components of the system, i.e., the technique was transparent to the IoT nodes. For example, firewalling, backbone/cloud, or non-constrained Software-Defined Networking (SDN) solutions.

- E2: Studies published before 2009.
- E3: Studies for which we could not access the full text.

APPLICATION OF CRITERIA The criteria I2, I3, and E2 were applied automatically on the digital databases searches. Then, we applied the semantic-dependent filtering criteria I1 and E1 in a two-step process. Firstly, only taking into account title, keywords, and abstract of the studies. Secondly, taking in account the full-text. In case of doubt in the first step, the study was included for the full-text selection step. Most studies were discarded during the first step.

3.5.4 *Data Extraction Process*

PROCESS Each of the 39 selected articles was read thoroughly by myself. The data extraction template evolved between Jun 2019 and April 2020. The 18 articles from the initial manual search were read and data extracted (refined) at least twice having a time span of at least three months between reads.

TEMPLATE The final template used to extract relevant data from each study is shown in Fig. 3.2.

3.5.5 *Data Synthesis Process*

The goal of the data synthesis process is to provide meaningful information about the current state of the art of MTD techniques for the constrained IoT. Particularly, the outputs of the data synthesis methods summarize the data results and shall provide convincing answers to the SLR-RQs of Sec. 3.5.1.

SYNTHESIS METHODS USED There are a variety of data synthesis methods [86]. In this work, the syntheses outputs are presented in the form of graphical plots, tables, and narrative synthesis, i.e., text. We synthesized both quantitative and qualitative aspects of the primary studies. An intermediate analytical step was necessary to synthesize

- Standard bibliography data:
  - Title, author, year, type of publication, venue.
- MTD technique name or brief description.
- Moving Parameter (MP).
- MTD technique taxonomy:
  - Data, Software, Runtime Environment, Platform, Network.
- Metrics:
  - Evaluate MP Shannon entropy metric (See Sec. 3.4.1).
  - Evaluate MP qualitative metrics (Defined in Sec. 3.4.2).
- Cryptography:
  - Is cryptography used?
  - Which cryptographic primitive is used?
  - What are the cryptographic inputs? (e.g., a key)
- Implementation:
  - Is the proposal implemented (even partially)?
- Evaluation:
  - Is the proposal evaluated?
  - How? Numerically, Simulation, Hardware prototype.
- Synthesis of the proposal with technical details (1-6 paragraphs).

Figure 3.2: Data Extraction Template.

some aspects (mostly qualitative) of the primary studies. In this process, we used existing MTD theory (e.g., MP, accepted taxonomies) and the *metrics* we developed in Sec. 3.4. The metrics allow a common frame of reference to synthesize and compare qualitative aspects of different studies.

3.6 SLR RESULTS

In this section, we present the results from the systematic review process. The SLR-RQs of Sec. 3.5.1 structure this section. Each subsection analyzes the results in the context of the RQs, and factually provides answers. Interpretive discussion is to be found in Sec. 3.8.

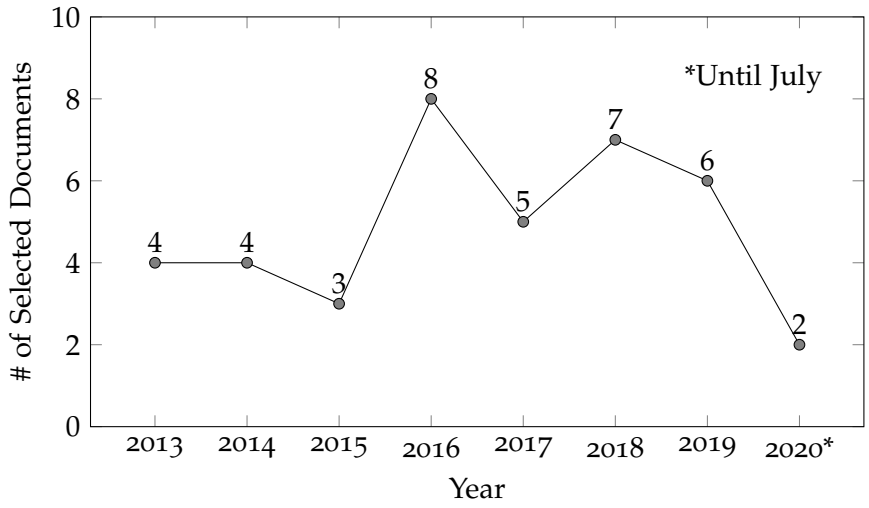


Figure 3.3: Number of selected documents per year (Total = 39).

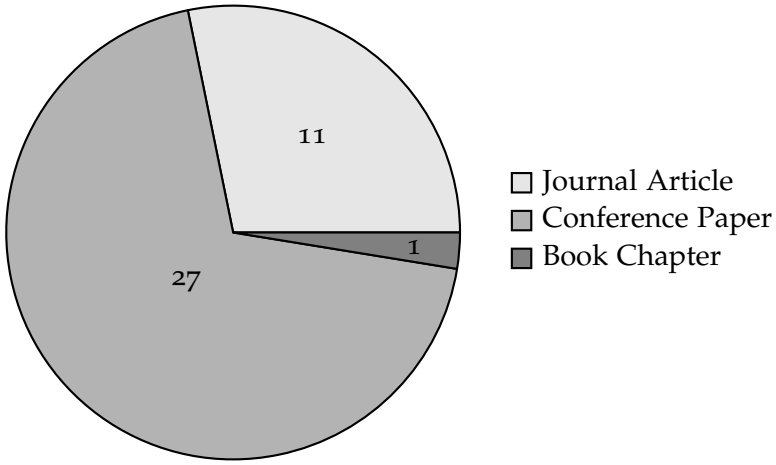


Figure 3.4: Number of documents by publication type.

3.6.1 *SLR-RQ-1: How many proposals of MTD techniques for IoT exist? (Status of Field of Study: Quantitative)*

The systematic review process, shown in Fig. 3.1, identified 39 documents containing 32 distinct proposals. There is not a one-to-one correspondence between documents and proposals. One proposal can spread among multiple documents, and one document can contain multiple proposals.

3.6.1.1 *Documents*

PER YEAR In Fig. 3.3 we plot the published documents per year. The first article is from 2013, two years after the first general-purpose MTD

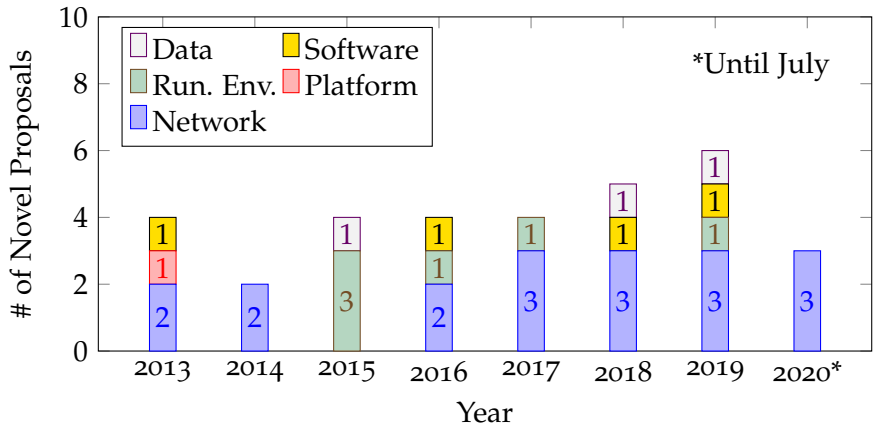


Figure 3.5: Number of novel proposals per year (Total= 32).

techniques that date from 2011. Excluding the year 2020, the (average $\pm$ standard deviation) number of publications per year is (5.3 ± 1.8). Since 2016 is (6.5 ± 1.3). Aside from the 2015-2016 increment, there is no clear upward trend, and the number of published documents per year is stable with post-2016 values.

VENUE, COUNTRIES, AND AFFILIATIONS The document publication type distribution is shown in Fig. 3.4, conference papers are predominant with a 69%. The top-3 countries are: USA (49%, 19 doc.), Finland (15%, 6 doc.), and Italy (15%, 6 doc.). The top affiliations are: University of Turku (13%, 5 doc.), Virginia Polytechnic Institute and State University (13%, 5 doc.), University of Naples Federico II (10%, 4 doc.), and George Mason University (10%, 4 doc.).

3.6.1.2 *Proposals*

PER YEAR Thirty-two novel proposal have been identified. In Fig. 3.5 we plot the novel proposals per year. A proposal is counted only once, taking the date of the first document that included it. Excluding the year 2020, the (average $\pm$ standard deviation) number of novel proposals per year is (4.1 ± 1.2). The minimum value was in 2014 (2 proposals) and the maximum in 2019 (6 proposals). The number of proposals per year is stable since 2013, with a slight upper trend of $\Delta = 1$ in the last two periods since the 2017-2018.

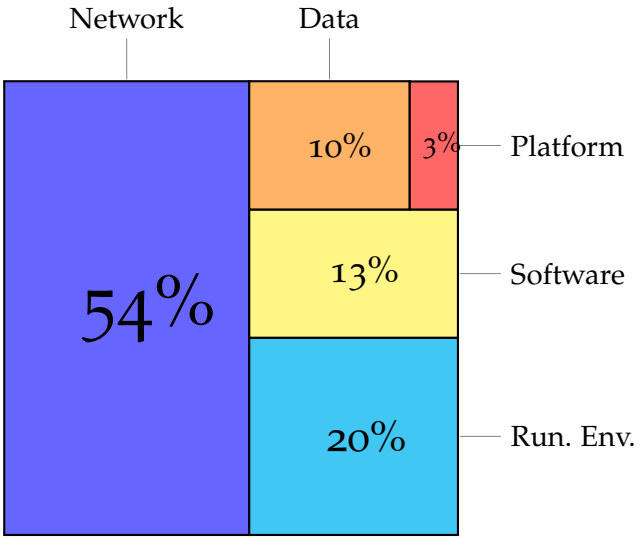


Figure 3.6: Taxonomy distribution of MTD techniques for IoT.

3.6.2 *SLR-RQ-2: What characteristics can be observed in the proposals? (Status of Field of Study: Qualitative)*

This question aims at highlighting qualitative aspects of the field of study of MTD for IoT techniques. We categorize, measure, and analyze technical properties of the techniques.

This section is divided in two parts. In the first, we use general MTD theory concepts presented in Ch. 2.2.3, in particular, the widespread taxonomy for MTD techniques. In the second, we use the entropy-related metrics we defined in Sec. 3.4 to describe and analyze the state of the art.

3.6.2.1 *MTD Taxonomy: Distribution and Trends*

We present the distribution of the techniques by MTD taxonomy in Fig. 3.6. *Network* techniques are predominant, with 54%. In the second position are dynamic *Runtime Environment* techniques with 20%. *Software* and *Data* techniques have a similar share with 13% and 10%, respectively. Notably, there is only one dynamic *Platform* technique (3%). Fig. 3.5 shows novel proposals per year and taxonomy. Excepting the year 2015, *Network* proposals have a constant rate of production and account for $\geq$ 50% even on a year-to-year basis. A relevant derived RQ is:

> *How do taxonomy distribution and trends compare between MTD for IoT and general MTD techniques?*

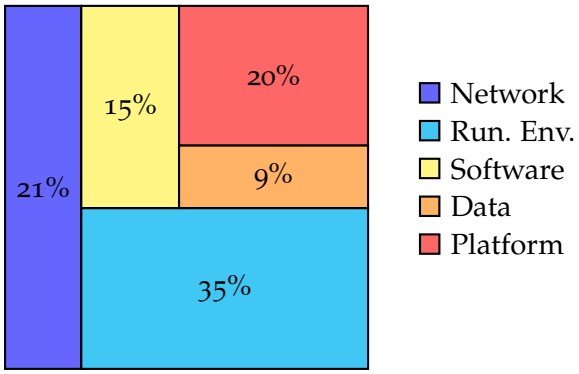


Figure 3.7: Taxonomy distribution of general MTD techniques.

To answer this question, we leverage on the general MTD survey from Lincoln MIT laboratory [190]. It dates from 2018 and comprises 89 distinct general purpose MTD techniques. The taxonomy distribution of the techniques is shown in Fig. 3.7. *Network* techniques are not predominant as in IoT; they account for 21%. *Runtime Environment* techniques take the most significant share with 35%. Notably, *Platform* techniques, almost non-existent in IoT systems, share the virtual second place with 20%. *Software* and *Data* techniques have similar values as in IoT with 15% and 9%, respectively. However, there is an increasing interest in general-purpose MTD *Network* techniques since 2015. We base this assertion on the rate of publications of the MTD Network techniques included in the 2020 survey of Sengupta et al. [160]. Also, recent MTD surveys [33, 177] focus on Network MTD solutions, which indicates a growing interest in the research community.

3.6.2.2 *MP Metrics: Shannon's entropy and other entropy-related metrics*

This subsection presents the results of applying several metrics related to the MP X. The metrics are defined in Sec. 3.4. To the best of our knowledge, this is the first MTD survey to apply the Shannon entropy metric to the studied techniques. The precise values of the metrics per technique can be found at the end of this section in Table 3.2. In Fig. 3.8, we present the histogram of the Shannon entropy $H(X)$ in bits of the techniques. Each bin aggregates values inferior to the label of the next bin, for example, in the bin '32' $\rightarrow 32 \leq H(X) < 64$. Neither *Platform* nor *Software* categories have techniques with 64 bits or more of Shannon's entropy. On the other hand, the rest of the categories have at least two techniques, each with 128 bits of entropy or more.

In the following, we synthesize results derived from the use of the novel qualitative metrics defined in this work. The goal is to highlight possible relationships between Shannon's entropy and other qualitative metrics of the MP.

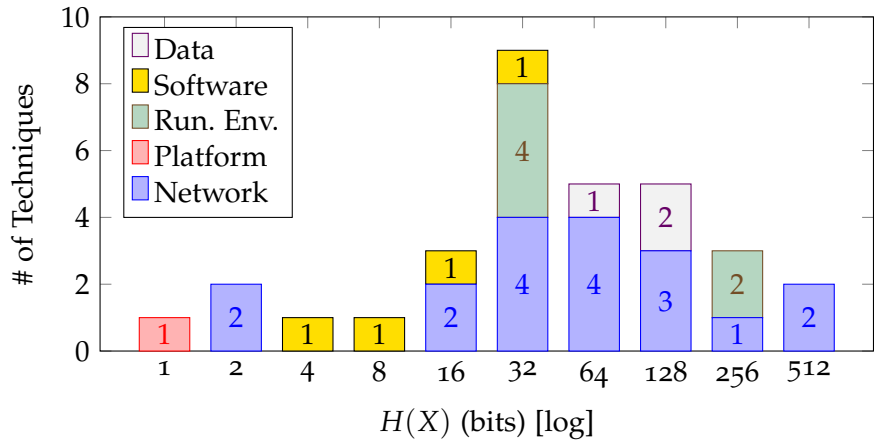


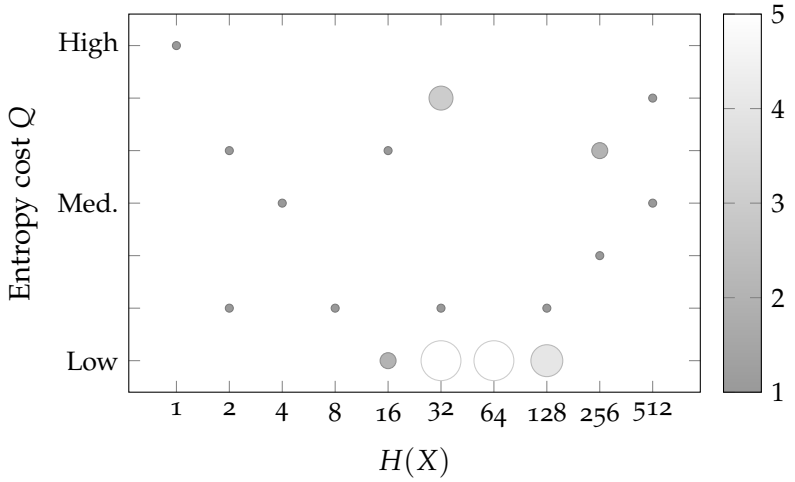
Figure 3.8: Histogram of Shannon's entropy of the techniques.

DEFINITION: ENTROPY COST Q To present the results, we aggregate the three system-centric metrics GEN , STO, and MOV, into the metric Q. We call Q the *entropy cost* because it captures how expensive, in terms of resources, is for the system the process of *generating*, *storing*, and *moving* the MP value. Each individual system-centric metric will be mapped to a value in $\{0,1,2\}$ from the original domain of $\{Low, Med., High\}$. From a system defense perspective, 0 is the most desirable value (lower cost) and 2 the least (higher cost). We define Q as the arithmetic sum of the system-centric metrics. Its value is in the range $\{0,1,...,6\}$. Again, the lower this value, the better from a system's perspective. Ideally, we want high values of entropy $H(X)$ at a low-cost Q. The results are shown in Fig. 3.9a.

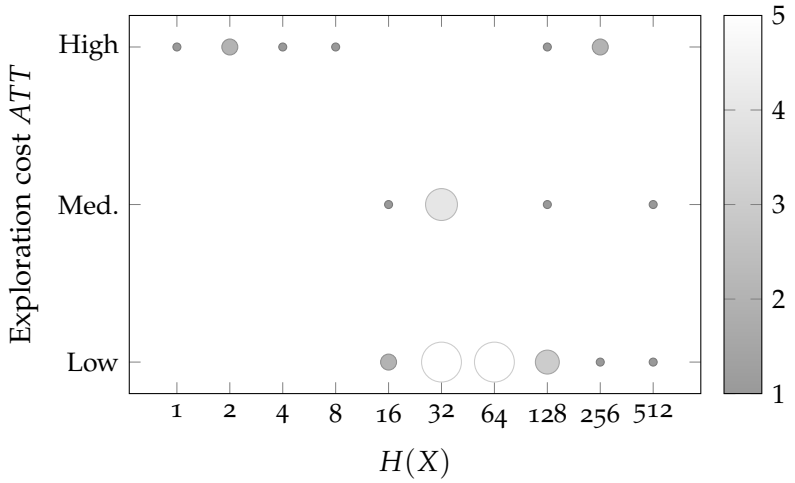
Similarly, the attacker-centric metric ATT is presented in Fig. 3.9b. From a system's perspective, the higher the ATT value, the better. A higher cost translates into a given attacker exploring-attacking fewer values of the MP using the same resources (e.g., time).

ANALYSIS For the most part, some expected correlations can be observed in the empirical data in Fig. 3.9. Those are:

- For high $H(X)$ techniques, the entropy cost Q should be low. This justifies the empirical feasibility of a technique with high entropy (i.e., the system is able to cope with the cost of generating, storing, and moving new values of this high-entropy MP X).
- For low $H(X)$ techniques, the exploration cost ATT should be high. This justifies the usefulness of a technique with low entropy (i.e., *low quantity but of high quality*).



(a) $H(X)$ and entropy cost metric (system-centric).



(b) $H(X)$ and exploration cost metric (attacker-centric).

Figure 3.9: Relationship between Shannon's entropy and other metrics (# of techniques per combination).

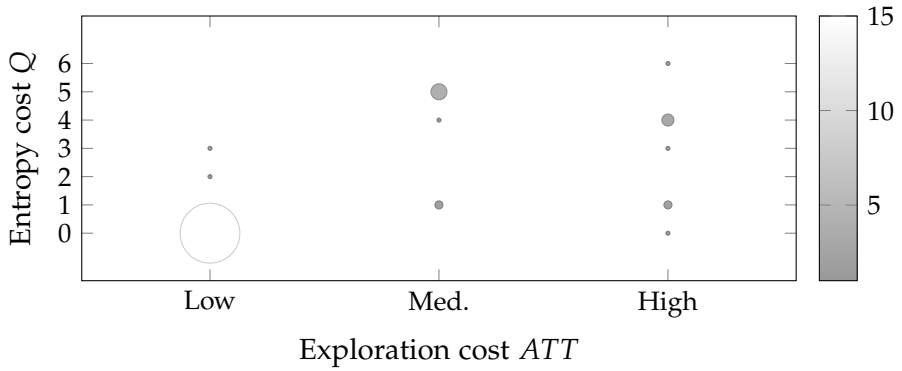


Figure 3.10: Number of techniques grouped by ATT and Q.

The first expectation is observed in Fig. 3.9a in techniques with $H(X) \leq 128$. Aside from some outliers (in $H(X) = 16$ and 32), we observe that as $H(X)$ grows Q decreases. However, techniques with $H(X) \geq 256$ reverse the trend with Med. to High values of Q. This is possible, but not desirable. It means that those techniques will be costly to implement in a real-world system. The second expectation is observed in Fig. 3.9b. In general, as an inherent trade-off, it is also expected that higher $H(X)$ will imply lower ATT. However, we find many (five) exceptions, especially in techniques with $H(X) \geq 128$ that have Med. to High values of ATT. This is desirable from a system point of view.

Finally, in Fig. 3.10 we plot ATT vs. Q metrics. Low ATT imply low entropy cost Q, 47% of techniques are in that case. Aside from that, there is no apparent correlation between them.

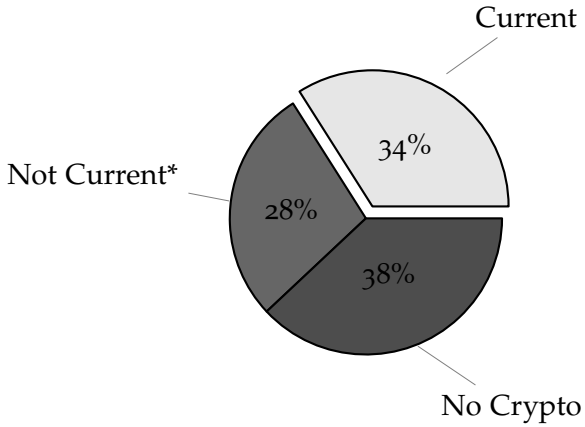
3.6.3 *SLR-RQ-3: How sound are the security foundations of the proposals?*

We extracted three cryptography-related information items[2] that we can relate to the security foundations of each technique. In Table 3.2 we present the raw extracted data in a per-technique basis. In this section, we synthesize those results using the following arguments:

- The fact that all techniques rely on a randomization process[3] (*Thesis*).
- A technique is *secure* only if it uses cryptographically strong randomness (*Hypothesis*).

2 *Is cryptography used? Which cryptographic primitive is used? What are the cryptographic inputs?*

3 Even the ones that are diversification-based use randomization either to create the variants or to select one of them.



*Deprecated or not-tested cryptography.

Figure 3.11: Cryptographic categories of the techniques.

In other words, we associate the *security foundation* of a proposal with the cryptographic primitive it uses. In order to do a proper assessment of each technique, in many cases, we required more detailed information than the three cryptography-related items. This complementary information was obtained from the "Synthesis of the proposal with technical detail" field in the data extracted.

Finally, we categorized each technique into one of the following *cryptographic* categories:

- *No Cryptography*: We consider these techniques to lack proper security foundations. Of the 32 distinct techniques, four explicitly do not use cryptography. They are instead based on algorithms from game theory, deterministic or stochastic optimization problems. Neither has any input entropy to the problem other than the system variables. Other studied techniques assume, sometimes even implicitly, a random process but do not give any detail about it. We grouped all these techniques into this category.

- *Not Current*: We consider these techniques to lack proper security foundations. Techniques in this category, either use cryptography that is known to be vulnerable (e.g., MD5), or proposed their custom-made cryptographic primitives or protocols but without security proofs.

- *Current*: We consider these techniques to have proper security foundations. These techniques use legacy or state-of-the-art cryptography with security proofs and not-known attacks (e.g., SHA256, HMAC, ChaCha20, Keccak).

The results are shown in Fig. 3.11. Only 34% of the techniques (11 out of 32) use *current* cryptographic primitives. The remaining majority (66%), uses not current or not cryptography at all.

3.6.4 *SLR-RQ-4: To what extent the proposals can be used in a real deployment?*

To answer this question, we use empirical evidence provided by the proposals in their corresponding publications. All of them provide, with varying levels of detail, a *design* specification. To provide an answer to this RQ, we put the focus on two other aspects of the proposed techniques: the *implementation* and *evaluation* of them. Some clarification about those aspects:

- An *implementation* of a proposal provides strong evidence on the feasibility of using it in a real IoT deployment. Some proposals implemented the technique in software and evaluated it in a simulated system (without using actual IoT hardware), while others used IoT hardware. Despite those differences, we consider any of them as proof of implementation. A technique is categorized as either implemented or not.

- An *evaluation* of a proposal provides evidence about the expected effectiveness or usability of it when deployed. *Evaluation* was divided into three non-exclusive sub-categories. *Theoretical*, if the evaluation was done analytically or numerically (e.g., for an abstracted mathematical aspect of the technique). *Simulation*, if the IoT system was simulated even partially (e.g., ContikiOS Cooja, NS-2). *Hardware* (HW), if the technique was evaluated using real IoT hardware.

An *evaluation* does not imply an *implementation*. For example, some authors evaluated a partial or abstracted component of the proposal (mostly theoretically or simulated).

RAW RESULTS The raw results are the following. For *implementation*, 50% percent of the techniques were implemented, and the rest were not. In Fig. 3.12, we show a Venn diagram of the *evaluation* status of the techniques. Only 19% were not evaluated at all. Of the rest, 44% were evaluated in simulation, 25% in hardware, and 22% theoretically.

To answer SLR-RQ-4, we define five exclusive categories that correspond to the evidence a technique provides to be used in a real IoT deployment. They are defined as follows, depending on the implementation and evaluation status of a technique:

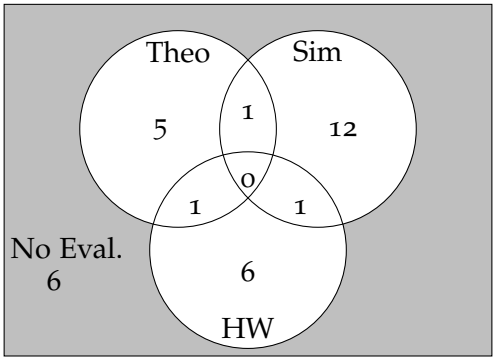


Figure 3.12: *Evaluation* status of the techniques (Total = 32).

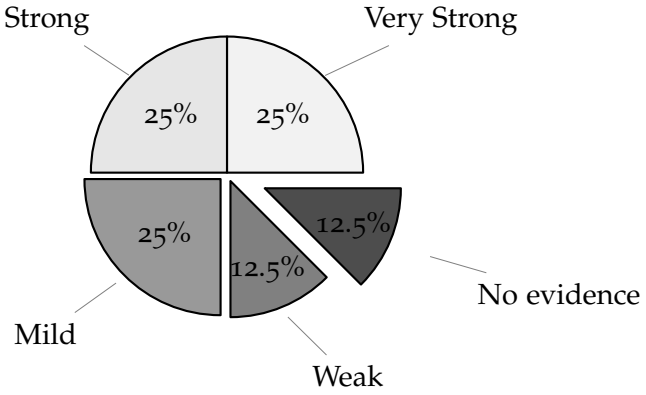


Figure 3.13: Distribution of categories of evidence about real IoT deployment of techniques.

- *Very Strong*. Implementation and hardware evaluation.
- *Strong*. Implementation without hardware evaluation.
- *Mild*. No implementation, but HW or Simulation evaluation.
- *Weak*. No implementation, but theoretical evaluation.
- *No evidence*. No implementation nor evaluation.

The results are shown in Fig. 3.13. Half of the 32 techniques provide Strong or Very Strong evidence about being used in a real IoT deployment. About 38% present Mild to Weak evidence. Finally, a minority of about 13% presents No evidence.

3.6.5 *SLR Results detailed per technique*

We summarize the raw data extracted from the publications on a per technique basis. The results can be seen in Table 3.2.

Table 3.2: MTD Techniques for the Constrained IoT

Reference(s)	Technique	Moving Parameter (X)	Taxonomy*	Metrics* $H(X)$	GEN	STO	MOV	ATT	Cryptography Primitive	Input	Impl.	Evaluation Theo	Sim	HW
[17, 29, 30]	Crypto-protocol reconf.	Network Crypto-Protocols	S	4	+	/	-	+	-				✓	
[17, 29, 30]	Firmware reconf.	OS Firmware (In local storage)	P	1	+	+	+	+	-		✓			
[137, 165, 199, 200]	μMT6D	IPv6 IID Addresses (64-bit)	N	64	-	-	-	-	SHA256	Key	✓		✓	
[69, 70, 91, 109, 140]	OS If. Div. Addr. Layout	Memory layout of linked binaries	R	32†	/	+	+	/	MD5	Salt	✓			✓
[69, 70, 91, 109, 140]	OS If. Div. Names	Symbol names of OS libraries	R	32†	-	/	-	/	SHA-2	Salt	✓			
[108]	Code Diversification	Code to store and execute	S	20**	/	/	+	/	-					
[2]	μScramble	Binary Linked Code	S	32†	/	+	+	/	LLVM PRNG	Seed	✓			✓
[2]	μSSP	Stack Canary (32-bit)	R	32	-	-	-	-	μRNG (Keccak[183])	Seed(SRAM)	✓			✓
[10]	IPv6-Multicast (SARCAST)	IPv6 Multicast Group-ID (80-bit)	N	32	-	-	-	-	SHA-1	Salt(32b)	✓		✓	✓
[56]	An ASLR Proposal	RAM addresss layout (ASLR)	R	52‡	-	-	-	-	-			✓		
[57]	SDN-IoT Topology Reconfiguration	Network Topology (Routing)	N	525§	+	/	+	/	-				✓	
[64]	Honeypots with Cellphones	Network Nodes Roles	N	47	+	/	+	/	-				✓	
[126]	AShA	MAC/IPv6 Address (16-bit)	N	16	-	-	-	-	HMAC	Key			✓	
[189]	ZD Game Theory Approach	Resource Node Locations	N	671¶	-	/	+	-	-			✓		
[80]	6HOP	P2P IPv6 IID (64b) + Port (16b)	N	160	-	-	-	-	An Unkeyed-Hash	Secret(512b)				
[138, 139]	DLSeF	Encryption Key for App. Data	D	128	-	-	-	-	*see note*‖	Key			✓	
[45]	Ephemeral	IPv6 IID Addresses (64-bit)	N	64	-	-	-	-	A Block Cipher	Key	✓		✓	
[197]	Stochastic Cost Minimization	Nodes that mutate Network Address	N	44††	-	-	-	-	-			✓		
[47]	Application Data re-Keying	Encryption Key for App. Data	D	64	-	-	-	-	LED Block Cipher	Key(64b)	✓		✓	
[129]	PHY-layer Divesification	PHY-layer Technology	N	2	+	+	-	+	-				✓	
[120]	APP-layer Protocol Diversification	Communication Protocol	S	8‡‡	-	/	-	+	A PRNG	Seed	✓			✓
[187, 188]	Re-keying with Side-Channel Attacks	Encryption Key	D	128	-	-	-	+	(Dziembowski [46])	Key				
[41]	Malware tolerant Mesh-Networks	Device Groups and Group-Keys	N	256¶¶	/	-	/	-	IRS [74] and *note*¶¶	Key			✓	
[4]	Identity Virtualization	Node IDs	N	64***	-	-	-	-	An Unkeyed-Hash	A Secret	✓		✓	
[110]	MAC Address Randomization	MAC Address (48-bit)	N	48	-	-	-	-	-			✓		
[63]	MAVR	Memory layout of linked binaries	R	256†††	/	+	/	+	A PRP	?	✓			✓
[6]	SDR defined PHY-layer	PHY-layer Modulation	N	3	-	/	-	+	-		✓	✓		
[93]	uOTA	P2P IPv6 IID (64b×2)	N	128	-	-	-	-	-		✓		✓	
[130]	AVRAND	Memory layout of linked binaries	R	256§§	/	+	/	+	A PRNG	Seed	✓			✓
[176]	SAD-SJ	PHY-layer TDMA Slot allocation	N	128	-	-	/	/	A PRP (Sym.Cipher-base)	Key		✓	✓	
[123]	UDP Port-Hopping	UDP Port Number	N	16	-	-	-	-	ChaCha20	Key+Nonce	✓	✓		✓
[123]	REST protoc. URIs Randomization	CoAP .well-known/core URI	N	120	-	-	-	-	ChaCha20	Key+Nonce				

* Taxonomy: Data (D), Software (S), Runtime Environment (R), Platform (P), Network (N).
Metrics Cost: Low (-), Med. (/), High (+).

† Assumed a 32-bit Salt/Seed.

** Assumed 1024 (2^{10}) Code Partitions, and 1024 Versions for each.

‡ Assumed 32KB RAM (2^{13} 32-bit addresses) and 4 regions to randomize.

§ Assumed 100 Nodes and number of possible Topologies ≈!100.

¶ Assumed 100 Nodes and 101 Resources to locate (100^{101} resource locations)

‖ Two custom-made crypto-protocols. Flaws: They re-use a 128-bit One-Time-Pad.

†† Assumed that 10 Nodes out of 100 can be chosen to mutate address.

‡‡ Assumed 200 protocols being used on the network.

¶¶ 2 privileged groups each with a 128b key. Authors suggest Burmester-Desmedt Group Key Agreement Protocol.

*** Assumed 64-bit length Node IDs.

††† Assumed a 256-bit Seed. Authors claim 6567 bits. Valid for 800 symbols $log(!800)$, but limited by input entropy.

§§ Assumed 256 bits of entropy for the Seed. Authors harvest entropy using an AVR timer and an oscillator.

3.7 LIMITATIONS OF THIS SLR STUDY

In this section, we assess the limitations of this review. One of the main limitations of this SLR study is that the 39 papers and the 32 distinct techniques are statistically limited to make conclusive claims. Other surveys in similar but established fields like "MTD in cloud computing" [177] worked with 95 papers. However, we believe that the amount of material included in the current SLR is sufficient to answer that MTD for the IoT is possible (RQ-1), and to identify the most prominent trends in the field (RQ-2). We will detail these answers in the next section.

As with any survey work, we might have left relevant papers out. Particularly difficult were *edge* cases where the publications did not explicitly mention MTD but used randomization or diversification of system components. If our inclusion criteria become too permissive, this survey might include entire research clusters that never mention MTD. However, we believe that we minimized the risk of letting out relevant work with the SLR methodology. Particularly useful were the snowballing techniques that iteratively capture related work. Furthermore, as the search method and inclusion-exclusion criteria are documented, future researchers could improve upon the current survey and address its shortcomings.

The evaluation of the metrics could also be contested. A subjective component is present in their evaluations. We had to make assumptions to evaluate the Shannon entropy, they are detailed in the footnotes of Table 3.2. The other four qualitative entropy metrics were evaluated by the experience/assessment of the authors. Even if we tried to be consistent among all techniques, there is still a subjective component on the final value. We tried to minimize the subjective bias for all of them. For Shannon's entropy, we made assumptions that we applied to every technique that needed them. For the qualitative metrics, we used coarse-grained (ternary) values. The metric values should be interpreted with a corresponding inherent uncertainty. We believe that, despite being approximate values, it is useful to have measurable quantities to compare different techniques.

3.8 SLR SUMMARY AND DISCUSSION

In this section, we summarize the answers to the SLR-RQs with an added component of interpretation. Also, we are in the condition to construct the answers to two of the general RQs of this thesis, RQ-1 and RQ-2, which we recapitulate here:

RQ-1: Is MTD for the constrained IoT possible?
RQ-2: What is the status of MTD techniques for IoT?

We indicate how each SLR-RQs contributes to their answer, but we summarize the RQs's answers in the conclusive section of this chapter.

SLR-RQ-1 *How many proposals of MTD techniques for IoT exist?* Thirty-two distinct techniques. This figure was not evident prior to this survey. The previously identified corpus in the MTD literature was of about a dozen techniques. This answer contributes to provide evidence to answer RQ-1 in the affirmative, and a partial answer to RQ-2 focusing on this quantitative fact.

SLR-RQ-2 *What characteristics can be observed in the proposals?* In contrast to SLR-RQ-1, SLR-RQ-2 focus on qualitative aspects of the field of study, particularly, MTD taxonomies distribution and metrics. This answer contributes to provide a partial answer to RQ-2.

First, we categorized the IoT techniques according to the MTD taxonomies. It was relevant to find that Network-category techniques account for more than half of them, while in the non-IoT MTD, they account for 20%. This can be explained by the importance of the network component in IoT systems, which translates into an effort to protect it.

Secondly, we applied the entropy-related metrics. We used the Shannon entropy of the MP $H(X)$, in conjunction with the entropy cost Q and the entropy exploration cost ATT. Approximately 69% of the techniques are comprised between $16 \leq H(X) \leq 128$. They are mostly of the Network, Runtime Environment, or Data categories. For the most part, both the entropy cost Q and the exploration cost ATT are Low. These techniques fall into a reasonable compromise among all the metrics. For system designers, at equal Q and ATT, we recommend prioritizing the higher Shannon entropy techniques, e.g., the ones with 128 bits.

SLR-RQ-3 *How sound are the security foundations of the proposals?* We justified that if deprecated or no cryptography is used, the security foundation of a proposal is not convincing. The results show that only 34% of the techniques use current cryptography. It is a low value, considering that improving security is the main objective of an MTD technique. This answer contributes to provide a partial answer to RQ-2.

Measuring the *security* of a system is a challenging task and depends on many factors. In general, a particular technique should define precise security goals, an attacker and system model for a theoretical

evaluation, or implement a real attacker, system, and define a use case for a more empirical evaluation. Comparing different techniques is not straightforward. We simplified this comparison problem by taking into account the cryptographic primitives of each technique. We assumed that if not current cryptography is used, an attacker can eventually replicate the system's movement and neutralize the effect of the MTD.

SLR-RQ-4 *To what extent the proposals can be used in a real deployment?* We looked for evidence in the publications themselves about the proposed technique's real-world usability. We used the proofs of implementation and degrees of evaluation as indicators. The results are encouraging, 50% of techniques provided strong or very strong evidence about their usability. Only %13 did not provide any evidence. This answer contributes to provide evidence to answer RQ-1 in the affirmative, and a partial answer to RQ-2.

3.9 CONCLUSION

In this chapter, we conducted a SLR about MTD techniques for the constrained IoT. An in-depth study of this field was missing in the literature, and answers to the RQ-1 and RQ-2 were not evident. In addition, we developed entropy-based metrics of empirical applicability that can be useful in future MTD-related applications beyond the scope of this survey and thesis.

RQ-1 Firstly, we can answer RQ-1 in the affirmative, by the evidence gathered in the answers to the SLR-RQs-1,4. MTD for the constrained IoT is a feasible cyberdefense technique. A large enough number of MTD techniques exist, with strong evidence about their implementation and evaluation.

RQ-2 Secondly, the answer to RQ-2 is not binary, and it takes from the answers to the SLR-RQs-1,2,3,4, i.e., the SLR results. In synthesis, 32 distinct MTD for IoT techniques exist and Network-layer-oriented techniques account for 54%. Half of all the techniques present strong evidence about their real-world deployment, and the majority of the techniques (64%) have weak security foundations.

PERSPECTIVES: A GAP IN SECURITY This SLR shows that the state of the art of MTD techniques for the constrained IoT is still immature: even if most of the techniques have convincing empirical evidence about their real-world deployability (*usability*), the cryptographic foun-

dations (*security*) of most are weak. MTD for IoT is a reality, but future work should prioritize providing convincing security foundations and keep providing real-world deployment evidence.

One of this thesis' main goals is to improve the resilience of constrained IoT systems. More precisely, we want to achieve this through the use of MTD techniques. The affirmative answer to RQ-1 was a necessary condition to keep working on this *thesis*. Fortunately, other researchers explored this goal, and our answer to RQ-2 provided us with a state of the art of MTD techniques for the constrained IoT. In the current chapter, we proved that *usable* proposals are a fact, but most techniques have shortcomings in their cryptographic foundations. This situation motivate us to formulate the following RQ:

> **RQ-3**: How to create *usable* and *secure* MTD techniques for the constrained IoT ?

This RQ will guide, for the most part, the rest of this thesis. In the next chapter, we will analyze it and inquire about the implications of a thorough answer. We will break RQ-3 down into several RQs related to general *design* aspects of MTD techniques and security best practices, and provide some of the answers.

Research Questions

RQs answered:

- **RQ-1**: Is MTD for the constrained IoT possible?
- **RQ-2**: What is the status of MTD for IoT techniques?
  - SLR-RQ-1 How many proposals of MTD techniques for IoT exist?
  - SLR-RQ-2 What characteristics can be observed in the proposals?
  - SLR-RQ-3 How sound are the security foundations of the proposals?
  - SLR-RQ-4 To what extent the proposals can be used in a real deployment?

RQs raised:

- **RQ-3**: How to create *usable* and *secure* MTD techniques for the constrained IoT ?

Part II

ABSTRACTION

The second part of this memoir is about the *design* of usable and secure MTD techniques for the constrained IoT.

We explore *what* are the most suitable IoT systems' components to become MPs, and *how* to move them in a distributed way. We propose IANVS, a generic and modular framework that can be used to instantiate particular MTD techniques.

4

DESIGNING MTD TECHNIQUES FOR A RESILIENT IOT

> Il semble que la perfection soit atteinte non quand il n'y a plus rien à ajouter, mais quand il n'y a plus rien à retrancher.
>
> Antoine de Saint-Exupéry[a]
> (*Terre des Hommes, 1939*)

a I do not pretend to attain perfection, but to adhere to the principle of simplicity.

4.1 INTRODUCTION

In the previous chapter, we acknowledged the feasibility of MTD techniques for the constrained IoT (RQ-1), and depicted the state of the art of the field (RQ-2). However, a lack of sound cryptographic founda-

tions in most of the proposals lead us to pose the question of how to create *usable* and *secure* MTD techniques for the constrained IoT (RQ-3).

In this chapter, we focus on fundamental MTD *design* questions (See Ch. 2.2.3) that need to be explored in order to answer RQ-3. We formalize them as follows:

RQ-3.D1: What are suitable MPs in IoT systems?
RQ-3.D2: How to move distributed MPs in IoT systems?
RQ-3.D3: When to move the MP?

Each RQ will be treated in a separate section of this chapter. First, we identified IoT systems' components suitable for MTDs (RQ-3.D1) corresponding to the MTD Network taxonomy.

Secondly, the answer to "how to move?" (RQ-3.D2) lead us to the definition of a generic and modular MTD framework we called IANVS[1]. Its building blocks correspond to fundamental cryptographic or network security fields. Thus, an instantiation of our framework can leverage on well-established security solutions. This framework is an essential contribution of this thesis and will be used in the third part of this manuscript to instantiate concrete MTD techniques for IoT systems.

Finally, we include a brief discussion on how to achieve synchronized MP movement (RQ-3.D3) over distributed systems, leveraging on components present in our framework.

Contributions of this Chapter

- The definition of IANVS, a modular framework that can instantiate MTD techniques suitable for constrained IoT systems. The framework can synchronize MPs over distributed systems. It is also designed to be crypto agile, i.e., the cryptographic primitives and protocols are meant to be replaced/instantiated with state-of-the-art variants.
- The identification and analysis of components in IoT systems that have potential to become MPs. We focused on Network components using a five-layer model (physical, link, network, transport, and application).

4.2 WHAT TO MOVE? EXPLORING THE NETWORK DOMAIN

"What target to move?" is one of the first questions an MTD system designer has to answer. In this section, we study Network components

1 Pronounced *Ianus*, in honor of the roman god of changes.

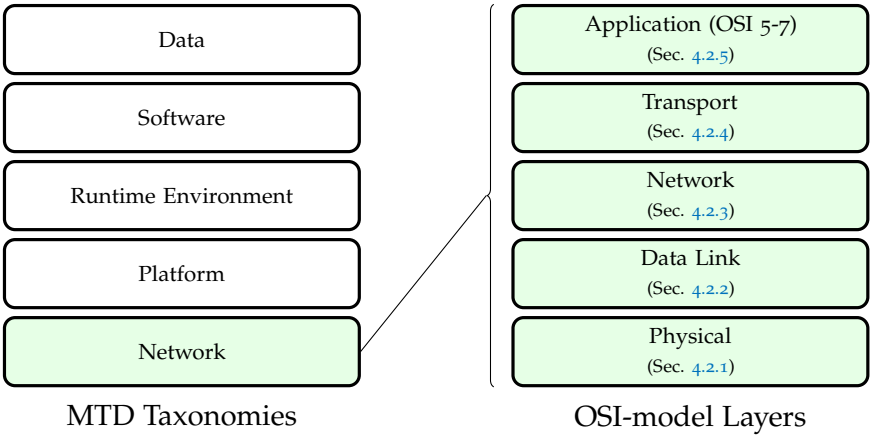


Figure 4.1: MTD Taxonomies and the compacted (five-layer) OSI model's system layers, with a reference to the section in which is studied.

of IoT systems that are suitable for MTD techniques and can provide a particular answer to this design question (RQ-3.D1).

MOTIVATION FOR NETWORK-CENTRICITY In the previous chapter, we observed that most MTD solutions for IoT choose a Network MP. This predominance can be explained by the vital role the network component plays in an IoT system. As seen in the background chapter (Ch. 2.3), a single IoT node does not have many capabilities, but complex services are possible when many are used. Paraphrasing the well-known expression, "the system is more than the sum of its nodes". Therefore, it is not surprising that both cyber attacks [124] and cyber defense [7] efforts are concentrated on the network component of IoT systems.

STRUCTURE OF THIS SECTION In Fig 4.1, we observe the MTD taxonomies with an expanded view on the "Network" category using a compacted five-layer OSI model's division. In the following, we will explore each network layer with concrete examples of IoT systems' components that can be used for MTD techniques, and a mention to existing proposals identified in the previous chapter's SLR.

4.2.1 *Physical Layer*

In Table 4.1, we can observe some common components of the physical layer and its variants.

Table 4.1: Phyisical-layer (OSI L1) components and its variants for IoT.

Component	Variants
Transmission medium	Radio waves Wired transmission media, Sound (atmosphere), Optical wireless comm. (Visible, IR, UV), Molecular comm. (e.g., pheromones, ethanol).
Radio modulation	PSK, FSK, ASK, QAM.
Carrier wave frequency	868 MHz, 2.4 GHz, 5.8GHz, 3kHz (VLF, non-ISM), 12-18 GHz (Ku-band, non-ISM).
Multiplexing	Time-based; Frequency-based: OFDM. Spread Spectrum techniques: DSSS, FHSS. Code-division multiple access (CDMA). "Time-Slotted Channel Hopping" (IEEE TSCH).
Phyisical topology	Mesh, star, bus, point-to-point, ring/tree.
Commercial techonology	IEEE 802.15.4. LPWAN (SubGHz): LoRa, SigFOX, NB-IoT. LTE-Cato/M. IEEE 802.11ah (SubGHz) IEEE 802.11ax (2.4GHz). Bluetooth (2.4 GHz).

OPEN MEDIUM AND JAMMING Wireless communication is a key technology for the IoT. Due to its open nature, the physical layer of wireless systems is a high-priority target for an adversary whose goal is to disrupt the normal behavior of the system. In particular, jamming attacks are one of the most straightforward and effective types of attacks [196]: information flow of the system is stopped or severely disturbed. Anti-jamming [61] is a field of research that can benefit from the MTD paradigm. One of the surveyed IoT techniques illustrate this [176], and we will explore this field in one of the last chapters of this memoir.

PERSPECTIVES: RADIO-AGILITY In general, having multiple radio technologies in the same constraned IoT node is the exemption. Nevertheless, more and more commercially-available hardware (i.e., the same integrated circuit) supports dual-band wireless communications, e.g., a SubGHz (long range) and a 2.4 GhZ technology is present in the Texas Instrument CC1350 microcontroller. In the same frequency range, anti-jamming spread spectrum techniques seem promising. Software-Defined Radio (SDR) technology, combined with MTD, could be a game-changer at this layer if it becomes economically accessible. Three of the studied IoT techniques in the previous chapter have a MP at the physical layer [6, 129, 176].

4.2.2 *Data Link Layer*

In Table 4.2, we depict some common components of the data link layer and its variants.

L2 PANORAMA L2 Address randomization has previously been explored in MTD for IoT [110, 126]. Information security at layer two is also becoming common in the standards (e.g., amendments to IEEE 802.15.4) and security use cases and have many parameters that can be changed. Of the other mentioned L2 components, parameters inside a Medium access control method, or metrics for L2 routing could be suitable to apply an MTD.

4.2.3 *Network Layer*

In Table 4.3, we depict some common components of the Network layer and its variants.

Table 4.2: Link-layer (OSI L2) components and its variants for IoT.

Component	Variants
Medium access control	Collision recovery: ALOHA, R-ALOHA. Collision avoidance: CSMA/CA, MACAW. Collision-free: MS-ALOHA, (CDMA, OFDMA).
L2 Routing ("Mesh-Under")	LOAD, HWMP/IEEE 802.11s, STP, B.A.T.M.A.N.
Metrics for L2 Routing	ETX, RSSI, Fuzzy LQE, Four Bit.
Frame format	802.15.4, LoRaWAN, Profinet, Ethernet, custom.
L2 Address	EUI-64, MAC-48.
L2 Security	(*In general, L2 framing standards define their own secure format*).

THE MOST EXPLORED LAYER The OSI and TCP/IP models' "Network" layer is the most explored by attackers and existing MTD for IoT techniques [10, 45, 57, 80, 93, 126, 164, 197]. The IPv6 protocol is central to this layer. It allows remote parties to interact with the IoT network. As discussed in the background chapter, this *connectivity* is critical to provide new services. Unfortunately, it is also an entry point for attackers that no longer need to be physically close to the system [92].

A PREFERED ATTACK TARGET As disused in the background chapter, aside from the IPv6 protocol, two other protocols are enablers of the IoT: the RPL routing protocol and the 6LoWPAN IPv6 header compression protocol. In terms of existing attacks, [134] focuses only on attacks targeting these two protocols. In more general IoT systems' attack literature [43, 124], the Network attacks account for the majority of the studied cases. As stated before, this layer concentrates most of effort both from attackers and defenders.

FOCUS ON EXISTING IOT MTD AT THIS LAYER We map the techniques from the SLR to their corresponding L3-component MP. First, eight out of a total of eighteen are L3-Network techniques (two being cross-layer), this corroborates the prominence of this layer. Then, among those eight, seven target the IPv6 protocol [10, 45, 80, 93, 126,

Table 4.3: Network-layer (OSI L3) components and its variants for IoT.

Component	Variants
Logical topology	Mesh, Star, Hybrid.
L3 Routing ("Route-Over")	RPL, LOADng, BABEL (RFC 6126), LEACH/MR-LEACH, AODV, DSE, OLSR.
L3 Routing - Metrics	ETX, RSSI, Fuzzy LQE, Four Bit.
L3 Routing - RPL Parameters	Objective Functions (OF): OF0, MRHOF, new OFs (*different OFs can lead to different L3 topologies*).
IPv6 Compression	No compression, 6LoWPAN, SCHC (RFC 8724).
IP Address	Static/EUI-64, DHCP.
L3 Security (IPSec)	IKEv2, ESP (8 variants), ESP and AH (8 variants).

164, 197], of which three leverage on -but do aim to protect- the RPL protocol [10, 45, 126]. Finally, the remaining one deals with the logical topology of the network leveraging on the SDN paradigm [57].

FUTURE AND PRESENT We consider that this layer still has much potential for MTD techniques. Notably, security techniques that target the RPL protocol have not been explored. The SDN paradigm is also very promising. However, most SDN solutions are not adapted for the IoT constraints. Further research effort on making SDN for IoT a reality is needed. As regards IP address randomization, it is established as the most mature MTD technique for IoT and can be used in real deployments.

4.2.4 *Transport Layer*

In Table 4.4, we depict some common components of the Transport layer and its variants.

UDP'S PREDOMINANCE AND L4 PANORAMA UDP is the transport protocol by choice in most constrained IoT systems. Security services at this layer, like DTLS or more recently cTLS, are also used in real

Table 4.4: Transport-layer (OSI L4) components and its variants for IoT.

Component	Variants
Transport protocol	UDP, TCP, SCTP.
Transport protocol - Ports	well-known ports, port randomization, port-knocking (specific sequences or authenticated packets).
Transport-layer Security	DTLS, TLS, cTLS 1.3.
(D)TLS "Cipher Suite"	Choice of cipher suite: +50 exist. (*Combination of authentication and encryption protocols and algorithms*).

deployments. However, security alternatives to achieve end-to-end security are also emerging at the upper application layer as discussed in Ch. 2.4.3. There are two MTD techniques for the IoT at this layer [80, 123], both targeting UDP ports. One of them is our own, and we will describe it in a latter chapter. Aside from port-related MPs, we identify that parameters related to DTLS (like cipher suites and associated keys) could be suitable for implementing MTDs.

4.2.5 *Application Layer*

Finally, in Table 4.5, we depict some common components of the Application layer[2] and its variants.

A RICHNESS OF COMPONENTS Application layer components are abundant in the IoT. As highlighted before, the most prominent is the RESTful protocol CoAP (See Ch. 2.3.2.2). In terms of security-related protocols (See Ch. 2.4.3), application-layer solutions are emerging like COSE and OSCORE. They offer great flexibility as security parameters can be set-up on a per-message basis (as opposed to transport layer DTLS). MTD techniques could exploit this per-message flexibility. Finally, OAuth authorization tokens are central to the the ACE authorization framework for IoT, and those tokens have many parameters that are suitable for MTDs.

Five MTD for IoT techniques have MPs in this layer [4, 41, 64, 123, 189]. Because of its flexibility and the growing importance of IoT research effort in the application layer, we believe this layer has great potential to be used as a MP source for novel MTD techniques.

2 We grouped OSI's layers 5 to 7, equivalent to the App. layer of the TCP/IP model

Table 4.5: Application-layer (OSI L5-7) components and its variants for IoT.

Component	Variants
Application Protocol	CoAP, MQTT, LWM2M, HTTP, XMPP.
Media-Type	CBOR, JSON, Binary, XML, HTML, JavaScript, JPG.
Resource IDs in RESTful interfaces (e.g. URLs)	Resource discovery, well-known resources, resource directories.
Application-layer Security	COSE, OSCORE, JOSE.
App-layer Security: COSE	Crypto algorithm change, key change, cascade encryption.
ACE-OAuth Framework for IoT	*proof-of-possession* tokens, *bearer* tokens. Many tokens' components can be modified: expiration, associated authorizations, associated key.
Application plain text (Syntax)	We can modify the syntax while maintaining the semantics.

4.2.6 *Network-layers Summary*

In this section, we reviewed common Network components of IoT systems. We divided this section using a compacted five-layer OSI model and at each layer we identified relevant components and its variants. These components can potentially become MPs in MTD techniques if enough variants could be realistically used. We also referenced existing MTD for IoT techniques that use MP of each layer, leveraging on our SLR.

Overall, components such as L2 and IP addresses as MPs have been extensively explored in the literature of MTD for IoT, ten out of eighteen proposals randomizes either the L2 or L3 address. This validates their suitability for becoming MPs. However, we identified many other components that are promising for MTD and have not been thoroughly explored, most notably, application-layer components.

In the following section, we develop a generic distributed MTD framework that can be -potentially- used to implement any distributed MP. In the third part of this thesis, we will use it to define three novel techniques at three different network layers: physical, transport, and application.

4.3 HOW TO MOVE? IANVS, AN MTD FRAMEWORK FOR THE IOT

In this section, we provide an answer to the question of "How to move distributed MPs in IoT systems?" (RQ-3.D2) by introducing a generic framework called IANVS. Indeed, there is more than one answer possible to this RQ. In our approach, we seek to be as generic as possible while keeping the abstractions closely related to topics well-studied, solved or resolvable, and implementable in constrained IoT systems.

4.3.1 *Presentation and Rationale*

We answer RQ-3.D2 with a proposal called IANVS (pronounced *Ianus*, in honor of the roman god of changes). IANVS is a generic MTD framework suitable for IoT systems. Our framework has a modular design. Its components can be adapted according to the specific constraints and requirements of a particular system.

IANVS abstracts, generalizes, and links common components of MTD strategies. A concrete MTD strategy design can use IANVS as an

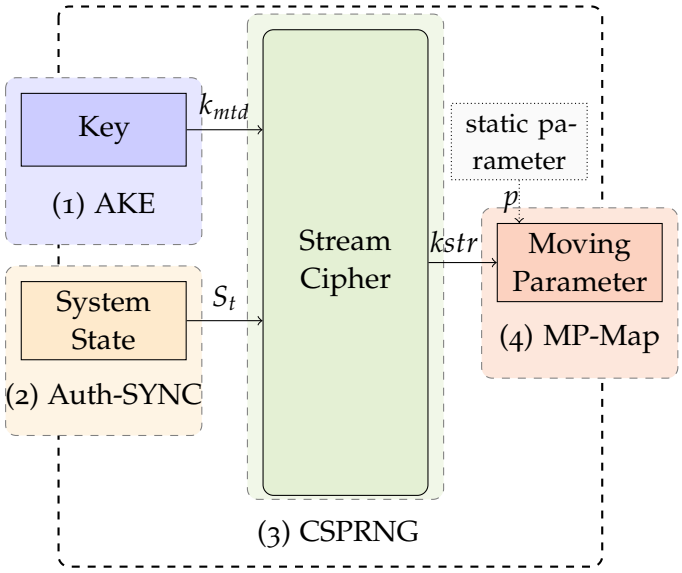


Figure 4.2: IANVS MTD Framework components.

archetype to build upon it. The main goal of IANVS is to leverage future design and implementation work, with the aim of having more secure and usable MTD for the constrained IoT (RQ-3). For example, components may be identical among different concrete proposals and could be re-used. Security proofs of a specific MTD strategy may ease or validate those of another one. In the third part of this memoir, we present three concrete MTD strategies that use IANVS as a design skeleton.

4.3.2 *IANVS Schema*

In Fig. 4.2, we present a diagram of IANVS and its four components:

1. AKE: An Authenticated Key Establishment mechanism.
2. Auth-SYNC: An Authenticated state Synchronization mechanism (e.g., authenticated time).
3. CSPRNG: A Cryptographically Secure Pseudo-Random Number Generator, from the stream ciphers family [147].
4. MP-Map: A transformation that outputs values in the MP domain with equiprobability (e.g., uniform hashing).

The AKE and Auth-SYNC components provide two inputs to the framework (i.e., a cryptographic key and a system state), and the MP-Map produces the output, i.e. the MP value. Optionally, a static parameter value p can be an MP-Map input.

4.3.3 *Components Details*

In the following, we detail each IANVS component and their interactions. We explicit the security requisites for the inputs and outputs of each component. We also provide examples of suitable candidates to instantiate concrete implementations of those components in IoT systems.

(1) The AKE component provides a cryptographic key (k_{mtd}). The k_{mtd} is *secret* and must only be shared by the trusted parties of an MTD strategy. AKE is related to the secure key bootstrapping problem. Key bootstrapping is a hard problem to solve and, in general, relies on pre-shared cryptographic material or a trusted third party. Suitable AKE component candidates for the constrained IoT setting are AKE protocols like EDHOC [156], our previous publication about nonce-based AKE within the ACE-Oauth IoT framework [122], or a run of the DTLS Handshake protocol with a lightweight cipher suite.

(2) The Auth-SYNC component provides a system state value (S_t). The S_t must be *authenticated* and *fresh*, but not necessarily secret. Auth-SYNC is closely related to the secure time synchronization problem. There is currently a lack of suitable solutions for the constrained IoT. We will assess the state of the art of secure time synchronization in IoT and provide a suitable proposal in Ch. 5 this manuscript.

(3) The CSPRNG component is at the core of the IANVS framework. It requires two inputs: (i) a cryptographic key (k_{mtd}), and (ii) an authenticated system state (S_t). They are provided by the AKE and Auth-SYNC components, respectively. The CSPRNG produces one output, a CSPR binary key-stream (*kstr*). A stream-cipher [13] must be used as a Cryptographically Secure Pseudo-Random Number Generator (CSPRNG). A stream-cipher produces long binary outputs with strong cryptographic guarantees. Also, many IoT suitable options exist [131] like software-based ChaCha20, or AES in Counter Mode (CTR)[3] that is present in most modern IoT hardware[4].

(4) The MP-Map component outputs the MP configuration value. It takes as inputs the key-stream (*kstr*) from the CSPRNG and -optionally- a static parameter value p. It maps the inputs to a value in the MP domain. The MP-Map component is related to the problems of Map structures, i.e., a collection of *key-value* pairs. If there is only one MP in the domain, one possible solution is to use hash tables, i.e., rely on a hash function to do the transformation. The Uniform Hashing Assumption must be approximated, i.e., every *key* should be mapped to a *value* domain with equiprobability.

3 It behaves as a stream-cipher.
4 In the form of AES-CCM crypto-processors.

4.3.4 *Security considerations*

The security of a complex system is not additive but "as strong as the weakest link". In general, security is not even composable. The composition of secure sub-systems can lead to an insecure system. In the current schema form, IANVS uses a *secret key-authenticated nonce* pair and a stream cipher to produce a key-stream. In a single-MP system, this output can be almost directly used as the MP value. The cryptographic properties of the MP movement will be as strong as that of the underlying stream cipher. However, as an example of non-composability, suppose using an AES-Cipher Block Chaining (CBC) *block* cipher instead. In that case, a time-based (i.e. a predictable nonce) AUTH-Sync IANVS instantiation will be broken because AES-CBC requires the nonce to be unpredictable for an attacker.

In the current memoir, we do not present a formal method proof of the security of IANVS, because the proof will depend on the particular instantiation of each module. However, in Ch. 5 we present a formal proof of an instantiation of the AUTH-Sync component. We will use IANVS, for the most part, in MTD techniques where the stream-cipher is the central component, and we focus on attacks targeting the MP domain. However, we write this subsection to make the reader aware of the aforementioned shortcomings and to advocate for formal proofs of crypto protocols and systems.

To conclude this section, we remark again that IANVS provides a generic design framework, and many decisions are -intentionally- left open. They will have to be answered within the context of a particular MTD technique.

4.4 WHEN TO MOVE? SYNCHRONIZED MOVEMENT IN DISTRIBUTED SYSTEMS.

This section aims to provide a partial answer to the question of *when to move?* (RQ-3.D3). We designed IANVS to help in the design and implementation of MTD strategies in *distributed* systems. The AKE and Auth-SYNC components provide this distributed capability, and enable to make effective the moving decision.

THE NEED FOR SYNCHRONIZED MOVEMENT A particular MTD strategy in a distributed system requires that the involved parties (e.g., IoT nodes) agree on the MP value over *time*. This means that the decision of *when to move?* must be effectively *executed* in a synchronized way. If not, some system nodes will have an incorrect representation of

the current MP value, with the potential implication that the system's services might not be fulfilled.

SYNCHRONIZATION USING AUTH-SYNC As depicted in Fig. 4.2, the current MP value is determined[5] by the values of the Key k_{mtd} (provided by the AKE component) and the System State S_t (provided by the Auth-SYNC component). Thus, k_{mtd} and S_t must be synchronized in order for the MP value to also be in a consistent state. The AKE component is meant to be executed rarely, e.g., potentially only once in the lifetime of the node. Having to re-key should only be needed in exceptional cases. For example, when all the possible values of S_t were used, one node should be revoked from the MTD strategy, or the key has been compromised. On the contrary, the Auth-SYNC component is meant to be executed often, e.g., potentially once per MP movement. The Auth-SYNC component should be used to synchronize the MP movement.

TIME-BASED MOVEMENT: AUTH-SYNC WITH TIME If the nodes have Real-Time Clock (RTC) capabilities, time-based synchronization solutions are applicable. It suffices to agree on a period of movement, run a secure time synchronization protocol, and then the MP movement can be triggered internally by the nodes. The time synchronization protocol might be executed again, depending on the synchronized internal clock accuracy needed by a particular MTD strategy.

EVENT-BASED MOVEMENT: AUTH-SYNC WITHOUT TIME If the nodes do not have RTC capabilities, the MP movement should be actively triggered by a node in the system and distributed to the others. A Master-Slave solution could be applied. The protocol should ensure that the trigger message is authenticated and fresh. To guarantee freshness (i.e., to avoid replay attacks), at least a two-message nonce-based protocol is needed. The protocol will be executed at every MP movement.

TIME, EVENT, OR HYBRID-BASED MOVEMENT? As seen in Ch. 2.2.3, there are three MTD general design solutions to decide *when to move?*: time, event, or hybrid-based solutions. Time-based MTDs will lead to proactive techniques, while event-based MTDs provide the means to define reactive or adaptive techniques (e.g., interacting with an intrusion detection system). The most advanced MTD systems will have an adaptive hybrid-approach. In this sub-section, we explored

5 After fixing an implementation of CSPRNG and MP-Map components

how both fundamental time or event solutions can be achieved in distributed systems using the Auth-SYNC component of IANVS.

A PARTIAL ANSWER Further design or implementation details, such as the period of movement or which events trigger the movement, will depend on the IoT use case and particular MTD technique. Also, other aspects of *when to move?* such as how do different period values affect an MTD system service or mitigate attacks, need other levels of abstraction (higher or lower) and system-attacker models not present in the current chapter. We consider the current subsection a partial design answer to RQ-3.D3.

4.5 CONCLUSION

In this chapter, we answered three fundamental MTD *design* questions originated from RQ-3. In order to design usable and secure MTD techniques, *what* components (RQ-3.D1), *how* (RQ-3.D2), and *when* (RQ-3.D3) to move them in IoT systems?

WHAT We focused on IoT system's Network components, and identified many suitable to become MPs for MTDs. Particularly, novel MTD solutions that target application-layer, SDR-leveraged, and SDN-leveraged components have the most potential. L2 and L3 addresses as MP have been extensively explored in the MTD for IoT literature.

HOW AND WHEN The IANVS MTD framework is our *design* answer to "how to move?" (RQ-3.D2), and also provides the elements to implement a particular answer to "when to move?" (RQ-3.D3). The KISS[6] principle -that advocates for simplicity- inspired our design. We designed IANVS to be modular. Each module corresponds to a research field on its own. Thus, a concrete implementation can leverage on state-of-the-art and well-established solutions.

CONCRETE USE CASES AND TECHNIQUES The design is only the first step in the creation of *usable* and *secure* MTD techniques for the IoT (RQ-3). In our SLR we advocated for the empirical usability of the MTD for IoT techniques because a concrete MTD technique should solve a concrete problematic. The prototype of IANVS was conceived as a solution to a concrete problematic (i.e., insider-node jamming in IoT), even if for technical-narrative purposes we presented it before this concrete application. In the rest of this memoir, we decrease from the

6 "Keep It Simple, Stupid"

level of abstraction in this chapter, and approach more concrete use cases and solutions.

The final goal of this manuscript is to improve the resilience of IoT systems though the use of MTD-based techniques *in the context of real use cases*. To formalize this research goal, we pose the following RQ, still derived from RQ-3:

RQ-3.I: How to instantiate MTDs in concrete IoT use cases?

This RQ will guide the third and final part of this memoir. IANVS is an essential contribution of this thesis and will be at the core of the *concrete* MTD techniques for IoT systems that will follow.

Research Questions

RQs answered:

- **RQ-3**: How to create *usable* and *secure* MTD techniques for the constrained IoT? (*Design* considerations)
  - **RQ-3.D1**: What are suitable MPs in IoT systems?
  - **RQ-3.D2**: How to move distributed MPs in IoT systems?
  - **RQ-3.D3**: When to move the MP? (*partially*)

RQs raised:

- **RQ-3**: How to create *usable* and *secure* MTD techniques for the constrained IoT? (*Instantiation*)
  - **RQ-3.I1**: How to instantiate MTDs in concrete IoT use cases?

Part III

CONSTRUCTION

The third part of this memoir is about the *instantiation* of usable and secure MTD techniques for the constrained IoT.

We define a secure time synchronization protocol and three concrete IANVS-based Network MTD techniques with MPs from the application, transport, and physical layers.

5

A SECURE TIME SYNCHRONIZATION PROTOCOL FOR IOT

5.1 INTRODUCTION

> But time
> Keeps flowing like a river
> To the sea
>
> "Time" by The Alan Parsons Project
> (*The Turn of a Friendly Card*, 1980)

In the previous chapter, we provided considerations and proposals about the *design* of MTDs in IoT systems. This chapter opens the third

and final part of this manuscript, which is about the *instantiation* of concrete MTD proposals to concrete IoT problems (RQ-3.I1).

The IANVS framework (See Ch. 4.3) will be the golden thread of this last part of the memoir. The latter chapters instantiate three IANVS-based MTD techniques, while the current one deals with the time synchronization problem. A fully developed IANVS-based MTD technique for IoT systems requires the instantiation of IANVS's four components (i.e., AKE, Auth-SYNC, CSPRNG, and MP-Map). In this chapter, we do not present an MTD proposal, but we tackle the fundamental problem of *secure time synchronization*. In other words, we provide a solution for the instantiation of a *time-based* Auth-SYNC component (See Ch. 4.4).

Time synchronization is a fundamental service for a wide variety of IoT applications, including security-related ones. Particularly, it is a prerequisite to time-based MP movement for MTDs in distributed systems. However, there is no standardized nor lightweight *secure* time synchronization solution suitable for IoT systems. This chapter presents our solution to this problem, the LATe synchronization protocol. Our proposal is agnostic to underlying communication technologies and leverages on IETF open standards. To enhance the readability of this chapter, we include the LATe messages' encoding definitions in Appendix B.

Contributions of this Chapter and its Appendix

- We define the LATe synchronization protocol. Our protocol provides a solution for secure time synchronization in constrained IoT systems.
- We provide a computer-aided proof of the security claims of LATe using the Scyther tool and discuss real-world attacks, mitigations, and implementation issues.
- We define LATe messages' application-layer encoding using IETF's CBOR and how to secure them with COSE. This messages' syntax contributions are in Appendix B.

5.2 STATE OF THE ART OF SECURE TIME SYNCHRONIZATION

MOTIVATION FOR IOT SECURE TIME Synchronized time is needed in several IoT applications, from time-stamping of sensor data to the establishment of authenticated secure channels. However, many time synchronization protocols are not secure: they assume existing secured communication channels. The establishment of secure channels, in

most cases, assumes a secure source of time e.g., to assure freshness of transactions. This creates a *circular dependence* problem that has already been spotted on the standardization community. Time protocols are being designed to overcome this, such as the IETF's work-in-progress Network Time Security (NTS) [51]. However, NTS or secure-versions of existing standardized time protocols, are not designed for the IoT constraints.

NON-SECURE STANDARDIZED TIME SYNCHRONIZATION Prominent standardized time synchronization protocols are the IETF Network Time Protocol (NTP) [27], IEEE 1588 Precision-Time-Protocol (PTP), and satellite-based Global Navigation Satellite System (GNSS). An excellent overview of time synchronization protocols over packet-switched networks is done in [101], it also analyses security threats and solutions. Moussa et al. [121] focus on time synchronization for the smart grid and its security requirements.

SECURE STANDARDIZED TIME SYNCHRONIZATION Current standardized solutions to achieve secure time synchronization include *Annex K* of PTP, and authenticated mode of NTP. Design of secure time synchronization protocols from scratch is an active topic, such as the aforementioned NTS [51]. The IETF has released a document [118] that specifies the threats and security requirements for future time protocols. Current standardization efforts do not deal with the specific constraints of IoT, and focus mostly on precision and robustness at the expense of increased requirements at the node and network. A standard suitable for IoT is an unsolved problem.

SECURE NON-STANDARDIZED TIME SYNCHRONIZATION IN IOT Outside standardization bodies, the secure time synchronization problem has been prominently studied for WSNs [53][172][62][20]. WSNs share many of the IoT constraints[1]. However, the aforementioned solutions either require already loose time synchronization, use asymmetric cryptography, or they use nonces but requiring more messages exchanges than our proposed solution. On Section 5.6, we will compare them to our proposed solution. Furthermore, unlike LATe, none of the proposed secure time synchronization methods have been formally proven.

1 Constrained IoT systems can be WSNs.

5.3 THE LATE SYNCHRONIZATION PROTOCOL

In this section, we present the LATe synchronization protocol. We define the involved entities, protocol goals, and provide a functional description and the messages' semantics. In Appendix B, we provide the messages' encoding using IETF standards.

5.3.1 *Background*

The non-cryptographic part of the proposed protocol can be traced to Cristian's time synchronization protocol [37]. However, the problem that needs to be solved concurrently is related to security and is expressed in the following RQ:

> "How to assure the freshness and authentication of an exchange of information *in the absence of time-awareness*?"

The concept of authenticated and fresh exchange of information (i.e. the previous RQ) is generalized and solved by Bauer et al. [18] with the concept of *event-markers*.

Our proposal is intended to be the simplest possible to the secure time synchronization problem; namely, using an *event-marker* for a two-message protocol. Unlike existing and less-lightweight proposals, our contribution has the added value of using open standards suitable for the IoT and presenting a computer-aided security proof.

5.3.2 *Protocol Entities and Hypothesis*

The nonce-based LATe Synchronization Protocol is our proposal to securely bootstrap time, and involves two entities:

- *Time Client (TC)*: the entity that attempts to update its local time representation.
- *Time Server (TS)*: the entity that provides its local time representation.

Hypothesis: TC and TS have valid pre-shared cryptographic material, and the messages are transported over unsecured communication channels.

5.3.3 *Protocol Goals*

Functional Goal:

1. Provide an entity (i.e., the TC), with the time representation from a trusted party (i.e., the TS).

Security Goals:

1. *Data Authentication*. The time representation must be *data-origin authenticated*; i.e., coming from the intended party.

2. *Data Integrity*. The time representation must be *integrity-protected*; i.e., an alteration of the original information must be detected.

3. *Freshness*. The time representation must be *fresh*; i.e., it corresponds to the current run of the protocol and not replayed from an earlier run.

Design Goals:

1. *Lightweight*. Minimize the number of messages to exchange; minimize the cryptographic operations to execute (in terms of complexity, that will be equivalent to minimize CPU processing power-time needed at the entities); minimize the information to exchange and provide a compact-representation of the information over the channel[2].

2. *Agnostic to underlying communication technologies*. The protocol messages should be easily transported over any underlying communication technology (wired, wireless, Ethernet, IP, non-IP, datagram oriented, etc)[3].

3. *Cryptographic agility*. The crypto-primitives used by the protocol must be easily interchangeable, e.g., ready for future algorithms, or if an attack is discovered, easy to replace the current with another.

Non-goal: Precise fine-grained time synchronization is not a goal. E.g., it is not a goal to synchronize at the order of μs; but rather at ms, s, or even minutes. The time synchronization error will be of the same order of magnitude than the network's round-trip delay time.

2 Not a semantic goal but strictly related with the syntax of the protocol.
3 Idem footnote 2.

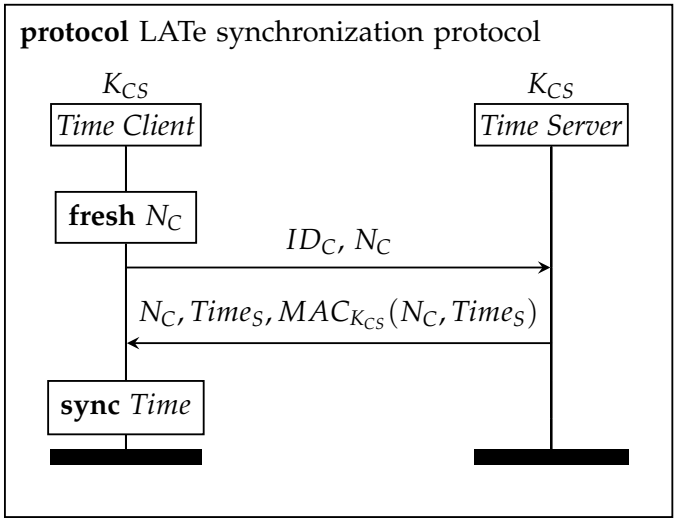


Figure 5.1: LATe synchonization protocol diagram. K_{CS} is a symmetric pre-shared key between TC and TS. ID_C is the identity representation of TC. N_C is a nonce generated by TC.

5.3.4 *Definition*

The LATe synchronization protocol consists of two messages exchanged between a TC and a TS. K_{CS} is a symmetric pre-shared key between TC and TS. $MAC_{K_k}(m)$ is a MAC of message m using shared key K_k.

A LATe protocol run happens as follows:

1. *TC* generates a random nonce N_C
2. *TC sends to TS Message 1*. Containing: ID_C the identity representation of *TC*, and N_C.
3. *TS sends to TC Message 2*. Containing: N_C, $Time_S$ the local time representation of *TS*, and $MAC_{K_{CS}}(N_C, Time_S)$ a message authentication code of N_C and $Time_S$ using the key K_{CS}
4. *TC* can synchronize its internal time representation according to Subsection 5.3.5

The protocol is more formally described in Fig. 5.1.

5.3.5 *Time Calculation*

The TC will have to run the following steps to achieve authenticated time synchronization:

1. Timestamp when it sends *Message 1*: T_1.
2. Validate *Message 2*, this involves:
   a) Verify that nonce N'_C in *Message 2* matches N_C sent in *Message 1*. (*Freshness*)
   b) Verify *data authentication and integrity*. TC calculates $MAC_{K_{CS}}(N_C, Time_S)$, and compares the result with the MAC value in *Message 2*.
3. Calculate the *Round Trip Time (RTT)*, as $RTT = T_2 - T_1$. T_2 is the local time of TC when performing this calculation.
4. Set the internal time representation T_C, as $T_C = Time_S + \frac{RTT}{2}$. The associated uncertainty of this value is $\pm\frac{RTT}{2}$.

5.4 LATE'S FORMAL-METHOD VERIFICATION USING SCYTHER

5.4.1 *State of the Art of Security Protocols Verification*

PROVABLE SECURITY AND FORMAL METHODS There are currently two main approaches to verify security protocols: the *provable security* and the *formal method* approach. *Provable security* defines a rigorous framework to define and prove (*theorem-proof*) cryptographic properties from a *mathematical* point of view, proving a protocol secure is hard on the provable security approach, and although there is criticism to this approach [89] it is still regarded as the most sound proof possible for a protocol. The *formal method* approach proposes a simpler model to describe and analyze cryptographic protocols, by abstracting basic properties (e.g., encryption), it assumes perfect cryptography (e.g., the crypto-primitives can not be broken), and the attacker capabilities need to be modeled also (and restricted), then logical flaws can be found on such model. Several *formal methods* exist; the most known is the Burrows-Abadi-Needham (BAN) logic or *logic of beliefs*, and is deprecated: flaws have been found on protocols that have been proved secure on the BAN logic.

COMPUTER-AIDED TOOLS State-of-the-art approaches include the automatic falsification or verification of protocols with computer-aided tools like: Coq, CertiCrypt, EasyCrypt and CryptoVerif, all these aimed to achieve or help to manually achieve *computational security* -a subset of provable security-, in which the proof of security is *reduced* to the computational infeasibility of solving some mathematical problems for

an adversary e.g., semi-prime factorization[4]; on the other hand, tools like ProVerif, Scyther, and Tamarin are all three on a higher abstraction level (*formal methods* assuming a particular attacker model, e.g., the Dolev-Yao and perfect cryptography), they provide a weaker proof than a computational security one, but is easier to model complex cryptosystems.

5.4.2 *The Scyther tool and a formal proof of LATe*

We chose the Scyther tool [36] to present a formal proof of LATe. The reasoning behind this choice is Scyther's simplicity to model cryptosystems, the attacker model found adequate to our setting, and the possibility to find concrete attacks. Scyther assumes perfect (or black-box) cryptography: the cryptoprimitives can not be broken. Another important assumption is the Dolev-Yao adversary model [44]. In Dolev-Yao an adversary has complete control over the communication channel: it can eavesdrop, intercept; modify, delete, and insert any message; the adversary could be a legitimate user of the network.

To prove LATe's security claims using the Scyther tool, we needed to accomplish two non-straightforward tasks: (1) express a MAC function (the crypto primitive does not exist); (2) express the security-authentication goals claimed.

To represent a MAC function over message m we use two primitives: $Enc_k(m)$ symmetric encryption of message m using key k, and a non-cryptographic hash function $H(m)$ (a hash function on Scyter is a one-way-function and known to every agent). Then, to obtain the keyed $MAC_k(m)$ of a message m we chose to an encrypt-then-hash method, as follows $H(Enc_k(m))$. The captured semantical meaning is that only an agent in possession of the key k is able to produce this one-way function output over m.

Regarding the modeling of the *authentication* and *freshness* claims, Scyther offers the check of *secrecy* of a variable m, and the following notions of authentication: *aliveness*, *weak agreement*, *non-injective agreement* and *non-injective synchronization*. *Non-injective synchronization* requires that *all protocol messages occur in the expected order with the expected values*. Proving *non-injective synchronization* will implicitly include aliveness, weak agreement and non-injective agreement. For a deep analysis on authentication hierarchies and precise definitions see [106] and [35].

Finally, the LATe synchronization protocol defined using the Security Protocol Description Language (SPDL) from Scyther is shown in Listing 5.1.

4 These methods cannot find particular attacks just prove they exist

Listing 5.1: LATe Protocol in Scyther's SPDL.

```
# LATe: Authenticated Time Synch Protocol

hashfunction H1;
usertype TimeStamp;

protocol LATe(I,R)
{
 role I   # Time Client - Initiator
 {
  fresh Na : Nonce;
  var T : TimeStamp;

  send_1(I,R,I,Na);
  recv_2(R,I,Na,T,H1({Na,T}k(I,R)));#encrypt-then-hash

  claim_I1(I,Nisynch); #encrypt-then-hash
  claim_I2(I,Niagree);
  claim_I3(I,Alive);
  claim_I4(I,Weakagree);
 }

 role R # Time Server - Responder
 {
  var Na : Nonce;
  fresh T : TimeStamp;

  recv_1(I,R,I,Na);
  send_2(R,I,Na,T,H1({Na,T}k(I,R)));#encrypt-then-hash
 }
}
```

5.4.3 *Results*

We verify our protocol using Scyther v1.1.13 compiled from source running on OS Ubuntu 17.04 x64. The Scyther settings are: Maximum number of runs 0 (unbounded), Matching type "find all type flaws", advanced parameters were left to default values. The results are the following: all claims have been verified (*Nisynch*, *Niagree*, *Alive* and *Weakagree*). Notably we achieved *non-injective synchronization* for the protocol. Secrecy of the *server time* was not a goal. The *data authentication-integrity* claims are satisfied by these results. However, the *non-injective synchronization* does not guarantee, by itself, the *freshness* goal of the LATe protocol, we will discuss this in Section 5.5.

5.5 ATTACKS, MITIGATIONS AND REAL-WORLD ISSUES

This section studies possible attacks, its mitigations, and discuss other real-world issues that affect the LATe protocol.

5.5.1 *Replay-attack, Injectivity and the Freshness claim*

Our protocol satisfies the notion of *non-injective synchronization*, however, this is not enough to claim resilience to *replay-attacks*. This kind of attacks can be formally ruled out by the notions of *injective agreement* and *injective synchronization*. *Injective-synchronization* is the strongest notion of authentication on the model we are using and -informally- is defined as follows: *"an Initiator I considers a protocol injectively synchronizing if the protocol (non-injective) synchronizes and each run of I corresponds to a unique run of Responder R."*. The *freshness* goal of our protocol is strictly related to the *injectivity* property. The question arises if our protocol satisfies *injective synchronization*, while we will not make a formal proof, that will involve to prove the LOOP property proposed in [35], but an affirmative response can be done, informally justified by observing that every client run will have a unique and unpredictable Nonce N_i which is used in all the messages exchanges with the Server in that run. This guarantees a one-to-one correspondence between all the messages of the same run, and message from others runs will not be able to be injected. On the formal model, a response that matches the nonce on the request, corresponds to the current run of the protocol and not another, it is *fresh*.

5.5.2 *Real nonces and pre-play attack*

The *injective synchronization* claim, who assure *freshness*, relies on the (idealized) properties of the Nonce as being unique and unpredictable. On practice this will not be the case, and the guarantees will be limited by the randomness quality of the nonce generation and by its length (not infinite). Shorter nonces will be more prone to collisions and *pre-play* attacks e.g., an attacker obtaining all possible nonce responses from the server, will be able to reply these responses -with old values of time- to any future client run of the protocol. To mitigate this risk one straightforward solution is to use longer nonces: e.g., 128-bits (the MAC-tag should also be increased accordingly). To make *pre-play* attacks infeasible (i.e. an attacker will not be able to obtain responses from the server to inject on the client) we define a stronger version of the protocol that includes the authentication of the first message

as shown on Listing 5.2. **Avoiding randomness**: Authentication of the first message allows another refinement, the nonce does not need to be random, and a counter (i.e. a sequence number) will suffice; the counter value must be stored on persistent memory to avoid being reset by an attacker.

Listing 5.2: LATe w/MAC of first message (N_C can be a counter).

1 $C \rightarrow S : ID_C, N_C, MAC_{K_{CS}}(ID_C, N_C)$
2 $S \rightarrow C : N_C, Time, MAC_{K_{CS}}(N_C, Time)$

5.5.3 *Reflection Attack*

Another attack can be done if the TC also acts as a TS: on the original LATe protocol an attacker can use a message generated by the actor in the Time Server role, to be injected in another run of the protocol with the same actor acting as a TC. The modified version on Listing 5.2 does not suffer from this attack. This can also be avoided if the second message includes the recipient's ID in the MAC.

5.5.4 *Symmetric Cryptography: Server Key-Management Issues*

The use of symmetric cryptography comes at a burden at the server: it has to keep a copy of all clients' keys. We assume an IoT setting where the constrained node (i.e. TC) has a well-known trusted party which it uses for many purposes e.g., an *Authorization Server* (AS) as defined in IETF's ACE-OAuth framework [155]. On such a setting, the AS can also act as a TS. LATe has also the flexibility to use asymmetric crypto to relieve the TS key management issues if fits better the envisioned IoT use case.

5.5.5 *Protocol refinement*

Using the Scyther tool we verified that the same security claims from the original LATe synchronization protocol are hold true in a protocol using a more compact Message 2. By omitting the Nonce in the response, but still using it to calculate the MAC, all the security claims hold still true and we achieve a non-negligible gain in message size. This can be done only if we assume that a TC can run only one concurrent run of the protocol (i.e. when receiving a response it

can assume implicitly the nonce to use to calculate the MAC), this assumption is reasonable.

To conclude this section, we gather all the mitigations proposed for attacks and this optimization, and the LATe v2 synchronization protocol in Listing 5.3.

Listing 5.3: LATe v2 synchronization protocol.

1 $C \rightarrow S : ID_C, N_C, MAC_{K_{CS}}(ID_C, N_C)$
2 $S \rightarrow C : Time, MAC_{K_{CS}}(ID_C, N_C, Time)$

The authentication of the first message that mitigates completely pre-play attacks, can also be used to mitigate Denial-of-Service attacks at the server-side. A version that does not authenticate the first message is still useful on real environments, if the users are aware of the pre-play and nonce considerations of Section 5.5.2.

5.6 COMPARISON OF TIME SYNCHRONIZATION PROTOCOLS

BASELINE To define a common *baseline* to compare several time synchronization protocols, we do not take in account underlying layers overhead (e.g., IEEE 802.15.4), but only application data. We also simplify the encoding of the messages, assuming no overhead for metadata, and we assume the following data sizes: a *Timestamp* representation is 4 bytes, a *Node Identity* is 2 bytes, a *Nonce* is 8 bytes, and a *MAC* is 8 bytes. In E-SPBS [20] an ECDSA signature is 48 bytes; In Guo et al. [62] we assume an Unspecified Signature being of 16 bytes, and non-cryptographic hash 16 bytes; In [53][172] syn-ack information of 1 byte. In Table 5.1, we show the results.

STANDARDS, OTHER ORDER OF MAGNITUDE We also calculated values for *NTS Extensions for NTPv4* after Key Establishment [51]: 2 Messages; 134 bytes avg. msg. size; 268 total bytes; 2 AEAD (symmetric) operations. And for *PTP with Annex-K* after Security Association: 4 Messages; 128 bytes avg. msg. size; 512 total bytes; $4 \times MAC$. Both are one order of magnitude greater due to the calculations taking in account real applicative messages and not simplified encoding.

LATE MINIMIZES In a battery-powered constrained IoT node, *energy* is the scarcest resource and, simplifying, the total bytes to be exchanged over the radio is the most important factor to minimize. LATe minimizes both the number of messages and the total bytes count,

Table 5.1: Secure time synchronization protocols baseline comparison.

Protocol	Nr. of Msg.	Avg. msg. size (Bytes)	Total Bytes	Crypto Ops. at Node
SPS [53]	2	21	41	$1 \times MAC$ $1 \times Nonce$
E-SPS [53]	3	17	50	$1 \times MAC$ $1 \times Nonce$
TinySeRSync [172]	2	21	42	$2 \times MAC$
Guo et al. [62]	3	39	116	$2 \times Signature$ $1 \times MAC$
E-SPBS [20]	3	35	104	$1 \times Signature$ $1 \times Nonce$
LATe	2	15	30	$1 \times MAC$ $1 \times Nonce$
LATe v2	2	15	30	$2 \times MAC$ $1 \times Nonce$

needing $\approx$ 25% less application data exchange than the second-lowest Secure Pairwise Synchronization Protocol (SPS)[53]. This percentage will vary if we include other protocols' overhead, or change the application data representation estimations; however, LATe will still be strictly inferior. In terms of cryptographic burden, LATe is also the lightest, with one MAC operation and one nonce generation.

5.7 CONCLUSION

In this chapter, we provided a solution to the fundamental problem of *secure time synchronization* bootstrapping: the nonce-based LATe synchronization protocol. Our protocol was designed for constrained IoT systems; it minimizes the number and size of messages, and the number of cryptographic operations needed at the synchronizing node.

This chapter also highlighted the use of computer-aided tools to prove the security claims of network protocols. We used a formal method that -even if not as mathematically sound as provable security- allows to efficiently detect security flaws early, in the design process. There is a large amount of time and research effort to be done in order

to go from a protocol model/design to its effective implementation in IoT hardware. We provide in Appendix B a bit-by-bit detailed encoding of LATe's messages using application-layer IETF's standards CBOR and COSE. We also discussed some real-world implementation issues, attacks, and mitigations, that were not captured by the formal method's model. The issues that arise when instantiating a design into reality are universally valid. The particular use case will determine the level of effort put into defining a model and validating a particular instantiation according to some goals.

In respect to this memoir's main RQs, LATe does not fully answer the question of "How to instantiate MTDs in concrete IoT?" (RQ-3.I1). But, as we discussed in this chapter's introduction (Sec. 5.1), it fulfills a necessary condition of a complete answer. LATe is a concrete answer to the instantiation of a time-based *Auth-AKE* component for IANVS-based MTD techniques.

In the remaining chapters, we will present three concrete MTD techniques -albeit two deliberately similar-, that will provide more concrete answers to RQ-3.I1.

Research Questions

RQs answered (partially):

- **RQ-3.I1**: How to instantiate MTDs in concrete IoT use cases?
  - [Auth-SYNC] We instantiated a secure time synchronization protocol, suitable for time-based *Auth-SYNC* components of IANVS-based MTDs (*Design*, and formal-method security *Evaluation*).

6

TWO IANVS-BASED NETWORK MTD TECHNIQUES

6.1 INTRODUCTION

In this chapter, we define two concrete MTD techniques that instantiate the IANVS framework (Ch. 4.3). They share all design components except for the MP-Map and corresponding MP. Both are from the Network category. The MP in the first technique is the UDP port number of a service (transport layer), while in the second one a CoAP well-known URI (application layer).

The question of "How to instantiate MTDs in concrete IoT use cases?" (RQ-3.I1) is partially answered. The proposals provide evidence on using IANVS as a base design to develop concrete and usable MTD techniques. The use case that motivated both proposals is an IoT node hosting a CoAP server and an attacker perpetuating a DoS attack targeting the CoAP server's "`.well-known/core`" URI. The techniques are complementary, and both aim at mitigating the same DoS attack but at different network layers.

We provide the *design* of both techniques. We also detail the evaluation testbed we deployed and the DoS attacker model for the envisioned use case. However, we *implemented* and *evaluated* only the UDP-port

technique in a real IoT platform. We provide the implementation source code and experimental data in a public repository [149].

Contributions of this Chapter

- The definition of two IANVS-based MTD network techniques. (1) A transport-layer technique with UDP port numbers as MP (i.e., port-hopping). (2) An application-layer technique with a CoAP resource's URI as MP.
- The implementation of the UDP port-hopping technique in Pycom LoPy4 IoT nodes as a hardware platform. We share the source code in [149].
- The evaluation of the the UDP port-hopping technique in a real testbed. We defined a probabilistic model that predicts the MTD's effectiveness against the reconnaissance phase of a DoS attack. The empirical results were corroborated.

6.2 DESIGN

6.2.1 *Definition of Common Components*

In order to instantiate IANVS, its four components should be clearly defined. In the MTD literature, the MP is one of the most important qualities that define a particular technique. In IANVS, the MP-Map component determines the MP. In this chapter, we propose two Network-based techniques that differ only in the MP-Map component. The rest of the components are the same and defined as follows:

- *AKE*: A symmetric PSK of 128-bits.
- *Auth-SYNC*: We use periodic MP changes and the NTP[1] protocol to synchronize the RTC of the MTD distributed nodes.
- *CSPRNG*: We use ChaCha20 with 20 rounds. The two inputs are the unmodified 128-bit key from AKE, and a 64-bit nonce derived form the NTP time.

1 NTS for the NTP, or a LATe implementation, must be used in a real deployment.

6.2.2 *Technique I: Single Port-Hopping*

This technique corresponds to the *Network* category of MTD techniques. The MP in this techniques is the UDP port number of a service. TCP and UDP port number pseudo-randomization has been previously proposed in the literature [12, 99, 107], and is known as *port-hopping*. Well-known port numbers are necessary for network services discovering and use. However, the static nature allows for straightforward DoS attacks [103] (e.g., flooding a well-known port). Also, they are the entry point for adversaries in the *reconnaissance* phase of more sophisticated attacks that target higher layers.

MP-Map definition. UDP port numbers range from 0 to 65535 (16-bits). The MP domain cardinality is thus $|MP| \leq |2^{16}|$. We offer a technique for a single-port hopping. Multiple port-hopping poses additional challenges and is discussed later. If the hopping-port is the only UDP port open, it is straightforward to use the 16-bits for port-hopping ($|MP| = |2^{16}|$). Let p be the well-known port number to transform. We apply a bit-wise `xor` with the first 16-bits of the ChaCha20 output *kstr*. This transformation is equivalent to the use of the ChaCha20 stream-cipher to encrypt p. The hopping port p_{mtd} equals $p \oplus kstr_{0..15}$. If other UDP ports are open, standard non-hopping ports may have a port range of 0-32767 (15-bits). Thus, the p_{mtd} should range from 32768 to 65535 ($|MP| = |2^{15}|$). The 16th bit of p_{mtd} should be set to 1.

About Multi-Port Hopping. Multiple MPs in the same domain and codomain require a more complex IANVS MP-Map transformation. The transformation should be invertible, which is the case for the `xor` operation (bijective). However, *security* issues arise depending on the construction chosen. The current proposal, if used for multiple-ports, is prone to a simple well-known attack of stream ciphers: nonce-reuse. If the same *kstr* is used to `xor` different inputs, it becomes a two-time pad. This has security-related consequences[2]. Therefore, a transformation should be used where a nonce-reuse is not that severe (e.g., nonce misuse-resistant).

6.2.3 *Technique II: CoAP `/.well-known/core` URI*

This technique corresponds to the *Network* category of MTD techniques. The MP in this technique is the CoAP [163] "`/.well-known/core`" resource URI [162]. This resource is mandatory to implement for a CoAP Server. If a Client sends a GET request to the `/.well-known/core` URI,

2 The `xor` of the cipher-texts equals the `xor` of the plain-texts.

the Server responds with a payload that contains a set of resources available. For an IoT node with 2 resources, the size of the payload is of ≈50 Bytes. If an attacker wants to perform a simple DoS attack on a remote CoAP Server, sending GET requests to the `/.well-known/core` is one of the most straightforward ways to achieve it. If the node is energy-constrained, the increased use of the network interface will also lead to battery-exhaustion. Our proposal aims at mitigating these type of remote DoS attacks.

MP-Map definition. The CoAP protocol encodes a GET request to the `/.well-known/core` URI as two Uri-Path CoAP Options. They contain an ASCII-encoding of `.well-known` and `core` and have an Option Length of 11 and 4 Bytes, respectively. We propose to `xor` the ASCII-encoding with the *kstr* of ChaCha20. The Server must only respond if the GET request corresponds to the current MTD representation of `/.well-known/core`. The MP moves in a codomain of 15 Bytes ($|MP| = |2^{120}|$). A collision can happen with a non-MTD URI of 11+4 bytes with a probability of $2^{-120} \approx 10^{-36}$. As opposed to UDP port-hopping ($|MP| = |2^{16}|$), this has a low probability of realization. If zero-collision is needed, other measures have to be taken, e.g., avoid combinations of Uri-Paths of 11+4 Bytes length.

Security Considerations. **MTD for a single CoAP resource URI is not a replacement for application-layer security or an authorization framework**. For example, DTLS, OSCORE, or ACE-OAuth should be used to achieve security services such as confidentiality, authentication, or authorization. In general, MTD is not a replacement for information security. However, it is a complementary measure that can improve the system's resilience at a negligible cost. For example, if an IANVS-based MTD technique is hopping the CoAP default UDP port (5683) at the IoT node; with almost no increased cost, it can also be used to apply MTD to the `/.well-known/core` URI. A multi-layer proposal can use the first 2 Bytes of the *kstr* for port-hopping and the following 15 Bytes for the CoAP URI.

6.2.4 *Proposals Summary*

We provided two proposals that illustrate the use of IANVS to instantiate concrete MTD techniques (RQ-3.I1). Both are compatible with legacy non-MTD components in the system, as neither modifies the underlying network protocols they are applied to. The composition of multi-layer MTD techniques was briefly discussed. The incurred incremental cost for additional Network-layers is negligible once IANVS is already in place. Table 6.1 resumes the proposals discussed in this section.

Table 6.1: Proposed MTD techniques using IANVS with ChaCha20.

Network Protocol	Moving Parameter(s)	|MP|	MP-Map
UDP	port number	2^{16}	$p \oplus kstr$
CoAP	`/.well-known/core` URI	2^{120}	$p \oplus kstr$
UDP+CoAP	*(both above)*	2^{136}	$p_1 \| \| p_2 \oplus kstr$

6.3 IMPLEMENTATION AND EVALUATION

In this section, we implement and evaluate the IANVS-based proposals. We share the source code and data in [149]. We use a real IoT Hardware platform in an IP Network. We expose the nodes to a remote attacker performing a DoS attack. We measure the effectiveness of the MTD proposal in terms of the reconnaissance-phase mitigation of the attack.

6.3.1 *System: IoT Hardware Platform*

We use Pycom LoPy4 nodes [105] as a hardware platform. A LoPy4 node has an Espressif ESP32 SoC (32-bit architecture @240 MHz, 520KiB RAM), an RTC, 4MB of external RAM and 8MB of Flash. It has Wi-Fi, Bluetooth, Sigfox, and LoRA (Semtech SX1276) as physical network capabilities. We chose it because a node can run MicroPython code and has many network interfaces; this allows for flexible and fast prototyping. We used the Expansion Board 3.0 to flash the LoPy4 from a UNIX-based PC, and power it through USB.

6.3.2 *Attacker Model*

The attacker is remote. It is physically external to the IoT network but has IP access to it. The attacker knows the IP address of a target IoT device that hosts a CoAP Server over UDP. The attacker's goal is to perform a DoS attack targeted at this IoT node. In order to do so, it floods the target node with CoAP GET `/.well-known/core` messages over UDP.

6.3.3 *Experimental Setup*

The setup is shown in Fig.6.1. The LoPy4 nodes use the Wi-Fi interface. They are one-hop from the wireless access point. The attacker uses

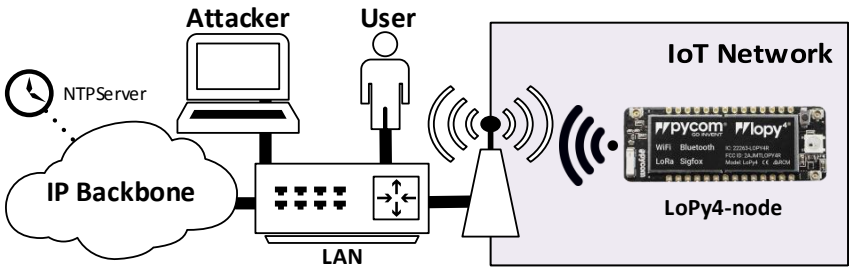


Figure 6.1: Experimental Setup.

a Lenovo ThinkPad-i7 T460 PC running an Ubuntu 19.10 OS. It is connected to the LAN using an Ethernet 100BASE-TX port. The User has the same configuration. For time synchronization, we used an external NTP Server.

6.3.4 *Experiment: UDP Port-Hopping Effectiveness against Reconnaissance-Phase of Attack*

The goal of this experiment is to measure to which degree the UDP port-hopping can mitigate the reconnaissance phase of an attack.

6.3.4.1 *Hypothesis*

The attacker cannot eavesdrop other packets from the LAN. It knows that MTD is applied to the CoAP UDP port, and the range of ports used for hopping. It does not know the PSK nor the period of movement. He knows the NTP Time, but he cannot spoof the NTP Server.

Reconnaissance-phase Success/Fail: If the attacker sends a UDP packet with a given port to the target IoT node (*udp-ping*), it will learn if that port is in use or not (*success* or *fail*).

6.3.4.2 *Probabilistic Model*

We define N as the number of ports used for port-hopping ($N \leq 2^{16}$). The actual port in use is uniformly chosen over N. Reconnaissance success for the attacker after a single *udp-ping* over N possible ports can be modeled as a random variable (r.v.) that follows a Bernoulli distribution with probability $p = \frac{1}{N}$. Over a single MTD period, the attacker can perform n number of *udp-pings*. The attacker cannot discard previously tested ports because it does not know when the port changes. Thus, the *udp-pings* are independent and identically distributed. Then, the number of reconnaissance successes over a

Table 6.2: UDP port-hopping experiment parameters.

N (#ports)	MTD period P (seconds)
512	{5,70,177,365,589}
1024	{10, 140, 355, 730, 1179 }
2048	{21, 281, 710, 1461, 2357 }

Attacker Speed = 2 attacks/s

single MTD period follows a Binomial distribution[3] $B(n, p)$, with n number of trials, and p probability of success of a single trial.

6.3.4.3 *Implementation and Execution*

We implemented port-hopping for LoPy4 nodes as specified in Sec. 6.2.2. We used `microCoAPy`[4] library and Joachim Strömbergson's `chacha.py`[5]. The attacker code uses the `hping3` packet generator tool. It randomizes the chosen port using the bash function `$RANDOM` (15-bits); if needed, applies a modulo N operation to restrict the result to the port-hopping range.

The LoPy4 logs internally if the used port was found for a given MTD period. In this study, we focus on the probability of the port not being found at all. We tested several combinations of port-hopping range N, and MTD period lengths P. The attacker's *udp-ping* period is fixed to `500ms` (2 att./s). The parameters for the experiments are in Table 6.2. In our experiments $n = P \times 2$ att./s.

For each tuple (N, P) of experiment parameters, we ran between 120 and 600 periods (samples). The total net run-time of the experiments is around 480 hours or 20 days. As an example, 120 runs of the ($N = 2048$, $P = 2357$) experiment have a net run-time $\approx$ 78 hours.

6.3.4.4 *Results*

In Fig. 6.2, we can see the results. For each tuple (15 in total), we calculated the empirical probability (i.e., the relative frequency) of a sample with zero successful attacks. To measure the uncertainty of this value, we partition the tuple-experiment sample set in 5+ equally-sized subsets. We calculated the standard deviation of the empirical probability of zero successful attacks from each subset. The theoretical

3 $\sum_{i=1}^{n} Bernoulli_i(p) \sim Binomial(n, p)$, with n = number of trials.
4 https://github.com/insighio/microCoAPy
5 https://github.com/secworks/chacha/

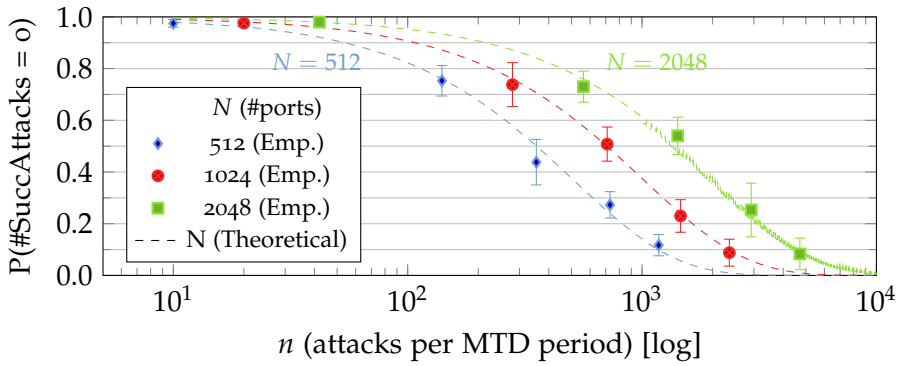


Figure 6.2: Port-Hopping: Empirical probability of zero successful attacks over one MTD period as a function of attacks per period, for different #ports N.

probability $P(\#SuccAttacks = 0)$ from the corresponding Binomial distributions $B(n, p = 1/N)$ is also plotted.

6.3.4.5 *Analysis*

As expected, the empirical data fits the probabilistic model. The parameters that determine the underlying probability are N and n. N can be controlled directly by the system; in this port-hopping technique, it should be maximized as there is no additional cost for the system. The parameter n depends on the attacker-defender relationship between P and attacker speed. Adjusting P to arbitrarily small values will incur increased costs for the system (e.g., offline time, fine-grained synchronization); it should be determined according to the particularities of the real system and use case.

6.4 CONCLUSION

In this chapter, we presented two IANVS-based MTD techniques suitable for IoT systems. We implemented IANVS in software and evaluated one of the MTD techniques in real IoT hardware. The results show that MTD-based techniques can effectively mitigate otherwise trivial attacks. We shared the source code and results publicly.

Regarding this memoir's main RQs, this chapter provided two instances of MTDs in concrete IoT use cases (RQ-3.I1), and validated the IANVS framework as a fundamental tool to help in their *design* and *implementation*[6]. It is not possible to give a definitive/exhaustive answer to RQ-3.I1. However, we illustrated how to leverage elements defined

6 E.g., by reusing already implemented components.

in Ch. 4 to make the instantiation of concrete MTDs a less-challenging task. I.e., by not starting from the MTD fundamentals but focusing on practical aspects of a use case. Some future work perspectives were hinted, like multiple MPs in the same domain, but we will treat them in the last chapter of this thesis.

In the following chapter, we present our final major contribution: an IANVS-based MTD technique targeting the physical layer of wireless systems.

Research Questions

RQs answered:

- **RQ-3.I1**: How to instantiate MTDs in concrete IoT use cases?
  - [IANVS-based] We instantiated a transport-layer MTD technique with UDP port numbers as MP (*Design*, hardware *Implementation*, and theoretical and hardware *Evaluation*).
  - [IANVS-based] We instantiated an application-layer MTD technique with CoAP ".well-known/core" resource's URI as MP (*Design*).

7

A PHY-LAYER MTD: PHYSICAL-LAYER RESILIENCE TO INSIDER ATTACKS IN IOT NETWORKS

7.1 INTRODUCTION

In this chapter, we present a Network-physical-layer MTD technique that instantiates the IANVS framework (Ch. 4.3). In conjunction with the techniques from the previous chapter, they constitute our contributions motivated by the question of "How to instantiate MTDs in concrete IoT use cases?" (RQ-3.I1).

Previously, we stated the importance of the Network component in IoT systems (Ch. 2.3), reflected on the results of our SLR (Ch. 3) and this memoir's focus on Network MTD techniques (Ch. 4-6). This chapter is not an exception, and is motivated by a problematic common to any WCS.

Wireless communication is a key technology for IoT systems. Due to its open nature, the physical layer of WCSs is a high-priority target for an adversary whose goal is to disrupt the system's normal behavior. In particular, jamming attacks are one of the most straightforward and effective types of attacks: by emitting a signal over the channel, information flow of the system is stopped or severely disturbed.

In this final contribution, we propose a IANVS-based AJ technique to improve the jamming resilience of IoT systems leveraging on the Direct-Sequence Spread Spectrum (DSSS) radio-modulation technique. The sequences of the proposed WCS are generated in an ad-hoc, independent, and distributed way. We show probabilistically that the generated sequences have *robust*[1] Cross-Correlation (CC) properties. We define a multi-user system model to evaluate the Bit Error Rate (BER) of our proposal in the presence of jammers. We define two types of jammers, a classical band-limited Gaussian noise jammer and an insider smart jammer with knowledge of one Spreading Sequence (SS) used in the system (i.e. an *insider-node attack*).

1 These properties will determine that the system is *usable* and *secure*.

Contributions of this Chapter

- We propose an IoT-oriented AJ Network-physical-layer MTD technique that uses CSPRNGs for the randomization of DSSS spreading sequences.
- We study the CC statistical and probabilistic properties of large CSPR sequence sets and uniformly random sequence sets. This fundamental study is lacking in the WCS literature.
- We evaluate the jamming resilience of our proposal using a model implemented in MATLAB. We expose our system to an insider smart jammer and validate that the attack is mitigated. Insider-smart-jammer AJ resilience and CC of sequences are analytically linked.

7.2 MOTIVATION

WIRELESS PHYSICAL LAYER Although a secure system involves security mechanisms at many of its components, the Network layer is fundamental. More precisely, the *network physical* (PHY) layer is arguably the most important resource of an IoT system to be protected. First, the PHY layer is the enabler of the distributed capabilities of IoT systems; upper-layer services rely on it. Second, most IoT networks are wireless, and the open nature of this transmission medium makes the PHY layer an easily-accessible *target resource* for an attacker. Among the existing PHY layer attacks, *jamming* is one of the most basic and effective.

JAMMING A *jammer* introduces a signal into a shared medium to disturb legit communication between nodes in the system. The consequence of a successful jamming attack is that the information flow of the system is disrupted. Even more, as a consequence of Shannon's limit on any communication channel [161], an attacker with enough power will *always* be successful in jamming a target system with limited power (e.g. a constrained IoT node). Therefore, using resources into a jamming attack is an effective strategy from an attacker's point of view [171]. Correspondingly, AJ defense mechanisms should be a priority from a system's perspective.

SPREAD-SPECTRUM TECHNIQUES Spread-spectrum techniques are well-known for their AJ capabilities [50, 167]. Two prominent spread-

spectrum techniques are Frequency-Hopping Spread Spectrum (FHSS), and DSSS. Both techniques rely on a pre-shared sequence between transmitter and receiver to (de)spread the signal in the time-frequency domains. A jammer without the knowledge of the pre-shared sequence cannot power-efficiently jam the transmission. State-of-the-art IoT radio uses spread-spectrum techniques. IEEE 802.15.4 uses DSSS, and defines a Time Slotted Channel Hopping (TSCH) mode based on FHSS [72]. Long-range LoRa modulation uses patented Chirp Spread-Spectrum (CSS) [88].

ROBUST IOT AJ IS LACKING However, none of the precious mentioned spread-spectrum systems were designed with AJ as a primary objective. On the one hand, LoRa uses well-known spreading-parameters to do CSS, allowing for trivial jamming [11]. On the other hand, 802.15.4 in DSSS mode uses spreading-sequences not only fixed but also too short for providing any AJ guarantee. Even if 802.15.4-TSCH provides an FHSS framework which allows a system to use a custom or standardized *hopping schedule*, most of 802.15.4-TSCH networks in the literature can be jammed [32]. There is a lack of IoT systems designed with AJ as one of their primary objectives. Furthermore, insider-node jamming attacks are a real threat to heterogeneous-IoT systems. A malicious insider-node has knowledge of the public network parameters of the system. Thus, it can efficiently jam nodes that share the same AJ parameters.

PROPOSAL DESIGN RATIONALE In this chapter, we propose a novel IoT-suitable AJ mechanism that uses the IANVS MTD framework as a base design, and that leverages on spread-spectrum techniques. By design, our proposal proactively mitigates insider-node jamming attacks. Our proposal randomizes the spreading-sequences used by the nodes in a DSSS system. Every pair of communicating nodes will have a unique pairwise spreading-sequence, only known by them. The novelty of our proposal relies on two factors. First, the spreading-sequences are generated using CSPRNGs; thus, cryptographically strong randomness claims of the generated sequences are assured. Second, the generation process is done following a decentralized and independent process[2].

7.3 BACKGROUND

In this section, we describe three subjects needed to develop this chapter's contribution: CC of Sequences, PR Sequence Sets, and CSPRNGs.

2 A necessary condition to provide insider-jammer resilience

7.3.1 *Cross-Correlation of Sequences*

7.3.1.1 *Definitions*

The correlation is a measure of the linear similarity between two sequences. If both sequences are identical, the term auto-correlation is used. Otherwise, the term CC is used.

The CC is used mostly on the signal processing field, and we distinguish it from the generic notion of *correlation* used on statistics (e.g. the Pearson's correlation coefficient). Generally speaking, the CC is a mathematical operation on two *functions*. In this chapter, we deal with series/sequences and the notion of *discrete* CC is relevant. The discrete CC for the sequences s_1 and s_2, written as $s_1 \star s_2$, is defined as:

$$(s_1 \star s_2)[n] \stackrel{\text{def}}{=} \sum_{m=-\infty}^{\infty} \overline{s_1[m]} s_2[m+n]$$

Furthermore, the discrete *circular* CC is relevant for periodic sequences of period L. The circular CC between two periodic sequences s_1 and s_2, written as $s_1 \circledast s_2$, is defined as:

$$(s_1 \circledast s_2)[n] \stackrel{\text{def}}{=} \sum_{m=0}^{L-1} \overline{s_1[m]} s_2[m+n] \tag{7.1}$$

Where $\overline{s_1[m]}$ is the complex conjugate of $s_1[m]$ and n the displacement. Normalized values of the circular CC are obtained if the result is divided by the maximum auto-correlation value.

In this chapter, we study sequences of length L, that are used in a DSSS system as periodic sequences of period L. Therefore, the circular CC concept is extensively used and will be simply referred as CC.

7.3.1.2 *Fast Cross-Correlation calculation*

The CC of $s_1[n]$ and $s_2[n]$, written $s_1[n] \star s_2[n]$, is equivalent to the convolution of $\overline{s_1[-n]}$ and $s_2[n]$, where $\bar{s}$ corresponds to the complex conjugate of s. This equality allows to use the *convolution theorem* to obtain:

$$(s_1 \star s_2) = \mathcal{F}^{-1}\left\{ \overline{\mathcal{F}\{s_1\}} \cdot \mathcal{F}\{s_2\} \right\} \tag{7.2}$$

Where $\cdot$ denotes point-wise multiplication, $\mathcal{F}$ stands for the Fourier transform, and $\mathcal{F}^{-1}$ for the inverse Fourier transform. This equivalence in conjunction with the *fast Fourier transform* allows for efficient computation of CC values in current computing hardware.

7.3.1.3 *Practical importance for WCSs*

Correlation properties of the sequences have a direct impact on many fundamental properties and performance metrics of WCSs. For example, low CC is desirable for CDMA systems, and low auto-correlation is desirable for signal acquisition and multi-path interference rejection. Thus, the correlation of sequence sets for WCSs is a widely studied subject [60, 184]. In general, families of sequence sets for WCSs are designed with low correlation properties [58, 154]. One well-known example is orthogonal sequence sets: the sequences in the set have a CC of zero.

The literature on CC of sequence sets for wireless communication [151, 169, 184, 201] characterizes a given family of sequence sets by the maximum CC value of all the pairs of sequences within a set ϕ_{max}. In most cases, due to the impossibility of computing exact-values, lower bounds for ϕ_{max} are given. Some well-known bounds are Sidelnikov's [166], and Welch's [193]. For a given pair of sequences (x, y), a useful CC concept is the *cross-correlation spectra* $\phi_{x,y}$. The $\phi_{x,y}$ measures the CC values evaluated for every possible shift of one of the sequences. The set size is also a fundamental characteristic of a family of sequence sets besides the CC. Ideally, for multi-user WCSs we want low ϕ_{max} and large sets. Nevertheless, there is generally a trade-off ϕ_{max} vs. set size: a low ϕ_{max} value implies a small set size.

7.3.2 *Pseudo-Random Sequence Sets for WCSs*

PR sequence sets for WCSs is a well-studied topic in the literature [184, 201]. A PR sequence complies with some randomness criteria. Golomb's Randomness postulates [59] are widely accepted criteria in the WCS literature.

Several families of PR sequence sets exist. The Feedback Shift Register (FSR)-based is the most prominent family of PR sequence sets. An FSR is a hardware component that consists of a chain of flip-flops sharing the same clock. The output of one flip-flop is also connected to the input of the next one. A Linear-Feedback Shift Register (LFSR) is an FSR in which the input to the first flip-flop is a linear function of the

previous FSR state. For a Non-Linear-Feedback Shift Register (NLFSR), the input is a non-linear function of the previous FSR state.

An LFSR uses simple hardware components and produces uniformly distributed sequences with high-throughput. As a result, LFSR-families of sequence sets are historically the most used and studied [59]. However, an LFSR has weaknesses in terms of cryptanalysis due to its linear nature. For example, remaining portions of a sequence can be predicted with partial knowledge of its elements using the Berlekamp-Massey algorithm [111]. Notably, this predictability is not a desirable property from an AJ perspective. If a jammer knows the sequence of a transceiver, it can jam it in a power-efficient way [8].

Other families of PR sequence sets for WCSs have gained attention in recent years due to the predictability of LFSR-based sequences. These families include Legendre/Jacobi sequences [31, 38], NLFSR-based De Bruijn sequences [169, 170, 191], and chaotic sequences [90, 112, 181]. However, these families of sequence sets for WCSs still have factors that impact on their randomness properties. Either because of non-randomness-related design objectives or because of inherent functional limitations. Jacobi sequences aim at low auto-correlation properties by design. De Bruijn sequences in a set are not independent of each other, and the sets are generated with low CC design objectives. In other words, non-negligible information can be known of other De Bruijn sequences in the set if one sequence is known. Chaos theory-based sequences have practical-use design challenges [181]. For example, chaotic sequences are inherently non-periodic, and this forces either robust-synchronization of the chaotic system state, or complex non-coherent methods for demodulation. Besides, their use for cryptography use is proven to be immature and broken [3, 102], which implies their randomness properties are compromised.

7.3.3 *CSPRNGs and Stream Ciphers*

A CSPRNG is a functional block that takes some input parameters (i.e., a *key* and a *nonce*) and produces PR output suitable for cryptographic use [14, 16]. The same input parameters will consistently produce the same output.

Stream ciphers [147] are symmetric ciphers. With a given *key*, a stream cipher generates a CSPR stream of bits called a *keystream*. The keystream is independent of the message to be encrypted. The Cha-Cha20 [21] is a stream cipher designed to be fast on pure-software implementations. It is used as a state-of-the-art cipher in Internet security protocols such as IKE-IPsec [125] and TLS [98]. Cha-Cha20 uses a 256-bit *key* and a 64-bit *nonce* as inputs. It can output 2^{41} bits

(a) IoT AJ MTD network: every pair of nodes use unique PHY parameters.

(b) Insider attack: node *d* compromised. Jamming resilient PHY layer links.

Figure 7.1: IoT MTD Network

(274 GBytes) of PR data (keystream). In this work, we use Cha-Cha20 as a CSPRNG. First, because in terms of security, it is a well-established stream cipher. Second, because it is software-optimized [39], and this offers great flexibility for the dynamic IoT systems we target.

7.4 PROPOSAL

In this section, we present our IANVS-based MTD mechanism targeted at the Network physical layer of an IoT system. The main objective of our proposal is to improve the jamming resilience in WCSs. In particular, we want to proactively mitigate the impact on the system of insider-node jamming attacks.

7.4.1 *Overview-Rationale*

An example of a target IoT network is illustrated in Fig. 7.1a: circles represent IoT nodes, and edges are communication links. Every link between a pair of communicating nodes in the IoT network has distinct physical layer parameters defined by our MTD proposal. These pairwise parameters are only known by each pair. This physical link diversity is represented with different edge colors in Fig. 7.1a.

To illustrate the potential AJ advantages of such a system, consider an *insider attack*. In this kind of attack, one of the legitimate nodes in a network becomes an adversary. Fig.7.1b shows node *d*, as an insider attacker in an IoT network. In a *legacy* IoT network, all the nodes share the physical layer parameters; therefore, an insider attacker can potentially become a very power-efficient jammer. In our proposed MTD system, an insider attacker's jamming impact is mitigated because it has no perfect knowledge of the physical layer parameters for every link in the network. As stated before, *every link* between a pair of nodes has unique physical layer parameters known only by each pair.

7.4.2 *IANVS-based MTD: CSPR Sequences for DSSS*

This proposal is based on IANVS (Ch. 4.3). Its components are defined as follows:

- *AKE*: We assume a pairwise symmetric PSK of 256-bits, unique to each pair of communicating nodes.
- *Auth-SYNC*: We assume a system-wide authenticated synchronized System State S_t of 64-bits, with no randomness requisites.
- *CSPRNG*: We use ChaCha20 and AES-CTR. The inputs to the CSPRNGs are the 256-bit key from AKE and the 64-bit S_t.
- *MP-Map*: We truncate the CSPRNG output (keystream) to L bits. The result is used as a periodic SS, ss_{mtd}, for DSSS modulation between the pair of communicating nodes.

The rest of this contribution focus on the properties of the CSPRNG and MP-Map components output. The CC properties of the produced SSs (i.e. the MP) and the AJ mitigation in a modeled WSN system facing an insider-node jammer.

7.4.3 *Implications of PHY Randomization*

The Cryptographically Secure Pseudo-Randomization of DSSS spreading sequences effectively mitigates insider jamming attacks, as will be evaluated in Sec. 7.6. This insider AJ resilience comes with a trade-off in terms of multi-user performance. The CC values of the spreading-sequences in a DSSS system determine the multi-user performance, the lower, the better. Because our proposal randomizes the sequences in a decentralized and independent way, there are no low-cross-correlation guarantees for the system.

However, we can statistically study the CC values of large CSPR sequence sets. Furthermore, because of the good randomness properties of CSPRNGs, any given CSPR sequence set can be probabilistically characterized. The CC of the CSPR sequences is not only the determining factor of multi-user performance but also of the insider AJ mitigation. This CC statistical study is presented in the following section. .

7.5 CROSS-CORRELATION OF CSPR SEQUENCE SETS

In this section, we study the statistical distribution of CC values of large CSPR sequence sets. We prioritize an empirical approach. First, we generate large CSPR sequence sets. Second, we calculate the CC of all pairwise sequence combinations in the set. Finally, we calculate the ECDF of the CC values. We also provide analytical results that validate the empirical study. We conclude this section with a CC comparative study with other families of PR sequence sets.

7.5.1 *Motivation*

For a given family of sequence sets for WCSs, the CC and the Cardinality (i.e., number of elements) of the sets are two of the most important characteristics. They determine the multi-user capabilities of the system. As stated in Sec. 7.4.3, only probabilistic statements can be made about the cross-correlation values of CSPR sequence sets. To the best of our knowledge, no work in the literature studies this problem with enough depth. This section deals with this fundamental study. These results are used in Sec. 7.6 to characterize the AJ capabilities of our proposal analytically.

7.5.2 *Sequence Sets Generation*

Let $S_{(L,CSPRNG)}$ be a generated sequence set. The sequences in the set are binary sequences of a fixed length L, and were generated using the same CSPRNG. The characteristics and generation-input parameters of the sets are the following:

- Cardinality (set size): 1024
- L (bits): {128, 256, 512, 768, 1024, 2048, 4096, 8192, 16384, 32768}
- CSPRNG: {ChaCha20 [21], AES-CTR[3]}
- CSPRNG Inputs:
  - Key: $\{(0)_{10}, (1)_{10}, (2)_{10}, ..., (1023)_{10}\}$ (256-bit length and zero-padded)
  - System State: {0} (64-bit)

3 AES is a block cipher, but in CTR mode behaves as a stream cipher.

For example, the set $S_{(128,ChaCha20)}$ is composed of 1024 sequences $\{s_0, s_1, ..., s_{1023}\}$ of length 128. A sequence s_i ($i \in \{0, 1, ..., 1023\}$) is generated using ChaCha20 as CSPRNG. And its inputs are: *Key* = $(i)_{10}$, and *System State* = 0 ; the output is truncated to *128* bits.

A generated binary sequence $\{b_1, ..., b_L\}$ has unipolar encoding with elements $b_i \in \{1, 0\}$, from now on we will work with sequences in bipolar encoding where $b_i \in \{1, -1\}$. Also, a given sequence of length L will be used in our communication system as a periodic sequence with period L; thus, length and period are equivalent terms in the rest of the section.

7.5.3 *Normalized Cross-Correlation Calculation*

The circular CC (Eq. 7.1) between two bipolar binary sequences of length L, evaluated at a displacement n, takes integer values comprised in $\{-L, ..., 0, ..., L\}$. In order to compare CC properties of sequences of different length L, the Normalized CC is useful. It is obtained dividing the CC value by the maximum possible value, in our case L. Furthermore, we take the absolute value of this result, as in terms of sequence-signal interference, the sign of the CC value is irrelevant. The expected Normalized CC values will be comprised in $\{0, \frac{1}{L}, \frac{2}{L}, ..., 1\}$, where the value of 0 is associated with an orthogonal sequence, and 1 with the max value, i.e., the same sequence.

Let $|NCC_{(x,y)[n]}|$ be the absolute value of the Normalized circular Cross-Correlation (NCC) . For a given pair of sequences (x, y) of length L, and evaluated at a fixed displacement (time shift) $n \in \{0, 1, ..., L-1\}$, this is defined as:

$$|NCC_{(x,y)[n]}| = \left| \frac{1}{L} \sum_{m=0}^{L-1} \overline{x[m]} y[m+n] \right| \tag{7.3}$$

The NCC is a scalar value. For every generated set $S_{(L,\text{CSPRNG})}$, we calculate the NCC for every pair of sequences (x, y) in the set, and for every displacement $n \in \{0, ..., L-1\}$. Let $CC\_S_{(L,CSPRNG)}$ be the set that contains all the calculated NCCs. This new set contains at least 50 million NCC elements.

To calculate the NCC values (Eq. 7.3), we use the convolution theorem (Eq. 7.2) in conjunction with fast Fourier transforms using an Intel Core i7-6600U CPU @ 2.60GHz x 4 with 16 GB RAM. For sets of large

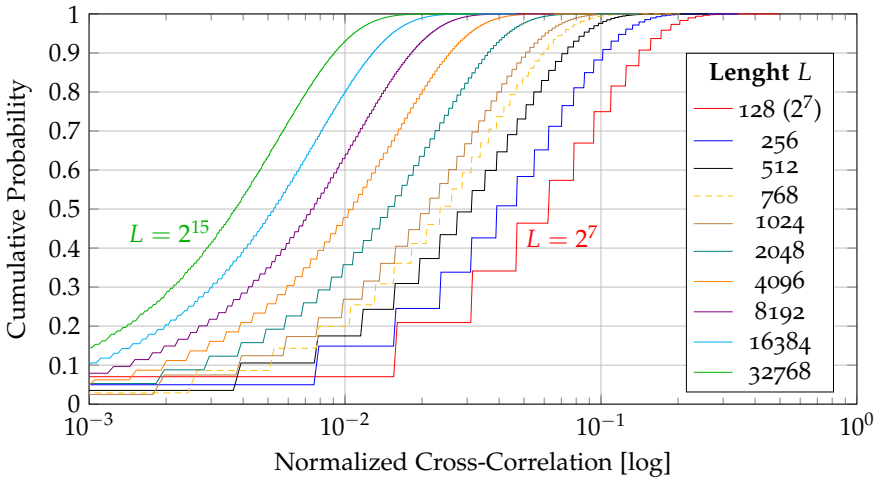


Figure 7.2: ECDF of Normalized CC of ChaCha20-generated sequence sets, for different sequence length L.

sequence length L we lowered the cardinality, but this does not affect the statistical relevance of the results[4].

7.5.4 *Statistical Results and Probability Analysis*

In Fig. 7.2 we show the ECDFs of NCC values of ChaCha20 generated sets $S_{L,ChaCha20}$. Every ECDF was calculated with between 50 and 500 million NCC values. The ECDFs of AES-CTR generated sets are visually indistinguishable from the ChaCha20. The statistical similarity is expected, as by design a CSPRNG is indistinguishable from a *True RNG*[5]; and, by the Glivenko-Cantelli Theorem [182], both ECDFs converge to the same Cumulative Distribution Function (CDF).

It can be observed in Fig. 7.2, that the greater the length L of the sequences in the set S_L, the lower the NCC values for almost all the percentiles (i.e., x-value for a given cumulative probability) of the ECDFs. This was expected, informally the longer the pseudo-random sequences, the lower the probability of two sequences being similar to each other. For example, the 80th percentile $P_{80} = k$ (i.e., 80% of the NCC values are $\leq k$) is for S_{128}, $k \approx 0.12$; for S_{256}, $k \approx 0.08$; and for S_{512}, $k \approx 0.06$. This is not true for every percentile. For some percentiles lesser than P_{15}, we can see that the ECDFs intersect each other. The step-nature of the ECDFs explains these counter-intuitive results. It is worth noting that this step-nature of the ECDFs is not due

4 Glivenko-Cantelli's Theorem [182] states that the ECDF of a random variable converges uniformly to the Cumulative Distribution Function (CDF) of the underlying-unknown distribution.

5 A generator of truly uniformly distributed bit string (i.e., a *Bernoulli process* with $p = 0.5$), or in cryptographic terms a *random oracle*.

to a limited number of sample variables NCC (at least 50 million), but to the inherent discrete values the NCC takes for binary sequences of a fixed length L.

Ultimately, we want to predict the NCC distribution of *any* given CSPR sequence set. The randomness properties of CSPRNGs and the Glivenko-Cantelli Theorem [182] give us strong statistical guarantees that any random set of CSPR sequences will follow the ECDFs shown in Fig. 7.2. Furthermore, we strengthen this statement with a pure analytical-probability approach that is found in the next Section 7.5.5. The single hypothesis is that a CSPRNG output resembles a true uniformly random process where every produced bit has an equal probability of being 1 or -1. The results validate the statistical analysis. For $L = 128$ the ChaCha20 ECDF corresponds to the Analytical CDFs with a point-wise precision of ± 0.0001. An important analytical result is that a given percentile P_n is a function of $\sqrt{L}$. This result will have practical AJ system design consequences, as will be explored in Sec. 7.6.

7.5.5 *Analytical CC Distribution of CSPR Sequences*

A CSPRNG of length L must be statistically indistinguishable from a Bernoulli Process of L trials. Using this equivalence, we develop an analytical probabilistic study of the CC of CSPR sequences.

Let $s = \{b_1, b_2, ..., b_L\}$ be a CSPR binary sequence of length L, where every bit b_i is a random variable (r.v.) that follows a Bernoulli distribution with $p = 0.5$.

Let $s_1 = \{x_1, ..., x_L\}$ and $s_2 = \{y_1, ..., y_L\}$ be two independent CSPR binary sequences, with L even. The circular CC of s_1 and s_2, evaluated at $n = 0$, is:

$$(s_1 \circledast s_2)[0] \stackrel{\text{def}}{=} \sum_{m=1}^{L} \overline{s_1[m]} s_2[m+0] = \sum_{m=1}^{L} x_m y_m = \sum_{m=1}^{L} b_m \tag{7.4}$$

Where b_m is a r.v. that also follows a Bernoulli distribution[6] with p=0.5. The probability distribution of the sum of two or more independent r.v. is equivalent to the convolution of their individual distributions. Particularly, the sum of two Bernoulli r.v. results in a r.v. with Binomial Distribution of 2 trials. It is well known that $\sum_{n=1}^{L} Bernoulli(p) \sim B(n, p)$, where $B(n, p)$ is a Binomial where n is the number of trials. With this result, we develop Eq. 7.4. The resulting random variable follows a Binomial distribution $B(L, 0.5)$. However, this is true only if the original domain-support of the Bernoulli distribution where $k \in \{0, 1\}$,

6 Multiplication of two independent Bernoulli variables is also a Bernoulli.

but in our signal processing setting, we use binary bipolar values $k \in \{-1, 1\}$. We apply a change of variable (c.v.) to transform the result of the studied r.v. $X \sim B(L, 0.5)$ with support $x \in \{0, 1, 2, 3, \ldots, L\}$, to our signal processing setting support where the CC result will take only even values $y \in \{-L, \ldots, -4, -2, 0, 2, 4, \ldots, L\}$. The c.v. is $y = 2(x - \frac{L}{2}) \longrightarrow x = \frac{y+L}{2}$. The resulting r.v. represents the CC value of two CSPR sequences, and its Probability Mass Function (PMF) is symmetrical with respect to zero. We need one more transformation because we are interested in the absolute value of the CC, $|CC|$. The $|CC|$ support will be $k \in \{0, 2, 4, \ldots, L\}$ with k even. Because of the symmetry of the CC r.v. PMF, the $|CC|$ r.v. PMF is straightforward; we double the probability of all the positive values, except for zero. More precisely, the PMF of the absolute cross-correlation $|CC|$ of two CSPR sequences of even length L is, as a function of the taken value $k \in \mathbb{N}_0$:

$$\mathbb{P}(|CC| = k) = \begin{cases} \dfrac{L!}{(L/2)!(L/2)!} \left(\dfrac{1}{2}\right)^L & \text{if } k = 0, \\ \dfrac{L!}{\left(\frac{L+k}{2}\right)! \left(\frac{L-k}{2}\right)!} \left(\dfrac{1}{2}\right)^L \times 2 & \text{if } 0 < k \leq L, \, k \text{ even}, \\ 0 & \text{otherwise.} \end{cases}$$

The CDF can be either calculated directly or expressed in terms of the regularized incomplete beta function. However, we take another approach to further describe the probabilistic properties of the $|CC|$ of CSPR sequences.

Using the De Moivre-Laplace theorem, we can approximate the Binomial distribution $X \sim B(L, 0.5)$ with a Normal distribution $N(\mu, \sigma)$ of mean $\mu = L/2$, and standard deviation $\sigma = \sqrt{L/4}$. This results in $X \sim B(L, 0.5) \sim N(\frac{L}{2}, \sqrt{L/4})$. We apply the same c.v. as in the discrete case to transform the result to our CC r.v. domain, a horizontal shift of $-L/2$, and a horizontal dilation by a factor of 2. This results in $CC \sim N(0, \sqrt{L})$. This continuous approximation takes into account the odd values of the horizontal axis. Because our domain support $k \in \{0, 2, 4, \ldots, L\}$ only has even values, we need to multiply $\times 2$ the approximated probability, excepting for $k = 0$. A last transformation is needed to obtain the absolute value and represent the $|CC|$ r.v., the result closely resembles a half-normal distribution. Finally, the *approximation* of the PMF of $|CC|$ is:

$$\mathbb{P}(|CC| = k) \approx \begin{cases} \frac{\sqrt{2/\pi}}{\sqrt{L}} & \text{if } k = 0, \\ \frac{\sqrt{2/\pi}}{\sqrt{L}} e^{-\frac{k^2}{2L}} \times 2 & \text{if } 0 < k \leq L, \text{ and } k \text{ even}, \\ 0 & \text{otherwise.} \end{cases}$$

We validated using Wolfram Mathematica sofware that the approximation is correct at least with 3 significant digits for $L \geq 128$. Some important characteristics of the $|CC|$ distribution of CSPR sequences of length L are shown in Table 7.1.

Table 7.1: Properties of the $|CC|$ of CSPR sequences of length L

Support	$k \in \{0, 2, 4, \ldots, L\}$
Mean	$\approx \sqrt{L}\sqrt{2/\pi}$
Variance	$\approx L(1 - 2/\pi)$
Median	$= \lfloor\sqrt{L/2}\rfloor$ or $\lceil\sqrt{L/2}\rceil$ (the even value)
Mode	$= 2$
CDF $F(k)$	$\approx \text{erf}\left(\frac{k}{\sqrt{L}\sqrt{2}}\right)$

Finally, for having results that correspond to the Normalized $|CC|$ ($|NCC|$), with support $k' \in \{0, \frac{2}{L}, \frac{4}{L}, .., 1\}$, the change of variable $k = k'L$ should be done.

7.5.6 *Comparison with NCC of other PR families*

In this section, we compare the obtained results against other families of PR sequence sets. The literature generally characterizes a given family with the maximum value of the NCC of all the sequences in a set, NCC_{max}.

The NCC_{max} expresses the worse-case value of the NCC of all pairwise sequence combinations in a set, for every relative sequence displacement. All other pairwise combinations have lower NCC values. When realized, the NCC_{max} affects two pairs of communicating nodes, not the whole system. An NCC_{max} will be realized when the specific pair(s) of nodes communicate at the same time, and for a specific relative sequence time-shift (displacement).

For PR families in the bibliography, there is hard-bounds by design for the NCC_{max}. For the CSPR sequence sets we propose in this chapter, we have no hard-bounds (i.e., any value is possible, albeit with different probability), but a probabilistic estimation can be given. In Table

Table 7.2: **NCC$_{max}$** for different families of PR sequence sets

PR Family	NCC_{max}	Set Size	Seq. Length L
Gold	0.130	257	255
Kasami (large set)	0.130	4 112	255
Kasami (small set)	0.067	16	255
De Bruijn (low-CC [169])	0.130	16	256
CSPR (C.I. 95%)	0.305	16	256

7.2, we show NCC_{max} values of CSPR[7] sequence sets compared with other PR families. For CSPR-ChaCha20 sequences of length $L = 256$, the NCC mean value is ≈ 0.050[8] and the median equals 0.044194174.

The NCC_{max} for CSPR sequences with a Confidence Interval (C.I.) 95% is more than double compared with other families. This higher NCC is undesirable for constant high-throughput systems with a centralized-star topology (i.e., cellular networks, where a central node communicates with all others). For a constrained IoT use case, where traffic is packet-based and sporadic, the impact of the theoretical NCC_{max} on the system performance is not that relevant because: (1) the probability of realization of the event is low (0.00088796%[9]), and (2) if it happens, the impact on the system performance will be over a single packet. Thus, arguably, the statistical distribution of CC of sequences (i.e. the statistical distribution of the CC spectra) is a better suited tool than the single-value NCC_{max} (i.e., a low-probability and short-lived event) to predict the expected (e.g., mean) performance of packet-based low-throughput systems like the constrained IoT.

7.6 EVALUATION: AJ RESILIENCE OF PROPOSAL

In this section, we present numerical results that measure the AJ resilience of our proposal. System AJ resilience is measured in terms of BER as a function of jammer signal power. We use MATLAB to simulate a multi-node DSSS system. First, we present the system and attacker model. Then, we evaluate the system AJ resilience against two types of attackers (jammers). (A) A Broadband Noise Jammer that represents

7 $P_{99.999926}(NCC) = 0.305$, combined probability for 256 values of the spectra -taken as, *i.i.d* variables- and for a set of 16 sequences $(0.99999926)^{256\times\binom{16}{2}} = 0.9555$.

8 Standard deviation $= 0.449$

9 For a given instant in time, with the 16 sequences being used at the same time, $P(NCC_{\texttt{at least 1 pair}} \geq NCC_{max}) \leq 1 - (0.99999926)^{120}$

a baseline for jammer power efficiency, i.e., any other DSSS jamming strategy will be preferable for the attacker. (B) An *Insider Smart Jammer* that represents an upper-bound for jammer power efficiency. The latter jamming scenario is the most relevant. It instantiates our insider-node jamming attack hypothesis and measures the degree of jamming mitigation of our proposed CSPRNG-based DSSS system. Finally, the Insider Smart Jammer AJ resilience is analytically linked to the CC properties studied in Section 7.5.

7.6.1 *System Model*

The system model is shown in Fig. 7.3. The system is composed of n Transmitter (TX)-nodes, a jammer, one Receiver (RX)-node, and a channel modeled as Additive White Gaussian Noise (AWGN). The Binary Phase-Shift Keying (BPSK) modulation in our setting is the conversion from unipolar {1,0} to bipolar {1,-1} of a signal sample. The communication process is as follows:

(1) A TX-node sends a random digital signal modulated with BPSK.

(2) A binary SS is applied after BPSK, and the transmission is in base-band.

(3) Other signals and AWGN are added.

(4-6) The RX-node demodulates the received signal by:

  (4) de-spreading it using a synchronized version of the SS,

  (5) applying an *integrate-and-dump* correlator, and

  (6) making a decision based on the sign of the signal to determine the BPSK symbol.

ABOUT THE AWGN CHANNEL MODEL The channel model does not account for fading, frequency selectivity, multi-path, nonlinearity, or dispersion. This simplification is not in demerit of the jammer. An AWGN channel highlights the relationship between the jammer and the nodes' signal power in the demodulation process. Most power-independent phenomena on a more realistic channel will be to the disadvantage of the jammer power efficiency because a receiver will be optimized to compensate the channel's effect on signals from legit nodes. For example, a frequency and phase-shifted jamming signal will have less impact on the induced BER at the demodulation process of a legit node, as compared to an in-phase and frequency version of the same jamming signal.

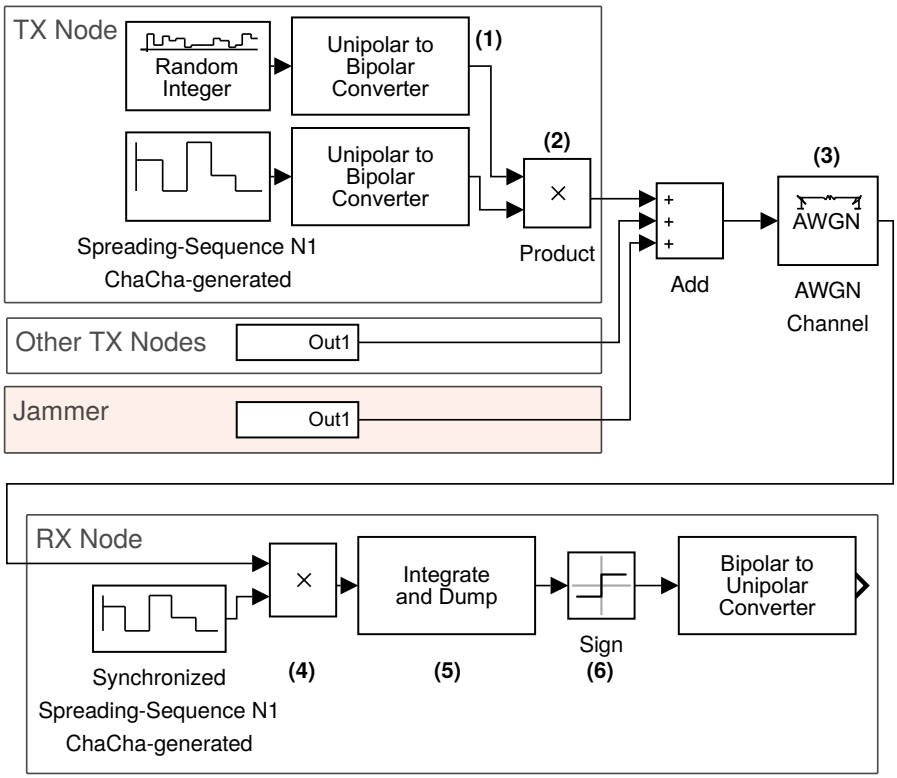


Figure 7.3: DSSS-CDMA System Model

7.6.2 *Attacker Model*

The attacker is a *jammer*. The jammer has access to the communication medium and can insert arbitrary signals. Its goal is to produce errors in the demodulation process of the receiver. Eavesdropping, tampering, or forgery of information is not a goal. The jammer has enough energy, power, and bandwidth to cover the entire band of the system with a signal of arbitrary power. The jammer can be further defined by the type of signal it inserts in the channel. We evaluate two types of jammers: a BBN Jammer and an Insider Smart Jammer (coherent and synchronous), both further detailed in Sections 7.6.3 and 7.6.4, respectively.

7.6.3 *Baseline Evaluation: Broadband Noise Jammer*

A BBN jammer places a random noise signal over the full width of the TX-node communication spectrum. A BBN strategy raises the noise level at the receiver and is a direct attack on the channel capacity of any communication system [133]. A BBN represents a *baseline* in terms

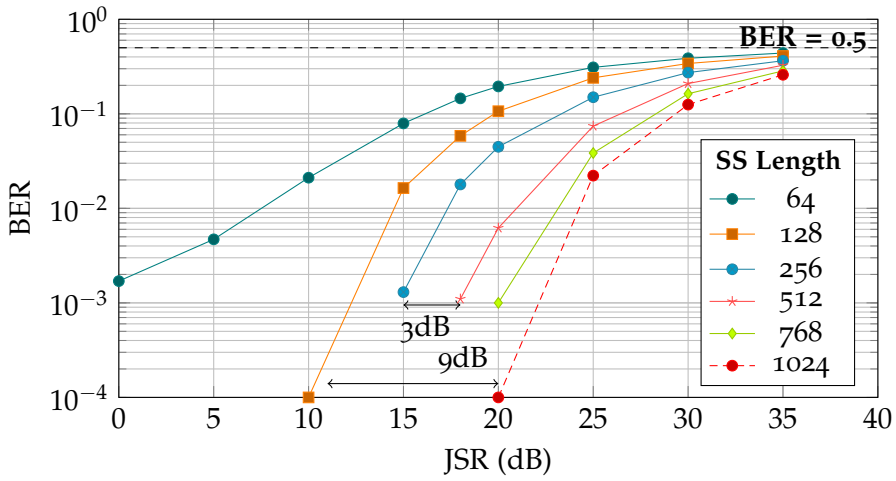


Figure 7.4: BBN jammer resilience

of jammer power efficiency for an *effective* jamming strategy against a DSSS system[10][133].

ATTACKER HYPOTHESIS The jammer knows the center frequency and bandwidth of the signal.

SIMULATION We used MATLAB/Simulink R2017a to model the system described in Sec. 7.6.1. There are 4 TX-nodes. We chose that number because it is representative of a one-hop-link in an IoT mesh network. Even if an IoT network has more nodes, for a given node, only nodes physically close have a signal-power relevance. Every TX-node constantly sends random binary data sampled at 1000 times/sec (1kbps), and we simulate 10 seconds (i.e., 10000 bits). Every TX-node uses a different SS previously generated with ChaCha20. The jammer power J_p is measured in relative terms against a single TX-node power S_p, this ratio is expressed as the JSR = J_p/S_p in dB[11]. The AWGN channel power is included in the JSR. We calculate the BER at the RX-node. We repeat the simulation with different SS lengths. The results are shown in Fig. 7.4, every data-point is a simulation run.

ANALYSIS Obtained results are consistent with the DSSS AJ theory [133]. When we double the SS length, we also double the signal bandwidth, and we obtain a ≈+3dB gain in resilience to Gaussian Noise for a fixed BER. The exception is the SS codes of length 64 for JSR<15dB. We explain this weaker resilience as a result of the higher CC values of the SS of the other legit TX-nodes. For this work, we consider a

10 Other jamming strategies such as Narrowband noise are not considered *effective*.
11 Power Decibel: $10\log_{10}(P1/P2)$

BER ≤ 0.1 to be acceptable for a digital communication system under jamming [133].

7.6.4 *Upper-Bound Evaluation: Insider Smart Jammer*

In this section, we define and model an *insider smart jammer*. We find it useful to explicit again our setting and evaluation goal.

SETTING AND GOALS The jammer is not limited in terms of energy; i.e., it can apply a jamming signal with power JSR persistently in time. There is no reactive AJ mechanism in our current evaluation. Thus, it is inevitable that for a given JSR value, the jammer will *defeat* (BER ≥ 0.1) the system. *An insider smart jammer will defeat most AJ systems with a JSR$\approx o$*. When the insider knows the AJ parameters of a single node, he can compromise the whole system's parameters. These AJ systems will be compromised either because the nodes share the same AJ parameters (e.g., hopping-schedule), or because the system uses a PR sequence set with no cryptographically secure pseudo-randomness properties (e.g., LFSR-generated, low-CC De Bruijn). In our proposal, the knowledge of one spreading sequence by an attacker does not imply the compromise of the other spreading sequences in the system. From a system perspective then, it is relevant to study the case in which the attacker has gained knowledge of the spreading sequence of *one node*. Then, measure the impact over the BER of the rest of the nodes in the system. Our system proposal was designed with the main objective of proactively mitigating this insider-node attack, and this section measures to which degree this is achieved.

The insider smart jammer represents an upper-bound for jammer power efficiency, under certain hypothesis.

ATTACKER HYPOTHESIS The jammer:

- Has perfect knowledge of the system (e.g., BPSK), except for the SSs used by the nodes.
- Is synchronized (time) and in-phase (*coherent*) with the SSs of the system.
- Knows the SS of *one* node, i.e., a compromised node.
- Can not compromise the SSs of other nodes[12].

12 See Appendix. C.1 for justification of this hypothesis.

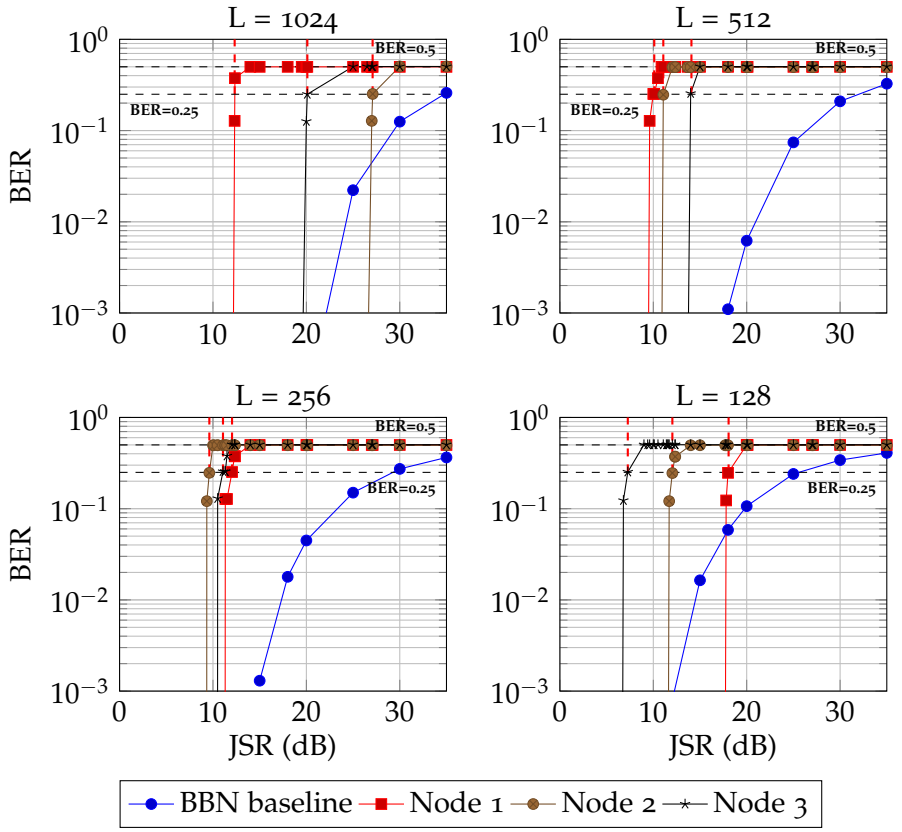


Figure 7.5: Smart jammer resilience for different SS Lengths L.

SIMULATION The set-up is the same as the BBN jammer. The jammer is modeled as a TX-node, which uses the SS of one compromised node. We measure the BER of non-compromised nodes for different values of JSR. The AWGN channel has a Signal-to-Noise Ratio (SNR) > 0 compared with a TX-node and is negligible. We plot the BBN jammer results for reference. The results are shown in Fig. 7.5.

ANALYSIS In all cases, the BER vs. JSR curve for individual nodes shows similar behavior. Let J_n be a threshold value unique to a $node_n$. Then, $BER = 0$ if $JSR < J_n$; followed by a steep positive slope at $JSR \approx J_n$, from $BER = 0$ to 0.5 ; finally, a constant $BER = 0.5$ for $JSR > J_n + \Delta JSR$. Not surprisingly, this threshold value J_n is related to the CC value between the spreading sequences of the jammer and the $node_n$.

In the following, we provide an informal explanation of the relationship between CC and $JSR_{BER=0.25}$ and refer to [19] for a general analytical expression. This relationship is valid for an integrate-and-dump DSSS correlator, and BPSK modulation. In our case, the jammer is in phase and frequency with the target signal. This maximizes the impact of the jammer in the target signal. Suppose that (A) the NCC

of the spreading-sequences is 1 (i.e., is the same sequence). In that case, a jammer with equal power as the TX-node ($JSR = 0dB$) sending random bits will achieve a $BER = 0.25$ at the RX-node. This BER value is explained because of the RX-node with equal probability $= 0.5$ decoding either the TX-node's bit or the jammers' bit, which in turn has a probability $= 0.5$ of being the same as the TX-node bit. This gives a total probability of decoding the correct bit of $p = 0.75 \rightarrow BER = 0.25$. (B) In the other extreme case, if the NCC is 0 (i.e., orthogonal sequences) theoretically there is no value for JSR that will affect the BER. (C) In the most general case, for an NCC between 0 and 1, say an NCC of a ratio $\frac{1}{N}$, the jammer needs N times more power to achieve the same effect as a $NCC = 1$. We formalize this relationship that derives from the work of [19], in the following Eq. 7.5:

$$JSR_{BER=0.25}(NCC) = 10\log_{10}(\frac{1}{NCC})[dB] \tag{7.5}$$

For example, if the $NCC = 0.1$ between the sequences of a jammer and a node, a jammer needs 10 times more signal power than the TX-node ($JSR = 10dB$) to achieve a $BER = 0.25$ at the RX-node.

For each node in our evaluation, we calculated the NCC and the theoretical JSR for $BER = 0.25$. In Fig. 7.5 we mark this theoretical value with a vertical dashed line from $BER = 0.25$ to 1. In all cases, the simulated results correspond to the predicted theoretical values.

SYNTHESIS We want to characterize the insider smart jammer resilience of a generic system implementing our proposal. In order to do so, we use (1) the study of the NCC for CSPR sequence sets from Section 7.5, and (2) Equation 7.5 that analytically relates $NCC \longleftrightarrow JSR_{BER=0.25}$. With these two elements, we can probabilistically describe the smart jamming resilience of BPSK-DSSS systems that use CSPR sequence sets. We transform the empirical values of NCC of CSPR sequences using Equation 7.5. The obtained results are shown in Fig. 7.6. With this ECDF representation, we can quickly determine the percentage (i.e., percentiles P_i) of the nodes in a generic system that will have a $BER \leq 0.25$, for a given JSR and sequence length L. An important observation is that contiguous curves in Fig. 7.6, where L is related by a factor of 2, are approximately +1.5dB apart ($\sqrt{2}$ in linear terms). In Appendix C.2, we confirm this observation empirically, and the results of Section 7.5.5 validate this relationship analytically.

APPROXIMATION OF BEHAVIOR FOR $BER \leq 0.1$ As stated before, an AJ resilient system should have a $BER \leq 0.1$ under a jamming attack. However, our study is relevant for $BER \leq 0.25$ because Eq. 7.5 relates $NCC \longleftrightarrow JSR_{BER=0.25}$. Nodes with BER values between $(0.1, 0.25]$ should be excluded in our study. We need an expression

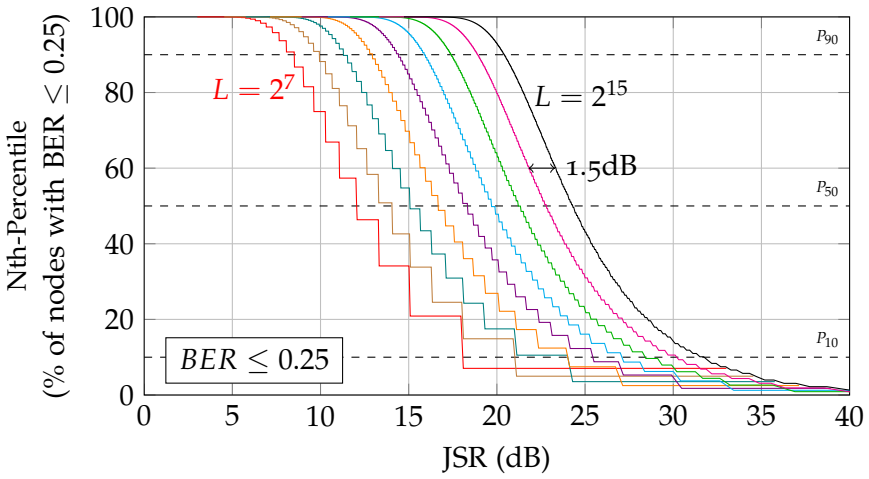


Figure 7.6: Percentiles for nodes with predicted $BER \leq 0.25$ as a function of an insider smart jammer power JSR, for different SS length $L \in \{2^7, 2^8, ..., 2^{15}\}$.

that relates $NCC \longleftrightarrow JSR_{BER=0.10}$. We were unable to find an exact analytical expression that predicted the simulated $BER = 0.10$ as a function of NCC. Yet, we found an upper-bound approximation. If we apply $-0.5dB$ to the analytical $JSR_{BER=0.25}$, in all simulated cases $BER = 0.0000$. Therefore, results in Fig. 7.6 will approximate $P(BER \leq 0.1)$, if we apply an horizontal-displacement $\Delta = -0.5dB$ to the ECDFs. This approximation is used to draw general conclusions about insider jammer AJ resilience of our system in the next section.

7.6.5 *Evaluation Summary*

Results from this section provide useful insights about how a system implementing our proposal will perform against jamming in general. We defined two scenarios that represent a baseline and an upper bound for jammer power efficiency. Any other effective jamming strategy against a DSSS system will fall in-between these.

The length L of the SSs is the most important factor for jamming resilience. Longer SSs provide better BER of the overall system in both scenarios. However, increasing L comes at a non-negligible cost: either bandwidth is increased for a fixed bit-rate, or bit-rate is lowered for a fixed bandwidth.

With the results from this section, we can quantify the jamming resilience we add to a CSPR-DSSS system by increasing L. With this information, a system designer can choose an appropriate trade-off between AJ and bandwidth/bit-rate. For the BBN jammer, results are well-known from DSSS theory: if we double L, the jammer needs ap-

proximately double the power (JSR $+3dB$). Also, a BBN jammer affects all the nodes of the network homogeneously independently of the SSs. The insider smart jammer scenario is different. The smart jammer affects each node differently. Hence, we can only give a probabilistic characterization of the AJ capabilities of the system. The cross-correlation properties of the CSPR SSs are a fundamental factor that determines the system resilience, Fig. 7.6 resumes the link between the statistical properties of CSPR SSs and smart jammer resilience.

To summarize, the provided jamming resilience of our proposal as a function of the CSPR sequences length L is:

- BBN Jammer resilience $\mathcal{O}(L)$
- Smart Jammer resilience $\mathcal{O}(\sqrt{L})$ [13]

Resilience performance against *any other* jamming attack should fall-in between those values. For the sake of completeness, for a fixed bit-rate, the bandwidth of a DSSS system is $\mathcal{O}(L)$; and for a fixed bandwidth, the bit-rate of a DSSS system is $\mathcal{O}(1/L)$.

7.7 RELATED WORK

This section reviews related work in two topics. First, we review work on correlation studies of PR sequences. Second, we introduce some works that use CSPR mechanisms for the design of AJ WCSs.

7.7.1 *Correlation of Pseudo-Random Sequences*

The study of correlation properties of PR sequence sets has been focused on the families highlighted in Sec. 7.3.2. Those families are LFSR-based, De Bruijn sequences [169, 170, 191], Legendre/Jacobi sequences [31, 38], and Chaotic Sequences [90, 112, 181]. The most studied family is LFSR-based. A classical CC reference is the work of Swarte et al. [151]. More recent work is given by Zepernick et al. [201], it also covers other PR families. For a particular PR family, we refer the reader to the corresponding cited works of a given family. *The CSPR Sequence Sets use throughout this work does not correspond to any PR family in the literature*.

Notwithstanding the fundamental and well-studied topic of the uniformly random probability distribution, there is a lack of studies

13 Justification for the square-root relationship between smart jamming resilience and L is given in Sec. 7.5.5 and Appendix C.2.

on the CC properties of uniformly-distributed sequence sets. Schotten et al. [154] gave an analytical formula for the auto-correlation of true-random sequences (i.e., a Bernoulli process) with the assumption of Golay's ergodicity postulate. However, to the best of our knowledge, no such analytical equivalent exists for CC characterization of uniformly-random sequence sets. Pöpper et al. [135] provided an empirical characterization of the CC of *random codes*[14], giving three percentiles (P_{95}, $P_{99.99}$, P_{100}) for sequences of length [$128 \leq L \leq 1024$], and sets with cardinality 1000. From the pure-mathematics field, Kuipers et al. [97] studied properties of an operation similar to the *convolution* on topological spaces linked to the uniform distribution. However, Kuipers et al. focused on demonstrating linear relationships between the operations.

7.7.2 *CSPR-based AJ WCSs*

The closest proposal in the literature to ours is NATO's unclassified work by F. Hermanns [66] (German Patent [65]). It proposed to use AES-OFB cipher[15] output as *code-hopping* (CH) sequences. The proposal is a hybrid DSSS-FHSS system. Unlike classical DSSS, where a central carrier frequency is known, CH also "hops" the center carrier frequency, thus doing FHSS. If originally available bandwidth was unused by the transceivers, this approach effectively increases AJ resilience. However, if the transceivers already used DSSS over the whole available bandwidth, the gain of this approach is yet to be evaluated. Hermanns evaluates the linear-complexity of AES-OFB sequences, empirically measures the auto-correlation spectra for a sequence, and evaluates the multi-user and AJ performances with a simulated system (the simulation platform is not disclosed). The jammer model is not fully specified for the AJ evaluation. We assume a synchronous and coherent jammer model was used because its results were consistent with ours. For a sequence length of 1000, it measured a *security gain* of +10 dB at the $BER = 10^{-3}$ level.

T. Song et al. [168] focus on a single link of a CDMA communication under *disguised jamming* (equivalent to the smart jammer, but not an insider). They propose to use AES (mode of operation not detailed) to encrypt LFSR-generated PR Sequences. They use the term *Secure Scrambling*[16] to refer to this operation. The legit parties share secret keys of 128, 192, or 256 bits. They focus on an analytic study of the impact of a disguised jammer who does not know the sequence, and

14 Not specified how they were generated, but probably with a CSPRNG.
15 Which behaves functionally as a stream-cipher.
16 *Scrambling* is also used in classical wireless literature to refer to long DSSS **PN!** (PN!) sequences, not necessarily cryptographically secure.

the system using the Arbitrarily Varying Channel (AVC) model. It is relevant to note, in the context of our work, that the generic analytical results obtained by Song et al. in the AVC model can also be applied to a single-link of our proposal, as an AES-output should behave as a CSPR output. They conclude that the secure scrambling method improves the resilience against disguised jamming for a single link.

FHSS work by M. Tiloca et al. [175] is proposed in the context of IEEE 802.15.4 in TSCH mode. They created a CSPRNG based on AES-CTR. The CSPRNG output is used to execute their Secure Link Permutation (SLP) algorithm, namely a pseudo-random permutation. SLP output pseudo-randomly determines the TSCH schedule. Also, they periodically change the TSCH schedule. They implemented their SLP algorithm in Contiki OS with TSCH, and evaluated it in TelosB IoT nodes. They compared the results against a fixed non-AJ TSCH schedule. Their proposal effectively defeats a selective-jammer, with negligible energy and packet-delivery-ratio penalties.

D. Torrieri [179] discusses the concept of *maneuver keys* in the context of MTD applied to a DSSS system. The work is a high-level design of such a system: the system nodes share a given group *key* k_g that they use as the sequence for DSSS modulation. How this key is generated and distributed is not detailed. The system is reactive, i.e., Intrusion Detection System (IDS) is present in the network. Moreover, if a jammer node is detected, it will be isolated from the rest of the system by initiating a secure group re-keying that excludes it. Later, Torrieri [178] re-categorized this proposal within the cyber-defense concept of *cyber maneuvers*.

To the best of our knowledge, no multi-link wireless system AJ proposal in the literature is *proactively* resilient to an *insider-node attack*. The system proposed by Hermanns [66] is potentially resilient, but it lacks sufficient detail about the distributed mechanisms and properties of the system (e.g. how the SSs are generated and distributed), AJ was evaluated for a single-link. Song et al. [168] AES-based *secure scrambling* proposal is, at the core, similar to our proposal. However, as [66], it focuses on a single link, the distributed mechanisms to generate SSs/codes for multiple users are not detailed, nor the system-wide AJ properties evaluated. Thus, under the hypothesis of an insider attacker that has knowledge of the secret parameters of one legit node, the system resilience for other nodes can not be estimated (e.g., discovering the LFSR used as input can compromise the other nodes). In Tiloca et al. [175], all nodes share the same frequency-hopping schedule: the calculation is *distributed*, but not *independent*. Inevitably, an insider attacker can efficiently jam the whole system. Finally, in Torrieri [179] all nodes share the same SS. In case of an insider jammer, all links will be efficiently jammed, defeating the system. The system

will only recover *after* some time when the IDS isolates the jammer. This is a *reactive* strategy to mitigate insider attacks. It excludes the node from the network only *after* some given process. In other words, jamming-detection is needed. Jamming-detection is a prominent field in the AJ literature and is used in reactive AJ strategies.

All works cited in this section, including our own, assume pre-shared secrets between the sender and receiver to execute AJ spread-spectrum techniques. In a real-word setting this hypothesis is not always true. In many IoT use cases, previously-unknown nodes have to bootstrap ad-hoc mesh networks. Those nodes do not share common cryptographic material. Physical-layer AJ bootstrapping is a hard problem to solve. AJ systems without pre-shared secret information are needed. This fundamental topic is called *keyless jam resistance*. We refer the reader to Kang et al. [81] for a recent survey on DSSS-based keyless AJ. And to J. Tao et al. [79], and C. Pöpper et al. [135], for pairwise and broadcast communication proposals, respectively.

7.8 DISCUSSION

In this section, we discuss security-related issues, our system proposal in the IoT context, and deployment challenges.

The main novelty of our DSSS system proposal is that it uses pairwise CSPR SSs generated in a *distributed* and *independent* way. Inherited by the IANVS-design, the proposal also has crypto agility; for example, it can be instantiated with future software-based IoT-friendly crypto primitives instead of ChaCha20.

The study of the CC properties of CSPR sequence sets proved to be fundamental to evaluate the AJ resilience of our proposal. First, we found that an in-depth study of the CC of independent uniformly distributed random binary sequences was missing in the literature; thus, we provided both an empirical statistical characterization of the CC values of large CSPR sequence sets and a probabilistic study. Then, we designed and evaluated our system in MATLAB; notably, against a power-efficient insider jammer. Finally, we linked the CC properties with the AJ resilience and characterized the AJ resilience of a generic system implementing our proposal.

7.8.1 *Security-related issues of PR Sequence Sets*

The AJ capabilities of a spread-spectrum wireless system depend on the *secrecy* of the sequences used. A jammer with knowledge of the

spreading parameters can attack the system very power-efficiently [8, 168]. In terms of security (i.e., keeping the secrecy), the shortcomings of AJ solutions using legacy PR sequences are twofold. First, LFSR-based sequences (e.g., Gold codes for 3GPP-UMTS) can be brute-forced in a computationally reasonable time (Berlekamp-Massey [111, 168]); this affects the secrecy of only one sequence. Secondly, PR sequence sets are composed of not independent sequences (e.g., to guarantee cross-correlation). Thus, knowing one sequence leaks information about the others. This second issue is very relevant to our insider attacker setting; For example, the knowledge of one De Bruijn low-CC code will ease the task of breaking the other codes in the set. The first issue can be addressed by using non-LFSR codes, and has been explored in the bibliography (e.g., De Bruijn, AES-based codes). However, the second one can only be addressed by sets where all the sequences are independent of each other (e.g., no cross-correlation constraints). Or proposal, CSPR codes generated independently, addresses both weaknesses. To the best of our knowledge, our work is the first one to propose sequence sets for WCSs composed of independent sequences, which we called CSPR Sequence Sets.

7.8.2 *Non-security impacts of CSPR Sequence Sets*

Our proposed CSPR-based sequence generation process prioritizes as design factors cryptographically secure randomness and independence of sequences. This has an impact in the CC of the sequences in the sets. There are no hard-guarantees about CC max values. We refer the reader to Sec. 7.5.6 about the implications in comparison to other families of sequences. To complete the CC discussion, we add that one of the main contributions of our work is the study in Sec. 7.5 of the CC properties of CSPR sequences sets that was lacking in the bibliography. We have not fully-studied the consequences they have for real systems in the current work (i.e., we focused on their AJ capabilities and evaluated for an AWGN-model). However, their statistical properties look promising, specially for low-throughput and systems with many nodes (or with potential rotation of codes per user as in our generic MTD proposal). For example, the mean value ($\sqrt{2/\pi L}$) and PMF of the NCC as a function of the sequence length L, can be used to have an estimation of the performance of large IoT systems implementing MTD-dynamic CSPR sequences. Also, because of the perpetual rotation of the used sequences in an MTD technique, the CC statistics properties are relevant even for a single node. Regarding the performance of the system when not under jamming, Hermanns [66] studied AES-based sequences for systems with 1 to 40 nodes and found that they perform as Gold Codes.

Another trade-off could be the computational cost of generating CSPR sequences as compared to LFSR-based solutions. Advances in hardware and software allow for suitable implementations of CSPRNGs on IoT devices that make this not being an issue. Most IoT System-on-Chips have AES Hardware Modules, and software-based solutions -like ChaCha20- are fast on IoT devices [39].

7.8.3 *Relevance of This Proposal for IoT Systems*

The AJ resilience against an insider jammer can only be measured in terms of probability. The pseudo-randomness and independence of sequences at the core of our system design are the main causes. In contrast, classical wireless systems precisely determine many properties *a priori*, i.e., with probability 1. For example, maximal CC or max-bounds for multi-hop latency. However, this lack of hard-values certainty is not a big drawback for the MTD IoT systems we target. A large number of nodes and MTD-inspired rotation of sequences over time are both properties that make any particular system to converge statistically to the CC -and thus AJ- properties we studied.

Furthermore, the cyber-defense objective in our IoT setting is not to protect a single (or limited number of) primary wireless targets, like a satellite link, a radar system, or cellphone-users. Instead, the *IoT system as a whole* is the target to defend. In other words, we care about the *service* the IoT system provides, and not the individual IoT nodes. For example, we can design a system to provide a given service, even if only 10% of the IoT nodes (or any given node only 10% of the time) will be resilient to a +25dB powerful jammer. The service could be provided, albeit in a degraded mode; This is what we understand as a resilient system. In constrained-node IoT networks “strength lies in (big) numbers".

7.8.4 *Key deployment challenges*

A real-world implementation of our proposal will have to deal with the synchronization of SSs for signal demodulation at the RX-node. This problem can be solved because CSPR sequences have good auto-correlation properties [154]. Also, the *bootstrapping* of the system, i.e. the IANVS's AKE and Auth-SYNC components, is not a trivial problem. In this respect, keyless jam resistance [81] techniques can be used to protect the physical layer and execute higher-layer communication protocols (e.g., EDHOC key establishment, LATe). Another point to be discussed is the availability of DSSS technologies in real-world IoT

systems. IoT hardware uses mostly narrow-band technologies, where FHSS techniques are dominant. IEEE 802.15.4 has a DSSS mode, but the spreading-sequences have length $L = 16$, which is not long enough for robust AJ. However, SDR technology has yet a bigger role to play in the future of IoT [85]. Cost-affordable SDR technologies will add to the synergy of new-services enabled by the IoT. In this context, alternating between DSSS, FHSS, or Long-Range radio will be a matter of executing some lines of computer code in already-deployed IoT nodes.

7.9 CONCLUSION

In this chapter, we presented an IANVS-based MTD technique targeting the network physical layer of IoT systems. We provided a DSSS AJ solution that is proactively resilient to insider-node jamming attacks.

Regarding this memoir's main RQs, this chapter provided another instance of an MTD in a concrete IoT use case (RQ-3.I1). Our proposal was motivated by the fact that IoT systems are inherently exposed to *jamming attacks*, and that most of the off-the-shelf IoT radio technologies do not have robust AJ properties. Moreover, insider-node attacks will defeat most AJ solutions and are a real threat in the heterogeneous IoT ecosystem.

The main novelty of our proposal is the use of DSSS Spreading Sequences independently generated with CSPRNGs leveraging on the IANVS framework. We implemented and evaluated the proposed system and attacker-jammer in simulation using MATLAB. The experimental results validated the insider jammer resilience claim of our proposal. We explain the results -and generalize them- by the CC properties of CSPR sequence sets, for which we provided an in-depth statistical and probabilistic study that was missing in the literature.

This chapter concludes our contributions and the third part of this memoir. In the following chapter, we present some general conclusions and future work perspectives that close this manuscript.

Research Questions

RQs answered:

- **RQ-3.I1**: How to instantiate MTDs in concrete IoT use cases?
  - [IANVS-based] We instantiated a physical-layer MTD technique with the DSSS SSs as MP (*Design*, simulation *Implementation*, and simulation *Evaluation*).

CONCLUSION

8

CONCLUSION AND PERSPECTIVES

In this chapter, we conclude this dissertation by offering a summary of its contributions and providing future work perspectives.

8.1 CONCLUSION

The main research goal of this dissertation was *to improve the resilience of the constrained IoT*. More precisely, to improve it through the use of the MTD cyber defense paradigm. Before this work, MTD was an established technique in non-IoT systems, but its feasibility in constrained IoT systems was uncertain.

Part I (Ch. 2-3) of this dissertation, *exploratory* by nature, scrutinized the field of MTD techniques for the constrained IoT. First, in Chapter 2 we presented fundamental background on the MTD paradigm, the constrained IoT domain, and IETF's IoT network security protocols. In this early chapter, we acknowledged the network component as the key enabler of constrained IoT systems. MTD for IoT was almost nonexistent in the literature, and the RQ arised: *Is MTD for the constrained IoT possible?*

In Chapter 3, we performed a SLR of MTD techniques for the constrained IoT. We identified thirty-two distinct techniques, of which two-thirds were not identified in previous surveys. We provided an evidence-based assessment of the security status of the techniques and

a validation of the feasibility of MTD for IoT. Our SLR was the first MTD review to focus on the techniques' cryptographic primitives. Besides, we developed four novel entropy-based *metrics* that we applied in conjunction with Shannon's entropy to characterize the state of the art. These metrics have applications beyond the scope of the review, and constitute a minor contribution in MTD's *Evaluation* research field (2.2.3). In summary, we identified: a predominance of MTD network techniques, that most of the techniques provide strong evidence about their deployment, and a generalized lack of sound security foundations. These results motivated the rest of our research contributions. We continued focusing on network-based research. We foresaw many open opportunities, even if it was the most explored field. Finally, the SLR results set a research baseline: to keep contributing with *usable* techniques while improving their *security* foundations. We synthesized these motivations in the RQ that guided the rest of the dissertation: *How to create usable and secure MTD techniques for the constrained IoT?*

The remaining two parts of the dissertation had a *constructive* purpose. Part II (Ch. 4), focused on general *design* aspects for the creation of *usable* and *secure* MTD techniques for the constrained IoT, while Part III (Ch. 5-7) on the *instantiation* of these MTDs in concrete IoT use cases.

In Chapter 4, we addressed fundamental questions about the *design* of MTD techniques in the constrained IoT context. We tackled one by one the RQs of *what* are suitable MPs in IoT systems, *how* and *when* to move them. First, we identified more than 30 components in IoT systems that have the potential to become MPs. We focused on Network components using a five-layer model (physical, link, network, transport, and application) and corroborated that network-based components have great potential for MTDs. Then, we proposed IANVS as an answer to the *how* to move those components. Our proposal is a modular framework that can instantiate MTD techniques suitable for constrained IoT systems. It is composed of four building blocks that correspond to fundamental cryptographic or network security fields: *AKE* to AKE, *Auth-SYNC* to authenticated and fresh information exchange (e.g. secure time synchronization), *CSPRNG* to CSPRNGs, and *MP-Map* to mathematical maps. Thus, a concrete instantiation of IANVS can leverage on well-established solutions and is crypto-agile by design. For example, in the future, a component's instance can be replaced with a more lightweight or robust state-of-the-art variant, maintaining the security and functional objectives of the concrete IANVS-based MTD without the need of major design changes. Finally, we discussed how to achieve synchronized MP movement leveraging on the Auth-SYNC component to execute event- or time-based answers to the *when* to move.

The IANVS framework is one of our fundamental contributions and was the "leitmotiv" of the remaining part of the memoir.

Chapter 5 starts Part III and introduces the LATe synchronization protocol. Our nonce-based secure time synchronization proposal is a solution for the instantiation of a time-based Auth-SYNC IANVS component suitable for constrained IoT systems. LATe is a client-server two-message protocol, and optimizes the size of the messages -leveraging on CBOR and COSE IETF standards- and the cryptographic operations needed at the client. We provided a computer-aided formal method proof of the security claims of LATe using the Scyther tool. We also discussed real-world implementation issues, attacks, and mitigations, that were not captured by the formal method's model.

In Chapter 6, we presented two concrete Network MTD techniques that instantiate the IANVS framework. Both were motivated by the same threat use case: a remote DoS attacker targeting a constrained IoT node that hosts a CoAP server. The first proposal is an MTD technique with UDP port numbers as MP, also known as port-hopping, and aims at mitigating the attack at a transport-layer level. The second proposal is an MTD technique where the MP is the CoAP "`.well-known/core`" resource's URI. This URI is the high-level target resource of the DoS attack. Thus, this second technique aims at mitigating the attack at the application layer. We implemented the UDP port-hopping technique in Pycom LoPy4 IoT nodes as a hardware platform and shared the source code. We evaluated the UDP port-hopping technique in a real testbed. We measured its effectiveness to mitigate the reconnaissance phase of the attack (i.e. port-scanning). We defined a probabilistic model to predict the MTD's effectiveness that corroborated the empirical results. This chapter provided two MTDs in concrete IoT use cases and illustrated how to leverage on IANVS elements to ease their instantiation. For example, by not starting from the MTD design fundamentals and by reusing implemented components.

Finally, in Chapter 7, we presented a Network physical-layer MTD technique. Unlike the previous chapter 's techniques that target end-to-end connectivity and attacks, this one focus on the first "hop" of the communication channel. Indeed, if an attacker disrupts any network layer in an end-to-end communication's path, no communication nor exchange of information is possible: the IoT system will not provide its service. In this chapter, we were motivated by the threat use case of insider-node attacks and *jamming*. As stated before, jamming is a very straightforward and effective attack: by emitting a signal over the channel, information flow of the system can be stopped or severely disturbed. The laws of electromagnetism and Shannon-Hartley's theorem guarantee the attacker's success if it has enough signal power. Moreover, we assumed an insider-node jamming attack that has knowl-

edge of the network-known parameters of AJ techniques. Our solution for this setting was a IANVS-based AJ technique that improves the insider-node jamming resilience of IoT systems and leverages on the DSSS radio-modulation technique. The main novelty of our proposal resided in the use of DSSS SS independently generated with CSPRNGs. Sequence sets generated this way were not present in WCS literature and we named this family "CSPR". We studied the CC statistical and probabilistic properties of large CSPR sequence sets and uniformly random sequence sets, and showed that CSPR sequences have robust CC properties. This fundamental study was lacking in the WCS literature. Finally, we evaluated the jamming resilience of our proposal using a model implemented in MATLAB. We exposed our system to an insider smart jammer and validated that the attack was mitigated. Insider-smart-jammer AJ resilience and CC of sequences were analytically linked.

This dissertation contributed to establish MTD as a cyber defense technique for IoT systems. There are no final "victories" in the endlessly changing cybersecurity field, but MTD is a welcome player for the defender's side. MTD's motivation is about the inevitable defeat of static systems facing adaptive adversaries. In a second-order degree, this will eventually apply to the MTD-based designs present in this memoir. For the most part, we practiced the principles of openness and reproducibility in our contributions. Not only because we believe it is the most robust way to approach security and research in general, but in the hope that they will evolve in the hands and minds of the people that try to keep the never-ending cybersecurity "battle" at the cyber defender's advantage.

We finish this subsection with a summary of the RQs that guided this dissertation and the chapters in which they were addressed.

Research Questions and their addressing Chapters

RQ-1: Is MTD for the constrained IoT possible? → Ch. 3.

RQ-2: What is the status of MTD for IoT techniques? → Ch. 3.

RQ-3: How to create *usable* and *secure* MTD techniques for the constrained IoT? → Ch. 4 - 7.

- **RQ-3.D1**: What are suitable MPs in IoT systems? → Ch. 4.2.
- **RQ-3.D2**: How to move distributed MPs in IoT systems? → Ch. 4.3.
- **RQ-3.D3**: When to move the MP? → Ch. 4.4.
- **RQ-3.I1**: How to instantiate MTDs in concrete IoT use cases?
  - [IANVS-based] We instantiated a physical-layer MTD technique with the DSSS SSs as MP (*Design*, simulation *Implementation*, and simulation *Evaluation*). → Ch. 7.
  - [IANVS-based] We instantiated a transport-layer MTD technique with UDP port numbers as MP (*Design*, hardware *Implementation*, and theoretical and hardware *Evaluation*). → Ch. 6.
  - [IANVS-based] We instantiated an application-layer MTD technique with CoAP ".well-known/core" resource's URI as MP (*Design*). → Ch. 6.
  - [Auth-SYNC] We instantiated a secure time synchronization protocol, suitable for time-based *Auth-SYNC* components of IANVS-based MTDs (*Design*, and formal-method security *Evaluation*). → Ch. 5.

8.2 FUTURE WORK

In the following, we describe some future research axes that we identified during this dissertation.

8.2.1 *Security: Fundamentals, Design, Proofs, and Openness.*

The SLR identified a lack of sound security fundamentals of most MTD techniques for the constrained IoT. Novel security solutions should re-use, as much as possible, established security blocks, at least crypto

primitives. As illustrated by the LATe protocol development, designing new security-related protocols is not straightforward. The heterogeneity of constrained IoT use cases inevitably leads to the development of novel protocols tailored for them. Even if sound cryptoprimitives are used, their composition may lead to an non-secure solution (See an extended discussion in Ch. 4.3.4). Computer-aided security proofs should be integrated as much as possible in the design of protocols. These semi-automated proofs can find security-flaws early in the development of protocols.

However, security can never be guaranteed with total certainty. A security proof or model (system's and attacker's) will never be able to capture all the nuances of reality. Side-channels attacks and real adaptive attackers are certainly not constrained by a theoretical model. The IoT research community should work together, prioritize re-usable components, automating proofs, source-code sharing—all elements in line with Kerckhoffs's white-box open cryptography principles. In the hope to leverage the work of future researchers, and avoid as much as possible *(re)inventing a, probably square-shaped, wheel*.

8.2.2 *MTD Techniques (What?): Unexplored MPs, SDR, and SDN.*

There are several possibilities for the design of novel MTD for IoT techniques using MPs from unexplored taxonomies. The SLR identified almost a complete lack of techniques in the *Platform* category. We explain this by the inherent limitations of constrained IoT hardware. Novel techniques could leverage on legacy MTD, where *Platform* techniques account for 20%, and adapt the most suitable proposals. Also, IoT *Data* techniques are under-explored. This kind of techniques can be instantiated in real use-cases as it is a field more related to information theory where IoT constraints can be more easily overcome.

Finally, *Network* techniques are predominant in the state of the art, but -even in that category- there are still many opportunities that we identified in Ch. 4.2. Software-Defined technologies like SDR or SDN are particularly promising, once they will become economically or technically possible for the IoT. They will allow the implementation of MTD techniques at the physical layer and the logical topology level, respectively, without the need to redeploy nodes. Routing protocols for multi-hop IoT networks like RPL can also gain in robustness by implementing MTD-inspired mechanisms.

8.2.3 *MTD Techniques (How?): IANVS-II, Multiple MPs in same domain, adaptive and cross-layer MTDs.*

IANVS's intent is to open-up possibilities in the development of concrete MTD techniques and possibilities for refinements of the framework itself. We identified limitations on possible instantiations of the framework if two MPs are moving in the same domain, briefly discussed in the port-hopping proposal (Ch. 6.2.2). We are currently working in an iteration of IANVS. The *CSPRNG* and *MP-Map* components could leverage on pseudo-random permutations and nonce misuse-resistant solutions. Additional challenges remain when mapping MPs into a smaller domain than the original, i.e., collisions can happen. The IANVS's *CSPRNG* and *MP-Map* components definitions and interactions need to be further studied.

In this dissertation, we only studied periodic MTD movement allowed by the *Auth-SYNC* component; but the component also enables active triggering of MTD movement. These IANVS-based solutions are interesting to develop because will lead to the instantiation of active or reactive MTDs that may be suitable to mitigate adaptive attackers.

Finally, cross-layer MTDs, were almost not explored in this dissertation. A common MTD framework can leverage the implementation charge in a single IoT node (e.g. re-using binary code). However, special precautions should be taken to make the security of one layer's MTD *independent* of the other layer's. In other words, if one layer's MTD is penentrated, the other MTD should still be secure (e.g. use at least different secret keys from the *AKE* component).

8.2.4 *Evaluation: The need for usable MTD metrics, security and system-performance/cost (trade-offs).*

To which degree is an MTD technique desirable? Metrics can help with the answer. Unfortunately, most MTD metrics are of difficult empirical applicability, as discussed in Ch. 3.4. Empirical evaluation of MTD techniques is a field that needs more development. This fact motivated the definition of the entropy-related metrics -and Shannon's entropy- focusing on practical applicability, but our metrics are only a first step and can certainly be refined. Evaluation is about the security of a system, but also about the system performance to provide its intended functional goal (non-security metrics). There is always a trade-off between security and functional goal performance, i.e., security measures increase the system's workload. Non-security metrics of empirical applicability should be developed too. Both definitions of novel security and non-security metrics will provide non-trivial

challenges due to the heterogeneous nature of MTD techniques. But, even non-generic metrics applied to a single MTD can help system designers and implementers to make relevant trade-off decisions about that particular MTD.

8.2.5 *DSSS physical modulation with CSPR sequences.*

The use of independently generated CSPR sequences for WCS proven to be promising for AJ purposes, also validated by their CC properties (Ch. 7). The proposal of CSPR sequence sets opens many research opportunities. Multi-user WCS should be thoroughly evaluated, we provided a theoretical estimation and a four-nodes simulation, but more dense networks should be evaluated -at least- in simulation. Also, CSPRNGs can be used to generate non-binary types of sequences; for example, complex number sequences $\{\pm 1, \pm i\}$. Other modulations than BPSK can leverage on them. In terms of the system simulation, non-AWGN channel models could be used if the non-AJ properties of the system are of interest, e.g., multi path, or fading. Synchronization of CSPR SS is a field that will need to be studied, and has good perspectives because CSPR sequences are known to have good auto-correlation properties [154]. In terms of jamming attacks, the definition of other adversarial settings and a comparative study of other AJ proposals in these settings seems interesting –although implementing different AJ proposals will be challenging–. Those adversarial settings can include dynamic attacker-system interactions. For those scenarios, our MTD IANVS-based proposal has the elements needed to design and implement an *adaptive* AJ defense strategy. Indeed, we did not study the dynamic component of our proposal. Finally, a hardware instantiation of our proposal in SDR technologies will be the a milestone that can open-up more possibilities. This SDR instantiation will allow to implement dynamic reconfiguration of modulation properties (e.g., not only DSSS) and MTD parameters (e.g., different periods of movement), and allow for its evaluation in real-world IoT use cases and jammers.

APPENDIX

A

RÉSUMÉ EN FRANÇAIS

Le principal objectif de recherche de cette thèse est d'améliorer la résilience de l'IoT contraint. Plus précisément, de l'améliorer par l'utilisation du paradigme de cyberdéfense MTD.

La première Partie (Ch. 2-3), *exploratoire* par nature, a examiné le domaine des techniques MTD pour les contraintes IoT. Tout d'abord, au Chapitre 2, nous avons présenté le contexte fondamental du paradigme MTD, du domaine IoT contraint et des protocoles IETF de sécurité du réseau IoT. Dans ce premier chapitre, nous avons reconnu que la composante réseau est le principal élément facilitateur des systèmes IoT contraints. Les techniques de MTD pour l'IoT étaient presque inexistantes dans la littérature, et le RQ (Question de Recherche) s'est imposé: *Est-il possible d'utiliser MTD pour l'IoT contraint?*

Au Chapitre 3, nous avons effectué un SLR (Revue Systématique de la Littérature) de techniques MTD pour le IoT contraint. Nous avons identifié trente-deux techniques distinctes, dont deux tiers n'avaient pas été identifiées lors des enquêtes précédentes. Nous avons fourni une évaluation fondée sur des éléments probants de l'état de sécurité des techniques et une validation de la faisabilité de l'MTD pour l'IoT. Notre SLR a été le premier examen de MTD à se concentrer sur les primitives cryptographiques des techniques. En outre, nous avons développé quatre nouvelles *métriques* basées sur l'entropie que nous avons appliquées en conjonction avec l'entropie de Shannon pour caractériser l'état de l'art. Ces métriques ont des applications qui dépassent le cadre de l'examen et constituent une contribution mineure dans le domaine de recherche MTD sur le sujet *Évaluation* (2.2.3). En résumé, nous avons identifié: une prédominance des techniques de type réseaux MTD, que la plupart des techniques fournissent des preuves solides de leur déploiement, et un manque généralisé de bases de sécurité solides. Ces résultats ont motivé le reste de nos contributions.Nous avons continué à nous concentrer sur la recherche basée sur le composant réseau. Nous avons prévu de nombreuses opportunités ouvertes, même si c'est le domaine le plus exploré. Enfin, les résultats de l'étude SLR ont établi une base de référence pour la recherche: continuer à contribuer avec des techniques *utilisables* tout en améliorant leurs bases *sécurité*. Nous avons synthétisé ces motivations

dans le RQ qui a guidé le reste de la thèse: *Comment créer des techniques MTD utilisables et sécurisées pour l'IoT contraint?*

Les deux autres parties de la thèse avaient un but *constructif*. La deuxième Partie (Ch. 4), s'est concentrée sur les aspects généraux du *design* pour la création des techniques MTD *utilisables* et *sécurisées* pour les IoT contraints, tandis que la Partie III (Ch. 5-7) sur l'*instantiation* de ces MTDs dans des cas d'utilisation concrets IoT.

Au Chapitre 4, nous avons abordé des questions fondamentales sur la *conception* des techniques MTD dans le contexte contraint IoT. Nous avons abordé une à une les RQs de de savoir *quels* (*what*) sont les MPs appropriés dans les systèmes IoT, *comment* (*how*) et *quand* (*when*) les déplacer. Tout d'abord, nous avons identifié plus de 30 composants dans les systèmes IoT qui ont le potentiel de devenir MPs. Nous nous sommes concentrés sur les composants de réseau en utilisant un modèle à cinq couches (physique, lien, réseau, transport et application) et avons corroboré que les composants basés sur le réseau ont un grand potentiel pour l' MTDs. Ensuite, nous avons proposé l'IANVS comme réponse au *comment* (*how*) déplacer ces composants. Notre proposition est un framework modulaire qui peut instancier des techniques MTD adaptées aux systèmes IoT contraints. Il est composé de quatre blocs de construction qui correspondent à des domaines fondamentaux de la cryptographie ou de la sécurité des réseaux: *AKE* à AKE, *Auth-SYNC* à l'échange d'informations authentifiées et fraîches (par exemple la synchronisation du temps sécurisée), *CSPRNG* à CSPRNGs, et *MP-Map* aux applications (*maps*) mathématiques. Ainsi, une instanciation concrète de l'IANVS peut s'appuyer sur des solutions bien établies et est crypto-agile par conception. Par exemple, à l'avenir, l'instance d'un composant peut être remplacée par une variante de pointe plus légère ou plus robuste, ce qui permet de conserver les objectifs de sécurité et de fonctionnalité de l'MTD concrète basée sur l'IANVS sans devoir procéder à des modifications majeures de la conception. Enfin, nous avons discuté de la manière de réaliser des mouvements du MP synchronisé en s'appuyant sur le composant Auth-SYNC pour exécuter des solutions à la question de *quand* se déplacer basées sur des événements ou basées sur le temps.

Le framework IANVS est l'une de nos contributions fondamentales et a été le «leitmotiv» de la partie restante du mémoire.

Le Chapitre 5 commence la Partie III et présente le protocole de synchronisation LATe. Notre proposition de synchronisation temporelle sécurisée basée sur le protocole nonce est une solution pour l'instanciation d'un composant Auth-SYNC IANVS basé sur le temps et adapté aux systèmes IoT contraints. LATe est un protocole client-serveur à deux messages, et optimise la taille des messages -en s'appuyant sur les standards IETF CBOR et COSE- et les opérations cryptographiques

nécessaires au niveau du client. Nous avons fourni une méthode formelle assistée par ordinateur prouvant les affirmations de sécurité de LATe à l'aide de l'outil Scyther. Nous avons également discuté des problèmes réels de mise en œuvre, des attaques et des mesures d'atténuation, qui n'étaient pas pris en compte par le modèle de la méthode formelle.

Au Chapitre 6, nous avons présenté deux techniques concrètes de MTD du type Réseau qui instancient le framework de l'IANVS. Toutes deux ont été motivées par le même «cas d'utilisation de la cybermenace»: un attaquant DoS (Déni de Service) distant ciblant un nœud IoT contraint qui héberge un serveur CoAP. La première proposition est une technique MTD avec des numéros de port UDP comme MP, également connue sous le nom de saut de ports (port-hopping), et vise à atténuer l'attaque au niveau de la couche transport. La deuxième proposition est une technique MTD où le MP est le CoAP «`.well-known/core`» URI, c'est-à-dire la ressource cible de haut niveau de l'attaque; notre technique vise donc à atténuer l'attaque au niveau de la couche application. Nous avons mis en œuvre la technique de saut de port UDP dans des nœuds Pycom LoPy4 IoT comme plate-forme matérielle et partagé le code source. Nous avons évalué la technique de saut de port UDP dans un banc d'essai réel. Nous avons mesuré son efficacité pour atténuer la phase de reconnaissance de l'attaque (c'est-à-dire le port-scanning ou balayage des ports). Nous avons défini un modèle probabiliste pour prédire l'efficacité de la MTD qui a corroboré les résultats empiriques. Ce chapitre a fourni deux MTDs dans des cas concrets d'utilisation de l'IoT et a illustré comment tirer parti des éléments de l'IANVS pour faciliter leur instanciation. Par exemple, en ne partant pas des principes fondamentaux de conception de l'MTDs et en réutilisant les éléments mis en œuvre.

Enfin, au Chapitre 7, nous avons présenté une technique MTD de couche physique du réseau. Contrairement aux techniques du chapitre précédent qui visent la connectivité et les attaques de bout en bout, celle-ci se concentre sur le premier «saut» (*hop*) du canal de communication. En effet, si un attaquant perturbe une seule couche du réseau sur le trajet d'une communication de bout en bout, aucune communication ni aucun échange d'informations n'est possible: le système IoT ne fournira pas son service. Dans ce chapitre, nous avons été motivés par le «cas d'utilisation de la cybermenace» que constituent les attaques de nœuds internes et le *brouillage* (*jamming*). Comme indiqué précédemment, le brouillage est une attaque très simple et efficace: en émettant un signal sur le canal, le flux d'informations du système peut être arrêté ou gravement perturbé. Les lois de l'électromagnétisme et le théorème de Shannon-Hartley garantissent le succès de l'attaquant s'il dispose d'une puissance de signal suffisante. De plus, nous avons supposé une attaque par brouillage d'un nœud interne qui a la connais-

sance des paramètres des techniques du AJ (Anti Brouillage) connus par le réseau. Notre solution pour ce cas a été une technique de AJ basée sur IANVS qui améliore la résilience au brouillage par des nœuds internes aux systèmes IoT et tire parti de la technique de modulation radio DSSS (étalement de spectre par séquence directe). Notre technique atténue de manière proactive les attaques de brouillage. La principale nouveauté de notre proposition réside dans l'utilisation des SS (séquences d'étalement) pour DSSS générés indépendamment avec CSPRNGs. Les jeux de séquences générés de cette manière n'étaient pas présents dans la littérature de WCS (systèmes de communication sans fil) et nous avons nommé cette famille «CSPR». Nous avons étudié les propriétés statistiques et probabilistes des grands ensembles de séquences CSPR et des ensembles de séquences uniformément aléatoires, et nous avons montré que les séquences CSPR ont des propriétés de CC (corrélations croisée) robustes. Cette étude fondamentale faisait défaut dans la littérature sur les WCS. Enfin, nous avons évalué la résilience au brouillage de notre proposition en utilisant un modèle mis en œuvre dans MATLAB. Nous avons exposé notre système à un brouilleur interne intelligent et validé que l'attaque était atténuée. La résilience AJ face à un brouilleur interne intelligent et la CC des séquences ont été analytiquement liées.

B

LATE SYNCHRONIZATION PROTOCOL: SYNTAX

B.1 LATE MESSAGE ENCODINGS

The protocol consists of two messages encoded with IETF's CBOR. COSE is used to cryptographically protect the second message. We define two new CBOR objects: *TIC Information* and *TOC Response*. Those objects are CBOR Maps which consist of *key-value* pairs of information. Additionally, to give semantic meaning to the objects without relying on external information we assign a CBOR Tag to each of the objects. CBOR Tag values range between ± 65536, and are registered in the Internet Assigned Numbers Authority. Tags in the 1-23 range take one byte when encoded -but all are allocated-; tags in the 24-255 range take two bytes: we chose values in this range.

B.1.1 *Message 1 - TIC Information*

The message will consist of a new CBOR MAP *TIC Information* as defined in Table B.1, we propose the CBOR Tag 59 to describe a *TIC Information* object.

About the nonce generation. The Nonce must be at least 64-bits and cryptographically secure randomness is needed, a pseudo-random number generator may be used if the seed has sufficient entropy, for details see [1].

The *Key-ID* is an opaque identifier of the key to be used by the server, it is the equivalent of the client's identity. The *Alg* field allows cryptoagility, some recommended algorithms are `HMAC w/SHA-256 truncated to 64 bits` (using a 256-bit pre-shared-key), `AES-CBC-MAC` or `AES-CMAC` (for both, 128-bit key will suffice). The client can explicitly request for a time server, e.g. in cases where the message is dealing

Table B.1: CBOR Map "TIC Information" object definition

Parameter name	**CBOR Key**	**Value Type**	**Description**
nonce	4	binary string	A random nonce
kid	5	binary string	Key-ID is an opaque value and identifies the cryptographic key to be used in the response
alg (optional)	6	int	Identifies the cryptographic algorithm to be used in the response
server (optional)	7	string	Identifies the intended Server for time synchronization (Absulute URI)

Table B.2: CBOR Map "TOC Information" object definition

Parameter name	CBOR Key	Value Type	Description
time	3	unsigned int	Time representation information
nonce	4	binary string	A random nonce

with intermediate nodes. In Listing B.1 we show a TIC Information object in human-readable CBOR diagnostic notation.

Listing B.1: *TIC Information* in CBOR diagnostic notation

```
{ nonce:h'73616E206C6F7265',
  kid  :h'0001',
  alg  :4/*HMAC w/SHA-256 truncated to 64 bits*/}
```

The binary representation of the same *TIC Information* object is found in Listing B.2 the size of the message is 19 bytes.

Listing B.2: *TIC Information* CBOR object (19 Bytes)

```
D83B                  # tag(59) (TIC Info.)
  A3                  # map(3)

    04                # unsigned(4) (=nonce)
    48                # bytes(8)
    73616E206C6F7265  # Nonce Value

    05                # unsigned(5) (=kid)
    42                # bytes(2)
    0001              # Key-ID Value

    06                # unsigned(6)  (=alg)
    04                # unsigned(4)
```

B.1.2 *Message 2 - TOC Response*

The message consists of a new CBOR MAP *TOC Information* as defined in Table B.2, we propose the CBOR Tag 60 to describe a *TOC Information* object. The TOC Information object contains the representation of the *time* from the server and a *nonce*.

The *TOC Response* object needs to include a Message Authentication Code, this security service will be provided by COSE using a `COSE_Mac0` object. A TOC Response authenticated and wrapped in COSE can be found in Listing B.3 in CBOR diagnostic notation.

Listing B.3: *TOC Information* in CBOR diagnostic notation

```
{protected: { /* Protected header of COSE_Mac0 Object*/
     kid: h'0001',
     alg: 4 /* HMAC w/ SHA-256 truncated to 64 bits */
   },
 payload   : { /* TOC Response CBOR MAP*/
     time  : 1477307841,
     nonce : h'73616E206C6F7265'
  },
 tag       : h'36f5afafobab5d43' /* MAC Code*/}
```

C

A PHY-LAYER MTD: UNBREAKABLE SS AND ASYMPTOTIC AJ EVALUATION

C.1 UNBREAKABLE SPREADING SEQUENCES

The smart jammer's hypothesis *the jammer can not gain knowledge of the SS of another node of the system* limits the capabilities of the jammer. This assumption is motivated by the dynamic nature of the attacker-system relationship, and by imposing it, we are simplifying this dynamism. If we assume that the jammer can gain knowledge of an unknown SS, we have to estimate a Δtime= t_{attack_ss} needed for it. From an attacked node perspective, once the attacker knows the sequence, we are *defeated* (BER = 0.5 for JSR$\approx$0). If we use the MTD aspect of our proposed system, we can mitigate this attack: the MTD system has to change the SS of the nodes with a periodicity $T_{ss\_movement} < t_{attack\_ss}$. We simplify the attacker model with two possible states: either knows the SS of a node ($t \geq t_{attack\_ss}$) or does not ($t < t_{attack\_ss}$). As stated before, in our proposal breaking one SS does not imply breaking the other SSs. This is due to the independence in the generation of the CSPR sequences, knowing one sequence does not leak any information about other SS (unlike other PR proposals in the literature). From a system perspective, it is very relevant to study the case in which the attacker has gained knowledge of *one node* SS (or equivalently, using a uniformly random SS), and measure the impact over the BER of the system excluding the compromised node, as done in Chapter 7.6.4.

C.2 ASYMPTOTIC AJ EVALUATION

A smart jammer affects each node differently, and the way it affects each node is related to the NCC between the jammer and node sequences (Eq. 7.5). We use the results from Chapter 7.6.3 and Chapter 7.6.4 to calculate percentiles for nodes with $BER \leq 0.1$ for the Smart Jammer (SJ), and BBN jammer scenarios. We present the results in

Fig. C.1. The ECDF of the NCC entirely determines the SJ case. Some counter-intuitive results happen on the P_{10} for $L \in \{2^9, 768, 2^{10}\}$ due to the discrete nature of the NCC values. Aside from that, for a given percentile we observe that if we double the length L we gain $\approx +1.5dB$ ($\sqrt{2}$ in linear terms) in jammer resilience.

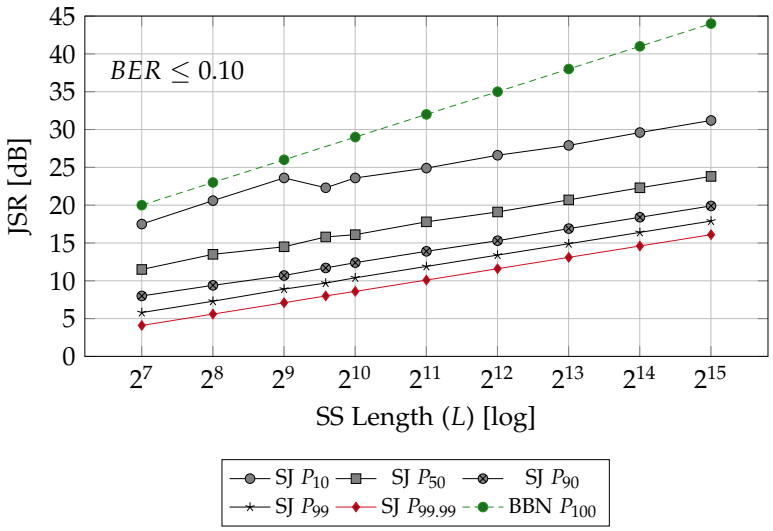


Figure C.1: Percentiles of nodes with $BER \leq 0.1$ for a Jammer with power JSR (dB), as a function of the SS length L.

BIBLIOGRAPHY

[1] Donald E. Eastlake 3rd, Steve Crocker, and Jeffrey I. Schiller. *Randomness Requirements for Security*. RFC 4086. June 2005. DOI: 10.17487/RFC4086. URL: https://rfc-editor.org/rfc/rfc4086.txt.

[2] Ali Abbasi, Jos Wetzels, Thorsten Holz, and Sandro Etalle. "Challenges in Designing Exploit Mitigations for Deeply Embedded Systems." In: *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE. 2019. DOI: 10.1109/EuroSP.2019.00013.

[3] A Akhavan et al. "Cryptanalysis of "an improvement over an image encryption method based on total shuffling"." In: *Optics Comm.* (2015).

[4] M. Albanese, A. De Benedictis, S. Jajodia, and K. Sun. "A moving target defense mechanism for MANETs based on identity virtualization." In: *2013 IEEE Conference on Communications and Network Security, CNS 2013* (2013). cited By 25, pp. 278–286. DOI: 10.1109/CNS.2013.6682717.

[5] Roger Alexander, Anders Brandt, JP Vasseur, Jonathan Hui, Kris Pister, Pascal Thubert, P Levis, Rene Struik, Richard Kelsey, and Tim Winter. *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*. RFC 6550. Mar. 2012. DOI: 10.17487/RFC6550. URL: https://rfc-editor.org/rfc/rfc6550.txt.

[6] F. Almoualem, P. Satam, J.-G. Ki, and S. Hariri. "SDR-Based Resilient Wireless Communications." In: *Proceedings - 2017 IEEE International Conference on Cloud and Autonomic Computing, ICCAC 2017* (2017). DOI: 10.1109/ICCAC.2017.18.

[7] Omar Alrawi et al. "Sok: Security evaluation of home-based iot deployments." In: *2019 IEEE Symp. on Security and Privacy*. IEEE. 2019.

[8] SaiDhiraj Amuru et al. "Optimal jamming against digital modulation." In: *IEEE Transactions on Information Forensics and Security* 10 (2015).

[9] Ioannis Andrea, Chrysostomos Chrysostomou, and George Hadjichristofi. "Internet of Things: Security vulnerabilities and challenges." In: *2015 IEEE Symposium on Computers and Communication (ISCC)*. IEEE. 2015, pp. 180–187.

[10] Kevin Andrea, Arda Gumusalan, Robert Simon, and Hugh Harney. "The design and implementation of a multicast address moving target defensive system for internet-of-things applications." In: *MILCOM 2017-2017 IEEE Military Communications Conference (MILCOM)*. IEEE. 2017. DOI: 10.1109/MILCOM.2017.8170748.

[11] Emekcan Aras et al. "Exploring the security vulnerabilities of LoRa." In: *2017 3rd IEEE International Conference on Cybernetics*. IEEE. 2017.

[12] Gal Badishi et al. "Keeping denial-of-service attackers in the dark." In: *International Symposium on Distributed Computing*. Springer. 2005.

[13] Subhadeep Banik et al. "Towards low energy stream ciphers." In: *IACR Transactions on Symmetric Cryptology* (2018), pp. 1–19.

[14] Elaine Barker et al. "Recommendation for Random Number Generation Using Deterministic Random Bit Generators." In: *NIST SP* 800-90A R1 (2015).

[15] Elaine B Barker, Don Johnson, and Miles E Smid. "SP 800-56A." In: *Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography (Revised), National Institute of Standards & Technology, Gaithersburg, MD* (2007).

[16] Lawrence E Bassham et al. "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications." In: *NIST SP* 800-22 Rev 1a (2010).

[17] E. Battista, V. Casola, A. Mazzeo, and N. Mazzocca. "SIREN: A feasible moving target defence framework for securing resource-constrained embedded nodes." In: *International Journal of Critical Computer-Based Systems* 4.4 (2013). cited By 3, pp. 374–392. DOI: 10.1504/IJCCBS.2013.059053.

[18] R. K. Bauer, T. A. Berson, and R. J. Feiertag. "A key distribution protocol using event markers." In: *ACM Transactions on Computer Systems* 1.3 (1983), pp. 249–255. ISSN: 07342071. DOI: 10.1145/357369.357373.

[19] Phongnawin Benprom et al. "Analysis of convolutional coded direct sequence spread spectrum CDMA system with a BPSK jamming signal." In: *8th ECTI*. IEEE. 2011.

[20] Chafika Benzaid et al. "An Enhanced Secure Pairwise Broadcast Time Synchronization Protocol in Wireless Sensor Networks." In: *Euromicro Int. Conf. on Parallel, Distributed, and Network-Based Processing* (2014). DOI: 10.1109/PDP.2014.114.

[21] Daniel J Bernstein. "ChaCha, a variant of Salsa20." In: *SASC*. 2008.

[22] Sandeep Bhatkar et al. "Address Obfuscation: An Efficient Approach to Combat a Broad Range of Memory Error Exploits." In: *USENIX*. 2003.

[23] Sharon Boeyen, Stefan Santesson, Tim Polk, Russ Housley, Stephen Farrell, and Dave Cooper. *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. RFC 5280. May 2008. DOI: 10.17487/RFC5280. URL: https://rfc-editor.org/rfc/rfc5280.txt.

[24] Carsten Bormann, Mehmet Ersue, and Ari Keränen. *Terminology for Constrained-Node Networks*. RFC 7228. May 2014. DOI: 10.17487/RFC7228. URL: https://rfc-editor.org/rfc/rfc7228.txt.

[25] Carsten Bormann and Paul E. Hoffman. *Concise Binary Object Representation (CBOR)*. RFC 7049. Oct. 2013. DOI: 10.17487/RFC7049. URL: https://rfc-editor.org/rfc/rfc7049.txt.

[26] Pearl Brereton, Barbara A Kitchenham, David Budgen, Mark Turner, and Mohamed Khalil. "Lessons from applying the systematic literature review process within the software engineering domain." In: *Journal of systems and software* 80.4 (2007), pp. 571–583.

[27] Jack Burbank, William Kasch, and Professor David L. Mills. *Network Time Protocol Version 4: Protocol and Algorithms Specification*. RFC 5905. 2010. DOI: 10.17487/rfc5905.

[28] Gui-lin Cai et al. "Moving target defense: state of the art and characteristics." In: *Frontiers of Information Technology & Electronic Engineering* 17.11 (2016).

[29] Valentina Casola, Alessandra De Benedictis, and Massimiliano Albanese. "A moving target defense approach for protecting resource-constrained distributed devices." In: *2013 IEEE 14th International Conference on Information Reuse & Integration (IRI)*. IEEE. 2013, pp. 22–29. DOI: 10.1109/IRI.2013.6642449.

[30] Valentina Casola, Alessandra De Benedictis, and Massimiliano Albanese. "A multi-layer moving target defense approach for protecting resource-constrained distributed devices." In: *Integration of Reusable Systems*. Springer, 2014, pp. 299–324. DOI: 10.1007/978-3-319-04717-1_1410.1007/978-3-319-04717-1_14.

[31] Zhixiong Chen, Xiaoni Du, and Guozhen Xiao. "Sequences related to Legendre/Jacobi sequences." In: *Information Sciences* 177 (2007).

[32] Xia Cheng, Junyang Shi, and Mo Sha. "Cracking the Channel Hopping Sequences in IEEE 802.15.4e-Based Industrial TSCH Networks." In: *4th IoTDI 2019* (2019).

[33] Jin-Hee Cho, Dilli P Sharma, Hooman Alavizadeh, Seunghyun Yoon, Noam Ben-Asher, Terrence J Moore, Dong Seong Kim, Hyuk Lim, and Frederica F Nelson. "Toward Proactive, Adaptive Defense: A Survey on Moving Target Defense." In: *IEEE Communications Surveys & Tutorials* (2020).

[34] Fred Chong, Ruby Lee, A Acquisti, W Horne, C Palmer, A Ghosh, D Pendarakis, W Sanders, E Fleischman, H Teufel III, et al. "National cyber leap year summit 2009: Co-chairs' report." In: *NITRD Program* (2009). URL: https://www.nitrd.gov/nitrdgroups/images/b/bd/National%5C_Cyber%5C_Leap%5C_Year%5C_Summit%20%5C_2009%5C_CoChairs%5C_Report.pdf.

[35] C. J F Cremers et al. "Injective synchronisation: An extension of the authentication hierarchy." In: *Theoretical Computer Science* 1-2 (2006). ISSN: 03043975. DOI: 10.1016/j.tcs.2006.08.034.

[36] Casimir Joseph Franciscus Cremers. "The Scyther Tool: Automatic Verification of Security Protocols." In: *Computer Aided Verification* 5423 (2008), pp. 414–418. ISSN: 0302-9743. DOI: 10.1007/978-3-540-70545-1_38.

[37] Flaviu Cristian. "Probabilistic clock synchronization." In: *Distributed Computing* 3.3 (1989), pp. 146–158. ISSN: 01782770. DOI: 10.1007/BF01784024.

[38] Ivan Bjerre Damgård. "On the randomness of Legendre and Jacobi sequences." In: *Conf. on the Theory and App. of Cryptography*. Springer. 1988.

[39] Fabrizio De Santis, Andreas Schauer, and Georg Sigl. "ChaCha20-Poly1305 authenticated encryption for high-speed embedded IoT applications." In: *Proceedings of the Conference on Design, Automation & Test in Europe*. European Design and Automation Association. 2017, pp. 692–697.

[40] Dr. Steve E. Deering and Bob Hinden. *Internet Protocol, Version 6 (IPv6) Specification*. RFC 8200. July 2017. DOI: 10.17487/RFC8200. URL: https://rfc-editor.org/rfc/rfc8200.txt.

[41] Michael Denzel and Mark Dermot Ryan. "Malware Tolerant (Mesh-) Networks." In: *International Conference on Cryptology and Network Security*. Springer. 2018, pp. 133–153. DOI: 10.1007/978-3-030-00434-7_7.

[42] Jyoti Deogirikar and Amarsinh Vidhate. "Security attacks in IoT: A survey." In: *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*. IEEE. 2017, pp. 32–37.

[43] Jyoti Deogirikar and Amarsinh Vidhate. "Security attacks in IoT: A survey." In: *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*. IEEE. 2017, pp. 32–37.

[44] D. Dolev and a. C. Yao. "On the security of public key protocols." In: *22nd Annual Symposium on Foundations of Computer Science* M (1981). ISSN: 0272-5428. DOI: 10.1109/SFCS.1981.32.

[45] Jessye Dos Santos, Christine Hennebert, JC Fonbonne, and Cédric Lauradoux. "Ephemeral: Lightweight pseudonyms for 6LoWPAN MAC addresses." In: *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE. 2016, pp. 1–6. DOI: 10.1109/PIMRC.2016.7794800.

[46] Stefan Dziembowski et al. "Towards sound fresh re-keying with hard (physical) learning problems." In: *Annual International Cryptology Conference*. Springer. 2016, pp. 272–301.

[47] AbdelRahman Eldosouky and Walid Saad. "On the cybersecurity of m-health iot systems with led bitslice implementation." In: *2018 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE. 2018, pp. 1–6. DOI: 10.1109/ICCE.2018.8326298.

[48] Pasi Eronen, Yoav Nir, Paul E. Hoffman, and Charlie Kaufman. *Internet Key Exchange Protocol Version 2 (IKEv2)*. RFC 5996. Sept. 2010. DOI: 10.17487/RFC5996. URL: https://rfc-editor.org/rfc/rfc5996.txt.

[49] Roy T Fielding. *Architectural styles and the design of network-based software architectures*. Vol. 7. University of California, Irvine Irvine, 2000.

[50] Paul G Flikkema. "Spread-spectrum techniques for wireless communication." In: *IEEE Signal Processing Magazine* 14.3 (1997).

[51] Daniel Fox Franke, Dieter Sibold, and Kristof Teichel. *Network Time Security for the Network Time Protocol*. Internet-Draft draft-ietf-ntp-using-nts-for-ntp-11. Work in Progress. IETF, Mar. 2018. 29 pp.

[52] Ala Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari, and Moussa Ayyash. "Internet of things: A survey on enabling technologies, protocols, and applications." In: *IEEE communications surveys & tutorials* 17.4 (2015), pp. 2347–2376.

[53] Saurabh Ganeriwal, Christina Pöpper, Srdjan Čapkun, and Mani B. Srivastava. "Secure Time Synchronization in Sensor Networks." In: *ACM Transactions on Information and System Security* 11.4 (2008), pp. 1–35. ISSN: 10949224. DOI: 10.1145/1380564.1380571.

[54] Oscar Garcia-Morchon, Sye Keoh, Sandeep Kumar, Rene Hummen, and Rene Struik. *Security Considerations in the IP-based Internet of Things*. Internet-Draft draft-garcia-core-security-00. Work in Progress. Internet Engineering Task Force. URL: https:

//datatracker.ietf.org/doc/html/draft-garcia-core-security-00.

[55] Oscar Garcia-Morchon, Sandeep Kumar, and Mohit Sethi. *Internet of Things (IoT) Security: State of the Art and Challenges*. RFC 8576. Apr. 2019. DOI: 10.17487/RFC8576. URL: https://rfc-editor.org/rfc/rfc8576.txt.

[56] Mengmeng Ge, Jin B Hong, Walter Guttmann, and Dong Seong Kim. "A framework for automating security analysis of the internet of things." In: *Journal of Network and Computer Applications* 83 (2017), pp. 12–27. DOI: 10.1016/j.jnca.2017.01.033.

[57] Mengmeng Ge, Jin B Hong, Simon Enoch Yusuf, and Dong Seong Kim. "Proactive defense mechanisms for the software-defined Internet of Things with non-patchable vulnerabilities." In: *Future Generation Computer Systems* 78 (2018). DOI: 10.1016/j.future.2017.07.008.

[58] M Golay. "The merit factor of long low autocorrelation binary sequences (Corresp.)" In: *IEEE Transactions on Information Theory* 28 (1982).

[59] Solomon Golomb. *Shift Register Sequences (3rd Rev. Edition)*. World Scientific, 2017.

[60] Solomon W Golomb and Guang Gong. *Signal design for good correlation: for wireless communication, cryptography, and radar*. Cambridge Univ. Press, 2005.

[61] Kanika Grover, Alvin Lim, and Qing Yang. "Jamming and anti–jamming techniques in wireless networks: a survey." In: *International Journal of Ad Hoc and Ubiquitous Computing* 17.4 (2014), pp. 197–215.

[62] Longhua Guo et al. "A Lightweight Secure Time Synchronization Mechanism for ISO/IEC/IEEE 21451 Sensor Networks." In: *IEEE Precision Clock Synchronization for Measurement, Control, and Communication*. 2015. ISBN: 9781467375955.

[63] J. Habibi, A. Gupta, S. Carlsony, A. Panicker, and E. Bertino. "MAVR: Code Reuse Stealthy Attacks and Mitigation on Unmanned Aerial Vehicles." In: *Proceedings - International Conference on Distributed Computing Systems* 2015-July (2015). cited By 13, pp. 642–652. DOI: 10.1109/ICDCS.2015.71.

[64] Amal O Hamada, Mohamed Azab, and Amr Mokhtar. "Honeypot-like Moving-target Defense for secure IoT Operation." In: *IEEE 9th Annual Inf. Technology, Electronics and Mobile Comm. Conf. (IEMCON)*. IEEE. 2018. DOI: 10.1109/IEMCON.2018.8614925.

[65] Frank Hermanns. *Protected spread spectrum signal transmission system for multiple-access messages uses pseudo-random sequences as expansion codes*. German Patent DE102004013884B4. 2004.

[66] Frank Hermanns. "Secure and Robust Tactical Communications Based on Code-Hopping CDMA (CH-CDMA)." In: *NATO/OTAN, Germany, Report No. RTO-MP-IST-083* (2008).

[67] Thomas Hobson, Hamed Okhravi, David Bigelow, Robert Rudd, and William Streilein. "On the challenges of effective movement." In: *Proceedings of the First ACM Workshop on Moving Target Defense*. ACM. 2014, pp. 41–50.

[68] Jin B Hong, Simon Yusuf Enoch, Dong Seong Kim, Armstrong Nhlabatsi, Noora Fetais, and Khaled M Khan. "Dynamic security metrics for measuring the effectiveness of moving target defense techniques." In: *Computers & Security* (2018).

[69] Shohreh Hosseinzadeh, Sami Hyrynsalmi, and Ville Leppänen. "Chapter 14 - Obfuscation and diversification for securing the Internet of Things (IoT)." In: *Internet of Things*. Elsevier, 2016, pp. 259–274. DOI: 10.1016/B978-0-12-805395-9.00014-9.

[70] Shohreh Hosseinzadeh, Sampsa Rauti, Sami Hyrynsalmi, and Ville Leppänen. "Security in the internet of things through obfuscation and diversification." In: *2015 International Conference on Computing, Communication and Security (ICCCS)*. IEEE. 2015, pp. 1–5. DOI: 10.1109/CCCS.2015.7374189.

[71] Shohreh Hosseinzadeh, Sampsa Rauti, Samuel Laurén, Jari-Matti Mäkelä, Johannes Holvitie, Sami Hyrynsalmi, and Ville Leppänen. "Diversification and obfuscation techniques for software security: A systematic literature review." In: *Information and Software Technology* 104 (2018), pp. 72–93.

[72] IEEE. "802.15.4e Standard for Local and metropolitan area networks. Part15.4: Low-Rate Wireless Personal Area Networks(LR-WPANs)A.1:MAC sublayer." In: (2012).

[73] IETF. *Authentication and Authorization for Constrained Environments (ace)*. 2020. URL: https://datatracker.ietf.org/wg/ace/about/.

[74] Gene Itkis. "Intrusion-resilient signatures: generic constructions, or defeating strong adversary with minimal assumptions." In: *International Conference on Security in Communication Networks*. Springer. 2002.

[75] Sushil Jajodia et al. *Moving target defense: creating asymmetric uncertainty for cyber threats*. Vol. 54. Springer Science & Business Media, 2011.

[76] Sushil Jajodia et al. *Moving Target Defense II: Application of Game Theory and Adversarial Modeling*. Vol. 100. Springer Science & Business Media, 2012.

[77] Sushil Jajodia and Kun Sun. "MTD 2014: First ACM Workshop on Moving Target Defense." In: *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. CCS '14. Scottsdale, Arizona, USA: ACM, 2014, pp. 1550–1551. ISBN: 978-1-4503-2957-6. DOI: 10.1145/2660267.2660385. URL: http://doi.acm.org/10.1145/2660267.2660385.

[78] Cullen Jennings, Zach Shelby, Jari Arkko, Ari Keränen, and Carsten Bormann. *Sensor Measurement Lists (SenML)*. RFC 8428. Aug. 2018. DOI: 10.17487/RFC8428. URL: https://rfc-editor.org/rfc/rfc8428.txt.

[79] Tao Jin et al. "Zero pre-shared secret key establishment in the presence of jammers." In: *ACM MobiHoc '19*. 2009.

[80] Aljosha Judmayer, Johanna Ullrich, Georg Merzdovnik, Artemios G Voyiatzis, and Edgar Weippl. "Lightweight address hopping for defending the IPv6 IoT." In: *Proceedings of the 12th International Conference on Availability, Reliability and Security*. ACM, 2017, pp. 1–10. DOI: 10.1145/3098954.3098975.

[81] Taeho Kang et al. "A survey of security mechanisms with direct sequence spread spectrum signals." In: *Journal of Comp. Science and Eng.* (2013).

[82] Stephen Kent. *IP Authentication Header*. RFC 4302. Dec. 2005. DOI: 10.17487/RFC4302. URL: https://rfc-editor.org/rfc/rfc4302.txt.

[83] Stephen Kent. *IP Encapsulating Security Payload (ESP)*. RFC 4303. Dec. 2005. DOI: 10.17487/RFC4303. URL: https://rfc-editor.org/rfc/rfc4303.txt.

[84] Dorene Kewley, Russ Fink, John Lowry, and Mike Dean. "Dynamic approaches to thwart adversary intelligence gathering." In: *Proceedings DARPA Information Survivability Conference and Exposition II. DISCEX'01*. Vol. 1. IEEE. 2001, pp. 176–185.

[85] Athar Ali Khan et al. "Cognitive-radio-based internet of things: Applications, architectures, spectrum related functionalities, and future research directions." In: *IEEE wireless communications* 24.3 (2017).

[86] Barbara Ann Kitchenham, David Budgen, and Pearl Brereton. *Evidence-based software engineering and systematic reviews*. Vol. 4. CRC press, 2015.

[87] Tero Kivinen. *Minimal Internet Key Exchange Version 2 (IKEv2) Initiator Implementation*. RFC 7815. Mar. 2016. DOI: 10.17487/RFC7815. URL: https://rfc-editor.org/rfc/rfc7815.txt.

[88] Matthew Knight and Balint Seeber. "Decoding LoRa: Realizing a modern LPWAN with SDR." In: *Proceedings of the GNU Radio Conference*. 2016.

[89] Neal Koblitz and Alfred J. Menezes. "Another look at "provable security"." In: *Journal of Cryptology* 20.1 (2007), pp. 3–37. ISSN: 09332790. DOI: 10.1007/s00145-005-0432-z.

[90] Tohru Kohda et al. "Statistics of chaotic binary sequences." In: *IEEE Transactions on information theory* (1997).

[91] Lauri Koivunen, Sampsa Rauti, and Ville Leppänen. "Applying internal interface diversification to IoT operating systems." In: *2016 International Conference on Software Security and Assurance (ICSSA)*. IEEE. 2016. DOI: 10.1109/ICSSA.2016.7.

[92] Constantinos Kolias, Georgios Kambourakis, Angelos Stavrou, and Jeffrey Voas. "DDoS in the IoT: Mirai and other botnets." In: *Computer* 50.7 (2017), pp. 80–84.

[93] A.I. Kouachi, S. Sahraoui, and A. Bachir. "Per Packet Flow Anonymization in 6LoWPAN IoT Networks." In: *2018 International Conference on Wireless Networks and Mobile Communications, WINCOM 2018* (2019). DOI: 10.1109/WINCOM.2018.8629719.

[94] Thorsten Kramp, Rob Van Kranenburg, and Sebastian Lange. "Introduction to the Internet of Things." In: *Enabling Things to Talk*. Springer, Berlin, Heidelberg, 2013, pp. 1–10.

[95] Dr. Hugo Krawczyk and Pasi Eronen. *HMAC-based Extract-and-Expand Key Derivation Function (HKDF)*. RFC 5869. May 2010. DOI: 10.17487/RFC5869. URL: https://rfc-editor.org/rfc/rfc5869.txt.

[96] Hugo Krawczyk. "SIGMA: The 'SIGn-and-MAc'approach to authenticated Diffie-Hellman and its use in the IKE protocols." In: *Annual International Cryptology Conference*. Springer. 2003, pp. 400–425.

[97] Lauwerens Kuipers et al. "Unifom Distribution in Topological Groups: Convolution of sequences." In: *Uniform distribution of sequences*. John Wiley & Sons, 1974. Chap. 4.

[98] Adam Langley et al. *ChaCha20-Poly1305 Cipher Suites for Transport Layer Security (TLS)*. RFC 7905. June 2016. DOI: 10.17487/RFC7905.

[99] Henry CJ Lee and Vrizlynn LL Thing. "Port hopping for resilient networks." In: *IEEE 60th Vehicular Technology Conference.* IEEE. 2004.

[100] Cheng Lei, Hong-Qi Zhang, Jing-Lei Tan, Yu-Chen Zhang, and Xiao-Hu Liu. "Moving target defense techniques: A survey." In: *Security and Communication Networks* 2018 (2018).

[101] Martin Lévesque and David Tipper. "A Survey of Clock Synchronization Over Packet-Switched Networks." In: *IEEE Communications Surveys and Tutorials* 18.4 (2016), pp. 2926–2947. ISSN: 1553877X. DOI: 10.1109/COMST.2016.2590438.

[102] Chengqing Li. "Cracking a hierarchical chaotic image encryption algorithm based on permutation." In: *Signal Processing* 118 (2016), pp. 203–210.

[103] Lulu Liang et al. "A denial of service attack method for an iot system." In: *2016 8th International Conference on Information Technology in Medicine and Education (ITME)*. IEEE. 2016.

[104] Rafael Lopez and Dan Garcia-Carrillo. *EAP-based Authentication Service for CoAP*. Internet-Draft draft-marin-ace-wg-coap-eap-06. Work in Progress. Internet Engineering Task Force, Oct. 2017. 21 pp. URL: https://datatracker.ietf.org/doc/html/draft-marin-ace-wg-coap-eap-06.

[105] *LoPy4: a quadruple bearer MicroPython enabled development board*. 2020. URL: https://pycom.io/product/lopy4/.

[106] G. Lowe. "A hierarchy of authentication specifications." In: *Proceedings 10th Computer Security Foundations Workshop* (1997), pp. 31–43. ISSN: 1063-6900. DOI: 10.1109/CSFW.1997.596782.

[107] Yue-Bin Luo et al. "Effectiveness of port hopping as a moving target defense." In: *7th International Conf. on Security Tech.* IEEE. 2014.

[108] Kaleel Mahmood and Devu Manikantan Shila. "Moving target defense for Internet of Things using context aware code partitioning and code diversification." In: *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*. IEEE. 2016, pp. 329–330. DOI: 10.1109/WF-IoT.2016.7845457.

[109] Petteri Mäki, Sampsa Rauti, Shohreh Hosseinzadeh, Lauri Koivunen, and Ville Leppänen. "Interface diversification in IoT operating systems." In: *The 9th International Conference on Utility and Cloud Computing*. ACM. 2016. DOI: 10.1145/2996890.3007877.

[110] A. Marttinen, A.M. Wyglinski, and R. Jantti. "Moving-target defense mechanisms against source-selective jamming attacks in tactical cognitive radio MANETs." In: *2014 IEEE Conference on Communications and Network Security, CNS 2014* (2014). cited By 3, pp. 14–20. DOI: 10.1109/CNS.2014.6997460.

[111] James Massey. "Shift-register synthesis and BCH decoding." In: *IEEE Transactions on Information Theory* (1969).

[112] Gianluca Mazzini et al. "Chaotic complex spreading sequences for asynchronous DS-CDMA. I. System modeling and results." In: *IEEE Trans. on Circuits and Systems* (1997).

[113] David McGrew and Daniel Bailey. *AES-CCM Cipher Suites for Transport Layer Security (TLS)*. RFC 6655. July 2012. DOI: 10.17487/RFC6655. URL: https://rfc-editor.org/rfc/rfc6655.txt.

[114] David McGrew, Daniel Bailey, Matthew Campagna, and Robert Dugal. *AES-CCM Elliptic Curve Cryptography (ECC) Cipher Suites for TLS*. RFC 7251. June 2014. DOI: 10.17487/RFC7251. URL: https://rfc-editor.org/rfc/rfc7251.txt.

[115] Carlo Maria Medaglia and Alexandru Serbanati. "An overview of privacy and security issues in the internet of things." In: *The internet of things*. Springer, 2010, pp. 389–395.

[116] Francesca Meneghello, Matteo Calore, Daniel Zucchetto, Michele Polese, and Andrea Zanella. "IoT: Internet of Threats? A survey of practical security vulnerabilities in real IoT devices." In: *IEEE Internet of Things Journal* 6.5 (2019), pp. 8182–8201.

[117] Daniel Migault, Tobias Guggemos, Carsten Bormann, and David Schinazi. *ESP Header Compression and Diet-ESP*. Internet-Draft draft-mglt-ipsecme-diet-esp-07. Work in Progress. Internet Engineering Task Force, Mar. 2019. 47 pp. URL: https://datatracker.ietf.org/doc/html/draft-mglt-ipsecme-diet-esp-07.

[118] Tal Mizrahi. *Security Requirements of Time Protocols in Packet Switched Networks*. RFC 7384. 2014. DOI: 10.17487/rfc7384.

[119] Gabriel Montenegro, Jonathan Hui, David Culler, and Nandakishore Kushalnagar. *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*. RFC 4944. Sept. 2007. DOI: 10.17487/RFC4944. URL: https://rfc-editor.org/rfc/rfc4944.txt.

[120] Brice Morin et al. "Engineering Software Diversity: A Model-Based Approach to Systematically Diversify Communications." In: *Proceedings of the 21th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems*. MODELS '18. ACM. Copenhagen, Denmark, 2018. ISBN: 9781450349499. DOI: 10.1145/3239372.3239393.

[121] Bassam Moussa, Mourad Debbabi, and Chadi Assi. "Security Assessment of Time Synchronization Mechanisms for the Smart Grid." In: *IEEE Communications Surveys and Tutorials* 18.3 (2016), pp. 1952–1973. ISSN: 1553877X. DOI: 10.1109/COMST.2016.2525014.

[122] Renzo E Navas, Manuel Lagos, et al. "Nonce-based authenticated key establishment over OAuth 2.0 IoT proof-of-possession architecture." In: *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*. IEEE. 2016.

[123] Renzo E. Navas, Håkon Sandaker, Frédéric Cuppens, Nora Cuppens, Laurent Toutain, and Georgios Z. Papadopoulos. "IANVS: A Moving Target Defense Framework for a Resilient Internet of Things." In: *2020 IEEE Symposium on Computers and Communications (ISCC)* (Rennes, France). IEEE, July 2020, pp. 1–6. DOI: 10.1109/ISCC50000.2020.9219728.

[124] Mukrimah Nawir, Amiza Amir, Naimah Yaakob, and Ong Bi Lynn. "Internet of Things (IoT): Taxonomy of security attacks." In: *2016 3rd International Conference on Electronic Design (ICED)*. IEEE. 2016, pp. 321–326.

[125] Yoav Nir. *ChaCha20, Poly1305, and Their Use in the Internet Key Exchange Protocol (IKE) and IPsec*. RFC 7634. Aug. 2015.

[126] Francesca Nizzi, Tommaso Pecorella, Flavio Esposito, Laura Pierucci, and Romano Fantacci. "IoT Security via Address Shuffling: The Easy Way." In: *IEEE Internet of Things Journal* 6.2 (2019), pp. 3764–3774. DOI: 10.1109/JIOT.2019.2892003.

[127] Hamed Okhravi et al. *Survey of Cyber Moving Target Techniques*. Tech. rep. MIT Lincoln Laboratory Lexington United States, 2013.

[128] Hamed Okhravi, Thomas Hobson, David Bigelow, and William Streilein. "Finding focus in the blur of moving-target techniques." In: *IEEE Security & Privacy* 12.2 (2013), pp. 16–26.

[129] Jesus Pacheco, Cihan Tunc, and Salim Hariri. "Design and evaluation of resilient infrastructures systems for smart cities." In: *2016 IEEE International Smart Cities Conference (ISC2)*. IEEE. 2016, pp. 1–6. DOI: 10.1109/ISC2.2016.7580756.

[130] Sergio Pastrana, Juan Tapiador, Guillermo Suarez-Tangil, and Pedro Peris-López. "Avrand: A software-based defense against code reuse attacks for avr embedded devices." In: *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*. Springer. 2016, pp. 58–77.

[131] Merly Annie Philip et al. "A survey on lightweight ciphers for IoT devices." In: *2017 International Conference on Technological Advancements in Power and Energy (TAP Energy)*. IEEE. 2017, pp. 1–4.

[132] Stjepan Picek, Erik Hemberg, and Una-May O'Reilly. "If You Can't Measure It, You Can't Improve It: Moving Target Defense Metrics." In: *Proceedings of the 2017 Workshop on Moving Target Defense*. 2017, pp. 115–118.

[133] Richard Poisel. "Jamming Techniques." In: *Modern Communications Jamming Principles and Techniques*. Artech House, 2011. Chap. 8.

[134] Pavan Pongle and Gurunath Chavan. "A survey: Attacks on RPL and 6LoWPAN in IoT." In: *2015 International conference on pervasive computing (ICPC)*. IEEE. 2015, pp. 1–6.

[135] Christina Pöpper. "Jamming-resistant Broadcast Communication without Shared Keys." In: *USENIX security Symposium* (2009).

[136] J. Postel. *User Datagram Protocol*. RFC 768. Aug. 1980. DOI: 10.17487/RFC0768. URL: https://rfc-editor.org/rfc/rfc768.txt.

[137] Tanner Preiss, Matthew Sherburne, Randy Marchany, and Joseph Tront. "Implementing dynamic address changes in contikios." In: *International Conference on Information Society (i-Society 2014)*. IEEE. 2014, pp. 222–227. DOI: 10.1109/i-Society.2014.7009047.

[138] Deepak Puthal, Surya Nepal, Rajiv Ranjan, and Jinjun Chen. "A dynamic key length based approach for real-time security verification of big sensing data stream." In: *International conference on web information systems engineering*. Springer. 2015, pp. 93–108. DOI: 10.1007/978-3-319-26187-4_7.

[139] Deepak Puthal, Surya Nepal, Rajiv Ranjan, and Jinjun Chen. "DLSeF: A Dynamic Key-Length-Based Efficient Real-Time Security Verification Model for Big Data Stream." In: *ACM Trans. Embed. Comput. Syst.* 16.2 (Dec. 2016). ISSN: 1539-9087. DOI: 10.1145/2937755.

[140] Sampsa Rauti, Lauri Koivunen, Petteri Mäki, Shohreh Hosseinzadeh, Samuel Laurén, Johannes Holvitie, and Ville Leppänen. "Internal Interface Diversification as a Security Measure in Sensor Networks." In: *Journal of Sensor and Actuator Networks* (2018). DOI: 10.3390/jsan7010012.

[141] Shahid Raza, Simon Duquennoy, and Göran Selander. *Compression of IPsec AH and ESP Headers for 6LoWPAN Networks*. Internet-Draft draft-raza-6lo-ipsec-04. Work in Progress. Internet Engineering Task Force, Mar. 2016. 11 pp. URL: https://datatracker.ietf.org/doc/html/draft-raza-6lo-ipsec-04.

[142] Eric Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.3*. RFC 8446. Aug. 2018. DOI: 10.17487/RFC8446. URL: https://rfc-editor.org/rfc/rfc8446.txt.

[143] Eric Rescorla, Richard Barnes, and Hannes Tschofenig. *Compact TLS 1.3*. Internet-Draft draft-ietf-tls-ctls-00. Work in Progress. Internet Engineering Task Force, Apr. 2020. 17 pp. URL: https://datatracker.ietf.org/doc/html/draft-ietf-tls-ctls-00.

[144] Eric Rescorla and Tim Dierks. *The Transport Layer Security (TLS) Protocol Version 1.2*. RFC 5246. Aug. 2008. DOI: 10.17487/RFC5246. URL: https://rfc-editor.org/rfc/rfc5246.txt.

[145] Eric Rescorla and Nagendra Modadugu. *Datagram Transport Layer Security Version 1.2*. RFC 6347. Jan. 2012. DOI: 10.17487/RFC6347. URL: https://rfc-editor.org/rfc/rfc6347.txt.

[146] Ron Ross, Victoria Pillitteri, Richard Graubart, Deborah Bodeau, and Rosalie McQuaid. *Developing Cyber Resilient Systems: A Systems Security Engineering Approach (SP 800-160 Vol. 2)*. Tech. rep. National Institute of Standards and Technology, 2019.

[147] Rainer A. Rueppel. *Analysis and Design of Stream Ciphers*. Springer-Verlag, 1986. ISBN: 978-3-642-82867-6. arXiv: arXiv:1011.1669v3.

[148] Anthony E Sale. "Lorenz and Colossus [military cryptography]." In: *Proceedings 13th IEEE Computer Security Foundations Workshop. CSFW-13*. IEEE. 2000, pp. 216–222.

[149] Håkon Sandaker and Renzo E. Navas. *IANVS: An MTD Framework for a Resilient IoT. Port-hopping code and data.* Version v1.0.0. May 2020. DOI: 10.5281/zenodo.3839686. URL: https://doi.org/10.5281/zenodo.3839686.

[150] Nico Saputro, Samet Tonyali, Abdullah Aydeger, Kemal Akkaya, Mohammad A Rahman, and Selcuk Uluagac. "A Review of Moving Target Defense Mechanisms for Internet of Things Applications." In: *Modeling and Design of Secure Internet of Things* (2020), pp. 563–614.

[151] Dilip V Sarwate and Michael B Pursley. "Crosscorrelation properties of pseudorandom and related sequences." In: *Proceedings of the IEEE* 68.5 (1980).

[152] Martin De Saulles. *IoT Statistics - Information Matters.* 2019. URL: https://informationmatters.net/internet-of-things-statistics/.

[153] Jim Schaad. *CBOR Object Signing and Encryption (COSE)*. RFC 8152. July 2017. DOI: 10.17487/RFC8152. URL: https://rfc-editor.org/rfc/rfc8152.txt.

[154] Hans Dieter Schotten and Hans Dieter Lüke. "On the search for low correlated binary sequences." In: *AEU-Inte. Journal of Electronics and Comm.* (2005).

[155] Ludwig Seitz, Göran Selander, Erik Wahlstroem, Samuel Erdtman, and Hannes Tschofenig. *Authentication and Authorization for Constrained Environments (ACE) using the OAuth 2.0 Framework (ACE-OAuth)*. Internet-Draft draft-ietf-ace-oauth-authz-35. Work in Progress. Internet Engineering Task Force, June 2020. 87 pp. URL: https://datatracker.ietf.org/doc/html/draft-ietf-ace-oauth-authz-35.

[156] Göran Selander et al. *Ephemeral Diffie-Hellman Over COSE (EDHOC)*. I-D draft-selander-lake-edhoc-00. Work in Progress. IETF, Nov. 2019. 74 pp.

[157] Göran Selander, John Preuß Mattsson, and Francesca Palombini. *Ephemeral Diffie-Hellman Over COSE (EDHOC)*. Internet-Draft draft-ietf-lake-edhoc-01. Work in Progress. Internet Engineering Task Force, Aug. 2020. 58 pp. URL: https://datatracker.ietf.org/doc/html/draft-ietf-lake-edhoc-01.

[158] Göran Selander, John Preuß Mattsson, Francesca Palombini, and Ludwig Seitz. *Object Security for Constrained RESTful Environments (OSCORE)*. RFC 8613. July 2019. DOI: 10.17487/RFC8613. URL: https://rfc-editor.org/rfc/rfc8613.txt.

[159] Jayasree Sengupta, Sushmita Ruj, and Sipra Das Bit. "A Comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT." In: *Journal of Network and Computer Applications* 149 (2020), p. 102481.

[160] Sailik Sengupta, Ankur Chowdhary, Abdulhakim Sabur, Adel Alshamrani, Dijiang Huang, and Subbarao Kambhampati. "A survey of moving target defenses for network security." In: *IEEE Communications Surveys & Tutorials* (2020).

[161] Claude Elwood Shannon. "A mathematical theory of communication." In: *Bell tech. journal* (1948).

[162] Zach Shelby. *Constrained RESTful Environments (CoRE) Link Format*. RFC 6690. Aug. 2012. DOI: 10.17487/RFC6690.

[163] Zach Shelby, Klaus Hartke, and Carsten Bormann. *The Constrained Application Protocol (CoAP)*. RFC 7252. June 2014. DOI: 10.17487/RFC7252. URL: https://rfc-editor.org/rfc/rfc7252.txt.

[164] Matthew Sherburne et al. "Implementing moving target IPv6 defense to secure 6LoWPAN in the internet of things and smart grid." In: *Proc. of the 9th Annual Cyber and Information Security Research Conf.* ACM. 2014.

[165] Matthew Sherburne, Randy Marchany, and Joseph Tront. "Implementing moving target ipv6 defense to secure 6lowpan in the internet of things and smart grid." In: *Proceedings of the 9th Annual Cyber and Information Security Research Conference*. ACM. 2014, pp. 37–40. DOI: 10.1145/2602087.2602107.

[166] V. M. Sidelnikov. "On mutual correlation of sequences." In: *Probl. Kybern.* 24 (1971).

[167] Marvin K Simon et al. *Spread spectrum communications handbook*. MG-Hill, 1997.

[168] Tianlong Song, Kai Zhou, and Tongtong Li. "CDMA system design and capacity analysis under disguised jamming." In: *IEEE Transactions on Information Forensics and Security* 11.11 (2016), pp. 2487–2498.

[169] Susanna Spinsante et al. "Binary De Bruijn sequences for DS-CDMA systems: analysis and results." In: *EURASIP Journal on Wireless Communications and Networking* (2011).

[170] Susanna Spinsante et al. "De Bruijn binary sequences and spread spectrum applications: A marriage possible?" In: *IEEE Aerospace and Electronic Systems Magazine* (2013).

[171] Spase Stojanovski et al. "Efficient attacks in industrial wireless sensor networks." In: *International Conf. on ICT Innovations*. Springer. 2014.

[172] Kun Sun et al. "TinySeRSync: secure and resilient time synchronization in wireless sensor networks." In: *Proceedings of the 13th ACM conference on Computer and communications security* (2006), p. 264. DOI: 10.1145/1180405.1180439.

[173] Joshua Taylor, Kara Zaffarano, Ben Koller, Charlie Bancroft, and Jason Syversen. "Automated effectiveness evaluation of moving target defenses: metrics for missions and attacks." In: *Proceedings of the 2016 ACM Workshop on Moving Target Defense*. 2016, pp. 129–134.

[174] Nikola Tesla. *System of signaling*. US Patent 725,605. Apr. 1903.

[175] Marco Tiloca et al. "DISH: DIstributed SHuffling against selective jamming attack in IEEE 802.15.4e TSCH networks." In: *ACM TOSN* (2018).

[176] Marco Tiloca, Domenico De Guglielmo, Gianluca Dini, and Giuseppe Anastasi. "SAD-SJ: A self-adaptive decentralized solution against Selective Jamming attack in Wireless Sensor Networks." In: *2013 IEEE 18th Conference on Emerging Technologies & Factory Automation (ETFA)*. IEEE. 2013, pp. 1–8.

[177] Matheus Torquato and Marco Vieira. "Moving Target Defense in Cloud Computing: A Systematic Mapping Study." In: *Computers & Security* (2020), p. 101742.

[178] Don Torrieri. "Cyber maneuvers and maneuver keys." In: *2014 IEEE Military Communications Conference*. IEEE. 2014, pp. 262–267.

[179] Don Torrieri, Sencun Zhu, and Sushil Jajodia. "Cyber maneuver against external adversaries and compromised nodes." In: *Moving Target Defense II*. Springer, 2013, pp. 87–96.

[180] Hannes Tschofenig and Thomas Fossati. *Transport Layer Security (TLS) / Datagram Transport Layer Security (DTLS) Profiles for the Internet of Things*. RFC 7925. July 2016. DOI: 10.17487/RFC7925. URL: https://rfc-editor.org/rfc/rfc7925.txt.

[181] CK Tse et al. "Chaos-based digital communication systems." In: *Operating Principles, Analysis Methods and Performance Evaluation* (2003).

[182] Howard G Tucker. "A generalization of the Glivenko-Cantelli theorem." In: *The Annals of Mathematical Statistics* 30.3 (1959), pp. 828–830.

[183] Anthony Van Herrewege and Ingrid Verbauwhede. "Software only, extremely compact, Keccak-based secure PRNG on ARM Cortex-M." In: *2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC)*. IEEE. 2014. DOI: 10.1145/2593069.2593218.

[184] Juan M Velazquez-Gutierrez and Cesar Vargas-Rosales. "Sequence sets in wireless communication systems: A survey." In: *IEEE Communications Surveys & Tutorials* 19.2 (2017).

[185] John Vollbrecht, James D. Carlson, Larry Blunk, Dr. Bernard D. Aboba, and Henrik Levkowetz. *Extensible Authentication Protocol (EAP)*. RFC 3748. June 2004. DOI: 10.17487/RFC3748. URL: https://rfc-editor.org/rfc/rfc3748.txt.

[186] Mališa Vučinić, Jonathan Simon, Kris Pister, and Michael Richardson. *Constrained Join Protocol (CoJP) for 6TiSCH*. Internet-Draft draft-ietf-6tisch-minimal-security-15. Work in Progress. Internet Engineering Task Force, Dec. 2019. 53 pp. URL: https://datatracker.ietf.org/doc/html/draft-ietf-6tisch-minimal-security-15.

[187] S. Vuppala, A. E. Mady, and A. Kuenzi. "Moving Target Defense Mechanism for Side-Channel Attacks." In: *IEEE Systems Journal* (2019), pp. 1–10. DOI: 10.1109/JSYST.2019.2922589.

[188] S. Vuppala, A. E. Mady, and A. Kuenzi. "Rekeying-based Moving Target Defence Mechanism for Side-Channel Attacks." In: *2019 Global IoT Summit (GIoTS)*. IEEE. 2019, pp. 1–5. DOI: 10.1109/GIOTS.2019.8766426.

[189] Shengling Wang, Hongwei Shi, Qin Hu, Bin Lin, and Xiuzhen Cheng. "Moving Target Defense for Internet of Things Based on the Zero-Determinant Theory." In: *IEEE Internet of Things Journal* (2019). DOI: 10.1109/JIOT.2019.2943151.

[190] Bryan C Ward et al. *Survey of Cyber Moving Targets Second Edition*. Tech. rep. MIT Lincoln Laboratory Lexington United States, 2018.

[191] Chirag Warty et al. "De Bruijn sequences as secure spreading codes for wireless communications." In: *ICACCI*. IEEE. 2013.

[192] Rolf H Weber. "Internet of Things–New security and privacy challenges." In: *Computer law & security review* 26.1 (2010), pp. 23–30.

[193] Lloyd Welch. "Lower bounds on the maximum cross correlation of signals (corresp.)" In: *IEEE Transactions on Information theory* 20 (1974).

[194] Wikipedia. *Heraclitus. Philosophy. Fire*. 2020. URL: https://en.wikipedia.org/wiki/Heraclitus#Fire.

[195] Claes Wohlin. "Guidelines for snowballing in systematic literature studies and a replication in software engineering." In: *Proceedings of the 18th international conference on evaluation and assessment in software engineering*. 2014, pp. 1–10.

[196] Wenyuan Xu, Wade Trappe, Yanyong Zhang, and Timothy Wood. "The feasibility of launching and detecting jamming attacks in wireless networks." In: *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*. 2005, pp. 46–57.

[197] Su Yao, Ziwei Li, Jianfeng Guan, and Yang Liu. "Stochastic Cost Minimization Mechanism based on Identifier Network for IoT Security." In: *IEEE Internet of Things Journal* (2019). DOI: 10.1109/JIOT.2019.2961839.

[198] Vahid Zangeneh and Mehdi Shajari. "A cost-sensitive move selection strategy for moving target defense." In: *Computers & Security* 75 (2018).

[199] Kimberly Zeitz, Michael Cantrell, Randy Marchany, and Joseph Tront. "Designing a micro-moving target ipv6 defense for the internet of things." In: *2017 IEEE/ACM Second International Conference on Internet-of-Things Design and Implementation (IoTDI)*. IEEE. 2017, pp. 179–184. DOI: 10.1145/3054977.3054997.

[200] Kimberly Zeitz, Michael Cantrell, Randy Marchany, and Joseph Tront. "Changing the game: A micro moving target ipv6 defense for the Internet of Things." In: *IEEE Wireless Communications Letters* 7.4 (2018), pp. 578–581. DOI: 10.1109/LWC.2018.2797916.

[201] Hans-Jurgen Zepernick and Adolf Finger. *Pseudo random signal processing: theory and application*. John Wiley & Sons, 2005.

[202] Jianjun Zheng et al. "A survey on the moving target defense strategies: An architectural perspective." In: *Journal of Comp. Science and Tech.* (2019).

[203] Jianjun Zheng and Akbar Siami Namin. "A survey on the moving target defense strategies: an architectural perspective." In: *Journal of Computer Science and Technology* 34.1 (2019), pp. 207–233.

[204] Rui Zhuang. "A theory for understanding and quantifying moving target defense (Doctoral Dissertation)." PhD thesis. Kansas State University, 2015.

[205] Rui Zhuang, Alexandru G Bardas, Scott A DeLoach, and Xinming Ou. "A theory of cyber attacks: A step towards analyzing MTD systems." In: *Proceedings of the Second ACM Workshop on Moving Target Defense*. ACM. 2015.

[206] Rui Zhuang, Scott A DeLoach, and Xinming Ou. "Towards a theory of moving target defense." In: *Proceedings of the First ACM Workshop on Moving Target Defense*. ACM. 2014, pp. 31–40.

DECLARATION

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other University. This thesis is the result of my own work and includes nothing which is the outcome of work done in collaboration, except where specifically indicated in the text.

L'Hermitage, December 2020

Renzo E. Navas

COLOPHON

This document was typeset using the typographical look-and-feel `classicthesis` developed by André Miede and Ivo Pletikosić. The style was inspired by Robert Bringhurst's seminal book on typography "*The Elements of Typographic Style*". `classicthesis` is available for both LaTeX and LyX:

https://bitbucket.org/amiede/classicthesis/

Final Version as of January 11, 2021 (classicthesis v4.6).

DOCTORAT BRETAGNE LOIRE / MATHSTIC

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Titre : Amélioration de la Résilience de l'Internet des Objets Contraint

Mots-clés : Internet des Objets, Défense par Cible Mouvante, Sécurité Réseaux, Antibrouillage

Résumé : Les systèmes de l'Internet des Objets (IoT) sont de plus en plus déployés dans le monde réel, mais leur sécurité est en retard par rapport à l'état de l'art des systèmes non IoT. La Défense par Cible Mouvante (MTD) est un paradigme de cyberdéfense qui propose de randomiser les composants des systèmes, dans l'intention de faire échec aux cyberattaquants qui s'appuyaient auparavant sur la nature statique des systèmes. Les attaquants sont désormais limités par le temps. Le MTD a été mis en œuvre avec succès dans les systèmes conventionnels, mais son utilisation pour améliorer la sécurité des IoT fait encore défaut dans la littérature. Au cours de cette thèse, nous avons validé le MTD comme paradigme de cybersécurité adapté aux systèmes IoT. Nous avons identifié et synthétisé les techniques MTD existantes pour l'IoT en utilisant une méthode d'examen systématique de la littérature, et nous avons défini et utilisé quatre nouvelles métriques liées à l'entropie pour mesurer des propriétés qualitatives des techniques MTD. Ensuite, nous avons proposé un framework générique de MTD distribué qui permet l'instanciation de stratégies MTD concrètes adaptées aux contraintes de l'IoT. Enfin, nous avons avons conçu un protocole de synchronisation du temp authentifié, et instancié trois techniques MTD particulières : deux dans les couches supérieures du réseau (portant sur le saut de ports et sur des interfaces RESTful d'applications) - et validé l'une d'entre elles dans du matériel réel-, et la troisième dans la couche physique pour obtenir des systèmes IoT résistants aux brouillage par des nœuds internes en utilisant des techniques d'étalement du spectre par séquence directe avec des séquences pseudo-aléatoires cryptographiquement fortes.

Title: Improving the Resilience of the Constrained Internet of Things

Keywords: Internet of Things, Moving Target Defense, Network Security, Anti-Jamming

Abstract: Internet of Things (IoT) systems are increasingly being deployed in the real world, but their security lags behind the state of the art of non-IoT systems. Moving Target Defense (MTD) is a cyberdefense paradigm that proposes to randomize components of systems, with the intention of thwarting cyber attacks that previously relied in the static nature of systems. Attackers are now constrained by time. MTD has been successfully implemented in conventional systems, but its use to improve IoT security is still lacking in the literature. Over the course of this thesis, we validated MTD as a cybersecurity paradigm suitable for IoT systems. We identified and synthesized existing MTD techniques for IoT using a systematic literature review method, and defined and used four novel entropy-related metrics to measure MTD techniques qualitative properties. Secondly, we proposed a generic distributed MTD framework that allows the instantiation of concrete MTD strategies suitable for the constraints of the IoT. Finally, we designed an secure time synchronization protocol, and instantiated three particular MTD techniques: two at the upper network layers (e.g. port-hopping, and application RESTful interfaces) -and validated one of them in real hardware-, and the third one at the physical layer to achieve IoT systems resilient to insider attacks/jamming by using Direct Sequence Spread-Spectrum techniques with cryptographically-strong pseudo-random sequences.