Open access • Journal Article • DOI:10.1109/JIOT.2020.3040358

MTD, Where Art Thou? A Systematic Review of Moving Target Defense Techniques for IoT — Source link

Renzo E. Navas, Frédéric Cuppens, Nora Cuppens, Laurent Toutain ...+1 more authors

Institutions: Centre national de la recherche scientifique, École Polytechnique de Montréal

Published on: 15 May 2021 - IEEE Internet of Things Journal (Institute of Electrical and Electronics Engineers (IEEE))

MTD, Where Art Thou? A Systematic Review of Moving Target Defense Techniques for IoT

Renzo E. Navas, *Student Member, IEEE,* Frédéric Cuppens, *Member, IEEE,* Nora Boulahia Cuppens, *Member, IEEE,* Laurent Toutain, *Member, IEEE,* and Georgios Z. Papadopoulos, *Member, IEEE*

***Abstract*—*Context:* 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, successfully implemented in conventional systems, that could improve IoT security.**

***Objective:* Identify and synthesize existing MTD techniques for IoT and validate the feasibility of MTD as a cybersecurity paradigm suitable for IoT systems.**

***Method:* We use a systematic literature review method to search and analyze existing MTD for IoT techniques up to July 2020. We evaluated the existing techniques in terms of security foundations and real-world deployability using the evidence they provide. We define and use entropy-related metrics to categorize them. This is the first MTD survey to use Shannon's entropy metric empirically.**

***Results:* Thirty-two distinct MTD for IoT techniques exist: 54% are Network-layer-based, 50% present strong evidence about their real-world deployment, and 64% have weak security foundations.**

***Conclusion:* MTD for IoT is a feasible cyberdefense approach. A variety of proposals exist, with evidence about their implementation and evaluation. Nevertheless, the MTD for IoT state of the art is still immature: the security foundations of most existing proposals are weak. Novel techniques should prioritize providing convincing security foundations and real-world deployment evidence.**

***Index Terms*—Internet of Things, Moving Target Defense, Cyber Security, Metrics, Entropy, Systematic Literature Review**

I. Introduction

THE Internet of Things (IoT) is a reality. Billions of IoT devices are already deployed in real-world environments, and the number increments every year [1]. IoT systems are used in a wide variety of use cases like agriculture, city infrastructure services, industrial automation, personal health, and home usage. Modern societies are increasingly reliant on IoT systems. Their widespread usage also translates into them being a high-value target for cyberattackers [2]. From a cyberdefense perspective, the constrained nature of most IoT devices imposes additional challenges to protect them. On the one hand, most legacy security techniques are not straightforward to use and need to be adapted. On the other hand, defining novel security mechanisms is a task that requires lots of research effort and validation; a flaw in their design or implementation can have serious consequences. Research effort in IoT security has risen in recent years, but many challenges still remain open [3].

Moving Target Defense (MTD) has been proposed as a cyber-defense paradigm in 2009 [4]. It is motivated by the inherent disadvantage at which static systems are when facing cyberattackers. With enough time and a static target, attackers will eventually find and exploit vulnerabilities of the system. In other words, it is not a question of *if* but *when* the system will be penetrated. By acknowledging this fact, MTD proposes to inherently make systems dynamic in order to limit the cyberattackers in the domain of *time*. Against an MTD system, the attacker has limited time to find and exploit vulnerabilities. A vulnerability found -but not exploited yet- may not be present in the next system state. Even if a vulnerability is exploited, the future state of the system may neutralize its effects. This is a desirable feature for resilient systems.

In 11 years since the inception of MTD, more than 80 distinct techniques have been proposed [5]. Also, several MTD techniques survey articles have been published [5]–[11]. However, limited work has been published about MTD targeted at IoT systems. The most recent peer-reviewed MTD surveys [9], [11], identify less than five IoT-specific MTD techniques. A recent book chapter [12] focuses on MTD for IoT and effectively identifies around a dozen techniques. MTD for IoT is an acknowledged promising field of study [9], [12]. However, we believe there is still a lack of an in-deep survey of its state of the art.

In this work, 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 a *systematic literature review* methodology, widely employed in the Medical science fields, but adapted for the Software Engineering fields by P. Brereton, B.A. Kitchenham et al. [13]. 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

Manuscript received August 17, 2020; revised October 20, 2020; accepted November 12, 2020.

Renzo E. Navas is with IMT Atlantique, Lab-STICC, UMR CNRS 6285, F-35700 Rennes, France (e-mail: renzo.navas@imt-atlantique.fr).

Frédéric Cuppens and Nora Boulahia Cuppens are with the Department du Genie Informatique et Genie Logiciel, École Polytechnique de Montréal, Montreal, QC H3T1J4, Canada (e-mail: frederic.cuppens@polymtl.ca; nora.boulahia-cuppens@polymtl.ca).

Laurent Toutain and Georgios Z. Papadopoulos are with IMT Atlantique, IRISA, UMR CNRS 6074, F-35700 Rennes, France (e-mail: laurent.toutain@imt-atlantique.fr; georgios.papadopoulos@imt-atlantique.fr).

methodology aims at producing evidence-based answers to clearly defined *Research Questions*.

To the best of our knowledge, this is the first MTD survey to empirically use Shannon's entropy metric for the studied MTD techniques. We also developed other entropy-related metrics to measure qualitative aspects that are not captured by the Shannon entropy.

Contributions In summary, the major contributions of this work are the following:

- A *Systematic* Review of MTD techniques for IoT. This review enforces the guidelines by Kitchenham et al. [14]. This is the first peer-reviewed MTD survey focused on IoT. Moreover, 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 studied techniques.
- The definition of four new entropy-related metrics and their empirical application. Moreover, this is the first review to make practical use of Shannon's entropy as a metric. The metrics have applications beyond the scope of this article.

The rest of this work is structured as follows. Background and related work are presented in Sec. II. The entropy-related metrics are defined in Sec. III. The methodology used in this survey is documented in Sec. IV and the results in Sec. V. Some discussion and future work perspectives are offered in Sec. VI. Finally, Sec. VII concludes this work.

II. Background and Related Work

In this section, we first present the MTD cyberdefense paradigm. Then, we focus on MTD techniques: definitions, design principles, and their taxonomy. Finally, we highlight related work about MTD surveys and the systematic review approach.

A. The MTD Paradigm

MTD is a cyberdefense paradigm that proposes to proactively dynamize systems' components to thwart cyber attackers that rely on the static nature of them. Leaked system information is now ephemeral, and time is a constraint for attackers.

a) 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 [15] in a precursor idea of the Frequency-Hopping Spread-Spectrum techniques for wireless communication systems. Defense through constant change in the Internet era can also be found at least since 2001: a U.S. DARPA project explored dynamic IP address and TCP port numbers [16]. However, it is not until 2009 that the term "MTD" was coined and explicitly proposed as a cyberdefense paradigm by the Networking and Information Technology Research and Development (NITRD) program in the context of a U.S. National Cyber-Defense Summit [17].

b) MTD Definition: R. Zhuang et al. [18] concisely define MTD as "constantly changing a system to reduce or move the attack surface available for exploitation by attackers". The NITRD program originally defined the MTD paradigm as follows [19]:

> MTD enables us to create, analyze, evaluate, and deploy mechanisms and strategies that are diverse and that continually shift and change over time to increase complexity and cost for attackers, limit the exposure of vulnerabilities and opportunities for attack, and increase system resiliency.

c) Rationale: MTD acknowledges that vulnerabilities are present in any system and that there is an information asymmetry between static systems and attackers. Because information about the system does not expire, cyber attackers with enough resources –*time* in particular– will eventually find an exploit, develop and launch a successful attack. The defeat of a static system facing a persistent attacker is ineluctable.

MTD tries to equilibrate this information asymmetry by limiting the *time* validity of –possibly leaked– system information. MTD's security 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 a 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*. A system's attack surface is "the subset of the system's resources that an attacker can use to attack the system" [20].

The components-attack surface's constant "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 shifted, and the vector of attack may no longer be valid. Even if the attack is successful at t_1, the MTD movement may limit the attack's effectiveness; e.g., the system at $t_1+\Delta t$ is no longer vulnerable. This MTD's *game-changer* property can be resumed [17] as "attacks only work once if at all".

MTD contrasts with systems' security measures that try to keep the attack surface small (i.e., attack surface reduction). In the software domain, these approaches remove bugs at the source, identify malicious attacks against deployed software, and patch software as rapidly as possible [19]. However, the "perfect" software approach does not scale to the increasing complexity current system's software. The patch distribution approach is standard practice in modern systems, but it has proven difficult to be ahead of the attackers. Avoiding exposed vulnerabilities should still be a priority, but MTD is proposed as a game-changing *(pro)active* approach that can complement that standard *reactive* practices. Furthermore, defenders do not entirely know a system's attack surface (e.g., zero-day vulnerabilities). Thus, those *unknown* vectors of attack can not be reduced in number nor quality, but at least they can be "moved" by MTD techniques.

d) Literature: The MTD literature can be divided into three fields [7]: *theory* [18], [21]–[23], *strategy* [5]–[11], and

evaluation [24]–[26]. The *Strategy* field covers concrete MTD *techniques* that can be implemented in real systems. This survey focuses on this field, with the particularity of being applicable to constrained IoT systems, and we detail it in the next subsection.

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 [18], a Cyber Attack Theory [22], and their interaction [23].

The MTD *Evaluaton* field deals with methods to evaluate and quantify the effectiveness of MTD systems. This research field 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 guide the design and implementation of novel or existing MTD techniques.

B. MTD Techniques, Design Principles, and a Taxonomy

An MTD technique is an instance of the MTD paradigm detailed for a specific system and use case. We define the Moving Parameter (MP) as a component of the system that will be changed-adapted over time by the MTD technique.

In this subsection, we focus on general design principles shared by the more than 100 distinct techniques that exist and were previously identified and analyzed in the literature. 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. [7], 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 [5], [6] and is composed of the following categories:

- **Dynamic Data:** Techniques that change the format, encoding, or representation of application data, i.e., same semantics with different syntax.
- **Dynamic Software:** Techniques that change an application's binary code dynamically, e.g., binary objects shuffling, application diversification.
- **Dynamic Runtime Environment:** Techniques that change the execution environment dynamically, e.g., RAM addresses, instruction set.
- **Dynamic Platform:** Techniques that change the computing platform properties, e.g., CPU architecture, OS, virtual machine instance.
- **Dynamic Networks:** Techniques that change network properties, e.g., protocols, addresses, topology.

As stated before, this taxonomy is based on the MP's system layer and is widely used in MTD literature. We use these categories extensively in the current work.

C. Related Work

MTD was proposed in late 2009, and the first peer-reviewed survey of MTD techniques appeared in 2013 [6]. Since then, many generic surveys of MTD techniques have been published [5]–[11] 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. Zheng et al. [9] 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. [11] 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. [12] 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.

In respect to the systematic literature review approach, none of the aforementioned surveys used it. However, Torquato et. al [27] conducted a systematic mapping study[1] of *MTD in cloud computing*. Hosseinzadeh et al. [28] performed a systematic review of *Diversification and obfuscation techniques for software security*, a broader topic than MTD.

III. METRICS

In this section, we define the metrics that are employed in the current literature review. Several metrics for MTD have been proposed [24]–[26]. However, in general, they are of difficult applicability to concrete and heterogeneous strategies. In this survey, we use metrics related to the entropy of the moving parameter. These metrics have the property of being applicable to the surveyed MTD techniques with a reasonable effort.

[1]There are differences between a *systematic literature review* 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 systematic review has a narrower subject, and fewer studies will be included. Sometimes, the term *systematic review* is used for what is technically a *mapping study*.

A. Shannon Entropy of the Moving Parameter

This metric is based on the *maximum* Shannon's entropy of the Moving Parameter (MP) of an MTD technique. Works by Zhuang et al. [18], and Hobson et al. [21] already used Shannon's entropy concept for MTD systems in a theoretical way. In this article, we present our own approach but refer the reader to those works for more information.

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:

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

Large values of $H(X)$ are desirable for MTD sytems. This assumption is defined as the *MTD Entropy Hypotesyis* [18], also Hobson et al. [21] defines that an MTD technique is *unpredictable* iff $H(X) \gg 0$.

In this work, 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 [18]: 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, $0 \leq H(X) \leq H(Y)$.

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.
- 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.

B. Qualitative Entropy-related Metrics: Definitions

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.

In order to capture some of these qualitative differences, we define four novel metrics: GEN, STO, MOV, and ATT.

TABLE I: MTD Entropy-related Metrics.

Metric	Description	Possible values
$H(X)$	Shannon Entropy of 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}
Q	g(GEN) + s(STO) + m(MOV)	{0, 1, ... , 5, 6}
ATT	Cost of an attack, assuming a state x_i	{Low, Med., High}

Besides, we define an indirect metric Q derived from the first three. The metrics 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 I.

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 (e.g., 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 metric Q aggregates the three system-centric metrics GEN, STO, and MOV. In order to sum direct metrics, they should be expressed in the same dimension or be dimensionless. For a given technique, each individual system-centric metric can be mapped to a dimensionless value in $\{0, 1, 2\}$ from the original domain of $\{Low, Med., High\}$. Formally, we use the functions g, s, and m to convert the direct metrics to a dimensionless scalar value. From a system defense's perspective, 0 is the most desirable value (lower cost) and 2 the least (higher cost). We define Q as the arithmetic sum of the mapped dimensionless system-centric metrics. 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. Its value is in the range $\{0, 1, ..., 6\}$ and the lower this value, the better from a system's perspective. Ideally, we want MTD techniques with high values of entropy $H(X)$ at a low-cost Q.

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 (*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.

C. Qualitative Entropy-related Metrics: From Definitions to Empirical Use and Evaluation

a) The Third Value: All the qualitative metrics, but Q, are of ternary value: Low, Med., and High. This choice is justified because of the inherent uncertainty and difficulty of measuring them for concrete and heterogeneous MTD techniques. Binary values were discarded because of being too coarse-grained, it was hard to define the limiting threshold, and high uncertainty was present in values *close* to this threshold. An ordered ternary system mitigates these issues. We can approach the estimation in a binary-way, but if the estimation proves not conclusive (e.g., because the value is close to the binary threshold), the ternary value between the two extremes

can be assigned. In this survey, we approached the evaluation binarily and use the *Med.* value that way.

b) Dimension of Cost: Furthermore, in Sec. III-B we defined the metrics as an estimation of the *cost* of the entity's limited resources, but we did not assign a dimension to this *cost*, e.g., time-seconds, information-bit, volume-μm^3, currency-dollars. The definition is intentionally generic, because what is a *limiting* resource for one entity in a concrete MTD setting, may not be limiting in another one. Thus, this generic definition allows the metrics to be adaptable to a variety of settings.

However, to compare two or more MTD techniques –like in this survey–, the metric definition has to be *consistent* among them. In other words, for an empirical use of the qualitative metrics, more definitions are needed to express this *cost*.

Each direct qualitative metric has to be further defined in terms of a commensurable quantity (i.e., same dimension) that will be consistently used among all the techniques under study. In this survey, we used the following *cost* dimensions for the direct metrics: *time* for GEN, MOV, and ATT; and *information* (e.g., bits) for STO.

As for the indirect metric Q, it synthesizes the three direct system-centric metrics in a single scalar value. Consequently, Q conveys less information than the three metrics (i.e., a three-dimensional vector), but allows for more straightforward comparisons and visual representations of the overall system's entropy cost. In this survey, the three functions g, s, and m that map the values from the direct metrics to the same codomain are simply: $\{Low \mapsto 0, Med. \mapsto 1, High \mapsto 2\}$.

c) Estimation of Cost for concrete MTDs: Once the dimension of each direct metric is defined, a fundamental question should be answered: *how to evaluate a direct metric for a set of MTD techniques?*

In an ideal setting, we should dispose of baseline hardware for the IoT system and the attacker. Then, implement all the MTD techniques into consideration and for the attacker to implement the state-of-the-art attack that corresponds to each technique. In this ideal hardware case, we could measure the cost directly (*time* and *bits*), and even use more fine-grained values (e.g., $\mathbb{R}_{\geq 0}$). A second approach by simulation could ease the task, but a simulation model valid for all the heterogeneous MTD techniques' use cases (e.g., physical modulation, firmware image exploits, network addresses, application resources) does not exist.

Neither of the aforementioned approaches is empirically practicable in a survey, some of the reasons are: (i) The required engineering person-hours to implement all the techniques under consideration will be hard to acquire; for example, in the current survey, only %22 of the articles implemented in hardware the sole technique under its consideration. (ii) Some techniques in the literature do not provide all the design details to implement them in real hardware.

However, consider the following example, as similarly stated in the intro of Sec. III-B: We can sensibly agree that there is one ore more orders of magnitude in the GEN and MOV time-cost difference when: (A) the MP is a 32-bit RAM value inside a node (e.g., $\leq 100msec$) (B) the MP is a 100 node physical topology arrangement (e.g., $\gg 100msec$). Even if we can not precisely measure the *time* value, 32 information bits are considerably "less costly" to generate and move than a 100 physical IoT nodes' topology. In other cases, the techniques may be of the same order of magnitude. This order-of-magnitude comparison can be attempted with any pair of techniques and metric.

Yet, the question remains: *how to capture–evaluate these differences or similarities in orders of magnitude for every heterogeneous MTD technique in a set?* In this survey, we use a *relative metrics* approach and state some hypotheses to approach a methodology to make this comparison. We detail this evaluation method in the remainder of this section.

d) Evaluation in this Survey – Approach: First, we do not attempt to evaluate the direct metrics into an absolute value, but in *relative* terms to the set under evaluation.

Second, the following general hypotheses–assumptions were used, if needed: (i) A system with 100 nodes. (ii) A node with 32 KB of RAM and 250 KB flash.

Then, for the set of MTD techniques to evaluate, the approach was the following:

1) Define a technique as representative of the "Low" value.
2) Define a technique as representative of the "High" value.
3) For each remaining technique, determine to which category–order of magnitude it pertains, as follows:
   a) If a similar technique has been categorized, use that same category; unless the current technique presents a game-changing technical innovation (in that case, continue evaluation).
   b) If the technique has empirical evaluation elements, use them to match the order of magnitude with "Low" or "High" techniques. Else, continue.
   c) If no empirical evidence nor similar technique is present, the evaluators can extrapolate the cost for the metric and technique under study using their knowledge in the field and the technique's article (or a "synthesis" of the article). For example, propagating MP changes that impact a distributed system will be more costly than changes that impact only one node, or hardware-based MPs are more resource-consuming that information-based MPs. This is a non-methodical *subjective* step.
   d) If no conclusive evaluation can be done, categorize the technique as "Med.".

In this survey, this evaluation was done by three of the authors. First, we agreed on the representative "Low" and "High" techniques for each metric. In this case, a same representative reference was chosen for all metrics (GEN, STO, MOV, and ATT): "Low" $\equiv$ "IPv6 64-bit Addresses randomization" by Sherburne et al. [29], and "High" $\equiv$ "OS firmware reconfiguration" by Casola et al. [30]. Then, the evaluation was done independently using the extracted data for each technique (See Listing 1). Moreover, at least one author that read the full article was available to answer questions to the others, if needed, in the subjective extrapolation step (3c). Finally, the evaluation results were synthesised using simple majority and using "Med." in case of no majority.

IV. Methodology

The survey methodology used in this work is based on the systematic literature review guidelines by Kitchenam et al. [14]. There are three main phases in a systematic review process:

1) *Planning*: Involves specifying the research questions 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 article.

The conducting phase, with detail on the search and selection processes, is illustrated in Fig.1. In the following, we explicit the research questions and detail the protocol and execution of the conducting phase.

A. 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 Research Questions (RQ) that guide this review:

RQ1: How many proposals of MTD techniques for IoT exist?
RQ2: What characteristics can be observed in the proposals?
RQ3: How sound are the security foundations of the proposals?
RQ4: To what extent the proposals can be used in a real IoT deployment?

The research questions are ordered from the more generic to the more particular. The first two are broad research questions. We separated them because RQ1 focuses on quantitative facts while RQ2 on qualitative ones. The last two, RQ3 and RQ4, inquire into technical qualitative properties of the proposals. They are useful to give an assessment of the *maturity* of the MTD for IoT field.

B. Search Process

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

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 systematic literature review. In Sep. 2019 the set consisted of 18 articles, and we estimated a systematic literature review meaningful. In July 2020, we did a manual update to include recently published work.

Reproducibility. Manually including article [31], the automated search, and the snowballing method will yield the same results without the need of this manual set.

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) Snowballing: Snowballing is a search technique that identifies potential additional articles to include in a systematic literature review [32]. Snowballing is a complementary technique to the automated string search process and requires a *start set* of papers that are known to be relevant and will be included in the literature review. Snowballing's rationale is that papers about the same subject will reference each other and by following these interconnections, we can identify relevant work that may not have been included in the start set. Backward snowballing identifies (past) articles that are on the reference list of an included study. Forward snowballing identifies (future) articles that refer to an included study. These additional articles will go through the *selection process* to determine if they are relevant or not. The snowballing process can be applied iteratively over the new set of selected articles. 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.

C. 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. 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.

The exclusion criteria are:

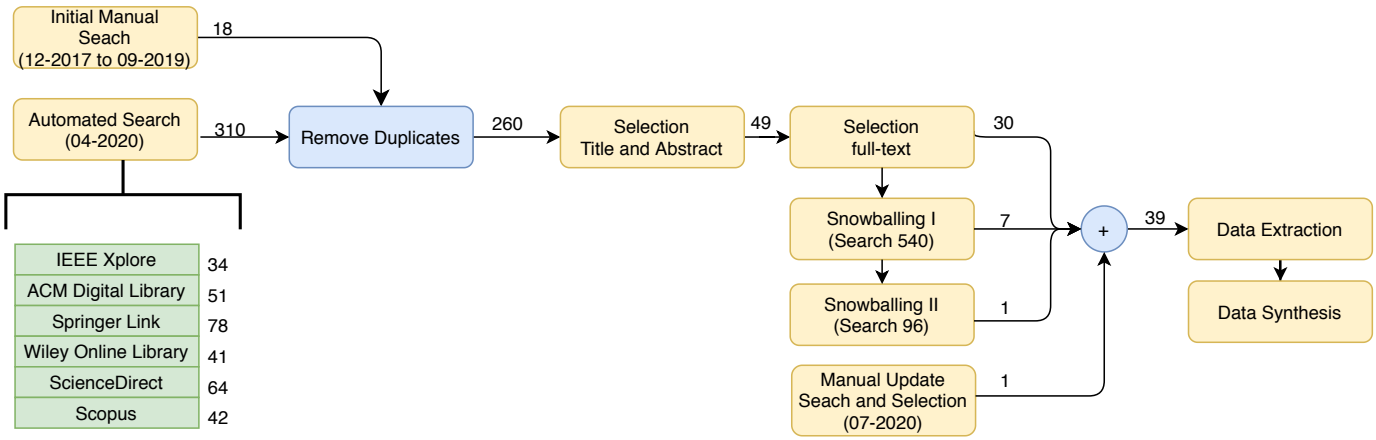


Fig. 1: Conducting the review: detail on the search and selection processes. The number of articles after an activity is represented in labels at the exit edges.

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

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.

D. Data Extraction Process

Each of the 39 selected articles was read thoroughly by Renzo E. Navas. The data extraction template evolved between Jun 2019 and October 2020. The final template used to extract relevant data from each study is shown in Listing 1. 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.

E. 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 research questions of Sec. IV-A. There are a variety of data synthesis methods [14]. 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 some aspects (mostly qualitative) of the primary studies. In this process, we used existing MTD theory (e.g., moving parameter, accepted taxonomies) and the *metrics* we developed in Sec. III. The metrics allow a common frame of reference to synthesize and compare qualitative aspects of different studies.

Listing 1: Data Extraction Template.

- Standard bibliography data:
  - Title, author, year, type of publication, venue.
- MTD technique name or brief description.
- Moving Parameter (MP).
- MTD technique taxonomy (Sec. II-A):
  - Data, Software, Runtime Environment, Platform, Network.
- Metrics:
  - Evaluate MP Shannon entropy metric (See Sec. III-A).
  - Evaluate MP qualitative metrics (Defined in Sec. III-B)
- 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.
- IoT Software:
  - What IoT OS or firmware is used?
- Synthesis of the proposal with technical details (1-6 pars.).

V. Results

In this section, we present the results from the systematic review process. The Research Questions (RQs) of Sec. IV-A 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 the last paragraph addressing a RQ.

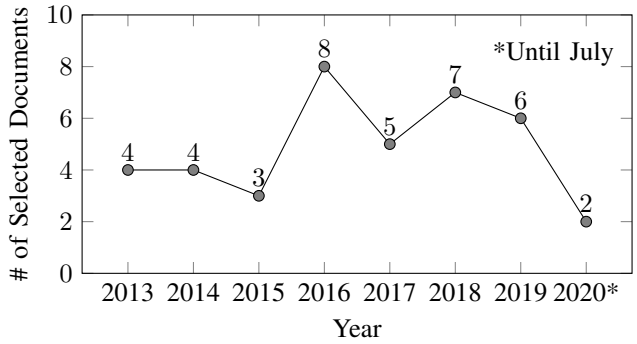


Fig. 2: Number of selected documents per year (Total = 39).

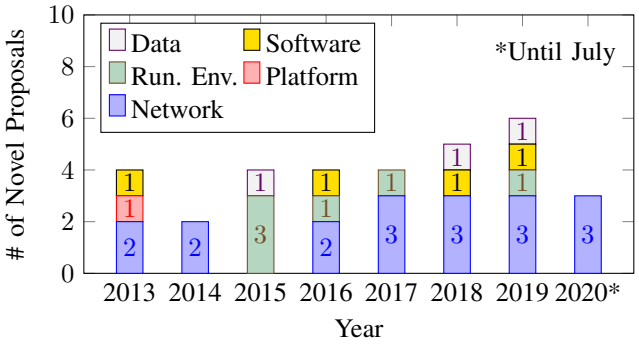


Fig. 4: Number of novel proposals per year (Total= 32).

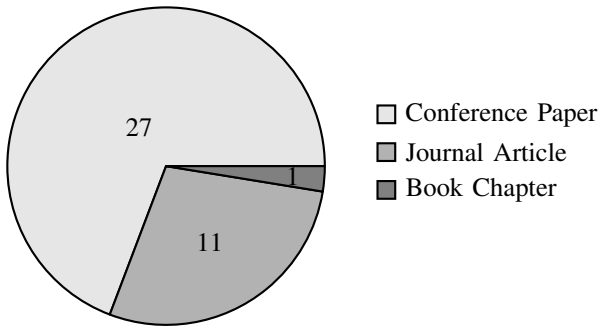


Fig. 3: Number of documents by publication type.

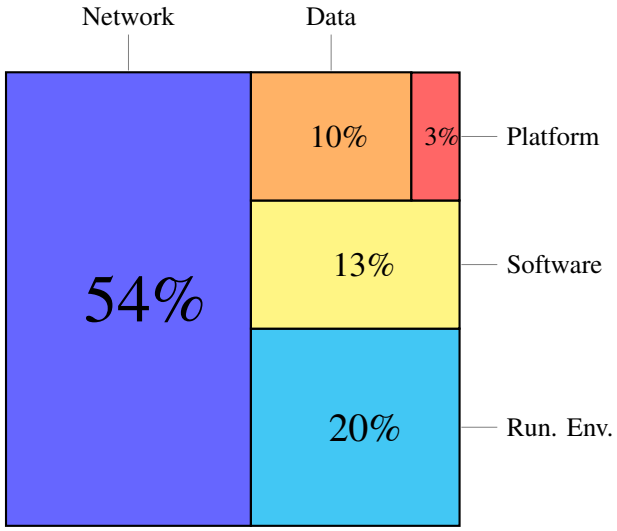


Fig. 5: Taxonomy distribution of MTD techniques for IoT.

A. RQ1: How many proposals of MTD techniques for IoT exist? (Status of Field of Study: Quantitative)

The systematic review process, shown in Fig. 1, identified 39 documents containing 32 distinct proposals.

1) Documents: In Fig. 2 we plot the published documents per year. The first article is from 2013, two years after the first general-purpose MTD 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. The document publication type distribution is shown in Fig. 3, 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.).

2) Proposals: 32 novel proposal have been identified. In Fig. 4 we plot the novel proposals per year. 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. 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.

Interpretive discussion. The figure of thirty-two distinct techniques was not evident prior to this survey. The identified corpus in the MTD literature was of about a dozen techniques.

B. RQ2: What characteristics can be observed in the MTD for IoT techniques? (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. We use general MTD theory concepts presented in Sec. II and the metrics we defined in Sec. III.

1) MTD Taxonomy. Distribution and Trends: We present the distribution of the techniques by MTD taxonomy in Fig. 5. *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. 4 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 research question is:

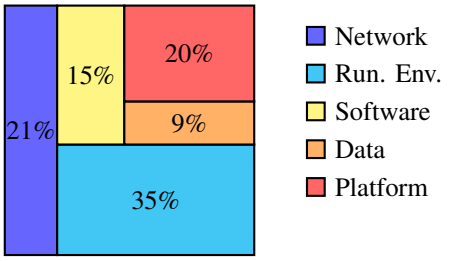


Fig. 6: Taxonomy distribution of general MTD techniques [5].

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

To answer this question, we leverage on the general MTD survey from Lincoln MIT laboratory [5]. It dates from 2018 and comprises 89 distinct general purpose MTD techniques. The taxonomy distribution of the techniques is shown in Fig. 6. *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. [10]. Also, recent MTD surveys [11], [27] focus on Network MTD solutions, which indicates a growing interest in the research community.

2) Moving Parameter. Shannon Entropy and other Metrics: This subsection presents the results of applying several metrics related to the Moving Parameter (MP) X. The metrics are defined in Sec. III. 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 II. In Fig. 7, 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 moving parameter. The results are shown in Fig. 8. The metric Q that aggregates the system-centric metrics is shown in Fig. 8a. From a system's perspective, the lower the Q value, the better. Similarly, the attacker-centric metric ATT is presented in Fig. 8b. 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. 8. 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 moving parameter 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*).

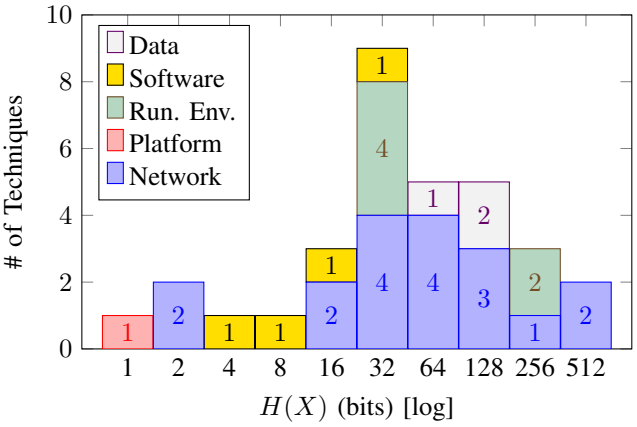
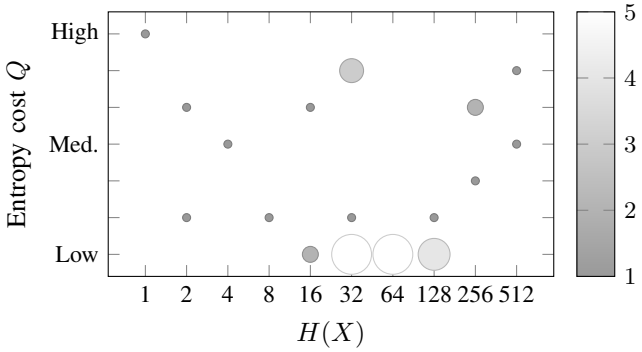
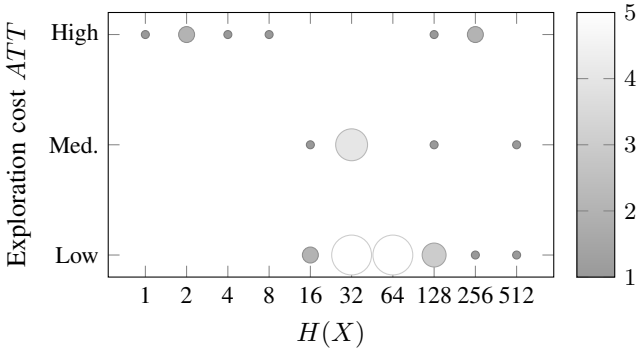


Fig. 7: Histogram of Shannon's entropy of the techniques.



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



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

Fig. 8: Relationship between Shannon's entropy and other metrics (# of techniques per combination).

The first expectation is observed in Fig. 8a 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.

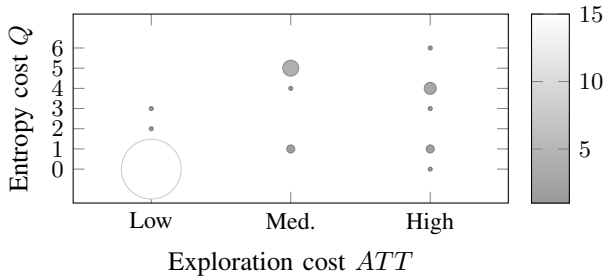


Fig. 9: Number of techniques grouped by ATT and Q.

It means that those techniques will be costly to implement in a real-world system. The second expectation is observed in Fig. 8b. 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. 9 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.

Interpretive discussion. Around 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.

C. RQ3: 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 II 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*).

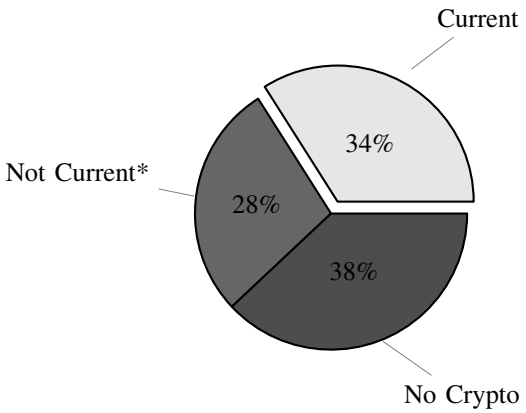
In other words, we associate the *security foundation* of a proposal with the cryptographic primitive it uses. In order to do a proper assessment of each technique, in many cases, we required more detailed information than the three cryptography-related items. This complementary information was obtained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

[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.

Fig. 10: Cryptographic categories of the techniques.

The results are shown in Fig. 10. 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.

Interpretive discussion. The goal of an MTD technique is to improve the security of an IoT system. To measure the security of a system is a challenging task and depends on many factors. Comparing different techniques is not straightforward. We simplified this comparison problem by taking into account the cryptographic primitives of each technique. If not current cryptography is used, the security foundation of the proposal is not convincing, an attacker can eventually replicate the system's movement and neutralize the effect of the MTD. The results show that only 34% of the techniques use current cryptography. It is a low value considering security is the main objective of an MTD technique.

D. RQ4: To what extent the proposals can be used in a real IoT deployment?

To answer this question, we use empirical evidence provided by the proposals in their corresponding publications. All of

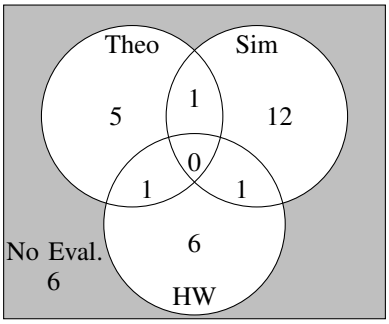


Fig. 11: *Evaluation* status of the techniques (Total = 32).

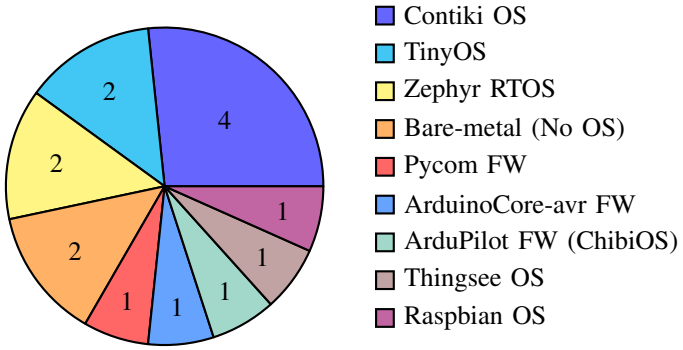


Fig. 12: IoT OS or firmware (FW) used in a technique's implementation or evaluation (Total = 15).

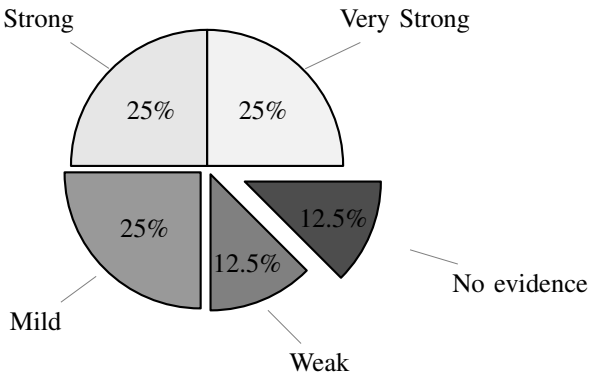


Fig. 13: Distribution of categories of evidence about real IoT deployment of techniques.

them provide, with varying levels of detail, a *design* specification. 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 subcategories. *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). If applicable, we also surveyed the IoT software used by an implementation or evaluation.

The raw results are the following. For *implementation*, 50% percent of the techniques were implemented, and the rest were not. In Fig. 11, 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. Finally, in Fig. 12, we show the distribution of the IoT OSs or firmwares used by the techniques' evaluation or implementation. Contiki OS was the preferred IoT software, with 27% of techniques using it.

To answer RQ4, 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:

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

The results are shown in Fig. 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.

Interpretive discussion. An MTD technique has an important empirical component because it is meant to be used in a real deployment. Though theoretical proposals push forward the field and eventually lead to more empirical variants, we believe that the empirical component should be prominent in the MTD for IoT. Highly theoretical MTD proposals, feasible in legacy systems, may never be usable in constrained IoT systems. The results of this section are encouraging: 50% of techniques provided strong or very strong evidence about their usability and only 13% did not provide any evidence.

E. 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 II. *Assumptions*: To evaluate Shannon's entropy $H(X)$, we made assumptions about many proposals. We detailed the assumptions in the table's footnotes. Those proposals either lacked design-implementation details or where generic, and we instantiate them assuming a baseline IoT system of 100 nodes and nodes with 32 KB of RAM and 250 KB flash. As for the qualitative metrics, we detailed the procedure in Sec. III-C.

TABLE II: MTD for IoT Techniques

Ref.	Technique	Moving Parameter (X)	Tax.*	Metrics*					Cryptography		Impl.	Evaluation		
				$H(X)$	GEN	STO	MOV	ATT	Primitive	Input		Theo	Sim	HW
[30], [33], [34]	Crypto-protocol reconf.	Network Crypto-Protocols	S	4	+	/	-	+	-				✓	
[30], [33], [34]	Firmware reconf.	OS Firmware (In local storage)	P	1	+	+	+	+	-		✓			
[29], [35]–[37]	μMT6D	IPv6 IID Addresses (64-bit)	N	64	-	-	-	-	SHA256	Key	✓		✓	
[38]–[42]	OS If. Div. Addr. Layout	Memory layout of linked binaries	R	32†	/	+	+	/	MD5	Salt	✓			✓
[38]–[42]	OS If. Div. Names	Symbol names of OS libraries	R	32†	-	/	-	/	SHA-2	Salt	✓			
[43]	Code Diversification	Code to store and execute	S	20**	/	/	+	/	-					
[31]	μScramble	Binary Linked Code	S	32†	/	+	+	/	LLVM PRNG	Seed	✓			✓
[31]	μSSP	Stack Canary (32-bit)	R	32	-	-	-	-	μRNG (Keccak [44])	Seed(SRAM)	✓			✓
[45]	IPv6-Multicast (SARCAST)	IPv6 Multicast Group-ID (80-bit)	N	32	-	-	-	-	SHA-1	Salt(32b)	✓		✓	✓
[46]	An ASLR Proposal	RAM addresss layout (ASLR)	R	52‡	-	-	-	-	-			✓		
[47]	SDN-IoT Topology Reconfiguration	Network Topology (Routing)	N	525§	+	/	+	/	-				✓	
[48]	Honeypots with Cellphones	Network Nodes Roles	N	47	+	/	+	/	-				✓	
[49]	AShA	MAC/IPv6 Address (16-bit)	N	16	-	-	-	-	HMAC	Key			✓	
[50]	ZD Game Theory Approach	Resource Node Locations	N	671¶	-	/	+	-	-			✓		
[51]	6HOP	P2P IPv6 IID (64b) + Port (16b)	N	160	-	-	-	-	An Unkeyed-Hash	Secret(512b)				
[52], [53]	DLSeF	Encryption Key for App. Data	D	128	-	-	-	-	*see note*‖	Key			✓	
[54]	Ephemeral	IPv6 IID Addresses (64-bit)	N	64	-	-	-	-	A Block Cipher	Key	✓		✓	
[55]	Stochastic Cost Minimization	Nodes that mutate Network Address	N	44††	-	-	-	-	-			✓		
[56]	Application Data re-Keying	Encryption Key for App. Data	D	64	-	-	-	-	LED Block Cipher	Key(64b)	✓		✓	
[57]	PHY-layer Divesification	PHY-layer Technology	N	2	+	+	-	+	-				✓	
[58]	APP-layer Protocol Diversification	Communication Protocol	S	8‡‡	-	/	-	+	A PRNG	Seed	✓			✓
[59], [60]	Re-keying with Side-Channel Attacks	Encryption Key	D	128	-	-	-	+	(Dziembowski [61])	Key				
[62]	Malware tolerant Mesh-Networks	Device Groups and Group-Keys	N	256¶¶	/	-	/	-	IRS [63] and *note*¶¶	Key			✓	
[64]	Identity Virtualization	Node IDs	N	64***	-	-	-	-	An Unkeyed-Hash	A Secret	✓		✓	
[65]	MAC Address Randomization	MAC Address (48-bit)	N	48	-	-	-	-	-			✓		
[66]	MAVR	Memory layout of linked binaries	R	256† † †	/	+	/	+	A PRP	?	✓			✓
[67]	SDR defined PHY-layer	PHY-layer Modulation	N	3	-	/	-	+	-		✓	✓		
[68]	uOTA	P2P IPv6 IID (64b×2)	N	128	-	-	-	-	-		✓		✓	
[69]	AVRAND	Memory layout of linked binaries	R	256§§	/	+	/	+	A PRNG	Seed	✓			✓
[70]	SAD-SJ	PHY-layer TDMA Slot allocation	N	128	-	-	/	/	A PRP (Sym.Cipher-base)	Key		✓	✓	
[71]	UDP Port-Hopping	UDP Port Number	N	16	-	-	-	-	ChaCha20	Key+Nonce	✓	✓		✓
[71]	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 $\approx$!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.

VI. Discussion

In this section, we summarize the state of the art of MTD for IoT with an added component of interpretation. We assess the limitations of the review and identify future research opportunities.

A. The state of the art of MTD for IoT

One of this survey's main objectives was to highlight MTD as a viable defense paradigm for IoT. First, we identified the proposals that exist. Secondly, once the IoT *corpus* identified, we categorized the techniques using standard MTD taxonomies and compared the result with a well-known corpus of legacy MTD techniques. The predominance of network techniques in the IoT field was expected and validated.

Then, we evaluated more fine-grained qualitative factors of the techniques. A hypothesis was that the MTD for IoT state of the art is not as *mature* as the non-IoT counterpart. Nevertheless, MTD is technically possible (*usability*) and desirable (*security*). For example, non-IoT solutions have sound implementation and experimental foundations. In contrast, IoT systems have many software and hardware constraints that, a priori, make the MTD techniques harder to implement. To find answers to this *usability* question, we empirically studied the techniques' implementation and evaluation status. Furthermore, we assessed the techniques' cryptographic primitives and linked deprecated or lack of cryptography usage as a sign of weak *security* foundations. The overall results in terms of usability and security of the techniques were mixed, validating the hypothesis that the field is still in development. Nonetheless, a non-minority number of techniques also proved that MTD for IoT can be usable and have solid security foundations.

Finally, another axis of research was the usage of *metrics* to evaluate other qualitative factors of the corpus. More precisely, we used metrics related to the entropy of the MP. To the best of our knowledge, this is the first survey to apply the Shannon entropy of the MP in an empirical way. However, Shannon's entropy was not entirely appropriate to make fair comparisons among techniques with MPs of very different nature[4]. To complement it, we defined metrics in the line of Kolmogorov complexity [72], i.e., to evaluate the amount of resources needed to do a set of actions over the MP. The novel metrics capture various qualitative factors of the entropy of the MP. This allowed us to provide a more comprehensive characterization of the studied corpus using entropy as a common denominator.

B. Limitations of this study

One of the main limitations of this 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" [27] worked with 95 papers. However, we believe that the amount of material included in the current survey is sufficient to identify the most prominent trends.

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

[4]For example, a new 128-bit address is not qualitatively the same as a new 100-nodes physical network topology configuration.

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 systematic review 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 II. 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.

C. Research opportunities

In this subsection, we identify some research opportunities. First, there is almost a complete lack of MTD for IoT techniques in the Platform category that can be explained by the inherent limitations of the IoT hardware. Novel techniques could leverage on legacy MTD, where Platform techniques account for 20%, and adapt the most suitable proposals.

Secondly, the usage of the entropy *metrics* can go beyond the current survey. Naturally, it would be interesting to use the metrics to measure the non-IoT MTD techniques and compare their state of the art with that of the MTD for IoT. In a per technique basis, metrics usage can help system designers make relevant trade-off decisions.

Thirdly, novel proposals that use current cryptography and explicitly prove cryptographic aspects of the proposal are required. They are a minority in the existing techniques. In terms of implementation and evaluation, the current state of the art is more developed. However, more hardware evaluations -with available source code- will be beneficial for the community and help to the overall establishment of MTD for IoT as a mature field of research.

Finally, we highlight MTD as a suitable cyberdefense paradigm for IoT. Current techniques have a mixed level of maturity. Despite that, many have a working implementation, hardware evaluation, and current cryptography. Network techniques are predominant, but there are still many opportunities, even in that field. For example, only one [47] SDN-based technique for IoT exist[5]. Overall, there are still many unexplored areas. The MTD for IoT field could leverage from existing legacy-MTD or explore novel ideas only possible within the IoT paradigm.

[5]Many SDN-based techniques apply to the legacy network and not to the IoT system. Thus, they are not tailored for IoT but are legacy SDN solutions.

VII. Conclusion

Moving Target Defense is acknowledged as a promising cyberdefense paradigm for IoT systems, but an extensive state of the art was missing in the literature. To fill this gap, we conducted a systematic review of MTD techniques for IoT systems. We identified and analyzed thirty-two different techniques, of which half were from the Network category. Furthermore, most of the techniques have convincing empirical evidence about their real-world deployability. We used this fact to validate the feasibility of MTD for IoT. However, only a third of them have sound cryptographic foundations, indicating that their security-related aspects need improvement. MTD for IoT is a reality, and future work should prioritize providing strong security foundations for the proposed techniques. Research effort could also be well invested in techniques of the least explored categories: Platform and Data. Within the Network techniques, constrained-SDN is a promising technology that can enable many novel MTDs. In respect to the MTD metrics field of research, we developed entropy-based metrics of empirical applicability that, in conjunction with Shannon's entropy, can be useful in future MTD-related applications beyond the scope of this survey.

References

[1] M. D. Saulles. (2019) Iot statistics - information matters. [Online]. Available: https://informationmatters.net/internet-of-things-statistics/

[2] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "Ddos in the iot: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.

[3] M. A. Khan and K. Salah, "Iot security: Review, blockchain solutions, and open challenges," *Future Generation Computer Systems*, 2018.

[4] A. Ghosh *et al.*, "Moving target defense co-chair's report-national cyber leap year summit 2009," *Tech. Rep., Federal NITRD Program*, 2009.

[5] B. C. Ward *et al.*, "Survey of cyber moving targets second edition," MIT Lincoln Laboratory Lexington United States, Tech. Rep., 2018.

[6] H. Okhravi, T. Hobson, D. Bigelow, and W. Streilein, "Finding focus in the blur of moving-target techniques," *IEEE Security & Privacy*, vol. 12, no. 2, pp. 16–26, 2013.

[7] G.-l. Cai *et al.*, "Moving target defense: state of the art and characteristics," *Frontiers of Information Technology & Electronic Engineering*, vol. 17, no. 11, 2016.

[8] C. Lei, H.-Q. Zhang, J.-L. Tan, Y.-C. Zhang, and X.-H. Liu, "Moving target defense techniques: A survey," *Security and Communication Networks*, vol. 2018, 2018.

[9] J. Zheng and A. S. Namin, "A survey on the moving target defense strategies: an architectural perspective," *Journal of Computer Science and Technology*, vol. 34, no. 1, pp. 207–233, 2019.

[10] S. Sengupta, A. Chowdhary, A. Sabur, A. Alshamrani, D. Huang, and S. Kambhampati, "A survey of moving target defenses for network security," *IEEE Communications Surveys & Tutorials*, 2020.

[11] J.-H. Cho, D. P. Sharma, H. Alavizadeh, S. Yoon, N. Ben-Asher, T. J. Moore, D. S. Kim, H. Lim, and F. F. Nelson, "Toward proactive, adaptive defense: A survey on moving target defense," *IEEE Communications Surveys & Tutorials*, 2020.

[12] N. Saputro, S. Tonyali, A. Aydeger, K. Akkaya, M. A. Rahman, and S. Uluagac, "A review of moving target defense mechanisms for internet of things applications," *Modeling and Design of Secure Internet of Things*, pp. 563–614, 2020.

[13] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," *Journal of systems and software*, vol. 80, no. 4, pp. 571–583, 2007.

[14] B. A. Kitchenham, D. Budgen, and P. Brereton, *Evidence-based software engineering and systematic reviews*. CRC press, 2015, vol. 4.

[15] N. Tesla, "System of signaling." Apr. 14 1903, uS Patent 725,605.

[16] D. Kewley, R. Fink, J. Lowry, and M. 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.

[17] F. Chong, R. 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," *NITRD Program*, 2009. [Online]. Available: https://www.nitrd.gov/nitrdgroups/images/b/bd/National_Cyber_Leap_Year_Summit_2009_CoChairs_Report.pdf

[18] R. Zhuang, S. A. DeLoach, and X. Ou, "Towards a theory of moving target defense," in *Proceedings of the First ACM Workshop on Moving Target Defense*. ACM, 2014, pp. 31–40.

[19] C. Security, I. A. I. +Networking, I. T. Research, and D. Subcommittee, "Nitrd csia iwg cybersecurity game-change research and development recommendations," *Federal Plan*, 2010. [Online]. Available: https://www.nitrd.gov/pubs/CSIAIWG-Cybersecurity-RD-Recommendations-052010.pdf

[20] S. Jajodia *et al.*, *Moving target defense: creating asymmetric uncertainty for cyber threats*. Springer Science & Business Media, 2011, vol. 54.

[21] T. Hobson, H. Okhravi, D. Bigelow, R. Rudd, and W. Streilein, "On the challenges of effective movement," in *Proceedings of the First ACM Workshop on Moving Target Defense*. ACM, 2014, pp. 41–50.

[22] R. Zhuang, A. G. Bardas, S. A. DeLoach, and X. 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.

[23] R. Zhuang, "A theory for understanding and quantifying moving target defense (doctoral dissertation)," Ph.D. dissertation, Kansas State University, 2015.

[24] S. Picek, E. Hemberg, and U.-M. 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.

[25] J. B. Hong, S. Y. Enoch, D. S. Kim, A. Nhlabatsi, N. Fetais, and K. M. Khan, "Dynamic security metrics for measuring the effectiveness of moving target defense techniques," *Computers & Security*, 2018.

[26] V. Zangeneh and M. Shajari, "A cost-sensitive move selection strategy for moving target defense," *Computers & Security*, vol. 75, 2018.

[27] M. Torquato and M. Vieira, "Moving target defense in cloud computing: A systematic mapping study," *Computers & Security*, p. 101742, 2020.

[28] S. Hosseinzadeh, S. Rauti, S. Laurén, J.-M. Mäkelä, J. Holvitie, S. Hyrynsalmi, and V. Leppänen, "Diversification and obfuscation techniques for software security: A systematic literature review," *Information and Software Technology*, vol. 104, pp. 72–93, 2018.

[29] M. Sherburne, R. Marchany, and J. 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.

[30] V. Casola, A. De Benedictis, and M. 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.

[31] A. Abbasi, J. Wetzels, T. Holz, and S. Etalle, "Challenges in designing exploit mitigations for deeply embedded systems," in *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2019.

[32] C. 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.

[33] V. Casola, A. De Benedictis, and M. Albanese, *A multi-layer moving target defense approach for protecting resource-constrained distributed devices*. Springer, 2014, pp. 299–324.

[34] E. Battista, V. Casola, A. Mazzeo, and N. Mazzocca, "Siren: A feasible moving target defence framework for securing resource-constrained embedded nodes," *International Journal of Critical Computer-Based Systems*, vol. 4, no. 4, pp. 374–392, 2013, cited By 3.

[35] T. Preiss, M. Sherburne, R. Marchany, and J. Tront, "Implementing dynamic address changes in contikios," in *International Conference on Information Society (i-Society 2014)*. IEEE, 2014, pp. 222–227.

[36] K. Zeitz, M. Cantrell, R. Marchany, and J. 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.

[37] ——, "Changing the game: A micro moving target ipv6 defense for the internet of things," *IEEE Wireless Communications Letters*, vol. 7, no. 4, pp. 578–581, 2018.

[38] S. Hosseinzadeh, S. Rauti, S. Hyrynsalmi, and V. 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.

[39] S. Hosseinzadeh, S. Hyrynsalmi, and V. Leppänen, *Chapter 14 - Obfuscation and diversification for securing the Internet of Things (IoT)*. Elsevier, 2016, pp. 259–274.

[40] L. Koivunen, S. Rauti, and V. Leppänen, "Applying internal interface diversification to iot operating systems," in *2016 International Conference on Software Security and Assurance (ICSSA)*. IEEE, 2016.

[41] P. Mäki, S. Rauti, S. Hosseinzadeh, L. Koivunen, and V. Leppänen, "Interface diversification in iot operating systems," in *The 9th International Conference on Utility and Cloud Computing*. ACM, 2016.

[42] S. Rauti, L. Koivunen, P. Mäki, S. Hosseinzadeh, S. Laurén, J. Holvitie, and V. Leppänen, "Internal interface diversification as a security measure in sensor networks," *Journal of Sensor and Actuator Networks*, 2018.

[43] K. Mahmood and D. M. 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.

[44] A. Van Herrewege and I. 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.

[45] K. Andrea, A. Gumusalan, R. Simon, and H. 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.

[46] M. Ge, J. B. Hong, W. Guttmann, and D. S. Kim, "A framework for automating security analysis of the internet of things," *Journal of Network and Computer Applications*, vol. 83, pp. 12–27, 2017.

[47] M. Ge, J. B. Hong, S. E. Yusuf, and D. S. Kim, "Proactive defense mechanisms for the software-defined internet of things with non-patchable vulnerabilities," *Future Generation Computer Systems*, vol. 78, 2018.

[48] A. O. Hamada, M. Azab, and A. Mokhtar, "Honeypot-like moving-target defense for secure iot operation," in *IEEE 9th Annual Inf. Technology, Electronics and Mobile Comm. Conf. (IEMCON)*. IEEE, 2018.

[49] F. Nizzi, T. Pecorella, F. Esposito, L. Pierucci, and R. Fantacci, "Iot security via address shuffling: The easy way," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3764–3774, 2019.

[50] S. Wang, H. Shi, Q. Hu, B. Lin, and X. Cheng, "Moving target defense for internet of things based on the zero-determinant theory," *IEEE Internet of Things Journal*, 2019.

[51] A. Judmayer, J. Ullrich, G. Merzdovnik, A. G. Voyiatzis, and E. 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.

[52] D. Puthal, S. Nepal, R. Ranjan, and J. 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.

[53] ——, "Dlsef: A dynamic key-length-based efficient real-time security verification model for big data stream," *ACM Trans. Embed. Comput. Syst.*, vol. 16, no. 2, Dec. 2016.

[54] J. Dos Santos, C. Hennebert, J. Fonbonne, and C. 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.

[55] S. Yao, Z. Li, J. Guan, and Y. Liu, "Stochastic cost minimization mechanism based on identifier network for iot security," *IEEE Internet of Things Journal*, 2019.

[56] A. Eldosouky and W. 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.

[57] J. Pacheco, C. Tunc, and S. Hariri, "Design and evaluation of resilient infrastructures systems for smart cities," in *2016 IEEE International Smart Cities Conference (ISC2)*. IEEE, 2016, pp. 1–6.

[58] B. 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*, ser. MODELS '18. ACM, 2018.

[59] 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.

[60] ——, "Moving target defense mechanism for side-channel attacks," *IEEE Systems Journal*, pp. 1–10, 2019.

[61] S. Dziembowski *et al.*, "Towards sound fresh re-keying with hard (physical) learning problems," in *Annual International Cryptology Conference*. Springer, 2016, pp. 272–301.

[62] M. Denzel and M. D. Ryan, "Malware tolerant (mesh-) networks," in *International Conference on Cryptology and Network Security*. Springer, 2018, pp. 133–153.

[63] G. Itkis, "Intrusion-resilient signatures: generic constructions, or defeating strong adversary with minimal assumptions," in *International Conference on Security in Communication Networks*. Springer, 2002.

[64] M. Albanese, A. De Benedictis, S. Jajodia, and K. Sun, "A moving target defense mechanism for manets based on identity virtualization," *2013 IEEE Conference on Communications and Network Security, CNS 2013*, pp. 278–286, 2013, cited By 25.

[65] A. Marttinen, A. Wyglinski, and R. Jantti, "Moving-target defense mechanisms against source-selective jamming attacks in tactical cognitive radio manets," *2014 IEEE Conference on Communications and Network Security, CNS 2014*, pp. 14–20, 2014, cited By 3.

[66] J. Habibi, A. Gupta, S. Carlsony, A. Panicker, and E. Bertino, "Mavr: Code reuse stealthy attacks and mitigation on unmanned aerial vehicles," *Proceedings - International Conference on Distributed Computing Systems*, vol. 2015-July, pp. 642–652, 2015, cited By 13.

[67] F. Almoualem, P. Satam, J.-G. Ki, and S. Hariri, "Sdr-based resilient wireless communications," *Proceedings - 2017 IEEE International Conference on Cloud and Autonomic Computing, ICCAC 2017*, 2017.

[68] A. Kouachi, S. Sahraoui, and A. Bachir, "Per packet flow anonymization in 6lowpan iot networks," *2018 International Conference on Wireless Networks and Mobile Communications, WINCOM 2018*, 2019.

[69] S. Pastrana, J. Tapiador, G. Suarez-Tangil, and P. 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.

[70] M. Tiloca, D. De Guglielmo, G. Dini, and G. 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.

[71] R. E. Navas, H. Sandaker, F. Cuppens, N. Cuppens, L. Toutain, and G. Z. Papadopoulos, "Ianvs: A moving target defense framework for a resilient internet of things," in *2020 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2020, pp. 1–6.

[72] A. N. Kolmogorov, "On tables of random numbers," *Theoretical Computer Science*, vol. 207, no. 2, pp. 387–395, 1998.