

Review

A Survey on Recent Advanced Research of CPS Security

Zhenhua Wang , Wei Xie ^{*}, Baosheng Wang, Jing Tao and Enze Wang 

College of Computer, National University of Defense Technology, Changsha 410073, China; wzh15@nudt.edu.cn (Z.W.); bswang@nudt.edu.cn (B.W.); ellen5702@aliyun.com (J.T.); wangenze18@nudt.edu.cn (E.W.)

* Correspondence: xiewei@nudt.edu.cn

Abstract: Cyber-physical systems (CPSs) are next-generation intelligent systems that integrate computing, communication, and control. Malicious attacks on CPSs can lead to both property damage and casualties. Therefore, it is worth surveying CPS security by reviewing and analyzing the latest high-quality related works. In this paper, we provide an overview of the CPS security studies from the last five years and select 142 related works from A- or B-level conferences/journals recommended by the China Computer Federation (CCF). First, we review the main contents of the selected papers and classify them into 24 topics. Then, we analyze hotspots and trends of CPS security technologies in three dimensions: (1) architecture layers (perception, network, and application); (2) application scenarios (smart grids, health care, smart transportation, smart homes, and general grids); and (3) MADC (Measure, Attack, Defense, and Control) types. Finally, we also perform a statistical analysis in terms of paper publication times, author institutes, countries, and sponsors to show the current worldwide CPS security research situation.

Keywords: CPS security; survey; classification; architecture layer; application scenario; security attribute



Citation: Wang, Z.; Xie, W.; Wang, B.; Tao, J.; Wang, E. A Survey of the Recent Advanced Research in CPS Security. *Appl. Sci.* **2021**, *11*, 3751. <https://doi.org/10.3390/app11093751>

Academic Editor: José Machado

Received: 24 March 2021

Accepted: 17 April 2021

Published: 21 April 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Cyber-physical systems (CPSs) integrate sensing, computation, control, and networking into physical objects and infrastructure to connect them to the Internet and to each other. The reductions in hardware and computing costs make it possible to connect more embedded devices to the network and share their data. Thus, recent years have witnessed increasing developments in CPS technologies. Various application scenarios, such as power grids, secure water treatment, the Internet of Vehicles (IoV), health care, and smart homes, are implemented with CPS techniques. These technologies have brought great convenience to production and life. However, these systems are also related to personal privacy and even security. Therefore, they are expected to be robust against different existing and unknown attacks; however, these systems face many challenges with respect to attacks.

First, the complexity of CPS components, objects, and communication systems creates great challenges for the security of CPSs. Different sensors, actuators, and control systems need to cooperate and require valid authentication and attestation. Various participants make different operation requests to the controller for distinctive scenarios. Apart from traditional communication methods, more approaches (e.g., vibration, light, and electromagnetism) are utilized to transfer information between different devices. Therefore, any minor miss can lead to severe information leakage or overprivileged permissions. A better understanding of CPS security in a unified framework can offer an overall view for researchers. Accordingly, we can pinpoint the weakness of CPS and propose effective defense schemes against such attacks.

Another notable issue is the constrained resources of embedded devices, including memory, computation, and power. Many devices in CPS are deployed in physically inaccessible places and lack follow-up maintenance. Hence, these resource-limited devices cannot afford computing-intensive tasks, which makes their security more challenging. In

most embedded devices, common security mitigation is not supported. Therefore, we need lightweight cryptography methods, authentication protocols, and intrusion detection to reduce the computational costs and extend their service life. A typical example is wearable devices, which gather people's sports, health, and other routine data [1]. However, due to the restricted battery and computation capability, many manufacturers and app developers do not focus on security concerns. These privacy issues put users' privacy at risk.

The heterogeneity and constraints of CPS modules make security protection challenging. Therefore, a comprehensive survey is required to help us identify weak points and new explorations in CPS. In [2,3], the authors make a functional analysis of security properties for smart home devices. There are also surveys about intrusion detection systems [4], the Internet of Things in industries [5], and authentication schemes [6]. These surveys provide in-depth studies but do not build a comprehensive CPS security framework. Some surveys [7–9] propose cross-layer security structures to summarize CPS security. However, these works do not focus on the latest high-quality papers and point out the newest trends and shortages. Compared with previous work, our study focuses on high-level papers and proposes a comprehensive CPS security analysis framework. However, the analysis papers are recommended by CCF and CPS-related papers in journals. Some seminal work may be omitted.

In this paper, we surveyed the literature on CPS security under a unified security framework consisting of three orthogonal dimensions, as shown in Figure 1. MADC is the study of measure, attack, defense, or control. To help researchers obtain the latest trend, we studied papers published in level A/B security journals/conferences that were recommended by the CCF. We gathered related papers with the keywords (IoT, CPS and embedded) and classified them into technology categories and application scenarios with 24 subclasses. Then, we explored the challenges and trends in all the studies and identified the root reasons for them. Furthermore, we also provided a non-tech analysis of CPS security research.

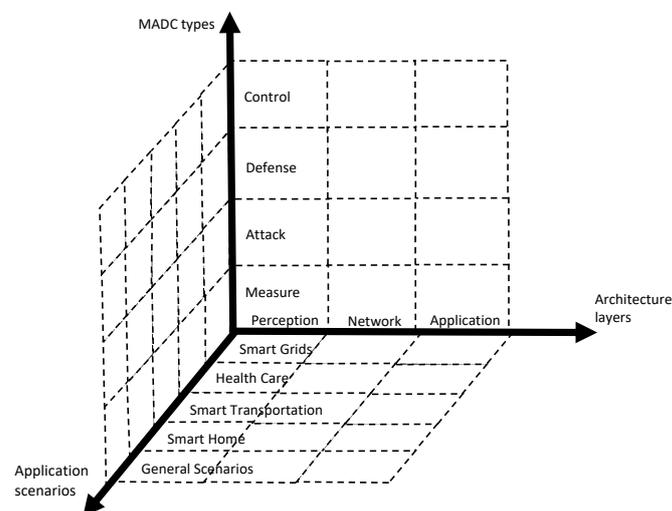


Figure 1. CPS security framework with three orthogonal dimensions.

Contribution The contributions of our work are as follows.

- We review the research progress of CPS security over the past five years based on 142 high-quality papers published in the CCF recommended level A/B journals/conferences.
- We analyze the selected papers from three aspects: architecture layers, application scenarios, and MADC types. In each subspect, we summarize the main challenges and give detailed research trends.
- We provide an overview of the global CPS security research situation in terms of publication time, organization/country of authors, and funding supporters of the papers.

The rest of the paper is organized as follows: Section 2 discusses the methodology of data collection and data classification. Section 3 reviews the 142 papers from 24 research topics and introduces each paper's primary research purposes and approaches. Section 4 analyzes CPS security with a unified security framework. In each dimension, we study the main research results and security features. Then, we show the non-tech analysis of CPS security research. Finally, our paper is concluded in Section 5.

2. Methodology

2.1. Data Collection

To reveal the recent global trends and frontiers in CPS security, we gathered related papers published in level A/B journals/conferences that were recommended by the CCF over the past five years. To discover papers about CPS security, we leveraged keywords, such as IoT, CPS, and embedded, as filters. Some conferences have a CPS security sections, which helped us to determine the related papers.

2.2. Classification

We first reviewed the main content of 142 papers from 24 concentrated research topics. Then, we classified the related papers from three dimensions: architecture layers, application scenarios, and MADC types. The horizontal axis represents the architecture layers, and at each layer, we distinguish detailed technologies. The vertical axis represents the application scenarios. They are smart grids, health care, smart transportation, smart homes, and general scenarios (cloud servers, industrial control systems (ICSs), and embedded devices). We marked different MADC types with four different colors: measure in green, attack in red, defense in blue, and control in brown. The classification results are shown in Table 1.

Table 1. Classification of related works.

	Perception Layer		Network Layer			Application Layer		
	Underlying Sensor	Lightweight Crypto	IoT Protocol	Traffic Analysis	Authentication	Access Control	Vulnerability Analysis	Trusted Computing
Smart Grids	Shekari et al. [10]			Formby et al. [11]			Dabrowski et al. [12], Garcia et al. [13], Soltan et al. [14], Huang et al. [15]	
Health Care			Chen et al. [16]	Mitchell and Chen [17]				
Smart Transportation	Cheng et al. [18]	Thomas et al. [19], Zhang et al. [20]	Van et al. [21]	Birnback et al. [22]	Ibrahim et al. [23], Abera et al. [24]	Dash et al. [25]	Sun et al. [26]	Hasan et al. [27]
Smart Homes	Choi et al. [28], Birnback et al. [29], Sikder et al. [30], Sikder et al. [31], Ronen and Shamir [32], Carlini et al. [33], Wang et al. [34]		Wu et al. [35], Rondon et al. [36], Ronen et al. [37]		Tian et al. [38], Agadakos et al. [39], Xu et al. [40], Tian et al. [41], Benadjila et al. [42]	Schuster et al. [43], Fernandes et al. [44], Jia et al. [45], Wang et al. [46], Petracca et al. [47], He et al. [48], Chen and Zhu [49]	Chen et al. [50], Zhang et al. [51], Celik et al. [52], Ding et al. [53], Celik et al. [54], Balliu et al. [55], Wang et al. [56], Kumar et al. [57], Zhou et al. [58], Waraga et al. [59]	Truong et al. [60], Chen et al. [61]
Cloud Servers		Yang et al. [62]			Fernandes et al. [63], Schulz et al. [64]	Bastys et al. [65], Fotiou et al. [66], Pereira et al. [67], Chen et al. [68], Garamvolgyi et al. [69], Shafagh et al. [70]	Wang et al. [71]	Pawlick and Zhu [72], Leiba et al. [73], Feng et al. [74]
General	ICS Lanotte et al. [75], Feng et al. [76], Krishnamurthy et al. [77], Herzberg and Kfir [78]		Yu et al. [79], Mikulskis et al. [80], Kim et al. [81], Krentz and Meinel [82], Kim et al. [83]	De et al. [84], Wu and Wang [85], Lee et al. [86], Abhishek et al. [87], Chen et al. [88], Yoon et al. [89], Stylianopoulos et al. [90], Tabrizi et al. [91]			Antonioli et al. [92], Keliris et al. [93], Corteggiani et al. [94]	
Embedded Devices	Zhai et al. [95], Anand and Saxena [96], Chhetri et al. [97], Sun and Tay [98], Hong et al. [99]	Zhang et al. [100], Shi et al. [101], Li et al. [102], Liu et al. [103], Liu and Seo [104], Mangia et al. [105], Azar et al. [106], Sancus 2.0 [107]	Celosia and Cunche [108], Zuo et al. [109], Bezawada et al. [110], English et al. [111], Cojocar et al. [112], Migault et al. [113], Han et al. [114], Sluganovic et al. [115]	Antonakakis et al. [116], Herwig et al. [117], Farooq and Zhu [118], Torabi et al. [119], Vervier and Shen [120], Fachkha et al. [121]	Asokan et al. [122], Jin et al. [123], Ibrahim et al. [124], Ghaeini et al. [125], Yan et al. [126], Gotzfried et al. [127], Ammar et al. [128], Clements et al. [129], Chatterjee et al. [130], Tan et al. [131], Kohnhauser et al. [132]	Maroof et al. [133], Rullo et al. [134], Abbasi et al. [135], Clements et al. [136]	Srivastava et al. [137], Zheng et al. [138], Muench et al. [139], Chen et al. [140], Feng et al. [141], Xu et al. [142], Gustafson et al. [143], Yao et al. [144], Nadir et al. [145]	Muhlberg et al. [146], Xu et al. [147], Xu and Capitza [148], Maene et al. [149]

3. Reviews

Before analyzing CPS frameworks, we divided the selected papers into 24 subcategories according to the type of technology or application scenarios. Then, we summarized the research background, technology approaches, and experimental results of each paper. The 24 subcategories are shown in Table 2.

Table 2. List of 24 subcategories.

Access Control Section 3.1	Authentication Section 3.2	Attestation Section 3.3	Trusted Computing Section 3.4
Security Strategy Section 3.5	Fuzzing Section 3.6	Botnet Section 3.7	Firmware Analysis Section 3.8
Apps Analysis Section 3.9	Audit Framework Section 3.10	Reverse Engineering Section 3.11	Fingerprinting Section 3.12
Anomaly Detection Section 3.13	Intrusion Detection Section 3.14	Overprivileged Permission Detection Section 3.15	Lightweight Cryptography Section 3.16
Memory Protection Section 3.17	Software Security Section 3.18	Protocol Security Section 3.19	Pairing Security Section 3.20
New Attacks Section 3.21	Data Privacy Section 3.22	Blockchain Section 3.23	Novel Defense Section 3.24

3.1. Access Control

Access control systems help ensure that the right people have access to the right places at the right times. In CPS scenarios, the focuses are control apps, underlying sensors, cloud servers, and remote blinding. Apart from security policies, researchers also leverage technologies (such as hardware-based trustworthy computing, taint analysis, and flow tracking) to achieve effective access control.

SmartAuth [38] is a user-centric, semantic-based authorization framework. First, it gathers security-sensitive information from the source code or annotations of IoT control apps and uses NLP and program analysis techniques to extract the security policy from these descriptions. Then, SmartAuth shows the required privileges to the user and requests authorization through a user-friendly interface, which can help to verify the current behavior with what the app claims to do. The authors evaluate SmartAuth on 180 available SmartApps and find that it can detect overprivileged apps with minimal performance overhead.

Schuster et al. [43] introduced a situational access control method in an IoT system. They introduced “environmental situation oracles”(ESOs), which are independent middleware between underlying sensors and access-control policies. The ESO design obeys the principle of least privilege, as ESOs cannot access the implementation details but only the abstract predicate (e.g., “at home or outsider”). Moreover, different ESOs provide the same API for clients, so they can be substituted directly to suit different policies.

SRM [16] is a secure remote monitoring framework that provides fine-grained control and privacy protection with hardware-based trustworthy computing technology (e.g., Intel SGX). To defend a packet-drop attack from an untrusted cloud server, SRM presents a lightweight “heartbeat” protocol. This protocol repeals previous entrusted key materials in the enclave if a valid heartbeat is missed in the defined time window. The authors implemented a prototype of SRM and tested it on an SGX enabled Intel platform, and the results show that it is feasible in practice and supports efficient access control over monitoring services.

Chen et al. [68] studied the security challenges in IoT remote blinding. They used a state-machine model to evaluate ten real-world remote binding cases. Then, they found many questionable designs, such as in usage of static device identifiers, weak device credentials, and weak cloud-side access control. These issues can lead to sensitive data leakage/injection, binding occupation, connection breaking, or even device hijacking. They also propose some mitigation solutions, such as the use dynamic device IDs, giving revoking permission to an individual who is already connected to the device and never delivering sensitive information during remote blinding.

FlowFence [44] is an access control framework that controls the flow between sources and sinks. It splits application codes into two modules: sensitive and nonsensitive. This framework adds extra data flows in the app structure, which can help apps deny undeclared

flows, including implicit flows. A prototype of this framework runs on an LG Nexus 4 and accepts all declared flows while rejecting the other flows with an acceptable memory overhead, QM call latency, and serialization overhead.

ContextIoT [45] is a context-based permission system that gathers context information and supports fine-grained action identification. First, it observes the control flow context of a sink in apps. Then, it uses taint analysis to label the data and judge malicious operations. The authors tested ContextIoT on 283 real-world SmartApps, and the results showed that ContextIoT can recognize all illegal contexts with negligible operation delays.

ProvThings [46] is a platform-centric malicious detector that generates a data provenance graph across different apps and devices. Unlike FlowFence [44] and ContextIoT [45], it tracks data flows while considering multiple devices and apps, allowing it to track complex interactions between IoT agents. ProvThings was evaluated on 236 SmartApps, and it traces all the provenance of known attack scenarios with negligible runtime and storage costs.

Fotiou et al. [66] proposed a fine-grained access control solution for IoT users, especially opportunistic users (guests). They used decentralized identifiers (DIDs) to grant or revoke guest privileges, and the DID documents are stored in decentralized systems (e.g., blockchains, distributed ledgers, and P2P networks). Moreover, this scheme is tracking-resistant, as guests only use their DIDs to access sources.

SEPD [67] is an access control model that addresses the challenge of policy administration and trust inspiration in a public sphere. The owner of a space announces access control policies considering users' histories of presence in the space. It also designs a policy language based on a temporal constraint network (TCN). SEPD supports anonymous resource sharing by structuring the authorization scheme in the style of a distributed trust management framework.

AWare [47] is an access control framework for Android to prevent malicious applications from abusing authorizations to gather privacy information stealthily. It links applications' operation requests with user input events and asks the user to authorize sensitive operations unambiguously. To reduce the number of explicit user authorizations, it also reuses such authorizations in duplicate scenarios. The authors implemented and evaluated AWare in a laboratory-based testbed. The results show that AWare can help users avoid authorizing unwanted operations with 2.28 decisions per application on average.

Bastys et al. [65] proposed a privacy protection scheme for IFTTT applets. They develop an access control and information flow analysis framework based on the JavaScript information flow tracking tool JSFlow [150]. Then, they classified the applet into private and public parts, thus breaking the information flow from private sources to public sinks. The authors evaluated 279,828 IFTTT applets spanning 400 services, 30% of which were related to stealthy privacy attacks.

He et al. [48] proposed a new access-control mechanism that focuses on IoT capabilities. It better fits users' expectations rather than per-device granularities. The authors set a schedule for authentication that reduces impacts from falsely allowing or denying access. In a 425-participant online user study, they found apparent differences in desired access-control policies for different capabilities among disparate groups. Their work supports richer capabilities and interactions.

3.2. Authentication

Authentication is the process of verifying whether someone (or something) is who (or what) it is declared to be. In CPS, the types and behaviors of users are very diverse. To solve this problem, researchers mainly enhance the security of authentication in protocols and mark users accurately through their behaviors.

Agadakos et al. [39] built a system, called Icelus, to locate users and model users' movements as an extra factor of authentication. Utilizing the many devices carried by users, this method overall considers a devices' location. It builds a model to identify the probability that a user is located in a particular location and is safer than models based

on a user's credentials only. Experiments with several smart devices (smartphones and wearables) show that this method has a lower false-rejection rate than smartphone-based location-based authentication (LBA) and rejects illegal access with few errors.

Current discovery protocols (e.g., Multicast DNS, Apple AirDrop, and Bluetooth Low Energy) do not consider much privacy control and often leak sensitive information. Hence, Wu et al. [35] proposed two private service discovery and mutual authentication protocols. The protocols design key primitives to encrypt the communication between the sender and receiver for verifying identity, and they can protect clients from connecting to an impostor service. Deployed in the vanadium distributed application framework, the authors measured the protocol on multiple platforms (Intel Edison, Raspberry Pi, smartphones, laptops, and desktops), and the results show that the end-to-end latency overhead is acceptable compared to that of SIGMA-I.

Virtual U [40] uses public photos of a target user to mislead 3D facial models and passes state-of-the-art face authentication. It uses VR devices to display a user's synthetic face and adds animations (e.g., blinking or smiling) to trick liveness detectors. The authors tested the proposed framework with 20 participants and on five face authentication systems. Only in two cases did all the systems withstand the attack.

VulCAN [21] is a novel vehicle authentication protocol that uses lightweight trusted computing to authenticate the message and attest the software component. This guarantees that only a chain of processing events can trigger the critical actuator event. The authors also provided an extended application scenario that shows the security guarantees against attackers with arbitrary code execution abilities on participating ECUs.

Physically unclonable functions (PUFs) are likely to be critical hardware primitive, as they can guarantee unique identities. Chatterjee et al. [130] adapt the idea combining double arbiter PUF, identity-based encryption (IBE), and keyed hash function to present a lightweight authentication and key exchange protocol. This identification protocol does not need to store the PUF response at the verifier clearly, so the main attack point in vulnerable verifiers cannot work. Utilizing the proposed protocol, the authors implement a secured video camera, which successfully resists the man-in-the-middle attack via IP spoofing.

Jin et al. [123] proposed Proof of Aliveness (PoA) to detect the survival of IoT devices. Assuming that attackers can forge aliveness proofs, they used one central authentication server with several clients and a hash chain (one-way function chains)-based authentication method to prevent replay attacks. Due to the limited length of one-way function (OWF) chains, the authors combined several OWF chains generated by a pseudorandom generator to build an efficient PoA. Moreover, they evaluated the PoA constructions on Raspberry Pis, and the proof generation time was approximately microseconds.

3.3. Attestation

Attestation is defined as trustworthy evidence or proof of something. In the case of a cybersecurity system, it means a user can be confident about what they are receiving from a device. Researchers have proposed effective attestation schemes for IoT devices and designed new attestation protocols.

SEDA [122] was the first attestation scheme for device swarms and leverages the common assumption to block physical attacks. It utilizes two new attestation architectures for embedded systems, SMART [151] and TrustLite [152], to implement two concrete instantiations, and its efficiency compared to traditional attestation was shown. With the implementations and simulations of large swarms with up to 1,000,000 devices, the results indicate that SEDA performs better than attesting each device separately.

Schulz et al. [64] proposed a remote attestation scheme, boot attestation, which is particularly considerable for lightweight and resource-limited embedded devices. In boot attestation, software integrity measurements are immediately authenticated during boot instead of working in a secure environment, thus reducing the traditional requirements for cryptography and storage. They also described extensions for key provisioning and attestation towards untrustworthy third-party verifiers to increase practicality and completeness.

US-AID [23] is an attestation scheme, particularly for autonomous and dynamic networks of IoT devices. US-AID combines continuous in-network attestation with a key exchange mechanism and Proofs-of-non-Absence to verify overall network integrity and effectively detects disconnecting physical attacks. With proof-of-concept implementations on autonomous drone swarms and extensive simulations, US-AID demonstrates its advantages in terms of energy and run-time.

Current trigger-action platforms will be compromised if the OAuth tokens are leaked and misused by an adversary. Fernandes et al. [63] proposed a decentralized trigger-action platform (DTAP), which uses the security principle of decentralized action integrity. This platform contains an untrusted cloud service and multiple trusted user clients that use transfer tokens (XTokens) to automatically obtain a rule-specific token and transmit it to the cloud service for rule execution. The authors evaluate DTAP with various micro- and macrobenchmarks, and the evaluation shows that the additional latency in rule execution time is modest while reducing throughput by 2.5%.

Ibrahim [124] proposed attestation schemes that work for large networks of embedded devices. These schemes detect remote malware infestations and physical and run-time attacks in different smart environments and autonomous systems.

Abera et al. [24] presented DIAT, a unique method that allows the identification of the accuracy of data in autonomous collaborative networks. DIAT utilizes data integrity attestation, modular attestation, and a new description of execution paths to make devices interact securely and efficiently even when some nodes are compromised. After testing a drone formation, the authors also evaluated the scheme in a simulation environment to determine its capability for large-scale systems.

PAtt [125] is a framework that utilizes remote software attestation to verify the integrity of physical processes controlled by the programmable logic controller (PLC). PAtt monitors minor changes in operation sequences to remotely verify the integrity of the control logic without a traditional trust anchor. The authors realized the proposed system on a controlled robot arm and detected PLC logic changing and spoofed sensor reading attacks with 97% accuracy.

MTRA [131] is a multiple-tier remote attestation protocol that considers both TPM-enabled IoT devices and IoT devices that cannot support TPMs. It adopts a one-way hash chain to defend against wormhole attacks as well as rainbow attacks. Additionally, it deploys an online-offline notification mechanism to protect devices from time-of-check-to-time-of-use attacks assisted by the memory randomization technique. The authors implemented MTRA on TPM-enabled devices (i.e., Odroid-XU4) and non-TPM devices (i.e., Raspberry Pi), and the performance evaluation proved that MTRA is more suitable for lightweight devices than existing remote attestation protocols.

PASTA [132] is a new attestation protocol for autonomous embedded systems, especially low-end embedded devices. It is fully decentralized, as each prover initiates a new token generation periodically to ensure freshness, which makes network disruptions or arbitrary device outages tolerable. The authors implemented the proposed protocol and conducted measurements in a simulated network. The results proved that PASTA is practical in large networks with millions of devices and is also able to robustly detect physical attacks.

Yan et al. [126] presented an efficient attestation scheme against physical attacks (EAPA) for IoT devices. EAPA uses a distributed attestation mode to attest devices by their direct neighbors, which cuts the total run-time to $O(1)$. In addition, the authors introduced an accusation mechanism to report compromised devices and designed a unique key update method, which makes the proposed scheme more efficient and secure. Compared to DARPA [153], SCAPI [154] and US-AID [23], EAPA has the lowest energy cost and run-time consumption in a large-scale network.

3.4. Trusted Computing

With trusted computing, a computer consistently behaves in expected ways, and those behaviors are enforced by computer hardware and software. Trusted computing allows devices to offer improved security over that which is currently available. How to make CPS more secure with trusted computing is a hot topic of current research.

Muhlberg et al. [146] proposed a unique method to trust assessment based on Sancus [155], a minimal hardware-only trusted computing base and protected module architecture. The trust assessment modules are deployed directly on IoT nodes, and these modules execute departures from the unprotected OS. Moreover, the modules can inspect the unprotected domain and report measurements of the node's trustworthiness to a trust management system. By assessing the Contiki OS running on a Sancus-enabled TI MSP430 microcontroller, this inspection mechanism is proven to be efficient in implementation with an acceptable runtime overhead.

Pawlick and Zhu [72] proposed a concept of trust that uses game theory to decide whether to trust a message from other components or cloud service that may be compromised. They modeled an interaction among the administrator of a cloud service, an attacker, and a device that decides whether to trust signals from the vulnerable cloud. The framework includes a simultaneous signaling game with the FlipIt game, and their equilibrium outcomes also influence each other. In the experimental period, they utilized the Gestalt Nash equilibrium (GNE) to design a trust mechanism for a cloud-assisted insulin pump. Without the help of historical data, the GNE provides a risk threshold to judge the confidence levels of messages from the cloud.

Truong et al. [60] proposed a unique trust evaluation mechanism, experience–reputation (E–R), to prevent malicious users from spreading corrupted or falsified data. In the E–R model, they assessed the quality of contributed data from users with two trust indicators, experience and reputation, and selected the most trustworthy MCS users to gather shared data. To evaluate the trust-based scheme, they deployed several recruitment schemes in an MCS testbed consisting of three types of user models, and the results highlight its strength in helping MCS services to detect intelligent malicious users. Moreover, the proposed recruitment mechanisms were implemented in a real-world IoT service, the Wise-IoT project.

Leiba et al. [73] proposed an incentivized and decentralized IoT software update delivery network, which is based on trustless proof of distribution. In this network, a vendor uses smart contracts to ensure that distributors obtain digital currency as a reward for delivering an update. To ensure untrusted data exchange, the authors utilized a zero-knowledge contingent payment protocol to establish trust between the IoT device and the software update distributor. This method can significantly increase the number of distributors, thus allowing scale out.

CIDER [147] is a novel architecture that can recover IoT devices within a short amount of time, even when attackers have taken full control of all devices. With new trusted computing primitives, namely, *gated boot* and *reset trigger*, the administrator can instruct CIDER to reset the compromised devices compulsorily and to install the patched firmware. The authors implemented a prototype of CIDER on three popular IoT platforms spanning the range from high to low end, and the evaluation shows that CIDER is compatible with current firmware and that the boot-up delay and runtime overhead are negligible.

In 2016, Chen et al. [61] proposed a new adaptive trust management protocol to help social IoT applications choose the best trust parameter settings in communication with others. Their work strikes a balance between trust convergence and trust fluctuation. The authors used two real-world social IoT service composition applications to verify their table-lookup method by analyzing the results dynamically and showed the feasibility of the adaptive trust management scheme.

Xu and Capitza [148] proposed a novel consensus algorithm, RATCHETA, which increases the maximum tolerable faulty nodes to 50% and lowers the message complexity. RATCHETA adapts a hybrid fault model, which assumes a more trustworthy subsystem among the less trusted parts, to prevent equivocation, such as sending inconsistent mes-

sages to different recipients. Moreover, it also guarantees an upper bound of the memory usage and message size and uses UDP multicast to avoid packet loss detection and retransmission. The authors implemented RATCHETA with its trusted subsystem built on top of ARM TrustZone, and the experimental results show that with 20% message omissions, it costs less than one second on average to reach a consensus on a 10-node group.

Contego-TEE [27] is a framework that prevents control spoofing attacks on a real-time embedded platform. It uses a trusted computing environment to guarantee the validity of protection mechanisms even when the host OS is corrupted. It also leverages the invariant real-time nature and domain-specific features to detect malicious control signals. Tested under a control spoof and DoS attack, the robotic vehicle equipped with Contego-TEE still worked regularly with negligible extra overhead in execution time.

PoTN [74] is a lightweight blockchain consensus protocol with proof-of-trust negotiations to identify the compromised fixed miners. With negotiation rules, a trusted random selection algorithm is introduced to select proposers and validators in a round of block creation while avoiding more communication overload of consensus protocols. As the proposers know nothing about each other, collusion to fake blocks among the proposers can be avoided. After simulating the peer-to-peer consensus processes, the results show that PoTN performs better in accuracy, network overload, and efficiency than traditional consensus protocols, Tendermint [156] and PoR [157].

Maene et al. [149] proposed and studied a new hardware method, called Atlas, to guarantee security and isolation even when the operating system of an embedded system is compromised. Atlas relies on its zero-software trusted computing base to prevent any data or code leaks and provides confidential shared memory as a secure communication channel without a dynamic key exchange. They implemented Atlas based on the LEON3 softcore processor, and the FPGA-based evaluation showed that their method leads to minimal cycle overhead at the cost of a reduced maximum frequency.

3.5. Security Strategy

A security strategy is a higher level of application design. Researchers try to balance security and availability in the whole cycle of development and maintenance.

Maroof et al. [133] proposed a software security framework, called PLuggable and Reprogrammable (PLAR). PLAR remotely monitors all devices plugged in the IoT network and reprograms the devices to match the security standard through its life cycle. Moreover, PLAR checks a device's security policies and overrides weak configurations. The authors implemented a prototype on an IP camera, and the prototype successfully detected the authentication vulnerability while mitigating it.

In a complex IoT network, users need to establish security by minimizing the cost with bounded rationality. Chen and Zhu [49] therefore introduced the Gestalt Nash equilibrium (GNE) solution concept, which models users with a sparse cognition vector. With the help of the GNE, users can measure the security level of neighboring IoTs, make mature decisions, and build their cognitive networks holistically. They further computed the GNE with a proximal-based algorithm, and the results revealed several situations that match real-world observations.

Rullo et al. [134] proposed a Pareto-optimal security resource allocation scheme to guarantee the proper functioning of an IoT-based system with reasonable resource overhead. They calculated the best defender strategy by formulating it as a linear optimization problem. Then, they measured its efficiency and effectiveness aspects. For large network topologies, they proposed a divide-et-impera method that decomposes the initial problem into smaller subproblems. After that, they used parallel computing techniques to solve the subproblems as small instances.

3.6. Fuzzing

Fuzzing can be used to discover potential vulnerabilities by sending random input to applications and observing their behavior. It can trigger unexpected vulnerabilities that

are difficult to find with conventional analysis. Researchers have applied fuzz technology to firmware, mobile apps, protocols, and other fields. Furthermore, they have improved the versatility and efficiency of the technology.

FirmFuzz [137] is an open-source framework that automatically emulates and dynamically analyzes Linux-based firmware images. It employs a greybox-based generational fuzzing approach coupled with static analysis and system introspection to help fuzz the emulated firmware images while monitoring the firmware's running status. With the help of host emulation, the authors emulated and dynamically analyzed 32 images scraped from 27 unique vendor websites. They discovered seven previously undisclosed vulnerabilities spanning six different devices: two IP cameras and four routers and reported four new CVEs.

FIRM-AFL [138] is the first high-throughput greybox fuzzer for IoT firmware. It supports three CPU architectures, mipsel, mipseb, and armel, so most firmware images can be emulated in a system emulator. With augmented process emulation, the performance bottleneck caused by system-mode emulation is solved by switching between user-mode and full-system emulation. After testing on 288 images, the evaluation results show that the throughput of FIRM-AFL can find 1-day or 0-day vulnerabilities 8.2 times higher than system-mode emulation-based fuzzing on average.

Traditional testing techniques rely on observable crashes of a program, but memory corruption is often less visible on embedded devices due to their unique architecture. Muench et al. [139] analyzed those differences on several categories of embedded devices and showed the difficulty of detecting memory corruptions. The authors further described and evaluated six heuristics that can be applied during the analysis of an embedded device to detect previously undetected memory corruptions. Based on Avatar and PANDA, they conducted numerous experiments to prove that live analysis can improve the fuzzing process of embedded systems.

IoTFuzzer [50] was the first firmware-free fuzzing framework, as many devices' firmware is not easily extracted or decoded. This fuzzing framework communicates with a device through their official mobile apps to reuse program-specific logic and to mutate the test case without relying on any knowledge about its protocol specifications. With a lightweight monitor that gathered the target IoT device's status, the authors evaluated 17 real-world IoT devices running on different protocols and successfully identified 15 memory corruption vulnerabilities, including eight new vulnerabilities.

IoTHunter [79] is the first gray-box fuzzer for fuzzing stateful protocols in IoT firmware. With a multistage message generation technique, IoTHunter can fuzz several stateful protocols (e.g., snmp, ftp, ssl, bgp, and smb) and has a high test case validation rate. The authors evaluated IoTHunter with a set of real-world programs, and the results proved that compared with black-box fuzzer-boofuzz, the performance has 2.2×, 2.0×, and 2.5× increases, respectively, in function coverage, block coverage, and edge coverage. They found five unknown vulnerabilities in the firmware of home router Mikrotik, which have been reported to the vendor.

3.7. Botnet

Due to weak protection and the huge number of embedded devices, APT organizations often control numerous embedded devices to build botnets. Therefore, it is significant to study the formation mechanism and behavior characteristics of botnets.

Antonakakis et al. [116] provided a longitudinal retrospective analysis of Mirai's growth history and its DDoS victims. The authors gathered and analyzed raw data from various aspects, including network packets, port scans, IoT honeypots, C2 milkers, DNS queries, DNS clusters, and aggregate histories of DDoS attacks. They confirmed that Mirai has launched over 15,000 attacks against both high-profile and seeming unrelated targets, such as Krebs on Security, game servers, telecoms, and anti-DDoS providers. Mirai may represent a giant change in the evolutionary development of botnets, and it reveals the absence of security in the IoT space.

Hajime is a new botnet that is similar to Mirai in IoT devices, and Herwig et al. [117] surveyed it. Hajime sends control commands and updates attack exploits through a public peer-to-peer system, which makes the botnet more resilient and quickly increases its size and power. Through detailed active scanning of Hajime's peer-to-peer infrastructure, they measured the bot size, bot churn, bot location, device composition, payload updates, and track vulnerabilities over time. Considering that there is no way to satisfactorily stop Hajime's C&C without damaging the quality of a BitTorrent's DHT, the authors made their code and data publicly available to help people patch vulnerable IoT devices.

Farooq and Zhu [118] proposed an analytical model to study the device-to-device (D2D) propagation of malware in wireless IoT networks. Inspired by dynamic population processes and Poisson point process theory, the authors set up a mean-field dynamical system to capture the malware infiltration process and control command propagation in networks. They also considered the overhead caused in patching devices and used the mean-field equilibrium in the population to solve the problem.

To clear compromised devices in the botnet, De et al. first proposed *AntiIoTic* [158], which utilizes the vulnerabilities in the infected devices and creates a white botnet to secure them. However, this method has legal issues, as the owners' explicit consent is missing. Therefore, in *ANTIBIOTIC 2.0* [84], the authors solved the problem and introduced fog computing to improve its predecessor. In the N-tier architecture of fog computing, every *ANTIBIOTIC* fog node cleans the infected device from malware and fixes the vulnerabilities (e.g., closing unnecessary ports and changing original passwords). By evaluating it on a real-world router, the authors proved the capability of *ANTIBIOTIC Bot* in securing and sanitizing.

Torabi et al. [119] studied malicious IoT device behaviors on an internet scale from a network telescope perspective. They obtained IoT device information and darknet traffic from online resources and inferred the compromised IoT devices interacting with the darknet. By characterizing the traffic generated by unsolicited IoT devices, they discovered new malware families whose target is vulnerable devices. Overall, the authors highlighted the large-scale insecurities of the IoT paradigm and pinpointed the risks of new malware variants.

In this paper, Vervier and Shen [120] studied the behavior of cybercriminals by operating low- and high-interaction IoT honeypots. They found that both the diversity and complexity of IoT botnets are increasing, while the Mirai malware family is still dominant. An increasing number of software vulnerabilities are used to exploit targeting devices, which makes the IoT malware ecosystem rapidly grow.

Fachkha et al. [121] presented a formal preprocessing probabilistic model to analyze internet-scale probing activities of more than 20 frequently used CPS protocols. This model uses likelihood models to filter out the noise of darknet traffic and leverages temporal analysis as well as context triggered piecewise hashing to report probing activities. Experimenting with 50 GB of darknet data, the authors disclosed more than 9000 hidden malicious CPS events coming from stealthy adversaries.

3.8. Firmware Analysis

Firmware analysis is mainly divided into two categories: static and dynamic analysis. In static analysis, symbolic execution and flow analysis are optimized to address firmware characteristics. In dynamic analysis, the research focus is to simulate firmware operations in a more general way.

FIRMADYNE [140] is an open-source automated dynamic analysis framework for identifying vulnerabilities in Linux-based embedded firmware. First, *FIRMADYNE* crawled from various vendor websites to download firmware images and gathered 23,035 firmware images spanning 42 vendors. Second, it used a binwalk API to extract the kernel and root filesystem to identify the hardware architecture. Next, it leveraged the corresponding QEMU full system emulator to launch the service. With 60 known exploits and 14 unknown

vulnerabilities that were discovered by the framework, it evaluated 9486 firmware images successfully, and 887 firmware images were vulnerable to one or more exploits.

Genius [141] is a numeric-feature-based search scheme that improves the effectiveness of cross-platform bug searches. It learns higher-level numeric features from representative control flow graphs (CFGs) and then searches for bugs based on the learned higher-level features. To speed up the process, it converts the CFGs into high-level numeric feature vectors, which can be indexed by locality sensitive hashing [159]. After evaluating 8126 devices with 420,558,702 functions across three architectures and 26 vendors, the results showed that Genius can execute a query in 1 s on average. Practically, Genius searched 154 vulnerabilities in the two latest firmware images from D-LINK and found 103 potential vulnerabilities, 16 of which have been confirmed.

Gemini [142] is a neural network-based method for detecting the similarity of firmware. It leverages a graph embedding network to convert the attributed control flow graph (ACFG) of images into an embedding and uses a neural network model to train the datasets of firmware and vulnerabilities. Specifically, with similarity detection, Gemini uses previous CVEs to identify more vulnerable firmware images with higher precision and lower time overhead than the state-of-the-art methods, i.e., Genius [141].

Inception [94] is a complex framework for testing complete real-world embedded firmware. The inception translator generates and merges high-level and low-level semantic code to preserve the semantics of the source code, thus improving the effectiveness of detecting vulnerabilities. Based on KLEE, an inception symbolic virtual machine performs symbolic execution, and handles memory abstractions of peripherals and interruptions with several strategies. Finally, the inception debugger redirects memory accesses to the peripherals on real hardware from a virtual machine. Evaluated on four real-world open-source and three industrial applications, Inception found eight crashes and two previously unknown vulnerabilities, which proves its ability to assist embedded image testing.

Avatar [160] can allow a user to emulate the firmware of an embedded device, which facilitates dynamic analysis. By injecting a software proxy, Avatar can execute instructions in the emulator while communicating with the physical hardware. Avatar2 [161] further supports orchestrating executions among multiple testing environments. It can organize different systems by “moving” the execution of binary code from one to another.

PRETENDER [143] is a framework that creates models of peripherals to allow the execution of firmware in a fully emulated environment. It first records the execution process in the MMIO region and locates the boundaries of each distinct periphery in the memory space of devices. Then, it uses multiple iterations of linear regression modeling to obtain proper models for each memory location. Tested on three hardware platforms with different embedded CPUs, PRETENDER successfully allows rehosting and survivable execution on six example firmware and supports smart fuzzing.

Gerbil [144] is a binary analysis framework to identify privilege separation vulnerabilities. First, it extracts the loading information from IoT firmware to know which MCU model is used in this firmware. Then, it slices the most vulnerable part of IoT firmware and uses symbolic execution to explore the execution path. With library function recognition, it avoids path explosion and reveals an indirect call to find deeper paths. Based on the firmware’s call graph, Gerbil identifies the command functions that are invoked by this overprivileged shared function. The authors evaluated Gerbil on 106 firmware images, and 69 of them have privilege separation vulnerabilities, which can lead to malicious firmware replacement or denials of service.

3.9. App Analyses

Apps are important platforms for users to view and control device statuses. While bringing convenience to users, apps also increase new attack surfaces. Researchers have paid attention to security problems in apps, such as information leakage and unauthorized access.

Previous studies have revealed various security faults, which allow malicious smart home apps to possess more privileges than in the original design, in the Samsung SmartThings platform. HoMonit [51] leverages side-channel inference capabilities to monitor SmartApps from encrypted wireless traffic. It compares the SmartApps activities inferred from the encrypted traffic with their expected behaviors obtained from their source code or UI interfaces and provides a dataset of 60 misbehaving SmartApps. After analyzing 181 official SmartApps, HoMonit finds 60 malicious SmartApps, which either perform overprivileged accesses or conduct event-spoofing attacks.

SAINT [52] is a static taint analysis tool for sensitive information tracking in SmartThings apps. SAINT first translates IoT source code into an intermediate representation (IR) and then identifies sensitive sources and sinks to detect sensitive data flows. After being tested on IoTBench, an IoT-specific test corpus containing 168 official and 62 third-party SmartThings apps, SAINT flags out 138 apps that leak at least one kind of sensitive data.

IoTMon [53] is a framework that guarantees safe interaction controls across apps. It leverages static program analysis and the natural language processing (NLP) technique to analyze intra-app actions and identify all interaction chains. IoTMon also assesses the inter-app interaction chains for their safety risk. The authors evaluated IoTMon on 185 official apps and found 162 unknown inter-app interaction chains, 37 of which posed high risks to physical environment security.

IoTGuard [54] is a framework that protects users from dangerous and harmful device states with three components: code instruments, data collectors, and security services. IOTGUARD first inserts extra logic to characterize an app's information at runtime. Then, the data collector stores the app's information in a dynamic model that represents the individual or unified behavior of the app depending on whether the app interacts with other apps. Finally, when receiving an app's information, the security service checks if the app's action passes a policy and decides to notify the app with a rejecting or passing message. IOTGUARD was evaluated on 65 real-world targets, and it enforced 11 policies and blocked 16 abnormal device states in 11 (17%) apps.

Balliu et al. [55] proposed a semantic framework to study cross-app interaction security. They presented an extensional condition as well as a syntactic condition for safe cross-app interactions. Moreover, they introduced a flow-sensitive security-type system to guarantee the confidentiality and integrity of implicit interactions and priorities. Finally, the authors used commercial apps to prove the practical effectiveness of the proposed framework.

To discover IoT devices' vulnerabilities caused by component reuse, Wang et al. [56] automatically analyzed mobile companion apps on a large scale. They collected apps from the Google Play store and then leveraged device interface analysis, imprint analysis, and fuzzy hash analysis to detect similar devices from four aspects: software, hardware, protocol, and backend services. Finally, they marked 324 devices spanning 73 vendors as potentially vulnerable, and 164 devices were confirmed, with an accuracy of approximately 50%.

3.10. Audit Framework

For complex CPS devices, we need to perform a comprehensive safety inspection. In some studies, the authors designed comprehensive testing frameworks with open source tools. These frameworks can cover physical interfaces, protocol security, firmware analysis, etc.

Kumar et al. [57] provided a large-scale analysis of the vulnerabilities in IoT devices for smart homes. In this research, they used a WiFi inspector (a tool included by antivirus products from Avast) to scan internal IoT devices and identified the device types through a set of expert rules and a supervised classification algorithm. After testing 83 M IoT devices in 16 M homes worldwide, the results show that astronomical numbers of devices use weak or default passwords on FTP, Telnet, or HTTP administration and are still vulnerable to known attacks.

Nadir et al. [145] proposed an auditing framework for vulnerabilities in IoT systems, which is based on open-source tools. This framework mainly focuses on communication,

firmware, and hardware. Communication analyses check the credential management vulnerabilities in web interfaces and network services. In firmware analysis, the framework searches hardcoded credential, backdoor or compromised third-party libraries. Hardware analysis examines the security of physical interfaces, sensors and memory. As an example, the authors analyzed an IP camera with the proposed framework and found many undetected vulnerabilities.

Waraga et al. [59] developed an extensible testbed to assess the vulnerabilities in IoT devices with open-source tools. The automatic testbed first gathers information on the target device and then uses all possible security tools (e.g., nmap, metasploit, binwalk, tshark, and SSLScan) to test the device. Finally, it generates a test report to show the security assessment results. The authors also tested a wireless camera and smart bulb with the testbed to prove its abilities.

MiniCPS [92] is a simulator framework that provides a real-time network, programmable logic controllers (PLCs), and physical-layer interactions in CPS for security research. MiniCPS uses Mininet as the lightweight network emulation part and extends it to simulate important CPS components (e.g., PLCs). Furthermore, MiniCPS provides a simple API to capture physical-layer interactions. The authors used a water treatment testbed and a custom SDN controller to test the modeling capability of MiniCPS and successfully developed offenses and defenses for MitM attacks.

Snout [80] is a pen-testing framework for various IoT protocols. It supports four simple radio protocols (ZigBee, ZWave, WiFi, and Bluetooth) for information gathering and device enumeration. Moreover, it can also assess the vulnerabilities, replay packets, and fuzz packets of specific protocols.

3.11. Reverse Engineering

As many embedded devices do not open their source code, we need to reverse the firmware to find bugs. The main challenge is in addressing the different instrument sets and identifying the logic. Recent studies have performed interesting work in IoT firmware binaries and protocols.

Many smart home platforms connect tens of IoT devices via mobile apps and IoT clouds, but they do not provide enough security safeguards. Zhou et al. [58] analyzed five widely used smart home platforms and found that the interactions among the participating entities (i.e., devices, IoT clouds, and mobile apps) have not been seriously considered. Combining firmware reverse engineering, network traffic interception, and blackbox testing, they identified a set of unexpected state transitions that could lead to remote device substitution, remote hijacking, remote DoS, illegal occupation, and firmware theft. Several new vulnerabilities were discovered in real-world smart home platforms.

ICSREF [93] is a reverse engineering framework for PLC binaries and is widely used in industrial control systems (ICSs). It can reverse varied PLC binaries without previous knowledge and provide the fingerprint of a binary. Therefore, ICSREF can be deployed to analyze PLC malware and identify malicious code authorship attribution through fingerprinting. To evaluate the correctness of ICSREF, the authors built a database of PLC program binaries with source code, which can be used as benchmarks by the community.

MISMO [26] is used to reverse embedded binary code in IoT control application domains. Specifically, reverse engineering performs a symbolic semantic-matching algorithm to match the target subroutine. The outcomes can also be used in vulnerability analysis and security mitigation. The authors evaluated MISMO on more than 2k real-world firmware binaries from six application domains (drones, autonomous cars, smart homes, robotics, 3D printers, and the Linux kernel). The experimental results proved that MISMO can accurately obtain a binary code's algorithm-level semantics and found a zero-day vulnerability in the most recent Linux kernel.

PIE [112] is a parse and processing logic identifier that checks the security of closed source embedded devices. It combines data flow and template matching to analyze parser-like binary code. When used correctly, it can identify the user input of firmware components

or extract protocols and discover memory-related bugs in input processing programs. The authors evaluate PIE on four real-world devices, and it extracted all commands from their protocol parsers, including hidden commands (backdoor).

3.12. Fingerprinting

A device fingerprint is the collected information from the software and hardware of a remote computing device for the purpose of identification. Device fingerprints play an important role in identity authentication. Based on device characteristics, researchers have designed efficient and accurate fingerprint algorithms.

Celosia and Cunche [108] investigated how to use a generic attribute (GATT) profile from Bluetooth-low-energy devices to create fingerprints. After analyzing a dataset of more than 13,000 profiles, they proved that the GATT profile can be used to identify devices uniquely. They also mentioned that sensitive information (e.g., device type, device model, device manufacturer, and user's name) can be inferred from the values of some characteristics. Finally, they suggested mitigating those issues by restricting access to values of characteristics or minimizing the exposure of the GATT profile.

Formby et al. [11] proposed two methods to generate unique fingerprints for devices in industrial control system (ICS) networks to enhance current intrusion detection methods. The first method leverages cross-layer data processing times, and the second uses their unique physical properties and operation times. Through a combination of the dataset from a live power substation and controlled lab experiments, the accuracy of the first and second methods achieve 99% and 92%, respectively. Both methods resist simple forgery attacks alongside traditional IDS systems.

BleScope [109] is a tool for generating BLE device fingerprints by extracting static UUIDs from companion mobile apps. As the devices' UUIDs and their hierarchies are hardcoded in plaintext in apps, static analysis on an app can be used for UUID fingerprinting. Through evaluations on all the free BLE IoT apps from the Google Play store, BleScope identified 1757 vulnerable mobile apps. Among the 5822 real BLE devices identified in a region, 94.6% were fingerprintable by attackers, while 7.4% of them were vulnerable to unauthorized access. Moreover, the authors proposed methods to mitigate these attacks at the app level, channel level, and protocol level, such as obfuscating the app, broadcasting disrupting signals, or generating dynamic UUIDs.

Bezawada et al. [110] presented an approach that uses IoT device behavioral fingerprinting to identify devices. Certain command and response sequences in protocols describe a device's behavior. The authors use machine learning tools to learn local features and detect similar device types. They further validated the approach with multiple machine learning classifiers, and the classifiers reported an identification rate of 93–100% and a mean accuracy of 99% across all the experiments.

3.13. Anomaly Detection

Anomaly detection is the identification of rare events, behaviors, or observations that deviate from a system's normal behavior. Unlike conventional systems, there are more operating characteristics in CPSs. In addition to conventional traffic and memory analysis, researchers also tend to use various physical features to find anomalies.

As public IoT devices still use low-level security protection strategies, Wu and Wang [85] provided a collaborative security detection approach based on game theory for distributed IoT systems. They used a consensus protocol to gather neighbors' local profiles and reached an agreement in the security scheme. The method adopts game theory to consider the confrontation between the defender and the attacker and attempts to achieve maximum security protection for the entire system. Simulations on a test network topology with 50 nodes and 297 edges proved that the approach is valuable for defending against DDoS attacks.

MDSClone [86] is a new clone detection method based on multidimensional scaling (MDS). It detects clones without relying on the geographical positions of nodes and is suitable for hybrid networks comprising both static and mobile nodes, which avoids

considering any specific mobility pattern. Moreover, three other techniques (i.e., CIPMLO, TI, and SMEBM) were proposed to speed up the core part of MDSClone, and the detection algorithm can be parallelized to improve the performance. Their comprehensive analytical and experimental evaluations showed that the clone detection probability of MDSClone is almost 100% with a shorter detection delay.

In clustering Internet of Things (IoT) networks, an adversary could attack the relay associated with all nodes to compromise the whole system. To detect this kind of attack, Abhishek et al. [87] presented hybrid intrusion detection systems by comparing the observed packet drop probabilities with their long-term expected values. The generalized likelihood ratio test proved that the unicast and broadcast models can achieve a negligible false alarm and ignorable detection probability.

Chen et al. [88] proposed a new method, which is based on analysis of runtime data logs and automatic model construction of CPS, to detect attacks before any damage is done. The authors systematically mutated software components and obtained traces of sensor data. Then, they learned an SVM-based model from the data and used the classifier to monitor the runtime status. They evaluated the proposed approach on the secure water treatment (SWaT) testbed, and the results showed that it can detect 85% of 55 network and code modification attacks from data logs generated at runtime.

Yoon [89] developed a network inspector for IoT devices to detect potential attacks. The author used IoT packet sequences as input nodes to the artificial neural network (ANN) and calculated the probability of attacks. Testing on real IoT samples with backdoor scripts showed that input nodes comprising more dimensions produce lower error rates.

Pattern matching plays an integral role in network intrusion detection systems (NIDS). However, with the increasing complexity and functionality of NIDS, pattern matching requires more time and energy consumption. Stylianopoulos et al. [90] proposed a new pattern matching architecture, which brings new and exciting opportunities for algorithm design, based on embedded GPUs. They evaluated the algorithms on a heterogeneous device and found that GPU-based pattern matching algorithms have competitive performance compared to a CPU and consume half as much energy as the CPU-based variants.

Ali et al. proposed two novel hazard analysis techniques, which are helpful in abnormal detection. In [162], the authors leveraged the Failure Modes, Effects, and Criticality Analysis to detect and prevent sensor failure with KB. In addition, in [163], the authors specially addressed the challenges that collaboration between multiple CPSs brings complexity, uncertainty, and variability. They designed a new tool (CPS Tracer) which generates the fault traceability graph to explore hazards with variability.

3.14. Intrusion Detection

Intrusion detection is a device or software application that monitors a network or systems for malicious activity or policy violations. In CPS, researchers explore intrusion with state machines, model checking, or machine learning. They apply the technology to new areas, such as water distribution systems, power grid substations, and drone systems.

As the security of a medical cyber physical system (MCPS) is essential to patients' health, Mitchell and Chen [17] proposed a behavior-rule specification-based technique to detect attackers while reducing the false alarm probability. This methodology transforms behavior rules to a state machine; therefore, a device against normal behavior can easily be checked. After testing on medical monitoring devices, the experimental results demonstrated that the intrusion detection technique can cope with more complicated and underlying attackers compared with the other two existing anomaly-based methods in pervasive healthcare applications.

Zhai et al. [95] proposed a self-organizing map (SOM)-based approach to detect abnormal program behavior in commercial off-the-shelf embedded devices, especially those that cannot be updated conventionally. The proposed method utilizes cycle per instruction (CPI) to extract corresponding program counter (PC) values and uses these to pinpoint malicious behaviors with an unsupervised SOM. Experiments on a typical

low-cost ARM-based embedded development board with 104 programs showed that this method can classify unknown program behaviors with over 98.4% accuracy.

Tabrizi et al. [91] proposed a systematic security analysis framework to detect attackers' actions without previous threat models. First, they used rewriting logic to model all the functions and define transitional rules of the states. Then, they defined attacker actions (e.g., dropping packets and replaying messages) that translate the system into an unsafe state. Finally, model checking was used to find an attacker's actions in all the sequences. In this paper, the authors took a real smart meter as an example and found astronomical numbers of attacks with cheap commodity hardware.

Lanotte et al. [75] studied the timing aspects of CPS to detect integrity and DoS attacks on sensors or actuators. To picture these attacks, they defined a hybrid process calculus and defined a threat model for both CPS and cyber-physical attacks. They also assessed the influence of a successful attacks on a CPS and estimated possible quantification of the chances of success of an attack.

Birnbaach et al. [22] demonstrated a drone detection system with cheap and easily obtained hardware to prevent drones from invading residents' privacy. The authors derived the statistical metrics of the movements of a drone from the communications between the drone and its controller. They used these data to detect attackers who tried to bypass detection by changing speed or flight patterns. After it was tested on two popular consumer drone models and with four kinds of approaches, this system was proven to detect the presence of a drone at a minimal distance of 48 m.

DICE [28] is an automatic method for detecting and identifying faulty IoT devices with context extraction. First, it computes sensor correlation and the transition probability between sensor states known as context. Then, the system monitors the sensor's status transition and analyzes sensor data to detect and identify faults. The experimental results on several real-world smart home environments confirmed that DICE can effectively detect and identify device faults of different types with high accuracy.

Feng et al. [76] proposed a novel framework that was designed to systematically generate invariant rules from industrial control system (ICS) operational logs to detect malicious behavior. In this paper, the authors used ICS's general control dynamics with several machine learning and data mining techniques to analyze the ICS's physical process variables. With two real-world tests, a water distribution system and a water treatment plant, it was proven that this framework can successfully derive broader meaningful invariant rules than those defined manually and perform better in anomaly detection.

Wang et al. [71] proposed a new method for detecting interruler vulnerabilities in trigger-action platforms. First, they leveraged an NLP-aided technique to infer interruler information flows from the manual of triggers and actions. Then, they introduced an analysis framework, called iRuler, which utilizes satisfiability modulo theory (SMT) solving and model checking to analyze information flow graphs and discover interruler vulnerabilities. They evaluated 315,393 IFTTT applets, and 66% of the rulesets were linked to inter-rule vulnerabilities (e.g., condition bypassing or blocking, action looping, action conflicts, and action duplication).

To prevent power grid attacks or reduce their damaging consequences, researchers have developed effective intrusion detection systems (IDSs) based on supervisory control and data acquisition (SCADA) networks. However, the SCADA system is still vulnerable to complicated attacks. To solve this problem, Shekari et al. [10] presented a radio frequency-based distributed intrusion detection system (RFDIDS). This system leverages radio frequency (RF) emissions to monitor power grid substation activities as a side-channel signal, which cannot be spoofed or played back. The simulation and experimental results verified that four types of extracted diagnostic information can be effectively leveraged to detect specific power grid attacks.

3.15. Overprivileged Permission Detection

Overprivileged permission means the system declares extra permissions but does nothing within its function. It is unobtrusive but can lead to information leakage or more severe attacks. Recently, researchers have proposed frameworks to detect this problem in sensors, apps, etc.

Cheng et al. [18] proposed Orpheus, an anomalous detection framework for data-oriented attacks, which is stealthy because it does not corrupt the program's control flow. To detect abnormal control behaviors caused by data-oriented attacks, they proposed an event-aware finite-state automaton (eFSA) model at the system call level. The authors evaluated their prototype in three real-world cases with data-oriented attacks, and the performance showed that the time overhead is negligible for state transition integrity checking.

Current smart devices leverage various sensors to provide friendly service; nevertheless, attackers can also gather user privacy data or transfer malware by just accessing the generic sensor API. In this paper, 6thSense [30] was proposed; it is a context-aware intrusion detection system. 6thSense creates a contextual model to monitor sensor data changes from different tasks and identify harmless and malicious sensor behaviors. It utilizes three different machine learning models (Markov chain, Naive Bayes, and LMT) to detect malicious behavior associated with sensors. After an evaluation on data from over 50 real users, 6thSense was proven to be robust against three sensor-based threats in smartphone scenarios (malicious triggers, information leakage, and data theft) with higher accuracy and lower overhead.

Event sensors play a critical role in smart homes, but the data from these sensors are not always reliable, leading to event spoofing. The goal of Peeves [29] is to learn smart home event signatures for various sensing modalities to protect against event sensor faults and complicated attackers. It recognized 22 event types with 48 physical sensors and achieved excellent classification results for 9 out of 22 events without false alarms.

Apps are widely used to control devices in smart home systems (SHSs). Nevertheless, current security mechanisms do not consider user activities and sensor-device-user interactions holistically to detect malicious behaviors. Therefore, Sikder et al. proposed Aegis [31], a context-aware security framework for smart home systems to identify malicious and benign behaviors. The authors evaluated Aegis's efficacy and performance in three real-life environments and measured Aegis' security against five different malicious behaviors. The results show that Aegis can detect bad behavior in SHSs with over 95% accuracy and secure SHSs with all kinds of smart home layouts, device configurations, installed apps, and user policies.

3.16. Lightweight Cryptography

Lightweight cryptography is an encryption method that features low computational complexity. It aims at expanding the applications of cryptography to embedded devices. Recently, researchers have proposed or improved lightweight cryptography algorithms for different scenarios.

Given the limitations of processing power, energy, and memory, a light and quick cryptography algorithm is needed in an IoT system to provide high-level security. Liu et al. [103] chose a family of lightweight elliptic curves in an IoT scenario that extracts the advantages of the emerging Montgomery and twisted Edwards curves. To solve the problem of resource limitation, the authors proposed two different versions of this algorithm: high-speed and memory-efficient versions; both versions are robust against timings and SPA attacks. In their work, they developed a new approach for energy consumption evaluation depending on the performance and communication costs between objects, which could be useful for further research.

LEDs are AES-like lightweight ciphers of two key sizes that are widely used in IoT devices. Li et al. [102] proposed a ciphertext-only fault analysis method with six distinguishers (SEI, GF, GF-SEI, ML, HW, and MAP) on LEDs in the random nibble-oriented fault model. The simulation experiments proved that the analysis could restore

64-bit and 128-bit secret keys of LEDs with over 99 percent probability by injecting fault locations into the deeper round.

Shi et al. [101] proposed an ultralightweight encryption scheme to protect resource-limited embedded devices. In an untrusted environment, this encryption algorithm is robust against white-box attacks without requiring massive memory use. It requires a small amount of static data, ranging from 48 to 92 KB for 10–32 rounds. After being theoretically analyzed, both the security and efficiency of this scheme were proven to be very good.

Given that many developers improperly encapsulate crypto functions into IoT devices, CRYPTOREX [100] is implemented to analyze the problem of crypto misuses on a large scale. CRYPTOREX utilizes stack layout recovery and dynamically updates the API list to tackle function arguments during static taint analysis across multiple architectures. When it was executed on 521 firmware images with 165 predefined crypto APIs, CRYPTOREX successfully found 679 crypto misuse issues in total, which shows that 24.2% of firmware images disobey at least one misuse rule.

Liu and Seo [104] proposed unique elliptic curves, namely, nothing upon my sleeves (NUMS) curves, and implemented the NUMS256, NUMS379, and NUMS384 curves on two kinds of embedded devices. The authors combined the individual computational strengths of the twisted Edward and Montgomery curves and the efficient pseudo-Mersenne primes. Their experimental results also showed that NUMS requires fewer memory resources than the widely used Curve25519 and executing NUMS in constant time can prevent an embedded device from timing and simple-side channel attacks.

Zhang et al. [20] proposed an efficient image encryption algorithm based on the architecture of diffusion and confusion. First, the initial conditions of the hyperchaotic Chen system were determined by plain imaging, which guarantees changes in chaotic sequences. Then, the plain images were encrypted by the chaotic sequence with pixel scrambling, image encryption, and cyclic shift. The XOR operation was also constructed to improve the correlation between pixels, thus resisting plaintext attacks. The simulation results and theoretical analysis proved that this scheme is efficient in key sensitivity, correlation, information entropy, and computation time and can defend against differential and chosen-plaintext attacks.

Compressed sensing is simple lightweight cryptography, but its security is not perfect. Mangia et al. [105] adapted an encoder to the statistical features of an acquired signal. Although attackers can partly compromise security, the authors still announced that this method shows good robustness against ciphertext-only attacks (COAs) and known plaintext attacks (KPs). With theoretical considerations and numerical evidence, the results showed that the effort needed to reveal the original signal is well beyond what an adversary could bear, especially for KPs.

TRAKS [19] is a unified key manager and distribution scheme for the European Rail Traffic Management System (ERTMS). It uses pseudorandom functions (PRFs) and a shared secret to generate a dynamic key, which reduces the deployment overhead. Moreover, it is backward compatible with the existing scheme used by ERTMS and accounts for the standardization of postquantum cryptography, as the ERTMS is designed with a long lifespan. The security of the proposed key generation protocol is discussed with a game-based approach and implementation in EuroRadio. TRAKS is proven to be as secure as the existing scheme while supporting additional key management benefits.

Azar et al. [106] proposed a communication and obfuscation management architecture (COMA) to securely activate obfuscated circuits made in untrusted foundries. COMA does not need to store the obfuscation unlock key on the untrusted chip and changes the key after each unlock attempt. It protects communications to/from the COMA-protected device with two novel mechanisms for encrypted connection: ultrasecure and ultrafast. They demonstrated that COMA reduces the area overhead by 14% compared to the latest key management architectures while allowing them to identify each IC chip and remove the need to implement a secure memory in the untrusted foundry.

As the computational resources in IoT devices are constrained, Yang et al. [62] outsourced the cryptographic operations of attribute-based encryption (ABE) to clouds for the security of message distribution. Moreover, to keep the user's attributes from being leaked by cloud services, they used multiple clouds to complete the outsourced computations. The parallel-cloud scheme provides stronger security protection than the chain-cloud scheme but is less flexible and expressive. Evaluated with two commercial cloud services and IoT simulators, the proposed scheme was proven to be reliable and competent.

3.17. Memory Protection

Memory protection is a method to control memory access rights on a computer. The main purpose of memory protection is to prevent a process from accessing memory that has not been allocated to it. Researchers have applied commonly used memory analysis techniques to embedded devices and make new discoveries.

As the number of low-cost embedded devices increases, integrating hardware-based security mechanisms into such devices becomes challenging. Ammar et al. [128] provided a new concept of a security microvisor ($S\mu V$), which uses software virtualization and assembly-level code verification to provide memory isolation for such lightweight devices. This work also proposed a software-based remote attestation to detect malware-infected devices. After testing on an 8-bit AVR microcontroller, the results proved that $S\mu V$ guarantees security against memory crashes and maintains low costs in memory and power consumption.

EPOXY [129] is an LLVM-based embedded compiler that protects low-cost bare-metal systems with a new technique called privilege overlays. EPOXY adds a privilege overlay to enable privileges before instruction execution only when the operations are identified, thus separating the implementation of security decisions from the application design. A prototype implementation of EPOXY was evaluated on 75 benchmark applications and three real-world IoT applications. The performance results showed that EPOXY only has a 1.8% increase in execution time and a 0.5% increase in energy costs on average.

Abbasi et al. [135] investigated the deployment of common exploit mitigations (e.g., ESP, ASLR, and stack canaries) on 42 major embedded OSs. The results show that most lower-end embedded (so-called deeply embedded systems) OSs do not have exploit mitigations. They then introduced a mitigation method, called μ Armor, to bring mitigation baselines to the constrained embedded systems, and the overhead of the total resources was less than 5%.

English et al. [111] presented a series of PoCs for the DNS proxy module of Connman, a widely used network connection manager in IoT firmware. They used a crafted DNS response packet to crash the proxy module, which could lead to denial-of-service or remote command execution. These PoCs can successfully attack x86 and ARMv7 architectures under memory protection. With little modification, this code can be used to exploit stack overflow in other protocols.

ACES [136] is an extension of the LLVM compiler for enhancing the security of applications running on bare-metal systems. It first analyzes the bitcode generated by LLVM to draw a program dependence graph (PDG). Then, ACES utilizes a user-selected policy to compare different functionality codes and data. In a study of typical IoT applications, the test results showed that ACES compartments protect control-flow integrity in different compartments and reduce ROP gadgets with little runtime overhead.

3.18. Software Security

Software security is an idea implemented to protect software against malicious attacks and other hacking risks. For CPSs, researchers have designed security architectures to enhance security in IoT software. These architectures consider the balance of performance and safety.

After being first proposed in 2013 at USENIX, Sancus [155] is a security architecture that supports security extensibility in third-party software for an IoT network with a

hardware-only trusted computing base (TCB). In recent years, many types of research have been conducted with Sancus to guarantee the security of IoT devices. Furthermore, Sancus 2.0 [107] with an updated design and implementation was proposed and supports private deployment and more efficient cryptography. The authors developed and evaluated a prototype FPGA implementation to evaluate this scheme in proximal, colocated, and remote settings under surveillance, and the method was proven to defend against acoustic side-channel attacks.

Soteria [127] is a lightweight open-source solution for offline software protection. It is implemented based on Sancus [155], which provides the isolation, remote attestation, and secure linking of software. In addition, it ensures the confidentiality and integrity of software with hardware-supported integrity checks, which use loader modules to decrypt a protected software module only when the integrity of both is not compromised. After it was evaluated on an openMSP430 extension, the results showed that Soteria effectively supports confidential loading of code with only small load-time overhead.

3.19. Protocol Security

A protocol is a set of rules and conventions prescribing how participant components should communicate. Any data traffic between a device and the cloud (including information transmitted via mobile apps) should be examined to ensure that it is secured.

Due to the heterogeneity of IoT deployments, the capability to securely (re)program embedded devices over the air is confined, so SEDA [83] is designed to address confusion caused by over-the-air updates for various devices and firmware. To enable the secure multicast approach, SEDA modifies a well-known asymmetric broadcast encryption scheme, BGW, to improve communication and computation efficiency. An experiment conducted by the authors in Flocklab with the Cooja simulator proved SEDA's security against identified adversary models.

Rondon et al. [36] analyzed the vulnerabilities in high-definition multimedia interface (HDMI) networks and proposed an attack framework, HDMI-WALK. This framework leverages the protocol vulnerabilities in consumer electronics control (CEC), an important HDMI component, to implement five kinds of attacks: malicious CEC scanning, eavesdropping, WPA/WPA2 handshake theft, CEC packet sniffing, and broadcast DoS. Moreover, the authors also discussed defense mechanisms to mitigate HDMI-WALK attacks, such as removing CEC functions if unnecessary, designing an IDS for CEC, or applying machine learning-based approaches to identify abnormal behaviors.

Kim et al. [81] investigated several prominent IoT protocols and found many critical security problems in practical IoT settings. They used formal symbolic security models to analyze problems of such protocols under traditional or stronger adversaries. In particular, they extended the formal analysis to cryptographic denial-of-service (DoS) attacks and showed that IoT networks can be easily flooded with fake signatures or session reinitialization requests. With EC-JPAKE as an example, they used an AES-brute-force-based server puzzle (SP) and the cookie approach to defend against DoS attacks with lower computation and communication complexities.

Migault et al. [113] first proposed ESP header compression (EHC), a novel lightweight protocol that compresses ESP packets, and defined how to compress IPsec ESP and clear-text data for IoT scenarios. Then, they presented an EHC strategy, Diet-ESP, which has small energy overhead compared to the standard IPsec ESP and non-encrypted communications. As many IoT sensors continue working until their battery dies, Diet-ESP doubles the sensors' lifetime.

Ronen et al. [37] proposed a new attack model on smart lamps with bugs in the ZigBee protocol. They mainly focused on the Philips Hue bulbs, which use ZigBee Light Link (ZLL) as their communication protocol and provided over-the-air (OTA) upgrade functions. First, they used power analysis to recover the ZLL master key and the OTA update keys. Then, they generated valid encrypted and authenticated firmware to infect the adjacent

lamps. Therefore, with enough initially compromised lamps, a lightbulb worm can spread in a chain reaction over a large scale.

Krentz and Meinel [82] proposed the adaptive key establishment scheme (AKES), which makes handling reboots securely in 802.15.4. As the loss of anti-replay data and frame counters stored in RAM causes pairing problems, AKES stores these data in nonvolatile memory. AKES self-adaptively broadcasts HELLO messages to discover neighbors and start a new session with each neighbor to solve the storage issue mentioned above. The assessment results showed that the proposed scheme reduces the energy and memory consumed to establish session keys.

3.20. Pairing Security

Device pairing builds a secure communication channel between two previously unassociated devices communicating over some insecure channel. Researchers prefer to design secure and efficient pairing protocols that are suitable for various scenarios.

Perceptio [114] is a novel context-based pairing mechanism for IoT devices with different sensor types. Devices that are always deployed in the same place can perceive more of the same events over time. This mechanism uses sensory perception of events to calculate event fingerprint similarity as a threshold to detect attacker devices. The perception protocol is evaluated in different settings, and the results show that the fingerprint similarity between legitimate devices is 94.9% on average, which is higher than that of the invalid device (68.9%).

HoloPair [115] is a secure and usable pairing protocol for augmented reality (AR) headsets. First, the devices exchange their public keys over the insecure high bandwidth and then submit and reach an agreement on a specific instance of a weak hash. Finally, relying on an AR display's unobservability, the participants confirm the authenticity of the exchanged keys in the low-bandwidth visual channel. The authors accomplished a prototype of the system with two HoloLens devices and implemented a comprehensive user study with 22 participants. As the experiments show, most participants detected man-in-the-middle attacks and confirmed successful pairing in 8 s.

Anand and Saxena [96] developed a noisy vibration pairing scheme to cloak vibration sounds during pairing against eavesdropping attacks. In this scheme, the speaker emits a band-limited white noise-based masking signal and low-frequency tones through an external speaker during key transmission. The "colocated" adversary can compromise the on-board motion sensors (e.g., accelerometer) and learn the transferred secret via vibrations. Therefore, we need to inject fake accelerometer readings to cover the actual vibration effect.

3.21. New Attacks

While new CPS technologies are applied in many production and life scenarios, new attack methods also arise. Researchers study these attacks in depth and propose mitigation measures. Among the areas that receive the most attention are power grids, convert channels, USB security, and vehicle security.

3.21.1. Power Grid

Dabrowski et al. [12] proposed coordinated load-changing attacks based on the assumption that an adversary can create power fluctuations to attack a power grid with a botnet of zombie computers. The authors developed three different attacks against the power grid: static load, dynamic load, and inter-zone attacks. Their simulations show that between 2.5 and 9.8 million infected devices are sufficient to strike the European synchronous grid without relying on smart grid features to modulate power consumption.

HARVEY [13] is a new PLC rootkit whose target is the industrial power grid. It replaces the legitimate control commands from PLC output modules with optimally calculated malicious commands to the physical plant's actuators. HARVEY is implemented with a commercial PLC controller and evaluated on a real-world power grid testbed. The delay caused by attack logic is much less than the normal control logic, so HARVEY is feasible.

Soltan et al. [14] studied the collective effects caused by an IoT botnet of high wattage devices and found that such a botnet could lead to local outages or large-scale blackouts, depending on the scale of the compromised devices. The authors revealed a new type of potential attack on power grids: manipulation of demand via IoT (MadIoT) attacks and divided the MadIoT attacks into five variations. The authors evaluated the attack effectiveness via advanced simulators on real-world power grid models. The results proved that such attacks can also be used to damage a grid while benefiting a few utilities in the electricity market.

Huang et al. [15] re-evaluated the impact of MadIoT [14] attacks when power transmission grids encounter such attacks. A novel cascading outage analysis tool was leveraged to determine how the protection equipment in the power grid and protection algorithms respond to cascading events that could lead to a power blackout. The authors applied their tool to an extensive North American regional transmission interconnection system, including more than 5000 buses, to study the influence caused by MadIoT attacks. The authors found that embedding protections in the transmission grid's operation can prevent a system blackout.

3.21.2. Covert Channel

Ronen and Shamir [32] noticed that smart lights controlled remotely can be used to leak data from a highly secure building or even trigger seizures. Utilizing original APIs, they changed the lightness of LED bulbs without sharp changes, created a receiver to measure the exact duration and frequency of those flickers and then obtained the leaked information. After testing on different systems, including expensive systems (Philips HUE) to cheap systems (LimitlessLED), the authors found that this attack can be performed from over 100 m away and leak more than 10 KB per day.

Carlini et al. [33] discussed a new attack that utilizes hidden voice commands, which are unintelligible to human listeners but are interpreted as commands by a voice assistant. The authors evaluated these attacks under a black-box model and white-box model. Whether an attacker has complete knowledge of the speech recognition system or not, they can attack the system through a hidden voice that contains malicious commands. They also proposed passive and active defense methods to mitigate these attacks, including a protocol for informing a user when a device accepts a voice command, a verbal challenge–response protocol, and a machine learning method that can detect these attacks with 99.8% accuracy.

Krishnamurthy et al. [77] proposed a new method to use the analog emissions of physical instrumentation (e.g., actuators and sensors) to leak sensitive or process-specific information through side channels, such as power, electromagnetic, thermal, and acoustic channels. They also considered the controller's dynamics and its closed-loop characteristics to avoid triggering the CPS alarm system. They demonstrated the control-theoretic approach on a hardware-in-the-loop (HITL) testbed and emulated a chemical process plant's benchmark industrial control system. They successfully sent a message over the acoustic side channel with a motor and received it.

Chhetri et al. [97] tried to mitigate the information leakage problems in smart manufacturing systems. They optimized the design variables and machine process in the cyber-domain (e.g., slicing and tool-path generation algorithms) with the help of leakage-aware computer-aided manufacturing (CAM) tools. In a real-world manufacturing system, this methodology reduces mutual information by almost 30% in acoustic, power, and magnetic side channels and by more than half in vibration side channels.

Herzberg and Kfir [78] described a new scenario in which a sensor can coordinate with an actuator to launch an attack and built a provably secure covert channel from a chatty sensor to a corrupt actuator via only a threshold-logic control loop. The sensor leaks messages that are encoded with the time of the transition by adding small values of the reported measurements to the controller. Through the simulation developed by Ali [164] for a wastewater treatment process, the results showed that, even by anomaly-detection systems, corrupt covert sensors and covert actuators cannot be detected.

Wang et al. [34] proposed a novel approach that uses low-cost motion sensors to detect hidden voice commands against voice assistants. The authors leveraged audio and vibrant features to generate unique signatures, which can be easily deployed on smartphones with onboard motion sensors. Then, they used machine learning approaches to identify the statistical acoustic features and distinguish hidden voice commands from normal commands. The experimental results showed that the accuracy reaches 99.9% with low-cost motion sensors.

3.21.3. USB Security

GoodUSB [41] is a mediation architecture for the Linux USB stack that prevents USB stack from exposing a set of unrestricted device privileges and defends against BadUSB attacks. It verifies a new device's claim with a user's expectation, and, if verification fails, the device will be redirected to a honeypot virtual machine. The authors tested GoodUSB against rubber ducky and tensy, which are popular BadUSB tools. It blocks the actions of these devices' firmware with only nine more microseconds in the enumeration process on average.

WooKey [42] is an open-source USB encrypting mass storage device designed for user data encryption and protection. It combines a two-factor authentication scheme and a security-oriented microkernel implemented in Ada/SPARK to prevent pre-authentication attacks and memory corruption. Even with in depth protection, the read/write speed of Wookey is not affected.

3.21.4. Vehicle Security

Dash et al. [25] introduced a new stealthy attack model to robotic vehicle (RV) systems. They proposed false data injection, artificial delay, and switch-mode attacks against RVs under the surveillance of IDS as well as the corresponding launching algorithms. These techniques analyze the predefined detection threshold and monitoring window and adjust their input parameters to bypass detection. The authors also launched attacks on two real-world RV platforms and found that the attacks can deviate a drone from its orbit or cause it to crash when landing.

3.22. Data Privacy

CPS has reached unprecedented levels of performance and efficiency. However, it does not address security and privacy problems properly. Researchers have tried to balance system availability and data privacy through their studies.

To protect the privacy of a sensor's raw observation data, Sun and Tay [98] introduced the concept of privacy implications and an optimization framework considering both data and inference privacy. The authors proposed a two-stage local privacy mapping for each sensor and investigated the relationship between observation data and inference privacy metrics. The simulation and experimental results proved that such an architecture could protect privacy information and local differential privacy with affordable budgets.

Hong et al. [99] proposed an approach, namely, attacker location evaluation-based fake source scheduling (FSSE), to protect source location privacy in CPS. FSSE first builds a backbone to form a source–sink path regarding the source location privacy and transmission delay. Then, it selects nodes as the fake source to assess probable attacker positions. Through analysis and simulation, the authors showed that this method performed better than compared algorithms in stable privacy level, transmission efficiency, and energy consumption.

3.23. Blockchain

Blockchain is based on principles of cryptography, decentralization, and consensus, which ensure trust in transactions. It has unique safety features, so researchers have tried to integrate blockchain technology into CPS security.

Garamvolgyi et al. [69] proposed an initial approach to coordinate the usage of CPS elements from UML statecharts with generating smart contracts. The authors expressed the state of a CPS with UML statechart simple states and composite states (e.g., history states). While the current target platform is Ethereum, their approach can easily be extended to other blockchain platforms. The smart contract can define an authenticated participant given an action in the whole security system.

Safagh et al. [70] presented a blockchain-based access control framework for the IoT, which is tailored for distributed IoT data streams. They used the blockchain as an auditor to enable secure data storage and sharing. Secure and resilient access control management was enabled by utilizing the blockchain as an auditable and distributed access control layer to the storage layer. The system also addresses the challenges of efficient key distribution and management by utilizing cloud storage resources as storage nodes.

3.24. Novel Defense

In most scenarios, network attacks can not be discovered timely, and we need a more effective mitigation. Recent research focuses on the FORGE system, which generates fake but believable documents to impose additional costs on the attacker.

Chakraborty et al. [165] designed a simple but novel method to delay attackers who want to steal important documents. The authors leveraged multi-layer graphs (MLGs) to represent target documents. With this MLG representation, they proposed two novel contributions: firstly, using Meta-Centrality (MC) to measure the importance of a concept and secondly, generating a set of fake documents from a single one via Integer Linear Programming.

Han et al. [166] extended the forge work from textual component to non-textual components. The authors proposed the concept of a Probabilistic Logic Graph (PLG) which could express different parts of a document, such as charts, equations, and formulas. They also designed an approximation algorithm to address the problem of generating fake PLGs, which can effectively deceive an adversary.

4. Discussion

In this section, we analyze CPS security under a unified security framework. In Section 4.1, the technologies from three layers, perception, network, and application, are analyzed. In Section 4.2, the main challenges and mitigation in typical scenarios, such as smart grids, health care, smart transportation, and smart homes, are studied. Many studies did not target specific application scenarios, so we divide them into general scenarios, mainly including cloud servers, ICSs, and embedded devices. In Section 4.3, we summarize measure, attack, defense, and control for each representative CPS system. Moreover, we count the non-tech characteristics of the related papers in Section 4.4. Through the discussion, we can see the hotspot and trend in CPS security clearly.

4.1. Architecture Layers

4.1.1. Perception Layer

In the perception layer, the primary mission is to identify and measure objects and then collect and process the state information. Therefore, sensor attack and defense technology are the key research directions.

Accurate sensor data are the premise of CPS safety, without which a controller can be deceived and make incorrect decisions. Therefore, most sensor nodes need effective protection measures. To detect abnormal control behavior, researchers have proposed event-aware finite-state automaton (eFSA) [18] or context extraction [28] techniques. Furthermore, devices should be protected against the leakage of sensitive information to prevent sensor information abuse [98,99]. As some sensors can gather acoustic signals that humans cannot hear, an attacker can inject malicious commands with hidden voice commands [33,34]. To defend against such attacks, vendors should filter unnecessary audio channels in advance.

4.1.2. Network Layer

Network layers are used to receive the gathered data and transmit them to the control center or applications. As the basis of networks, protocols among diverse devices vary and face serious challenges. Encryption technology in a network needs to be improved for embedded devices. Moreover, traffic analysis is also significant in security defense.

(1) IoT protocol: Various devices (router, gateway, switching, cloud server, etc.) link objects via various protocols, such as ZigBee, ZWave, WiFi, Bluetooth, and HDMI (high definition multimedia interface). They can be divided into two basic types: network and data protocols, which should guarantee the integrity and credibility of packets when transferring data. Under the guidance of this principle, researchers have provided more secure protocols in patient monitoring [16], vehicle authentication [21], and house controls [35]. They have also investigated the vulnerabilities in existing widely used protocols and proposed corresponding solutions, such as HDMI [36], ZigBee [37], and EC-JPAKE [81].

(2) Lightweight Crypto: Cryptography is the cornerstone of information security. As the computation and memory resources of embedded devices are limited, these devices require more lightweight encryption algorithms. Some studies proposed lightweight encryption schemes for secure communication among resource-limited devices [101,103,104] or image encryption [20]. They also investigated the problem of crypto misuses [100] and performed ciphertext-only fault analysis on lightweight ciphers [102]. In addition, to efficiently manage a secret key among devices deployed distributively, a key manager and distribution scheme [19] is also needed to establish trusted communication.

(3) Traffic Analysis: Through traffic analysis, intrusion behaviors or abnormal statuses can be detected to help block an attack before it occurs. Typically, rule-based matching is the easiest and fastest way to detect attacks, but it cannot adapt to new attack models. Recent studies have attempted to solve this problem with new approaches, such as behavior-rule specification-based techniques [17], artificial neural networks (ANNs) [89], pattern matching accelerated by GPUs [90], and model-checking [91]. Moreover, they also leveraged fingerprinting techniques to enhance the intrusion detection scheme [11], as minor differences in devices can be used to identify spoofed command responses. These methods are well adapted to specific application scenarios with affordable overhead.

4.1.3. Application Layer

The application layer is the most comprehensive and interactive layer. It receives data from the network layer and utilizes them to support the required applications. In this process, the most important thing is to ensure the data's credibility, which requires effective authentication methods. As the participants and privileges in the IoT are more complex than ever, we also require access control schemes to protect systems. Vulnerabilities in applications must be found and repaired in time. Moreover, trusted computing in CPS is also an essential research field.

(1) Authentication: The main task of authentication is to verify the identity of users or devices. In CPS, more complicated factors are needed to verify an object. In the latest studies, users' movements [39], proofs of nonabsence [23], and modular attestation [24] are all considered attestation factors. More authentication methods can be used in CPS, but they also provide a broader attack surface for attackers. With the wide use of biometric identification, an attacker can leverage a target user's pictures to create 3D facial models and trick camera-based authentication systems [40].

(2) Access Control: Access control is widely used in security management to stop unauthorized users and formulate policies for granting privileges. The two most prominent scenes on access control are cloud services and mobile apps. More devices can handle computing and data storage missions to cloud services. To check their security issues, researchers have studied access control models in remote binding [68], IFTTT (if this then that) services [65], and resource sharing [67]. As more vendors develop mobile apps to control their devices, the access control issues in these apps are also worthy of attention. The design of apps should follow the principle of least privilege [43], and application

codes should be divided into sensitive and nonsensitive modules [44] to block undeclared flows. To support fine-grained context identification, researchers can track an app's data flows [45], even across multiple devices and apps [46].

(3) Vulnerability Analysis: To find vulnerabilities in CPS devices, we can learn technology from traditional software analysis, but the approach faces new challenges. Traditional analysis techniques, such as fuzzing [50,137,138], similarity detection [142], and control flow analysis [141,142], are still efficient in discovering unknown firmware bugs. Nevertheless, the various instruction sets and CPU architectures present new challenges. To solve this problem, researchers have developed effective reverse tools for mobile apps [58], PLC binary tools [93], and firmware binary tools [26,112] to identify processing logic. Additionally, a simulation environment [92,164] was also implemented for real-time networks and instruction simulations.

(4) Trusted Computing: In a real CPS environment, it is impossible to repair every device and block all attacks. In fact, some devices are no longer supported by vendors, as they have exceeded the service life. Therefore, we need trust computation to minimize the impact of a partially damaged system. Through security and isolation architecture, we can protect the data even when the host OS is compromised [27,149]. For each node in CPS, managers can judge their credibility through the experience–reputation model [60], game theory [72], or the trust assessment module [146]. Moreover, trusted computing could also recover faulty IoT devices in an untrusted environment [73] or even when attackers have control of all devices [147].

4.2. Application Scenarios

As CPS technologies have been widely used in multiple scenarios, smart grids, health care, smart transportation, and smart homes are hot topics in research. There are also some studies on more general areas, including cloud servers, industrial control systems, and embedded devices.

4.2.1. Smart Grids

A smart grid embeds more intelligence into a power grid and can monitor and control the grid in real time. Through these sensors and controllers, a smart grid can improve the reliability and efficiencies of operations. Moreover, smart grids can also handle bidirectional energy flows, which allows distributed generation and reduces power consumption.

Smart grids are responsible for the energy scheduling of numerous devices, and the greatest challenge is load-changing attacks. With the assistance of botnet, adversaries can create power fluctuations to paralyze a power grid [12]. Researchers have evaluated the impact carefully [14,15] and found that attackers could create large-scale blackouts by changing the energy load quickly. To defend against this kind of attack, they proposed a monitoring scheme that leverages radio frequency [10] to observe a power grid's status. This scheme can detect specific types of power grid attacks.

4.2.2. Health Care

The health care system uses various wearable/implantable devices to monitor and assess a patients' health conditions. The data can be accessed by a doctor remotely to give medical advice. Nevertheless, cyber-criminals can steal personal health information or interrupt the process of healthcare delivery. To protect the data, security professionals built a remote monitoring framework to provide fine-grained access control protection [16]. Moreover, researchers also leveraged specification-based techniques to detect the intrusion behaviors of medical devices [17].

4.2.3. Smart Transportation

Smart transportation integrates traffic data collection, transmission, and analysis technologies. These technologies are supported by various sensors and powerful electronic control units (ECUs). Smart vehicles and drones can cooperate in a wireless network and

therefore make traffic more efficient. All driving operations require the assistance of sensor information and control commands, which are also vulnerable to attackers.

In transportation security, it is essential to detect abnormal behaviors and prohibit unauthenticated operation. To achieve this goal, researchers have utilized the statistical movement metrics of a drone [22] and an event-aware finite-state automaton (eFSA) model [18] to detect hidden attackers. Even when nodes in a drone formation are compromised, safe interactions can be guaranteed by data integrity attestation [24].

4.2.4. Smart Homes

In smart homes, cameras, routers, water heaters, and light bulbs are connected to the home LAN, which can be controlled easily by mobile apps. Such smart devices operate in a relatively trusted environment, so many manufacturers do not give top priority to safety. However, security problems can still cause considerable damage to people's privacy.

The main security research issues are sensor privacy collection and app vulnerabilities. Researchers have proposed many frameworks that focus on sensor security. Peeves [29] and 6thSense [30] monitor the changes in sensor data to avoid event sensor faults. Aegis [31] checks sensor-device-user interactions to identify malicious behaviors. In app security checks, static taint analysis [52], inter-app interaction chain analysis [53], and cross-app interaction analysis [55] are useful in finding dangerous actions. Interestingly, as many components are reused in different apps, similarity detection [56] is efficient in vulnerability analysis.

4.2.5. General Scenarios

(1) Cloud Server: More devices hand over partial calculations and logical judgments to the cloud server. This operation enhances the computing power of devices, but it also leads to more security problems. IFTTT (if this then that) is a typical third-party application that has attracted security researchers' attention. Researchers provides a privacy protection scheme for IFTTT applets [65] and detected interruler vulnerabilities in them [71]. As a decentralized CPS network is compatible with the blockchain's design concept, the number of related studies has increased rapidly. Blockchain techniques, such as smart contracts [69,69], consensus mechanisms [74,148], and decentralized systems [66,70,70], have been applied to CPS security and have achieved good results.

(2) Industrial Control System: ICS refers to distributed control systems in many industries, such as water treatment [88] and IP cameras [130]. Programmable logic controllers (PLCs) are popular controllers designed for the industrial environment. To study the safety problems in PLC, researchers have designed MiniCPS [92], a PLC simulator; HARVEY [13], a PLC rootkit; and ICSREF [93], reverse engineering for PLC binary. Usually, ICS uses wireless or wired communication protocols according to the surrounding environments. Therefore, to discover unknown bugs, researchers have fuzzified several stateful protocols (e.g., snmp, ftp, ssl, bgp, and smb) [79,167] and evaluated simple radio protocols (e.g., ZigBee, ZWave, WiFi, and Bluetooth) [80]. They also optimized these protocols, such as the 802.15.4 protocol [82], constrained application protocol (CoAP), and MQ telemetry transport (MQTT) protocol [81], to enhance their reliability and robustness.

(3) Embedded Device: Embedded devices are usually parts of larger devices, such as switchers, routers, and gateways, for specialized tasks. Many of them are designed with a specialized OSs and are optimized to minimize computation and memory usage. As a result, traditional PC security solutions cannot address the challenges in embedded devices. There is not a one-size-fits-all security approach to embedded devices. Therefore, we must consider the risk of attack, possible attack vectors, and the cost of implementing a security solution. Features that need to be considered are listed in Table 3.

Table 3. The security features of embedded devices and the corresponding implementation.

Security Feature	Implementation
Access control	Unauthorized access to the device could be forbidden by a user-selected policy [136] or allocating security resources carefully [134].
Authentication	Researchers provided several attestation schemes for different purposes, such as autonomous and dynamic networks of IoT devices [23], device swarms [122], the integrity of physical processes [125], and the defense against physical attacks [126]. Moreover, memory protection schemes [128], proof of aliveness [123], and offline software protection [127] are also leveraged for authenticating communication between devices.
Secure communication	We encrypt communication from/to devices with lightweight crypto [102–104,104]. Insecure encryption algorithms should be avoided [100].
Intrusion & abnormal detection	System features could be leveraged to detect intrusion, such as cycle per instruction (CPI) [95].
Security management	Check a device's security policies and replace the weak configurations [133] to mitigate against known attacks.
Hidden fingerprinting	As an attacker can launch attacks based on the type of devices, hiding the fingerprinting information of devices can also protect them [108,109].

4.3. MADC Types

To construct an integrated prevention and control system, we need to understand the complete offense and defense processes. Researchers also do much work in security measures, new attack exploration, defense technology studies, and control framework updates. We summarize their works on MADC types: measure, attack, defense, and control.

4.3.1. Measure

The security measure is essential in CPS security and mainly includes misconfiguration detection, botnet discovery, and vulnerability analysis for apps, firmware, or ICSs. Security researchers have conducted large-scale measurements of existing security risks and potential program vulnerabilities in the CPS ecosystem. All of these studies help us to improve the security of CPS.

Security configuration mistakes are typical but cause severe security threats. The common situations are the misuse of cryptographic functions [100], hidden commands (backdoor) [112], and the lack of common exploit mitigation (e.g., ESP, ASLR, and stack canaries) [135]. Botnets have become one of the greatest threats to CPS security. Researchers have surveyed powerful botnets, such as Mirai [116] and Hajime [117]. They have also proposed analytical models to study the propagation mechanisms of botnets [118] and malicious behaviors on an internet scale [119]. Of these analyses, realistic honeypot [120] and efficient traffic analyses [121] are very useful. Interestingly, in [84], the authors leveraged a white botnet to recover compromised devices, which is a novel but efficient solution.

4.3.2. Attack

To improve defense systems, researchers also need to develop new attack methods. We categorize the attacks based on the targets' location. Attacks whose target consists of sensors and actuators are considered perception layer attacks, while attacks that focus on specific scenarios are application layer attacks. Attacks that focus on various protocols belong are network layer attacks.

In the perception layer, attackers try to steal information or send malicious commands through various covert channels. An adversary could remotely receive data by changing the lightness of smart lights [32], monitoring covert acoustic channels [77,78], and utilizing hidden voice methods to inject malicious commands [33,34]. Researchers have also tried to mitigate information leakage problems by optimizing the design variables and machine

process [97]. In the network layer, attacks on communication protocols, such as HDMI [36] and ZLL (ZigBee light link) [37], are also proposed. For the application layer, researchers have presented attack models for certain industrial control systems, such as robotic vehicle (RV) systems [25] and smart grids [12,14,15]. Moreover, audit frameworks are designed to implement systematic safety tests in embedded devices [59,145].

4.3.3. Defense

Most devices in CPS only have limited storage, battery, and transmission ranges. Therefore, it is a challenging mission to build lightweight warning systems for abnormal data or intrusion behaviors. In different structure layers, researchers have utilized different approaches to detect and block attacks. They used the system's operating characteristics in the perception layer, analyzed traffic in the network layer, and formulated defense policies in the application layer.

In the perception layer, researchers have created methods to detect abnormal status with operation features, such as cycle per instruction (CPI) [95], radio frequency (RF) [10], sensor–device–user interactions [31], and long-term expected values [87]. In the network layer, researchers modeled regular traffic first and then leveraged different methods to identify abnormal sequences, such as model checking [91], packet observation against long-term expected values [87], and artificial neural networks (ANNs) [89]. In the application layer, game theory [72,85] can be used to judge the reliability of distributed CPS, generate invariant rules from ICS operational logs [76], and check abnormal behaviors by transforming behavior rules to a state machine [17].

4.3.4. Control

With the development of computation and communication capabilities, CPS has become a complex, extensive system. The control of system security is no longer based on traditional performance detection but is combined with various control theories and technologies. Studies about control have mainly been performed at the application layer and include device pairing, cloud service management, access control, and trusted computing. The other works mainly address key management [19,82] and encrypted communication schemes [83,106].

Device pairing is a significant control task that is responsible for coordination between multiple devices. Researchers have designed pairing protocols for devices with different sensor types [114] and augmented reality (AR) headsets [115]. They also tried to develop a noisy vibration scheme against eavesdropping attacks [96]. Moreover, cloud computing scenes attract attention in security research, and the main studies are about autonomous collaborative networks [24], decentralized trigger–action platforms [63], and secure message distribution in clouds [62]. Access control and trusted computing are analyzed above, so we do not address them here.

4.4. Non-Tech Analysis

In this subsection, we provide a broad view of the global CPS security research situation from the non-tech aspect. We perform statistics on publication time, organization/country of authors, and funding supporters of the papers. Finally, we also show the cooperation among different universities and situations worldwide.

In security conferences and journals labeled A/B, the number of papers related to CPS security has increased dramatically over the past five years. In the first three years (2015–2017), the number doubled from 10 to 23. From 2018 to 2019, the number increased more rapidly: 2018 (32), 2019 (62). Figure 2 shows this trend.

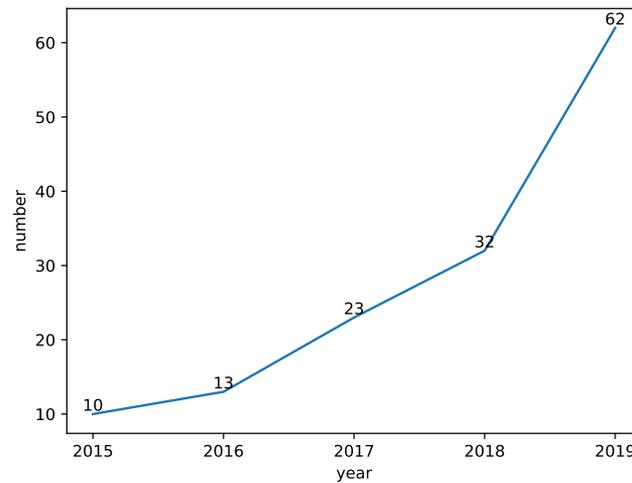


Figure 2. Time trend of the number of CPS security-related papers.

Figure 3 shows the contributions of different countries, and their proportions are shown in Figure 4. The United States made the most prominent contribution (60 papers), followed by China (20 papers). Germany, Canada, Singapore, and the United Kingdom can be ranked as the third echelon, with 15, 6, 5, and 5 papers, respectively. The UK and Singapore played a major role by publishing five papers, while the other countries published no more than three papers.

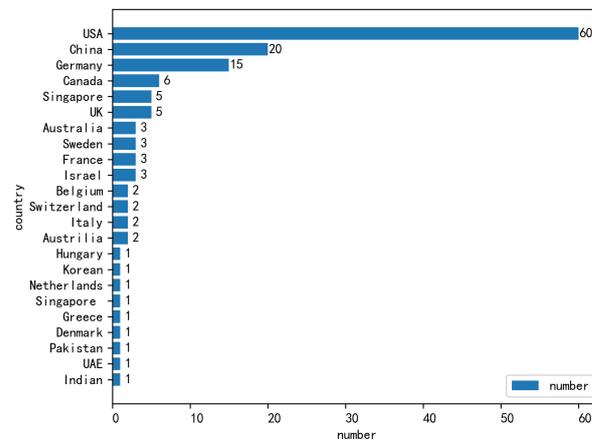


Figure 3. Number of papers from different countries.

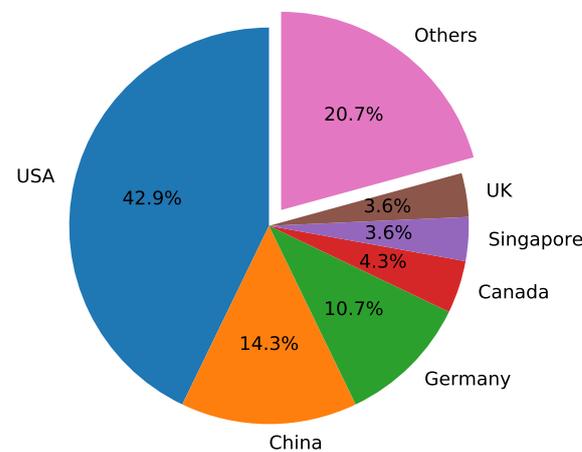


Figure 4. Proportion of papers from different countries.

We also analyze the sponsors and find that the top sponsors are the NSF, U.S. military, NSFC, DFG, etc. The NSF (Nation Science Foundation) is a federal agency in the United States that funds basic scientific research programs. The U.S. military, mainly including U.S. Army Research Office, the U.S. Office of Naval Research, Air Force Research Lab, DARPA, and Department of Defense, has also participated in many scientific research projects. The NSFC (National Natural Science Foundation of China) and the DFG (German Research Foundation) also sponsor many scientific research projects, whose proportions are 10% and 5%, respectively. Figure 5 shows the distributed situation with a pie chart, and Table 4 shows an explanation of the abbreviations.

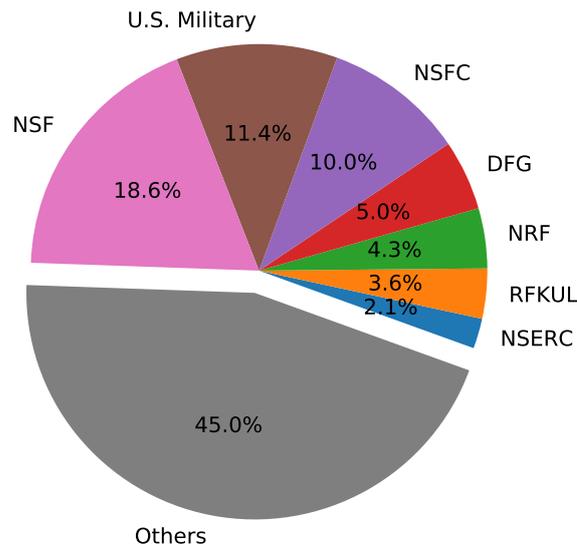


Figure 5. Main sponsors and their attributes.

Table 4. Explanation of the abbreviations.

Abbr	Expansion	Country
NSF	National Science Foundation	U.S.
USA Military	Army, Navy, Air Force, DARPA, DoD	U.S.
NSFC	National Natural Science Foundation of China	China
DFG	German Research Foundation	Germany
NRF	National Research Foundation	Singapore
RFKUL	Research Fund KU Leuven	Germany
NSERC	Natural Sciences and Engineering Research Council of Canada	Canada

The prevalence and vulnerabilities of CPS have drawn the attention of researchers. Many universities and institutes have created labs to perform CPS security-related studies in recent years. Figure 6 shows the leading university laboratories and institutes and their relations in the research field of CPS security.

The USA’s universities and research labs are leading the studies. Carnegie Mellon University developed the SmartAuth framework [38] in cooperation with the University of Chicago and Indiana University at Bloomington. The University of California cooperates with the Air Force Research Lab and utilizes high-level numeric feature vectors to find bugs in firmware images [141]. Cornell Tech, Columbia University, and Tel Aviv University from Israel proposed a situational access control framework for the IoT system [43].

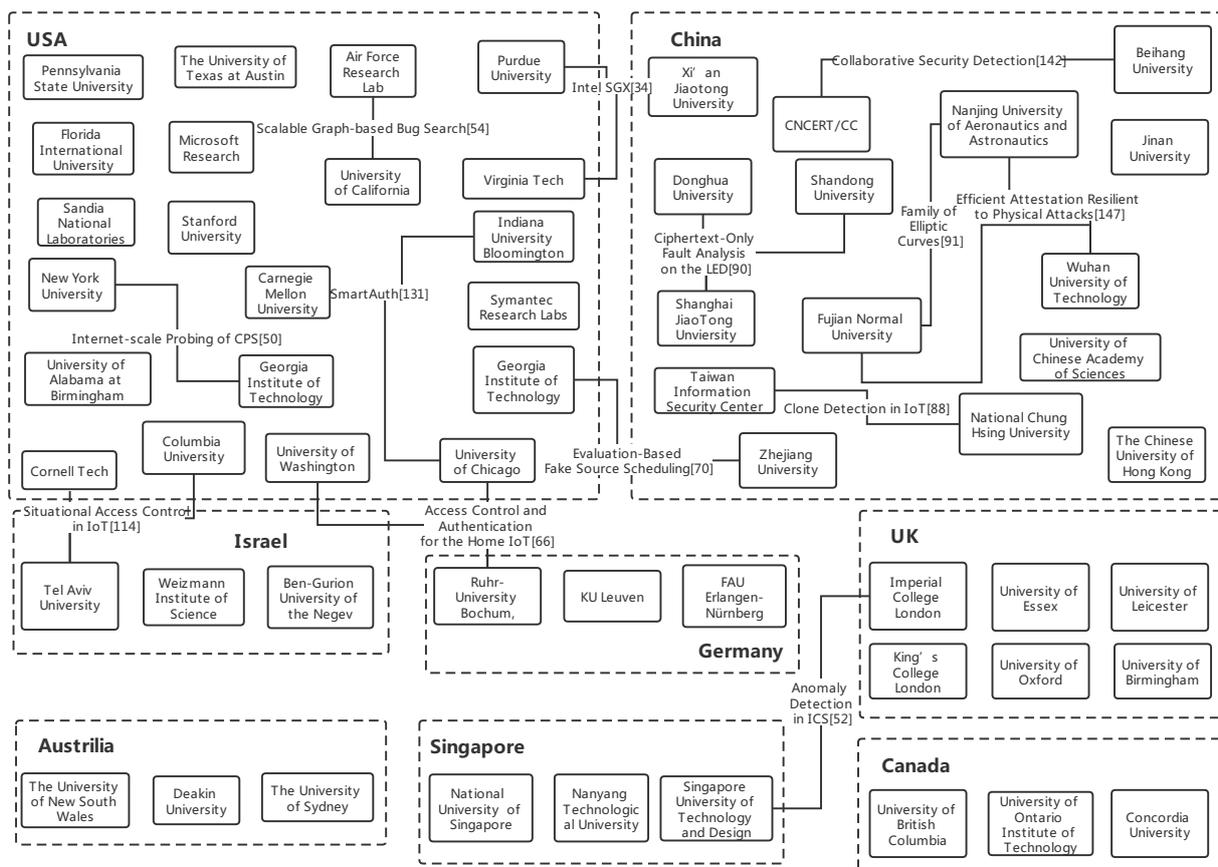


Figure 6. Main universities and labs studying the security of CPS and their cooperations.

China has also made significant contributions to CPS security research. Zhejiang University in cooperation with the Georgia Institute of Technology proposed a fake message scheduling to protect source location privacy [99]. Xi'an Jiaotong University, Purdue University and Virginia Tech improved the Intel[®] SGX with a “heartbeat” protocol to assist key revocation [16]. Donghua University, Shandong University, and Shanghai JiaoTong University performed a ciphertext-only fault analysis on the LED lightweight cryptosystem [102]. Moreover, the Taiwan Information Security Center and National Chung Hsing University introduced a clone detection method from multidimensional scaling [86].

Universities and laboratories from other countries, such as Israel, Singapore, the United Kingdom, Australia, Germany, and Canada, are also engaged in related research. All the research cooperation among countries is close. For example, Singapore University of Technology and Design developed an abnormal detection framework to detect malicious behavior with the Imperial College of London [76].

5. Conclusions

CPS security is a hot area of current security research, as it is closely related to real life. Although there are many studies on CPS security, more efforts are still needed to understand CPS security from a more comprehensive perspective. We provide a comprehensive survey based on 142 papers selected from A/B level conferences/journals recommended by the CCF and obtain a series of valuable conclusions. Technically, the current research hotspots of CPS security include authentication and access control schemes and analysis of firmware and apps. However, there are still many emerging fields, such as the security of CPS blockchains and cloud servers that require more research. Globally, the United States is in a leading position with respect to the most published papers, while China is catching up. In addition, the number of institutions/universities that contribute to CPS

security research has increased from 10 in 2015 to 62 in 2019, and international academic communication is getting closer than ever. Therefore, we hope to provide more insight and ideas to the research community through this review.

Author Contributions: Conceptualization, Z.W. and W.X.; methodology, W.X.; software, Z.W.; validation, E.W., B.W. and J.T.; funding acquisition, B.W. All authors have read and agreed to the published version of the manuscript.

Funding: The work was supported by the Natural Science Foundation of Hunan Province in China (2019JJ50729) and Natural Science Foundation of China (61902412, 61902416).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CPS Cyber Physical System

References

1. Arias, O.; Wurm, J.; Hoang, K.; Jin, Y. Privacy and security in internet of things and wearable devices. *IEEE Trans. Multi-Scale Comput. Syst.* **2015**, *1*, 99–109. [[CrossRef](#)]
2. Alrawi, O.; Lever, C.; Antonakakis, M.; Monroe, F. Sok: Security evaluation of home-based iot deployments. In Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 19–23 May 2019; pp. 1362–1380.
3. Stojkoska, B.L.R.; Trivodaliev, K.V. A review of Internet of Things for smart home: Challenges and solutions. *J. Clean. Prod.* **2017**, *140*, 1454–1464. [[CrossRef](#)]
4. Zarpelão, B.B.; Miani, R.S.; Kawakani, C.T.; de Alvarenga, S.C. A survey of intrusion detection in Internet of Things. *J. Netw. Comput. Appl.* **2017**, *84*, 25–37. [[CrossRef](#)]
5. Da Xu, L.; He, W.; Li, S. Internet of things in industries: A survey. *IEEE Trans. Ind. Inform.* **2014**, *10*, 2233–2243.
6. Yang, T.; Zhang, G.; Liu, L.; Yang, Y.; Zhao, S.; Sun, H.; Wang, W. New Features of Authentication Scheme for the IoT: A Survey. In Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things, London, UK, 11–15 November 2019; pp. 44–49.
7. Ashibani, Y.; Mahmoud, Q.H. Cyber physical systems security: Analysis, challenges and solutions. *Comput. Secur.* **2017**, *68*, 81–97. [[CrossRef](#)]
8. Madakam, S.; Lake, V.; Lake, V.; Lake, V. Internet of Things (IoT): A literature review. *J. Comput. Commun.* **2015**, *3*, 164. [[CrossRef](#)]
9. Ahmed, H.I.; Nasr, A.A.; Abdel-Mageid, S.; Aslan, H.K. A survey of IoT security threats and defenses. *Int. J. Adv. Comput. Res.* **2019**, *9*, 325–350. [[CrossRef](#)]
10. Shekari, T.; Bayens, C.; Cohen, M.; Graber, L.; Beyah, R. RFDIDS: Radio Frequency-based Distributed Intrusion Detection System for the Power Grid. In Proceedings of the Network and Distributed Systems Security (NDSS) Symposium 2019, San Diego, CA, USA, 24–27 February 2019.
11. Formby, D.; Srinivasan, P.; Leonard, A.; Rogers, J.; Beyah, R.A. Who's in Control of Your Control System? Device Fingerprinting for Cyber-Physical Systems. In Proceedings of the NDSS, San Diego, CA, USA, 21–24 February 2016.
12. Dabrowski, A.; Ullrich, J.; Weippl, E.R. Grid shock: Coordinated load-changing attacks on power grids: The non-smart power grid is vulnerable to cyber attacks as well. In Proceedings of the 33rd Annual Computer Security Applications Conference, San Juan, PR, USA, 4–8 December 2017; pp. 303–314.
13. Garcia, L.; Brassier, F.; Cintuglu, M.H.; Sadeghi, A.R.; Mohammed, O.A.; Zonouz, S.A. Hey, My Malware Knows Physics! Attacking PLCs with Physical Model Aware Rootkit. In Proceedings of the NDSS, San Diego, CA, USA, 26 February–1 March 2017.
14. Soltan, S.; Mittal, P.; Poor, H.V. BlackIoT: IoT botnet of high wattage devices can disrupt the power grid. In Proceedings of the 27th USENIX Security Symposium (USENIX Security 18), Baltimore, MD, USA, 15–17 August 2018; pp. 15–32.
15. Huang, B.; Cardenas, A.A.; Baldick, R. Not everything is dark and gloomy: Power grid protections against IoT demand attacks. In Proceedings of the 28th USENIX Security Symposium (USENIX Security 19), Santa Clara, CA, USA, 14–16 August 2019; pp. 1115–1132.
16. Chen, Y.; Sun, W.; Zhang, N.; Zheng, Q.; Lou, W.; Hou, Y.T. Towards efficient fine-grained access control and trustworthy data processing for remote monitoring services in iot. *IEEE Trans. Inf. Forensics Secur.* **2018**, *14*, 1830–1842. [[CrossRef](#)]
17. Mitchell, R.; Chen, R. Behavior rule specification-based intrusion detection for safety critical medical cyber physical systems. *IEEE Trans. Dependable Secur. Comput.* **2015**, *12*, 16–30. [[CrossRef](#)]
18. Cheng, L.; Tian, K.; Yao, D. Orpheus: Enforcing cyber-physical execution semantics to defend against data-oriented attacks. In Proceedings of the 33rd Annual Computer Security Applications Conference, San Juan, PR, USA, 4–8 December 2017; pp. 315–326.
19. Thomas, R.J.; Ordean, M.; Chothia, T.; De Ruiter, J. TRAKS: A Universal Key Management Scheme for ERTMS. In Proceedings of the 33rd Annual Computer Security Applications Conference, Orlando, FL, USA, 4–8 December 2017; pp. 327–338.

20. Zhang, M.; Peng, B.; Chen, Y. An Efficient Image Encryption Scheme for Industrial Internet-of-Things Devices. In Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things, London, UK, 11–15 November 2019; pp. 38–43.
21. Van Bulck, J.; Mühlberg, J.T.; Piessens, F. VulCAN: Efficient component authentication and software isolation for automotive control networks. In Proceedings of the 33rd Annual Computer Security Applications Conference, Orlando, FL, USA, 4–8 December 2017; pp. 225–237.
22. Birnbach, S.; Baker, R.; Martinovic, I. Wi-fly?: Detecting Privacy iNvasion Attacks by Consumer Drones. In Proceedings of the NDSS, San Diego, CA, USA, 26 February–1 March 2017.
23. Ibrahim, A.; Sadeghi, A.R.; Tsudik, G. Us-aid: Unattended scalable attestation of iot devices. In Proceedings of the 2018 IEEE 37th Symposium on Reliable Distributed Systems (SRDS), Salvador, Brazil, 2–5 October 2018; pp. 21–30.
24. Abera, T.; Bahmani, R.; Brassler, F.; Ibrahim, A.; Sadeghi, A.R.; Schunter, M. DIAT: Data Integrity Attestation for Resilient Collaboration of Autonomous Systems. In Proceedings of the NDSS, San Diego, CA, USA, 24–27 February 2019.
25. Dash, P.; Karimibiuki, M.; Pattabiraman, K. Out of control: Stealthy attacks against robotic vehicles protected by control-based techniques. In Proceedings of the 35th Annual Computer Security Applications Conference, San Juan, PR, USA, 9–13 December 2019; pp. 660–672.
26. Sun, P.; Garcia, L.; Zonouz, S. Tell Me More Than Just Assembly! Reversing Cyber-Physical Execution Semantics of Embedded IoT Controller Software Binaries. In Proceedings of the 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Portland, OR, USA, 24–27 June 2019; pp. 349–361.
27. Hasan, M.; Mohan, S. Protecting Actuators in Safety-Critical IoT Systems from Control Spoofing Attacks. In Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things, London, UK, 11–15 November 2019; pp. 8–14.
28. Choi, J.; Jeoung, H.; Kim, J.; Ko, Y.; Jung, W.; Kim, H.; Kim, J. Detecting and identifying faulty IoT devices in smart home with context extraction. In Proceedings of the 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Luxembourg City, Luxembourg, 25–28 June 2018; pp. 610–621.
29. Birnbach, S.; Eberz, S.; Martinovic, I. Peeves: Physical Event Verification in Smart Homes. In Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, ACM, London, UK, 11–15 November 2019; pp. 1455–1467.
30. Sikder, A.K.; Aksu, H.; Uluagac, A.S. 6thsense: A context-aware sensor-based attack detector for smart devices. In Proceedings of the 26th USENIX Security Symposium (USENIX Security 17), Santa Clara, CA, USA, 12–14 July 2017; pp. 397–414.
31. Sikder, A.K.; Babun, L.; Aksu, H.; Uluagac, A.S. Aegis: A context-aware security framework for smart home systems. In Proceedings of the 35th Annual Computer Security Applications Conference, San Juan, PR, USA, 9–13 December 2019; pp. 28–41.
32. Ronen, E.; Shamir, A. Extended functionality attacks on IoT devices: The case of smart lights. In Proceedings of the 2016 IEEE European Symposium on Security and Privacy (EuroS&P), Saarbrücken, Germany, 21–24 March 2016; pp. 3–12.
33. Carlini, N.; Mishra, P.; Vaidya, T.; Zhang, Y.; Sherr, M.; Shields, C.; Wagner, D.; Zhou, W. Hidden voice commands. In Proceedings of the 25th USENIX Security Symposium (USENIX Security 16), Denver, CO, USA, 22–24 June 2016; pp. 513–530.
34. Wang, C.; Anand, S.A.; Liu, J.; Walker, P.; Chen, Y.; Saxena, N. Defeating hidden audio channel attacks on voice assistants via audio-induced surface vibrations. In Proceedings of the 35th Annual Computer Security Applications Conference, San Juan, PR, USA, 9–13 December 2019; pp. 42–56.
35. Wu, D.J.; Taly, A.; Shankar, A.; Boneh, D. Privacy, discovery, and authentication for the internet of things. In *Lecture Notes in Computer Science, Proceedings of the European Symposium on Research in Computer Security, Heraklion, Greece, 26–30 September 2016*; Springer: Cham, Switzerland, 2016; pp. 301–319.
36. Rondon, L.P.; Babun, L.; Akkaya, K.; Uluagac, A.S. HDMI-walk: Attacking HDMI distribution networks via consumer electronic control protocol. In Proceedings of the 35th Annual Computer Security Applications Conference, San Juan, PR, USA, 9–13 December 2019; pp. 650–659.
37. Ronen, E.; Shamir, A.; Weingarten, A.O.; O’Flynn, C. IoT goes nuclear: Creating a ZigBee chain reaction. In Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2017; pp. 195–212.
38. Tian, Y.; Zhang, N.; Lin, Y.H.; Wang, X.; Ur, B.; Guo, X.; Tague, P. Smartauth: User-centered authorization for the internet of things. In Proceedings of the 26th USENIX Security Symposium (USENIX Security 17), Santa Clara, CA, USA, 12–14 July 2017; pp. 361–378.
39. Agadakos, I.; Hallgren, P.; Damopoulos, D.; Sabelfeld, A.; Portokalidis, G. Location-enhanced authentication using the IoT: Because you cannot be in two places at once. In Proceedings of the 32nd Annual Conference on Computer Security Applications, Los Angeles, CA, USA, 5–9 December 2016; pp. 251–264.
40. Xu, Y.; Price, T.; Frahm, J.M.; Monroe, F. Virtual U: Defeating face liveness detection by building virtual models from your public photos. In Proceedings of the 25th USENIX Security Symposium (USENIX Security 16), Denver, CO, USA, 22–24 June 2016; pp. 497–512.
41. Tian, D.J.; Bates, A.; Butler, K. Defending against malicious USB firmware with GoodUSB. In Proceedings of the 31st Annual Computer Security Applications Conference, Los Angeles, CA, USA, 7–11 December 2015; pp. 261–270.
42. Benadjila, R.; Michelizza, A.; Renard, M.; Thierry, P.; Trebuchet, P. WooKey: Designing a trusted and efficient USB device. In Proceedings of the 35th Annual Computer Security Applications Conference, San Juan, PR, USA, 9–13 December 2019; pp. 673–686.

43. Schuster, R.; Shmatikov, V.; Tromer, E. Situational access control in the internet of things. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 1056–1073.
44. Fernandes, E.; Paupore, J.; Rahmati, A.; Simionato, D.; Conti, M.; Prakash, A. Flowfence: Practical data protection for emerging iot application frameworks. In Proceedings of the 25th USENIX Security Symposium (USENIX Security 16), Denver, CO, USA, 22–24 June 2016; pp. 531–548.
45. Jia, Y.J.; Chen, Q.A.; Wang, S.; Rahmati, A.; Fernandes, E.; Mao, Z.M.; Prakash, A.; Unviarsity, S. ContextIoT: Towards providing contextual integrity to appified IoT platforms. In Proceedings of the NDSS, San Diego, CA, USA, 26 February–1 March 2017.
46. Wang, Q.; Hassan, W.U.; Bates, A.; Gunter, C. Fear and Logging in the Internet of Things. In Proceedings of the Network and Distributed Systems Symposium, San Diego, CA, USA, 18–21 February 2018.
47. Petracca, G.; Reineh, A.A.; Sun, Y.; Grossklags, J.; Jaeger, T. Aware: Preventing abuse of privacy-sensitive sensors via operation bindings. In Proceedings of the 26th USENIX Security Symposium (USENIX Security 17), Santa Clara, CA, USA, 12–14 July 2017; pp. 379–396.
48. He, W.; Golla, M.; Padhi, R.; Ofek, J.; Dürmuth, M.; Fernandes, E.; Ur, B. Rethinking access control and authentication for the home internet of things (IoT). In Proceedings of the 27th USENIX Security Symposium (USENIX Security 18), Baltimore, MD, USA, 15–17 August 2018; pp. 255–272.
49. Chen, J.; Zhu, Q. Interdependent strategic security risk management with bounded rationality in the internet of things. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 2958–2971. [[CrossRef](#)]
50. Chen, J.; Diao, W.; Zhao, Q.; Zuo, C.; Lin, Z.; Wang, X.; Lau, W.C.; Sun, M.; Yang, R.; Zhang, K. IoTfuzzer: Discovering Memory Corruptions in IoT Through App-based Fuzzing. In Proceedings of the NDSS, San Diego, CA, USA, 18–21 February 2018.
51. Zhang, W.; Meng, Y.; Liu, Y.; Zhang, X.; Zhang, Y.; Zhu, H. Homonit: Monitoring smart home apps from encrypted traffic. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 1074–1088.
52. Celik, Z.B.; Babun, L.; Sikder, A.K.; Aksu, H.; Tan, G.; McDaniel, P.; Uluagac, A.S. Sensitive information tracking in commodity IoT. In Proceedings of the 27th USENIX Security Symposium (USENIX Security 18), Baltimore, MD, USA, 15–17 August 2018; pp. 1687–1704.
53. Ding, W.; Hu, H. On the safety of IoT device physical interaction control. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 832–846.
54. Celik, Z.B.; Tan, G.; McDaniel, P.D. IoTGuard: Dynamic Enforcement of Security and Safety Policy in Commodity IoT. In Proceedings of the NDSS, San Diego, CA, USA, 24–27 February 2019.
55. Balliu, M.; Merro, M.; Pasqua, M. Securing Cross-App Interactions in IoT Platforms. In Proceedings of the 2019 IEEE 32nd Computer Security Foundations Symposium (CSF), Hoboken, NJ, USA, 25–28 June 2019; pp. 319–31915.
56. Wang, X.; Sun, Y.; Nanda, S.; Wang, X. Looking from the mirror: Evaluating IoT device security through mobile companion apps. In Proceedings of the 28th USENIX Security Symposium (USENIX Security 19), Santa Clara, CA, USA, 14–16 August 2019; pp. 1151–1167.
57. Kumar, D.; Shen, K.; Case, B.; Garg, D.; Alperovich, G.; Kuznetsov, D.; Gupta, R.; Durumeric, Z. All things considered: An analysis of IoT devices on home networks. In Proceedings of the 28th USENIX Security Symposium (USENIX Security 19), Santa Clara, CA, USA, 14–16 August 2019; pp. 1169–1185.
58. Zhou, W.; Jia, Y.; Yao, Y.; Zhu, L.; Guan, L.; Mao, Y.; Liu, P.; Zhang, Y. Discovering and understanding the security hazards in the interactions between IoT devices, mobile apps, and clouds on smart home platforms. In Proceedings of the 28th USENIX Security Symposium (USENIX Security 19), Santa Clara, CA, USA, 14–16 August 2019; pp. 1133–1150.
59. Waraga, O.A.; Bettayeb, M.; Nasir, Q.; Talib, M.A. Design and implementation of automated IoT security testbed. *Comput. Secur.* **2020**, *88*, 101648. [[CrossRef](#)]
60. Truong, N.B.; Lee, G.M.; Um, T.W.; Mackay, M. Trust evaluation mechanism for user recruitment in mobile crowd-sensing in the Internet of Things. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 2705–2719. [[CrossRef](#)]
61. Chen, R.; Bao, F.; Guo, J. Trust-based service management for social internet of things systems. *IEEE Trans. Dependable Secur. Comput.* **2015**, *13*, 684–696. [[CrossRef](#)]
62. Yang, L.; Humayed, A.; Li, F. A multi-cloud based privacy-preserving data publishing scheme for the internet of things. In Proceedings of the 32nd Annual Conference on Computer Security Applications, Los Angeles, CA, USA, 5–9 December 2016; pp. 30–39.
63. Fernandes, E.; Rahmati, A.; Jung, J.; Prakash, A. Decentralized action integrity for trigger-action IoT platforms. In Proceedings of the 2018 Network and Distributed System Security Symposium, San Diego, CA, USA, 18–21 February 2018.
64. Schulz, S.; Schaller, A.; Kohnhäuser, F.; Katzenbeisser, S. Boot attestation: Secure remote reporting with off-the-shelf iot sensors. In *Lecture Notes in Computer Science, Proceedings of the European Symposium on Research in Computer Security, Oslo, Norway, 11–15 September 2017*; Springer: Cham, Switzerland, 2017; pp. 437–455.
65. Bastys, I.; Balliu, M.; Sabelfeld, A. If this then what? Controlling flows in IoT apps. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 1102–1119.
66. Fotiou, N.; Pittaras, I.; Siris, V.A.; Polyzos, G.C. Enabling Opportunistic Users in Multi-Tenant IoT Systems using Decentralized Identifiers and Permissioned Blockchains. In Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things, London, UK, 11–15 November 2019; pp. 22–23.

67. Pereira, H.G.; Fong, P.W. SEPD: An Access Control Model for Resource Sharing in an IoT Environment. In *Proceedings of the European Symposium on Research in Computer Security*; Springer: Cham, Switzerland, 2019; pp. 195–216.
68. Chen, J.; Zuo, C.; Diao, W.; Dong, S.; Zhao, Q.; Sun, M.; Lin, Z.; Zhang, Y.; Zhang, K. Your IoTs Are (Not) Mine: On the Remote Binding Between IoT Devices and Users. In *Proceedings of the 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, Portland, OR, USA, 24–27 June 2019; pp. 222–233.
69. Garamvölgyi, P.; Kocsis, I.; Gehl, B.; Klenik, A. Towards Model-Driven Engineering of Smart Contracts for Cyber-Physical Systems. In *Proceedings of the 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, Luxembourg City, Luxembourg, 25–28 June 2018; pp. 134–139.
70. Shafagh, H.; Burkhalter, L.; Hithnawi, A.; Duquennoy, S. Towards blockchain-based auditable storage and sharing of iot data. In *Proceedings of the 2017 on Cloud Computing Security Workshop*, Dallas, TX, USA, 11–15 November 2017; pp. 45–50.
71. Wang, Q.; Datta, P.; Yang, W.; Liu, S.; Bates, A.; Gunter, C.A. Charting the Attack Surface of Trigger-Action IoT Platforms. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, London, UK, 11–15 November 2019; pp. 1439–1453.
72. Pawlick, J.; Zhu, Q. Strategic trust in cloud-enabled cyber-physical systems with an application to glucose control. *IEEE Trans. Inf. Forensics Secur.* **2017**, *12*, 2906–2919. [[CrossRef](#)]
73. Leiba, O.; Yitzchak, Y.; Bitton, R.; Nadler, A.; Shabtai, A. Incentivized delivery network of IoT software updates based on trustless proof-of-distribution. In *Proceedings of the 2018 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, London, UK, 24–26 April 2018; pp. 29–39.
74. Feng, J.; Zhao, X.; Lu, G.; Zhao, F. PoTN: A Novel Blockchain Consensus Protocol with Proof-of-Trust Negotiation in Distributed IoT Networks. In *Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things*, London, UK, 11–15 November 2019; pp. 32–37.
75. Lanotte, R.; Merro, M.; Muradore, R.; Viganò, L. A formal approach to cyber-physical attacks. In *Proceedings of the 2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, Santa Barbara, CA, USA, 21–25 August 2017; pp. 436–450.
76. Feng, C.; Palleti, V.R.; Mathur, A.; Chana, D. A Systematic Framework to Generate Invariants for Anomaly Detection in Industrial Control Systems. In *Proceedings of the NDSS*, San Diego, CA, USA, 24–27 February 2019.
77. Krishnamurthy, P.; Khorrami, F.; Karri, R.; Paul-Pena, D.; Salehghaffari, H. Process-aware covert channels using physical instrumentation in cyber-physical systems. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2761–2771. [[CrossRef](#)]
78. Herzberg, A.; Kfir, Y. The chatty-sensor: A provably-covert channel in cyber physical systems. In *Proceedings of the 35th Annual Computer Security Applications Conference*, San Juan, PR, USA, 9–13 December 2019; pp. 638–649.
79. Yu, B.; Wang, P.; Yue, T.; Tang, Y. Poster: Fuzzing IoT Firmware via Multi-stage Message Generation. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, London, UK, 11–15 November 2019; pp. 2525–2527.
80. Mikulskis, J.; Becker, J.K.; Gvozdenovic, S.; Starobinski, D. Snout: An Extensible IoT Pen-Testing Tool. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, London, UK, 11–15 November 2019; pp. 2529–2531.
81. Kim, J.Y.; Holz, R.; Hu, W.; Jha, S. Automated analysis of secure internet of things protocols. In *Proceedings of the 33rd Annual Computer Security Applications Conference*, Orlando, FL, USA, 4–8 December 2017; pp. 238–249.
82. Krentz, K.F.; Meinel, C. Handling reboots and mobility in 802.15. 4 security. In *Proceedings of the 31st Annual Computer Security Applications Conference*, Los Angeles, CA, USA, 7–11 December 2015; pp. 121–130.
83. Kim, J.Y.; Hu, W.; Shafagh, H.; Jha, S. Seda: Secure over-the-air code dissemination protocol for the internet of things. *IEEE Trans. Dependable Secur. Comput.* **2016**, *15*, 1041–1054. [[CrossRef](#)]
84. De Donno, M.; Felipe, J.M.D.; Dragoni, N. ANTIBIOTIC 2.0: A fog-based anti-malware for internet of things. In *Proceedings of the 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, Stockholm, Sweden, 17–19 June 2019; pp. 11–20.
85. Wu, H.; Wang, W. A game theory based collaborative security detection method for Internet of Things systems. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 1432–1445. [[CrossRef](#)]
86. Lee, P.Y.; Yu, C.M.; Dargahi, T.; Conti, M.; Bianchi, G. MDSClone: Multidimensional scaling aided clone detection in Internet of Things. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2031–2046. [[CrossRef](#)]
87. Abhishek, N.V.; Tandon, A.; Lim, T.J.; Sikdar, B. A GLRT-Based Mechanism for Detecting Relay Misbehavior in Clustered IoT Networks. *IEEE Trans. Inf. Forensics Secur.* **2019**, *15*, 435–446. [[CrossRef](#)]
88. Chen, Y.; Poskitt, C.M.; Sun, J. Learning from mutants: Using code mutation to learn and monitor invariants of a cyber-physical system. In *Proceedings of the 2018 IEEE Symposium on Security and Privacy (SP)*, San Francisco, CA, USA, 21–23 May 2018; pp. 648–660.
89. Yoon, J. Using a Deep-Learning Approach for Smart IoT Network Packet Analysis. In *Proceedings of the 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, Stockholm, Sweden, 17–19 June 2019; pp. 291–299.
90. Stylianopoulos, C.; Kindström, S.; Almgren, M.; Landsiedel, O.; Papatriantafyllou, M. Co-evaluation of pattern matching algorithms on IoT devices with embedded GPUs. In *Proceedings of the 35th Annual Computer Security Applications Conference*, San Juan, PR, USA, 9–13 December 2019; pp. 17–27.
91. Tabrizi, M. Formal security analysis of smart embedded systems. In *Proceedings of the 32nd Annual Conference on Computer Security Applications*, Los Angeles, CA, USA, 5–9 December 2016; pp. 1–15.

92. Antonioli, D. MiniCPS: A toolkit for security research on CPS networks. In Proceedings of the First, ACM Workshop on Cyber-Physical Systems-Security and/or Privacy, Denver, CO, USA, 16 October 2015; pp. 91–100.
93. Keliris, A. Icsref: A framework for automated reverse engineering of industrial control systems binaries. In Proceedings of the NDSS, San Diego, CA, USA, 18–21 February 2018.
94. Corteggiani, N.; Camurati, G.; Francillon, A. Inception: System-wide security testing of real-world embedded systems software. In Proceedings of the 27th USENIX Security Symposium (USENIX Security 18), Baltimore, MD, USA, 15–17 August 2018; pp. 309–326.
95. Zhai, X.; Appiah, K.; Ehsan, S.; Howells, G.; Hu, H.; Gu, D.; McDonald-Maier, K.D. A method for detecting abnormal program behavior on embedded devices. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 1692–1704. [[CrossRef](#)]
96. Anand, S.A.; Saxena, N. Noisy Vibrational Pairing of IoT Devices. *IEEE Trans. Dependable Secur. Comput.* **2018**, *16*, 530–545. [[CrossRef](#)]
97. Chhetri, S.R.; Faezi, S.; Al Faruque, M.A. Information leakage-aware computer-aided cyber-physical manufacturing. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2333–2344. [[CrossRef](#)]
98. Sun, M.; Tay, W.P. On the Relationship Between Inference and Data Privacy in Decentralized IoT Networks. *IEEE Trans. Inf. Forensics Secur.* **2019**, *15*, 852–866. [[CrossRef](#)]
99. Hong, Z.; Wang, R.; Ji, S.; Beyah, R. Attacker location evaluation-based fake source scheduling for source location privacy in cyber-physical systems. *IEEE Trans. Inf. Forensics Secur.* **2018**, *14*, 1337–1350. [[CrossRef](#)]
100. Zhang, L.; Chen, J.; Diao, W.; Guo, S.; Weng, J.; Zhang, K. CryptoREX: Large-scale Analysis of Cryptographic Misuse in IoT Devices. In Proceedings of the 22nd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2019), Beijing, China, 23–25 September 2019; pp. 151–164.
101. Shi, Y.; Wei, W.; He, Z.; Fan, H. An ultra-lightweight white-box encryption scheme for securing resource-constrained IoT devices. In Proceedings of the 32nd Annual Conference on Computer Security Applications, Los Angeles, CA, USA, 5–9 December 2016; pp. 16–29.
102. Li, W.; Liao, L.; Gu, D.; Li, C.; Ge, C.; Guo, Z.; Liu, Y.; Liu, Z. Ciphertext-only fault analysis on the LED lightweight cryptosystem in the internet of things. *IEEE Trans. Dependable Secur. Comput.* **2018**, *16*, 454–461. [[CrossRef](#)]
103. Liu, Z.; Huang, X.; Hu, Z.; Khan, M.K.; Seo, H.; Zhou, L. On emerging family of elliptic curves to secure internet of things: ECC comes of age. *IEEE Trans. Dependable Secur. Comput.* **2016**, *14*, 237–248. [[CrossRef](#)]
104. Liu, Z.; Seo, H. IoT-NUMS: Evaluating NUMS elliptic curve cryptography for IoT platforms. *IEEE Trans. Inf. Forensics Secur.* **2018**, *14*, 720–729. [[CrossRef](#)]
105. Mangia, M.; Pareschi, F.; Rovatti, R.; Setti, G. Low-cost security of IoT sensor nodes with rakesness-based compressed sensing: Statistical and known-plaintext attacks. *IEEE Trans. Inf. Forensics Secur.* **2017**, *13*, 327–340. [[CrossRef](#)]
106. Azar, K.Z.; Farahmand, F.; Kamali, H.M.; Roshanifard, S.; Homayoun, H.; Diehl, W.; Gaj, K.; Sasan, A. COMA: Communication and Obfuscation Management Architecture. In Proceedings of the 22nd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2019), Beijing, China, 23–25 September 2019; pp. 181–195.
107. Noorman, J.; Bulck, J.V.; Mühlberg, J.T.; Piessens, F.; Maene, P.; Preneel, B.; Verbauwhede, I.; Götzfried, J.; Müller, T.; Freiling, F. Sancus 2.0: A low-cost security architecture for IoT devices. *ACM Trans. Priv. Secur. (TOPS)* **2017**, *20*, 1–33. [[CrossRef](#)]
108. Celosia, G.; Cunche, M. Fingerprinting Bluetooth-Low-Energy Devices Based on the Generic Attribute Profile. In Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things, London, UK, 11–15 November 2019; pp. 24–31.
109. Zuo, C.; Wen, H.; Lin, Z.; Zhang, Y. Automatic Fingerprinting of Vulnerable BLE IoT Devices with Static UUIDs from Mobile Apps. In Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, London, UK, 11–15 November 2019; pp. 1469–1483.
110. Bezawada, B.; Bachani, M.; Peterson, J.; Shirazi, H.; Ray, I.; Ray, I. Behavioral fingerprinting of iot devices. In Proceedings of the 2018 Workshop on Attacks and Solutions in Hardware Security, Toronto, ON, Canada, 15–19 October 2018; pp. 41–50.
111. English, K.V.; Obaidat, I.; Sridhar, M. Exploiting Memory Corruption Vulnerabilities in Connman for IoT Devices. In Proceedings of the 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Portland, OR, USA, 24–27 June 2019; pp. 247–255.
112. Cojocar, L.; Zaddach, J.; Verdult, R.; Bos, H.; Francillon, A.; Balzarotti, D. PIE: Parser identification in embedded systems. In Proceedings of the 31st Annual Computer Security Applications Conference, Los Angeles, CA, USA, 7–11 December 2015; pp. 251–260.
113. Migault, D.; Guggemos, T.; Killian, S.; Laurent, M.; Pujolle, G.; Wary, J.P. Diet-ESP: IP layer security for IoT. *J. Comput. Secur.* **2017**, *25*, 173–203. [[CrossRef](#)]
114. Han, J.; Chung, A.J.; Sinha, M.K.; Harishankar, M.; Pan, S.; Noh, H.Y.; Zhang, P.; Tague, P. Do you feel what I hear? Enabling autonomous IoT device pairing using different sensor types. In Proceedings of the 2018 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 21–23 May 2018; pp. 836–852.
115. Sluganovic, I.; Serbec, M.; Derek, A.; Martinovic, I. HoloPair: Securing shared augmented reality using microsoft hololens. In Proceedings of the 33rd Annual Computer Security Applications Conference, Orlando, FL, USA, 4–8 December 2017; pp. 250–261.

116. Antonakakis, M.; April, T.; Bailey, M.; Bernhard, M.; Bursztein, E.; Cochran, J.; Durumeric, Z.; Halderman, J.A.; Invernizzi, L.; Kallitsis, M.; et al. Understanding the mirai botnet. In Proceedings of the 26th USENIX Security Symposium (USENIX Security 17), Santa Clara, CA, USA, 12–14 July 2017; pp. 1093–1110.
117. Herwig, S.; Harvey, K.; Hughey, G.; Roberts, R.; Levin, D. Measurement and Analysis of Hajime, a Peer-to-peer IoT Botnet. In Proceedings of the NDSS, San Diego, CA, USA, 24–27 February 2019.
118. Farooq, M.J.; Zhu, Q. Modeling, analysis, and mitigation of dynamic botnet formation in wireless iot networks. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 2412–2426. [[CrossRef](#)]
119. Torabi, S.; Bou-Harb, E.; Assi, C.; Galluscio, M.; Boukhtouta, A.; Debbabi, M. Inferring, characterizing, and investigating Internet-scale malicious IoT device activities: A network telescope perspective. In Proceedings of the 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Luxembourg City, Luxembourg, 25–28 June 2018; pp. 562–573.
120. Vervier, P.A.; Shen, Y. Before toasters rise up: A view into the emerging iot threat landscape. In *Lecture Notes in Computer Science, Proceedings of the International Symposium on Research in Attacks, Intrusions, and Defenses, Heraklion, Crete, Greece, 10–12 September 2018*; Springer: Cham, Switzerland, 2018; pp. 556–576.
121. Fachkha, C.; Bou-Harb, E.; Keliris, A.; Memon, N.D.; Ahamad, M. Internet-scale Probing of CPS: Inference, Characterization and Orchestration Analysis. In Proceedings of the NDSS, San Diego, CA, USA, 26 February–1 March 2017.
122. Asokan, N.; Brassler, F.; Ibrahim, A.; Sadeghi, A.R.; Schunter, M.; Tsudik, G.; Wachsmann, C. Seda: Scalable embedded device attestation. In Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, 1–3 October 2015; pp. 964–975.
123. Jin, C.; Yang, Z.; van Dijk, M.; Zhou, J. Proof of aliveness. In Proceedings of the 35th Annual Computer Security Applications Conference, San Juan, PR, USA, 9–13 December 2019; pp. 1–16.
124. Ibrahim, A. Collective Attestation: For a Stronger Security in Embedded Networks. In Proceedings of the 2018 IEEE 37th Symposium on Reliable Distributed Systems (SRDS), Salvador, Brazil, 2–5 October 2018; pp. 267–268.
125. Ghaeini, H.R.; Chan, M.; Bahmani, R.; Brassler, F.; Garcia, L.; Zhou, J.; Sadeghi, A.R.; Tippenhauer, N.O.; Zonouz, S. PAtt: Physics-based Attestation of Control Systems. In Proceedings of the 22nd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2019), Beijing, China, 23–25 September 2019; pp. 165–180.
126. Yan, W.; Fu, A.; Mu, Y.; Zhe, X.; Yu, S.; Kuang, B. EAPA: Efficient Attestation Resilient to Physical Attacks for IoT Devices. In Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things, London, UK, 11–15 November 2019; pp. 2–7.
127. Götzfried, J.; Müller, T.; De Clercq, R.; Maene, P.; Freiling, F.; Verbauwhede, I. Soteria: Offline software protection within low-cost embedded devices. In Proceedings of the 31st Annual Computer Security Applications Conference, Los Angeles, CA, USA, 7–11 December 2015; pp. 241–250.
128. Ammar, M.; Crispo, B.; Jacobs, B.; Hughes, D.; Daniels, W. S_μV—The Security MicroVisor: A Formally-Verified Software-Based Security Architecture for the Internet of Things. *IEEE Trans. Dependable Secur. Comput.* **2019**, *16*, 885–901. [[CrossRef](#)]
129. Clements, A.A.; Almkhahub, N.S.; Saab, K.S.; Srivastava, P.; Koo, J.; Bagchi, S.; Payer, M. Protecting bare-metal embedded systems with privilege overlays. In Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2017; pp. 289–303.
130. Chatterjee, U.; Govindan, V.; Sadhukhan, R.; Mukhopadhyay, D.; Chakraborty, R.S.; Mahata, D.; Prabhu, M.M. Building PUF based authentication and key exchange protocol for IoT without explicit CRPs in verifier database. *IEEE Trans. Dependable Secur. Comput.* **2018**, *16*, 424–437. [[CrossRef](#)]
131. Tan, H.; Tsudik, G.; Jha, S. MTRA: Multi-Tier randomized remote attestation in IoT networks. *Comput. Secur.* **2019**, *81*, 78–93. [[CrossRef](#)]
132. Kohnhäuser, F.; Büscher, N.; Katzenbeisser, S. A Practical Attestation Protocol for Autonomous Embedded Systems. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy (EuroS&P), Stockholm, Sweden, 17–19 June 2019; pp. 263–278.
133. Maroof, U.; Shaghghi, A.; Jha, S. PLAR: Towards a Pluggable Software Architecture for Securing IoT Devices. In Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things, London, UK, 11–15 November 2019; pp. 50–57.
134. Rullo, A.; Midi, D.; Serra, E.; Bertino, E. Pareto optimal security resource allocation for Internet of Things. *ACM Trans. Priv. Secur. (TOPS)* **2017**, *20*, 1–30. [[CrossRef](#)]
135. Abbasi, A.; Wetzels, J.; Holz, T.; Etalle, S. Challenges in designing exploit mitigations for deeply embedded systems. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy (EuroS&P), Stockholm, Sweden, 17–19 June 2019; pp. 31–46.
136. Clements, A.A.; Almkhahub, N.S.; Bagchi, S.; Payer, M. ACES: Automatic Compartments for Embedded Systems. In Proceedings of the 27th USENIX Security Symposium (USENIX Security 18), Baltimore, MD, USA, 15–17 August 2018; pp. 65–82.
137. Srivastava, P.; Peng, H.; Li, J.; Okhravi, H.; Shrobe, H.; Payer, M. FirmFuzz: Automated IoT Firmware Introspection and Analysis. In Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things, London, UK, 11–15 November 2019; pp. 15–21.

138. Zheng, Y.; Davanian, A.; Yin, H.; Song, C.; Zhu, H.; Sun, L. FIRM-AFL: High-throughput greybox fuzzing of iot firmware via augmented process emulation. In Proceedings of the 28th USENIX Security Symposium (USENIX Security 19), Santa Clara, CA, USA, 14–16 August 2019; pp. 1099–1114.
139. Muench, M.; Stijohann, J.; Kargl, F.; Francillon, A.; Balzarotti, D. What You Corrupt Is Not What You Crash: Challenges in Fuzzing Embedded Devices. In Proceedings of the NDSS, San Diego, CA, USA, 18–21 February 2018.
140. Chen, D.D.; Woo, M.; Brumley, D.; Egele, M. Towards Automated Dynamic Analysis for Linux-based Embedded Firmware. In Proceedings of the NDSS, San Diego, CA, USA, 21–24 February 2016; Volume 16, pp. 1–16.
141. Feng, Q.; Zhou, R.; Xu, C.; Cheng, Y.; Testa, B.; Yin, H. Scalable graph-based bug search for firmware images. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 24–28 October 2016; pp. 480–491.
142. Xu, X.; Liu, C.; Feng, Q.; Yin, H.; Song, L.; Song, D. Neural network-based graph embedding for cross-platform binary code similarity detection. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas, TX, USA, 30 October–3 November 2017; pp. 363–376.
143. Gustafson, E.; Muench, M.; Spensky, C.; Redini, N.; Machiry, A.; Fratantonio, Y.; Balzarotti, D.; Francillon, A.; Choe, Y.R.; Kruegel, C.; et al. Toward the Analysis of Embedded Firmware through Automated Re-hosting. In Proceedings of the 22nd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2019), Beijing, China, 23–25 September 2019; pp. 135–150.
144. Yao, Y.; Zhou, W.; Jia, Y.; Zhu, L.; Liu, P.; Zhang, Y. Identifying Privilege Separation Vulnerabilities in IoT Firmware with Symbolic Execution. In *Lecture Notes in Computer Science, Proceedings of the European Symposium on Research in Computer Security, Luxembourg, 23–27 September 2019*; Springer: Cham, Switzerland, 2019; pp. 638–657.
145. Nadir, I.; Ahmad, Z.; Mahmood, H.; Shah, G.A.; Shahzad, F.; Umair, M.; Khan, H.; Gulzar, U. An Auditing Framework for Vulnerability Analysis of IoT System. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Stockholm, Sweden, 17–19 June 2019; pp. 39–47.
146. Mühlberg, J.T.; Noorman, J.; Piessens, F. Lightweight and flexible trust assessment modules for the Internet of Things. In *Lecture Notes in Computer Science, Proceedings of the European Symposium on Research in Computer Security, Vienna, Austria, 21–25 September 2015*; Springer: Cham, Switzerland, 2015; pp. 503–520.
147. Xu, M.; Huber, M.; Sun, Z.; England, P.; Peinado, M.; Lee, S.; Marochko, A.; Mattoon, D.; Spiger, R.; Thom, S. Dominance as a New Trusted Computing Primitive for the Internet of Things. In Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 19–23 May 2019; pp. 1415–1430.
148. Xu, W.; Kapitza, R. RATCHETA: Memory-bounded Hybrid Byzantine Consensus for Cooperative Embedded Systems. In Proceedings of the 2018 IEEE 37th Symposium on Reliable Distributed Systems (SRDS), Salvador, Brazil, 2–5 October 2018; pp. 103–112.
149. Maene, P.; Götzfried, J.; Müller, T.; de Clercq, R.; Freiling, F.; Verbauwhede, I. Atlas: Application confidentiality in compromised embedded systems. *IEEE Trans. Dependable Secur. Comput.* **2018**, *16*, 415–423. [[CrossRef](#)]
150. Hedin, D.; Birgisson, A.; Bello, L.; Sabelfeld, A. JSFlow: Tracking information flow in JavaScript and its APIs. In Proceedings of the 29th Annual ACM Symposium on Applied Computing, Yeongju, Korea, 24–28 March 2014; pp. 1663–1671.
151. Eldefrawy, K.; Tsudik, G.; Francillon, A.; Perito, D. SMART: Secure and Minimal Architecture for Establishing a Dynamic Root of Trust. In Proceedings of the Network and Distributed System Security Symposium, San Diego, CA, USA, 5–8 February 2012.
152. Koeberl, P.; Schulz, S.; Sadeghi, A.R.; Varadharajan, V. TrustLite: A security architecture for tiny embedded devices. In Proceedings of the Ninth European Conference on Computer Systems, Amsterdam, The Netherlands, 14–16 April 2014; pp. 1–14.
153. Ibrahim, A.; Sadeghi, A.R.; Tsudik, G.; Zeitouni, S. DARPA: Device attestation resilient to physical attacks. In Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks, Darmstadt, Germany, 18–20 July 2016; pp. 171–182.
154. Kohnhäuser, F.; Büscher, N.; Gabmeyer, S.; Katzenbeisser, S. Scapi: A scalable attestation protocol to detect software and physical attacks. In Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks, Boston, MA, USA, 18–20 July 2017; pp. 75–86.
155. Noorman, J.; Agten, P.; Daniels, W.; Strackx, R.; Van Herrewewe, A.; Huygens, C.; Preneel, B.; Verbauwhede, I.; Piessens, F. Sancus: Low-cost trustworthy extensible networked devices with a zero-software trusted computing base. Presented at the 22nd USENIX Security Symposium (USENIX Security 13), San Jose, CA, USA, 26–28 June 2013; pp. 479–498.
156. Buchman, E. Tendermint: Byzantine Fault Tolerance in the Age of Blockchains. Ph.D. Thesis, University of Guelph, Guelph, ON, Canada, June 2016.
157. Alzahrani, N.; Bulusu, N. Towards true decentralization: A blockchain consensus protocol based on game theory and randomness. In *Lecture Notes in Computer Science, Proceedings of the International Conference on Decision and Game Theory for Security, Seattle, WA, USA, 29–31 October 2018*; Springer: Cham, Switzerland, 2018; pp. 465–485.
158. De Donno, M.; Dragoni, N.; Giarretta, A.; Mazzara, M. AntibloTic: Protecting IoT devices against DDoS attacks. In *International Conference in Software Engineering for Defence Applications*; Springer: Cham, Switzerland, 2016; pp. 59–72.
159. Stephens, N.; Grosen, J.; Salls, C.; Dutcher, A.; Wang, R. Driller: Augmenting fuzzing through selective symbolic execution. In Proceedings of the NDSS, San Diego, CA, USA, 21–24 February 2016.
160. Zaddach, J.; Bruno, L.; Francillon, A.; Balzarotti, D. AVATAR: A Framework to Support Dynamic Security Analysis of Embedded Systems' Firmwares. In Proceedings of the NDSS, San Diego, CA, USA, 23–26 February 2014; Volume 23, pp. 1–16.

161. Muench, M.; Nisi, D.; Francillon, A.; Balzarotti, D. Avatar 2: A multi-target orchestration platform. In Proceedings of the Workshop Binary Analysis Research (Colocated NDSS Symposium), San Diego, CA, USA, 18–21 February 2018; Volume 18, pp. 1–11.
162. Ali, N.; Hong, J.E. Failure detection and prevention for cyber-physical systems using ontology-based knowledge base. *Computers* **2018**, *7*, 68. [[CrossRef](#)]
163. Ali, N.; Hussain, M.; Hong, J.E. Analyzing Safety of Collaborative Cyber-Physical Systems Considering Variability. *IEEE Access* **2020**, *8*, 162701–162713. [[CrossRef](#)]
164. Ali, E. pH control using PI control algorithms with automatic tuning method. *Chem. Eng. Res. Des.* **2001**, *79*, 611–620. [[CrossRef](#)]
165. Chakraborty, T.; Jajodia, S.; Katz, J.; Picariello, A.; Sperli, G.; Subrahmanian, V. FORGE: A fake online repository generation engine for cyber deception. *IEEE Trans. Dependable Secur. Comput.* **2019**, *18*, 518–533. [[CrossRef](#)]
166. Han, Q.; Molinaro, C.; Picariello, A.; Sperli, G.; Subrahmanian, V.S.; Xiong, Y. Generating Fake Documents using Probabilistic Logic Graphs. *IEEE Trans. Dependable Secur. Comput.* **2021**. [[CrossRef](#)]
167. Yue, T.; Wang, P.; Tang, Y.; Wang, E.; Yu, B.; Lu, K.; Zhou, X. Ecofuzz: Adaptive energy-saving greybox fuzzing as a variant of the adversarial multi-armed bandit. In Proceedings of the 29th {USENIX} Security Symposium ({USENIX} Security 20), 15–17 July 2020; pp. 2307–2324. Available online: https://www.usenix.org/system/files/sec20fall_yue_prepub_0.pdf (accessed on 24 March 2021).