Knowledge-based Cyber Physical Security at Smart Home: A Review

AZHAR A. ALSUFYANI, Cardiff University, UK
OMAR RANA, Cardiff University, UK
CHARITH PERERA, Cardiff University, UK

Smart-home systems represent the future of modern building infrastructure as they integrate numerous devices and applications to improve the overall quality of life. These systems establish connectivity among smart devices, leveraging network technologies and algorithmic controls to monitor and manage physical environments. However, ensuring robust security in smart homes, along with securing smart devices, presents a formidable challenge. A substantial number of security solutions for smart homes rely on data-driven approaches (e.g., machine/deep learning) to identify and mitigate potential threats. These approaches involve training models on extensive datasets, which distinguishes them from knowledge-driven methods. In this review, we examine the role of knowledge within smart homes, focusing on understanding and reasoning regarding various events and their utility towards securing smart homes. We propose a taxonomy to characterize the categorization of decision-making approaches. By specifying the most common vulnerabilities, attacks, and threats, we can analyze and assess the countermeasures against them. We also examine how smart homes have been evaluated in the reviewed literature. Furthermore, we explore the challenges inherent in smart homes and investigate existing solutions that aim to overcome these limitations. Finally, we examine the key gaps in smart-home-security research and define future research directions for knowledge-driven schemes.

CCS Concepts: • **Human-centered computing → Ubiquitous and mobile computing**; • **Security and privacy → Human and societal aspects of security and privacy**; • **Computing methodologies → Knowledge representation and reasoning**.

Additional Key Words and Phrases: Internet of Things, Reasoning, Cyber-Physical Security, Smart Home

1 INTRODUCTION

Cyber-physical systems (CPSs) combine hardware and software for specific purposes. For example, actuators that function in the external environment and receive information from sensors are controlled by embedded computers and communication networks to be adaptable, autonomous, and efficient in smart spaces [70]. The level of embeddedness of these devices ranges from pervasive to ubiquitous computing [73].

In pervasive systems [98], the main characteristics are effectiveness in smart homes, invisibility, localized scalability, and the masking of uneven conditioning. Thus, the primary aim of pervasive computing is to seamlessly connect devices and applications. It assumes that the environment is intelligent such that it can a way that can detect any device that enters and exits from the environment and immediately provide information that users need to the device.

A ubiquitous system [73], which builds on the notion of free mobility by taking advantage of pervasive computing, can create dynamic models of multiple environments and apply its services accordingly. These systems contribute to the artificial intelligence (AI) field with high flexibility and effectiveness characteristics, and they have the ability to plan autonomously and intelligently [40].

CPSs enter various fields to perform functions such as security, safety, reliability, environmental monitoring, optimized service performance, and cost minimization. The typical applications of CPS

Authors' addresses: Azhar A. Alsufyani, Cardiff University, School of Computer Science and Informatics, Cardiff, CF24 4AX, UK, alsufyaniaa@cardiff.ac.uk; Omar Rana, Cardiff University, School of Computer Science and Informatics, Cardiff, CF24 4AX, UK, ranaof@cardiff.ac.uk; Charith Perera, Cardiff University, School of Computer Science and Informatics, Cardiff, CF24 4AX, UK, pererac@cardiff.ac.uk.

include ambient assisted living, transportation, power grids, agriculture, industrial maintenance, healthcare, robotics, pollution control, and communication technologies.

A smart home is a CPS application that plays a crucial role in human life from the perspectives of comfort, safety, management, security, and privacy. Moreover, the ultimate goal of a smart home is to improve the quality of life by developing all appliances in the house to become smart. Even though the concept of a smart home was discussed more than two decades ago, this technology has not achieved its main objectives and obstacles to its progress need to be investigated [19].

Between 2025 and 2030, the number of devices connected to the Internet of Things (IoT) will grow with economic value in various areas from $6.3 trillion to $12.6 trillion [86]. The responsibility for bolstering security has also increased as a result of extensive expansion. However, smart-home devices are prone to security threats and vulnerabilities. Thus, developers experience difficulty in maintaining the security of smart-home systems.

1.1 Existing Surveys

Several recent surveys focused on reviewing CPSs and smart homes separately. To the best of our knowledge, no reviews have been conducted on CPS in smart homes in terms of knowledge-based techniques. In this section, we discuss the novelty of our study and compare it to other surveys. Table 1 summarize the surveyed papers.

Ref. [5] discussed future technologies for smart houses based on the IoT. This study also highlighted the advantages of IoT-based smart-home devices in terms of quality, reliability, and security. In [90], the authors provided an overview of the demand-response potential of smart buildings and discussed the mechanisms to mitigate attacks at both the cyber and physical layers. Terence et al. [53] defined seven major requirements for building smart homes using IoT. Ref. [121] analyzed and predicted the main technological and scientific trends in the development of smart homes over the next decades. A survey [28] focused on securing smart homes by detecting abnormal home and user behavior in the homes and then responding to threats. Gong et al. [42] discussed the architecture and framework of smart buildings in cyber-physical social systems (CPSSs). Moreover, they proposed a CPSS-based smart-building operation framework. The framework proposed by Stojkoska and Trivodaliev [106] aimed to close the gap between modern smart homes and future IoT-based smart homes. Ref. [16] provided crucial insights into the privacy, security, and trust challenges associated with IoT devices in home environments. In [8], vulnerabilities in smart applications, the potential threats they pose, and the current state-of-the-art security mechanisms available to address these issues are presented.

The proliferation of communication between the cyber and physical worlds is a major challenge because a large amount of data are produced [25]. Tavcar and Horvath [106] reviewed data collection and analysis in real-time to support data-driven decision making. As presented in [46], existing solutions to protect the physical, communication, processing, and storage components of cyber systems, such as cryptography, intrusion-detection systems, and game theory, must be considered in specific areas, such as smart health, smart transportation, smart grids, smart homes, and public security. The authors also stressed the importance of human error. Ahmad et al. [2] discussed infrastructural transformation to smart cities, considering the CPS system as the pillar block for smart cities. For example, the smart-city ecosystem in India has been improved by building robust networking and enabling technological services. The analysis of security issues at various CPS-layer architectures as well as the risk assessment and methods of securing CPS were presented in [10]. Ref. [112] analyzed the main CPS security threats, vulnerabilities, and attacks, as well as cryptographic and non-cryptographic CPS security solutions. The threats were categorized based on the three layers of the CPS, and suggested solutions to these threats were addressed in [62]. Ukachi [85] defined cyber-physical security threats for smart buildings, the negative consequences of these threats, and the mitigation and defense mechanisms. In [110], authors discussed the security threats and challenges in several applications of CPS. Mishra et al. [80] provided a complete scientific review of many new technologies, including their objectives, scope, and purpose of next generation CPS. In [111], a novel integrated learning methodology is presented that effectively detects CPS attacks and accurately identifies the type of attack in real-time. Ref. [71] discussed the CPS challenges of building automation systems within the framework of the increasing openness and connectivity of intelligent buildings.

Table 1. Related review papers.

	Techniques		Domains		
Reference	Data-driven system	Knowledge-based system	IoT	CPS	Smart home
Mussab et al. [5]			✓		✓
Jessamyn et al. [28]	✓				✓
Tavcar and Horvath [108]		✓		✓	
Ahmad et al. [2]				✓	
Terence et al. [53]			✓		✓
Junjian et al. [90]				✓	
Adam et al. [121]					✓
Kai et al. [42]	✓				✓
Hadi et al. [46]	✓			✓	
Yosef and Qusay [10]		✓			✓
Yaacoub et al. [112]				✓	
Nam and Shailendra [62]			✓	✓	
Ukachi [85]				✓	✓
Amit et al. [110]				✓	
Stojkoska et al. [106]			✓		✓
Conti et al. [25]				✓	
Mishra et al. [80]	✓		✓	✓	
Wang et al. [111]	✓			✓	
Li et al. [71]				✓	
Buil-Gil et al. [16]			✓	✓	✓
Ansari et al. [8]			✓	✓	✓
Gaba et al. [39]	✓		✓	✓	
This article		✓	✓	✓	✓

1.2 Motivation

The proliferation of IoT devices has introduced new attack surfaces that bad actors can potentially exploit to gain unauthorized access to sensitive data or disrupt system operations [8] [16]. Smart homes are prime examples of IoT systems that are deeply embedded in our personal lives. The widespread use of smart home technologies has resulted in new conveniences, as well as unique security and privacy threats that must be addressed. Moreover, with the emergence of 6G [91], Industry 4.0 [59] and the integration of heterogeneous systems [35], data security and privacy issues have been exacerbated, further underscoring the need for rigorous security analyses. As these systems collect and process personal data, concerns about misuse and unauthorized access have arisen. This study examines the knowledge-management processes of smart-home operations from a cybersecurity perspective. It critically analyzes how smart-home technology obtains, stores, processes, interprets, and shares information. By reviewing these systems, we aim to identify vulnerabilities, security risks, and areas for improving user privacy and security.

1.3 Contributions

This work presents a comprehensive survey on cyber-physical security in smart homes, analyzing existing security techniques. It aims to guide researchers by identifying future research directions and leveraging smart home contexts to enhance security. The key contributions are summarized as follows:

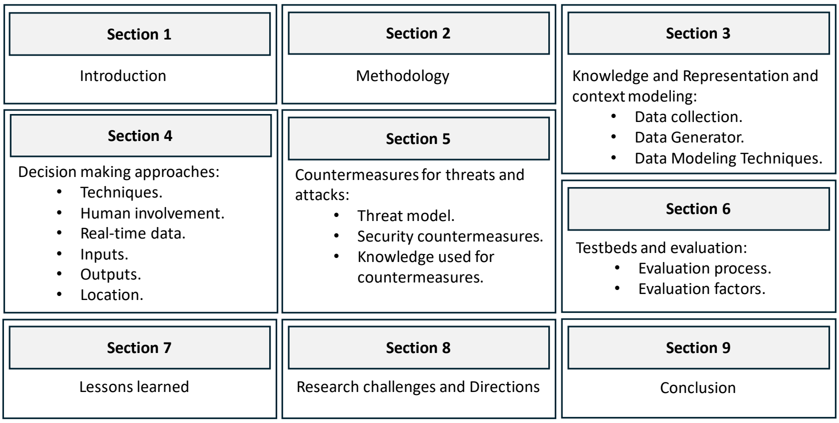


Fig. 1. Structure of this survey paper.

- We provide a detailed overview of knowledge representation and context-modeling methods used in smart homes to identify sensing and actuating data.
- We survey decision-making approaches, their input and output data, and the real-time data required for some of the proposed approaches. We also present a taxonomy of decision-making locations.
- We identify several threat models primarily for smart-home systems. Moreover, we discuss security countermeasures that can mitigate the proposed attacks and threats to smart homes.
- We articulate smart-home test-beds and evaluation methods concerning the number of users and devices, as well as the types of platforms and protocols used in each technique. We present figures showing the evaluation settings and goals.
- Furthermore, we summarize most of the lessons learned from the reviewed studies. Finally, we highlight the open challenges and discuss future research directions toward security in smart-home systems.

1.4 Review Structure

The reminder of this paper is structured as follows. Section 2 details the literature review method. Section 3 covers contextual information and modeling techniques. Section 4 reviews the reasoning mechanisms for smart home data. Section 5 describes countermeasures against attacks. Section 6 analyzes current evaluation methods. Section 7 presents lessons learned. Section 8 offers a summary and future research directions. Section 9 concludes the review. Figure 1 illustrates the overall structure and organization of this study.

2 METHODOLOGY

This review follows the steps proposed by Kitchenham [63] [64]. The literature search and article selection for evaluating knowledge-based cyber-physical security in smart homes published between 2012-2024 are illustrated in Figure 2. We focused on scientific databases including Google Scholar, Scopus, Springer, IEEE eXplore, ACM Digital Library, Wiley Interscience, and Taylor & Francis Online.

2.1 Research Questions

The use of CPS for smart homes has grown considerably over the last decade. However, to the best of our knowledge, no reviews have been done on knowledge-based mechanisms in smart

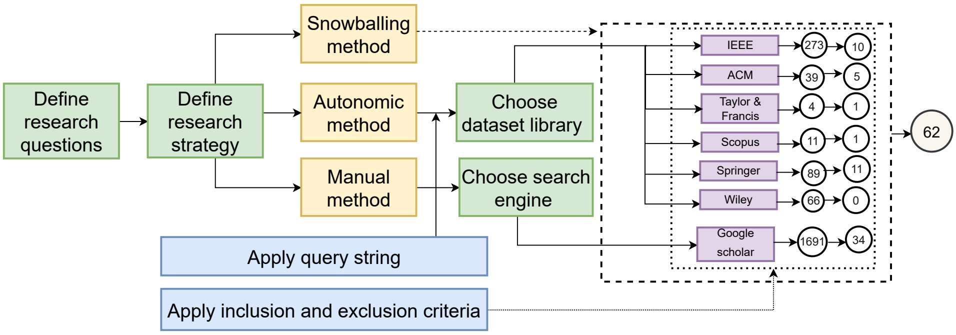


Fig. 2. Search and selection process.

homes. Hence, we aim to bridge this gap by reviewing this subject. Our analysis was guided by the following research questions (RQs):

- RQ1: How should model the smart-home knowledge be represented and modeled?
- RQ2: What are the decision-making techniques, and how do these techniques capitalize on the data produced in smart homes?
- RQ3: How is knowledge used for countermeasures against attacks and threats?
- RQ4: What evaluation strategies are implemented for evaluating smart-home systems?
- RQ5: What are the open issues that need to be investigated further regarding the security of smart homes?

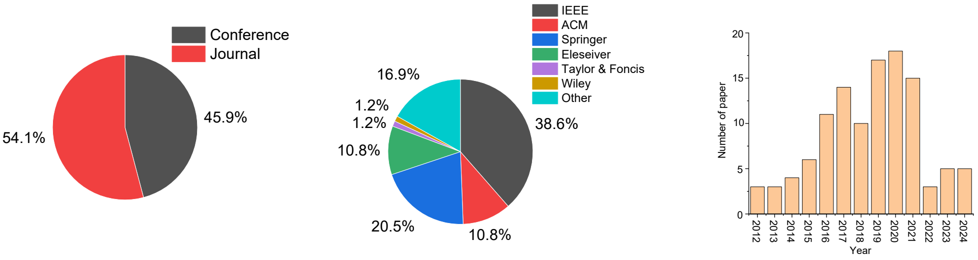
These research questions were answered by fulfilling the contributions of this study. Section 3 covers knowledge representation and context modeling (RQ1). Section 4 addresses smart-home reasoning with contextual information (RQ2) and countermeasures (RQ3). Evaluation methods concerning users, devices, protocols, and platforms are discussed for RQ4. Finally, RQ5 explores future security directions for smart-home systems. Table 2 outlines each research question and its rationale.

Table 2. The rationale behind the research questions.

Research question (RQ)	Rationale
RQ1: How to represent and model the smart home knowledge?	To define context types of smart home systems and find out the pieces of knowledge captured in these systems. With more focus on knowledge representation and modelling techniques.
RQ2: What are the decision-making techniques, and how do these techniques capitalize on data produced in the smart home?	This research question examines the decision-making approaches used in the smart home and explores its inputs, outputs, and location.
RQ3: What is knowledge used for countermeasures against attacks and threats?	This research question focuses on determining and discussing the countermeasures in the smart home against attacks and threats, which are mentioned in this paper.
RQ4: What evaluation strategies are practiced for evaluating smart home systems?	This research question aims to discover how the previous studies evaluate its proposed approaches.
RQ5: What are the open issues to be further investigated in regard to the security of smart homes?	To determine a possible research area in smart home security.

2.2 Data Analysis

Figure 3 provides a detailed overview of the publications reviewed during the literature evaluation. This visualization encompasses a 12-year timeframe to capture the most recent and relevant research developments in the field. Figure 3a represents percentage of the paper venues, which is nearly the same for both conferences and journals, at 46% and 54%, respectively. Considering the specific details in Figure 3b, the first striking feature is that the IEEExplore dataset had the highest proportion. Moreover, the most significant number of studies were published in the last three years (Figure 3c).



(a) The percentage of Conference venue to Journal.

(b) The percentage of selected articles base on publication.

(c) Number of studies published per year.

Fig. 3. A comprehensive overview of the articles reviewed in various academic and professional venues over a 12-year period.

3 KNOWLEDGE REPRESENTATION AND CONTEXT MODELLING

This section outlines the stages data undergoes before processing for decision-making. Figure 4 illustrates the three main data processing procedures in smart-home systems: collection (Subsection 3.1), generation (Subsection 3.2), and modeling (Subsection 3.3). Each stage's essential principles and procedures are thoroughly explained.

3.1 Data Collection

This subsection introduces context types in smart-home systems, explaining context-awareness and its role in enhancing system efficacy, efficiency, and relevance. It covers how smart-home devices collect information and discusses the manual and automatic data entry methods.

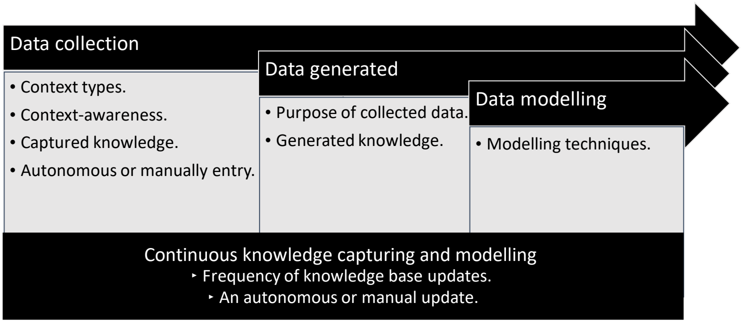

Data collection
• Context types.
• Context-awareness.
• Captured knowledge.
• Autonomous or manually entry.
Data generated
• Purpose of collected data.
• Generated knowledge.
Data modelling
• Modelling techniques.
Continuous knowledge capturing and modelling
‣ Frequency of knowledge base updates.
‣ An autonomous or manual update.

Fig. 4. Dimensions of the three steps through which data in smart homes passes. It demonstrates the principles covered in each level, as well as the method of updating this data based on survey papers.

3.1.1 Context Types. Context is information that can be used to describe the circumstances of an entity, such as a person, location, or object, and is deemed pertinent to the interaction between a user and an application; this is referred to as contextual information [1]. Dynamism, stochasticity, and heterogeneity are inherent characteristics of context [68]. As mentioned in [9], seven types of contextual information exist: personal, activity, physical, device, systematic, application, and environmental. Table 3 presents the different types of contexts used in smart homes. Personal context involves data about an individual's habits and behaviors within their home, enhancing user experience through smart home technologies. The activity context captures specific actions and movements of users and objects, improving experience via context-aware functionality. Physical context pertains to the operational states of devices and sensors, optimizing comfort and efficiency [82]. Device context includes operational characteristics and capabilities, ensuring smooth interoperability and resource optimization [51]. Systematic context categorizes user behavior patterns for standardized representation. Environmental context provides data on ambient conditions, contributing to awareness of living space conditions. For example, the Privacy via AnomaLy-detection System (PALS) [34] assumes a physical context, such as location, time, activity, and roles. In [72], the environmental context, such as temperature and dimmable light, were considered. Additionally, in [33], [104], [103], [94] [109] and [88], the context of a home environment was sensed, and in [92], the smart-home environment was sensed based on security functions. Ref. [102] the user-activity context to correctly understand the behavior of users. The environment and application contexts are described elsewhere [68]. Situational information provide the context for initializing a communication network [31]. The user context was captured in [113] to determine the user expectation from the behavior of applications, and Ref. [4] covered the personal data that should be protected. In [75], the systematic context for collecting network traffic was discussed. Regarding the application context, [36] and [119] are concerned with analyzing smart applications to detect over-privileges in the SmartThings framework or types of attacks. Additionally, [81], [56] and [17], described the application context. In [60], contextual information from the types of user and physical contexts was presented. The device context is described elsewhere [43]. Amir et al. [94] focused on the number of applications and users in smart homes.

3.1.2 Context-awareness. This describes a system with the ability to understand the intent of users to improve its efficiency. Sikder et al. [102] proposed an Aegies+ platform-independent context-aware security framework for smart homes that detects malicious activities. In [68], the authors presented a platform that enabled the development of context-aware applications that could be adapted autonomously at runtime. ContextIoT [56] is a context-aware permission model that restricts unauthorized device access and detects malicious activities in a smart home. A context-aware authentication framework introduced by Ashibani et al. [9] is being developed for smart-home applications to access devices.

3.1.3 Captured Knowledge. Determining the data type to be used in a decision is the first step in the decision-making process. In [101], 219 diverse policies were gathered from actual smart home users, including 33 limitation policies and 146 demand conflicts. Similarly, data were collected from 50 malicious users to assess the effectiveness of the proposed system against these threats. In [102], the sensor features and smart-home device states were obtained from day-to-day user activities as well as malicious data from the adversary model. In [34], cloud-service providers gathered information about a smart home from the sensed data. In [33], the data collected from IoT devices were stored remotely on the cloud and locally in the RES-Hub as a backup during a cloud outage, as well as to authenticate requests and issue commands that end devices can verify. Chi et al. [21] collected user-configuration information from applications and sent it to a cloud. In [107], features of home entities and set of concepts, devices capabilities, and security vocabulary were collected by cloud. ContexIoT [56] modifies the application code to add security-focused logic patches to the application to gather crucial running context. IOTGUARD [17] adds new logic to the source code of an application to collect data from it while it is running, including devices, events, actions, predicates that control device actions, and numerically valued properties of those actions. In [31],

Table 3. Smart Home Contexts.

Context	Reference Data Type	[72]	[101]	[102]	[34]	[33]	[21]	[81]	[107]	[56]	[32]	[17]	[31]	[113]	[75]	[104]	[89]	[9]	[119]	[94]	[60]
Personal	Location	✓																✓			
	Service provider				✓																
	Metadata	✓																			
	Status	✓																			
	Priority		✓																		
	Role				✓																
	Time				✓																
	Endpoint communication	✓			✓																
	User expectation of app's behavior													✓							
	Name															✓		✓			
	Gender															✓					
	Height/Weight														✓						
	User's calendar																	✓			
	Historical information																	✓			
	Number of users																			✓	
	Age																			✓	
Activity	User			✓																	
	Malicious			✓																	
	Location																				✓
	Posture																				✓
	Object																				✓
Physical															✓						✓
Device	Model	✓																			
	Description								✓		✓										
	Policies		✓																		
	States			✓		✓															
	Function								✓												
	Location								✓												
	Controller								✓												
	Control and flow attributes									✓											
	Trigger										✓										
	Action										✓										
Systematic																✓					
Application	Configuration information						✓														
	Semantic rules						✓														
	Meta data							✓					✓								
	Runtime information											✓									
	Situation information												✓								
	Abstract specification														✓						
	Behavior																		✓		
	Platform used																			✓	
	Number of application																			✓	
Environmental	Variable	✓																			
	Temperature																✓				

✓indicates this feature is present in the research.

application network connections were tracked. In [113], user expectations were derived from the behavior of a set of installed automation applications. The authors of [45] claimed that they collected raw data from smart-home devices in real time. In [119], wireless packets were gathered, including Z-wave and ZigBee data. SERENIoT [109] packet signatures from network traffic. Mahadewa et al. collected an abstract definition of application-layer protocols and the internal behaviors of entities [74] [75]. In [115], a monitoring system that gathers bathroom activities was proposed. The device state information gathered from the cloud was utilized by RES-Hub [33]. Moreover, infrastructure for smart homes is being developed, as shown in [81]. Ding et al. collected inter-app trigger-action interactions and physical-channel information from the application description [32]. In [104], data packets transferred over a network were obtained from a knowledge base. The user configurations were also obtained [88]. The approach in [9] uses static credentials, and contextual information. Additionally, exchanged-message semantic names are compiled in [120], and HoMonit, proposed in [119] collects wireless packets. In [29], the data gathered for the study comprises guidelines for developing Self-CPS, as well as existing reference models and architectures pertinent to this field.

3.1.4 Autonomous or Manual Entry. Two types of data-input methods are used during the collection process: manual and automatic. Lin et al. [72] suggested that an automatic manager reduces the manual input from users; however, in some circumstances, the user must still take action. In [101], users explicitly specified priorities and policies for smart-home devices. However, security analysts must provide input to HOMESCAN [75] [74]. In [34], the user provides feedback to an anomaly detection system. DepSys obtains user input to determine application priority and policy [81]. IOTGUARD requires user input for application configuration [17]. In smart-home applications, users set their expectations [113]. In [88], the user identifies the configuration of smart-home devices. However, the states of these devices are obtained autonomously [102]. An algorithm for automated categorization and decision making is proposed in [115]. An automatic instrumentation script was to extract the configuration information in [21]. The context-collection logic is responsible for gathering application variables [56]. HanGuard automatically sends situational information to the home router through a control channel [31]. According to Ashibani et al., this method does not require human interaction [9]. Additionally, HoMonit automatically captures the wireless traffic [119].

3.2 Data Generated

The data generated in smart homes empowers homeowners and residents with information and control, leading to increased efficiency, convenience, security, and well-being. This subsection discusses the purpose of smart-home data and the generated data.

3.2.1 Purpose of Knowledge. Smart-home data may be collected for several reasons. For example, [72] aimed to collect data to assist the system with reasoning about attacks and to respond appropriately to them. In [101], contextual information was used to identify user roles and consumer expectations for smart homes. Aegis built an activity context to distinguish between the benign and harmful uses of smart-home devices and sensors for various user behaviors and patterns [102]. HOMESCAN can discover security issues from its knowledge of smart-home implementation [75] [74]. Abnormalities in data collected from smart home device activities are presented in [34]. In [115], the goal was to recognize potentially life-threatening events. RES-Hub aims to provide resilience to smart homes when the cloud is unavailable [33]. Chi et al. collected application configurations to identify threats and minimise false alarms [21]. Depsys provides comprehensive solutions for specifying, detecting, and resolving conflicts in the home [81]. In [56], context information aide users in differentiating between benign and malicious behavior. Unanticipated physical interactions between applications have been addressed by IoTMon [32]. Celik et al. evaluated collected data based on a set of security and safety policies [17]. In [31], situation information was compared with policies to ensure that they originated from a legitimate home-area network phone. Expat captures this information from an installed application to ensure that user expectations are not violated [113]. The knowledge base in [104] is used to protect smart-home devices from network attacks. A user setup is required for devices to detect intrusions [88]. In [9], contextual information was used for the authentication process. This is used to infers policies in which entities gain access control over devices based on entity names [120]. Wireless traffic is used to detect security threats in smart home applications [119]. The goal of SERENIoT is to examine the network traffic to and from IoT devices to detect and prevent suspicious packets and connections [109].

3.2.2 Generated Knowledge. The data collected in smart homes provide important knowledge that can be used to make decisions. In [72], the gathered contextual data was used to develop resource description framework (RDF) triples to describe the relationships between the elements in smart homes. User priorities are produced from user credentials and device policies [101]. In [102], a context array of several user behaviors was constructed. A local labeled transition system (LTS) representation of system integration was generated from the collected traces [75] [74]. Access-control decisions were produced in the context of smart homes in [34]. Chi et al. built a risk-ranking model for cross-app interference threats [21]. In [81], the dependency information of a smart-home application was inferred. In [32], inter-app interaction chains were built using an application analysis. Expat was used to generate policies to be enforced on smart-home platforms [113]. Device-interaction rules were established to warn users of threats [88]. User-defined rules were used to create home-security policies, that were subsequently applied to devices [120]. The operations of SmartApps were derived from encrypted traffic in [119]. By separating packets and creating distinctive signatures, the behavior of the devices was retrieved in [109].

3.3 Data-Modelling Techniques

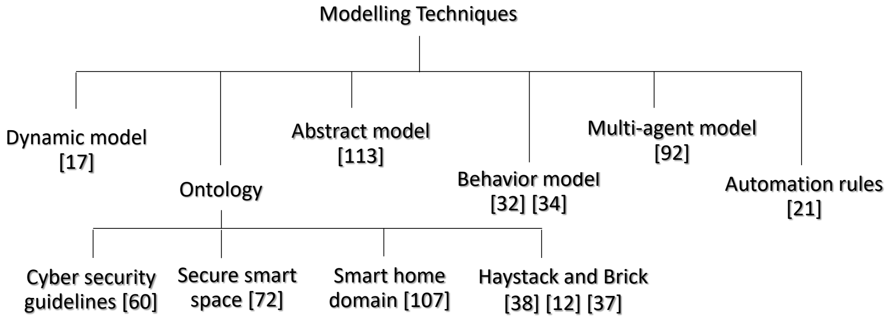


Fig. 5. Types of modelling techniques.

Figure 5 shows a taxonomy of the six different modelling approaches used to develop models and representations in the context of smart-home systems. Each modelling strategy has a distinct purpose and address different aspects of data modelling in smart-home spaces. An ontology known as secure smart-space ontology (SSSO) is used to model a smart-space as a knowledge [72]. Sofia et al. [34] represented the smart-home context in a knowledge graph to aid in defining of rules for controlling data access. In [21], each IoT application was represented by automation rules that adhered to the trigger-condition-action (TCA) paradigm to extract the application's rules for detecting threats. Tao et al. [107] proposed a smart-home domain ontology, an ontology-based device description model, and ontology-based security management to fulfill heterogeneity and security requirements. In [32], the interaction behavior of physical channels was modeled by assigning appropriate values to various physical channels to determine their distances from other. A unified dynamic model is proposed in [17] to represent the runtime-execution behavior of applications in states and transitions. In [113], an abstract model of an appified smart-home platform was suggested to represent user-expectation invariants. Khan et al. [60] suggested an ontology called the Cyber Security Guidelines Ontology (CSGO) to express knowledge about security rules for interoperability and comprehension among smart-home users. In [92], a smart-home network used a multi-agent approach to achieve shared security objectives. The Haystack [38] and Brick ontologies [12] [37] are unified semantic metadata standards for building assets and their interactions that enable the successful interoperability and automation of building systems and analytics. The RDF is a semantic technology that standardizes the definition and use of metadata [30]. The reference architecture for self-adaptive CPS was constructed by integrating the guidelines of the process based on software architecture methodology with the approach suggested in [29].

3.4 Continuous Knowledge Capturing and Modelling

This subsection focuses on the pattern of knowledge-based updating and manual and automatic updates. These findings provide insight into the accuracy of the retrieved data and whether human intervention is required.

3.4.1 Frequency of Knowledge-base Updates. After each requested service is completed in the smart system, the RDF triples are updated, as shown in [72]. User-priority and device-policy lists are updated based on the user expiration date in the system and each time another policy is issued, respectively [101]. Every time a new device or application is introduced to the system, the Aegis framework updates the training dataset [102]. HOMESCAN updates its collected information whenever new states are inferred [75] [74]. When PALS receives feedback from a user, it updates its knowledge graph [34]. Regular status updates from the cloud are sent to RES-Hub [33]. DepSys updates the dependency information of an application when conflicting dependencies are observed [81]. The environmental variables of the applications change during operation [56]. The security policies of HanGuard are updated when a mobile phone is connected to the network [31]. Expat modifies previous rules after creating an instrumented rule file [113].

3.4.2 Autonomous or Manual Updating. The manager is responsible for updating the RDF triples automatically [72]. Kratos automatically finds any expiration dates and new additional policies [101]. In [102], the training dataset was automatically updated when a new device was introduced. Additionally, HOMESCAN updates its knowledge automatically [75] [74]. In [56], when an application is executed, its environmental variables are updated automatically. In [109], the policies were automatically updated.

4 DECISION MAKING APPROACHES

The smart home context generates vast data from appliances and IoT devices, crucial for decision-making. Smart-home reasoning systems ensure occupants' efficiency and comfort by determining the best actions [77]. Self-adaptive systems make dynamic decisions to meet functional and non-functional criteria [97]. Effective decision-making improves quality, efficiency, reduces risk, and enhances outcomes. This section discusses decision-making approaches, their inputs and outputs, and whether they are performed locally or remotely. Key decision-making topics are also presented. Figure 6 presents a comprehensive taxonomy of decision-making approaches in smart home systems. It classifies these approaches based on input data types, incorporation of real-time data, output nature, decision-making location, and human intervention or interaction in the workflow.

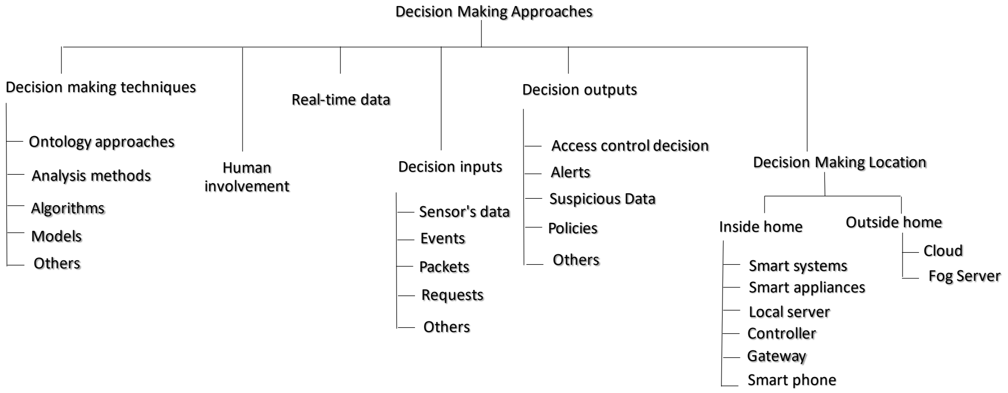


Fig. 6. Decision making approaches taxonomy.

4.1 Decision Making Techniques

Better decisions in line with smart-home concerns can be made by choosing the most appropriate method for certain situations. Building a successful and functional context inside a house requires excellent decision making. Table 4 summarizes the decision-making techniques used in these studies.

4.1.1 *Ontology Approaches.*

A secure smart-space ontology was suggested to help analyze and reason the state of the smart space in a way that is intelligible by machines [72]. A context-based ontology [60] was developed to help security managers make decisions on information security.

4.1.2 *Analysis Methods.*

In [75] and [74], a hybrid analysis that included dynamic testing, white-box analysis, and trace analysis was proposed. Jia et al. [56] suggested using taint analysis to monitor runtime data and then identify the data source while displaying contextual information to the user. Ding and Hu [32] used a risk-analysis mechanism to assess the dangers of discovered chains of inter-app interactions. Side-channel analysis was performed by HOMONIT to monitor encrypted wireless traffic [119]. To identify the weaknesses in the framework architecture, an empirical investigation of the SmartThings platform and its applications was conducted [36].

4.1.3 *Algorithms.*

Arun and Reza proposed a logic-based algorithm for detecting typical user behavior at access points and demanding user authentication [57]. IOTGUARD built a graph algorithm to extract the events and actions of applications [17]. Ref. [92] presented a multi-agent algorithm compromising agents representing multiple devices or subsystems in the home that work together to achieve a common goal. Furthermore, context-aware systems made decisions based on contextual information such as user location, behavior, and preferences [34] [9]. An activity-recognition algorithm used sensors and data analysis to recognize and anticipate human actions [102] [115].

4.1.4 *Models.*

Chi et al. presented a risk-ranking model that could assess the severity of identified cross-app interference risks [21]. The user-specific score threshold for voiceprint verification was calculated using a Gaussian mixture model (GMM) [96]. Individual agents in the Beliefs, Desires, and Intentions (BDI) model functioned as autonomous agents in decision making [92]. In [3], the STRIDE and DREAD models identified threats in the network.

Table 4. Decision Making Methods.

Techniques	Reference	Capabilities	Limitations
Ontology approaches	[72]	Dynamic access control with high performance capabilities.	The suggested default policies are not appropriate for all possible cases. Manual labor is still required even if the manager strives to minimise human intervention.
	[60]	Support automation.	There are limited input sources and a lack of implementation.
Analysis methods	[74] [75]	Effectively defend against various attack models.	Interacting with the system is necessary for the security analyst.
	[56]	Effectiveness in identifying attack.	Based on a user-made decision with overhead computation.
	[32]	Discover unexpected behaviors.	Based on user involvement with added overhead.
	[119]	Detect the misbehave of smart applications.	Introduce overhead computing.
	[36]	Used combination of analysis mechanism.	Conservative.
Algorithms	[57]	Local decision.	Based on user behavior only.
	[17]	Reduce the overhead.	Need user interaction.
Models	[21]	Users do not need to set security objectives.	False alarm.
	[96]	Used dynamic threshold method.	The likelihood of a false rejection rate still exists.
	[92]	An autonomous entity.	Cannot make decision on real time data.
	[3]	Conscious of potential physical harm.	
Others	[104]	Controls the user's firewall rules in a flexible manner.	Small scale of the test environment.
	[94]	Static risk level.	Human intervention.
	[115]	Adjustable to each user.	As the number of possibilities is limited, a false alarm may occur.
	[7]	Fog services.	No implementation in real-world environment.
	[89]	Reduce latency.	Suffer from delay.
	[88]	Increasing user awareness.	Prone to human errors.
	[45]	Real-time decision and low latency.	
	[58]	Done inside the smart home appliances.	Not evaluate its effectiveness.
	[43]	Convenient deployment.	User intervention caused delay.
	[109]	Distributed decision.	Complicated and take time to develop.
	[34]	Dynamic reasoning.	Coarse-grained policies.
Not mentioned	[101]	It has the ability to control a variety of individuals and devices.	If there are too many difficult conflicts, it could result in stalemate.
	[68]	Self-adapt and self-aware.	Complex.
	[81]	Flexible.	Non-critical safety system.
	[33]	Reliability.	Complex.
	[31]	Static phone operating system.	Flow decision cache limit.
	[113]	Support remote and local checking.	High overhead.
	[69]		Human intervention.
	[9]	Real-time and continuous authentication.	Human intervention.
	[120]	Both inputs and outputs are sent securely.	Single point failure for controller.
	[4]	Support local and remote security.	

4.1.5 Others. Security-management providers have been presented in order to identify and address the security/privacy risks for IoT [104]. The system proposed by Dutta et al. utilized semantic web technologies to make access-control choices [34]. In [115], an effective-reasoning module was proposed to identify user-critical scenarios and supply data for lifestyle-pattern reasoning and daily function-monitoring modules. ICN-iSapiens deploys intelligent monitoring and control applications effectively and efficiently by using information-centric networking [7]. Edge servers can offer localized computation and storage [89], and a configuration tool was proposed to help users develop device-interaction rules [88]. Mutual authentication between smart devices and smart-home gateways was proposed in [45]. The self-signing technology [58] maintains the integrity of its security framework. The proposed method for mediating network communication across devices in the same network was made possible by software-defined networking [43]. In [109], Blockchain was proposed to detect anomalies in networks. A risk-based permission system presented in [94] reduces malicious applications in the system [29]. De et al. proposed a control loop technique that entails ongoing system monitoring, data analysis, planning of necessary adaptation actions, and executing those actions to ensure the system can adapt to runtime changes autonomously [29].

4.2 Human Involvement

An automated manager significantly reduces human interaction while still requiring user entry [72]. Kratos [101] involved users in defining their policies and priorities in addition to resolving difficult conflicts in the system. In [74] and [75], a security analyst required specification-extraction process for the proposed system. When conflicts occur, DepSys may require human input to adjust the policy [81]. In [17], the user entered the application configuration and runtime prompts. A context-based ontology [60] involved the user in the modelling of security guidelines. Security and Privacy for In-home Networks (SPIN) keeps the user in control to stop undesirable traffic flows [69]. Manju and Albert introduced an approach that allowed users to configure smart-home devices within the internal network of smart homes [88]. In [9], the authentication information was input by users for a security configuration. Alshaboti et al. proposed a user-defined network policy that incorporates users into a security system [4].

4.3 Real-time Data

Real-time notifications are provided to users via Aegies+ [102]. A sensor knowledge graph contains real-time data gathered from smart-home sensors [34]. DepSys utilises real-time data to evaluate its approach [81]. The platform proposed in [7] was implemented using real-time data. In [89], the processing of sensor data in real-time was facilitated using edge servers. SecFHome enables a smart gateway to analyze collected data in real time [45]. Contextual data were obtained and evaluated in real time from secured sources [10]. HoMonit captures wireless channel packets and detects misbehaviors in real time [119]. Additionally, SERENIoT monitors IoT-devices data in real time [109].

4.4 Inputs for Decision-Making Process

Different inputs are required depending on the approach employed. However the following components are present in many smart-home decision-making methods.

4.4.1 Sensor Data. An effective reasoning module was suggested based on the sensor data that appear in the activity monitoring system proposed [115]. In [33], data collected from smart-home devices and regular cloud-status updates were employed as inputs to the proposed system.

4.4.2 Events. In [72], a succession of events that took place in a smart space was processed by an autonomic manager to assess the situation. In [17], sensors collected application-specific events and their accompanying actions and predicates, and then sent them to a hub/cloud-based processing device for analysis.

4.4.3 Packets. Data packets transmitted across the home network were analyzed in [120], and control packets of the local network were analyzed in [43]. Side-channel data from secured wireless traffic were utilized as the input in [119]. In addition, IoT-device IP packets were tracked in [109]. In [4], network-traffic analysis was used to make decisions.

4.4.4 Requests. In [45], user-access requests for smart-home devices were monitored to maintain secure communication. Tyche used application permission requests for device services as an entry into a risk-based permission model [94].

4.4.5 Others. The priority assignment information and device rules are entered by the user in [101]. System implementation, test cases, and prior knowledge serve as the foundation for HOMESCAN [74] [75]. In [34], decision-making was based on the user context and device type. The context reasoning of iCasa on its objectives, resources, and runtime architecture are presented in [68]; the configuration details are captured in [21]. DepSys analyzes the metadata of applications to discover threats [81]. In [57], this was based on user actions and behaviors when making decisions. Amadeo et al. [7] recorded stakeholder inputs, user preferences, and dynamic context-related aspects. Inter-procedural control and data-flow information defined the context of a smart-home in [56]. IoTMon can record physical interactions [32]. HanGuard gathers runtime data from the mobile devices of users [31]. In [96], the log ratio score was used by an analysis algorithm to make decisions. In [113], user expectations were identified. The design of the SmartThings programming framework was discussed in [36]. Threats related to networks were analyzed in [3]. The user and physical contexts were used to build an ontology [60]. In [104], network activity was monitored. The device-security features were captured in [69]. DBI were inputted into the proposed model in [92]. The devices connections were verified as secure in [88]. A module's codes are examined to ensure their integrity [58]. In [9], information on a user's location, profile, calendar, request time, and access activity patterns was gathered. Additionally, Kumar et al. captured data from home devices [67].

4.5 Outputs for Decision Making-Process

Decision-making procedures generate outputs that aid decision-making and reach conclusions; thus, the outputs of decision-making methods depend on their input.

4.5.1 Access-Control Decision. An in-context sensitive action is provided by ContexIoT [56]. The HanGuard router is responsible for making access decisions [31]. Tyche implemented risk-based access-control decisions for IoT systems. In [58], the authors supported access control to identify user permissions. Services and data accessed by platform components were discussed in [68].

4.5.2 Alerts. In [92], security alerts were sent to security agents when an attack on a smart-home network was detected. The user was informed of the incursion using the approach suggested by the authors [57]. In [88], unwanted contacts triggered intrusion alarms.

4.5.3 Suspicious Data. Sivaramman et al. proposed a security solution for detecting suspicious behaviors [104]. The SPIN system distinguishes between normal and suspicious behavior at the network level [69]. SERENIoT [109] differentiates between malicious packets and connections using security policies.

4.5.4 Policies. Adaptive security policies are applied to threats that occur in the proposed system [72]. Kratos uses a policy negotiation algorithm to resolve user disputes and optimize different conflict policies [101]. Moreover, Expat implements contextual access control policies for smart-home platforms for smart-home applications [113]. PALS provides context dependent access control policies [34].

4.5.5 Others. In [75] and [74], security issues were discovered in smart-home systems. The system proposed in [115] used a reasoning algorithm to generate daily activity reports and send notifications. RES-Hub generates commands according to user specifications [33]. HOMEGUARD produces analytical findings [21]. DepSys addresses the conflicts detected using smart-home technologies [81]. Real time services were provided in [7]. Ding et al. proposed an interaction-chains algorithm to measure the risk level of interactions [32]. IOTGUARD recognizes hazardous and insecure states in various applications [17]. The user-authentication method differentiates between legal and illegal users in the system [96] [9]. Design flaws were discovered in [36]. Risk assessment that identifying cyber-physical risks were presented in [3]. Security guidelines were provided in [60]. User commands were issued to carry out various activities in [89] and [120]. The flow [43] and security decisions [4] were generated from smart-home network services, and inappropriate behaviors were identified in applications [119].

4.6 Decision Making Location

Our analysis categorizes studies and approaches based on the location of decision-making within smart home solutions. We differentiate between decision-making processes occurring inside the home and those relying on external decision-making, as illustrated in Figure 6.

4.6.1 Inside the Home. The attack surface for potential cyberattacks can be reduced by securing devices and data inside the home. Thus, smart-home networks is more challenging for attackers. Six locations in the home process data.

- Smart Phone. HOMEGUARD [21] collects configuration information using a configuration collector to detect cross-app interference threats without requiring users to identify the security objectives. In [81], the authors collected an application's metadata to resolve conflicts at installation and during the run time. Although DepSys is flexible and allows dynamic program addition and removal at runtime, safety criteria were not considered. HanGuard suggested a monitoring on user phones to create access control for applications [31]. In [96], the parameters of the GMM were compared with the dynamic threshold score to distinguish between legitimate and unauthorized users.
- Gateway. SERENIoT [109] monitors network traffic to and from IoT devices to detect and block suspicious packets and connections. The reasoning module proposed in [115] uses raw data from (presence, humidity, and microphone) sensors to generate a daily activity report, trigger notifications, and send alerts. In some cases, a decision made in the gateway near a smart home may result in a false alarm. An autonomic manager [68] reasons over three types of models, which include available services, goals, and architecture, before making a decision to grant an application that is self-aware and self-adapted for the current situation, unless it is complex. In [33], when the cloud is unavailable, RES-Hub collects data from sensors and sends user specifications as commands to the actuators. A tool informing users of network intrusions was proposed by Pillai et al. In [45], a smart gateway made the decision to securely collect and process data transmitted by smart devices in real time [88]. In [9], a secure gateway was proposed based on gathering necessary contextual information and evaluating access to smart-home devices.
- Controller. Ref. [43] suggests the use of controller and non-controller devices. Controller devices only interact directly with controllers or the cloud to minimize privileges; thus, controllers issue requests from smart home devices. IoT controller devices in the home are used to control smart homes, as shown in Sovereign [120], which suggests that a local controller manages the authentication and access-control system.
- Local Server. Jose et al. [57] detected user activities and behaviors at multiple access points. These behaviors were compared with accepted user behavior to spot intrusions or attempted intrusions. This study analyzed and stored the database at home; however, this approach is not significantly more secure because of the possibility of hackers gaining access to the IoT devices. In [17], security services approved or rejected actions and used graph algorithms to reduce the burden of policy checking. To gather data from sensors and deliver commands to actuators, Qashlan et al. [89] developed smart-home multi-edge servers in addition to cloud storage. The edge nodes conducted transactions, whereas the cloud was used for extensive analysis and long-term archiving.
- Smart Appliances. In a study by Lin et al. [72], an autonomic manager was responsible for analyzing system events, such as user requests and threats, to produce adaptive security policies for IoT-based systems. The primary strength of this manager was dynamic, and responded to events in an adjustable manner instead of being dependent on default policies, which were inappropriate in some cases. The ontology suggested in [60] assisted the

security manager in making decisions regarding the user and physical situation for the smart-home devices. Kang et al. [58] provided security services for smart homes by ensuring device authentication, availability, and data integrity. They employed access control and self-signing mechanisms to defend themselves against threats.
- Smart-Home Systems. HomeScan [74] [75] seeks to identify as many security flaws as possible in partially implemented smart home integrations. Although it provides dynamic analysis, the security analyst must interact with the system to perform the required functionalities. The authors of [3] used threat analysis and risk assessment to identify threats and system-affected areas that investigators should focus on. The authors in [69] suggested the use of a privacy manager that allows users to manually prohibit IoT devices on a network that exhibit potentially unfavorable behavior.

4.6.2 Outside the Home. Sending data from smart home devices to cloud servers for analysis, storage, and extra processing allows for the remote processing of smart-home data on the cloud.

- Cloud. In [101], a policy manager evaluated device policies and user priorities collected by the backend server, started user negotiations to settle conflict needs, and created the final policies. The policy manager can manage different users and devices; however resolving difficult conflicts may become impossible if too many exist, thereby necessitating user engagement. Dutta et al. [34] proposed a cloud-service provider that is responsible for dynamic reasoning based on user context, devices, and attributes. In [56], a user could make an informed choice regarding the control flow, data flow, and runtime value, with the aid of the cloud-based permission service to conduct access-control operations. By examining SmartThings applications, IoTMon [32] directs developers and users to reduce the risk of inter-app interaction chains. Moosa et al. [113] proposed a satisfiability modulo theories (SMT) solver on a platform server to verify that policies satisfy the user expectations. The Samsung SmartThings programming framework was analyzed in [36] to identify design weaknesses using an empirical analysis including static analysis techniques, runtime testing, and manual analysis. In [104], the security management provider was responsible for identifying unusual network activities. A framework for multiple agents engaging in complicated reasoning is known as BDI modelling [92]. Within the cloud service layer, BDI reasoning for agents is used to detect network threats. In [67], the context of a smart home was responsible for the authentication procedure. Ref. [119] presented a system for monitoring the smart-home applications of SmartThings based on encrypted wireless traffic called HoMonit. In [4], the proposed security services monitored network traffic and issued security alerts. Tyche [94] proposed a permission-based model to categorizes access requests into three risk levels to assist users in decision makings.
- Fog Server. Fog computing extends cloud capabilities by offering computation and storage resources at the edge of the network, closer to IoT devices and end users [48]. Amadeo et al. [7] suggested remote cloud and fog layers to enable real-time systems to monitor and manage smart home applications.

5 COUNTERMEASURES FOR THREATS AND ATTACKS

In the following subsections, we present common countermeasures and best practices for protecting smart homes against different types of attacks, such as adversarial attacks. We discuss the threats specific to each category and the corresponding countermeasure approaches and strategies both in the literature and those used in existing systems. Finally, the knowledge employed in the countermeasures is discussed (Subsection 5.3). Figure 7 presents a comprehensive compilation of

12 threat models and the corresponding countermeasures specifically tailored to the smart-home domain, as proposed and analyzed by existing studies.

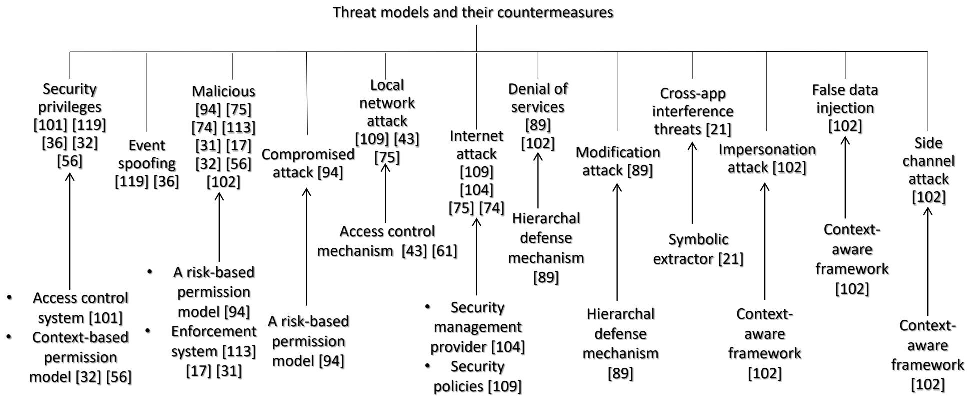

Threat models and their countermeasures
Security privileges [101] [119] [36] [32] [56]
Event spoofing [119] [36]
Malicious [94] [75] [74] [113] [31] [17] [32] [56] [102]
Compromised attack [94]
Local network attack [109] [43] [75]
Internet attack [109] [104] [75] [74]
Denial of services [89] [102]
Modification attack [89]
Cross-app interference threats [21]
Impersonation attack [102]
False data injection [102]
Side channel attack [102]
• Access control system [101]
• Context-based permission model [32] [56]
• A risk-based permission model [94]
• Enforcement system [113] [17] [31]
A risk-based permission model [94]
Access control mechanism [43] [61]
• Security management provider [104]
• Security policies [109]
Hierarchal defense mechanism [89]
Hierarchal defense mechanism [89]
Symbolic extractor [21]
Context-aware framework [102]
Context-aware framework [102]
Context-aware framework [102]

Fig. 7. Existing related studies provide examples of threat models for smart homes, along with accompanying countermeasures.

5.1 Threats Model

Smart homes face increasing threats as the number of connected gadgets rises. Therefore, creating threat models is essential to identify potential security vulnerabilities in these automated environments. Key highlights of common threat models for smart homes are presented below.

5.1.1 Security Privileges. Four types of security-privileges threats exist: First, over-privileged control, as in [101] [119], where smart devices are controlled by users in ways that are beyond what is necessary for their intended functioning, which may lead to unauthorized device access. Fernandes et al. [36] discussed architectural faults in SmartApps that result in overprivilege. Second, privilege abuse and unauthorised system changes by smart home users are potentially dangerous because they may lead to the installation of unknown applications [101]. The third is privilege escalation, which is the expelling of system users possessing devices [101]. Previous studies highlighted how malicious program can access unapproved devices and sensitive event data due to the attackers escalating their privileges and causing security issues [32] [56]. Finally, transitive privilege occurs because of the multi-user, multi-device smart-home access control, which is insufficient, inaccurate, or uncaring [101].

5.1.2 Event Spoofing. A false event is created and legitimately activates certain devices [119]. Because SmartThings lacks appropriate security measures, Fernandes et al. developed this attack against it [36].

5.1.3 Malicious. Ref. [94] presents a malignant SmartApp. HomeScan [75] [74] conducts analysis against malicious control points, malicious hub, and malicious smart-device attacks. Owing to design and implementation issues in applications, Ref. [113] also investigated in harmful applications. Soteris et al. described IoT devices that consider the local network as vulnerable to malicious software [31]. IOTGUARD [17] modeled harmful code that is added to an application or provides user access to a program that can lead to an unsafe situation. In [32] and [56], attacks occurred because a malicious application resulted in unexpected behavior. An application that modifies the status of connected devices in a specified manner to launch a malicious application was mentioned in [102].

5.1.4 Compromised Attack. A compromised SmartApp [94] is the outcome of approving permission requests without understanding the danger an application presents, and an application may also seek more privileges than required.

5.1.5 Local Network Attack. Ref. [109] considers malware in IoT devices from a local network. Additionally, a compromised system in the local network via vulnerable devices is studied in [43] [75]. In [61], the authors introduced a system that emphasizes access control mechanism to enhance security.

5.1.6 Internet Attack. Corentin and David considered Internet viruses in IoT devices [109]. An attack at the network level was defined in [104]. Eavesdropping, intercepting and changing control activities, and intercepting and changing administration activities are the three forms of network attacks researched in [75] [74].

5.1.7 Denial of Service (DOS) Attack. In [89], a DOS defined in which the attacker sends the target a high number of transactions to prevent the target from being available. Skider et al. presented a malicious application that terminates all active jobs on smart devices at a certain value [102].

5.1.8 Modification Attack. As mentioned by Qashlan et al. [89], an attacker may attempt to change or remove stored data for a specific person or device.

5.1.9 Cross-App Interference threats. Ten types of cross-app interference threats were identified by Chi et al. [21]. They are divided into two action-interference threats, four trigger-interference threats, and four condition-interference threats.

5.1.10 Impersonation Attack. Skider et al. measured Aegis's performance against impersonation attacks in which an attacker uses stolen code to open a smart lock by posing as a legitimate user after battery-monitoring software leaks the unlock code over SMS or an application that records voice commands and plays them back to pretend to be real users [102].

5.1.11 False-Data Injection. A smart home may contain a malicious application that uses falsified data to performed harmful actions on a smart-home device [102].

5.1.12 Side-Channel Attack. An installed smart home application with design flaws can undertake lawful but exposed side-channel actions that can be exploited by other applications in the system or an attacker [102].

5.2 Security Countermeasures

We classified studies based on their security purposes. Some focus on mitigating attacks and protecting smart homes, while others aim to detect malicious attacks or raise alerts. Table 5 presents the classification of countermeasures from related research.

In [101], policy negotiations and conflict resolutions were suggested to improve security. Aegis is a context-aware security system that monitors user behaviors in smart homes to identify malicious conduct [102]. Mahadewa et al. proposed a system to discover security flaws in the implementation of smart-home integration [74] [75]. In [34], an anomaly-detection engine was introduced to produce warnings regarding suspicious actions in a home environment. A reasoning module was proposed in [115] to identify user-critical scenarios and offer data for modules that monitor daily functions and lifestyle-pattern reasoning. HOMEGUARD [21] aims to detect cross-app interference threats in smart-home applications. Jose et al. [57] proposed a logic-based security method to detect

Table 5. The countermeasures purpose for proposed threat model.

Purpose	Studies	Threat model	Countermeasures
Detection	[21]	Action-interference threats, trigger-Interference threats, and condition-interference threats.	
	[102]	Impersonation, false data injection, side channel attack, DoS, and triggering a malicious app.	Context-aware framework [102].
	[74] [75]	Internet attack, local network attack and event spoofing.	
	[101]	Over privileged control, privilege abuse, privilege escalation, and transitive privilege.	Access control system.
Prevention	[67]	Message forgery, message replay, masquerade attack, device compromise, DoS threat, password guessing, and man in the middle attack (MIMA).	A secure session key-based unique addressing scheme (SSKUAI).
	[89]	Denial of Service and modification attacks.	A hierarchical defence mechanism.
	[31]	Malicious.	Enforcement systems.
	[32]	Privilege escalation and malicious.	Context-based permission system.
Mitigation	[94]	Malicious and compromised attack.	A risk-based permission model.
	[113]	Malicious.	Enforcement systems.
	[43]	Local network attack.	Access control mechanism.
Combination	[104]	Internet attack.	Security management provider.
	[119]	Over privileged control and event spoofing.	
	[36]		
	[56]	Privilege escalation and malicious.	Context-based permission system.
	[17]	Malicious.	Enforcement systems.
	[109]	Local network attack and internet attack.	Security policies.
	[61]	Unauthorized access.	Access control system.

intrusions in smart homes and provide alerts. IoT threats are recognized and countered by ContexIoT [56]. IOTGUARD [17], a dynamic policy-based enforcement system for the IoT, detects insecure device states and blocks them. Khan et al. [60] developed a context-based ontology that guides users to mitigate vulnerability risks. In [92], a multi-agent collaboration model was presented to detect threats in a smart-home network. Pillai et al. [88] developed an intrusion detection system to detect undesired actions in smart-home devices and alert users. In [52], authentication vulnerabilities caused by application-developer mistakes were reported. Zhang et al. observed a smart-home application to detect inappropriate behaviors and alert users via text messages [119]. Doan et al. proposed an RES-Hub using the OAuth 2.0 authentication and authorization architecture to ensure safe access and management over home services and devices while the cloud is down [33].

In [31], the HanGuard system was proposed to protect smart-home networks from mobile-application attacks. Sivaraman et al. recommended the use of software-defined technologies to protect IoT devices from unwanted network activity [104]. Ref. [89] used blockchain technology and edge computing to provide resistance against modifications and DoS attacks. Ref. [69] presented a platform that protects a home network by blocking traffic flow and devices that cause attacks. Guo et al. [45] proposed an authentication scheme to secure communication between the gateway and devices. In [58], a security framework that offers an integrity mechanism for preventing security risks by utilizing self-signing and access control approaches was introduced. A context-aware authentication framework was presented to secure communication and mitigate attacks [9]. Kumar et al. proposed a scheme to authenticate the communication between a user and smart home using a secure key session [67]. SERENIoT defends IoT devices from threats by blocking traffic that is different from the specifications [109]. In [7], enhanced security was achieved because data are kept at the network edge and hostile attacks have a lower chance of success. Threat-mitigation policies have been proposed to minimize risks [72] suggested threat mitigation policies. Ren et al. [96] developed a mobile-authentication system to reduce false rejection rates. Expat [113] safeguards appified smart-home systems from hazards posed by rogue or malfunctioning automation apps. To mitigate the affected devices, Goutam et al. established least-privilege policies [43]. An IPv4 address-resolution protocol (ARP) server was suggested as a security measure to counteract ARP spoofing attacks [4]. In [94], a proposed model for smart homes reduced the risk of overprivileged applications.

Table 5 provides an overview of the countermeasure against the proposed threat model. A secure session key-based unique addressing scheme [67], was proposed to monitor smart home IoT networks by altering the conventional IPv6 protocol. A hierarchical defense strategy was proposed [89] for resilience against modification and DoS attacks. Sivaraman et al. [104] introduced a security management provider entity that offers security and privacy to home IoT devices as a service. Kratos [101] proposed an access-control system that resolves conflicts between user requests to preserve smart home security. A context-based permission system, ContexIoT [56], overcomes the threat model by including data dependence in the context definition. In [94], a risk-based permission model that reduces malicious application attacks was presented. An enforcement system was proposed in [113] to prevent the installation of malicious applications. The IOTGUARD directly prevents dangerous and undesirable conditions in single, and multi-app contexts [17]. In [31], secure a smart-home network was secured by enforcing access control restrictions across user phones and IoT devices. Hesita deployed a least-privilege network strategy to reduce the danger of compromise in smart homes [43]. Thomasset et al. developed security policies for IoT devices to detect and block aberrant behaviors [109].

To improve the security of a smart home, access-control techniques must be included. The literature identifies five different access control techniques:

(1) Multi-user Access Control: Kratos [101] proposed a multi-user smart-home access-control system that addresses the diverse and conflicting demands of different users.
(2) Context-aware Access Control: The attribute-based access control (ABAC) model [66] determines the access control for devices and data in the smart-home environment [34]. ContexIoT [56] is a permission-based system that ensures the contextual integrity of IoT apps during operation. In [113], policies for fine-grained, contextual access control for smart-home platforms were proposed. The context-aware authentication framework introduced by Ashibani et al. [9] protects smart devices from unauthorized access by both anonymous and known users.
(3) Situation-aware Access Control: Demetriou et al. [31] gathered situational information via user-space applications to detect whether an authorized application was establishing a network connection with a target IoT device.
(4) Network Access Control: Distributed-access control networks were recommended by [89] to guard against unauthorized data access in smart-home systems that utilize ABAC. Hestia [43] is a default access control mechanism for devices in a smart-home network that is flexible to scale with changing smart-home environments and simple enough to be deployed at present. In [120], access-control policies for smart-home local networks that authenticate entities through data encryption and decryption were proposed. Mohammed et al. introduced static and dynamic access controls, both of which can be used to prevent or block malicious activities [4].
(5) Risk-based Access Control: People may perceive varying levels of danger as acceptable; thus, Tyche developed risk assessments of access control requests from applications by users [94].
(6) Security Role: In [101], five various roles in smart homes are indicated to understand user priority. By grouping applications into four categories—energy, health, security, and entertainment— a semantic-aware multilevel equivalence class-based policy first introduced in [81], reduces the cognitive strain on users. Each mobile device connected to the home

network has a role assigned by HanGuard, such as HAN user for accessing a specific home domain, admin roles for all domains, and guests for unregistered devices [31].

5.3 Knowledge Used for Countermeasures

Smart homes must be kept secure with the aid of defenses against attacks based on specific knowledge that has been gathered previously. In [72], countermeasures were applied based on the access policy, security/trust/threat levels, assessment policies, threat-mitigation policy, and contextual security information. IOTGUARD gathers application-specific data from its source code to enforce rules that prevent stop undesirable behaviors [17]. The router receives the runtime scenario from the user's phone and acts accordingly to enforce the policy [31]. User voiceprints were used in a dynamic-threshold technique to determine speaker ratings [96]. Expat [113] reviewed user-entered policies to ensure that they met user expectations. Using access-control rules, Sivaraman et al. developed a security management provider entity to secure IoT devices at the network level [104]. In [89], the rules and regulations upheld by blockchain miners and smart contracts were used to safeguard smart-home appliances. Devices in SPIN with security capabilities can prevent traffic from unreliable devices [69]. In [92], data gathered using the BDI model was used to achieve security. The rules define how the detecting device interacts with devices in the smart-home network to identify intrusions and undesirable behaviors [88]. SecFHome introduces an authentication mechanism to secure data after transferring the session keys [45]. Security dangers are reduced by defining the functions of each module in the suggested architecture [58]. Ashibani et al. proposed a context-aware authentication system for smart homes that uses the user's location, profile, calendar, and access-behavior patterns to enable access to home devices [9]. Secret keys and device identities were used as knowledge to secure communications over a smart-home network [67]. Sovereign leverages semantic names for resource identification, security implementation, and the definition of security rules [120]. Hestia implemented least-privilege policies to protect smart-home security [43]. In [4], network attacks are reduced by implementing various access-control measures. Physical-device operations are used as knowledge to assess its potential threats [94].

6 TESTBEDS AND EVALUATION

Testbeds and evaluations test and assess theories, models, and hypotheses, allowing researchers to verify if suggested concepts work in practical settings. The following sections discuss the evaluation procedures and the factors used in these strategies.

6.1 Evaluation Approaches

This section focuses on the evaluation strategies employed in selected studies. A summary of the evaluation methods is present in Table 6.

To validate the performance of an autonomic security manager, Lin et al. [72] proposed a case study of a conference room with a large number of events. In [101], a case study was conducted to evaluate the effectiveness and overhead of Kratos. The effectiveness and feasibility of Aegis+ were tested by building a smart-home testbed [102]. Case studies were implemented in [74] to find the security issues. Three case studies were conducted to assess the applicability of the risk-based approach [94]. In [3], a case study was conducted to evaluate the proposed model and provide a proof of concept for compromised devices.

An experiment was conducted in a laboratory to highlight the vulnerability of smart-home devices [103]. Lalanda et al. simulated the ICasa platform [68] to measure the complexity of services, timely execution, and adaptation cost. In [33], a demonstration was presented as a proof of concept for SmartThings devices. The authors of [21] developed experiments to prove that HOMEGUARD can detect cross-app interference threats. Static and run time analyses were used in [81] to detect

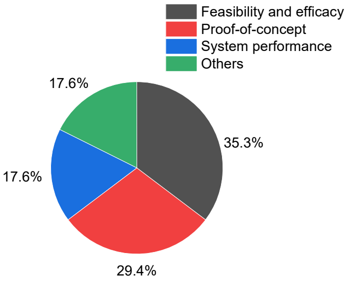


Fig. 8. Evaluation goals.

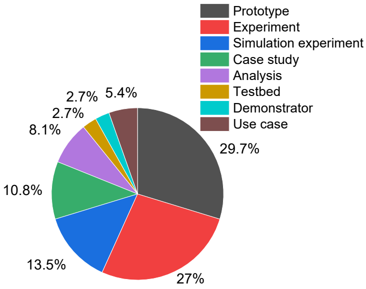


Fig. 9. Evaluation settings overview.

conflicts in a smart-home system. The CASAS dataset [27] was used for the runtime analysis, which lasted 34 days. The experiment developed by Jose et al. [57] was used to observe the user behavior at different access points in a studio apartment over a 30-day period. Ref. [107] evaluated the system performance by designing prototypes and proofs. Amadeo et al. deployed a testbed as proof of concept for the proposed framework [7]. A simulation was conducted to measure the overhead performance of IOTGUARD [17]. Demetriou et al. [31] designed a prototype and conducted an experiment to evaluate the HanGuard performance. In [96], an experiment was conducted to test the performance of voice-print verification. The effectiveness of Expat [113] was evaluated using its own testbed and dataset. An experiment was presented [75] to identify vulnerabilities by extracting the execution log and Wi-Fi traffic from the implementation of smart-home systems. As a proof-of-concept, Fernandes et al. [36] developed an empirical analysis to exploit a flawed design by building the SmartApps dataset and conducting a survey with 22 participants. As a proof of concept, SERENIoT simulated network compatibility across numerous days on Amazon AWS [109]. An attribute-smart contract-based edge scheme [89] was simulated to demonstrate its feasibility and efficiency in authenticating smart-home users and devices. As a proof of concept, Rafferty et al. proposed a use case illustrating the coordination of threat-response decisions between operational availability and security risk agents [92]. SceFHome was simulated in [45] to calculate communication and processing expenses. Security analysis proved the security of the proposed scheme [67]. Hu et al. conducted proof-of-concept studies to assess the security of smart home assistant applications [52].

A prototype was created to test the privacy, security, and performance of Sovereign [120] and evaluate the network performance of Hestia [43]. The HoMonit prototype was created by Zhang et al. to test the efficacy and efficiency of the suggested system [119]. In [4], the software defined networking (SDN)-based framework was prototyped by countering malicious network monitoring and ARP spoofing. Contexlot developed a prototype [56] on a dataset compromising 25 SmartApps for 22 attacks. The suggested network-centric method was prototyped to demonstrate its efficacy in protecting multiple smart-home devices deployed in the laboratory [104]. Ding et al. [32] implemented a prototype of over 185 official SmartThings applications. A prototype was developed to evaluate the efficacy of the network-intrusion detection system in a smart-home network [88]. In [9], a prototype was designed for a context-based authentication system to evaluate its flexibility. Lastdrager et al. implemented a prototype for the SPIN platform in their laboratory [69].

Figure 8 shows that the most common goal was proof of concept (10 publications), followed by system performance (eight publications). Feasibility, efficacy, and other goals were combined to form 12 studies. The terminology for evaluation methods varied significantly. Figure 9 presents an overview of the eight methods used. The most stated evaluation type was "prototype" with 11 publications. The second method was "Experiment." This was followed by "simulation experiment" and "case study" with five and four papers, respectively. Notably, the "analysis" method was used in three papers. However, the least-popular method were "testbed," "demonstrator," and "use case."

Table 6. Evaluation strategy.

Reference	Evaluation method	Evaluation goal	Location	Duration	Data type
[29]	Case study.	To assess the relevance, advantages, and limitations.			
[72]	Case study.	System performance for a large-scale smart space.			
[101]	Case study.	Effectiveness and performance overhead.			User demands.
[102]	Testbed.	The effectiveness and feasibility of system.		15 days.	Benign daily activities dataset (85,000 events).
[74]	Case studies.	To find security issues.			
[68]	Simulation experiment.	The study evaluates the fog-level platform's ability to manage context module complexity, the execution time for context modules, especially with conflict resolution, and the cost of dynamic context adaptability managed by the autonomous manager, in collaboration with Orange Labs.			
[33]	Demonstrator.	Proof-of-concept.			
[21]	Experiment.	Proof of concept.			Configuration information.
[81]	Static and run time analysis.	To measure the likelihood of true conflict between applications, number of runtime conflict, conflict resolution capability, and a level of conflict for each app.		34 days (runtime analysis).	Dependency information.
[57]	Experiment.	To track user activity at different access points.	Studio apartment.	A month.	Logical sensing parameters.
[107]	Prototype.	System performance.			
[7]	Experimental testbed.	Proof-of-concept.			
[56]	Prototype.	Proof-of-concept.			Control and data flow attributes of the app, and runtime values.
[32]	Prototype experiment.	Proof-of-concept.			Inter-app trigger-action interactions and physical channel information.
[17]	Simulation experiment.	System overhead performance.			Application's information.
[31]	Experiment.	System performance.			Situation information.
[96]	Experiment.	System performance.			Utterance from speaker.
[113]	Experiment.	System performance.			Rules and policies.
[75]	Experiment.	To identify the vulnerabilities.			Execution log and Wi-Fi traffic.
[36]	Empirical analysis.	Proof-of-concept.			
[3]	Case study.	Proof-of-concept.			
[104]	Prototype.	To demonstrate its effectiveness in safeguarding several smart home gadgets.	Lab.		
[89]	Simulation experiment.	Feasibility and efficiency of the system.			
[69]	Prototype.		Lab.		
[92]	Use case.	Proof-of-concept.			
[88]	Prototype.	System performance.			
[45]	Simulation.	Communication costs and computation costs performance.			
[9]	Prototype.	To show the flexibility of the security framework.			
[67]	Security analysis.				
[52]	Experiment.	Proof-of-concept.			
[120]	Prototype.	To assesses the privacy and security, and performance of system.			
[43]	Prototype.	Network performance.			
[109]	Simulation.	Proof-of-concept.		From 1 hour to multiple days.	
[119]	Prototype.	To evaluate the effectiveness and efficiency of the proposed system.			Wireless traffic.
[4]	Prototype.	To show the feasibility of the proposed framework.			
[103]	Experiment.		Lab.		
[94]	3 Case studies.				

6.2 Evaluation Factors

A variety of criteria should be considered when evaluating methods. The six variables affecting the evaluation are discussed in this subsection.

6.2.1 Devices. Smart home environments are equipped with an assortment of smart devices that provide comfort to the occupants of the home. By the end of this decade, the number of smart devices in our daily lives is expected to be in the order of billions [100]. Accordingly, access control for multiple-devices is considered a daunting challenge in smart homes. Table 7 presents an overview of the devices used in the smart-home context. For example, 17 different devices were considered in Ref. [101]. A real-world smart environment can be created using 14 distinct types of commercially available sensors, devices, and controllers [102]. In [31], four devices were selected (three actuators and one sensor) for real-world testing. The smart home in Ref. [113] was equipped with 18 devices. In [36], 132 devices that were compatible with SmartThings, called device handlers, were downloaded. The authors used the Philips Hue light bulb and Nest smoke alarm to demonstrate the value of having IoT protection provided by the security-management provider as a value-add service [104]. Five types of devices were used in the implementation of smart-home environments [9]. In [119], data was collected from seven ZigBee devices and four Z-Wave devices. In [109], a smart-home context was simulated using 53 various devices. Additionally, 29 devices were used to simulate the smart home [17].

Table 7. Smart home devices, users and applications.

Reference	Devices Number	Devices list	Users Number	Applications Number
[101]	17	Smart home hub, smart light, smart lock, smart camera, smart thermostat, motion sensor, door sensor, and temperature sensor.	43	10
[102]	6-24	Sensors, controllers (smartphone, tablet, and voice-controlled smart), and devices (smart light, smart lock, etc.).	20	
[31]	4	WeMo Switch and WeMo Motion, the WeMo in.sight.AC1, and My N3rd		55
[113]	18			15
[36]	132			499
[104]	2	Philips Hue light-bulb and the Nest smoke-alarm.		
[9]	5	Single-board computer, wireless router, smart switch, smart light hub, smart bulb		
[119]	11	ZigBee devices and Z-Wave devices		30
[109]	53			
[117]			7	
[17]	29			65

6.2.2 Platforms. Various types of platforms are used in smart homes. For example, the Samsung SmartThings platform [44] is implemented in [101], [33], [56], [119], [94], [36], [32] and [21]. It has the largest market share in consumer IoT and supports the greatest number of open-source apps and smart home devices. Google Home was used in [102]. Moreover, in [102], the Samsung SmartThings platform was selected for the purpose of developing a single-platform smart-home environment in which all devices share the same access point. However, in multi-platform smart-home systems, where the gadgets for smart homes are deployed as separate entities and no common access point is considered during installation, Amazon Alexa, Philips Hue, LIFX smart bulbs, and Samsung SmartThings were used. The ICasa platform [87] [68] offers a suitable model for development, as well as a number of tools for interfacing with heterogeneous devices, gathering and displaying contextual data, and enabling the dynamic deployment of components and applications. The proposed multi-layer cloud platform [107] for IoT-based smart homes consisted of a public cloud provided by Amazon EC2 and two private smart-home cloud platforms supported by DGUT and Canbo. The ICN-iSapiens platform [7] provided real-time services while obscuring the diversity of the IoT devices. Celik et al. evaluated their system using the SmartThings platform and IFTTT [55] trigger-action platform [17]. The authors of [113] integrated their prototype into the OpenHAB smart-home platform [84], which was used to the automate the interactions between smart devices. An open-source measurement platform called SPIN [50] builds a dynamic and user-friendly data model for IoT devices in the home network used in [69].

6.2.3 Applications. Amit et el. [101] installed ten different official SmartThings applications that control other devices. In [31], home-area network IoT devices were connected with WiFi/Internet using 55 different Android applications. Fifteen automated applications were installed on a smart-home platform [113]. In [36], 499 SmartApps were downloaded from the SmartThings app store, and a thorough examination was performed. From the SmartThings public GitHub repository [105], 30 SmartApps were chosen, where 20 SmartApps worked with ZigBee devices and ten connected Z-Wave devices [119]. In [17], 35 SmartThings and 30 IFTTT market-vetted applications (65 applications) were used to evaluate smart homes. Table 7 summarizes the number of devices used in smart-home environments in the literature.

6.2.4 Protocols. Communication protocols (ZigBee and Wi-Fi) used by Philips Hue, LIFX, and Chromecast were analyzed to extract an end-to-end specification for detecting security vulnerabilities [74]. To link the devices to the hub, Tam et al. utilized message queuing telemetry transport [13]) via TCP/IP; however, Bluetooth, ZigBee, and Z-Wave were used as the communication protocols, and the OAuth 2.0 authorization protocol [79] [47] was used to authenticate SmartApps APIs to ensure that the Web App had access to the devices it needed [33]. In [7], the CCN-Lite software [78] used a simple CCNx/named data networking (NDNx) protocol, which was chosen to facilitate ICN connections between different boards in smart homes and used IEEE 802.11g to communicate wirelessly with devices. In [36], the OAuth protocol was used by the client-side Web IDE and SmartThings backends to analyze attacks. Kumar et al. [67] proposed a modification of the IPv6 protocol and used the Diffie Hellman key-exchange protocol to secure communication in a smart-home network. In [120], a lightweight NDN [26] protocol that safeguards data by securing device-to-device communication was implemented. Wei et al. [119] detect misbehaving SmartApps by checking the wireless (Z-Wave and ZigBee) traffic between a SmartThings hub and devices. In [103], the issues of the universal plug-n-play protocol used by devices to communicate with home gateways were discussed.

6.2.5 Events. An event is anything that occurs in the smart-home system that alters its state. Aegis+ [102] notifies users of any malicious activity in real time by comparing a dataset comprising over 85,000 events collected from a user's daily activities with 24 different datasets for a total of over 15,000 events. In [72], a sequence of 160 events was used to validate the investigation. Yunhan et al. [56] evaluated the system on a dataset compromising 283 SmartApps by injecting device events to trigger 916 events handling logic. In [36], sensitive information was not adequately protected by the SmartThings event subsystem, which devices use to interact asynchronously with SmartApps through events. For each IFTTT rule to be mapped to an IoT app, Celik et al. [17] extract the events (86, 30) and actions (78, 30) from SmartThings applications and IFTTT trigger-action applets, respectively. Ref. [119] proved that the lack of event integrity protection in the SmartThings architecture leads to event-spoofing attacks.

6.2.6 Users. Users of smart homes often share installed smart-home devices in a multi-user scenario, as a typical house consists of multiple people (see Table 7). In [101] smart-home data were collected from 43 real-life users. In [102], information was acquired from 20 users, and various users simultaneously conducted daily tasks. In [117], a smart-home was prototyped with seven households, including couples, roommates, and families with children of various ages. The needs and preferences of smart home users are defined in Refs. [117], [49], [116], and [101] to include a fine-grained access-control system to prevent overprivileged challenges, a role-based access control system to restrict access to devices and applications in a home setting, and location and time-based access control for transient users in a communal setting. Automation rules aim to reconcile competing requests, and users accept per-device roles for private rooms in a shared environment. Aegis+ [102] analyzed user activity using a context pattern to identify concurrent operations carried out by several people and devices in a smart-home system. The proposed bathroom-monitoring system [115] provided the users' daily activities, personal-care routines, and lifestyle habits as knowledge for the reasoning module. Ref. [3], explained the issues stemming from the disclosure of behavioral patterns, such as the exchange of private information, insurance-related fraud, and burglary.

7 LESSONS LEARNED

As vast amounts of data are created [25], securing communication between the cyber and physical worlds is challenging. This requires addressing user awareness, device security, network risks, and malicious programs. In this section, we explain why current strategies are inadequate and present the following observations based on peer-reviewed articles.

Users. The CSGO has been suggested to help users perform security guidelines automatically [60]. Furthermore, access-control systems that define user roles and privileges depending on smart-home conditions are being studied to avoid conflicting requests between users. Most current research is directed at creating whole processes in smart-home automation, even though adding the user to the loop of operations would make the user more aware of important faults in the system.

Devices. SDN [65] was proposed [31] to protect IoT devices. Qashlan et al. proposed a decentralization authentication scheme to secure IoT devices [89]. In [58], a security framework for smart devices was proposed to maintain the integrity of the module codes. A risk-based permission model was proposed to classify device operations and mitigate risks [94]. However, these methods were limited against specific types of attacks.

Networking. In [104], a range of services was provided at the network level, such as security, taking advantage of SDN technology. A distributed system for protecting home networks from hacked devices was presented [69]. A multi-agent collaboration model was used to represent each entity in a smart-home network as an agent to collaboratively achieve security [92]. A modification of the IPv6 protocol to secure smart-home IoT networks proposed in [67]. In [120], the NDN model was used to secure device-to-device communication. Access-control policies were enforced to reduce communication with networks [43]. Both [109] and [119] proposed detection systems to monitor traffic in a smart-home network. A network access-control framework was enforced at the network level of a smart home [4]. As a smart-home network serves as the Internet's primary point of contact with the outside world, numerous security threats can be launched against it. Studies must be conducted to combat such attacks, which are becoming increasingly frequent.

Applications. Side-channel inference [119] monitors the activities of SmartApps to discover misbehavior. In [21], cross-app interference threats were recognized using SMT, which treated the problem as an automated theorem problem. In [113], an SMT solver was used to verify the satisfiability of the policy. Dependency detection and resolution at installation and runtime were implemented to check for conflicts across applications [81]. Patching was used in context-based permission systems [56] and policy-based enforcement systems [17], which increased the performance overhead. These methods can be integrated to further improve defense accuracy.

8 RESEARCH CHALLENGES AND DIRECTIONS

This section outlines four research challenges in smart-home systems identified in our review (See Table 8): integrating self-adaptation in smart homes, on-edge data processing, lack of adequate testbeds and evaluation, and beyond-detection-method techniques. These areas require further exploration and the development of novel techniques and solutions.

Table 8. A summary of research directions and current challenges.

Research direction	Current challenges	Possible Solutions
Self-adaptive security: Self-adaptive security encompasses a proactive and dynamic approach to cybersecurity, enabling systems to autonomously adjust and reconfigure their security mechanisms and countermeasures in real-time, based on evolving conditions, emerging threats, and fluctuating risk levels within the operational environment [20].	The deployment of unified security solutions is made more difficult by the heterogeneous nature of the linked devices, each of which has different capabilities, protocols, and vulnerabilities.	Monitor-Analyze-Plan-Execute-Knowledge (MAPE-k) [15] method and multi-agent mechanism [76].
On edge security: In this approach, edge devices are equipped with computational, storage, and communication capabilities traditionally associated with cloud servers [83].	Transferring data to cloud-based systems can introduce significant latency, as the process of transmitting and processing the information over the network can be time-consuming. Moreover, relying solely on cloud infrastructure for data handling raises concerns regarding data security and privacy.	Edge processing.
Testbeds and evaluation: It processes serve as benchmarking tools, providing a systematic and objective means to quantify the uniqueness, effectiveness, and practical applicability of the research contributions.	Using only theoretical models or simulations might make it difficult to fully understand the complexities and details of real-world applications.	Real-world evaluation approach.
Cyber-physical anomaly detection: The ability to identify when anything goes wrong or when an unusual event occurs [93].	threat explanations remains an understudied and largely unexplored area.	Intelligence gathering.

8.1 Self-adaptive Security

Smart devices are heterogeneous and each has a different set of capabilities in terms of sensing and actuation. Smart spaces may be hacked, exposing privacy and security or rendering the entire area a hostile environment in which ordinary tasks are impossible. Therefore, securing smart spaces can be challenging because of device heterogeneity, continuous changes in context, and limited device resources. Self-adaptive security is crucial for smart-home systems because it offers real-time threat detection, flexibility in response to changing threats, resource optimization, and a smooth user experience. Ensuring that smart homes are robust in against a constantly shifting threat landscape helps safeguard user security and privacy. Self-adaptive security measures are becoming increasingly important as smart-home technology develops. To address this problem, smart devices should be dynamically configured to perform the corresponding task. The monitor-analyze-plan-execute-knowledge (MAPE-k) [15] method and multi-agent mechanism [76] show future directions for further research to tackle this challenge. These techniques can monitor smart-home networks and devices continually for any unusual activity or security breaches while automating security decisions and actions, reducing the reliance on human decision-makers who are frequently prone to error. Moreover, the MAPE-K framework has been extended to facilitate human-machine teaming [22], thereby enabling effective collaboration and communication between automated systems and human operators. A notable example demonstrating the incorporation of the human-machine teaming concept can be observed in the domain of unmanned aerial systems [23] [18], where autonomous drones and human controllers work in tandem, leveraging their respective strengths and capabilities. In addition, ontologies (such as W3C SSN [24], W3C BOT [95], and W3C IoT-Lite [14]) were used to model contextual information in the smart-home environment.

8.2 On-Edge Security

The computation and storage of data produced in a smart-home are saved on cloud backend servers. The large volume of traffic generated by the widespread use of mobile videos and online social-media applications has led to the big-data concept [11]. Thus, managing big-data-driven networks in cloud environments is critical [118]. Consequently, edge computing or fog computing [83] is an emerging technology in which edge devices provide the capabilities of a cloud server to perform functions including communication, storage, and control. Using edge computing is a possible direction for ensuring the security and safety of the cyber-physical system without requiring cloud services. Edge security for smart-home systems is of paramount importance, and represents a critical area for future work to handle the increasing amount of data generated and processed in smart homes. Similarly, it enables devices to continue operating autonomously in the case of network failures or disturbances. Edge systems are crucial for maintaining low latency in real-time applications, such as smart lighting or home automation, while safeguarding the integrity and confidentiality of data. As the adoption of smart-home technology continues to grow, further research and development in edge security is vital for addressing the unique challenges and opportunities presented by this rapidly evolving field. Smart-home systems can decrease the quantity of sensitive information communicated to remote servers or third-party services by implementing certain data-processing and decision-making functions closer to the source [41]. The potential security benefits offered by edge-computing capabilities must be recognized. By performing filtering and processing operations on data collected from nearby sources, edge devices can effectively eliminate noise, irrelevant information, and outliers [99]. Consequently, only relevant high-quality data are transmitted to the cloud after edge processing. Enhanced data quality optimizes the performance of analytical models and enables the development of smart-home products and services tailored to user needs and contexts.

8.3 Testbeds and Evaluations

Researchers can compare their suggested solutions or methodologies with those that already exist owing to testbeds and evaluations. The uniqueness and efficacy of the research are evaluated using this benchmarking. Moreover, insights into the generalizability of the research can be gained through test beds and evaluation results. Researchers can verify whether their findings hold true in various settings, populations, or circumstances. As discussed in Section 6, most of the reviewed studies on smart-home security utilized prototypes, experiments, case studies, analyses, and simulation experiments to evaluate their approaches. However, testbeds, demonstrators, and use cases also played a minor role in evaluations. A real-world evaluation is required to achieve the most realistic results. This is one of the most challenging issues in this field. Therefore, smart-home security techniques must be implemented in real scenarios. For benchmarking purposes, a real-world IoT testbed should be created using Arduino and Raspberry Pi sensor nodes. Each sensor node has several different sensors and computational capabilities. In addition, to validate the system performance, we conducted experiments using real-life datasets. Different types of datasets exist based on their usage, such as IoT smart-home devices (YourThings dataset [6] and CASAS dataset [27]), smart home applications ([56] [119] [81] [36] [114]), and the IoT network intrusion dataset [54]. Therefore, collecting datasets based on common cases of security use is important. The number of data-sets is determined by the quality of each data-set and repeatability of the results.

8.4 Cyber-physical Anomaly Detection

Anomaly-detection techniques are used to signal to users that something occurred incorrectly in a smart home [93]. The security, privacy, and safety of smart-home systems depend heavily on cyber-physical anomaly detection. To address the changing threats and vulnerabilities, efficient anomaly-detection systems must be created and implemented as smart-home technology continues to expand and become more complicated. This is a viable area for further research to support the development and use of smart-home systems. However, to the best of our knowledge, a threat explanation has not yet been investigated for cyber-physical security in smart-home systems. As a result, a unique challenge arises in discovering and exploring incidents that take advantage of the entire gamut of smart-home contexts. To accomplish this, intelligence gathering functionality is a promising topic for further research. This may provide the locations of suspected cyber-physical threats by collecting more evidence and information to detect anomalous incidents. In such cases, the system must capture infrastructure knowledge and capabilities to improve the smart home's understanding of potential threats. Because of the relevance of detecting anomalous incidents in real time based on contextual information, more effort should be given to this field. The security, privacy, and safety of smart-home systems depend heavily on cyber-physical anomaly detection. To address the changing threats and vulnerabilities, efficient anomaly-detection systems must be created and implemented as smart-home technology continues to expand and become more complicated. This is a viable area for further research to support the development and use of smart-home systems.

9 CONCLUSIONS

CPSs play a crucial role in smartness and digitization by integrating the cyber and physical worlds. This has resulted in the emergence of numerous applications in various fields. For example, smart homes are the primary domain of CPS, consisting of many smart devices and applications in the interest of providing services to maintain household comfort. Smart-home environments face many challenges in terms of their functional and nonfunctional requirements. Numerous solutions using artificial intelligence mechanisms have been proposed. These methods have limitations, including concentrating on a single issue rather than providing a comprehensive solution or suggesting remedies that need to be updated. Therefore, a complete solution that develops with the evolving vulnerabilities of smart homes is required.

In this review, we analyzed and evaluated the knowledge employed in smart homes to comprehend and analyze their experiences. We proposed a taxonomy that defines the classification of decision-making locations. We presented the main countermeasures against attacks and threats in smart homes. We also discussed the evaluations of smart homes from the past to the present. We have reviewed the security of smart homes on different platforms and applications. In addition, we analyzed various aspects of the challenges of smart homes and how current solutions overcome these limitations. Finally, we examined four research gaps related to smart homes from a knowledge-based perspective that requires further research.

REFERENCES

[1] Gregory D Abowd, Anind K Dey, Peter J Brown, Nigel Davies, Mark Smith, and Pete Steggles. 1999. Towards a better understanding of context and context-awareness. In *International symposium on handheld and ubiquitous computing*. Springer, 304–307.

[2] Md Ahmad, Mohd Abdul Ahad, M Afshar Alam, Farheen Siddiqui, Gabriella Casalino, et al. 2021. Cyber-Physical Systems and Smart Cities in India: Opportunities, Issues, and Challenges. *Sensors* 21, 22 (2021), 7714.

[3] Nikolay Akatyev and Joshua I James. 2019. Evidence identification in IoT networks based on threat assessment. *Future Generation Computer Systems* 93 (2019), 814–821.

[4] Mohammed Al-Shaboti, Ian Welch, Aaron Chen, and Muhammed Adeel Mahmood. 2018. Towards secure smart home IoT: Manufacturer and user network access control framework. In *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*. IEEE, 892–899.

[5] Mussab Alaa, Aws Alaa Zaidan, Bilal Bahaa Zaidan, Mohammed Talal, and Miss Laiha Mat Kiah. 2017. A review of smart home applications based on Internet of Things. *Journal of Network and Computer Applications* 97 (2017), 48–65.

[6] Omar Alrawi, Chaz Lever, Manos Antonakakis, and Fabian Monrose. 2019. Sok: Security evaluation of home-based iot deployments. In *2019 IEEE symposium on security and privacy (sp)*. IEEE, 1362–1380.

[7] Marica Amadeo, Antonella Molinaro, Stefano Yuri Paratore, Albino Altomare, Andrea Giordano, and Carlo Mastroianni. 2017. A Cloud of Things framework for smart home services based on Information Centric Networking. In

2017 IEEE 14th International Conference on Networking, Sensing and Control (ICNSC). IEEE, 245–250.

[8] Adeeb Mansoor Ansari, Mohammed Nazir, and Khurram Mustafa. 2024. Smart Homes App Vulnerabilities, Threats, and Solutions: A Systematic Literature Review. *Journal of Network and Systems Management* 32, 2 (2024), 1–62.

[9] Yosef Ashibani, Dylan Kauling, and Qusay H Mahmoud. 2017. A context-aware authentication framework for smart homes. In *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*. IEEE, 1–5.

[10] Yosef Ashibani and Qusay H Mahmoud. 2017. Cyber physical systems security: Analysis, challenges and solutions. *Computers & Security* 68 (2017), 81–97.

[11] Rachad Atat, Lingjia Liu, Jinsong Wu, Guangyu Li, Chunxuan Ye, and Yi Yang. 2018. Big data meet cyber-physical systems: A panoramic survey. *IEEE Access* 6 (2018), 73603–73636.

[12] Bharathan Balaji, Arka Bhattacharya, Gabriel Fierro, Jingkun Gao, Joshua Gluck, Dezhi Hong, Aslak Johansen, Jason Koh, Joern Ploennigs, Yuvraj Agarwal, et al. 2016. Brick: Towards a unified metadata schema for buildings. In *Proceedings of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments*. 41–50.

[13] Andrew Banks and Rahul Gupta. 2014. MQTT Version 3.1. 1. *OASIS standard* 29 (2014), 89.

[14] Maria Bermudez-Edo, Tarek Elsaleh, Payam Barnaghi, and Kerry Taylor. 2015. Iot-lite ontology. *W3C Member Submission, W3C, November* (2015).

[15] Yuriy Brun, Giovanna Di Marzo Serugendo, Cristina Gacek, Holger Giese, Holger Kienle, Marin Litoiu, Hausi Müller, Mauro Pezzè, and Mary Shaw. 2009. Engineering self-adaptive systems through feedback loops. In *Software engineering for self-adaptive systems*. Springer, 48–70.

[16] David Buil-Gil, Steven Kemp, Stefanie Kuenzel, Lynne Coventry, Sameh Zakhary, Daniel Tilley, and James Nicholson. 2023. The digital harms of smart home devices: A systematic literature review. *Computers in Human Behavior* (2023), 107770.

[17] Z Berkay Celik, Gang Tan, and Patrick D McDaniel. 2019. IoTGuard: Dynamic Enforcement of Security and Safety Policy in Commodity IoT.. In *NDSS*.

[18] Theodore Chambers, Michael Vierhauser, Ankit Agrawal, Michael Murphy, Jason Matthew Brauer, Salil Purandare, Myra B Cohen, and Jane Cleland-Huang. 2024. HIFuzz: Human Interaction Fuzzing for Small Unmanned Aerial Vehicles. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–14.

[19] Marie Chan, Daniel Estève, Christophe Escriba, and Eric Campo. 2008. A review of smart homes—Present state and future challenges. *Computer methods and programs in biomedicine* 91, 1 (2008), 55–81.

[20] Salim Chehida, Eric Rutten, Guillaume Giraud, and Stéphane Mocanu. 2024. A model-based approach for self-adaptive security in CPS: Application to smart grids. *Journal of Systems Architecture* 150 (2024), 103118.

[21] Haotian Chi, Qiang Zeng, Xiaojiang Du, and Jiaping Yu. 2020. Cross-app interference threats in smart homes: Categorization, detection and handling. In *2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, 411–423.

[22] Jane Cleland-Huang, Ankit Agrawal, Michael Vierhauser, Michael Murphy, and Mike Prieto. 2022. Extending MAPE-K to support human-machine teaming. In *Proceedings of the 17th Symposium on Software Engineering for Adaptive and Self-Managing Systems*. 120–131.

[23] Jane Cleland-Huang, Theodore Chambers, Sebastian Zudaire, Muhammed Tawfiq Chowdhury, Ankit Agrawal, and Michael Vierhauser. 2024. Human–machine Teaming with Small Unmanned Aerial Systems in a MAPE-K Environment. *ACM Transactions on Autonomous and Adaptive Systems* 19, 1 (2024), 1–35.

[24] Michael Compton, Payam Barnaghi, Luis Bermudez, Raul Garcia-Castro, Oscar Corcho, Simon Cox, John Graybeal, Manfred Hauswirth, Cory Henson, Arthur Herzog, et al. 2012. The SSN ontology of the W3C semantic sensor network incubator group. *Journal of Web Semantics* 17 (2012), 25–32.

[25] Marco Conti, Sajal K Das, Chatschik Bisdikian, Mohan Kumar, Lionel M Ni, Andrea Passarella, George Roussos, Gerhard Tröster, Gene Tsudik, and Franco Zambonelli. 2012. Looking ahead in pervasive computing: Challenges and opportunities in the era of cyber–physical convergence. *Pervasive and mobile computing* 8, 1 (2012), 2–21.

[26] NDN Specification Contributors. 2022. *NDN Packet Format Specification version 0.3*. Retrieved 2022 August 2022 from https://named-data.net/doc/NDN-packet-spec/current/changelog.html#version-0-3

[27] Diane J Cook and Maureen Schmitter-Edgecombe. 2009. Assessing the quality of activities in a smart environment. *Methods of information in medicine* 48, 05 (2009), 480–485.

[28] Jessamyn Dahmen, Diane J Cook, Xiaobo Wang, and Wang Honglei. 2017. Smart secure homes: a survey of smart home technologies that sense, assess, and respond to security threats. *Journal of reliable intelligent environments* 3, 2 (2017), 83–98.

[29] Marcos Paulo de Oliveira Camargo, Gabriel dos Santos Pereira, Daniel Almeida, Leandro Apolinario Bento, William Fernande Dorante, and Frank Jose Affonso. 2024. Ra4self-cps: a reference architecture for self-adaptive cyber-physical systems. *IEEE Latin America Transactions* 22, 2 (2024), 113–125.

[30] Stefan Decker, Sergey Melnik, Frank Van Harmelen, Dieter Fensel, Michel Klein, Jeen Broekstra, Michael Erdmann, and Ian Horrocks. 2000. The semantic web: The roles of XML and RDF. *IEEE Internet computing* 4, 5 (2000), 63–73.

[31] Soteris Demetriou, Nan Zhang, Yeonjoon Lee, XiaoFeng Wang, Carl A Gunter, Xiaoyong Zhou, and Michael Grace. 2017. HanGuard: SDN-driven protection of smart home WiFi devices from malicious mobile apps. In *Proceedings of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. 122–133.

[32] Wenbo Ding and Hongxin Hu. 2018. On the safety of iot device physical interaction control. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. 832–846.

[33] Tam Thanh Doan, Reihaneh Safavi-Naini, Shuai Li, Sepideh Avizheh, and Philip WL Fong. 2018. Towards a resilient smart home. In *Proceedings of the 2018 workshop on IoT security and privacy*. 15–21.

[34] Sofia Dutta, Sai Sree Laya Chukkapalli, Madhura Sulgekar, Swathi Krithivasan, Prajit Kumar Das, and Anupam Joshi. 2020. Context sensitive access control in smart home environments. In *2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*. IEEE, 35–41.

[35] Abdessalam Elhabbash, Yehia Elkhatib, Vatsala Nundloll, Vicent Sanz Marco, and Gordon S Blair. 2024. Principled and automated system of systems composition using an ontological architecture. *Future Generation Computer Systems* 157 (2024), 499–515.

[36] Earlence Fernandes, Jaeyeon Jung, and Atul Prakash. 2016. Security analysis of emerging smart home applications. In *2016 IEEE symposium on security and privacy (SP)*. IEEE, 636–654.

[37] Gabe Fierro. 2024. *Brick*. https://brickschema.org/

[38] Brian Frank. 2023. *Haystaack*. https://brickschema.org/

[39] Shivani Gaba, Ishan Budhiraja, Vimal Kumar, Sheshikala Martha, Jebreel Khurmi, Akansha Singh, Krishna Kant Singh, Sameh S Askar, and Mohamed Abouhawwash. 2024. A systematic analysis of enhancing cyber security using deep learning for cyber physical systems. *IEEE Access* (2024).

[40] Ilche Georgievski and Marco Aiello. 2016. Automated planning for ubiquitous computing. *ACM Computing Surveys (CSUR)* 49, 4 (2016), 1–46.

[41] Fatemeh Golpayegani, Nanxi Chen, Nima Afraz, Eric Gyamfi, Abdollah Malekjafarian, Dominik Schäfer, and Christian Krupitzer. 2024. Adaptation in Edge Computing: A review on design principles and research challenges. *ACM Transactions on Autonomous and Adaptive Systems* (2024).

[42] Kai Gong, Jianlin Yang, Xu Wang, Chuanwen Jiang, Zhan Xiong, Ming Zhang, Mingxing Guo, Ran Lv, Su Wang, and Shenxi Zhang. 2022. Comprehensive review of modeling, structure, and integration techniques of smart buildings in the cyber-physical-social system. *Frontiers in Energy* (2022), 1–21.

[43] Sanket Goutam, William Enck, and Bradley Reaves. 2019. Hestia: simple least privilege network policies for smart homes. In *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks*. 215–220.

[44] Rachel Gunter. 2017. *Making Sense of Samsung's SmartThings Initiative*. https://marketrealist.com/2017/12/making-sense-samsungs-smartthingsinitiative

[45] Yimin Guo, Zhenfeng Zhang, and Yajun Guo. 2022. SecFHome: Secure remote authentication in fog-enabled smart home environment. *Computer Networks* 207 (2022), 108818.

[46] Hadi Habibzadeh, Brian H Nussbaum, Fazel Anjomshoa, Burak Kantarci, and Tolga Soyata. 2019. A survey on cybersecurity, data privacy, and policy issues in cyber-physical system deployments in smart cities. *Sustainable Cities and Society* 50 (2019), 101660.

[47] Dick Hardt. 2012. *The OAuth 2.0 authorization framework*. Retrieved 3 August 2022 from https://tools.ietf.org/html/rfc6749

[48] Abhishek Hazra, Pradeep Rana, Mainak Adhikari, and Tarachand Amgoth. 2023. Fog computing for next-generation internet of things: fundamental, state-of-the-art and research challenges. *Computer Science Review* 48 (2023), 100549.

[49] Weijia He, Maximilian Golla, Roshni Padhi, Jordan Ofek, Markus Dürmuth, Earlence Fernandes, and Blase Ur. 2018. Rethinking Access Control and Authentication for the Home Internet of Things (IoT). In *27th USENIX Security Symposium (USENIX Security 18)*. 255–272.

[50] Cristian Hesselman. 2017. *SPIN: A user-centric security extension for inhome networks*. https://www.sidnlabs.nl/en/newsand-blogs/spin-a-user-centric-security-extension-for-in-home-networks

[51] Masoumehsadat Hosseini, Heiko Mueller, and Susanne Boll. 2024. Controlling the Rooms: How People Prefer Using Gestures to Control Their Smart Homes. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–18.

[52] Hang Hu, Limin Yang, Shihan Lin, and Gang Wang. 2020. Security vetting process of smart-home assistant applications: A first look and case studies. *arXiv preprint arXiv:2001.04520* (2020).

[53] Terence KL Hui, R Simon Sherratt, and Daniel Díaz Sánchez. 2017. Major requirements for building Smart Homes in Smart Cities based on Internet of Things technologies. *Future Generation Computer Systems* 76 (2017), 358–369.

[54] K Hyunjae, Dong Hyun Ahn, Gyung Min Lee, Jeong Do Yoo, Kyung Ho Park, and HK Kim. 2019. IoT network intrusion dataset. *IEEE Dataport* (2019).

[55] IFTTT. 2022. *Every thing works better together*. https://ifttt.com/

[56] Yunhan Jack Jia, Qi Alfred Chen, Shiqi Wang, Amir Rahmati, Earlence Fernandes, Zhuoqing Morley Mao, Atul Prakash, and SJ Unviersity. 2017. ContexloT: Towards providing contextual integrity to appified IoT platforms.. In *NDSS*, Vol. 2. San Diego, 2–2.

[57] Arun Cyril Jose and Reza Malekian. 2017. Improving smart home security: Integrating logical sensing into smart home. *IEEE Sensors Journal* 17, 13 (2017), 4269–4286.

[58] Won Min Kang, Seo Yeon Moon, and Jong Hyuk Park. 2017. An enhanced security framework for home appliances in smart home. *Human-centric Computing and Information Sciences* 7, 1 (2017), 1–12.

[59] Victor R Kebande and Ali Ismail Awad. 2024. Industrial Internet of Things Ecosystems Security and Digital Forensics: Achievements, Open Challenges, and Future Directions. *Comput. Surveys* 56, 5 (2024), 1–37.

[60] Yasir Imtiaz Khan and Maryleen U Ndubuaku. 2018. Ontology-based automation of security guidelines for smart homes. In *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*. IEEE, 35–40.

[61] Pimal Khanpara, Kruti Lavingia, Rajvi Trivedi, Sudeep Tanwar, Amit Verma, and Ravi Sharma. 2023. A context-aware internet of things-driven security scheme for smart homes. *Security and Privacy* 6, 1 (2023), e269.

[62] Nam Yong Kim, Shailendra Rathore, Jung Hyun Ryu, Jin Ho Park, and Jong Hyuk Park. 2018. A survey on cyber physical system security for IoT: issues, challenges, threats, solutions. *Journal of Information Processing Systems* 14, 6 (2018), 1361–1384.

[63] Barbara Kitchenham, O Pearl Brereton, David Budgen, Mark Turner, John Bailey, and Stephen Linkman. 2009. Systematic literature reviews in software engineering–a systematic literature review. *Information and software technology* 51, 1 (2009), 7–15.

[64] Barbara Kitchenham, Rialette Pretorius, David Budgen, O Pearl Brereton, Mark Turner, Mahmood Niazi, and Stephen Linkman. 2010. Systematic literature reviews in software engineering–a tertiary study. *Information and software technology* 52, 8 (2010), 792–805.

[65] Diego Kreutz, Fernando MV Ramos, Paulo Esteves Verissimo, Christian Esteve Rothenberg, Siamak Azodolmolky, and Steve Uhlig. 2014. Software-defined networking: A comprehensive survey. *Proc. IEEE* 103, 1 (2014), 14–76.

[66] D Richard Kuhn, Edward J Coyne, Timothy R Weil, et al. 2010. Adding attributes to role-based access control. *Computer* 43, 6 (2010), 79–81.

[67] Pankaj Kumar and Lokesh Chouhan. 2021. Design of secure session key using unique addressing and identification scheme for smart home Internet of Things network. *Transactions on Emerging Telecommunications Technologies* 32, 5 (2021), e3993.

[68] Philippe Lalanda and Catherine Hamon. 2020. A service-oriented edge platform for cyber-physical systems. *CCF Transactions on Pervasive Computing and Interaction* 2, 3 (2020), 206–217.

[69] Elmer Lastdrager, Cristian Hesselman, Jelte Jansen, and Marco Davids. 2020. Protecting home networks from insecure IoT devices. In *NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 1–6.

[70] Edward A Lee. 2008. Cyber physical systems: Design challenges. In *2008 11th IEEE international symposium on object and component-oriented real-time distributed computing (ISORC)*. IEEE, 363–369.

[71] Guowen Li, Lingyu Ren, Yangyang Fu, Zhiyao Yang, Veronica Adetola, Jin Wen, Qi Zhu, Teresa Wu, K Selcuk Candan, and Zheng O'Neill. 2023. A critical review of cyber-physical security for building automation systems. *Annual Reviews in Control* 55 (2023), 237–254.

[72] Changyuan Lin, Hamzeh Khazaei, Andrew Walenstein, and Andrew Malton. 2021. Autonomic security management for iot smart spaces. *ACM Transactions on Internet of Things* 2, 4 (2021), 1–20.

[73] Kalle Lyytinen and Youngjin Yoo. 2002. Ubiquitous computing. *Commun. ACM* 45, 12 (2002), 63–96.

[74] Kulani Mahadewa, Kailong Wang, Guangdong Bai, Ling Shi, Yan Liu, Jin Song Dong, and Zhenkai Liang. 2019. Scrutinizing implementations of smart home integrations. *IEEE Transactions on Software Engineering* (2019).

[75] Kulani Tharaka Mahadewa, Kailong Wang, Guangdong Bai, Ling Shi, Jin Song Dong, and Zhenkai Liang. 2018. Homescan: scrutinizing implementations of smart home integrations. In *2018 23rd International Conference on Engineering of Complex Computer Systems (ICECCS)*. IEEE, 21–30.

[76] James McNaull, Juan Carlos Augusto, Maurice Mulvenna, and Paul McCullagh. 2012. Multi-agent system feedback and support for ambient assisted living. In *2012 Eighth International Conference on Intelligent Environments*. IEEE, 319–322.

[77] Dagmawi Neway Mekuria, Paolo Sernani, Nicola Falcionelli, and Aldo Franco Dragoni. 2021. Smart home reasoning systems: a systematic literature review. *Journal of Ambient Intelligence and Humanized Computing* 12, 4 (2021), 4485–4502.

[78] Eric Sesterhenn Michael Frey, Cenk Gündogan. 2018. *CNN LITE*. Retrieved 3 August 2022 from https://github.com/cn-uofbasel/ccn-lite

[79] Microsoft. 2018. *The OAuth 2.0 authorization protocol*. Retrieved 3 August 2022 from https://oauth.net/2/

[80] Ayaskanta Mishra, Amitkumar V Jha, Bhargav Appasani, Arun Kumar Ray, Deepak Kumar Gupta, and Abu Nasar Ghazali. 2023. Emerging technologies and design aspects of next generation cyber physical system with a smart city

application perspective. *International Journal of System Assurance Engineering and Management* 14, Suppl 3 (2023), 699–721.

[81] Sirajum Munir and John A Stankovic. 2014. Depsys: Dependency aware integration of cyber-physical systems for smart homes. In *2014 ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 127–138.

[82] Gervais Ntezicyimanikora. 2024. Smart Surveillance System With Anomaly Detection at Home. *Available at SSRN 4714486* (2024).

[83] Babatunji Omoniwa, Riaz Hussain, Muhammad Awais Javed, Safdar Hussain Bouk, and Shahzad A Malik. 2018. Fog/edge computing-based IoT (FECIoT): Architecture, applications, and research issues. *IEEE Internet of Things Journal* 6, 3 (2018), 4118–4149.

[84] openHAB. 2022. *openHAB empowering the smart home.* https://www.openhab.org

[85] Ukachi Osisiogu. 2019. A review on cyber-physical security of smart buildings and infrastructure. In *2019 15th International Conference on Electronics, Computer and Computation (ICECCO)*. IEEE, 1–4.

[86] Collins Patel Michael. 2022. *IoT value set to accelerate through 2030: Where and how to capture it.* Retrieved 1 November 2022 from https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/iotvalue-set-to-accelerate-through-2030-where-and-how-to-capture-it

[87] Ada Diaconescu Philippe Lalanda, Julie McCann. 2020. *Pervasive Computing in Practice.* https://self-star.imag.fr

[88] Manju Mohan Pillai and Albert Helberg. 2021. Improving Security in Smart Home Networks through user-defined device interaction rules. In *2021 IEEE AFRICON*. IEEE, 1–6.

[89] Amjad Qashlan, Priyadarsi Nanda, and Xiangjian He. 2020. Security and privacy implementation in smart home: Attributes based access control and smart contracts. In *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 951–958.

[90] Junjian Qi, Youngjin Kim, Chen Chen, Xiaonan Lu, and Jianhui Wang. 2017. Demand response and smart buildings: A survey of control, communication, and cyber-physical security. *ACM Transactions on Cyber-Physical Systems* 1, 4 (2017), 1–25.

[91] Vu Khanh Quy, Abdellah Chehri, Nguyen Minh Quy, Nguyen Dinh Han, and Nguyen Tien Ban. 2023. Innovative trends in the 6G era: A comprehensive survey of architecture, applications, technologies, and challenges. *IEEE Access* 11 (2023), 39824–39844.

[92] Laura Rafferty, Farkhund Iqbal, Saiqa Aleem, Zhihui Lu, Shih-Chia Huang, and Patrick CK Hung. 2018. Intelligent multi-agent collaboration model for smart home IoT security. In *2018 IEEE international congress on internet of things (ICIOT)*. IEEE, 65–71.

[93] Md Motiur Rahman, Deepti Gupta, Smriti Bhatt, Shiva Shokouhmand, and Miad Faezipour. 2024. A Comprehensive Review of Machine Learning Approaches for Anomaly Detection in Smart Homes: Experimental Analysis and Future Directions. *Future Internet* 16, 4 (2024), 139.

[94] Amir Rahmati, Earlence Fernandes, Kevin Eykholt, and Atul Prakash. 2018. Tyche: A risk-based permission model for smart homes. In *2018 IEEE Cybersecurity Development (SecDev)*. IEEE, 29–36.

[95] Mads Holten Rasmussen, Maxime Lefrançois, Georg Ferdinand Schneider, and Pieter Pauwels. 2021. BOT: the building topology ontology of the W3C linked building data group. *Semantic Web* 12, 1 (2021), 143–161.

[96] Honglei Ren, You Song, Siyu Yang, and Fangling Situ. 2016. Secure smart home: A voiceprint and internet based authentication system for remote accessing. In *2016 11th International Conference on Computer Science & Education (ICCSE)*. IEEE, 247–251.

[97] Huma Samin, Nelly Bencomo, and Peter Sawyer. 2022. Decision-making under uncertainty: be aware of your priorities. *Software and Systems Modeling* (2022), 1–30.

[98] Mahadev Satyanarayanan. 2001. Pervasive computing: Vision and challenges. *IEEE Personal communications* 8, 4 (2001), 10–17.

[99] Megha Sharma, Abhinav Tomar, and Abhishek Hazra. 2024. Edge computing for industry 5.0: fundamental, applications and research challenges. *IEEE Internet of Things Journal* (2024).

[100] Nicholas Shields. 2022. *THE US SMARTHOMEMARKET REPORT: Systems, apps, and devices leading to home automation.* Retrieved July 28 2022 from https://www.businessinsider.com/the-us-smart-home-market-report-systems-apps-and-devices-leading-to-home-automation-2018-3-19?r=US&IR=T

[101] Amit Kumar Sikder, Leonardo Babun, Z Berkay Celik, Abbas Acar, Hidayet Aksu, Patrick McDaniel, Engin Kirda, and A Selcuk Uluagac. 2020. Kratos: Multi-user multi-device-aware access control system for the smart home. In *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks.* 1–12.

[102] Amit Kumar Sikder, Leonardo Babun, and A Selcuk Uluagac. 2021. Aegis+ A Context-aware Platform-independent Security Framework for Smart Home Systems. *Digital Threats: Research and Practice* 2, 1 (2021), 1–33.

[103] Vijay Sivaraman, Dominic Chan, Dylan Earl, and Roksana Boreli. 2016. Smart-phones attacking smart-homes. In *Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks.* 195–200.

[104] Vijay Sivaraman, Hassan Habibi Gharakheili, Arun Vishwanath, Roksana Boreli, and Olivier Mehani. 2015. Network-level security and privacy control for smart-home IoT devices. In *2015 IEEE 11th International conference on wireless and mobile computing, networking and communications (WiMob)*. IEEE, 163–167.

[105] SmartThings. 2018. *SmartThings Public GitHub Repo.* https://github.com/SmartThingsCommunity/SmartThingsPublic

[106] Biljana L Risteska Stojkoska and Kire V Trivodaliev. 2017. A review of Internet of Things for smart home: Challenges and solutions. *Journal of cleaner production* 140 (2017), 1454–1464.

[107] Ming Tao, Jinglong Zuo, Zhusong Liu, Aniello Castiglione, and Francesco Palmieri. 2018. Multi-layer cloud architectural model and ontology-based security service framework for IoT-based smart homes. *Future Generation Computer Systems* 78 (2018), 1040–1051.

[108] Jože Tavčar and Imre Horvath. 2018. A review of the principles of designing smart cyber-physical systems for run-time adaptation: Learned lessons and open issues. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 49, 1 (2018), 145–158.

[109] Corentin Thomasset and David Barrera. 2020. SERENIoT: Distributed Network Security Policy Management and Enforcement for Smart Homes. In *Annual Computer Security Applications Conference*. 542–555.

[110] Amit Kumar Tyagi and N Sreenath. 2021. Cyber physical systems: Analyses, challenges and possible solutions. *Internet of Things and Cyber-Physical Systems* (2021).

[111] Di Wang, Fangyu Li, Kaibo Liu, and Xi Zhang. 2024. Real-time cyber-physical security solution leveraging an integrated learning-based approach. *ACM Transactions on Sensor Networks* 20, 2 (2024), 1–22.

[112] Jean-Paul A Yaacoub, Ola Salman, Hassan N Noura, Nesrine Kaaniche, Ali Chehab, and Mohamad Malli. 2020. Cyber-physical systems security: Limitations, issues and future trends. *Microprocessors and microsystems* 77 (2020), 103201.

[113] Moosa Yahyazadeh, Proyash Podder, Endadul Hoque, and Omar Chowdhury. 2019. Expat: Expectation-based policy analysis and enforcement for appified smart-home platforms. In *Proceedings of the 24th ACM symposium on access control models and technologies*. 61–72.

[114] Moosa Yahyazadeh, Proyash Podder, Endadul Hoque, and Omar Chowdhury. 2019. *Expat Github Repository.* https://github.com/expat-paper/expat.git

[115] Nikola Zaric, Milutin Radonjic, Milica Pejanovic-Djurisic, and Igor Radusinovic. 2015. An example of monitoring system with reasoning module for ambient assisted living applications. In *IEEE EUROCON 2015-International Conference on Computer as a Tool (EUROCON)*. IEEE, 1–6.

[116] Eric Zeng, Shrirang Mare, and Franziska Roesner. 2017. End user security and privacy concerns with smart homes. In *Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017)*. 65–80.

[117] Eric Zeng and Franziska Roesner. 2019. Understanding and Improving Security and Privacy in Multi-User Smart Homes: A Design Exploration and In-Home User Study. In *28th USENIX Security Symposium (USENIX Security 19)*. 159–176.

[118] Engin Zeydan, Ejder Bastug, Mehdi Bennis, Manhal Abdel Kader, Ilyas Alper Karatepe, Ahmet Salih Er, and Mérouane Debbah. 2016. Big data caching for networking: Moving from cloud to edge. *IEEE Communications Magazine* 54, 9 (2016), 36–42.

[119] Wei Zhang, Yan Meng, Yugeng Liu, Xiaokuan Zhang, Yinqian Zhang, and Haojin Zhu. 2018. Homonit: Monitoring smart home apps from encrypted traffic. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. 1074–1088.

[120] Zhiyi Zhang, Tianyuan Yu, Xinyu Ma, Yu Guan, Philipp Moll, and Lixia Zhang. 2022. Sovereign: Self-contained Smart Home with Data-centric Network and Security. *IEEE Internet of Things Journal* (2022).

[121] Adam Zielonka, Marcin Woźniak, Sahil Garg, Georges Kaddoum, Md Jalil Piran, and Ghulam Muhammad. 2021. Smart homes: How much will they support us? A research on recent trends and advances. *IEEE Access* 9 (2021), 26388–26419.