electronics

MDPI

Article

Automated Vulnerability Scanning and Prioritisation for Domestic IoT Devices/Smart Homes: A Theoretical Framework

Diego Fernando Rivas Bustos [1], Jairo A. Gutierrez [1,*] and Sandra J. Rueda [2]

[1] School of Engineering, Computer and Mathematical Sciences, Auckland University of Technology, Auckland 1010, New Zealand
[2] Systems and Computing Engineering Department, Universidad de Los Andes, Bogotá 111711, Colombia; sarueda@uniandes.edu.co
* Correspondence: jairo.gutierrez@aut.ac.nz

Abstract

The expansion of Internet of Things (IoT) devices in domestic smart homes has created new conveniences but also significant security risks. Insecure firmware, weak authentication and weak encryption leave households exposed to privacy breaches, data leakage and systemic attacks. Although research has addressed several challenges, contributions remain fragmented and difficult for non-technical users to apply. This work addresses the following research question: How can a theoretical framework be developed to enable automated vulnerability scanning and prioritisation for non-technical users in domestic IoT environments? A Systematic Literature Review of 40 peer-reviewed studies, conducted under PRISMA 2020 guidelines, identified four structural gaps: dispersed vulnerability knowledge, fragmented scanning approaches, over-reliance on technical severity in prioritisation and weak protocol standardisation. The paper introduces a four-module framework: a Vulnerability Knowledge Base, an Automated Scanning Engine, a Context-Aware Prioritisation Module and a Standardisation and Interoperability Layer. The framework advances knowledge by integrating previously siloed approaches into a layered and iterative artefact tailored to households. While limited to conceptual evaluation, the framework establishes a foundation for future work in prototype development, household usability studies and empirical validation. By addressing fragmented evidence with a coherent and adaptive design, the study contributes to both academic understanding and practical resilience, offering a pathway toward more secure and trustworthy domestic IoT ecosystems.

Keywords: domestic Internet of Things (IoT); smart home security; automated vulnerability scanning; vulnerability prioritisation

Academic Editors: Irfan Awan, Amna Qureshi and Muhammad Shahwaiz Afaqui

Received: 13 December 2025
Revised: 16 January 2026
Accepted: 19 January 2026
Published: 21 January 2026

1. Introduction

The rapid expansion of Internet of Things (IoT) devices in domestic smart homes has introduced both new conveniences and significant cybersecurity challenges. Devices such as connected cameras, wearables and household appliances are increasingly embedded in everyday life, yet many are deployed with weak authentication and default credentials that make them vulnerable to compromise [1]. Insecure implementations of widely used protocols such as Zigbee and CoAP further intensify device exposure [2], while misconfigurations and poor lifecycle management amplify household risks [3]. Firmware flaws have also been shown to be readily exploitable in penetration test environments [4], and the resulting vulnerabilities can have severe privacy implications, such as leakage of sensitive

data from domestic surveillance devices [5]. Heterogeneity across manufacturers and protocols amplifies these issues, leaving households exposed to risks of Denial-of-Service and enumeration attacks when protections are inconsistently applied [6].

Although research in IoT security has advanced in many directions, the evidence base remains fragmented. Vulnerability studies, scanning techniques, prioritisation frameworks and protocol analyses are often published in isolation, making it difficult for non-technical households to benefit from a unified model.

The systematic literature review presented in Section 2 highlights four structural gaps:

- Vulnerability knowledge is dispersed, with inconsistent classifications across devices and studies;
- Scanning approaches such as penetration testing, fuzzing and traffic monitoring remain siloed, preventing their integration into a coherent process [4,7];
- Prioritisation models such as CVSS tend to emphasise technical severity while neglecting household-specific context such as device function, privacy impact and protocol exposure [1,5];
- Finally, weak standardisation across common IoT protocols, including MQTT, Zigbee, Z-Wave, BLE and CoAP, results in uneven protections across households [2,6].

These shortcomings underscore the need for a theoretical framework that systematically consolidates academic evidence into an artefact tailored to domestic IoT security.

The paper adopts a theoretical framework approach (as opposed to a conceptual framework) as it will use established theories to examine the issues associated with the four research gaps. The primary research question guiding this study is the following:

> *How can a theoretical framework be developed to enable automated vulnerability scanning and prioritisation for non-technical users in domestic IoT smart home environments?*

This study aims to (1) design a Vulnerability Knowledge Base that consolidates dispersed academic evidence into a coherent repository; (2) develop an Automated Scanning Engine that integrates multiple approaches into a single process; (3) design a Context-Aware Prioritisation Module that embeds household relevance into risk scoring; and (4) propose a Standardisation and Interoperability Layer that harmonises security features across common IoT protocols. Together, these objectives define a layered theoretical framework that responds directly to the gaps revealed in the literature.

The framework is conceived as an iterative refinement cycle rather than a fixed linear sequence. Feedback loops between its first three modules, vulnerability knowledge, automated scanning and context-aware prioritisation; allow new scanning outputs to update the knowledge base and influence future prioritisation decisions. This design ensures adaptability as domestic IoT ecosystems evolve and aligns with the Design Science Research Methodology principle of artefact evolution [8].

2. Literature Review

This section synthesises 40 academic sources to provide the theoretical grounding for this work. The discussion is organised in alignment with the layers of the expected theoretical framework: 1. Vulnerabilities in domestic IoT; 2. Automated vulnerability scanning tools; 3. Prioritisation strategies; and 4. Standardisation and interoperability. By identifying commonalities, limitations and gaps across these bodies of work, this literature review demonstrates the need for a domestic IoT security framework tailored for non-technical users.

2.1. Prevalent Vulnerabilities in Domestic IoT Devices

The literature shows that domestic IoT devices remain susceptible to vulnerabilities that undermine both functionality and household security. Early surveys classified vulnerabilities into categories spanning hardware, software and communication layers. Classic taxonomies remain influential, with ref. [9] identifying early gaps in trust, privacy and security. In the context of information security, "trust" is a measured belief that a system, user or device will behave in an expected, secure and authorised manner. For instance, ref. [1] demonstrated how insecure firmware updates, poor key management and default credentials leave devices easily compromised. Recent analyses also emphasise device-specific weaknesses in smart homes [1]. Similarly, ref. [10] identified systemic weaknesses including side-channel attacks, hardware Trojans and inadequate access control that expose smart home devices to escalating threats. Ref. [11] further highlighted how permission escalation within IoT hubs could compromise multiple devices in a single household, amplifying the risk from a single exploit. From a foundational perspective, ref. [12] emphasise the systematic nature of IoT risks, linking device constraints and ecosystem complexity to persistent exposure in consumer settings. Ref. [5] reinforced these findings, showing that vulnerability surfaces in consumer IoT remain fragmented, heterogeneous and largely unmitigated despite increased research attention.

Empirical research provides concrete evidence of these weaknesses. Ref. [1] tested commercially available smart home devices using open-source tools, identifying vulnerabilities that were not only numerous but also remotely exploitable. Their findings stress the importance of distinguishing between vulnerabilities that require physical access and those that can be exploited remotely. The former present lower risk to most consumers, while the latter can threaten millions of devices simultaneously. Large-scale incidents have demonstrated the catastrophic potential of remote compromise in consumer IoT. Hijacking campaigns against smart cameras and household devices have shown how poorly secured endpoints can be leveraged at scale to disrupt services and compromise household privacy [1,5]. Similarly, at DEF CON 2017, researchers disclosed 47 new vulnerabilities in 23 devices from 21 manufacturers, underscoring the persistent industry-wide exposure [1]. Databases such as Exploitee.rs and HardwareSecurity.org, referenced in [1], continue to catalogue vulnerabilities across more than 200 devices, nearly half of which are designed for smart home environments.

Weak authentication, unencrypted communications and outdated firmware have been reported for different types of IoT devices, including smart cameras, thermostats, smart lighting, and home appliances. Table 1 summarises vulnerabilities found in Domestic IoT devices.

Table 1. Summary of vulnerabilities in domestic IoT devices (authors' synthesis based on [1,5,9–11]).

Device Type	Common Vulnerabilities	Protocols	Risk Level
Smart Cameras	Weak/default credentials, stream hijacking	HTTP, RTSP	High
Thermostats	Remote manipulation, outdated firmware	Zigbee, Z-Wave	Medium–High
Smart Lighting	Unauthorised access, weak encryption	Bluetooth, Wi-Fi	Low–Medium
Home Appliances	Default credentials, unpatched software	HTTP, Telnet	High

This evidence demonstrates that vulnerabilities in domestic IoT vary significantly in impact, with remote exploits posing the greatest household risks. While existing research identifies these weaknesses comprehensively, there is no unified framework that contextualises them according to household exposure. These findings define Gap #1: Absence of a unified vulnerability framework for contextualising domestic IoT risks, addressed by Module #1: Vulnerability Knowledge Base in the proposed framework.

Having established the range of prevalent vulnerabilities in domestic IoT devices, the next section considers how automated scanning tools have been developed to identify such weaknesses and assess their potential impact.

2.2. Automated Vulnerability Scanning Tools

Automated scanning is a cornerstone of vulnerability discovery in IoT ecosystems, but existing tools differ significantly in scope, accuracy and household applicability. State-of-the-art surveys of IoT vulnerability scanning approaches highlight the limits of existing automated tools [13]. Traditional network scanners such as Nmap and Masscan remain foundational for device discovery and port analysis. These tools provide wide coverage and efficiency in enumerating services and identifying exposed ports, but they cannot detect deeper firmware vulnerabilities or device-specific flaws [14,15]. At the internet scale, Shodan enables global enumeration of IoT devices and services but is limited by its reliance on banner grabbing, which restricts the precision of vulnerability identification [6].

Dynamic and emulation-based analysis techniques attempt to address these shortcomings by inspecting device behaviour at runtime. Frameworks such as Avatar and Firmadyne enable the re-hosting of IoT firmware and symbolic execution, allowing for the discovery of hidden vulnerabilities such as authentication bypasses and insecure default services [1]. These approaches expose flaws that cannot be captured by surface-level scanning alone, but their reliance on resource-intensive processes and technical expertise makes them unsuitable for direct application in domestic settings [7,14,15]. Addressing resource constraints, ref. [16] proposes a scalable, lightweight AI-driven security framework that applies optimisation and game-theoretic strategies, highlighting pathways to reduce overhead while sustaining detection efficacy in constrained IoT settings. Automated penetration testing for smart home devices has been operationalised in prototype frameworks [17].

Recent research has sought to combine static and dynamic techniques to improve accuracy. Complementing these hybrid approaches, ref. [18] present a machine-learning-based cybersecurity framework for IoT devices that operationalises classifier-driven detection within practical deployment constraints, reinforcing the role of ML as a bridging mechanism between surface scanning and deeper behavioural analysis. Ref. [19] proposed an automated IoT assessment framework that integrates firmware re-hosting with network scanning, reducing blind spots across the analysis spectrum. Ref. [7] advanced this trajectory by introducing generative fuzzing tailored to IoT networks, capable of automatically generating new test cases that go beyond signature-based detection. At a broader level, ref. [20] review generative-AI applications for IoT security, indicating that model-driven generation can extend beyond fuzzing to support adaptive detection and mitigation strategies across heterogeneous devices. While these hybrid approaches improve comprehensiveness, they remain fragmented across toolchains and lack integration into user-friendly solutions for households. Complementary studies classify consumer IoT software vulnerabilities, improving scanning workflows [21]. Table 2 compares these different approaches.

Table 2. Comparison of vulnerability scanning approaches.

Approach	Examples	Strengths	Limitations	Suitability
Traditional	Shodan, Nmap, Masscan	Wide coverage, simple use	Outdated results, limited context	Moderate
Static/Firmware	OVER, firmware audits	Scalable, supply-chain visibility	Miss runtime issues	Low-Moderate
AI-Enhanced	ML-based scanning	High accuracy, adaptive	High computational cost, data dependent	Moderate-High
Penetration/Fuzzing	IoTective, AutoDES, GAN fuzzing	Exploit feasibility evidence	Complex, requires expertise	Low (research stage)

While Table 2 compares scanning approaches strictly in terms of technical attributes, Figure 1 presents a timeline that combines chronology with methodological categories. It illustrates when key contributions emerged, beginning with early academic contributions such as [2,11], which referenced surface-level enumeration tools such as Nmap and Shodan [1,5]. Scanning practices then evolved toward automated discovery, firmware analysis and auditing, culminating in AI-driven fuzzing.

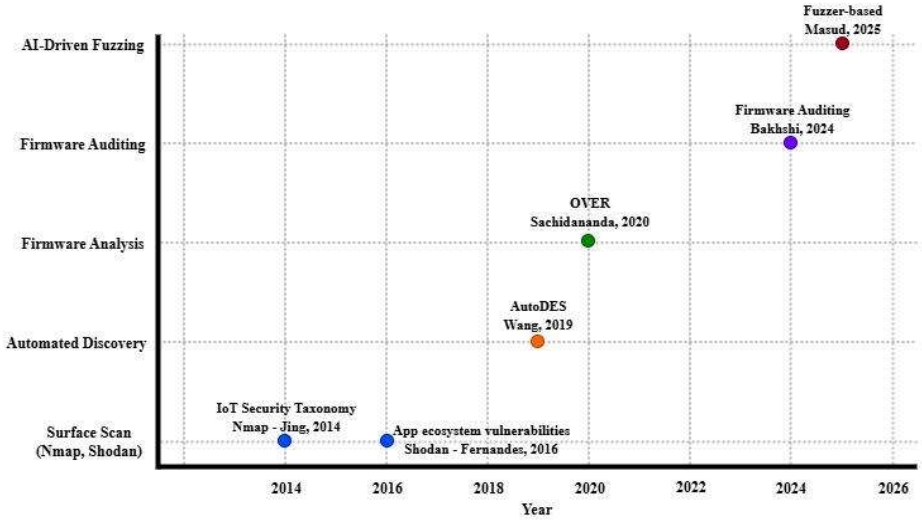

Figure 1. Timeline of scanning methods (2014–2025). It shows the evolution from surface-level enumeration through automated discovery and firmware analysis to AI-driven fuzzing. The combination of chronology and category layering highlights both temporal sequence and conceptual progression in the development of scanning approaches [2,6,7,11,14,15].

The evidence shows that automated scanning in IoT is simultaneously powerful and fragmented. Traditional tools provide shallow insight, while emulation-based and fuzzing methods achieve depth at the cost of usability. No framework consolidates these methods into an accessible form suitable for smart home environments. These findings define Gap #2: Fragmentation of automated scanning approaches, which remain tool-specific and poorly integrated, addressed by Module #2: Automated Scanning Engine.

While scanning tools help reveal a wide spectrum of flaws, the challenge of prioritising which vulnerabilities to address first requires a different perspective.

2.3. Prioritisation Strategies

Identifying vulnerabilities is only the first step; determining which to address first is equally critical for effective mitigation in domestic IoT environments. The most widely adopted system for prioritisation is the Common Vulnerability Scoring System (CVSS), which rates vulnerabilities based on exploitability, impact and access vectors. While CVSS remains a global standard, its application in smart home environments has been criticised for misrepresenting risk; "risk" is understood as the potential for loss or harm to an organisation's data or systems, calculated from the likelihood of a threat exploiting a vulnerability. For instance, a high CVSS score for a smart light bulb may receive more attention than a moderate vulnerability in a security camera, despite the latter presenting a far greater household security threat [5]. Risk scoring for IoT devices has also been modelled through fuzzy logic and optimisation, demonstrating alternative prioritisation strategies beyond CVSS [22].

Alternative frameworks attempt to correct these shortcomings. Firmware auditing reviews reveal gaps in prioritisation at the binary level, where vulnerabilities are often documented but not contextualised for remediation [15]. Ref. [23] applied machine-learning models to detect IoT attacks early, generating rankings derived from anomaly detection metrics. Predictive approaches have also been proposed to anticipate malicious behaviour in IoT devices, strengthening the link between identification and actionable prioritisation [24]. While technically robust, these methods still privilege exploit signatures and fail to account for device criticality in household settings. Ref. [3] developed AI-driven systemic risk models, connecting technical vulnerabilities to organisational and societal impacts. While this improves the systemic relevance of prioritisation, it lacks granularity at the level of household devices. More directly, ref. [25] introduced the CRASHED framework, explicitly targeting smart home contexts by incorporating device roles and exposure into prioritisation logic. Ref. [26] proposed the IoT Security Framework (ISF), which emphasised device interdependencies and ecosystem-wide risk rather than isolated technical vulnerabilities. Table 3 presents prioritisation frameworks.

Table 3. Prioritisation frameworks for IoT vulnerabilities (authors' synthesis based on [3,5,23,25]).

Framework/Approach	Criteria	Limitations	Relevance
CVSS	Severity, exploitability, impact	No user/device context	Baseline only
SAFER	Current/Future risk indicators	Complex, patch data required	Promising for device forecasting
ML-based	Traffic anomalies	Dataset dependent	Edge-friendly, incomplete
Dependency-based	Cascading risk propagation	Data intensive	Highlights systemic impact

The literature shows that prioritisation frameworks have yet to balance technical severity with household context. Without this, security resources risk being misallocated to less impactful flaws while more dangerous vulnerabilities remain unaddressed. This defines Gap #3: Over-reliance on technical severity in prioritisation frameworks, with limited recognition of household context, addressed by Module #3: Context-Aware Prioritisation. Figure 2 shows a conceptual model of technical vs. user-centric (user context) prioritisation.

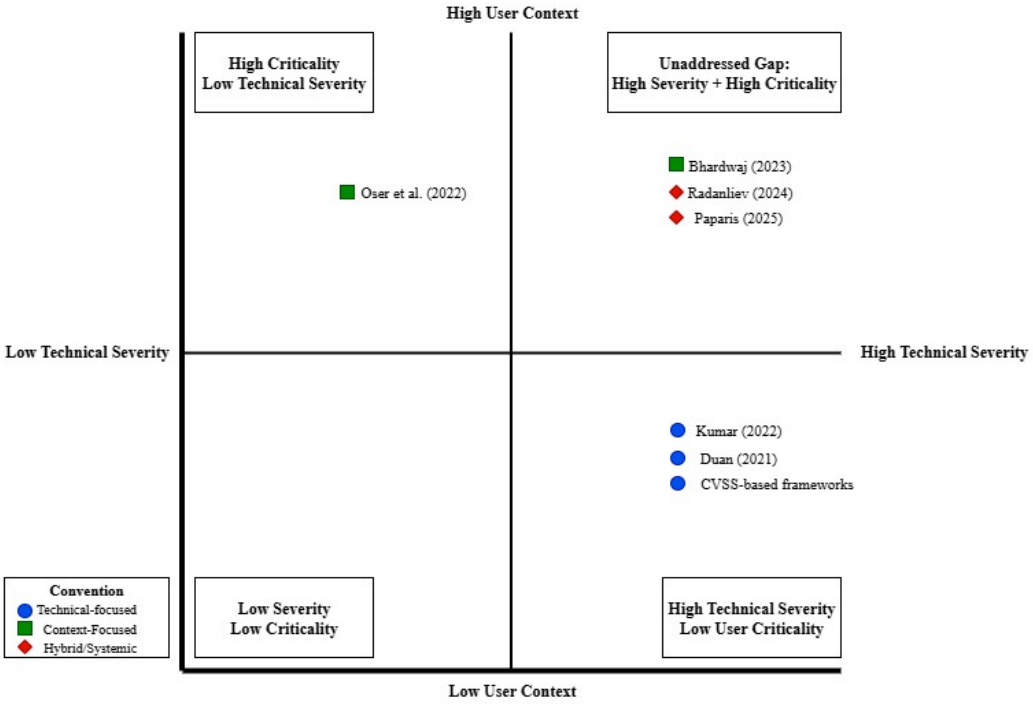

Figure 2. Conceptual model of technical vs. user-centric prioritisation (authors' own work, informed by [3,5,19,23,25,26]).

Although prioritisation strategies enhance the management of vulnerabilities, they rely on the quality of underlying communication protocols. Section 2.4 therefore examines the standards, interoperability and security features that underpin IoT ecosystems.

Despite these innovations, the overall picture remains fragmented. CVSS and ML-based systems are weighted heavily toward technical severity, while more context-aware approaches like CRASHED and ISF show promise but remain poorly integrated with widely used scoring systems. This misalignment creates a persistent blind spot: vulnerabilities that are simultaneously high-severity and high-criticality within households are not systematically prioritised.

Intrusion Detection and Anomaly Detection in Domestic IoT: While prioritisation frameworks identify which vulnerabilities matter most, effective protection in domestic IoT also depends on mechanisms that can detect exploitation attempts in real time. Intrusion detection systems (IDSs) and anomaly detection frameworks extend the security posture beyond scanning by monitoring device and network behaviour for malicious patterns.

Ref. [23] demonstrated that machine-learning-based IDS at the network edge can detect IoT botnet activity at an early stage, preventing attacks before household compromise occurs. Ref. [27] provided a systematic review of machine-learning approaches for IoT botnet detection, consolidating the role of classifiers such as random forests, SVMs and neural networks in anomaly detection. Ref. [28] introduced READ-IoT, a reliable anomaly detection framework designed to maintain event integrity in heterogeneous device environments. Ref. [29] proposed ADRIoT, an edge-assisted anomaly detection framework that distributes detection workloads, improving scalability for large domestic IoT networks. Ref. [30] advanced this direction with SARIK, a Kubernetes-based policy and security framework for IoT devices, enabling anomaly detection and mitigation in containerised environments. Ref. [31] contributed a hybrid deep-learning framework for IoT security, combining convolutional and recurrent models to improve anomaly detection accuracy. Extending this line, ref. [32] integrate CoviNet with a Granger-causality-inspired graph-neural approach to compress and analyse cloud-side IoT streams, improving anomaly detection

scalability for deployments that blend edge devices with cloud services. Ref. [33] extended the application of IDS beyond households, analysing vulnerabilities and intrusion detection strategies in smart city environments, highlighting how these methods can be transferred to domestic contexts. To improve interpretability for non-technical users, ref. [34] propose an explainable-AI design for smart home IDSs, indicating that transparent feature attributions can support user-centred remediation decisions.

Together, these studies confirm that IDS and anomaly detection techniques are essential complements to vulnerability identification and prioritisation, enabling proactive responses to evolving attacks. However, most remain highly technical, lacking the usability and integration required for household adoption, thereby reinforcing the need for a framework that translates advanced detection into user-accessible protection.

2.4. Protocols, Interoperability and Security Features

The security posture of domestic IoT ecosystems depends not only on device architecture but also on the protocols that interconnect them. These protocols embed varying levels of protection, yet their inconsistent adoption across devices and vendors creates systemic risks for households.

At the application layer, the Message Queuing Telemetry Transport (MQTT) protocol has become a dominant standard for lightweight messaging. It supports TLS encryption, but implementation is optional and many consumer devices ship with unencrypted configurations [1]. The Constrained Application Protocol (CoAP) was designed for constrained devices and provides Datagram Transport Layer Security (DTLS). However, its computational overhead makes it unsuitable for highly resource-constrained hardware, leading to limited deployment in practice [2].

At the network and transport layers, Zigbee and Z-Wave are widely used in smart homes. Zigbee integrates AES-128 encryption; however, published analyses report key-extraction and replay-style attacks that undermine reliability [2,10]. Z-Wave strengthened its security with the introduction of the S2 framework and Elliptic-curve Diffie–Hellman (ECDH) key exchange, yet legacy devices lacking these features remain prevalent in households [10]. Bluetooth Low Energy (BLE) supports multiple pairing and bonding mechanisms but continues to be susceptible to downgrade and sniffing attacks [10].

At the perception layer, devices such as sensors, RFID tags and hardware modules form the foundation of the IoT ecosystem. While critical to data collection, they typically operate under resource constraints and rely on proprietary or lightweight communication standards with limited encryption. As [2] noted, RFID and sensor networks are particularly exposed due to heterogeneous deployments and lack of unified standards. Ref. [10] further emphasised hardware-level vulnerabilities such as side-channel attacks, hardware Trojans and sensor spoofing, which remain outside the coverage of higher-layer security mechanisms.

In the networking field, “interoperability” means the ability of diverse systems or components (regardless of vendor or carrier) to seamlessly exchange and use data in real time. In a smart home context, the coexistence of networking protocols across layers in a single household intensifies risks, as hubs and gateways often link devices across standards. This creates complex interoperability chains where a weakness in one layer may cascade into others. For example, perception-layer spoofing of sensor data can propagate through network protocols into application-layer compromises. Ref. [2] highlighted that such cross-layer interactions amplify risks, particularly when proprietary extensions and vendor-specific implementations prioritise functionality over consistent security enforcement. According to early analyses of the IoT security landscape [12], protocol-layer protections

may be undermined by heterogeneous deployments and legacy implementations. Table 4 summarises the used protocols and their security features.

Table 4. IoT protocols and security features (authors' synthesis based on [2,4,6,10]).

Protocol	Security Features	Limitations
MQTT	TLS support, lightweight	Optional encryption, weak adoption
Zigbee	AES-128 encryption	Key extraction vulnerabilities
Z-Wave	S2 framework, ECDH	Legacy devices insecure
Bluetooth LE	Pairing/bonding modes	Susceptible to sniffing, downgrade attacks
CoAP	DTLS support	Resource-heavy for constrained devices

Figure 3 consolidates information per layer and highlights cross-layer risks, vulnerabilities that extend beyond a single protocol layer. Data leakage arises when perception-layer devices such as RFID tags or sensors transmit unencrypted information, exposing it as it moves through transport and application protocols. Spoofing occurs when malicious or compromised devices inject false data at the perception layer, which is then propagated by higher-layer protocols into application services. Weak enforcement reflects the inconsistent application of security features across layers; for example, even if CoAP enforces DTLS at the application level, its protections may be undermined by weaker or legacy encryption in underlying transport protocols such as Zigbee. These cross-layer risks underscore the need for integrated security enforcement, as weaknesses at one layer can compromise the resilience of the entire smart home ecosystem [2,10].

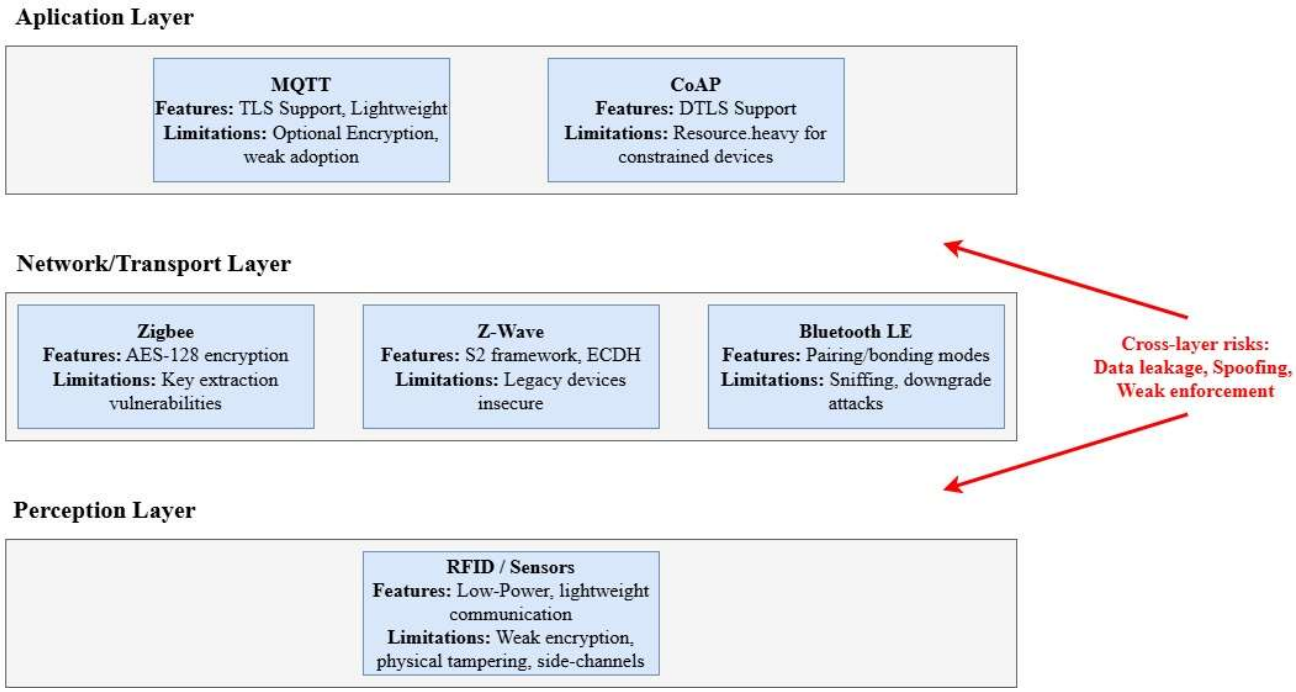

Figure 3. Interoperability and cross-layer risks (adapted from [2] with refinements based on [1,6,10]. Representation of application, transport/network and perception layers in domestic IoT, annotated with protocol security features and limitations. Arrows indicate how weaknesses propagate across layers.

The evidence demonstrates that IoT protocols and device communication mechanisms embed useful security features but fail in practice due to inconsistent adoption, legacy vulnerabilities and cross-layer risks. Weak encryption at the perception layer, fragmented adoption of secure transport standards and uneven enforcement of application-layer protections illustrate how vulnerabilities can cascade across layers. Adaptive policy frameworks dynamically adjust IoT security at the edge [35]. Fog computing research highlights unresolved privacy/security trade-offs [36]. These dynamics confirm Gap #4: Weak standardisation of IoT protocols, enabling interoperability failures and cross-layer vulnerabilities, a gap addressed by Module #4: Standardisation and Interoperability Layer.

Protocol- and interoperability-related weaknesses establish how systemic flaws persist across IoT layers. To integrate these observations into a coherent structure, Section 2.5 consolidates evidence across all subsections and maps it into research gaps and framework modules.

2.5. Evidence-to-Framework Traceability

The literature reviewed in Sections 2.1–2.4 confirms that research on IoT security has generated significant insights, but it also reveals persistent structural limitations that affect domestic applicability. To ensure transparency between the evidence base and the design of the proposed framework, it is essential to map contributions from the literature to unresolved gaps and then to the modules that will address them in Section 4.

Four major gaps emerge:

- Gap #1 (vulnerabilities): Absence of a unified vulnerability framework for contextualising domestic IoT risks;
- Gap #2 (scanning tools): Fragmentation of automated scanning approaches, which remain tool-specific and poorly integrated;
- Gap #3 (prioritisation): Over-reliance on technical severity in prioritisation frameworks, with limited recognition of household context;
- Gap #4 (frameworks and protocols): Weak standardisation of IoT protocols, enabling interoperability failures and cross-layer vulnerabilities.

Gaps 1 to 4 motivate a framework with four modules:

- Vulnerabilities highlight the need for Module #1: Vulnerability Knowledge Base;
- Scanning tools confirm the need for Module #2: Automated Scanning Engine;
- Prioritisation strategies demonstrate the need for Module #3: Context-Aware Prioritisation Module;
- Protocols underscore the necessity of Module #4: Standardisation and Interoperability Layer.

Table 5 consolidates the connections between research gaps identified from the literature and the framework modules.

The evidence presented demonstrates that the proposed framework is not speculative but a structured response to gaps systematically identified in the literature. Each module emerges directly from deficiencies observed across prior work, ensuring academic rigour and practical relevance for domestic IoT environments.

The analysis across vulnerabilities, scanning tools, prioritisation strategies and protocols shows that while progress has been made in understanding domestic IoT risks, critical gaps remain unresolved. The persistence of these issues indicates that individually, solutions cannot adequately protect households; instead, a coherent framework is required to integrate the diverse contributions of prior research.

Table 5. Research gaps identified from the literature.

Contribution Cluster	Supporting Evidence	Gap Identified	Framework Module
Vulnerabilities in Domestic IoT Devices	[1,5,9–11,37,38]	Gap #1: No unified framework exists to systematically categorise and contextualise vulnerabilities in household settings.	Module #1: Vulnerability Knowledge Base
Automated Scanning Tools	[1,6,7,14,15,19, 23,27–31,33]	Gap #2: Scanning approaches are fragmented across tools, leaving blind spots and lacking household-ready integration.	Module #2: Automated Scanning Engine
Prioritisation Frameworks	[3,5,15,22–26]	Gap #3: Over-reliance on technical severity metrics; household context and device criticality are underrepresented.	Module #3: Context-Aware Prioritisation Module
Protocols and Interoperability	[1,2,10,35,36]	Gap #4: Weak standardisation, legacy protocols and cross-layer risks undermine consistent security enforcement.	Module #4: Standardisation & Interoperability Layer

3. Methodology

This section explains the selection of the Design Science Research Methodology (DSRM) as the guiding methodology, describes the use of a Systematic Literature Review (SLR), the curated dataset of 40 academic sources, and introduces the mapping strategies (evidence-to-framework) that ensure transparency and traceability.

3.1. Design Science Research Methodology (DSRM)

This work employs the Design Science Research Methodology (DSRM) to guide the construction of a theoretical framework for automated vulnerability scanning and prioritisation in domestic IoT devices. The DSRM provides a structured process for developing artefacts that both addresses identified problems and maintains academic rigour.

The methodology consists of six stages [8,39]: (1) Problem Identification and Motivation; (2) Define Objectives of a Solution; (3) Design and Development; (4) Demonstration; (5) Evaluation; and (6) Communication. Each stage is tailored to the context of IoT security:

Problem Identification and Motivation: The literature review in Section 2 revealed fragmentation across four domains: vulnerabilities, scanning tools, prioritisation models and standardisation. These challenges directly affect households adopting smart home technologies, motivating the development of a framework that consolidates these aspects into a coherent structure.

Define Objectives of a Solution: The primary objective is to design a theoretical framework that integrates automated scanning, context-aware prioritisation and standardisation principles.

Design and Development: The framework is constructed through an evidence-to-module mapping process. Thematic tables and figures from Section 2 provide the design in-

puts. Each identified gap (summarised in Tables 5 and 6) is mapped to a framework module, resulting in a layered model that systematically addresses domestic IoT security challenges.

Table 6. Mapping of vulnerability sources to knowledge base functions (authors' own work, based on SLR dataset).

Source	Vulnerability Focus	Contribution to Knowledge Base
[1]	Weak authentication and default credentials	Defines credential-related vulnerability class
[2]	Protocol-level weaknesses in Zigbee, CoAP	Provides cross-protocol vulnerability evidence
[4]	Firmware flaws identified via penetration testing	Adds classification of lifecycle/firmware vulnerabilities
[5]	Privacy-related vulnerabilities (e.g., cameras, wearables)	Expands taxonomy with privacy/data exposure vulnerabilities
[3]	Device misconfiguration risks	Creates explicit category for configuration-related vulnerabilities
[6]	Insecure network services	Defines service-exposure vulnerabilities in household IoT
[9]	Foundational taxonomy of IoT trust, privacy and security	Provides early conceptual classification supporting consolidation
[38]	Smart camera vulnerabilities	Adds household-critical device category to taxonomy
[37]	Socio-technical digital harms in smart homes	Extends taxonomy beyond technical flaws to include user harms

Demonstration: Demonstration is limited to conceptual validation by showing that the framework adequately addresses the gaps identified in the literature. Practical prototyping or empirical validation is deferred to future work.

Evaluation: Evaluation aims to ensure the proposed framework is rigorous and credible. Following the guidance on evaluation within the DSR body of work [8], this study emphasises traceability and construct validity across the artefact's modules. Two forms of assessment are employed: (i) Internal consistency check—verifying that each framework module directly addresses the gaps identified in the dataset; and (ii) Traceability assurance—employing Table 5 (research gaps identified from the literature and mapping to framework modules) to ensure transparent alignment between sources, gaps and design modules.

Communication: The final stage communicates the artefact and its contribution to academic and practitioner audiences.

3.2. Systematic Literature Review Process

The review was conducted in accordance with PRISMA 2020 guidelines, ensuring transparency and replicability. The process involved the following:

- Database searches: Targeted searches were performed in leading scientific databases (ACM Digital Library, IEEE Xplore, ScienceDirect and SpringerLink) using structured strings focused on IoT security and vulnerability management. The primary search string combined three clusters of terms: (1) device/domain scope ("Internet of Things" OR "IoT" OR "smart home" OR "domestic IoT"); (2) vulnerability dimension ("vulnerability scanning" OR "automated vulnerability detection" OR "vulnerability prioritization" OR "vulnerability prioritisation"); and (3) framework context ("cybersecurity framework" OR "security framework" OR "theoretical framework"). Complementary search strings were applied to capture additional studies on protocols, risk scoring

and prioritisation models. Abstract-level filters ensured a domestic IoT focus (e.g., "smart home").

- Screening and eligibility: Titles, abstracts and keywords were screened against predefined inclusion and exclusion criteria to ensure relevance to domestic IoT ecosystems.
- Deduplication and quality appraisal: Duplicate records were removed and retained studies were assessed for credibility and scholarly rigour. Inclusion criteria required studies to be peer-reviewed journal articles, conference papers or book chapters published in English between 2015 and 2025, explicitly addressing vulnerabilities, scanning, prioritisation or protocol security in domestic IoT contexts. Exclusion criteria removed studies on industrial IoT, non-networked devices, physical hardware vulnerabilities, non-English publications and grey literature without peer review.
- Selection of dataset: From an initial pool of 722 records, a final dataset of 40 academic sources was established.

This process is documented in the PRISMA 2020 Flow Diagram (Figure 4), which illustrates the numbers of records identified, screened, excluded and retained.

The curated dataset consists of 40 peer-reviewed academic sources covering the four thematic clusters:

- Vulnerabilities in Domestic IoT Devices (e.g., default credentials, firmware flaws, weak encryption);
- Automated and AI-Driven Scanning Approaches (e.g., Nmap, Shodan, Avatar, Firmadyne, generative fuzzing);
- Prioritisation Frameworks (e.g., CVSS, CRASHED, ISF, ML-based anomaly detection);
- Protocols and Interoperability Models (e.g., MQTT, Zigbee, Z-Wave, CoAP, BLE).

The literature was examined to identify limitations, blind spots and systemic challenges. Examples include fragmented vulnerability taxonomies, siloed scanning tools, prioritisation models that lack household context and inconsistent adoption of secure protocols. These weaknesses were consolidated into the four structural gaps described in Section 2.

3.3. Limitations

While the methodological design of this work ensures rigour and transparency, several limitations must be acknowledged:

Absence of Primary Data Collection: This research relies exclusively on secondary data in the form of published academic literature. No primary data were collected from households, device vendors or security practitioners. While this ensures methodological consistency and avoids the ethical complexities of human participation, it also means that user-centric considerations are inferred indirectly from prior studies rather than directly validated through empirical field work.

Dependence on Published Sources: The curated dataset consists of 40 peer-reviewed academic sources derived from an initial pool of 722 records. Although these sources were carefully selected for relevance and academic quality, they remain subject to the inherent limitations of publication cycles and research reporting. For example, even when looking to use academic sources published between 2015 and 2025 (including academic sources from the SLR), emerging vulnerabilities or proprietary industry practices may not yet be reflected in the academic literature. As a result, the framework is based on the best available evidence but may require updating as the IoT security landscape evolves.

Scope and Contextual Boundaries: The framework is explicitly tailored to domestic IoT ecosystems. Its design emphasises household devices, user accessibility and non-technical contexts. While some principles may be applicable to broader IoT domains such as industrial control systems or healthcare, generalisability is limited. Caution should

therefore be exercised when extrapolating findings beyond the domestic setting without further adaptation and validation.

Evaluation Constraints: Evaluation in this work is restricted to conceptual validation, including internal consistency checks and traceability mechanisms. Although this ensures methodological rigour, it does not provide empirical testing of the framework in real-world deployments. Future research should extend this evaluation through expert validation, prototyping or pilot studies in household environments to further confirm the framework's practical applicability.

These limitations do not undermine the validity of the study but define the boundaries within which the findings should be interpreted.

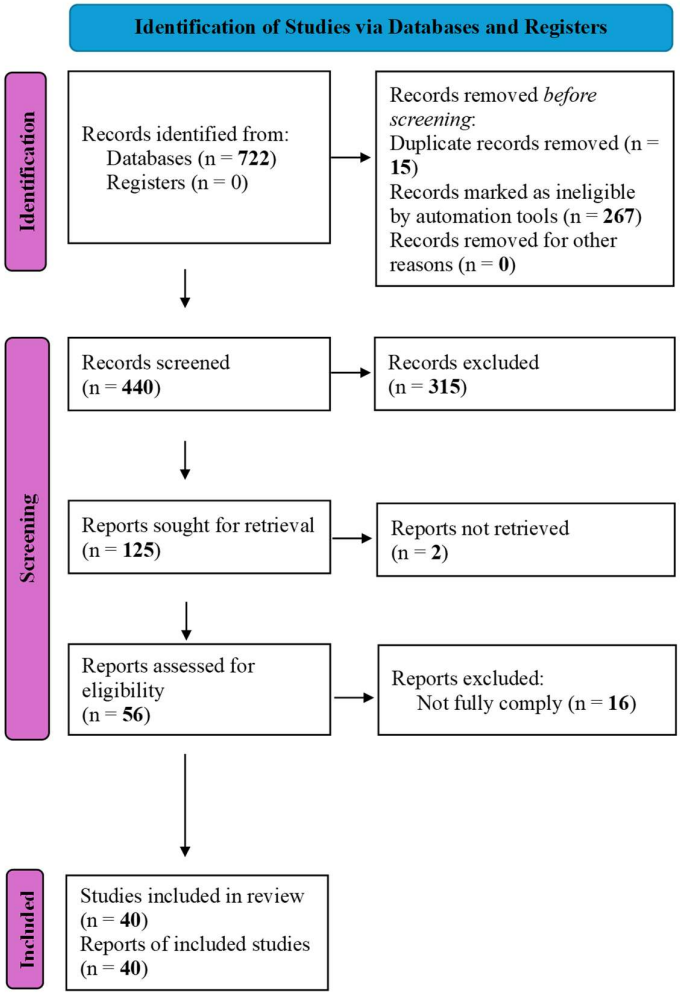

Figure 4. PRISMA 2020 Flow Diagram for study selection (adapted from [40]).

4. Theoretical Framework

This section presents the theoretical framework developed through the Design Science Research Methodology (DSRM) to address the following research question: How can a theoretical framework be developed to enable automated vulnerability scanning and prioritisation for non-technical users in domestic IoT smart home environments?

The framework is the culmination of the Systematic Literature Review (SLR) of 40 peer-reviewed academic sources presented in Section 2 and the methodological process detailed in Section 3. It consolidates vulnerabilities, integrates scanning tools, introduces user-contextual prioritisation and enforces protocol standardisation into a single model designed for the domestic IoT environment.

Consistent with the DSRM, the framework is a theoretical artefact. It provides constructs (vulnerability classes, scanning methods, prioritisation dimensions, protocol security features) and a model (the layered framework) that together address the four structural gaps identified in Section 2.

4.1. Framework Overview

The framework is designed as a layered system with four modules:

- Vulnerability Knowledge Base: Consolidates dispersed vulnerability evidence;
- Automated Scanning Engine: Operationalises detection using integrated methods;
- Context-Aware Prioritisation Module: Ranks vulnerabilities with household relevance;
- Standardisation and Interoperability Layer: Ensures secure integration across IoT protocols.

Although the framework is a theoretical artefact, it is not intended to function as a purely informational or alerting system. Its layered design enables vulnerability prioritisation outputs to inform intermediary enforcement mechanisms, such as home gateways or network controllers, which can automatically constrain the behaviour of compromised devices without requiring direct user intervention.

Figure 5 presents the high-level architecture of the system. This layered architecture reflects bottom-up logic: the Knowledge Base (Table 1) supplies evidence, the Scanning Engine (Table 2) operationalises detection, the Prioritisation Module (Table 3) contextualises results and the Interoperability Layer (Table 4) aligns controls across protocols.

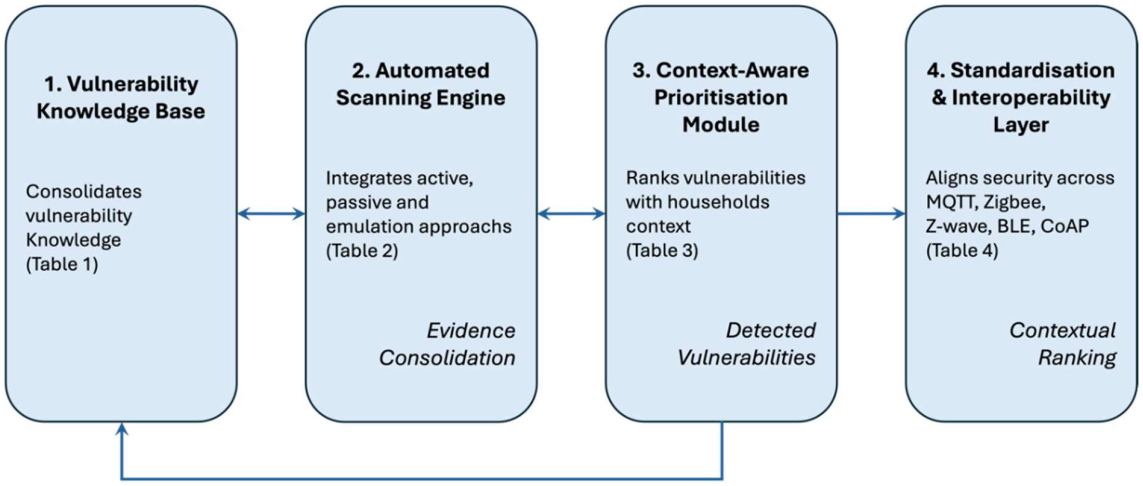

Figure 5. High-level framework architecture (authors' own work, based on synthesis of 40 academic sources).

Although the framework is illustrated as a four-layer sequence, its design is iterative rather than strictly linear. Bidirectional flows between Modules 1 and 2 and between Modules 2 and 3 allow scanning outputs to refine the Knowledge Base and prioritisation insights to adjust scanning processes. In addition, Module 3 can feed back directly into Module 1 if contextual analysis reveals missing categories or overlooked vulnerability evidence. These feedback loops make the framework an iterative refinement cycle, ensuring that the input reaching the Standardisation and Interoperability Layer (Module 4) has been validated and refined through earlier stages.

4.2. Framework Modules

4.2.1. Module 1—Vulnerability Knowledge Base. Gap Addressed: Fragmented and Dispersed Vulnerability Knowledge

As illustrated in Table 1 (Section 2), vulnerabilities in domestic IoT environments are numerous and varied, spanning device-level weaknesses, network exposure, firmware flaws and poor encryption practices. However, the literature reports these vulnerabilities in fragmented ways, often tied to specific devices, protocols or case studies, without a unifying taxonomy accessible to households. This fragmentation was formalised as Gap #1 in Table 5. Foundational taxonomies such as [9] provide the early conceptual structures for trust, privacy and security, reinforcing the rationale for consolidation.

The academic sources dataset shows consistent emphasis on this problem. For instance, ref. [1] highlighted weak authentication practices in household IoT devices, warning that default credentials remain a recurring entry point for attacks. Ref. [2] showed that vulnerabilities are aggravated by protocol-level weaknesses, such as insecure implementations of Zigbee and CoAP. Ref. [4] demonstrated how firmware flaws are exploited in penetration testing of IoT testbeds, while [5] drew attention to vulnerabilities with direct privacy consequences, such as data leakage from smart cameras. Similarly, ref. [3] noted that device misconfigurations often go unnoticed by end-users, leaving IoT ecosystems vulnerable. Ref. [6] identified insecure network services as a persistent exposure vector in domestic environments. Device-level analyses such as [37] on smart cameras highlight household-critical vulnerabilities that remain underrepresented in current classifications. Beyond technical weaknesses, socio-technical risks have been evidenced by [38], who documented digital harms associated with smart home adoption, including surveillance, coercion, and privacy loss.

The Vulnerability Knowledge Base module was therefore designed as the foundational layer of the framework. Its function is to consolidate these dispersed findings into a structured repository, ensuring that vulnerabilities identified across multiple devices, platforms and protocols are normalised into coherent categories. By grounding this consolidation in the evidence gathered from the academic sources used [1–6], the module directly responds to Gap #1 and establishes the baseline for subsequent framework layers.

High-level description: Module 1, shown in Figure 6, should retrieve information from vulnerability databases that are publicly available and format the obtained data according to a predefined template; to help non-technical users understand the meaning of a vulnerability, the proposed template has three classification keys that are easy to understand: device type, security and privacy impacts and required access. The template also includes a description that is not restricted to predefined values and can store information that is traditionally used to describe vulnerabilities, like attack vector and complexity.

The first key, device type, classifies devices into four categories:

- Home security: Groups devices that control household access like smart locks and security cameras;
- Smart appliances: Groups smart appliances that contribute to daily household tasks like washing machines, refrigerators, and vacuum cleaners;
- Environment controls: Groups devices that control a household environment like thermostats and lights;
- Management hubs: This category includes devices that can control other IoT devices; an example is a voice assistant that allows users to increase or decrease temperature.

The second key aims to highlight impact on security and privacy: the possible values are high, medium, and low. For example, if a smart lock has a vulnerability, the impact on security is high, and if a vacuum cleaner that takes household photos and sends them to a

centralised server has a vulnerability, the impact on privacy is high, while a smart light bulb has a low impact.

The third key describes the type of access required by an attacker. The possible categories are physical and network. These are values traditionally used in previous schemes such as CVSS; however, the proposed classification aims to highlight their relevance for a non-technical user.

The three proposed keys aim to help non-technical users understand the real meaning of IoT device vulnerabilities. The following figure describes the tasks that Module 1 must perform to meet this goal.

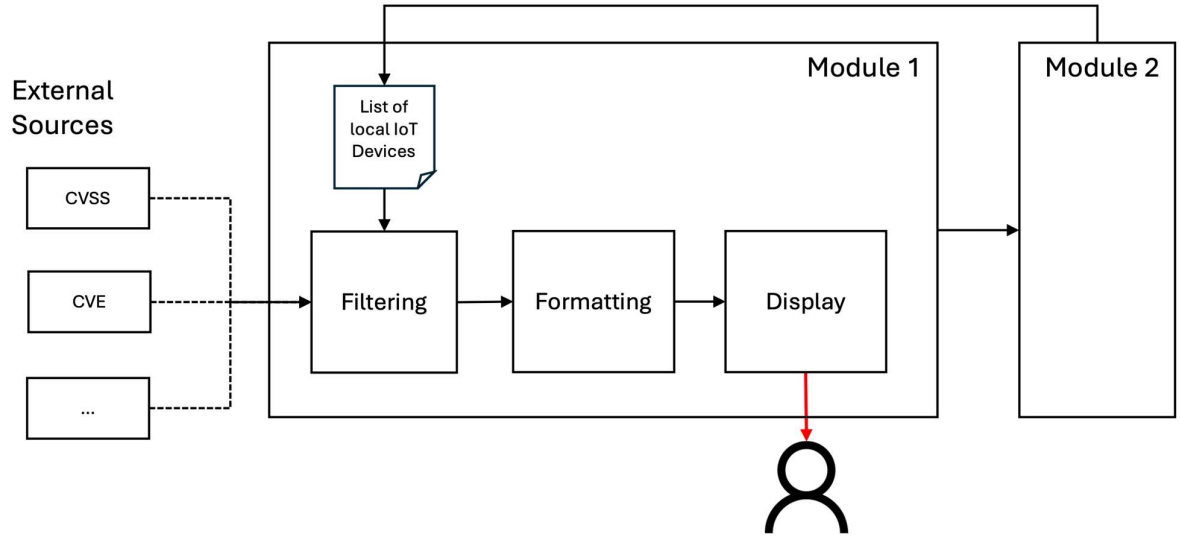

Figure 6. Module 1 retrieves vulnerability information from public databases, selects the ones related to local IoT devices, formats them according to a predefined template, and displays the results to users. The dotted lines indicate external inputs and the red arrows indicate user inputs/outputs.

4.2.2. Module 2—Automated Scanning Engine Gap Addressed: Fragmented and Siloed Scanning Tools

The Automated Scanning Engine is not intended to execute all scanning techniques continuously within the domestic environment. Instead, it is conceptualised as a modular detection layer that prioritises lightweight, traffic-centric and network-based monitoring approaches suitable for household deployment, while more resource-intensive techniques contribute evidence off-path to inform the framework.

As summarised in Table 2, the dataset revealed a diversity of scanning approaches (including penetration testing, network scanning, fuzzing and emulation) yet these approaches remain fragmented. Each tool or framework operates in isolation, addressing a narrow layer of the IoT attack surface. This fragmentation was formalised as Gap #2 in Table 5.

Multiple sources illustrate this problem. Ref. [4] demonstrated penetration testing of IoT testbeds, showing how firmware flaws can be uncovered but without integration into broader vulnerability discovery pipelines. Ref. [7] advanced fuzzing techniques to expose vulnerabilities in IoT networks, but these methods operate separately from device scanning tools. Ref. [6] proposed automated vulnerability discovery at the network level, but the focus remained on service enumeration without integration into higher-layer prioritisation. Ref. [4] explored traffic analysis for anomaly detection, while [1] addressed authentication weaknesses but did not link them to automated detection methods. Techniques such as CoviNet with graph-based temporal dependencies [32] can be leveraged within the passive-monitoring stream when traffic is relayed to cloud services.

Complementary studies have extended scanning toward intrusion detection and anomaly monitoring. Refs. [23,27] showed how machine-learning-based IDS at the network edge can detect botnet activity; ref. [28] introduced READ-IoT for reliable anomaly detection in heterogeneous IoT environments; ref. [29] proposed ADRIoT, an edge-assisted detection framework improving scalability; ref. [30] introduced SARIK, enabling containerised anomaly detection and policy enforcement; ref. [31] advanced hybrid deep-learning models for anomaly detection; and ref. [33] extended IDS analysis to smart city infrastructures, highlighting transferability to domestic contexts.

The Automated Scanning Engine module was designed to unify these siloed approaches. Through DSRM's design stage, the evidence in Table 2 was mapped to three complementary streams, active probing, passive monitoring and emulation/fuzzing. In line with this integration, ref. [18] demonstrate how ML pipelines can be embedded alongside probing and monitoring to elevate detection recall in consumer IoT scenarios. Incorporating IDS and anomaly detection into the passive monitoring stream further strengthens this design, ensuring that the engine captures both pre-deployment vulnerabilities and runtime exploitation attempts. By integrating methods highlighted in [4,6,7], supported by broader insights from [1,4], the module transforms fragmented tool capabilities into a single conceptual process. This design directly addresses Gap #2 and provides the operational layer of the framework. Design choices prioritise low overhead; for instance, ref. [16] shows how optimisation-aware, lightweight AI can retain performance under domestic device and bandwidth constraints.

High-level description: Module 2 is designed to integrate data from external sources and internally generated data. The module must first automatically detect IoT devices in a household, then for each detected device it must identify the type, retrieve related information from external databases, and perform available analysis. The external and internal data is unified based on device identification and automatically synthesised based on an ML-based classifier. The list of detected devices is sent to Module 1 to be used there to retrieve available vulnerability knowledge for this specific set of devices and, since the detection of IoT devices is dynamic, the list changes when IoT devices are added or removed.

The following Figure 7 describes the tasks that Module 2 must perform.

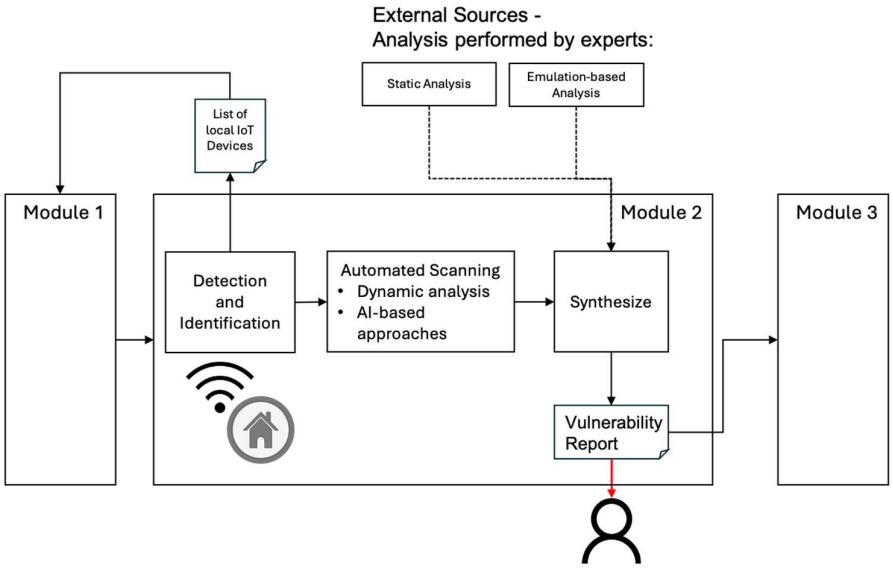

Figure 7. Module 2 detects local IoT devices, scans them looking for known vulnerabilities and synthesises results. The dotted lines indicate external inputs and the red arrows indicate user inputs/outputs.

4.2.3. Module 3—Context-Aware Prioritisation Module Gap Addressed: Prioritisation Frameworks Lacking Household/User Context

As synthesised in Table 3, existing prioritisation frameworks such as CVSS, SAFER, ML-based approaches and dependency-based methods provide mechanisms for scoring vulnerabilities but remain largely technical. They rank based on severity and exploitability but omit contextual household factors such as device function, role in daily life or privacy implications. This was identified as Gap #3 in Table 5, where lack of user-centric focus was a recurring shortcoming across multiple sources.

The academic sources used provide evidence of this limitation. Ref. [4] highlighted how camera and wearable device vulnerabilities pose privacy risks not captured by standard severity scores. Ref. [3] emphasised that misconfigurations in household devices could have different impacts depending on whether the device served a safety-critical role or not. Ref. [6] noted that network exposure measures fail to account for user-facing consequences. Ref. [22] proposed fuzzy-logic and optimisation-based scoring methods that better capture uncertainty in IoT configurations, offering more nuanced prioritisation than CVSS alone. Ref. [7] illustrated that fuzzing results often identify technical weaknesses without guidance on their household relevance. Figure 8 shows the conceptual flow of the Automated Scanning Engine.

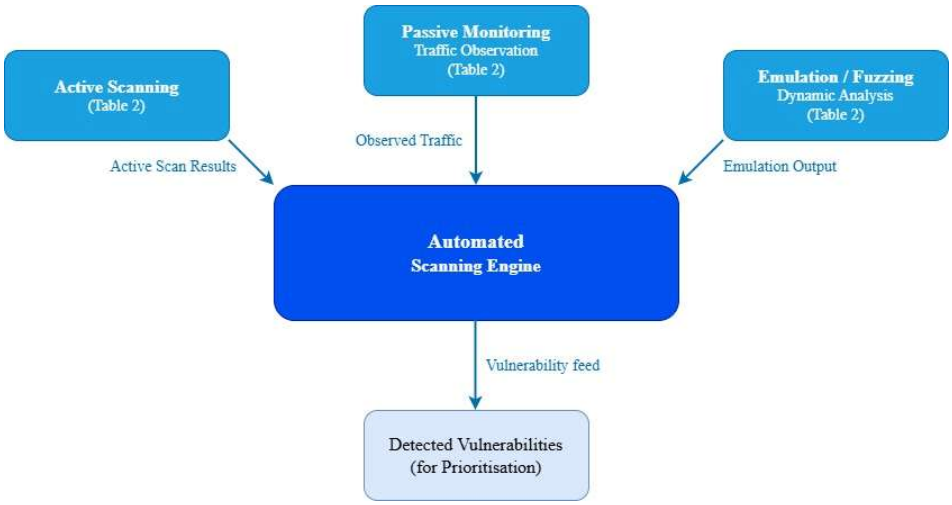

Figure 8. Conceptual flow of the Automated Scanning Engine (authors' own work, based on synthesis in Table 2).

Ref. [15] showed that firmware-level vulnerabilities are frequently catalogued without prioritisation logic, reinforcing the need to integrate lifecycle context into household scoring. Refs. [1,2] similarly observed that vulnerabilities at the protocol level (e.g., weak authentication or insecure Zigbee sessions) carry household impacts that severity-only models overlook. Ref. [24] advanced predictive analytics approaches, demonstrating how vulnerability prioritisation can be informed by models that forecast exploitation likelihood in specific device categories.

The Context-Aware Prioritisation Module therefore extends existing frameworks by embedding contextual dimensions into prioritisation outputs. Through DSRM mapping, evidence from Table 3 was classified into two categories:

- Retain technical dimensions (e.g., severity, exploitability);
- Embed household context (device criticality, privacy impact, protocol exposure).

This transformation, directly informed by the academic dataset, ensures that prioritisation is not only technically valid but also meaningful for non-technical users. Explainability

is critical to user-centric decisions; insights from explainable-AI IDS design [34] can be surfaced in prioritisation outputs to justify rankings to end-users.

High-level description: Module 3 is designed to assess risk by combining technical dimensions, like severity and exploitability, already defined by well-known vulnerability databases, like CVSS, with relevance in the household context. To include relevance, the module (i) asks users to manually evaluate the impact if a particular device does not function in their household; and (ii) receives privacy impact from Module 1.

The traditional formula to compute risk is Risk = Probability of occurrence × Impact.

Probability of occurrence can be computed based on the factors already available from public databases. These factors include attack complexity, privileges required, and user interaction.

CVSS already includes metrics regarding impact: confidentiality impact, integrity impact and availability impact. These values are used to compute impact in the context of a household; however, they are complemented with relevance and impact on privacy. Figure 9 describes the tasks that Module 3 must perform.

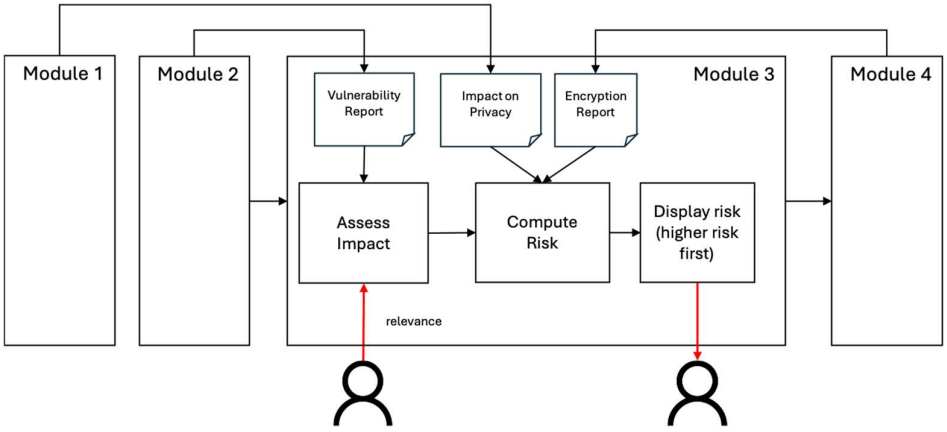

Figure 9. Module 3 assesses impact considering the context in households. The red arrows indicate user inputs/outputs.

4.2.4. Module 4—Standardisation and Interoperability Layer Gap Addressed: Lack of Protocol Standardisation and Inconsistent Adoption

As summarised in Table 4 (Section 2), household IoT ecosystems rely on diverse communication protocols, including MQTT, Zigbee, Z-Wave, BLE and CoAP, which vary significantly in their embedded security features. This heterogeneity results in fragmented protections across devices, complicating household security. The issue was formalised as Gap #4 in Table 5, where inconsistent protocol adoption was shown to be a systemic barrier to secure IoT.

The dataset provides clear evidence for this gap. Ref. [1] highlighted persistent weaknesses in authentication and encryption across common IoT protocols, stressing the risks of insecure default implementations. Ref. [2] offered comparative analysis of Zigbee, CoAP and other protocols, showing how insecure design decisions propagate across devices and layers. Ref. [4] demonstrated that protocol vulnerabilities can be exploited in penetration testing environments, while [3] emphasised that misconfigured devices often fail to enforce even the minimal protections offered by protocols. Ref. [6] further noted that network-level inconsistencies expose households to DoS and enumeration risks. Ref. [35] expanded this evidence by showing how adaptive policy frameworks can dynamically enforce protocol security at the edge, reducing inconsistency across heterogeneous devices.

The Standardisation and Interoperability Layer, as shown in Figure 10, addresses these fragmented protections by providing a central module that ensures consistent enforcement

of security controls across protocols. Through DSRM's design mapping, protocol-level vulnerabilities reported in Table 4 were translated into requirements for standardisation (e.g., encryption, authentication, cross-layer consistency). Evidence from [1,2], reinforced by [3,4,6], provides the foundation for this design. Ref. [36] complement this foundation by analysing fog computing deployments, highlighting unresolved privacy and security trade-offs that arise when protocols lack integrated enforcement. The module thus directly closes Gap #4 by embedding interoperability and standardisation into the framework.

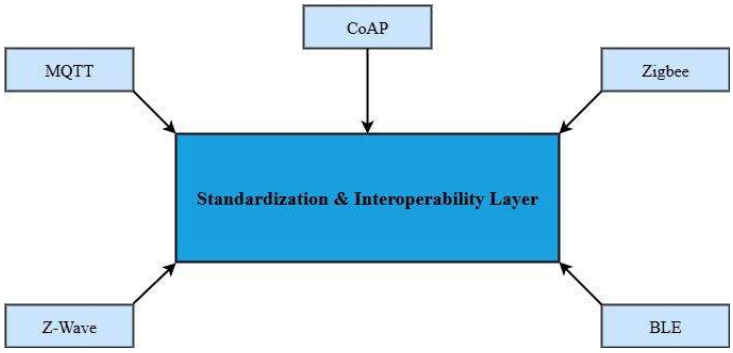

Figure 10. Cross-layer interoperability integration (authors' own work, informed by synthesis in Table 4).

High-level description: Module 4 is designed to detect the IoT protocols that are being used in a real household, and automatically look for information regarding their security features. A local agent must establish, based on the collected data, whether the used protocols support encryption, hash codes, and authentication. Other features like key management and policy management are left as future work as they require agents with complete access to the analysed IoT devices and their software. Additionally, since wireless protocols like MQTT and CoAP can run encryption algorithms, but do not run them by default, the module must dynamically monitor whether the exchanged packets are encrypted.

The following Figure 11 describes the tasks that Module 4 must perform.

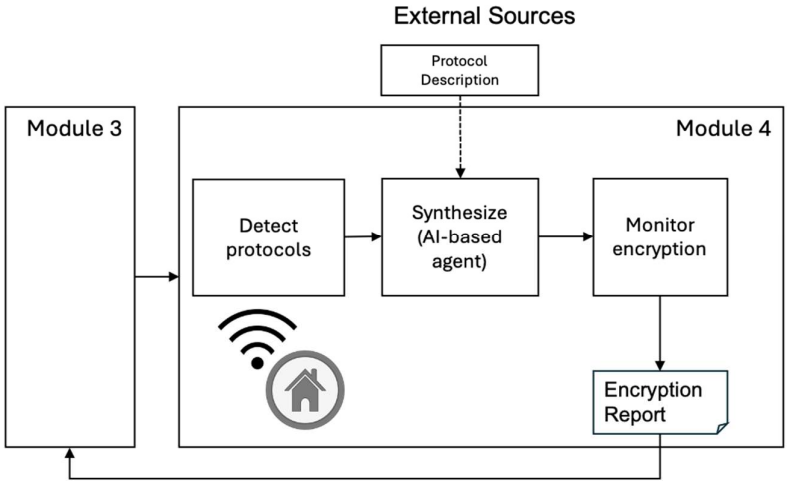

Figure 11. Module 4 identifies the IoT protocols used in a household and asses their security features.

4.3. Framework Evaluation

As explained in Section 3, evaluation of the framework follows DSRM Stage 5. Two complementary mechanisms are applied: internal consistency check and traceability assurance. This ensures that the framework is both conceptually sound and transparently derived from the dataset of 40 academic sources.

Internal Consistency Check: Each module was verified against the specific gap it was designed to address. For example, Module 1, Vulnerability Knowledge Base, responds directly to the fragmented vulnerability reporting documented by [1,2]. Module 2, Automated Scanning Engine, integrates scanning approaches that are otherwise siloed, as illustrated by [4,6,7]. Module 3, Context-Aware Prioritisation, embeds contextual criteria absent in traditional frameworks, as highlighted by [3,4]. Module 4, Standardisation and Interoperability Layer, enforces cross-protocol alignment, directly addressing the heterogeneity documented by [1,2].

This consistency check confirms that the framework has no "orphan gaps", as every shortcoming identified in Section 2 is closed by one design element in Section 4.

Traceability Assurance: The second evaluation mechanism involved ensuring that each module is traceable back to the outputs of Section 2 and the tools described in Section 3:

Module 1 draws directly on vulnerabilities classified in Table 1.

Module 2 integrates scanning methods catalogued in Table 2.

Module 3 extends prioritisation models summarised in Table 3.

Module 4 enforces alignment across protocols compared in Table 4.

This mapping is made explicit in Table 5.

4.4. The Framework

The framework makes two key contributions: First, it provides a theoretical contribution by integrating vulnerability knowledge, scanning approaches, prioritisation dimensions and protocol security features into a single layered model (Figures 5–7, Tables 6–8). This closes the four structural gaps identified in Section 2 and demonstrates the systematic transformation of literature evidence into a design artefact. Second, it offers practical contributions for domestic IoT security. By embedding contextual prioritisation criteria, the framework produces outputs that are meaningful for non-technical users, while also informing vendors and policymakers of the importance of cross-protocol standardisation.

Table 7. Mapping of scanning sources to engine functions (authors' own work, based on SLR dataset).

Source	Scanning Approach	Contribution to Scanning Engine
[4]	Penetration testing on IoT testbeds	Represents active probing of device firmware and services
[7]	Fuzzing for IoT networks	Adds emulation-based detection of edge-case vulnerabilities
[6]	Network-level automated discovery	Contributes service enumeration and network exposure analysis
[5]	Traffic monitoring for anomalies	Provides passive monitoring component
[1]	Authentication testing in consumer IoT	Adds credential-strength checks to scanning logic
[23,27]	ML-driven IDS at the network edge	Extends passive monitoring to detect botnet behaviours
[28]	READ-IoT anomaly detection	Adds reliability-focused anomaly detection to monitoring
[29]	ADRIoT edge-assisted detection	Improves scalability of runtime monitoring
[30]	SARIK container-based IDS	Introduces containerised anomaly detection and policy enforcement
[31]	Hybrid deep-learning anomaly detection	Enhances accuracy through combined DL models
[33]	IDS in smart cities	Demonstrates transferability of IDS methods to household contexts

Table 8. Extended prioritisation criteria for domestic IoT (authors' own work, based on SLR dataset).

Criterion	Evidence Source (Dataset)	Extension Beyond Technical Severity
Technical severity (baseline)	[6]	Retains CVSS-based severity as the foundation
Exploitability	[7]	Refines likelihood of exploitation using fuzzing and emulation results
Device criticality	[3]	Weighs devices differently depending on safety/privacy roles (routers, medical vs. entertainment devices)
Household function	[5]	Adds privacy/safety factors not included in standard severity models
Protocol exposure	[1,2]	Incorporates risks tied to insecure MQTT, Zigbee, CoAP implementations
Privacy impact	[5]	Explicitly accounts for user data leakage as a prioritisation factor
Uncertainty handling	[22]	Introduces fuzzy-logic/optimisation scoring for uncertain IoT contexts
Lifecycle/firmware context	[15]	Adds firmware auditing evidence to prioritisation logic
Predictive exploitation likelihood	[24]	Forecasts risk of exploitation for specific devices

Evaluations confirmed both internal consistency and traceability to the dataset of 40 academic sources, fulfilling the design objectives outlined in Section 3. The evaluation approach is aligned with DSR evaluation principles [41], focusing on internal coherence, transparency of evidence-to-design traceability and relevance to the application context.

Each module is explicitly supported by a cluster of sources:

- Module #1 consolidates vulnerability evidence [1–6,9,37,38];
- Module #2 integrates scanning methods, including IDS/anomaly detection [1,4,6,7,23,27–31,33];
- Module #3 embeds household context into prioritisation [1–4,6,7,15,22,24];
- Module #4 enforces interoperability [1–4,6,35,36].

This comprehensive mapping shows that all 40 sources from Section 2 are systematically embedded within the framework, ensuring transparency, coherence and academic rigour. The next section provides a comprehensive discussion of the framework's theoretical positioning, practical implications, limitations and directions for future research.

5. Discussion

The discussion analyses the extent to which the framework resolves the identified gaps and how it advances the academic and practical understanding of domestic IoT security. In line with the Design Science Research Methodology [8], the emphasis here is on evaluation through internal consistency and traceability. The analysis is therefore structured around the four identified gaps, followed by a cross-module integration and reflection on practical and methodological implications. This approach ensures coherence with the literature base while demonstrating how the framework contributes to the advancement of IoT vulnerability management.

5.1. Addressing Gap #1: Unified Vulnerability Knowledge Base

Gap #1 highlighted the absence of a unified framework for contextualising domestic IoT vulnerabilities. Prior research offered valuable but fragmented contributions. For

example, ref. [9] emphasised early shortcomings in privacy and trust, while [10] catalogued side-channel and hardware-level threats. Ref. [11] demonstrated the risks of privilege escalation in smart hubs and [12] framed risks across multiple IoT layers. Refs. [1,4] provided empirical evidence of device-specific weaknesses, such as insecure firmware and remote exploits in smart cameras, yet their findings were isolated to particular device categories.

By consolidating this dispersed knowledge into Module #1, the framework advances from fragmented taxonomies to a structured Vulnerability Knowledge Base. This repository synthesises device-level, protocol-level and ecosystem-level vulnerabilities into a format that can be interpreted not only by researchers but also by non-technical users. The contribution lies in shifting the focus from isolated vulnerabilities to a systematic classification tailored to household exposure. In doing so, the framework strengthens contextual relevance. Remote exploits in consumer devices are distinguished from vulnerabilities requiring physical access, enabling the prioritisation of risks that matter most to households [1]. Furthermore, by unifying empirical studies and taxonomic surveys, Module #1 bridges the gap between conceptual classifications [9,12] and operational evidence from penetration testing and vulnerability databases [5,11]. This integration is a key theoretical advancement, providing a coherent base upon which subsequent modules (scanning, prioritisation and interoperability) can operate effectively.

5.2. Addressing Gap #2: Fragmented Automated Scanning Approaches

Gap #2 identified the fragmentation of automated scanning approaches, which remain siloed across tools and are unsuitable for direct adoption in household contexts. Traditional enumeration methods such as Nmap and Masscan remain widely used for device discovery and port analysis [6]. These tools provide broad coverage but are limited to surface-level information and do not capture deeper vulnerabilities embedded in firmware or device configurations. Shodan extends enumeration to the internet scale but similarly relies on banner grabbing, restricting precision [14].

Dynamic and emulation-based methods have been proposed to overcome these limitations. Ref. [1] demonstrated how frameworks such as Avatar and Firmadyne enable re-hosting of firmware images to reveal authentication bypasses and hidden services. While powerful, these approaches require specialist expertise and computational resources that prevent their straightforward application in domestic households. The OVER framework developed by [14] extended static analysis to firmware, open-source software and mobile applications, surfacing systemic vulnerabilities such as hard-coded passwords and outdated components, but again did so without integration into user-friendly processes.

Recent research has sought to apply Artificial Intelligence (AI) to reduce blind spots and improve accuracy. Ref. [19] proposed a machine-learning pipeline for predicting missing CVSS metrics and combining them with attack graphs for system-level assessment. Ref. [7] advanced this trajectory through generative fuzzing that created new test cases from live traffic, enabling the discovery of previously undetected vulnerabilities. Ref. [27] contributed a systematic review of machine-learning approaches for IoT botnet detection, highlighting the role of classifiers such as random forests and neural networks in anomaly detection. These contributions show the potential of AI-driven scanning to bridge static and dynamic approaches, but they remain methodologically isolated. They are mentioned in this paper, as it is increasingly clear that AI is playing a significant role in dealing with zero-day threats, and research efforts are being directed at better integrating AI-powered solutions into security tools.

Practical prototypes have also highlighted the feasibility of tailored penetration testing for domestic IoT. Ref. [4] introduced IoTective, a tool capable of performing automated reconnaissance across Wi-Fi, Bluetooth and Zigbee, generating inventories of assets and

reporting potential vulnerabilities. Ref. [6] presented AutoDES, a framework for automated vulnerability discovery and exploitation, capable of producing evidence of exploitability against IoT binaries. While these contributions demonstrate practical feasibility, their outputs are not yet embedded into unified frameworks accessible to households.

Module #2 of the proposed framework responds directly to this fragmentation by consolidating these scanning strategies into a conceptual Automated Scanning Engine. Rather than privileging one method, it layers traditional enumeration, firmware analysis, AI-enhanced techniques and penetration-style fuzzing into a structured process. This integration ensures that households benefit from comprehensive coverage without being required to navigate the complexity of individual toolchains. The theoretical contribution is therefore the articulation of a coherent scanning model that bridges tool silos and grounds vulnerability detection within an evidence-based, layered architecture.

5.3. Addressing Gap #3: Over-Reliance on Technical Severity in Prioritisation

Gap #3 highlighted the trend of existing frameworks to privilege technical severity, often overlooking the household context that determines real-world impact. The Common Vulnerability Scoring System (CVSS) remains the dominant standard, but its focus on exploitability and impact vectors has been criticised for misrepresenting risks in consumer settings [5]. For example, a vulnerability with a high CVSS score in a low-criticality device such as a smart light bulb may be prioritised over a moderate-scored flaw in a security camera, even though the latter presents greater consequences for household privacy and safety.

Alternative approaches have been developed to address these shortcomings. Ref. [5] proposed the SAFER framework, which introduced Current and Future Device Security Risk Indicators (CDSRIs/FDSRIs) to capture both immediate and forecasted risk by incorporating vendor patch cadence and firmware update trends. Ref. [3] extended prioritisation to systemic levels through dependency-based models that quantified how vulnerabilities propagate across interconnected devices. Ref. [25] tailored prioritisation specifically to smart homes through the CRASHED framework, which embedded device roles and exposure into its logic. Similarly, ref. [26] proposed the IoT Security Framework (ISF), emphasising device interdependencies and ecosystem-wide risk rather than isolated technical flaws. Collectively, these contributions signal a shift towards more context-aware models, yet they remain disconnected from widely adopted scoring systems and are not consistently operationalised.

Module #3 of the proposed framework advances this discussion by embedding household context directly into prioritisation. The Context-Aware Prioritisation Module combines technical severity scores with additional criteria such as device criticality, privacy impact and protocol exposure. In doing so, it addresses the limitations of relying solely on CVSS and integrates insights from forecasting approaches [5], dependency-aware models [3] and smart home-focused frameworks [25,26]. The theoretical contribution lies in providing a layered prioritisation logic that balances technical severity with domestic relevance, ensuring that households allocate resources to vulnerabilities that matter most.

5.4. Addressing Gap #4: Weak Protocol Standardisation and Interoperability

Gap #4 focused on the lack of consistent standardisation and the resulting interoperability issues across domestic IoT protocols. Prior research has repeatedly shown that protocol heterogeneity amplifies systemic risk. Early analyses by [2] identified how weaknesses at one layer could cascade across others, emphasising the risks of inconsistent adoption of encryption and authentication. Ref. [10] reinforced these findings by highlighting vulnerabilities in Zigbee, Z-Wave and Bluetooth Low Energy, including key extraction

attacks and susceptibility to replay and sniffing. Ref. [1] provided empirical evidence that consumer devices frequently ship with optional or disabled encryption, undermining even well-established standards such as MQTT with TLS support.

More recent studies demonstrate how these protocol-level weaknesses create exploitable attack chains. Ref. [6] showed that legacy deployments of Zigbee and Z-Wave expose smart homes to risks of enumeration and Denial-of-Service attacks when protections are unevenly enforced. Ref. [4] found that misconfigurations in consumer hubs, such as unpatched Home Assistant deployments and outdated firmware in Zigbee bridges, intensify protocol interoperability flaws. Ref. [3] further argued that dependency chains across perception, network and application layers increase the likelihood of cascading failures, particularly in heterogeneous environments where vendors apply security unevenly.

Module #4 of the proposed framework directly responds to these challenges by establishing a Standardisation and Interoperability Layer. This module harmonises security practices across MQTT, CoAP, Zigbee, Z-Wave and BLE, embedding cross-layer resilience into the framework. It synthesises the academic evidence to ensure that weaknesses in one protocol do not undermine the protections of others, thereby reducing the systemic risks identified by [2]. The theoretical contribution lies in translating fragmented standards and protocol-specific insights into a unified interoperability layer that supports household security. By embedding standardisation as a dedicated module, the framework ensures that protocol heterogeneity is addressed not as an addition but as a core element of domestic IoT security.

Recent work provides concrete illustrations of how such protocol-aware standardisation can be operationalised at intermediary enforcement points. For example, ref. [42] present Wital, a whitelist-based IoT firewall that applies Manufacturer Usage Descriptions (MUD) at domestic gateways to constrain compromised devices while preserving legitimate functionality. This work is cited here as an illustrative exemplar of enforcement enabled by protocol harmonisation, complementing the proposed framework without constituting a required implementation component.

In this framework, protocol harmonisation does not imply unifying or modifying underlying communication protocols. Instead, it operates at the policy and behavioural level by aligning security-relevant expectations across heterogeneous protocols, enabling consistent constraint of anomalous or malicious behaviour without disrupting normal device functionality.

5.5. Cross-Module Integration and Contributions

While each module of the framework responds to a specific gap, their value emerges most clearly when considered in combination. Module #1, the Vulnerability Knowledge Base, provides the foundational repository that enables Modules #2 and #3 to function effectively. Without a consolidated classification of vulnerabilities, automated scanning outputs would remain fragmented, and prioritisation would lack contextual grounding [1,5,9]. Module #2, the Automated Scanning Engine, operationalises this repository by integrating enumeration, firmware analysis, AI-enhanced scanning and fuzzing into a coherent process [4,7,19]. Its layered design ensures that vulnerability evidence is comprehensive and diverse, feeding directly into Module #3.

The Context-Aware Prioritisation Module (Module #3) depends on the outputs of both Modules #1 and #2. By combining technical severity with household context, it advances beyond purely technical scoring models [5,25]. Notably, it links detection with decision-making, ensuring that vulnerabilities identified through scanning are assessed in relation to household relevance. Module #4, the Standardisation and Interoperability Layer, provides the systemic cohesion that enables the preceding modules to operate reliably

across heterogeneous protocols. By addressing weaknesses in MQTT, Zigbee, Z-Wave and BLE [2,6,10], it ensures that vulnerabilities and scanning results are not undermined by protocol-level inconsistencies.

Together, the modules form a layered artefact that addresses the identified gaps. Notably, these modules interact through feedback loops: scanning may highlight gaps in the Knowledge Base, while prioritisation may reveal the need to adjust scanning coverage or classifications. This iterative design reinforces the internal consistency of the framework, ensuring that vulnerability management is not only comprehensive but also self-correcting. This contrasts with prior approaches that remained isolated in scope, whether focused on machine-learning detection [27], systemic risk modelling [3] or firmware analysis [14]. The framework's novelty lies in its ability to integrate these contributions into a cohesive architecture tailored to domestic IoT contexts.

The theoretical contribution therefore extends beyond individual modules. By translating technical vulnerability evidence into household-relevant priorities, the framework supports non-technical users by enabling automated mitigation actions to be executed by intermediary systems, rather than relying on users to manually interpret or respond to high-priority alerts. It positions automated vulnerability scanning and prioritisation as an interconnected process that is transparent, evidence-driven and accessible for non-technical users. This aligns with the principles of Design Science Research Methodology [8], which emphasise artefacts that are both rigorously grounded in the literature and practically relevant.

5.6. Practical Implications for Domestic IoT Users

The practical value of the framework lies in its ability to lower the barriers faced by non-technical households when managing IoT security. Prior research has shown that consumers often underestimate systemic vulnerabilities in their smart home ecosystems, leading to under-preparedness against attacks [5,11]. By consolidating vulnerabilities into a structured knowledge base (Module #1), the framework equips households with an accessible repository of risks that can be understood without specialist expertise. This addresses the persistent problem of fragmented vulnerability reporting, which has historically been confined to expert audiences [1].

The Automated Scanning Engine (Module #2) further contributes to practical impact by providing households with a structured process that unifies traditional tools, firmware analysis, AI-enhanced scanning and Fuzzing methods. While advanced approaches such as Generative Fuzzing [7] or machine-learning-based anomaly detection [23,27] remain technically complex, their conceptual integration into the framework enables translation into lightweight implementations that can be adapted for household devices. Prototypes such as IoTective [4] already demonstrate the feasibility of simplified reconnaissance for end users, suggesting that household-ready scanning tools are achievable. By separating detection from contextual prioritisation, the framework ensures that lightweight scanning outputs can be augmented through household-specific contexts, enabling fast and deployable detection without sacrificing relevance or decision quality.

The Context-Aware Prioritisation Module (Module #3) ensures that consumers are not overwhelmed by technical risk scores. Based on device function, privacy impact and protocol exposure, the framework supports household decision-making that aligns with lived realities. For instance, while a high CVSS vulnerability in a smart bulb may appear urgent, Module #3 would direct attention to vulnerabilities in security cameras or home hubs where household privacy and integrity are more directly at risk [4,25].

The Standardisation and Interoperability Layer (Module #4) provides systemic protection for households by addressing protocol heterogeneity. In practice, this means that

weaknesses in Zigbee or Bluetooth are not left unmitigated but harmonised within a broader security posture [2,6,10]. For end users, this reduces the risk that misconfigured or legacy devices undermine the resilience of the entire household network.

The framework also has broader implications for policymakers, vendors and researchers. Policymakers may use the structure to design regulations that ensure minimum security baselines across protocols, while vendors may adapt the prioritisation logic to provide consumer-friendly vulnerability notifications. Researchers, in turn, can use the framework as a blueprint for empirical validation or as a foundation for developing deployable tools.

Collectively, these implications demonstrate that the framework offers tangible pathways for improving domestic IoT resilience. For households, the framework not only provides prioritised and actionable vulnerability lists but also evolves iteratively. The iterative nature of the design means that assessments are not static but adapt as new vulnerabilities and device contexts emerge, reducing the risk of blind spots over time. This dynamic quality makes the framework more resilient in practice, providing ongoing relevance for non-technical users.

By enabling enforcement at intermediary network points rather than within devices themselves, the framework supports active mitigation while preserving legitimate device behaviour. This approach ensures that non-technical users benefit from automated, protocol-aware constraint of malicious activity without manual configuration or risk of functional disruption.

5.7. Limitations

Although the framework demonstrates clear contributions, it is important to reflect critically on the methodological boundaries of this study. The work adopted a Design Science Research Methodology approach, which emphasises artefact construction based on systematic evidence [8]. Evaluation was therefore conducted conceptually, focusing on internal consistency and evidence-to-framework traceability rather than empirical deployment. This reflects a deliberate methodological choice but introduces limitations.

First, the study relied exclusively on secondary data from 40 peer-reviewed academic sources published between 2015 and 2025. While this dataset was carefully curated through a systematic literature review process, it necessarily excludes insights from grey literature, industry reports and unpublished empirical findings. Consequently, the framework may not capture the full scope of emerging vulnerabilities or proprietary tools used in practice [1,5].

Second, while the framework incorporates an iterative refinement cycle between Modules 1 and 3, this study did not empirically test how such feedback would operate in domestic environments. Although the design illustrates how scanning outputs can enrich the Knowledge Base and how prioritisation can inform both knowledge and scanning processes, these feedback loops remain theoretical. Future work should validate whether such iterative mechanisms can be effectively implemented in practice, either through household trials or automated system integration.

Third, the absence of empirical validation limits the immediate applicability of the framework. Although prior studies have demonstrated proof-of-concept tools such as IoTective [4], automated fuzzing frameworks [7] and dependency-based risk models [3], this work did not implement or test these methods in live smart home environments. As a result, claims about household usability remain theoretical. Future work should extend the framework through prototype development and user evaluation to confirm its practical effectiveness.

Fourth, the generalisability of the framework is constrained by the academic evidence base. While studies such as [23,27] demonstrate the adaptability of machine-learning techniques, most evaluations were conducted on testbeds or institutional networks rather than in domestic households. Similarly, protocol studies by [2,10] focused on broader IoT ecosystems, requiring careful adaptation to the domestic context. These limitations highlight the need for further empirical research that grounds the framework in real-world household deployments.

Beyond individual modules, the discussion demonstrated the integrative value of the framework. Together, the modules form a layered artefact that bridges micro-level vulnerabilities, meso-level scanning processes and macro-level systemic risks, offering a coherent and accessible structure for non-technical households. This discussion confirms that the framework responds directly to the identified research gaps and advances the field of domestic IoT security by integrating vulnerability knowledge, automated scanning, context-aware prioritisation and standardisation. It also establishes an iterative refinement cycle through which knowledge, scanning and prioritisation processes continuously inform one another, ensuring that the framework remains adaptive to emerging household contexts and vulnerabilities.

6. Conclusions and Future Work

Households face significant risks from insecure firmware, weak authentication, fragmented scanning tools and heterogeneous protocols [1,2,5]. In this context, this work identified four structural gaps in the literature: (1) dispersed vulnerability knowledge; (2) fragmented scanning approaches; (3) prioritisation frameworks focused narrowly on technical severity; and (4) weak standardisation across protocols.

Sections 3 and 4 described how these gaps were addressed through the Design Science Research Methodology [8], resulting in a four-module framework: the Vulnerability Knowledge Base, Automated Scanning Engine, Context-Aware Prioritisation Module and Standardisation and Interoperability Layer. Section 5 discussed how these modules collectively advanced both academic knowledge and household practice.

The modules of the framework operate as an integrated system: Module #1 provides evidence, Module #2 generates actionable data, Module #3 translates findings into household-relevant priorities and Module #4 secures the system across protocols. The originality of this framework lies not in isolated responses but in their interdependence, which transforms fragmented insights into a layered artefact for domestic IoT security strengthening. The framework operates as an iterative refinement cycle rather than a one-directional process: outputs from scanning and prioritisation continuously feed back into the Knowledge Base, ensuring that evidence and classifications evolve with emerging vulnerabilities.

6.1. Contributions

This study makes contributions at several levels:

Theoretical contribution: Building on Section 2, the framework integrates vulnerabilities, scanning methods, prioritisation strategies and interoperability challenges into a coherent design tailored for households. It bridges micro-level device risks, mid-level scanning and prioritisation and macro-level protocol resilience [3,27].

Practical contribution: Reinforced in Section 5, the framework lowers barriers for households by contextualising vulnerabilities, supports policymakers in defining baseline protections and guides vendors in developing consumer-friendly vulnerability reporting [5,11].

In relation to the research objectives stated in Section 1, the study defined a layered theoretical framework that responds directly to the gaps revealed by the literature. It did that by (1) designing a Vulnerability Knowledge Base that consolidated dispersed academic evidence into a coherent repository; (2) developing an Automated Scanning Engine that integrates multiple approaches into a single process; (3) designing a Context-Aware Prioritisation Module that embeds household relevance into risk scoring; and (4) proposing a Standardisation and Interoperability Layer that harmonises security features across common IoT protocols.

6.2. Limitations

Dataset constraints: The framework was built on 40 peer-reviewed sources from 2015 to 2025. While rigorous, this excluded grey literature and industry reports that may capture emerging threats [1,5].

Conceptual evaluation: The framework was assessed for internal consistency and traceability rather than tested in live households. Although prototypes such as IoTective [4] and Generative Fuzzing [7] show feasibility, practical validation remains to be carried out in future work. Also, while the framework embeds feedback loops across Modules 1 to 3, these mechanisms remain conceptual and untested. Future empirical work should evaluate how such iteration functions in live household environments.

Generalisability: Much of the reviewed evidence was derived from laboratory or institutional testbeds [23,27], requiring adaptation to domestic contexts. This highlights the need to develop tools for managing IoT devices in household contexts.

These limitations highlight opportunities for future research.

6.3. Future Work

Short-term: Develop a prototype implementation of the framework and conduct household usability studies to test its accessibility for non-technical users [11,25].

Medium-term: Integrate real-time intrusion and anomaly detection at the edge and evaluate interoperability under realistic attack conditions across multiple protocols [6,10,23,27].

Long-term: Expand the knowledge base with industry threat intelligence, disclosure databases and longitudinal data, while exploring adoption pathways for policymakers and vendors [1].

This staged approach ensures that the framework evolves from a conceptual artefact into an empirically validated and widely applicable tool. Each stage should incorporate iterative refinement, enabling the framework to evolve through cycles of testing, evaluation and knowledge-base updating. This ensures that household-level security remains responsive to new device types, vulnerabilities and usage contexts.

To conclude, this work has developed a theoretical framework that addresses fragmented approaches to domestic IoT security. By unifying vulnerabilities, consolidating scanning methods, embedding household context in prioritisation and harmonising protocol protections, the framework responds to the four gaps identified in Section 2 and answers the research question posed in Section 1.

Methodologically, the work demonstrates how Design Science Research Methodology and systematic synthesis can be used to generate a transparent artefact. Practically, it provides households, policymakers and vendors with a structured foundation for strengthening IoT resilience. While limited to conceptual evaluation, the framework sets a foundation for future empirical work that can translate academic insight into deployable solutions. As domestic IoT adoption accelerates, protecting households requires not isolated defences but integrated frameworks. The artefact developed here demonstrates one such pathway,

contributing to academic knowledge and laying the groundwork for real-world security in smart homes. By embedding an iterative refinement cycle within its design, the framework also establishes a mechanism for continuous learning and adaptation, aligning with Design Science Research Methodology principles of artefact evolution.

Author Contributions: Conceptualisation, D.F.R.B. and J.A.G.; methodology, D.F.R.B.; validation, D.F.R.B.; formal analysis, D.F.R.B.; investigation, D.F.R.B.; data curation, D.F.R.B.; writing—original draft preparation, D.F.R.B.; writing—review and editing, D.F.R.B., J.A.G. and S.J.R.; visualisation, D.F.R.B. and S.J.R.; supervision, J.A.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

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

References

1. Costa, L.; Barros, J.; Tavares, M. Vulnerabilities in IoT Devices for Smart Home Environment. In Proceedings of the 5th International Conference on Information Systems Security and Privacy, Prague, Czech Republic, 23–25 February 2019; pp. 615–622. [CrossRef]
2. Jing, Q.; Vasilakos, A.V.; Wan, J.; Lu, J.; Qiu, D. Security of the Internet of Things: Perspectives and challenges. *Wirel. Netw.* **2014**, *20*, 2481–2501. [CrossRef]
3. Radanliev, P.; De Roure, D.; Maple, C.; Nurse, J.R.C.; Nicolescu, R.; Ani, U. AI security and cyber risk in IoT systems. *Front. Big Data* **2024**, *7*, 1402745. [CrossRef] [PubMed]
4. Nordnes, K.; Lin, J.-C.; Lee, M.-C.; Chang, V. IoTective: Automated Penetration Testing for Smart Home Environments. In Proceedings of the 9th International Conference on Internet of Things, Big Data and Security, Angers, France, 28–30 April 2024; pp. 29–39. [CrossRef]
5. Oser, P.; van der Heijden, R.W.; Lüders, S.; Kargl, F. Risk Prediction of IoT Devices Based on Vulnerability Analysis. *ACM Trans. Priv. Secur.* **2022**, *25*, 1–36. [CrossRef]
6. Wang, Z.; Zhang, Y.; Tian, Z.; Ruan, Q.; Liu, T.; Wang, H.; Liu, Z.; Lin, J.; Fang, B.; Shi, W. Automated Vulnerability Discovery and Exploitation in the Internet of Things. *Sensors* **2019**, *19*, 3362. [CrossRef]
7. Masud, M.T.; Koroniotis, N.; Keshk, M.; Turnbull, B.; Kermanshahi, S.K.; Moustafa, N. Generative fuzzer-driven vulnerability detection in the Internet of Things networks. *Appl. Soft Comput.* **2025**, *174*, 112973. [CrossRef]
8. Peffers, K.; Tuunanen, T.; Rothenberger, M.A.; Chatterjee, S. A Design Science Research Methodology for Information Systems Research. *J. Manag. Inf. Syst.* **2007**, *24*, 45–77. [CrossRef]
9. Sicari, S.; Rizzardi, A.; Grieco, L.A.; Coen-Porisini, A. Security, privacy and trust in Internet of Things: The road ahead. *Comput. Netw.* **2015**, *76*, 146–164. [CrossRef]
10. Mosenia, A.; Jha, N.K. A Comprehensive Study of Security of Internet-of-Things. *IEEE Trans. Emerg. Top. Comput.* **2017**, *5*, 586–602. [CrossRef]
11. Fernandes, E.; Jung, J.; Prakash, A. Security Analysis of Emerging Smart Home Applications. In Proceedings of the 2016 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22–26 May 2016; pp. 636–654. [CrossRef]
12. Li, S.; Tryfonas, T.; Li, H. The Internet of Things: A security point of view. *Internet Res.* **2016**, *26*, 337–359. [CrossRef]
13. Amro, A. IoT Vulnerability Scanning: A State of the Art. In *Computer Security*; Katsikas, S., Cuppens, F., Cuppens, N., Lambrinoudakis, C., Kalloniatis, C., Mylopoulos, J., Anto, A., Gritzalis, S., Meng, W., Furnell, S., Eds.; Springer International Publishing: Cham, Switzerland, 2020; Volume 12501, pp. 84–99. [CrossRef]
14. Sachidananda, V.; Bhairav, S.; Elovici, Y. OVER: Overhauling vulnerability detection for IoT through an adaptable and automated static analysis framework. In Proceedings of the 35th Annual ACM Symposium on Applied Computing, Brno, Czech Republic, 30 March–3 April 2020; pp. 729–738. [CrossRef]
15. Bakhshi, T.; Ghita, B.; Kuzminykh, I. A Review of IoT Firmware Vulnerabilities and Auditing Techniques. *Sensors* **2024**, *24*, 708. [CrossRef]
16. Chaganti, K.C. A Scalable, Lightweight AI-Driven Security Framework for IoT Ecosystems: Optimization and Game Theory Approaches. *IEEE Access* **2025**, *13*, 72235–72247. [CrossRef]
17. Akhilesh, R.; Bills, O.; Chilamkurti, N.; Chowdhury, M.J.M. Automated Penetration Testing Framework for Smart-Home-Based IoT Devices. *Future Internet* **2022**, *14*, 276. [CrossRef]

18. Arabelli, R.; Buradkar, M.; Lakshmaji, K.; Dube, A.P.; Shiba, C.M.; Geetha, B.T. Machine Learning-Based Cybersecurity Framework for IoT Devices. In Proceedings of the 2024 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI), Chennai, India, 9–10 May 2024; pp. 1–6. [CrossRef]
19. Duan, X.; Ge, M.; Le, T.H.M.; Ullah, F.; Gao, S.; Lu, X.; Babar, M.A. Automated Security Assessment for the Internet of Things. In Proceedings of the 2021 IEEE 26th Pacific Rim International Symposium on Dependable Computing (PRDC), Perth, Australia, 1–4 December 2021; pp. 47–56. [CrossRef]
20. Aung, Y.L.; Christian, I.; Dong, Y.; Ye, X.; Chattopadhyay, S.; Zhou, J. Generative AI for Internet of Things Security: Challenges and Opportunities (Version 1). *arXiv* **2025**, arXiv:2502.08886. [CrossRef]
21. Nazzal, B.; Zaid, A.A.; Alalfi, M.H.; Valani, A. Vulnerability classification of consumer-based IoT software. In Proceedings of the 4th International Workshop on Software Engineering Research and Practice for the IoT, New York, NY, USA, 3 February 2022; pp. 17–24. [CrossRef]
22. Mashaleh, A.S.; Ibrahim, N.F.B.; Alauthman, M.; Almseidin, M.; Gawanmeh, A. IoT Smart Devices Risk Assessment Model Using Fuzzy Logic and PSO. *Comput. Mater. Contin.* **2024**, *78*, 2245–2267. [CrossRef]
23. Kumar, A.; Shridhar, M.; Swaminathan, S.; Lim, T.J. Machine learning-based early detection of IoT botnets using network-edge traffic. *Comput. Secur.* **2022**, *117*, 102693. [CrossRef]
24. Kalaria, R.; Kayes, A.; Rahayu, W.; Pardede, E.; Salehi, S.A. IoTPredictor: A security framework for predicting IoT device behaviours and detecting malicious devices against cyber attacks. *Comput. Secur.* **2024**, *146*, 104037. [CrossRef]
25. Paparis, G.; Zarras, A.; Farao, A.; Xenakis, C. CRASHED: Cyber risk assessment for smart home electronic devices. *J. Inf. Secur. Appl.* **2025**, *91*, 104054. [CrossRef]
26. Bhardwaj, A.; Kaushik, K.; Alshehri, M.; Mohamed, A.A.-B.; Keshta, I. ISF: Security Analysis and Assessment of Smart Home IoT-based Firmware. In *ACM Transactions on Sensor Networks*; Association for Computing Machinery: New York, NY, USA, 2023. [CrossRef]
27. Nazir, A.; He, J.; Zhu, N.; Wajahat, A.; Ma, X.; Ullah, F.; Qureshi, S.; Pathan, M.S. Advancing IoT security: A systematic review of machine learning approaches for the detection of IoT botnets. *J. King Saud Univ. Comput. Inf. Sci.* **2023**, *35*, 101820. [CrossRef]
28. Yahyaoui, A.; Abdellatif, T.; Yangui, S.; Attia, R. READ-IoT: Reliable Event and Anomaly Detection Framework for the Internet of Things. *IEEE Access* **2021**, *9*, 24168–24186. [CrossRef]
29. Li, R.; Li, Q.; Zhou, J.; Jiang, Y. ADRIoT: An Edge-Assisted Anomaly Detection Framework Against IoT-Based Network Attacks. *IEEE Internet Things J.* **2022**, *9*, 10576–10587. [CrossRef]
30. dos Santos, J.G.; Filho, G.P.R.; Meneguette, R.I.; Bonacin, R.; Pessin, G.; Gonçalves, V.P. Enhancing IoT device security in Kubernetes: An approach adopted for network policies and the SARIK framework. *Future Gener. Comput. Syst.* **2025**, *162*, 107485. [CrossRef]
31. Baswaraj, D.; Rahman, A.; Pandey, D.; Bhargavi, T.; Ismoilov, M.; Deepthi, N. A Hybrid Deep Learning Framework for IoT Security Enhancement and Anomaly Detection. In Proceedings of the 2025 3rd International Conference on Integrated Circuits and Communication Systems (ICICACS), Raichur, India, 21–22 February 2025; pp. 1–6. [CrossRef]
32. Begum, M.B.; Yogeshwaran, A.; Nagarajan, N.; Rajalakshmi, P. Dynamic network security leveraging efficient CoviNet with granger causality-inspired graph neural networks for data compression in cloud IoT Devices. *Knowl.-Based Syst.* **2025**, *309*, 112859. [CrossRef]
33. Bhardwaj, A.; Bharany, S.; Abulfaraj, A.W.; Ibrahim, A.O.; Nagmeldin, W. Fortifying home IoT security: A framework for comprehensive examination of vulnerabilities and intrusion detection strategies for smart cities. *Egypt. Inform. J.* **2024**, *25*, 100443. [CrossRef]
34. Dixit, M.; Siby, S.M.; Jenefa, J.; Vetriveeran, D.; Sambandam, R.K.; Vinodha, D. Theoretical Framework for Integrating IoT and Explainable AI in a Smart Home Intrusion Detection System. In Proceedings of the 2024 IEEE International Conference on Contemporary Computing and Communications (InC4), Bangalore, India, 15–16 March 2024; Volume 1, pp. 1–5. [CrossRef]
35. Halgamuge, M.N.; Niyato, D. Adaptive edge security framework for dynamic IoT security policies in diverse environments. *Comput. Secur.* **2025**, *148*, 104128. [CrossRef]
36. Alrawais, A.; Alhothaily, A.; Hu, C.; Cheng, X. Fog Computing for the Internet of Things: Security and Privacy Issues. *IEEE Internet Comput.* **2017**, *21*, 34–42. [CrossRef]
37. Buil-Gil, D.; Kemp, S.; Kuenzel, S.; Coventry, L.; Zakhary, S.; Tilley, D.; Nicholson, J. The digital harms of smart home devices: A systematic literature review. *Comput. Hum. Behav.* **2023**, *145*, 107770. [CrossRef]
38. Bhardwaj, A.; Bharany, S.; Osman Ibrahim, A.; Almogren, A.; Ur Rehman, A.; Hamam, H. Unmasking vulnerabilities by a pioneering approach to securing smart IoT cameras through threat surface analysis and dynamic metrics. *Egypt. Inform. J.* **2024**, *27*, 100513. [CrossRef]
39. Gregor, S.; Hevner, A.R. Positioning and Presenting Design Science Research for Maximum Impact1. *MIS Q.* **2013**, *37*, 337–355. [CrossRef]

40. Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, L.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; et al. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ* **2021**, *372*, n71. [CrossRef]
41. Peffers, K.; Rothenberger, M.; Tuunanen, T.; Vaezi, R. Design Science Research Evaluation. In *Design Science Research in Information Systems. Advances in Theory and Practice*; Peffers, K., Rothenberger, M., Kuechler, B., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7286, pp. 398–410. [CrossRef]
42. Kim, H.; Toh, W.X.; Hao, L.; Schulzrinne, H. Wital: A Whitelist-Based IoT Firewall for Mitigating Device Exploitation. In Proceedings of the 2024 IEEE International Performance, Computing, and Communications Conference (IPCCC), Orlando, FL, USA, 22–24 November 2024; pp. 1–2. [CrossRef]