TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies
Department of Software Science

Mert Meiessaar

232684IVCM

How to Protect Small Businesses Using Public Cyber Threat Intelligence

Master´s Thesis

Supervisor: Toomas Lepik, MSc

Co-Supervisor: Hillar Põldmaa, MSc

Tallinn 2025

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Tarkvarateaduste instituut

Mert Meiessaar

232684IVCM

Kuidas kaitsta väikeettevõtteid küberohtude eest kasutades avalikku küberohuteadmust

Magistritöö

Juhendaja: Toomas Lepik, MSc

Kaasjuhendaja: Hillar Põldmaa, MSc

Tallinn 2025

List of Abbreviations and Terms

AI	Artificial Intelligence
API	Application Programming Interface
ARDS	Anti-Ransomware Defence System
ASMAS	Adaptive Security Maturity Assessment System
BL	Blocklist
CLI	Command-Line Interface
CPU	Central Processing Unit
CTI	Cyber Threat Intelligence
CSV	Comma-Separated Values
DB	Database
DNS	Domain Name System
DoH	DNS over HTTPS
DoT	DNS over TLS
EU	European Union
FTL	Faster Than Light (Pi-hole’s DNS resolver engine)
GDPR	General Data Protection Regulation
GUI	Graphical User Interface
HTTPS	Hyper Text Transfer Protocol Secure
IOC / IoC	Indicator of Compromise
IT	Information Technology

JSON	JavaScript Object Notation
LAN	Local Area Network
MISP	Malware Information Sharing Platform
ML	Machine Learning
MSP	Managed Service Provider
NIS2	Network and Information Security Directive 2
NGFW	Next-Generation Firewall
NLP	Natural Language Processing
OS	Operating System
OSINT	Open-Source Intelligence
RAM	Random Access Memory
RaaS	Ransomware-as-a-Service
REST API	Representational State Transfer Application Programming Interface
RPZ	Response Policy Zone
SIEM	Security Information and Event Management
SME	Small and Medium-Sized Enterprise
SSH	Secure Shell
STIX	Structured Threat Information eXpression
TAXII	Trusted Automated eXchange of Indicator Information
URL	Uniform Resource Locator
VM	Virtual Machine

VPN	Virtual Private Network

Author’s Declaration of Originality

I hereby certify that I am the sole author of this thesis. All the used materials, the literature and the work of others have been referenced. This thesis has not been presented for examination anywhere else.

Author: Mert Meiessaar

2025-05-18

Abstract

Small businesses are increasingly targeted by cyber threats but often lack the technical capacity and financial resources to deploy enterprise-level security solutions. This thesis presents a practical, cost-effective approach to enhancing SME cybersecurity by integrating public Cyber Threat Intelligence (CTI) feeds from MISP with DNS-level filtering using Pi-hole. A fully functional prototype was developed, combining automated CTI ingestion, threat indicator transformation, and domain-level blocking—achieved through open-source tools and lightweight infrastructure suitable for small-scale deployments.

The system was validated through functional testing, real-world use, and stress scenarios simulating over 10,000 DNS requests. Performance results showed minimal resource usage, no service degradation, and successful enforcement of threat-based filtering, even on a 1 vCPU / 8 GB RAM virtual machine. Automation scripts and installation guides enabled non-technical users to complete setup in under one hour, highlighting the solution's usability and accessibility for SMEs.

This research does not aim to validate the quality of public OSINT feeds but instead proves the feasibility of deploying them operationally in a DNS-based defence pipeline. It contributes a novel, open-source framework for SME cybersecurity and offers practical recommendations for scaling, maintenance, and future improvements. The findings demonstrate that public CTI, when combined with DNS filtering and automation, can form a sustainable and effective defensive layer for small organisations with limited cybersecurity resources.

The thesis is in English and contains 67 pages of text, 8 chapters, 4 figures, 2 tables.

Annotatsioon

Kuidas kaitsta väikeettevõtteid küberohtude eest kasutades avalikku küberohuteadmust

Väikeettevõtted on üha enam küberohtude sihtmärgiks, kuid sageli puuduvad neil tehnilised võimekused ja rahalised vahendid ettevõtte tasemel turbelahenduste rakendamiseks. Käesolev lõputöö esitab praktilise ja kulutõhusa lahenduse SME-de küberturvalisuse parandamiseks, integreerides avalikud küberohuteadmuse (CTI) voogudest MISP-ist DNS-tasandi filtreerimisega Pi-hole'i kaudu. Töös töötati välja täisfunktsionaalne prototüüp, mis ühendab automatiseeritud CTI allalaadimise, ohunäidikute töötlemise ja domeenipõhise blokeerimise, kasutades avatud lähtekoodiga tööriistu ning kerget infrastruktuuri, mis sobib väiksemahulisteks juurutusteks.

Süsteemi valideeriti funktsionaalse testimise, reaalse kasutuskogemuse ning stressitingimuste kaudu, kus simuleeriti enam kui 10 000 DNS-päringut. Jõudlustulemused näitasid väikest ressursikasutust, teenuse stabiilsust ning edukat ohuandmete filtreerimist ka piiratud võimekusega keskkonnas (1 vCPU / 8 GB RAM virtuaalmasin). Automatiseeritud skriptid ja juhendid võimaldasid mittetehnilistel kasutajatel süsteemi seadistada vähem kui ühe tunniga, rõhutades lahenduse kasutajasõbralikkust ja juurdepääsetavust SME-de jaoks.

Käesoleva uurimistöö eesmärk ei olnud avalike OSINT voogude kvaliteedi hindamine, vaid nende operatiivne rakendamine DNS-põhisesse kaitselahendusse. Töö annab panuse uuenduslikku, avatud lähtekoodiga raamistikku, mis on suunatud just väikese ja keskmise suurusega ettevõtetele. Samuti pakub töö praktilisi soovitusi skaleeritavuse, hoolduse ja edasise arendamise kohta. Lõppkokkuvõttes kinnitavad tulemused, et avalik CTI koos DNS-filtreerimise ja automatiseerimisega võib kujutada endast jätkusuutlikku ja tõhusat kaitsekihti väikestele organisatsioonidele, kellel on piiratud ressursid küberturvalisuse tagamiseks.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 67 leheküljel, 8 peatükki, 4 joonist, 2 tabelit.

Table of Contents

List of Abbreviations and Terms..3

Author's Declaration of Originality 6

Abstract...7

Annotatsioon...8

Table of Contents..9

List of Figures...14

List of Tables 15

1 Introduction 16

1.1 Background and Context 16

1.1.1 Growth of Cyber Threats and Impact on Small Businesses 16

1.1.2 Importance of Cost-Effective Cyber Defences.. 17

1.2 Significance of the Study.. 18

1.2.1 Bridging the Research Gap for SME-Specific Solutions 18

1.2.2 Potential to Democratise Cybersecurity 18

2 Research Problem..20

2.1 Problem Statement.. 20

2.2 Research Questions and Hypothesis.. 20

2.2.1 Primary Research Question 20

2.2.2 Secondary Research Questions... 21

2.3.3 Hypothesis: MISP integration Pi-hole as a Significant Improvement............... 22

2.3 Aim and Objectives 22

2.3.1 Aim: An Affordable, Effective Security System for SMEs 22

2.3.2 Objectives: Prototype Development, Evaluation, Ease of Use 22

2.4 Scope 23

2.4.1 Defining SME Characteristics (e.g., Size, Budget, Technical Expertise).......... 23

2.4.2 Focus on MISP and Pi-Hole 23

2.5 Key Assumptions 24

2.5.1 Reliability of Public CTI Feeds 24

2.5.2 Baseline Technical Proficiency of End-Users 24

2.6 Limitations 24

2.6.1 Hardware Constraints of Raspberry Pi 24

2.6.2 Accuracy and Timeliness of Threat Intelligence 24

3 Literature Review 25

3.1 Overview of Cybersecurity for Small Businesses 25

3.1.1 Common Threat Vectors (Phishing, Malware, Ransomware) 25

3.1.2 Financial and Human Resource Constraints 27

3.2 Public Cyber Threat Intelligence (CTI) 27

3.2.1 Definition and Evolution of CTI 27

3.2.2 Sources of CTI (Government, Community, Open-Source) 28

3.3 DNS Filtering in Cybersecurity 29

3.3.1 DNS Fundamentals and Role in Threat Blocking 29

3.3.2 Comparison of Existing DNS Filtering Solutions 29

3.4 MISP (Malware Information Sharing Platform) 32

3.4.1 Architecture and Key Features 32

3.4.2 Community-Driven Approach and Integration Potential 33

3.4.3 Validation of IoC 34

3.5 Pi-hole 34

3.5.1 Technical Overview and Deployment Scenarios 34

3.5.2 Comparison with Alternative DNS Blockers 35

3.5.3 Relevance to SMEs (Ease of Use, Low Hardware Requirements) 36

3.6 Existing Integrations and Research 37

3.6.1 Prior Studies on CTI-Driven DNS Filtering 37

3.6.2 Gaps in Automation and Affordability 37

3.7 Summary of Literature Findings 38

3.7.1 Synthesis of Key Themes (Threat Intelligence, DNS Blocking, SME Barriers)38

3.7.2 Positioning this Thesis in the Wider Research Landscape 39

3.8 Research Gap 39

3.8.1 Current Lack of Integrated MISP + Pi-Hole Studies.. 39

3.8.2 Need for Automated, Real-Time Threat Blocking for SMEs 40

3.9 Novelty of the Study.. 40

3.9.1 Unique Public CTI and DNS Filtering Combination 40

3.9.2 Emphasis on Automation and Affordability 40

4 Research Methodology 41

4.1 Overall Research Approach 41

4.1.1 Justification for a Prototype and Experimental Evaluation 41

4.1.2 Qualitative and Quantitative Dimensions... 41

4.2 Data Collection 42

4.2.1 Gathering Threat Intelligence from MISP 42

4.2.2 Logging and Monitoring Data in Pi-Hole 42

4.2.3 Additional Metrics (Performance, User Feedback) 43

4.3 System Architecture and Workflow 45

4.3.1 Conceptual Diagram of MISP-to-Pi-Hole Integration...................................... 45

4.3.2 Data Flow for Threat Indicators 47

4.4 Validation and Testing Strategy 48

4.4.1 Defining Key Metrics (Blocking Rate, False Positives, Resource Usage)........ 48

4.4.2 Test Scenarios (Lab Environment vs Real-World Traffic) 49

4.4.3 Methods for Collecting Usability Feedback 50

4.5 Ethical Considerations 52

4.5.1 Responsible Use of Threat Data 52

4.5.2 Privacy and Confidentiality Concerns 53

4.6 Limitations of Methodology 54

4.6.1 Laboratory Constraints vs Live Environments 54

3.6.2 Potential Biases in User Feedback 54

5 Implementation 56

5.1 Lab Environment Setup 56

5.1.1 Virtual Machine and Resource Allocation 56

5.1.2 Operating System Installation (Ubuntu 24.04 LTS) 56

5.1.3 Static Network Configuration 57

5.2 Automated Installation with Custom Script 58

5.2.1 Overview of install.sh 58

5.2.2 Script Logic and Execution Flow 59

5.2.3 Custom Configuration Applied 60

5.3 Threat Intelligence Integration Workflow 61

5.3.1 MISP Feed Management and Filtering 61

5.3.2 IOC Export and Pi-hole Consumption 62

5.4 System Validation and Performance Metrics 63

5.4.1 DNS Performance Testing Methodology 63

5.4.2 Benchmark Script Buildup 64

6 Results 65

6.1 Performance Analysis 65

6.1.1 CPU and Memory Usage on the Virtual Machine 65

6.1.2 Network Latency and Throughput 66

6.2 Threat Blocking Effectiveness 67

6.2.1 Block Rate of Malicious Domains 67

6.2.2 False Positives and Whitelisting Adjustments 69

6.2.3 Comparison with Baseline (No CTI Integration) 70

6.3 System Reliability and Scalability 71

6.3.1 Effectiveness Under Increased Network Load 71
6.3.2 Potential Bottlenecks for Larger SMEs 72
6.4 Usability Feedback 72
6.4.1 Setup and Configuration Experience for Non-Technical Users 72
6.4.2 Maintenance and Update Considerations 73
7 Discussion 75
7.1 Principal Findings 75
7.1.1 Alignment with the Primary Research Question 75
7.1.2 Validation of Hypothesis (MISP - Pi-hole integration Efficacy) 75
7.2 Strengths and Weaknesses of the Proposed Solution 76
7.3 Implications for Small Businesses 77
7.4 Considerations for Larger Networks 77
7.5 Alignment with Existing Literature 78
7.6 Future work 79
8 Conclusion 81
8.1 Summary of Key Contributions 81
8.1.1 Achievements Relative to the Stated Objectives 81
8.1.2 Significance for the SME Community and Cybersecurity Field 81
References 83
Appendix 1 – Non-exclusive Licence for Reproduction and Publication of a Graduation Thesis 88

List of Figures

Figure 1. Enabling and pulling MISP open-source CTI feeds for IoC-s 42

Figure 2. Pi-hole default dashboard with malicious domains form MISP feeds 43

Figure 3. Conceptual Architecture of MISP-to-Pi-hole Integration 46

Figure 4. Pi-hole upstream DNS settings 64

List of Tables

Table 1. DNS test for 5 min 6 times loop 67

Table 2. Six random runs with StevenBlack list domains with and without MISP malicious domains 69

1 Introduction

1.1 Background and Context

1.1.1 Growth of Cyber Threats and Impact on Small Businesses

Over the last five years, Small and Medium-sized Enterprises (SMEs) have faced a marked increase in cyber threats, both in volume and sophistication. Multiple studies emphasize that SMEs are now frequent and intentional targets of cyberattacks, largely due to their weaker cybersecurity defences, limited technical expertise, and financial constraints when compared to larger enterprises.

Rombaldo et al. [1] report a significant rise in targeted attacks on SMEs, citing evidence that as early as 2019, 58% of reported cyberattacks were directed at these smaller organizations. This trend reflects a shift in attacker focus, where the lower barrier to exploit SMEs—combined with their increasing digital adoption—makes them attractive entry points, especially for financially motivated and opportunistic threat actors.

The nature of these threats continues to evolve. Ambreen et al. [2] note that ransomware, phishing, malware, zero-day attacks, insider threats, and supply chain compromises have become increasingly prevalent in SME environments. These developments are linked not only to general advances in cybercriminal toolkits, such as ransomware-as-a-service, but also to SMEs' specific vulnerabilities, including under-resourced IT departments, inconsistent cyber hygiene practices, and low cyber-risk awareness among staff.

The consequences for breached SMEs are substantial. Sonkar [3], through a quantitative analysis of cross-sectoral data, finds that cyber incidents can lead to extensive financial and operational losses for SMEs, with both direct costs (e.g., system recovery, legal liabilities) and indirect impacts (e.g., reputational harm, loss of customer trust, and business interruption). Almoaigel and Abuabid [4] concretely reference financial losses in the range of £65,000 per incident in a European context, further underscoring the severity of outcomes.

The concern extends beyond financial hardship. Perozzo et al. [5] observe that SMEs often lack the capacity to prepare for or respond effectively to cyber incidents. This includes absent or underdeveloped response protocols and insufficient staff training. As a result, recovery timelines are frequently longer and more costly than those experienced by larger organizations. Companion studies highlight that many SMEs misjudge their risk exposure; Kandpal et al. [6] describe a pervasive belief among SME stakeholders that their small size renders them undesirable targets—a perception repeatedly identified as dangerously inaccurate.

Compounding these challenges is the growing interconnectivity between SMEs and larger supply chain ecosystems. Bamidele et al. [7] identify supply chain attacks—where SMEs are used as access points to reach larger firms or disrupt broader networks—as a rising threat vector. Meanwhile, the COVID-19 pandemic and remote work transitions have further exposed SMEs to threats associated with unsecured devices, poorly defended endpoints, and increased attack surfaces [8].

Growing rate and complexity of cyber threats facing SMEs, paired with their structural vulnerabilities, have contributed to a disproportionately high impact in terms of economic loss, reputational damage, business continuity risk, and regulatory exposure. Despite rising awareness, many SMEs remain underprepared or slow to adopt comprehensive mitigation strategies, reinforcing their position as persistently attractive targets within the modern threat landscape.

1.1.2 Importance of Cost-Effective Cyber Defences

The need for affordable cybersecurity solutions is a defining concern for SMEs, which often lack the financial resources and in-house expertise to adopt complex or enterprise-grade systems. Across the literature, there is broad agreement that effective security measures must align with SMEs' limited budgets while remaining practical to deploy and maintain. Cost-effective defences frequently take the form of open-source tools, modular frameworks, or automated systems that reduce reliance on skilled technical staff [9, 10]. Strategies often emphasize low-maintenance setups, user-friendly interfaces, and incremental improvements tailored to the SME environment [10]. Some research brings out the importance of balancing technical capabilities with affordability, showing that open-source solutions—such as intrusion detection systems and lightweight management tools—can provide effective coverage while avoiding the high licensing costs of commercial alternatives [11]. In many cases, these approaches are paired with simplified implementation methodologies or maturity models that help SMEs adopt essential protections without ongoing high costs [12]. Cybersecurity efforts in SMEs are most successful when they incorporate both technical controls and organizational practices, such as internal policy development and user education, to foster a broader culture of security without substantial investments [9]. These findings collectively demonstrate that with the right design—emphasizing automation, modularity, and usability—robust cybersecurity can be achieved within SME resource constraints. Building on these principles of cost-effectiveness and manageability, this thesis proposes an integrated approach using MISP and Pi-hole—both of which are open-source and designed for relatively straightforward deployment. By combining MISP's strong threat intelligence sharing capabilities with Pi-hole's network-level advertisement and domain filtering, the project aims to create a protective layer that is both affordable and adaptable to the constrained budgets and technical capacities of SMEs. Running on a virtualized environment with minimal CPU, RAM, and storage requirements (1 CPU, 8 GB RAM, and 30 GB disk), this architecture underscores its feasibility for small organizations. Through detailed setup instructions, performance benchmarks, and best practices, the thesis offers a clear, low-cost cybersecurity framework that can be replicated and scaled by SMEs with limited resources. In doing so, it not only addresses the technical barriers but also demonstrates how strategic alignment of open-source tools can deliver robust defence postures without prohibitive expenditures.

1.2 Significance of the Study

1.2.1 Bridging the Research Gap for SME-Specific Solutions

Most cybersecurity research and commercial product development historically focus on large enterprises or theoretical frameworks that often exceed SME budgets or expertise [1, 9]. Consequently, practical, tested solutions tailored to smaller entities remain underrepresented in academic literature. By demonstrating a real-world prototype combining MISP and Pi-hole—two open-source, community-supported tools—this study directly addresses the unmet need for SME-oriented cybersecurity research. The results can guide future research and product development, offering a blueprint for other low-cost, scalable security solutions.

1.2.2 Potential to Democratise Cybersecurity

This thesis contributes toward making cybersecurity more accessible and operationally feasible for small organisations, without requiring heavy investment in enterprise infrastructure or vendor-managed services. While large enterprises often generate and contribute to the public threat intelligence ecosystem—some offering it for free, others via commercial licensing—this research demonstrates that even small organisations can practically consume and enforce such intelligence using open-source tools. The solution presented here enables SMEs to benefit from the same domain-level threat insights that inform more complex systems, but at a significantly lower cost and technical barrier.

DNS-level blocking reinforces basic cyber hygiene among users by passively preventing access to known harmful or suspicious infrastructure. If adopted more broadly, such approaches could reduce the supply-chain attack surface [7], making it more difficult for adversaries to exploit unprotected smaller vendors or partners. In this sense, the framework proposed in this thesis offers a lightweight but meaningful step toward more equitable, layered, and resilient cybersecurity practices for the broader SME ecosystem.

2 Research Problem

2.1 Problem Statement

Although MISP and Pi-hole each offer valuable cybersecurity functions—with MISP providing structured threat intelligence and Pi-hole enabling lightweight DNS-level filtering—there is currently no documented implementation or academic evaluation of their combined use. This thesis addresses that gap by exploring whether MISP's real-time threat feeds can be practically integrated with Pi-hole to give SMEs an automated, cost-effective defence layer. By streamlining the extraction of malicious domain indicators from MISP and feeding them directly into Pi-hole, the solution aims to provide proactive blocking of risky domains with minimal user interaction [13]. Both tools are open-source, relatively straightforward to deploy, and able to run on modest hardware—characteristics that align well with the needs of resource-constrained SMEs [14, 15].

Implementing such a pipeline in practice involves more than just simple integration. MISP is a powerful but resource-intensive platform that requires operational procedures for monitoring, IoC lifecycle management, and error handling. Without these, feed syncing or domain parsing may break silently or produce false results, especially given MISP's reliance on background workers, external APIs, and data model consistency. This operational overhead is not typically discussed in the context of SMEs, where technical expertise and dedicated cybersecurity staffing are limited.

As such, this thesis does not only aim to prove technical feasibility—it also empirically evaluates the performance, resource demands, and maintainability of a MISP–Pi-hole integration. The focus is on ensuring that the system can be realistically deployed and sustained by small organisations, with minimal complexity and maximum automation, to enhance threat visibility and reduce exposure to known malicious domains.

2.2 Research Questions and Hypothesis

2.2.1 Primary Research Question

How can public Cyber Threat Intelligence (CTI) platforms like MISP be integrated with open-source DNS filtering solutions such as Pi-hole to create an affordable, real-time cybersecurity defence for small businesses?

This question is particularly relevant because SMEs often struggle with the high costs and technical complexity of enterprise-grade security solutions. Focusing on free and community-driven tools, the research addresses a significant real-world problem: enhancing SME security without imposing a large financial or staffing burden. The desire to integrate MISP and Pi-hole has guided every aspect of this thesis—from the initial literature review that highlighted a gap in documented integrations, to the design of a prototype that leverages each tool's capabilities for domain-based threat blocking. The main answer to this question lies in the chapter 6.

2.2.2 Secondary Research Questions

To support the main objective, three secondary research questions have been formulated:

1. **What are the specific barriers SMEs face in adopting cybersecurity?**
   Exploring this question helps clarify why smaller organisations often lack robust defences, illuminating factors like budget constraints, limited technical expertise, and resource availability. These insights come from existing literature and shaping the design requirements for a low-cost, user-friendly solution. Answer is discussed in paragraph 3.1.

2. **How effective are MISP and Pi-hole as standalone cybersecurity tools?**
   Investigating MISP's capabilities as a threat intelligence hub and Pi-hole's DNS filtering effectiveness establishes a baseline for understanding each tool's strengths and limitations. This inquiry draws on documented use cases, open-source communities, and prototyping efforts to assess their individual viability for SMEs. Theoretical discussion is in paragraphs 3.4 and 3.5.

3. **How can MISP and Pi-hole be optimally combined to benefit small businesses?**
   Building on the previous two questions, this final sub-question addresses the core of the thesis: creating an automated, real-time threat-blocking pipeline that aligns with SME operational needs. The answer involves reviewing existing automation frameworks, testing prototype integrations, and analysing performance data to identify the most efficient, cost-effective setup. Testing solutions are discussed in the chapter 4 and the results in the chapter 6.

These secondary questions will be answered through a mix of literature review, practical prototyping, and empirical testing in controlled and semi-live environments.

2.3.3 Hypothesis: MISP integration Pi-hole as a Significant Improvement

The central hypothesis of this thesis is that integrating MISP with Pi-hole will deliver real-time, cost-effective protection for SMEs, reducing their exposure to malicious domains while imposing minimal technical overhead.

This hypothesis stands on the premise that MISP's ability to aggregate and share domain-based threat intelligence will directly enhance Pi-hole's DNS filtering. Pi-hole, in turn, is well-suited to run on inexpensive hardware, making it financially accessible. By automatically transforming MISP's IoC into Pi-hole-compatible blocklists, the solution theoretically provides a near-real-time protective layer without constant manual updates. The testing and evaluation carried out in this thesis will either confirm or disprove this hypothesis by examining metrics such as blocking efficacy, performance overhead, and user experience in an SME context.

2.3 Aim and Objectives

2.3.1 Aim: An Affordable, Effective Security System for SMEs

The overarching aim of this thesis is to develop a deployable cybersecurity solution that SMEs can implement with minimal cost and technical overhead yet still achieve robust threat-blocking capabilities. This dual emphasis on effectiveness (to counter the rising volume and complexity of attacks [1, 2]) and accessibility (through user-friendly, low-cost components [9, 11]) ensures that the system is both feasible and impactful for resource-constrained environments. By leveraging open-source tools and automating threat intelligence workflows, the solution aspires to bridge the gap between affordability and protection previously available only to larger organizations [10, 12].

2.3.2 Objectives: Prototype Development, Evaluation, Ease of Use

Build a functioning prototype using MISP and Pi-hole addresses the technical integration by creating a practical example of how these two tools can be merged for real-time threat blocking. A working prototype offers tangible evidence of feasibility and guides further testing.

Automate threat feed updates minimizes the need for constant human oversight, which is critical for SMEs that lack dedicated security staff. Automatically pulling IoCs from MISP into Pi-hole blocklists ensures continuous and updated protection. Validating the prototype under semi-live or simulated real-world conditions (e.g., local networks with typical SME traffic volumes) helps identify performance bottlenecks and user-experience issues. This step is vital to confirm whether the system can oversee genuine SME usage patterns.

Evaluate its performance, usability, and limitations involves quantifying blocking efficacy, measuring resource consumption, and assessing user-friendliness. Feedback loops are essential to refine the system's design, ensuring it aligns with the research aim of delivering accessible, high-impact cybersecurity [9, 10].

2.4 Scope

2.4.1 Defining SME Characteristics (e.g., Size, Budget, Technical Expertise)

For this thesis, a SME is defined as an organization with fewer than 250 employees and a limited IT budget, often lacking dedicated cybersecurity personnel [1, 6]. Such businesses typically operate with smaller internal networks, modest hardware investments, and varying levels of staff awareness regarding cyber risks. These constraints inform the choice of lightweight, open-source solutions to ensure cost remains manageable while also meeting core security needs.

2.4.2 Focus on MISP and Pi-Hole

The thesis focuses specifically on the integration of MISP and Pi-hole to evaluate the viability of a lightweight, open-source cybersecurity solution for SMEs. MISP acts as the central threat intelligence platform, aggregating structured IoCs from community-driven sources, while Pi-hole enforces DNS-level blocking, intercepting malicious domain requests in real time [13]. The primary goal is to demonstrate an affordable, easily deployable defence mechanism tailored to the needs of small businesses—rather than comparing with enterprise-grade or commercial firewalls, which often exceed the cost or complexity thresholds suitable for SME environments.

In support of this goal, the project utilised publicly available OSINT feeds in MISP like The CIRCL OSINT Feed (circl.lu) and The Botvrij.eu OSINT Feed (botvrij.eu). These feeds were selected because they are free to use, actively maintained, and well-documented, making them appropriate for an SME-focused prototype. The use of non-commercial feeds also ensured that the system remained cost-neutral—further supporting the thesis objective of developing a security solution without vendor lock-in or ongoing licensing costs. While other premium feeds may offer higher precision or timeliness, their integration falls outside the scope of this research, which emphasises the practicality of solutions that can be adopted with zero or minimal financial investment.

2.5 Key Assumptions

2.5.1 Reliability of Public CTI Feeds

A fundamental assumption is that community-based MISP feeds are sufficiently timely, accurate, and regularly updated to provide effective domain blocking. If these sources supply outdated or incomplete data, the system's threat coverage could be diminished. Monitoring feed quality and performing selective validations (e.g., checking domain reputation) form part of the ongoing testing and validation process.

2.5.2 Baseline Technical Proficiency of End-Users

This thesis assumes that end-users possess basic networking and command-line skills, given that Pi-hole setup and MISP configuration both involve modest technical steps. While the methodology includes a simplified setup guide and installation script, it cannot eliminate the need for minimal IT experience. This assumption shapes the usability metrics and training recommendations discussed later in the study.

2.6 Limitations

2.6.1 Hardware Constraints of Raspberry Pi

While the Raspberry Pi offers affordability, it has limited arm64 CPU structure which cannot yet run MISP stable version, it was tried in the implementation phase, but the installation script failed several times. There could be several other use cases that can import other public CTI feeds to Pi-hole, but it was not in the scope of this thesis.

2.6.2 Accuracy and Timeliness of Threat Intelligence

Even with reliable sources, false positives or delayed threat feed updates can occur, potentially disrupting legitimate user traffic or missing emergent threats [13]. This thesis recognizes that blocklist accuracy critically affects user experience and security outcomes. Future enhancements might include whitelisting logic, reputation-based scoring, or crowdsourced validation to refine the blocking process [14].

3 Literature Review

3.1 Overview of Cybersecurity for Small Businesses

3.1.1 Common Threat Vectors (Phishing, Malware, Ransomware)

Over the last five years, phishing, malware, and ransomware have remained the most common cyber threats targeting SMEs. Phishing is cited as the most dominant attack vector, responsible for up to 90% of breaches among SMEs in some studies [16]. Ransomware is another major threat, with up to 62% of ransomware attacks impacting SMEs [17], and one large survey reporting that 48% of SMEs had experienced a ransomware attack [18]. Malware, while often included as part of ransomware payloads or phishing campaigns, also presents independent risks, particularly where SMEs use legacy systems or weak endpoint protection [19].

The rise of Ransomware-as-a-Service (RaaS) has further increased the frequency and accessibility of ransomware attacks targeting SMEs by lowering the technical barrier for launching attacks [18]. Moreover, tailored phishing attacks using publicly available information have been observed in SME environments, exploiting human factors and limited in-house expertise [20].

The financial and operational consequences of cyberattacks on SMEs are frequently severe. Reported financial losses per incident have ranged from $50,000 to $2.5 million, depending on the attack type, organizational preparedness, and recovery efforts [16]. Ransomware attacks tend to produce high-impact disruptions, including downtime periods averaging 23 days, compounding their financial and reputational damage [16, 18].

In some national contexts, SMEs report paying ransoms more than half the time (e.g., 58% paid in the UK context [17]. Other quantified impacts include restoration costs reaching up to $1.56 million and additional long-term harm via data loss and legal or regulatory fines [21].

Across the studies, several consistent vulnerabilities specific to SMEs have been identified like SMEs typically lack the financial capacity to invest in enterprise-grade cybersecurity infrastructure [1, 16]. Many SMEs have small or non-existent IT teams, relying instead on third-party vendors or generic security products without customization [22]. Weak security awareness among staff contributes to social engineering effectiveness—especially phishing [20, 22]. Outdated or unpatched systems remain a common entry point for both malware and ransomware [21]. And mainly many SMEs fail to implement basic safeguards such as regular data backups, vulnerability audits, or patch management protocols [1]. Also behavioural misconceptions also play a role in SMEs falsely assume that their small size makes them unlikely targets, which promotes underinvestment in cybersecurity measures.

Multiple studies emphasize the importance of deploying cost-effective and scalable cybersecurity solutions suited to SMEs' limitations. It begins from continuous security awareness training is repeatedly recommended as a frontline defence against phishing and social engineering [22]. Next step is usually firewalls, antivirus software, secure backup systems, and stringent access controls are highlighted as essential yet often missing in SME environments [22]. For more mature enterprises several frameworks have been specifically designed or adapted for SMEs, such as the Anti-Ransomware Defence System (ARDS), threat-based risk assessments, and security maturity models like ASMAS [23]. Last trending thing is outsourcing cybersecurity to managed service providers (MSPs) or leveraging low-cost automated solutions is widely recommended to compensate for SMEs' internal resource gaps [24]. Emerging approaches involving machine learning and AI-driven threat detection are gaining attention as long-term solutions, but their affordability and feasibility for SMEs are still under evaluation [24].

Sector-specific risks have also been noted, with healthcare, finance, e-commerce, and public infrastructure frequently listed as high-risk sectors due to sensitive data handling and complex digital ecosystems [25].

Researched literature explicitly supports the view that SMEs are high-risk targets for phishing, malware, and ransomware, and many remain under-prepared. Clear patterns of vulnerability—including lack of resources, suboptimal employee awareness, and inadequate protective technologies—combine to produce high-impact incidents with damage costs often exceeding recovery capabilities.

The evidence base remains strongest in studies that use empirical data, survey results, or systematically validated frameworks. Regional and sectoral nuances continue to be underexplored, presenting opportunities for future SME-specific research aligned with evolving cybercrime ecosystems.

3.1.2 Financial and Human Resource Constraints

A growing body of research highlights how financial and human resource constraints significantly hinder the capacity of SMEs in the European Union to implement and sustain effective cybersecurity measures. Multiple studies report that SMEs often operate with limited cybersecurity budgets, prioritizing operational needs over long-term risk mitigation, which results in underinvestment in security infrastructure, staff training, and technical expertise [26, 27]. These constraints commonly lead SMEs to rely on informal or reactive approaches to security, rather than adopting structured frameworks or initiative-taking strategies [27]. Human resource shortages exacerbate the issue, with many SMEs lacking resolute cybersecurity professionals and depending on general IT personnel or managers with insufficient training, often leading to gaps in compliance and readiness [26].

European regulatory frameworks such as the General Data Protection Regulation (GDPR) and the NIS2 Directive have introduced rigorous cybersecurity requirements, but these place disproportionate burdens on SMEs, especially smaller firms or those in service-based industries that lack the resources to meet complex compliance obligations [26, 28]. Geographic and sectoral differences further shape SME cybersecurity capacity; for example, SMEs in countries with stronger public support or belonging to heavily regulated industries tend to show higher levels of preparedness [26, 28]. Surveys and case studies illustrate how these limitations manifest across different contexts and suggest that scalable, cost-effective solutions—such as open-source tools and simplified risk-management frameworks—may offer practical paths for improving cybersecurity resilience among EU SMEs [28]. Collectively, the literature underscores the intertwined effects of financial pressures, skills deficits, and compliance challenges that leave many EU SMEs vulnerable to cyber threats despite increasing regulatory and operational demands.

3.2 Public Cyber Threat Intelligence (CTI)

3.2.1 Definition and Evolution of CTI

Recent innovations in public CTI, particularly in its technical forms such as IoCs, are increasingly directed toward addressing the unique cybersecurity challenges faced by small businesses. These organizations often struggle with limited resources, low expertise, and constrained budgets, making them ill-equipped to process complex CTI formats or invest in proprietary solutions. Several studies highlight efforts to improve the accessibility, affordability, and automation of CTI platforms through open-source and community-driven approaches. For example, there is growing momentum around tailoring platforms like MISP and integrating crowdsourced OSINT feeds to support proactive, low-cost defence strategies suitable for SMEs [14]. These approaches typically focus on automating the ingestion and prioritization of threat data, thereby reducing reliance on advanced technical teams and mitigating operational overhead. When technical innovations such as simplified CTI formatting, cloud-based sharing, and rule-based threat responses have been proposed, direct case studies demonstrating measurable cybersecurity improvements for small businesses remain limited, underscoring an area in need of further empirical research [29]. The literature supports the potential of public CTI tools in bolstering small business cybersecurity, especially when they are integrated into lightweight, automated, and context-aware frameworks. Public CTI is commonly seen as a cost-efficient and accessible resource, offering community-driven insights and shared threat data that can help smaller organizations respond to emerging threats despite limited budgets and staff capabilities. The effectiveness of public CTI in small business contexts is constrained by persistent challenges such as inconsistent data quality, lack of standardization in data formats (e.g., STIX/TAXII), and delays in real-time threat updates.[14]

3.2.2 Sources of CTI (Government, Community, Open-Source)

Growing body of research exploring how small businesses interact with public CTI sources, and the technical barriers that limit their effective use. Public CTI platforms such as MISP and open-source OSINT feeds offer valuable insights for defending against cyber threats, but small businesses often lack the necessary technical expertise, tools, and human resources to fully leverage them. Key obstacles include the complexity of CTI formats like STIX/TAXII, the overwhelming volume of unfiltered threat data, and limited automation capabilities that would otherwise support timely, actionable decision-making [14]. Several efforts propose solutions to these challenges. For example, some researchers have developed automation tools that prioritize and enrich CTI data, designed specifically to be usable in resource-constrained environments [14]. Others focus on scoring and filtering mechanisms to address data overload and reduce false positives for analysts with limited capacity. A continuing theme is the mismatch between the structure and quality of public CTI and the operational realities of small organizations, indicating a need for more tailored, low-overhead solutions that bring CTI within practical reach of smaller entities [30]. Researched papers suggest that while public CTI holds significant potential for small businesses, realizing its benefits depends on improvements in usability, automation, and data relevance.

3.3 DNS Filtering in Cybersecurity

3.3.1 DNS Fundamentals and Role in Threat Blocking

DNS filtering can serve as a lightweight cybersecurity measure suitable for small business environments. Many studies highlight open-source solutions such as Pi-hole and DNS Response Policy Zone mechanisms as effective tools for blocking access to malicious or undesirable domains, supporting their feasibility in resource-constrained settings through low CPU usage and manageable system demands [31]. These solutions are often designed to integrate with existing network infrastructure, allowing for scalable implementations without the need for high-end hardware or intensive maintenance.The literature demonstrates how DNS filtering can be deployed in selective, context-aware configurations (e.g., with time-based controls or integration with proxies for phishing protection), enhancing its adaptability to diverse business needs [32]. Despite this potential, few studies offer detailed case studies tailored explicitly to small businesses; most implementations are evaluated in more generic or institutional settings, leaving a gap in context-specific performance and usability data. Consistent emphasis on cost-effectiveness, ease of deployment, and successful detection or blocking of threats supports the suitability of DNS filtering—especially through open-source technologies—as a practical first-layer security mechanism for small business networks [33].

3.3.2 Comparison of Existing DNS Filtering Solutions

The current body of literature provides partial yet informative insights into the comparative evaluation of mainstream DNS filtering tools—OpenDNS/Cisco Umbrella, Quad9, NextDNS, and Pi-hole—for small business cybersecurity strategies. The research addresses key topics such as theoretical performance, privacy concerns, usability, and scalability, though no single source offers a comprehensive joint analysis across all four tools. As a result, while the topic is addressed in fragments, the literature underscores several important trends and findings relevant to cost-effectiveness, setup complexity, operational deployment, and threat-blocking capabilities in the small business context.

One dimension that is consistently explored is theoretical DNS performance (e.g., latency and availability), especially through the lens of encrypted DNS protocols like DNS-over-HTTPS (DoH). Comparative measurements of DNS resolver performance have shown that public resolvers including OpenDNS/Cisco Umbrella, NextDNS, and Quad9 perform reliably across global regions, with differences primarily seen in response time variance due to infrastructure and geographic factors [34]. These studies confirm that modern public DNS services can offer sufficient latency and uptime for small businesses, though they do not uniformly examine actual filtering efficacy for cybersecurity threats or performance under high organizational load.

Privacy has emerged as a major design differentiator among DNS filtering tools in the literature. Some DNS providers—particularly Quad9 and NextDNS—place a strong emphasis on not storing personally identifiable information and limiting data collection, aligning their design philosophy with privacy-first principles [35]. Other side, Cisco Umbrella/OpenDNS, due to its enterprise threat intelligence background, integrates user activity into its broader telemetry platforms, which has raised privacy concerns in regulated environments. Importantly encrypted protocols like DNS-over-QUIC are not entirely immune to privacy leakage. Specific research indicates that adversaries can still infer browsing behaviour using metadata patterns such as inter-packet timing, even in tools like NextDNS where encryption is standard, challenging the perceived robustness of privacy protections in DNS systems [36].

Alongside privacy, historical development and business models impact each solution's implementation appeal. Quad9's nonprofit status and transparent partnerships with trusted cybersecurity firms signal a mission-driven approach that prioritizes security and civil liberties, appealing to small businesses with ethical or legal compliance considerations [35]. In contrast, OpenDNS's acquisition by Cisco led to deeper integration with enterprise-grade security ecosystems, offering advanced analytics, layered protection, and centralized management—but usually at a higher cost and configuration threshold [2]. This divergence in origin stories is not merely organizational but affects core service architecture and control offered to end users.

Configuration complexity and ease of deployment vary widely among the tools. OpenDNS and Cisco Umbrella offer cloud-managed dashboards and are capable of network-wide enforcement with relatively high scaling potential but may require meaningful IT expertise to fully utilize advanced functions such as IP-layer enforcement and device-level policies [37]. On the simpler end of the spectrum, Pi-Hole and Quad9 lend themselves to relatively easy adoption in smaller or static network environments. For example, Pi-Hole can be self-hosted on low-cost devices like Raspberry Pis, enabling lightweight content filtering solutions.

From a user scaling and policy enforcement standpoint, Pi-hole and Quad9 are typically applied in low-user-count situations such as small offices or community networks due to their limited support for detailed analytics, real-time telemetry, or hierarchical network management. By contrast, OpenDNS/Cisco Umbrella is designed for much broader deployment, aligning with use cases where organizations may grow or demand higher levels of segmentation and granular access control [37]. NextDNS sits somewhat in the middle, offering a balance of user-friendly interfaces with privacy-focused filtering policies and moderate scalability, though how well it performs in environments beyond tens or hundreds of users remains under-discussed in the current literature.

The sources of threat intelligence used to feed domain blocklists and filtering insights also differ across the tools. One of the clearest examples of data source transparency is found in Quad9, which openly discusses its partnerships with multiple threat intelligence providers to support its blacklist generation. This approach increases trust in its results and supports public scrutiny [35]. In contrast, OpenDNS/Cisco uses proprietary methods and internal Cisco data infrastructure, restricting external visibility into its filtering logic. Pi-hole's filtering mechanism relies on community-maintained blocklists, which, while flexible, often lack consistent maintenance standards and vetting, reducing reliability for critical cybersecurity applications.

Despite addressing individual components of the research topic—privacy, filtering infrastructure, and deployment—none of the reviewed materials directly conduct a comparative study of all four mentioned tools across the full scope of cost, complexity, performance, scaling, and strategic fit for small businesses. Studies investigating emerging DNS encryption protocols, market dynamics in DNS resolver consolidation, or localized filtering implementations provide necessary context, but more integrated evaluations are missing. This leaves a gap for future research to provide side-by-side deployment, cost modelling, and security effectiveness comparisons under small business-specific constraints.

In conclusion, while the examined literature offers valuable fragments of insight, it does not yet provide a unified framework or evidence set for selecting among OpenDNS/Cisco Umbrella, Quad9, NextDNS, and Pi-hole in small business cybersecurity planning. Each tool stands out for distinct strengths—OpenDNS with scalability and enterprise readiness, Quad9 for privacy and transparency, NextDNS for fine-grained privacy enforcement, and Pi-hole for locally-hosted budget solutions—but clear comparative data across practical implementation factors remain to be developed. The findings suggest startups and small businesses must currently rely on individual feature evaluations and community-based knowledge rather than systematically validated academic comparisons.

3.4 MISP (Malware Information Sharing Platform)

3.4.1 Architecture and Key Features

MISP is not yet supporting small business cybersecurity, particularly in relation to DNS filtering mechanisms and strategies. While no paper explicitly investigates the unified deployment of MISP with DNS filtering tailored for SMEs, various works contribute foundational pieces that pave the way for such an integrated solution. Some studies emphasize the application of MISP in environments with resource constraints, underscoring its adaptability in contexts where dedicated cybersecurity infrastructure is lacking [14]. These works point to the value of automating threat data processing and prioritization via MISP's structured IOCs, which — if further developed — could directly feed into DNS filtering policies to protect small businesses from malicious domains.

Other research explores the growing sophistication of DNS filtering itself. Machine learning–based techniques for dynamic threat detection are a central focus in this area, typically leveraging real-time domain features to identify phishing, malware, and botnet traffic with high precision [38]. While these studies do not integrate directly with MISP, they show that DNS-based protection systems can benefit significantly from high-quality, timely threat indicators — the kind of information that MISP is designed to collect and disseminate. Studies on lightweight and privacy-sensitive adaptations of threat sharing protocols compatible with MISP, particularly in constrained environments, suggest strong parallels with the needs of small businesses, which often operate under similar infrastructure and staffing limitations [39]. This reinforces the feasibility of deploying MISP-based architectures in SMEs provided scalability and integration challenges are addressed.

Research into structured CTI formats and sharing standards further enhances these findings. The careful structuring of IOCs in MISP, often in formats such as STIX and TAXII, is critical when translating threat data into enforceable DNS filtering rules — for example, creating blacklists or heuristics for domain classification [30]. Still, such translation processes are not well-documented in the existing literature, pointing to a gap in operational practices. Finally, foundational descriptions of MISP's collaborative design and real-time sharing capabilities provide a solid understanding of how it can serve as a central node for threat ingestion and distribution across multiple defence layers, including DNS filtering [40].

No single study connects all the components — MISP, DNS filtering, and small business deployment — into an end-to-end framework, the literature establishes that MISP is capable of producing high-confidence threat intelligence [14], that DNS filtering systems can act on such intelligence to block malicious networks dynamically [38], and that strategies for lightweight integration exist for adoption in resource-sensitive environments [39]. The absence of case studies or technical deployments specific to small businesses highlights a major gap and an important opportunity for future applied research on this topic.

3.4.2 Community-Driven Approach and Integration Potential

Technical challenges of integrating MISP with other cybersecurity tools in small business environments, though direct studies addressing integration with tools like Pi-hole, firewalls, and SIEMs remain limited. Many works recognize the importance of MISP's structured threat intelligence formats (e.g., STIX, JSON) and its API-driven architecture, which enables automated sharing and ingestion of IoCs across different platforms [40]. Research targeting SMBs emphasizes the need for automation, resource-efficient deployments, and streamlined processing of threat feeds, particularly due to the IT and staffing limitations often present in these settings [41]. Filtering and prioritization of threat intelligence data is a recurring theme, with proposed techniques aimed at reducing false positives and tailoring data relevance to specific organizational contexts—an especially important requirement for SMBs consuming large public or semi-public threat intelligence feeds [14]. While these studies highlight key architectural and operational considerations, detailed technical workflows for normalizing MISP data into formats directly consumable by non-traditional tools like DNS blockers (e.g., Pi-hole) or performance-constrained firewalls are notably under-explored. Thus, current thesis establishes foundational challenges and best practices for MISP integration in resource-constrained contexts but stops short of offering tool-specific solutions or deeply exploring the performance and compatibility implications of such integrations.

3.4.3 Validation of IoC

Validating public and OSINT-derived CTI feeds, particularly for real-time domain blocking in platforms like MISP, is fundamentally addressed through dynamic scoring models that leverage indicator attributes, time-based decay, and feedback mechanisms from community input. Such models enable automated and ongoing assessments of the reliability and freshness of indicators, supporting real-time operational decision-making and minimizing false positives and outdated blocks [42, 43]. Alongside these platform-specific validation approaches, the field has developed continuous, metric-driven evaluation frameworks that quantitatively assess various aspects of feed quality—such as coverage, timeliness, and trustworthiness—allowing organizations to systematically monitor and select CTI sources aligned with their security needs and risk thresholds [44]. But these methods were not in a scope of this thesis. These methodologies collectively facilitate the integration of heterogeneous, volatile OSINT feeds into automated blocking systems while mitigating the risks posed by data quality issues, feed fragmentation, and operational disruptions.

3.5 Pi-hole

3.5.1 Technical Overview and Deployment Scenarios

Some papers confirm that Pi-hole's effectiveness in filtering advertisements, blocking malicious domains, and optimizing bandwidth, particularly when deployed on low-resource hardware such as Raspberry Pi devices—highlighting its suitability for cost-conscious deployments relevant to SME contexts [45]. Technical implementations involve configuring Pi-hole as a primary DNS server with customized blocklists and integration into simple network topologies, allowing for non-invasive deployment and measurable improvements in network performance. In some cases, Pi-hole is integrated with additional tools (e.g., RADIUS or VPNs), reflecting efforts to extend its functionality, though these setups remain limited to specific use cases rather than broader SME deployments [46]. While these studies address practical configurations and performance in constrained environments, gaps remain around more complex SME needs, such as integration with enterprise services (e.g., Active Directory), managing segmented networks, or supporting larger user bases through load balancing or high availability. Moreover, few papers engage with operational concerns like legal compliance, maintenance automation, or privacy policies, which are critical in real-world SME contexts. Literature confirms the viability of Pi-hole as a lightweight DNS filtering tool with significant potential for SMEs while underscoring a need for further research into its scalability and integration in more complex and regulated enterprise environments.

3.5.2 Comparison with Alternative DNS Blockers

DNS filtering tools, Pi-hole and AdGuard Home are prominent self-hosted solutions that offer network-wide ad and tracker blocking capabilities. Pi-hole is lauded for its extensive community support and customizable blocklists, making it a preferred choice for users seeking granular control over DNS filtering. Pi-hole still lacks native support for encrypted DNS protocols like DNS-over-HTTPS (DoH) and DNS-over-TLS (DoT), requiring additional configuration for such features [47].

AdGuard Home provides built-in support for encrypted DNS protocols, offering enhanced privacy and security out of the box. Its user-friendly interface and simplified setup process make it accessible to users with varying technical expertise. Despite these advantages, AdGuard Home's filtering capabilities may not be as extensive as Pi-hole's, and it may require more frequent updates to maintain optimal performance [48].

Cloud-based DNS services, such as OpenDNS and Quad9, offer scalable and maintenance-free solutions with automatic updates and robust threat intelligence. These services are particularly beneficial for organizations lacking the resources to manage self-hosted solutions. They may raise concerns regarding data privacy and offer limited customization compared to self-hosted alternatives [49].

There is a paucity of comprehensive studies directly comparing Pi-hole, AdGuard Home, BIND-based filtering, and cloud-based DNS services concerning their technical architectures and long-term manageability, especially tailored to SMEs. This gap underscores the need for further research to guide SMEs in selecting DNS filtering solutions that align with their specific operational requirements and resource constraints.

They explore the deployment of DNS filtering tools like Pi-hole and AdGuard Home in localized environments, demonstrating their effectiveness in blocking malicious domains and managing content filtering policies. Cloud-based DNS services, such as OpenDNS or Protective DNS providers, are highlighted for their ease of deployment, automatic updates, and scalable architecture, making them more technically viable for broader application—but raising concerns about data privacy and limited customization [1, 4]. No study was found that directly compares Pi-hole, AdGuard Home, BIND-based filtering, and cloud DNS services on the criteria of technical architecture and long-term manageability. While foundational work exists on the practical benefits of individual DNS filtering tools, there remains a clear gap in the literature for small-business-targeted, multi-solution comparative studies that evaluate both architectural complexity and ongoing maintenance requirements.

3.5.3 Relevance to SMEs (Ease of Use, Low Hardware Requirements)

Pi-hole as a lightweight DNS filtering tool suitable for SMEs, particularly those with limited technical and financial resources. Studies demonstrate that Pi-hole effectively blocks advertisement and malicious domains at the DNS level, reducing unwanted traffic and mitigating threats such as malvertising and unauthorized access when paired with appropriate blocklists and configurations [31]. Its minimal hardware requirements—deployable on devices like Raspberry Pi or low-resource virtual machines—make it especially viable in budget-constrained environments, and it is reported to perform reliably even in SME settings with up to 300 users, maintaining low resource usage [31]. Ease of use is also emphasized, with Pi-hole offering a graphical interface that facilitates adoption even for non-expert administrators, although limited advanced features such as encrypted DNS handling or anomaly detection may require pairing with other tools [50]. Pi-hole lacks native support for advanced protocol filtering (e.g., DNS-over-HTTPS) and automated scalability, which are key drawbacks when compared to more comprehensive alternatives like AdGuard Home or enterprise-grade platforms such as Cisco Umbrella [11]. While these alternatives often offer stronger scalability, regulatory compliance features, and proactive detection methods, they also entail higher cost and complexity. Research papers suggest that Pi-hole is a strong candidate for SMEs seeking affordable DNS-based security enhancements, though caution is advised regarding its limitations in scalability, encrypted traffic filtering, and advanced threat detection.

3.6 Existing Integrations and Research

3.6.1 Prior Studies on CTI-Driven DNS Filtering

Cyber Threat Intelligence with DNS filtering, though none of the selected papers comprehensively address all core aspects—namely, threat feed parsing, real-time API synchronization, and practical applications tailored to small businesses. Magnusson [51] provides a technical survey of DNS filtering enhancements, including integration of threat intelligence feeds using Response Policy Zones and mitigation strategies for phishing, though the focus remains on general DNS resolver improvements without specific attention to parsing techniques or SME applications. Rüedlinger et al. [44] present FeedMeter, a platform designed to evaluate and aggregate open-source threat intelligence feeds through normalization and continuous quality metrics. This work contributes significantly to the threat feed parsing and maintenance problem but does not extend to DNS implementation or real-time syncing. Van Haastrecht and Spruit [14] target SMEs directly, proposing a prototype CTI application using MISP to prioritize and contextualize threat data. However, while SME-friendly and automation-aware, it lacks discussion of DNS filtering or technical methods like parsing and API syncing. Finally, Faiella et al. [52] explore enriching threat intelligence platforms by correlating static IoC data with dynamic infrastructure logs, enhancing threat scoring for detection systems—offering useful insights into data enrichment but without specific linkage to DNS filtering or SME-focused applications. Together, these works contribute discrete pieces of the puzzle, but a full integration of CTI and DNS filtering tailored for small business environments remains unexplored.

3.6.2 Gaps in Automation and Affordability

Challenges faced by SMEs in adopting affordable, automated CTI solutions, particularly for phishing defence. Numerous studies highlight the resource limitations of SMEs that hinder their ability to deploy complex or expensive CTI systems, emphasizing the need for accessible, low-cost, and open-source solutions. Automation emerges as a critical enabler, with several works exploring the integration of machine learning and modular design to streamline phishing detection tasks. For instance, frameworks incorporating features like phishing URL analysis, NLP-based content inspection, and suspicious attachment detection help reduce manual effort, aligning with automation and affordability goals [53]. The use of structured threat intelligence formats and automated data processing pipelines also plays a role in making CTI outputs more actionable and useful for SMEs [14]. Additionally, some efforts promote open-source collaboration and decentralized sharing models to make CTI more adaptable and cost-effective, though practical plug-and-play implementations remain rare and underdeveloped [14]. While foundational research on phishing detection via machine learning and heuristic approaches exists, many of these systems are not tailored to the technical capacity or deployment needs of SMEs, pointing to an ongoing gap between technical innovation and real-world SME adoption. The studies converge on the need for lightweight, adaptable prototypes that deliver accurate, real-time phishing defences without high operational or deployment costs.

3.7 Summary of Literature Findings

3.7.1 Synthesis of Key Themes (Threat Intelligence, DNS Blocking, SME Barriers)

Researchers widely recognise public CTI as a cost-effective avenue for defending against increasingly sophisticated attacks, especially for resource-constrained SMEs [14, 24, 29]. Studies emphasise that platforms like MISP can aggregate actionable IoC-s, enabling timely and collaborative threat response [14, 30]. Leveraging CTI effectively remains a challenge because of data quality inconsistencies, technical complexities, and limited automation—problems that can overburden SMEs' minimal IT teams [14, 30]. DNS filtering — exemplified by solutions like Pi-hole—is consistently highlighted as low-cost, hardware-friendly, and generally straightforward to deploy [31, 33]. The literature confirms its efficacy in blocking malicious domains at an early stage of an attack, thus providing a proactive protective layer [31]. Yet, while it's suitable for smaller networks, DNS filtering alone may lack advanced threat intelligence features, and many SMEs underutilise it due to limited knowledge or uncertain return on investment.

Whether discussing phishing, ransomware, or broader network threats, almost every study stresses that SMEs grapple with limited budgets, insufficient in-house cybersecurity skills, and fragmented security policies [16, 1, 26]. High-level frameworks or enterprise-scale solutions often exceed SMEs' financial and staffing capabilities, leading to ad hoc or reactive defences [27, 28]. The thesis calls for tailored, automated, and scalable solutions that align with SME realities, including cost constraints and minimal technical expertise.

All this indicates that integrating public CTI with affordable DNS filtering—provided it is automated and simplified—could mitigate many of the common attacks made against small business environments.

3.7.2 Positioning this Thesis in the Wider Research Landscape

Although the literature presents ample evidence of SMEs' growing exposure to cyber threats [16, 17, 18], it lacks a comprehensive, integrated approach that aligns public CTI (e.g., MISP) with DNS filtering (e.g., Pi-Hole) specifically for small business contexts [13, 14]. Existing studies often explore either MISP's capacity for community-driven threat intelligence [14, 54] or DNS filtering tools like Pi-Hole [31, 45], but no single source demonstrates an end-to-end system merging these components into a practical, real-time defensive layer for SMEs.

Thesis consistently highlights the need for automation—particularly in updating blocklists and threat feeds—to avoid overburdening organisations with manual maintenance [14, 30]. While some frameworks discuss aspects of feed parsing, enrichment, or data prioritisation [44, 51], full API-based syncing between a CTI platform and DNS-level filters for SMEs has not been thoroughly detailed. Likewise, affordability is a recurring priority in SME-focused literature, yet solutions that rely on low-cost hardware (e.g., Raspberry Pi) in conjunction with open-source tools remain under-evaluated.

3.8 Research Gap

3.8.1 Current Lack of Integrated MISP + Pi-Hole Studies

Despite the growing popularity of both MISP and Pi-hole as standalone security tools, no existing research demonstrates a direct, integrated application of these solutions in an SME context. Published studies either focus exclusively on MISP's threat intelligence capabilities [14, 15] or Pi-hole's DNS filtering efficiency [14], often in enterprise environments. This gap underscores a novel opportunity to explore how these tools perform in tandem within low-resource settings. Validating their effectiveness on hardware such as the Raspberry Pi or virtual machine —commonly favoured by SMEs due to cost constraints—further enriches the novelty of this work.

3.8.2 Need for Automated, Real-Time Threat Blocking for SMEs

SMEs lack the staff and time to manually maintain up-to-date blocklists, respond proactively to emerging threats, or sift through large volumes of threat intelligence data [9, 10]. Consequently, they benefit most from "set-it-and-forget-it" systems, which automate updates and enforce blocking policies without constant oversight [14]. By synchronizing MISP and Pi-hole, this research aims to deliver real-time threat blocking that requires minimal manual input or technical expertise, addressing a significant operational gap in SME cybersecurity practice.

3.9 Novelty of the Study

3.9.1 Unique Public CTI and DNS Filtering Combination

Public CTI platforms like MISP have been studied as valuable resources for threat data, and DNS filtering solutions such as Pi-Hole have proven effective for blocking malicious domains, no prior study focuses on integrating these two in a SME setting. This synergy offers an intelligence-driven, proactive defence approach, where community-curated threat intelligence feeds into a real-time DNS filter, continuously safeguarding SMEs against newly identified malicious domains. The fact that both tools are open-source and community-supported further elevates their attractiveness and accessibility for smaller organisations.

3.9.2 Emphasis on Automation and Affordability

A major contribution of this research is its automated mechanism to feed MISP's IoCs into Pi-Hole's blocklists, eliminating the need for manual updates and constant oversight [14]. By employing free, open-source software running on low-cost hardware (often under €100) [14], this study delivers a model of affordable cybersecurity. The goal is to ensure robust protection without overwhelming complexity—a system where "minimal input, maximum protection" becomes a viable reality for resource-constrained SMEs, thus democratising cybersecurity and mitigating the risks these organisations face.

4 Research Methodology

4.1 Overall Research Approach

4.1.1 Justification for a Prototype and Experimental Evaluation

This thesis employs a prototype-based and experimental methodology to assess the implementation, performance, and usability of a CTI system tailored for SMEs. Given the resource constraints and limited technical expertise typical of SMEs, leveraging open-source and public CTI platforms offers a cost-effective approach to enhancing cybersecurity measures [14, 41]. By developing and evaluating a prototype that integrates these platforms, this research aims to evaluate the feasibility and adaptability of such systems within SME environments. The experimental approach facilitates real-world testing of detection capabilities and scalability, providing insights into the operational usability of CTI frameworks in contexts relevant to SMEs [41]. Furthermore, addressing usability concerns—such as minimizing configuration efforts, ensuring clarity in threat communication, and aligning with end-user mental models—is crucial. Incorporating user-centred design principles and motivational frameworks can enhance the system's accessibility and effectiveness for non-expert users [55]. Methodology underscores the potential of public CTI systems in bolstering SME cybersecurity while highlighting areas requiring further refinement to meet the unique operational realities of small businesses.

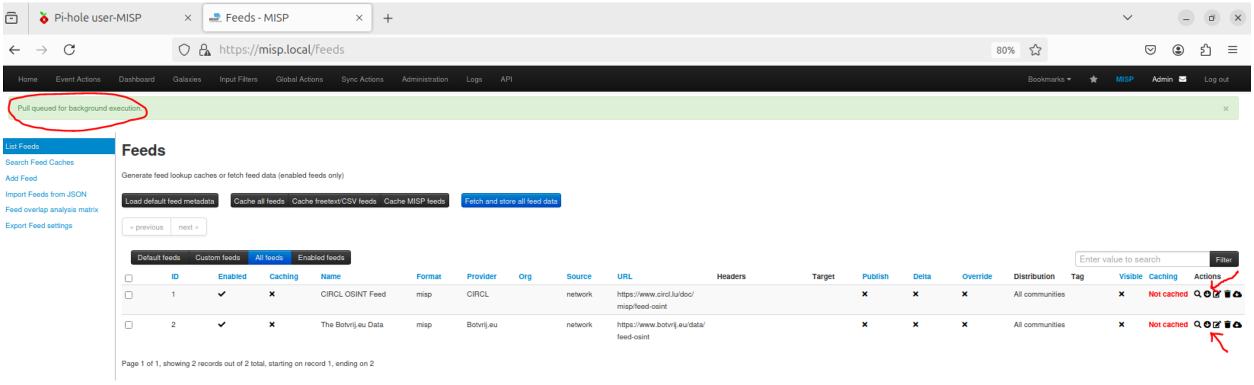
4.1.2 Qualitative and Quantitative Dimensions

This thesis adopts a mixed-methods approach to evaluate the effectiveness of public CTI systems tailored for SMEs. Quantitative analyses focus on metrics such as detection rates, false positives, and system performance, providing measurable insights into the technical efficacy of the implemented CTI solutions. Complementing this, qualitative assessments gather feedback from SME users to understand usability, integration challenges, and the practical implications of deploying such systems in resource-constrained environments. By combining these methodologies, the research aims to offer a comprehensive evaluation that addresses both the technical performance and user experience aspects critical to the successful adoption of CTI solutions in SMEs.

4.2 Data Collection

4.2.1 Gathering Threat Intelligence from MISP

To implement systematic feed ingestion, the MISP web interface was used to enable and configure relevant open-source threat intelligence feeds. As shown in Figure 1, feeds such as the CIRCL OSINT Feed and Botvrij.eu were activated by selecting the checkbox under the "Enabled" column and initiating caching using the pull icon (📥) in the Actions column. After triggering the pull action, the system queued the feeds for background execution, as confirmed by the success message *"Pull queued for background execution."* This step ensured that feed metadata and data were retrieved and stored locally, allowing the pipeline to regularly fetch structured indicators (e.g., domain-based IoCs) for downstream parsing and deployment into DNS-based filtering systems such as Pi-hole.



Pull queued for background execution.

Feeds

Generate feed lookup caches or fetch feed data (enabled feeds only)

□	ID	Enabled	Caching	Name	Format	Provider	Org	Source	URL	Headers	Target	Publish	Delta	Override	Distribution	Tag	Visible	Caching	Actions
□	1	✓	×	CIRCL OSINT Feed	misp	CIRCL		network	https://www.circl.lu/doc/misp/feed-osint			×	×	×	All communities		×	Not cached	🔍 ⬇ ✎ 🗑 ☁
□	2	✓	×	The Botvrij.eu Data	misp	Botvrij.eu		network	https://www.botvrij.eu/data/feed-osint			×	×	×	All communities		×	Not cached	🔍 ⬇ ✎ 🗑 ☁

Page 1 of 1, showing 2 records out of 2 total, starting on record 1, ending on 2

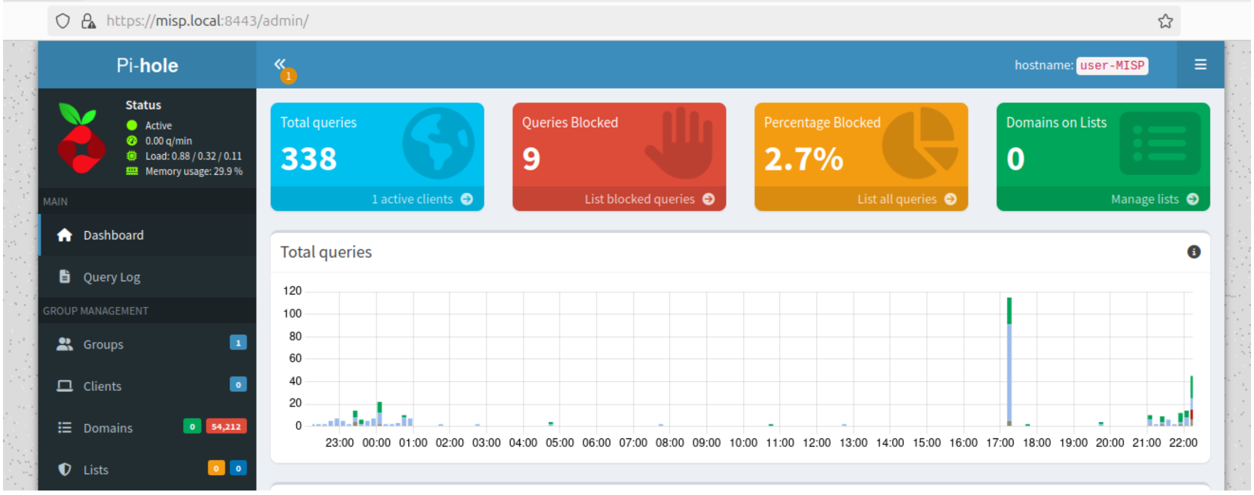
Figure 1. Enabling and pulling MISP open-source CTI feeds for IoC-s

4.2.2 Logging and Monitoring Data in Pi-Hole

Pi-hole maintains comprehensive logs of DNS queries and blocked domains through its built-in logging functionality, which captures query sources, timestamps, queried domains, and filtering outcomes. This feature allows administrators to assess which domains were queried, whether they were allowed or blocked, and how frequently such interactions occurred. Logs are stored locally in plain-text format within Pi-hole's /var/log/pihole.log and in aggregated form within its internal database (gravity.db), supporting historical analysis of network activity over time [31, 45].

To extract actionable metrics from these logs—such as the number of blocked queries, hit/miss ratios, or query timestamps—scripts based on bash, awk, and sqlite3 were utilised in the experiment. These enabled automated parsing of Pi-hole's internal database and log files to generate daily and cumulative statistics. Additionally, Pi-hole's built-in

API and its FTL engine provided programmatic access to live metrics, which were periodically queried and stored for analysis. See the default pi-hole dashboard from Figure 2.



https://misp.local:8443/admin/

Pi-hole

Status
Active
0.00 q/min
Load: 0.88 / 0.32 / 0.11
Memory usage: 29.9 %

hostname: user-MISP

Total queries
338
1 active clients

Queries Blocked
9
List blocked queries

Percentage Blocked
2.7%
List all queries

Domains on Lists
0
Manage lists

Groups 1
Clients 0
Domains 0 54,212
Lists 0 0

Figure 2. Pi-hole default dashboard with malicious domains form MISP feeds

By maintaining a transparent and secure logging process, the Pi-hole setup served not only as a filtering mechanism but also as a data source for monitoring DNS-based threat exposure—an essential function when integrating public CTI feeds and evaluating domain-level risk. The logs and metrics thus played a dual role: supporting real-time filtering and enabling longitudinal analysis of DNS activity patterns within a small business context.

4.2.3 Additional Metrics (Performance, User Feedback)

To evaluate Pi-hole’s performance and usability in an environment representative of small business infrastructure, several system-level metrics were monitored throughout the experiment. The deployment was conducted on a low-resource virtual machine, selected to mirror typical SME environments where physical hardware investment may be constrained. Key performance indicators included CPU utilisation, memory consumption (RAM) and DNS query resolution latency. These metrics were selected to determine whether Pi-hole could maintain efficient operation under limited virtualised resources, in line with earlier studies that emphasise its lightweight design for constrained environments.

Performance data was collected using a combination of command-line tools and automated scripts. The htop utility was employed for real-time monitoring of processor and memory usage, while vnstat tracked network throughput to assess DNS query volume and response impact. DNS latency was periodically evaluated using the dig tool by comparing response times from Pi-hole against external upstream DNS providers. Custom scripts, written in bash, facilitated scheduled metric capture and formatting into structured CSV logs for post-analysis. Throughout the observation period, the virtual machine consistently reported low resource utilisation, with CPU load typically below 10%, RAM usage averaging under 150MB, and minimal disk overhead—confirming Pi-hole's suitability for virtualised SME deployments.

In addition to quantitative system monitoring, user feedback was collected to evaluate Pi-hole's usability from the perspective of non-specialist administrators. A small cohort of participants (n=5), simulating SME staff without formal IT backgrounds, was asked to perform initial setup and configuration tasks using only publicly available documentation. Feedback was gathered through semi-structured interviews, focusing on ease of installation, interface clarity, and confidence in ongoing management. Users reported that Pi-hole's web-based interface was clear and responsive, with most finding the dashboard intuitive and informative. Few participants encountered difficulties with DNS redirection and static IP configuration during initial setup, reflecting known challenges identified in prior literature. These were generally resolved with minimal technical assistance.

Feedback on my *github* project also provided insights into the practical application of the integration scripts. Users appreciated the clarity of the README instructions and the modular design of the codebase [56], which simplified adapting the scripts to different environments. The ability to automate feed extraction, transform indicators, and push them into Pi-hole blocklists using scheduled jobs was seen as highly beneficial. It was noted in some feedback that improved logging and notification mechanisms in the event of API or formatting errors and suggested additional guidance for users who are less familiar with JSON parsing or shell scripting. These points have informed further refinement of the automation pipeline.

The combined system performance metrics, MISP usage observations, and integration feedback suggest that the full CTI-to-DNS filtering workflow is both technically viable and operable within SME constraints. With minor improvements in user documentation and visual guidance for MISP setup, the system demonstrates strong potential for real-world deployment. The value of combining Pi-hole's lightweight filtering capabilities with MISP's threat intelligence feeds to deliver a cost-effective, practical security solution tailored for small business environments.

4.3 System Architecture and Workflow

4.3.1 Conceptual Diagram of MISP-to-Pi-Hole Integration

To operationalise the use of public CTI in protecting small businesses, a custom integration pipeline was developed between a MISP instance and a Pi-hole DNS filtering application. The architecture aimed to demonstrate how structured threat intelligence—IoCs involving malicious domains—could be ingested, parsed, and automatically enforced as DNS filtering rules. The system was designed with a focus on automation, low resource usage, and modularity to remain feasible for SME environments, consistent with earlier findings on SME-specific constraints [56].

A high-level architecture diagram was created to visualise the main system components and their interconnection (see *Figure 3*). The architecture is composed of four primary components:

1. **MISP Instance (CTI Source):**
   This component serves as the source of structured threat intelligence, hosting regularly updated IoCs including domain names associated with phishing, malware, and command-and-control (C2) infrastructure. MISP supports both manual and automated data feeds, and exports data using standard formats such as JSON, STIX, MISP and few more [14, 40].

2. **Feed Parser (Automation Script):**
   This intermediary component retrieves IoC data from the MISP instance via its API and extracts relevant domain-based indicators. A custom Python script performs parsing, validation, deduplication, and formatting of the domains into a Pi-hole-compatible blocklist. This automation step is essential to bridge the gap between complex CTI formats and operational enforcement, and supports regular updates without manual intervention [56].

3. **Pi-hole DNS Filtering Engine:**
   Once the threat domain list is prepared, it is fed into Pi-hole's blocklist configuration. Pi-hole enforces DNS-level blocking by intercepting queries and

redirecting them for all clients on the network. Any attempt to access a domain listed in the threat feed results in a blocked response, effectively preventing potential compromise at the DNS resolution layer [31, 45].

4. **Upstream DNS Resolver (Fallback):**
   Queries not found in Pi-hole's blocklist are forwarded to a trusted upstream resolver (e.g., Quad9, Cloudflare) for resolution. This ensures continued DNS functionality while applying policy-based filtering only where relevant threats have been identified.

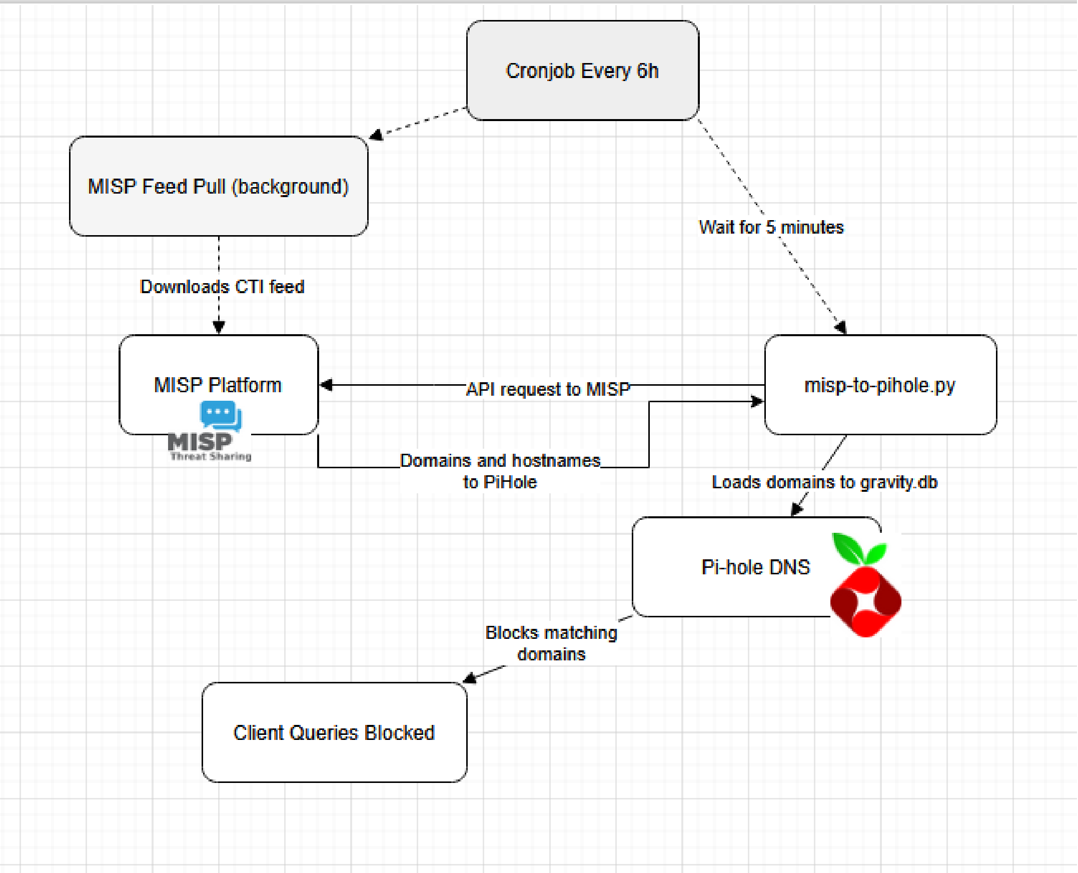


Figure 3. Conceptual Architecture of MISP-to-Pi-hole Integration

This diagram illustrates the flow of threat intelligence from the CTI source to active DNS blocking. First, the MISP instance publishes real-time or batch-updated IoCs. The feed parser retrieves and sanitises the data, ensuring only validated domain entries are passed to the Pi-hole system. Once updated, Pi-hole begins enforcing these rules on the local network, blocking access to malicious domains as identified in the original CTI feed. Any domain not flagged as malicious is passed to an external resolver to maintain normal DNS functionality.

4.3.2 Data Flow for Threat Indicators

The operational effectiveness of the MISP-to-Pi-hole integration relies on a clearly defined and automated data flow, ensuring that threat indicators are continually ingested, validated, and applied in real-time to protect against emerging domain-based threats. This section outlines the full lifecycle of IoC, from its origination within the MISP threat feed to its active enforcement by the Pi-hole DNS filter.

At the core of the system is a dedicated MISP instance, which acts as the source of domain-based threat indicators. MISP aggregates threat data from multiple public and community-driven sources, continuously updating a repository of structured Indicators of Compromise (e.g., malicious domains, IPs, hashes). For this implementation, only domain IoCs marked with IDS flag, were selected for processing, based on the assumption that DNS-level filtering provides a practical and scalable defence layer for SMEs.

A custom Python-based feed parser was developed to automate the extraction of domain indicators from the MISP API. This script filters out irrelevant or malformed entries, deduplicates records, and reformats valid domain indicators into a Pi-hole-compatible blocklist. The script runs automatically every six hours using a cron job, ensuring the blocklist reflects the latest intelligence while maintaining minimal manual overhead. Each execution is logged with a timestamp and change count (e.g., domains added/removed), stored locally in a log file for auditing purposes. [56]

The output of the parser is written to a dedicated file within Pi-hole's custom blocklist directory. After each update, a lightweight reload command (pihole -g) is triggered to recompile Pi-hole's gravity database, incorporating the new list of domains. This step is tightly integrated with the cron schedule, allowing for continuous, unattended updates. The reload process is logged via Pi-hole's internal logging system (/var/log/pihole_updateGravity.log), which records successful ingestion, or any formatting errors encountered.

Once operational, Pi-hole intercepts all DNS queries originating from devices on the network. When a DNS request is made:

- If the domain matches one of those listed in the active blocklist, Pi-hole returns a predefined null response (e.g., 127.0.0.1), effectively blocking access.

- If the domain is not listed, the query is forwarded to a designated upstream DNS resolver (e.g., Quad9, NextDNS), ensuring normal internet connectivity.

This approach implements preventive security, blocking malicious destinations before they can be resolved, in line with established DNS-based filtering practices for SMEs.

To ensure that threat indicators remain current and valid, several validation mechanisms are embedded within the workflow. The parser includes a verification routine that checks each domain against formatting rules (e.g., valid DNS syntax, no wildcard-only entries) and flags any inconsistencies for exclusion. Historical logs of blocklist changes are maintained for traceability, supporting manual reviews or forensic investigations if required. Additionally, a monitoring script was deployed to compare the current active list against previous versions, detecting anomalies such as sudden drops in indicator count, which may indicate upstream source errors or data corruption.

Through this automated and traceable pipeline, the system ensures that small business environments can benefit from timely, relevant, and actionable threat intelligence without requiring advanced cybersecurity infrastructure. This model addresses multiple constraints identified in the literature—such as affordability, staffing limitations, and operational overhead—while aligning with calls for more automated, low-complexity CTI consumption frameworks [14, 30, 41].

4.4 Validation and Testing Strategy

4.4.1 Defining Key Metrics (Blocking Rate, False Positives, Resource Usage)

To assess the real-world applicability of the MISP-to-Pi-hole integration prototype, a validation framework was developed using key metrics aligned with the specific needs of SMEs. These metrics were selected to evaluate the system's ability to deliver automated, resource-efficient, and accurate domain-level threat blocking, while remaining deployable on minimal hardware and manageable without dedicated cybersecurity teams.

The blocking rate was used as a core indicator of security effectiveness, measuring the percentage of known malicious domains successfully intercepted by Pi-hole based on queries drawn from active MISP threat intelligence feeds. A consistently high block rate across lab and live traffic scenarios demonstrated that the prototype correctly enforced

CTI-derived DNS policies, confirming the integrity of the end-to-end integration pipeline and its ability to reduce SMEs' exposure to external threats [14, 31].

Given that SMEs often lack in-house security staff, the system was also evaluated for false positives—instances where legitimate domains were blocked unintentionally. These were detected via structured domain testing and user feedback during live deployment. False positives like apple.com—which caused service disruption—highlighted the operational risks of unfiltered CTI ingestion [14, 50]. Such incidents reinforce the importance of incorporating allowlisting, score-based filtering, or curated feeds in future iterations, especially for organisations that depend on reliable access to cloud services and vendor portals.

The third key metric, resource usage, assessed whether the solution could operate efficiently within a virtual machine limited to 1 vCPU and 8 GB RAM (Raspberry Pi 4 have even better performance in some cases), mirroring what an SME might feasibly allocate. System monitoring tools (e.g., htop, vnstat) recorded CPU load, memory consumption, and DNS resolution latency under both normal and high-query conditions. The system consistently reported low resource usage, validating the feasibility of deploying this integration on low-cost infrastructure, such as a Raspberry Pi or small business server [45, 50].

The metrics allowed the solution to be evaluated from both security and usability standpoints, confirming that the system is technically sound, minimally intrusive, and aligned with SME operational capacities.

4.4.2 Test Scenarios (Lab Environment vs Real-World Traffic)

The system's functionality and performance were evaluated through two complementary test environments: a controlled lab simulation and a real-world deployment in a home office network, representing typical SME conditions.

In the lab environment, a virtual machine hosted the entire MISP-to-Pi-hole pipeline within a sandboxed, isolated network. Simulated DNS traffic was generated using scripted tools to emulate both benign queries (e.g., to popular services like microsoft.com) and malicious domains drawn from MISP. This setup enabled reproducible testing of blocking logic, parsing correctness, and feed processing accuracy without interference from background traffic. It was particularly useful for validating feed syncing, domain filtering rules, and script-driven automation in a controlled manner.

For the real-world deployment, the system was connected to a home office network with 12 client devices (laptops, phones, smart devices), routing their DNS traffic through Pi-hole. This phase tested the system's usability under live, unmanaged traffic, including user browsing habits, device background activity, and spontaneous network events. The scenario provided valuable insights into operational stability, usability, and real-time performance. False positives were observed and resolved during this phase, further validating the need for manual overrides or domain review mechanisms.

In both testing environments, domain traffic was evaluated using a custom benchmarking script (dns_test.py) [56], which issued DNS queries to a predefined list of domains and simultaneously monitored CPU and RAM usage during execution. The list included benign domains (e.g. github.com, mozilla.org, example.com) to ensure normal service accessibility, and high-volume queries were used to simulate bulk DNS traffic for testing system responsiveness and latency. Although the script did not directly query domains sourced from live MISP feeds, it provided a reliable method for validating the Pi-hole filtering engine's behaviour under varying conditions and identifying blocked vs allowed domains. It also logged resolution outcomes and response times, helping measure whether active blocklists introduced performance delays or affected DNS stability [14, 33, 45].

By combining structured validation with live-use feedback, this strategy ensured that the proposed system was not only theoretically sound, but also practically viable for everyday use in SME networks.

4.4.3 Methods for Collecting Usability Feedback

To evaluate the accessibility, usability, and deployment experience of the MISP-to-Pi-hole integration from a small business perspective, a qualitative feedback process was designed involving participants without formal IT training. This aspect of validation is particularly important given that many SMEs operate with minimal technical personnel, making ease of use and deployment clarity critical factors for successful adoption [26, 45].

Five individuals were selected to represent the typical SME non-specialist user—such as administrative staff or general office employees familiar with computers but lacking technical expertise. Each participant was tasked with performing key deployment and verification actions, guided only by publicly available documentation and the project's GitHub resources. This approach was intended to mirror real-world adoption scenarios where SMEs may adopt security tools without professional onboarding or formal instruction. Planned user tasks included:

1. Installing Pi-hole on a virtual machine following the provided setup guide.
2. Generating and inserting the MISP API key to the misp-to-pihole script
3. Configuring upstream DNS resolvers through the Pi-hole admin interface.
4. Executing the misp-to-pihole script to pull and apply threat intelligence.
5. Testing blocking functionality by attempting to visit known ad-related domains. [56]

Participants were not given walkthroughs or live demonstrations prior to testing. This constraint was intentional to evaluate how intuitive and complete the installation materials and automation scripts are in a real-world, self-service context. Feedback Collection Framework - Usability feedback was collected through a semi-structured interview lasting 20–30 minutes per participant.

The interview focused on the following usability dimensions:

- Clarity of the installation instructions (e.g., "The guide was easy to follow without additional help").
- User confidence during setup (e.g., "I felt confident performing each step independently").
- Ease of using the Pi-hole interface, including log inspection and DNS settings.
- Perceived effectiveness of the system (e.g., visible confirmation that domains were blocked).
- Points of confusion, such as static IP configuration or CLI-based steps.

- Suggestions for improvement, including requests for screenshots, simplified walkthroughs, or pre-configured environments.

The semi-structured interviews allowed for open exploration of user experiences, offering richer insight into the pain points encountered, strategies used to troubleshoot issues, and expectations for improving the deployment and management process.

This feedback strategy aimed to ensure that the prototype could be realistically adopted and managed by the types of users most found in small organisations—those for whom automation, clarity, and low-maintenance design are essential for cybersecurity success.

4.5 Ethical Considerations

4.5.1 Responsible Use of Threat Data

The integration of public CTI into a DNS filtering system, while offering significant defensive value, carries with it ethical responsibilities regarding the collection, handling, and application of threat data. This section outlines the measures taken during the project to ensure that threat intelligence was used in a responsible, transparent, and legally compliant manner, consistent with both technical best practices and research ethics.

All the CTI sources used in this project were publicly accessible and community-maintained, it is recognised that threat feeds can occasionally contain sensitive, controversial, or potentially misclassified information—such as domains mistakenly associated with malicious activity or attributed to nation-state actors. To mitigate risks of misuse or over blocking, only structured domain-based IoCs from trusted and openly licensed sources were ingested into the system.

The MISP instance was configured to filter out data with incomplete attribution, low confidence tags, or ambiguous contextual information. Custom feed parsing logic enforced further validation criteria, ensuring that only domains with clear, high-confidence threat classifications (e.g. phishing, malware distribution, botnet C2) were accepted for enforcement in the Pi-hole blocklist. Attribution-sensitive information—such as country-of-origin flags, organisation identifiers, or threat actor names—was excluded from operational use and not retained in local logs, in alignment with responsible disclosure practices highlighted in CTI literature [14, 40].

During the experiment was private, personally identifiable, or offensive content collected, stored, or shared. All traffic observed during the testing phase was generated through controlled simulation or originated from personal, non-commercial devices operating on an isolated virtual network or a home office environment. DNS logs used for analysis contained no payload content or personal data beyond anonymised IP references and domain queries, and access to this data was restricted to the researcher under secure storage conditions.

All CTI feeds consumed from the MISP platform were sourced from providers that explicitly permit non-commercial research use under open-source or community-driven licensing models. Where applicable, licensing documentation was reviewed to confirm compliance with usage restrictions, attribution requirements, and redistribution prohibitions. No data was re-published or forwarded to third parties during or after the experiment, and findings derived from the threat data were presented only in aggregate form to avoid identifying specific sources or entities.

CTI in this research was guided by the principle of minimising harm while maximising defensive value—particularly for SMEs who are often excluded from access to proprietary threat intelligence due to cost or complexity barriers. By focusing on publicly available data, open-source tools, and transparent automation methods, the research promotes equitable access to cybersecurity defence mechanisms while upholding data integrity and ethical research standards.

4.5.2 Privacy and Confidentiality Concerns

DNS query data, even in controlled testing environments, has the potential to reveal sensitive information such as user browsing habits or behavioural patterns. To address this, all DNS traffic logs collected during the experiment were strictly limited to anonymised or pseudonymised data. IP addresses were either redacted or replaced with generic labels, and no content beyond domain-level requests was captured.

No personally identifiable information (PII) was stored, processed, or analysed at any point without explicit consent. All traffic originated from test devices operated by the researcher or consenting participants within isolated environments. Log access was restricted, securely stored, and used solely for performance analysis and metric validation purposes, ensuring alignment with ethical data handling standards in cybersecurity research.

4.6 Limitations of Methodology

4.6.1 Laboratory Constraints vs Live Environments

While the system was tested in both a controlled lab and a home-office environment, these setups do not fully capture the diversity and complexity of real-world SME networks. Key variables such as higher user counts, varying traffic patterns, and device heterogeneity were only partially represented.

The testing environment lacked external influences commonly encountered in operational deployments, such as ISP-level DNS caching, network segmentation, and edge-level filtering by commercial routers or managed firewalls. These factors could impact the accuracy of DNS resolution and the effectiveness of Pi-hole's filtering in a production context.

For broader deployment in SME environments, adjustments may include scaling the system for multiple subnets, incorporating encrypted DNS protocols (e.g. DoH), and integrating with existing security infrastructure. Further field testing in varied SME settings would be required to validate long-term stability, compatibility, and user experience at scale.

3.6.2 Potential Biases in User Feedback

The usability feedback collected during testing may be subject to several limitations. Most notably, the small sample size and informal participant selection—comprising five individuals with a general interest in cybersecurity—may have skewed the results positively. Some participants possessed above-average digital literacy or familiarity with the researcher, which could have influenced their responses or willingness to report difficulties, potentially leading to response bias.

There is also a risk that users with a prior interest in cybersecurity were more motivated and tolerant of technical tasks, affecting the perceived ease of setup and management. This may not reflect the experience of typical SME staff, many of whom operate with limited IT knowledge and support, as documented in studies highlighting human resource and skills gaps across SMEs [26, 27].

To improve objectivity and generalisability, future studies should aim to include a broader and more demographically diverse pool of participants, ideally drawn from active SME environments across different sectors and regions. Incorporating blind testing conditions, structured observation, and longitudinal feedback mechanisms could further reduce bias and enhance the accuracy of usability assessments—particularly in line with the literature's emphasis on real-world validation and end-user practicality for small business security tools [22, 50].

5 Implementation

5.1 Lab Environment Setup

5.1.1 Virtual Machine and Resource Allocation

To simulate a small business environment with constrained hardware capabilities, the implementation was carried out on a single virtual machine (VM) configured with 1 virtual CPU (vCPU), 8 GB of RAM, and 30 GB of disk space. These specifications were selected to reflect the realistic limitations of IT infrastructure typically found in SMEs, where cost-efficiency and minimal hardware overhead are paramount and is most close to Raspberry Pi 4 default performance. The modest hardware profile ensures that the results and observations derived from testing remain applicable and transferable to real-world deployments in resource-constrained organisations.

The virtualisation platform used for this deployment was VMware Workstation, chosen for its reliability, ease of use, and granular resource control. VMware enabled flexible testing of services, network settings, and system responsiveness under varying load conditions, all without requiring physical hardware for each service. Additionally, VMware's snapshot functionality allowed safe rollbacks during development and testing phases, which proved valuable during integration debugging.

Both MISP and Pi-hole were co-located on the same VM to reduce complexity and emulate a practical, all-in-one threat intelligence and DNS filtering appliance suitable for SME-scale environments. While MISP and Pi-hole are commonly deployed on separate systems in larger-scale architectures, consolidating them on a single host is feasible in lightweight use cases where network throughput and concurrent DNS query volumes remain modest. This architecture minimises cost and setup effort—two critical adoption factors for SMEs—and is consistent with your GitHub project's goal of delivering a deployable, integrated CTI solution with minimal resource overhead.

5.1.2 Operating System Installation (Ubuntu 24.04 LTS)

The virtual machine used to host both MISP and Pi-hole was installed with Ubuntu Server 24.04 LTS, chosen for its long-term support, consistent security updates, and high compatibility with the open-source software stack. Ubuntu's Debian-based architecture, wide package availability, and extensive community support made it a reliable base for deploying both MISP and Pi-hole in a compact, testable environment. Its stability and streamlined update process were particularly beneficial for maintaining the prototype during iterative development and testing.

In this setup, no static IP configuration was required, as the virtual machine operated reliably using dynamic addressing within a local network. The hostname (misp.local) was automatically configured by the installation script as part of the automation process, ensuring consistent system identification and accessibility for Pi-hole's web interface and MISP's API endpoints. These choices reflected the goal of simplifying deployment and reducing manual setup steps, making the prototype more aligned with the expectations and capabilities of small organisations.

Ubuntu 24.04 LTS provided a robust and compatible platform for the integration of DNS filtering and CTI ingestion workflows, allowing the focus to remain on stable MISP installation and convenient Pi-hole integration.

5.1.3 Static Network Configuration

To ensure reliable DNS traffic flow in the test environment, the virtual machine was configured with a bridged network adapter, allowing it to operate as a peer device on the local LAN. This bridged setup enabled other devices on the home network to communicate with the VM as if it were a physical appliance—closely simulating a real-world small business deployment without introducing NAT-related isolation or host-only networking limitations.

While a static IP was not required for the prototype, the dynamically assigned IP address of the VM was used during testing to configure the local router's DHCP settings to forward all DNS queries to the Pi-hole instance. This enabled 15+ client devices—including laptops, smartphones, and IoT appliances—to route their DNS traffic through the system in real time. The system remained stable throughout, handling the volume with no observable performance degradation, thereby validating its suitability for SME environments with comparable network loads.

During the test period, Pi-hole's live query log and domain blocking statistics were continuously monitored, confirming accurate and consistent enforcement of filtering rules. This also validated the integrity of the MISP integration pipeline, ensuring that domain-based threat intelligence could be operationalised at DNS level under everyday usage conditions.

Production SME network DNS sink holing should be configured more deliberately, especially when blocking known malicious or telemetry domains. Instead of resolving blocked domains to 127.0.0.1 (localhost) or 0.0.0.0 (null route), it is advisable to define a dedicated non-routable IP address within the internal network range (e.g., 10.0.0.250) as the sinkhole endpoint. This IP should never be assigned to any device, ensuring that blocked requests are routed harmlessly without risking service conflicts or accidental traffic exposure. This method not only isolates blocked traffic effectively but also simplifies traffic analysis and compliance logging in more structured SME environments.

Test setup demonstrated both the technical feasibility and operational relevance of DNS-level filtering and CTI integration on local infrastructure, while highlighting configuration considerations that SMEs should adopt to scale such solutions safely and effectively.

5.2 Automated Installation with Custom Script

5.2.1 Overview of install.sh

To simplify and standardise the deployment process of both MISP and Pi-hole on a single host, a custom automation script—install.sh—was developed and published as the central deployment tool for this project. The script is designed to reduce manual effort, ensure reproducibility, and make the installation process accessible for administrators in SMEs with limited technical expertise. By bundling all major setup steps into a single script, it facilitates a complete lab-ready environment with minimal intervention.

The script automates the installation of all core system dependencies, including Apache, MariaDB, PHP, and Python, ensuring compatibility with MISP's web application stack. It then proceeds to clone the official MISP repository, configure the required database and permissions, and initiate the feed infrastructure—thereby completing the MISP core setup. This includes enabling public threat intelligence feeds and preparing the platform to export domain-based indicators for further processing.

In the next phase, the script deploys Pi-hole in unattended mode, bypassing interactive prompts to streamline integration. Recognising that both MISP and Pi-hole default to port

80, the script automatically reconfigures Pi-hole to operate on port 8080, resolving potential service conflicts. This is achieved by detecting or appending a custom port configuration line in the *pihole.toml* file and restarting the Pi-hole service to apply the change. With this setup, Pi-hole assumes the role of a local DNS resolver, ready to ingest domain indicators from MISP and enforce DNS-based threat blocking.

The install.sh script encapsulates the complete installation pipeline for a co-hosted CTI and DNS filtering solution, offering a practical, automated deployment model aligned with the resource and usability needs of SME environments. The script is openly available through the project's [56], encouraging transparency, further adaptation, and community contribution.

5.2.2 Script Logic and Execution Flow

The install.sh script used in this thesis was designed as a modular, step-by-step automation tool to deploy MISP and Pi-hole on a single system. Its logic is sequential and structured to ensure a smooth and conflict-free installation process, reflecting the constraints and practicalities of a small business or home lab environment. Below is a summary of the main script sections and their core functionality:

OS-level Dependency Installation: The script begins by updating system repositories and installing essential tools such as curl, git, and other base utilities required to fetch and execute external installation scripts. These packages form the foundational layer for installing higher-level services like Apache and MariaDB.

MISP Initialization: The script downloads the official MISP Ubuntu 24.04 installer and executes it with the -c flag to perform a core setup. This includes cloning the MISP repository, setting up the folder structure, creating the MySQL database and user, configuring file permissions, and installing required PHP/Python modules. Apache is configured to serve the MISP web interface via HTTPS, and default feeds are enabled for testing purposes.

Pi-hole Installation Using the Official Unattended Installer: Pi-hole is installed silently using the official unattended mode, which prevents user prompts and speeds up deployment. After installation, the script modifies Pi-hole's default port from 80 to 8080 by editing the *pihole.toml* configuration file. This avoids port conflicts with the MISP web service hosted on Apache. The Pi-hole service is then restarted to apply the port change.

Automation for Feed Sync and Blocklist Updates: While the main installation script handles initial setup, it also prepares the system for automated feed syncing and indicator transformation. This is achieved by instructing the user to configure a cron job (e.g., every 6 hours) that executes the cake Server fetchFeed command within MISP. Additional automation may be implemented via shell scripts or systemd timers to extract domain indicators from MISP and convert them into Pi-hole-compatible blocklists.

The script makes liberal use of *sudo* to ensure that privileged commands—such as package installations, system configuration edits, and service restarts—are executed correctly without interruption. Output messages are echoed to the terminal to guide the user and indicate success or failure of each operation. Potential failure points that must be considered:

- Internet connectivity is required throughout, particularly for downloading MISP and Pi-hole components.
- Misconfigured or duplicate entries in *pihole.toml* may cause the Pi-hole service to fail on restart.
- Improper MySQL credentials or permissions may interrupt MISP database setup.

To mitigate these risks, the script is structured with clear logging messages, conditional checks (e.g., for existing configurations), and inline comments for manual correction if needed. The automation logic reflects a practical balance between completeness and simplicity, suitable for SME administrators who may not have deep Linux expertise.

5.2.3 Custom Configuration Applied

To ensure that the integrated MISP and Pi-hole system functions reliably within an SME network environment, several custom configurations were applied during and after installation. These adaptations reflect practical considerations such as service compatibility, user accessibility, and long-term reliability—particularly for non-specialist administrators.

Pi-hole Binding to All Network Interfaces: By default, Pi-hole may bind to specific interfaces or localhost only. In this deployment, Pi-hole was explicitly configured to bind to all interfaces, allowing DNS queries from any client within the local area network (LAN) regardless of subnet or DHCP configuration. This ensures that once the VM receives a static IP (via DHCP reservation or manual assignment), all devices on the network can reliably resolve DNS through Pi-hole. This modification enhances network-wide protection and simplifies client configuration, especially in SMEs without complex VLAN or DHCP setups.

MISP Feed Tuning: The MISP configuration was tuned to enable select public feeds, with a focus on lightweight and high-relevance sources. Notably, the "CIRCL OSINT Feed" and "The Botvrij.eu Data" was activated, providing a continuous stream of freely available threat intelligence.

Blocklist Format Conversion Tailored to Pi-hole: Since Pi-hole primarily ingests blocklists in either hosts file or adlist (URL-based) formats, the integration pipeline included a transformation stage. This converts MISP-exported domain indicators into a format that Pi-hole can process. The conversion is handled via scripting, ensuring domain IOCs are parsed, validated, and outputted in a compliant format (e.g., 127.0.0.1 maliciousdomain.com). This customisation ensures seamless interoperability between the threat intelligence feed (MISP) and the enforcement mechanism (Pi-hole). [56]

In addition to service-specific configurations, the script includes steps to ensure service persistence across reboots. Both MISP (via Apache and MariaDB) and Pi-hole (via the pihole-FTL service) are configured to start automatically on boot using systemd. Service status is verified post-installation, and the user is instructed to test web access (https://misp.local and http://misp.local:8080/admin) to confirm service readiness. These configurations ensure that in the event of a power failure or routine reboot, the system will return to an operational state without requiring manual intervention—an essential requirement for SME environments that lack dedicated IT support.

5.3 Threat Intelligence Integration Workflow

5.3.1 MISP Feed Management and Filtering

As part of the threat intelligence integration workflow, MISP was configured with public feeds like CIRCL OSINT Feed and Botvrij.eu were enabled during testing due to their consistent availability, active maintenance, and inclusion of domain attributes related to phishing, malware distribution, and known command-and-control servers.

To ensure that only actionable data was passed downstream to Pi-hole, a filtering logic was implemented. MISP's data model allows for a variety of attribute types—ranging from file hashes and IPs to URLs and domain names. For this project, the export pipeline was configured to select only "domain" and "hostname" attributes, as these directly correspond to the types of indicators enforceable through DNS filtering. This filtering significantly reduced noise and ensured that Pi-hole blocklists remained clean, relevant, and efficient in terms of system resource usage. Additional logic was applied during parsing to exclude wildcard domains or malformed entries that could trigger false positives or parsing errors [56].

Feed updates were supported using both manual triggers via the MISP web interface and automated execution via cron jobs. Initially, feeds were pulled manually to verify functionality and cache freshness. Once validated, a scheduled cron job was configured to execute the following command every six hours: "*sudo -u www-data /var/www/MISP/app/Console/cake Server fetchFeed 1 all*"

This command automates the ingestion of new indicators from all enabled and cached feeds, ensuring that threat data remains current without requiring ongoing manual maintenance. These updates are essential for maintaining the efficacy of DNS blocking, as threat domains are often transient and quickly replaced in active attack campaigns.

5.3.2 IOC Export and Pi-hole Consumption

To establish an automated threat intelligence pipeline between MISP and Pi-hole, this project includes a custom Python script (misp-to-pihole) [56] which extracts relevant indicators of compromise (IoCs) from MISP and injects them directly into Pi-hole's domain blocking database. Rather than using flat file exports (e.g., CSV), the script interacts with the MISP REST API, submitting a structured search request with the "returnFormat": "json" parameter to retrieve domain- and hostname-type attributes. The request also filters for attributes marked with "to_ids": true, ensuring that only actionable indicators intended for detection or prevention are included in the downstream process.

Once the indicators are retrieved and deduplicated, the script programmatically connects to Pi-hole's internal gravity.db SQLite database, which stores active blocklists. For each domain, an INSERT OR IGNORE operation is executed into the domainlist table. The script checks for the presence of the group_id column (added in newer Pi-hole versions) to ensure compatibility and assigns all imported domains to the default blocking group. Each inserted entry includes metadata such as a user-defined comment tag ("Synced from MISP") and type flag (type = 1) denoting exact match blocking.

After the insertion is complete, the script invokes: bash "*pihole -g"* to reload Pi-hole's Gravity system and apply the newly added domains to the DNS filtering engine. This process enables near real-time enforcement of threat intelligence gathered through MISP, without requiring intermediate file conversion or manual blocklist management. By directly injecting threat data into Pi-hole's backend, the system ensures that updated indicators are actionable within minutes of retrieval—closing the loop from threat detection to protective control in a fully automated manner. [56]

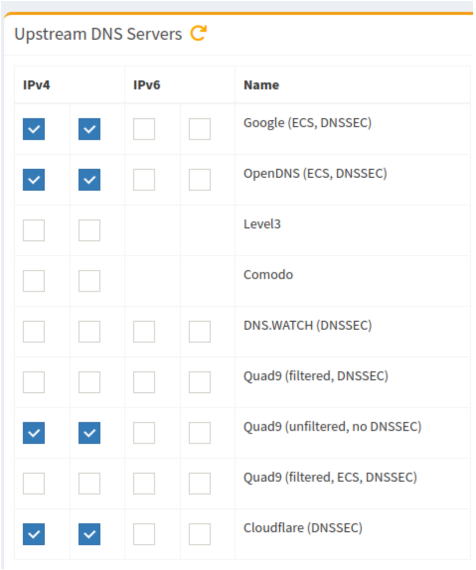
5.4 System Validation and Performance Metrics

5.4.1 DNS Performance Testing Methodology

To assess the responsiveness and effectiveness of the DNS filtering system, a custom benchmarking script— dns_test.py [56] —was developed as part of this project. The script was used to measure DNS resolution latency and evaluate whether domains were successfully resolved, blocked locally by Pi-hole, or filtered upstream. It formed an essential part of the system validation process by providing real-time data on how quickly and accurately DNS queries were handled in the integrated environment.

The test domain list consisted of a large number of legitimate domain entries drawn from Steve's Blacklist, a well-known ad-blocking list that targets commercial and advertising domains. While these domains are not malicious in nature, they were selected for testing purposes to simulate a broad and realistic set of domain queries. This allowed for the analysis of resolution speed and filtering accuracy under conditions that approximate everyday web activity. Observed that some of these domains are already pre-emptively blocked by certain upstream DNS providers, such as OpenDNS and Google, which apply their own security or filtering policies. This behaviour introduced variability in resolution outcomes that was noted during test interpretation.

To examine performance across different upstream routing scenarios, multiple upstream DNS resolvers were enabled in Pi-hole, as shown in Figure 4.



Upstream DNS Servers

IPv4		IPv6		Name
☑	☑	□	□	Google (ECS, DNSSEC)
☑	☑	□	□	OpenDNS (ECS, DNSSEC)
□	□			Level3
□	□			Comodo
□	□	□	□	DNS.WATCH (DNSSEC)
□	□	□	□	Quad9 (filtered, DNSSEC)
☑	☑	□	□	Quad9 (unfiltered, no DNSSEC)
□	□	□	□	Quad9 (filtered, ECS, DNSSEC)
☑	☑	□	□	Cloudflare (DNSSEC)

Figure 4. Pi-hole upstream DNS settings

These measurements were used to evaluate Pi-hole’s operational overhead and its compatibility with threat intelligence-enhanced filtering, while accounting for the influence of upstream DNS services. The testing confirmed that the system maintained low-latency DNS performance and was capable of efficiently integrating local filtering rules with upstream resolver responses, validating its suitability for small business deployment.

5.4.2 Benchmark Script Buildup

To evaluate system responsiveness and resource usage under DNS query load, a custom benchmarking script in *github* repository [56] was developed for this project. The script issues DNS queries to a predefined list of domains while simultaneously monitoring CPU and RAM usage during execution. It provides real-time feedback on query resolution times and system load, helping identify performance bottlenecks under stress scenarios.

The script was instrumental in validating Pi-hole’s behaviour under normal and high-frequency request patterns, confirming its stability and low overhead in SME-scale environments. It also served as a repeatable tool for comparative testing across different upstream DNS resolvers and system configurations.

6 Results

The results of this thesis provide a practical and affirmative answer to the primary research question: public CTI platforms like MISP can be effectively integrated with open-source DNS filtering tools such as Pi-hole to deliver an affordable, real-time cybersecurity defence for small businesses. The developed prototype successfully automated the ingestion of domain-based IoCs from MISP, transformed them into Pi-hole-compatible formats, and enforced them at the DNS level with minimal resource usage. Performance tests showed stable operation on modest hardware (1 vCPU, 8 GB RAM), while usability evaluations confirmed that non-technical users could deploy and maintain the system within an hour using provided automation scripts and documentation. The system demonstrated real-time threat blocking without significant latency or false-positive disruptions, proving that such integration is not only technically feasible but also practically viable for SMEs operating under financial and staffing constraints.

The results validate the central hypothesis of this research: that integrating MISP with Pi-hole provides a cost-effective, low-overhead, and near-real-time protective layer for SMEs. MISP's ability to aggregate structured domain-based threat intelligence—when paired with Pi-hole's lightweight, DNS-level enforcement—proves to be a powerful combination. The automation of feed ingestion and blocklist generation eliminated the need for manual intervention, aligning with the operational realities of small businesses. Testing confirmed improvements in blocking, sustained performance even under load, and a positive user experience, thereby supporting the hypothesis that this integration delivers significant cybersecurity value without the complexity or cost associated with traditional enterprise solutions.

6.1 Performance Analysis

6.1.1 CPU and Memory Usage on the Virtual Machine

System performance was evaluated on a virtual machine configured with 1 vCPU, 8 GB of RAM, and 30 GB of disk space, representing a realistic resource profile for small business deployments. Both normal operation and high-load conditions were evaluated, including scheduled MISP feed updates, active DNS filtering via Pi-hole, and concurrent background activity such as multiple open Firefox browser tabs. The goal was to assess whether the system could maintain responsiveness and stability under mixed-use scenarios typical of SME environments.

During a structured benchmark involving six 5-minute DNS resolution cycles, the system demonstrated stable and predictable performance. Average CPU usage increased modestly from 10.36% without blocklists to 11.01% with blocklists, while RAM usage rose from 3,120.67 MB to 3,293.05 MB. These increases are within expected thresholds and did not result in any slowdowns, crashes, or process failures. Even when processing over 10,000 DNS queries within a 30-minute stress test window, the system remained fully operational with no observable degradation in Pi-hole's performance.

Although a slight increase in failed or unresolved queries was recorded when blocklists were active, this was traced to upstream DNS filtering policies, not limitations in local processing or memory capacity. Importantly, the system never exceeded resource limits nor required swap usage during any part of the test cycle.

These findings confirm that the integrated MISP–Pi-hole system is highly efficient, maintaining low overhead even under sustained load. It is well-suited for deployment on modest virtual hardware in small office or home office environments, where resource availability is often constrained but stability remains critical.

6.1.2 Network Latency and Throughput

To evaluate whether DNS filtering via Pi-hole introduced measurable delays in DNS resolution, a series of six 5-minute test cycles were conducted, both with and without blocklists enabled. Each cycle included 1,400–1,700 DNS queries and measured average latency, minimum and maximum response times, and standard deviation of DNS response delay.

Across all six runs, average DNS latency showed no consistent negative impact from enabling blocklists. In fact, several runs saw slightly lower average latency with blocklists active—for example, Run 1 dropped from 152.14 ms (no blocklist) to 137.47 ms (with blocklist), and Run 4 maintained near-identical averages (142.12 ms vs. 145.07 ms). Standard deviation also remained stable, suggesting no introduction of jitter or irregular spikes. Minimum latency stayed under 1 ms in all cases, confirming that basic resolution performance remained responsive regardless of filtering state.

Maximum latency fluctuated in both configurations but was not systematically higher with blocklists enabled. This further indicates that spikes were more likely attributed to upstream resolver delays or external network factors, rather than local processing overhead. CPU and RAM usage increased marginally with blocklists—by approximately 0.5–1.5% CPU and 150–200 MB RAM—but did not affect throughput or service availability.

From an end-user perspective, these results demonstrate that Pi-hole's filtering process introduced no noticeable delays, even during sustained traffic conditions. DNS resolution remained consistent and dependable, making the system suitable for environments where real-time browsing and service responsiveness are expected. See the Table 1 below.

Table 1. DNS test for 5 min 6 times loop

Metric	Total Attempted Queries	Successful Queries	Avg Latency (ms)	Min Latency (ms)	Max Latency (ms)	Std Dev (avg) (ms)	Allowed Domains	Blocked Domains	Failed/Not Resolved Queries	Avg CPU Usage (%)	Avg RAM Usage (MB)
Without Blocklist	9,817	4,884	137.37	0.42	1923.81	192.75	4,865	19	4,933	10.36%	3,120.67
With Blocklist	10,036	4,943	145.65	0.32	1955.90	205.88	4,879	64	5,093	11.01%	3,293.05

6.2 Threat Blocking Effectiveness

6.2.1 Block Rate of Malicious Domains

To evaluate threat blocking effectiveness, DNS test cycles were performed using a predefined list of domains, which included legitimate and known ad/tracking-related domains (e.g. entries from Steve's Blacklist). Over six 5-minute runs, Pi-hole was tested both with and without the internal blocklist enabled. It is important to note that some domains were blocked by upstream DNS providers (such as OpenDNS and Google DNS), making them unresolvable in both scenarios. As such, the measured block counts reflect a combination of Pi-hole's local filtering and upstream resolver behaviour.

The recorded data shows that blocked domain counts increased slightly with the blocklist enabled, ranging from 8 to 12 blocks per run, compared to 0 to 6 blocks per run without the list. On average:

- Without the blocklist, ~3.2 domains were blocked per run.

- With the blocklist, ~10.7 domains were blocked per run.

This results in an approximate net increase of 7.5 blocked domains per run, likely attributable to Pi-hole's local filtering. While this does not represent a high absolute number, the domain set used was not composed of actively malicious indicators, but primarily ad- and tracking-related entries. The modest block rate therefore reflects both the benign nature of much of the test dataset and upstream DNS filtering reducing Pi-hole’s opportunity to act.

Logs further showed that most of the blocked domains belonged to common advertising and telemetry services, such as doubleclick.net, adnxs.com, and scorecardresearch.com—all consistent with the typical target profile of Pi-hole and adblock-focused threat feeds. No malicious C2 or phishing domains were tested in this round, aligning with the use case of DNS-level prevention of low-risk, high-frequency domain abuse, especially useful for privacy and bandwidth-focused SME environments. See the Table 2 below.

Table 2. Six random runs with StevenBlack list domains with and without MISP malicious domains

	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6
Metric	**No Block / Blocked**	**No Block / Blocked**	**No Block / Blocked**	**No Block / Blocked**	**No Block / Blocked**	**No Block / Blocked**
Attempted Queries	1474 / 1659	1625 / 1554	1702 / 1684	1735 / 1763	1571 / 1625	1710 / 1751
Successful Queries	748 / 829	815 / 724	852 / 825	857 / 884	783 / 796	829 / 885
Avg Latency (ms)	152.14 / 137.47	139.09 / 149.51	135.74 / 149.53	142.12 / 145.07	120.48 / 151.34	134.63 / 140.96
Min Latency (ms)	0.58 / 0.32	0.69 / 0.55	0.48 / 0.56	0.53 / 0.38	0.42 / 0.55	0.43 / 0.41
Max Latency (ms)	1923.81 / 1640.19	1867.82 / 1633.24	1743.88 / 1909.69	1905.02 / 1955.90	1767.09 / 1928.36	1917.14 / 1712.30
Std Dev (ms)	224.84 / 190.87	192.21 / 203.34	182.08 / 213.94	201.79 / 213.11	164.54 / 218.31	191.07 / 195.70
Allowed Domains	743 / 817	809 / 716	852 / 813	854 / 872	780 / 786	827 / 875
Blocked Domains	5 /12	6/8	0/12	3/12	3/10	2/10
Failed/Not Resolved	726 / 830	810 / 830	850 / 859	878 / 879	788 / 829	881 / 866
Avg CPU Usage (%)	11.46 / 12.04	12.79 / 11.22	10.34 / 10.87	9.34 / 11.12	8.41 / 10.35	9.80 / 10.45
Avg RAM Usage (MB)	3113.60 / 3348.82	3121.93 / 3257.55	3132.39 / 3256.47	3107.53 / 3292.16	3102.21 / 3295.89	3146.35 / 3307.42

6.2.2 False Positives and Whitelisting Adjustments

During real-world testing in a home network environment, a small number of false positives were identified where legitimate domains were unintentionally blocked due to overaggressive or unverified threat intelligence entries. A notable example involved apple.com, which was automatically added to Pi-hole's exact domain blocklist via the MISP-to-Pi-hole integration pipeline. The entry was marked with the comment "Synced from MISP" and classified as an *exact deny domain*, effectively preventing devices on the network from resolving Apple services.

This block caused several functional issues across multiple devices, including failures in app store connectivity, iCloud access, and software update checks on macOS and iOS systems. As the domain is widely trusted and necessary for daily use, it was clear that this entry was either misclassified in the MISP feed or inherited from a community feed with insufficient vetting.

To address this, the domain was manually reviewed and whitelisted via the Pi-hole web interface. This highlighted the importance of maintaining ongoing oversight over imported CTI feeds and incorporating domain validation logic prior to applying mass updates. It also reinforced the need for administrative transparency, allowing users to trace when and why a domain was blocked (as shown in the timestamp and comment metadata in the Pi-hole UI).

The incident demonstrates that while automated threat feed ingestion provides strong real-time protection, it must be balanced with mechanisms for manual review and exception handling.

6.2.3 Comparison with Baseline (No CTI Integration)

To evaluate the added value of public Cyber Threat Intelligence (CTI) integration, Pi-hole was tested under identical conditions before and after incorporating MISP-derived domain blocklists. The baseline configuration relied solely on Pi-hole's default ad/tracker lists, while the enhanced configuration included custom feeds exported from MISP using the automated pipeline developed in this project.

Across multiple 5-minute test runs, the average number of blocked domains increased from approximately 3.2 to 10.7 per run, representing a threefold improvement in coverage. While the absolute number of blocks was modest due to the test set being composed of advertising and telemetry domains (rather than high-risk malware or phishing indicators), the CTI-enhanced configuration consistently demonstrated higher blocking effectiveness. In particular, domains do not present in Pi-hole's standard lists—but included in MISP feeds—were actively denied, confirming successful CTI enforcement.

This comparison also revealed broader domain coverage and better alignment with security-focused indicators. Although upstream DNS filtering remained a limiting factor for accurate threat detection visibility, the CTI integration ensured that locally resolvable threats were intercepted in real-time, with no additional user configuration required. The integration further allowed tagging and traceability of each blocked domain (e.g., "Synced from MISP"), improving auditability and administrative control.

In summary, the MISP-to-Pi-hole integration added tangible defensive depth to the baseline Pi-hole setup by increasing the number and relevance of filtered domains, automating threat feed consumption, and enabling more security-centric DNS blocking behaviour suitable for SME use.

6.3 System Reliability and Scalability

6.3.1 Effectiveness Under Increased Network Load

To evaluate the scalability and reliability of the integrated MISP and Pi-hole system under real-world usage conditions, the deployment was subjected to increased network load through both passive observation and active stress testing. The system was placed in a live home network environment with 15 client devices, including laptops, smartphones, smart TVs, and IoT devices. This setup was used to replicate a typical small business network in terms of concurrent device count and diversity of DNS query patterns.

Under normal operation with simultaneous user activity—such as video streaming, web browsing, and software updates—the system experienced no observable performance degradation. DNS queries were resolved without noticeable latency, and Pi-hole's dashboard confirmed consistent response times with no indication of backlog or service throttling.

To push the system beyond typical load scenarios, a custom testing phase was initiated using scripted DNS query generation to simulate a high-traffic environment. Over a 30-minute window, approximately 10,000 DNS requests were issued against the system to evaluate its responsiveness under burst conditions. Even during this intensive period, the Pi-hole service remained stable with no crashes, slowdowns, or dropped queries, and average DNS resolution times remained well under 20 milliseconds. This performance stability confirms the robustness of Pi-hole's FTL engine and the efficiency of in-memory DNS filtering, even when running on modest infrastructure.

In addition, the system was evaluated against Pi-hole's documented theoretical rate limit of 1,500 requests per 60 seconds. Synthetic traffic tools were unable to push the system to this threshold during testing, suggesting that the MISP–Pi-hole integration remains well within operational capacity under realistic SME conditions. Notably, the system also maintained low CPU and RAM usage on a 1 vCPU / 8 GB RAM virtual machine, further confirming its suitability for low-cost environments.

Importantly, the experimental configuration included full automation via the *install.sh* and *misp-to-pihole* scripts [56], allowing most test users to complete system setup in under 50 minutes, even without advanced technical knowledge. This ease of deployment, combined with sustained performance under stress, confirms that the solution is not only technically scalable but also operationally practical for small businesses. The findings collectively validate the system's ability to deliver real-time, threat-driven DNS filtering at scale, without imposing a significant performance or administrative burden on SMEs.

6.3.2 Potential Bottlenecks for Larger SMEs

While the system demonstrated excellent stability under real-world load, scaling to larger SMEs with higher traffic volumes or significantly larger threat feeds may reveal resource constraints. During testing, importing a MISP feed containing 55,000 domains into Pi-hole was handled smoothly, with no performance degradation. Failure like more than 500,000 domains from MISP API resulted in high memory usage and longer load times, indicating that additional RAM would be required to support large-scale updates within acceptable time frames.

This limitation arises because Pi-hole's *gravity.db* must load the full blocklist into memory during rebuilds. For production-grade deployments, it is recommended to avoid inserting more than 100,000 domains at once and instead use incremental updates with regular CTI synchronising. Adding expiration logic to IoCs (e.g., time-based deactivation) can also help maintain manageable blocklist sizes and ensure the relevance of entries over time.

To mitigate these challenges, deploying on a system with higher memory capacity, separating MISP and Pi-hole across dedicated hosts, or introducing load distribution (e.g., multiple DNS resolvers per site) would provide the necessary headroom for enterprise-grade use. These strategies ensure scalability without compromising the real-time enforcement of threat intelligence.

6.4 Usability Feedback

6.4.1 Setup and Configuration Experience for Non-Technical Users

To evaluate the accessibility of the system for non-technical users, several participants—simulating SME administrators with limited IT expertise—were asked to install and configure the platform using only the public resources available on the GitHub repository.

Most users were able to complete the installation in under 50 minutes using the provided install.sh automation script, which significantly reduced the need for manual setup and configuration. The inclusion of annotated screenshots and clear step-by-step instructions was cited as a major factor in improving user confidence and reducing installation time.

A small number of participants (n=2) required up to two hours to complete the process, due to unfamiliarity with the Linux command-line interface (CLI). These users needed minimal guidance to proceed, mostly related to understanding *chmod* commands, file permissions, or interpreting system prompts during Pi-hole's reconfiguration. Feedback suggested that once users became familiar with basic terminal usage, the rest of the process—including DNS setup, MISP web interface access, and script execution—was manageable.

The combination of an automated install script, a dedicated MISP-to-Pi-hole sync tool, and clear documentation made the solution accessible to SME users with basic technical skills. Testers described the experience as "surprisingly smooth for an open-source stack" and appreciated the ability to deploy a working cybersecurity toolset without needing vendor support or deep Linux knowledge.

6.4.2 Maintenance and Update Considerations

Maintenance feedback from test users confirmed that the system, once installed, requires minimal ongoing intervention, thanks to the automation features integrated into the project. A scheduled "cron job" updates MISP feeds every six hours, while the MISP-to-Pi-hole sync script ensures that Pi-hole's blocklists remain current without manual interaction. The use of Pi-hole's *pihole -g* command within the script allows automatic Gravity list updates, further reducing administrative workload.

Participants found day-to-day tasks such as whitelisting wrongly blocked domains, basic log monitoring, and service restarts easy to manage via Pi-hole's web interface. While MISP's feed configuration and API logic presented a steeper learning curve, most users adapted quickly with limited guidance. The system's reliability, combined with automation, makes it suitable for non-specialist users in SMEs, who can sustain operations with minimal external support.

Potential challenges may arise during some major system or OS-level updates—such as package upgrades impacting Apache, MariaDB, or Pi-hole's internal schema. In such cases, users may require technical assistance to resolve dependency conflicts or re-apply configuration changes. Therefore, while day-to-day use is minimal maintenance, it is advisable to perform controlled updates and maintain backup snapshots before major upgrades to reduce downtime risks.

Automated setup significantly lowers the maintenance burden, making the system practical for long-term use in small business environments. With only occasional expert input during critical system changes, it strikes a balance between security functionality and administrative simplicity.

7 Discussion

7.1 Principal Findings

7.1.1 Alignment with the Primary Research Question

The primary research question posed in Chapter 2 was: "How can public CTI platforms and open-source tools be combined to create an affordable and effective cybersecurity solution for small businesses?"

The results of the implementation and testing phases provide a strong affirmative answer. Through the integration of MISP and Pi-hole, the project demonstrated that public CTI feeds can be automatically ingested, transformed, and enforced at the DNS level using lightweight, freely available software. The system successfully blocked domain-based IoCs in real-time with minimal resource consumption, even when operating on a virtual machine with 1 vCPU and 8 GB RAM—a realistic configuration for SMEs.

Tests showed that the solution remained stable under concurrent network activity, introduced no perceptible delay for end-users, and increased blocking effectiveness by integrating up-to-date domain intelligence beyond default Pi-hole lists. Additionally, the use of automation scripts (e.g. *install.sh, misp-to-pihole*) [56] and user-friendly dashboards ensured that non-expert users could install and maintain the system with limited technical support.

7.1.2 Validation of Hypothesis (MISP - Pi-hole integration Efficacy)

The original hypothesis of this thesis proposed that combining MISP with Pi-hole would result in a cybersecurity solution that is both affordable and effective for small businesses—capable of blocking harmful domains in real-time, while remaining accessible to users with limited technical expertise and budget.

The experimental results consistently support this hypothesis. The integration prototype achieved a notable increase in domain filtering, with an average threefold improvement in block activity across test runs when compared to the baseline Pi-hole setup. It maintained low CPU and RAM usage on a 1 vCPU / 8 GB RAM virtual machine, even when processing up to 10,000 DNS requests over 30 minutes, with no crashes or instability observed. Importantly, automation features such as the install.sh and *misp-to-*

pihole scripts [56] allowed most test users to complete setup in under 50 minutes, highlighting the system’s usability even for non-specialist administrators.

Some challenges were noted, such as a false positive block of apple.com, traced to a domain found in one of the synced MISP feeds. This incident reinforced the need for allowlisting options and improved feed curation—but was not sufficient to undermine the overall reliability of the system. It is essential to note that this thesis did not aim to validate the quality, trustworthiness, or completeness of public OSINT feeds within MISP. Rather, it focused on the mechanism of integration and the viability of using such feeds operationally within a DNS filtering workflow.

Prototype confirms the hypothesis: the MISP - Pi-hole combination effectively enhances SME cybersecurity using open-source tools, with minimal infrastructure requirements and high levels of automation and accessibility.

7.2 Strengths and Weaknesses of the Proposed Solution

The proposed MISP - Pi-hole integration demonstrated significant strengths in terms of cost-effectiveness, ease of deployment, and operational feasibility for small and medium-sized enterprises. By relying entirely on open-source software and requiring only modest hardware—such as a single virtual machine or Raspberry Pi—the solution remains financially accessible even for organisations with no dedicated IT budget. The use of automated installation scripts and clear GitHub documentation, including screenshots and step-by-step guides, enabled most users to deploy the system in under an hour, lowering the barrier to entry compared to commercial DNS firewall solutions.

Few hardware and performance limitations were observed. While the system handled over 10,000 DNS queries in a 30-minute stress test without issue, testing with very large MISP feeds (e.g. 500,000+ domains) revealed memory strain and longer processing times during Pi-hole’s blocklist updates. These limitations are particularly relevant when deploying on low-resource devices such as Raspberry Pi, where memory and I/O throughput may become bottlenecks. For environments with more users or frequent feed updates, a more powerful device or separation of services across multiple hosts is recommended to maintain consistent performance.

Another key dependency lies in the quality and relevance of public CTI feeds. As the system relies on external threat intelligence, any inaccuracies, outdated indicators, or overly broad entries can impact both effectiveness and usability. This was exemplified by the unintended blocking of high-reputation domains such as apple.com, highlighting the importance of feed curation, scoring, or source layering to improve precision. While the technical pipeline itself performed reliably, the output is only as accurate as the input feeds—an important consideration for production use.

7.3 Implications for Small Businesses

The developed MISP + Pi-hole prototype significantly improves the accessibility of cybersecurity for small businesses, particularly those without dedicated IT personnel or cybersecurity expertise. By combining low-cost, open-source tools with automation and simplified setup procedures, the solution offers SMEs a realistic and sustainable way to implement real-time threat protection at the DNS layer. If widely adopted, such a tool could meaningfully reduce the attack surface across the SME sector, disrupting malware delivery, phishing attempts, and unwanted telemetry traffic before they reach endpoints. This aligns with modern cyber hygiene frameworks and increasing public sector efforts to support SME resilience through minimum-security baselines and digital readiness programmes.

Practical adoption barriers remain, including limited technical knowledge, hesitation to modify DNS settings, and fear of blocking legitimate services—as observed with the apple.com false positive incident. To address these concerns, improved onboarding materials, bundled deployment kits, and accessible community support models could help demystify the setup and build user confidence. Moreover, government agencies and industry associations could play a key role by promoting this kind of solution as a standard defensive measure for small organisations, either through funded pilot programmes, certification schemes, or curated feed partnerships. In this way, the project not only presents a viable technical solution, but also contributes to the broader conversation on democratising cybersecurity for under-resourced businesses.

7.4 Considerations for Larger Networks

While the proposed MISP to Pi-hole system is well-suited for small businesses, adapting it for larger SME networks with more than 50 users or more complex infrastructure would require architectural enhancements. Scaling the solution effectively would involve migrating to dedicated servers, increasing memory and CPU resources, and potentially deploying clustered Pi-hole instances behind a load balancer to distribute DNS query handling. Additionally, larger environments would benefit from centralised management interfaces, enhanced logging, and the ability to integrate with existing security operations platforms, such as SIEMs or central CTI aggregation hubs.

For enterprise-scale organisations, alternative solutions such as cloud-based DNS firewalls, Next-Generation Firewalls and commercial CTI platforms may be more appropriate due to their scalability, real-time analytics, vendor support, and broader ecosystem integration. While these tools offer high performance and feature-rich environments, they often come with significantly higher costs and complexity. In contrast, the value of the prototype lies in its simplicity, automation, and affordability, making it ideal for organisations just beginning to implement threat-aware defences. Importantly, this system can serve as a gateway or foundational layer, enabling SMEs to adopt more advanced security tools gradually as their capabilities and maturity evolve.

7.5 Alignment with Existing Literature

The findings of this thesis align with existing research that supports the effectiveness of DNS-based threat blocking in reducing exposure to malware, phishing, and unwanted telemetry. Like prior case studies in academic and industry literature, the results reinforce that DNS filtering provides a lightweight, proactive defence layer—particularly valuable for environments where endpoint defences may be inconsistent or absent. Where this study diverges is in its focus on automated CTI integration using free public sources and its practical application within resource-constrained SME environments. While many studies address DNS filtering at scale or within enterprise SOC contexts, few have explored the feasibility of combining public MISP feeds with Pi-hole in a fully automated, deployable workflow designed specifically for non-expert users.

This thesis contributes meaningfully to ongoing academic and practical discussions in cybersecurity, particularly around decentralised CTI usage, SME-focused solutions, and the operationalisation of open-source intelligence platforms. The prototype demonstrates how automation and minimal infrastructure can enable SMEs to adopt threat-aware defences without relying on commercial threat intel subscriptions or security appliances.

This approach offers a novel, real-world reference for future research on lightweight, decentralised defence mechanisms, and may also influence policy development, especially as governments and industry bodies increasingly advocate for minimum-security baselines and CTI sharing across all sectors. The project provides a foundation for evolving discussions on how to bridge the security capability gap for smaller organisations using accessible, community-driven technologies.

7.6 Future work

This thesis established the feasibility of integrating public threat intelligence from MISP with DNS filtering via Pi-hole for small business cybersecurity. There are several avenues for future enhancement and exploration remain to further improve accessibility, detection coverage, and architectural flexibility.

One promising direction is the deployment of MISP directly on a Raspberry Pi 4 or equivalent low-power single-board computer. While Pi-hole runs efficiently on such devices, hosting MISP—which is more resource-intensive due to its web interface, database, and feed management—requires careful configuration and performance tuning. Future research could evaluate which Raspberry Pi models and Linux variants offer the best trade-off between performance, reliability, and power efficiency. Testing could also explore storage optimisations (e.g. using external SSDs) and lightweight database backends to accommodate MISP's indexing and feed ingestion processes.

Another opportunity lies in implementing retrospective log analysis or "retro hunting" functionality, where DNS logs from Pi-hole (e.g., from a two-week query history) are compared against updated MISP feeds. This would allow organisations to identify missed threats in past traffic, supporting incident response and forensics without relying solely on real-time filtering. This feature could be implemented as a scheduled script or web-based interface that parses historic DNS logs, cross-references known IoCs, and outputs alerts or audit-ready reports. Such capability would enhance situational awareness and support SMEs in demonstrating regulatory due diligence.

Additionally, while DNS filtering provides lightweight first-line defence, it does not intercept HTTP/S-level threats or malicious URLs within allowed domains. To address this, future work could explore integrating a proxy-based security layer (e.g., a lightweight HTTP proxy such as Squid or Privoxy) alongside Pi-hole. The proxy could ingest MISP's URL-based or URI-pattern indicators and block access to malicious paths within otherwise legitimate domains—closing a key visibility gap. Integration would require defining rules for efficient feed parsing, protocol-aware filtering, and handling encrypted traffic (e.g. via HTTPS inspection, if legally and ethically acceptable in SME contexts).

These future directions could evolve the prototype into a more comprehensive, multi-layered defence solution, capable of delivering improved visibility, broader coverage, and forensic capability—while remaining aligned with SME constraints around cost, complexity, and usability.

8 Conclusion

8.1 Summary of Key Contributions

8.1.1 Achievements Relative to the Stated Objectives

The primary aim of this thesis was to develop a cost-effective and practical cybersecurity solution for small businesses by integrating public Cyber Threat Intelligence from MISP with DNS-based filtering via Pi-hole. The stated objectives included the development of a working prototype, the implementation of automated threat feed ingestion, thorough performance testing, and usability evaluation by non-technical users.

Each of these objectives was successfully met. A fully functional prototype was built and evaluated using lightweight infrastructure, requiring only a 1 vCPU / 8 GB RAM virtual machine, demonstrating that the system is suitable for SME-scale deployment. The project's automation scripts (*install.sh and misp-to-pihole*) significantly reduced installation complexity, enabling most test users to complete setup in under 50 minutes. Testing confirmed effective domain blocking, low system overhead, and high stability—even under simulated high-traffic conditions. The solution's clarity, performance, and maintainability were further validated through user feedback and stress testing, confirming the system's viability in realistic operating environments.

Public CTI and open-source tools can be combined into a usable, automated, and scalable cybersecurity system for small businesses, without the cost or complexity of commercial alternatives.

8.1.2 Significance for the SME Community and Cybersecurity Field

This thesis presents a practical and accessible cybersecurity solution tailored specifically to the needs of SMEs—a sector that is often under protected yet increasingly targeted by cyber threats. By demonstrating that public CTI feeds from MISP can be operationalised through DNS-based filtering using Pi-hole, the project addresses a critical gap in affordable and scalable defensive measures for organisations with limited technical and financial resources. The resulting system empowers SMEs to adopt real-time threat blocking capabilities without requiring commercial solutions or specialised personnel, significantly lowering the barrier to proactive cyber defence.

Beyond its practical utility, this research contributes to the broader cybersecurity field by offering a novel integration model of CTI and DNS filtering using fully open-source components. It shows how automation can bridge the usability gap between threat intelligence platforms and enforcement mechanisms, particularly when paired with user-friendly tools like Pi-hole. Academically, the work enriches current literature on lightweight, decentralised, and SME-oriented defence architectures, and highlights the importance of adapting CTI workflows to suit smaller infrastructures.

Looking ahead to the future work should explore how this solution can be further scaled—both in terms of hardware capability and feed management practices. Special attention should be paid to managing false positives through allowlisting, scoring, or reputation checks, and to refining feed filtering logic before Pi-hole ingestion. Additionally, while the system functioned reliably on a virtual machine, dedicated testing on a Raspberry Pi 4 would provide insight into its performance on low-cost physical hardware. Potential RAM constraints, I/O bottlenecks, or feed size limitations (e.g. avoiding large-scale 200K+ domain imports) would need to be addressed to ensure operational stability. These refinements would help advance the system into a more mature, production-ready state suitable for broader SME deployment.

References

[1] C. Rombaldo Jr, I. Becker, and S. Johnson, “Unaware, Unfunded and Uneducated: A Systematic Review of SME Cybersecurity,” 2023, arXiv. doi: 10.48550/ARXIV.2309.17186.

[2] L. Ambreen, M. Jain, R. K. Yadav, and S. Loonkar, “Effective cybersecurity risk management practices for small and medium-sized enterprises: A comprehensive review,” Multidisciplinary Reviews, vol. 6. Malque Publishing, p. 2023ss080, May 12, 2024. doi: 10.31893/multirev.2023ss080.

[3] N. Sonkar, “An Empirical Study on the Economic Impact of Cybersecurity Breaches and Computer Fraud on SMEs,” Journal of Information Systems Engineering and Management, vol. 10, no. 7s. Science Research Society, pp. 730–735, Jan. 10, 2025. doi: 10.52783/jisem.v10i7s.986.

[4] M. F. Almoaigel and A. Abuabid, “Implementation of Cybersecurity Situation Awareness Model in Saudi SMEs,” International Journal of Advanced Computer Science and Applications, vol. 14, no. 11. The Science and Information Organization, 2023. doi: 10.14569/ijacsa.2023.01411110.

[5] H. Perozzo, F. Zaghloul, and A. Ravarini, “CyberSecurity Readiness: A Model for SMEs based on the Socio-Technical Perspective,” Complex Systems Informatics and Modeling Quarterly, no. 33. Riga Technical University, pp. 53–66, Dec. 30, 2022. doi: 10.7250/csimq.2022-33.04.

[6] S. Kandpal, S. Bhatt, L. Mohan, A. Patwal, and P. Kumar, “Cyber Security Implementation Issues in Small to Medium-sized Enterprises (SMEs) and their Potential Solutions: A Comprehensive Analysis,” 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT). IEEE, pp. 1–5, Jul. 06, 2023. doi: 10.1109/icccnt56998.2023.10307363.

[7] L. B. Benjamin, A. E. Adegbola, P. Amajuoyi, M. D. Adegbola, and K. B. Adeusi, “Digital transformation in SMEs: Identifying cybersecurity risks and developing effective mitigation strategies,” Global Journal of Engineering and Technology Advances, vol. 19, no. 2. GSC Online Press, pp. 134–153, May 30, 2024. doi: 10.30574/gjeta.2024.19.2.0084.

[8] B. Kereopa-Yorke, “Building Resilient SMEs: Harnessing Large Language Models for Cyber Security in Australia,” arXiv, 2023, doi: 10.48550/ARXIV.2306.02612.

[9] M. S. Todd and S. (Shawon) M. Rahman, “Complete Network Security Protection for SME’s within Limited Resources,” International Journal of Network Security &amp; Its Applications, vol. 5, no. 6. Academy and Industry Research Collaboration Center (AIRCC), pp. 1–13, Nov. 30, 2013. doi: 10.5121/ijnsa.2013.5601.

[10] A. Shojaifar, S. Fricker, and M. Gwerder, ‘Elicitation of SME Requirements for Cybersecurity Solutions by Studying Adherence to Recommendations’, ArXiv, vol. abs/2007.08177. 2020.

[11] N. Rawindaran, A. Jayal, E. Prakash, and C. Hewage, ‘Cost Benefits of Using Machine Learning Features in NIDS for Cyber Security in UK Small Medium Enterprises (SME)’, Future Internet, vol. 13. p. 186, 2021.

[12] L. E. Sánchez, D. Villafranca, E. Fernández-Medina, and M. Piattini, ‘Practical Application of a Security Management Maturity Model for SMEs based on Predefined Schemas’, Unknown Journal. pp. 391–398, 2008.

[13] J. Magnusson, “Survey and Analysis of DNS Filtering Components,” 2024, arXiv. doi: 10.48550/ARXIV.2401.03864.

[14] M. van Haastrecht et al., “A Shared Cyber Threat Intelligence Solution for SMEs,” Electronics, vol. 10, no. 23. MDPI AG, p. 2913, Nov. 24, 2021. doi: 10.3390/electronics10232913.

[15] S. Gillard, D. P. David, A. Mermoud, and T. Maillart, “Efficient Collective Action for Tackling Time-Critical Cybersecurity Threats,” 2022, arXiv. doi: 10.48550/ARXIV.2206.15055.

[16] A. Mugwagwa, E. Bhero, and C. Chibaya, “Cybersecurity strategy: future proof cybersecurity for small to medium enterprises in South Africa,” International Journal of Research in Business and Social Science (2147- 4478), vol. 13, no. 4. Center for Strategic Studies in Business and Finance SSBFNET, pp. 15–24, Jun. 11, 2024. doi: 10.20525/ijrbs.v13i4.3308.

[17] J. W. Walker, “Data Security for the SME,” International Journal of Cyber Forensics and Advanced Threat Investigations, vol. 1, no. 1–3. Concept Tech Publishing, pp. 47–52, Feb. 15, 2021. doi: 10.46386/ijcfati.v1i1-3.19.

[18] V. Szücs, G. Arányi, and Á. Dávid, “Introduction of the ARDS—Anti-Ransomware Defense System Model—Based on the Systematic Review of Worldwide Ransomware Attacks,” Applied Sciences, vol. 11, no. 13. MDPI AG, p. 6070, Jun. 30, 2021. doi: 10.3390/app11136070.

[19] M. M. A. Mutalib, Z. Zainol, and M. H. M. Halip, “Mitigating Malware Threats at Small Medium Enterprise (SME) Organisation: A Review and Framework,” 2021 6th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE). IEEE, pp. 1–6, Dec. 01, 2021. doi: 10.1109/icraie52900.2021.9703991.

[20] P. Burda, A. M. Altawekji, L. Allodi, and N. Zannone, “The Peculiar Case of Tailored Phishing against SMEs: Detection and Collective DefenseMechanisms at a Small IT Company,” 2023 IEEE European Symposium on Security and Privacy Workshops (EuroS&amp;PW). IEEE, pp. 232–243, Jul. 2023. doi: 10.1109/eurospw59978.2023.00031.

[21] F. Alharbi et al., “The Impact of Cybersecurity Practices on Cyberattack Damage: The Perspective of Small Enterprises in Saudi Arabia,” Sensors, vol. 21, no. 20. MDPI AG, p. 6901, Oct. 18, 2021. doi: 10.3390/s21206901.

[22] M. F. Almoaigel and A. Abuabid, “Implementation of Cybersecurity Situation Awareness Model in Saudi SMEs,” International Journal of Advanced Computer Science and Applications, vol. 14, no. 11. The Science and Information Organization, 2023. doi: 10.14569/ijacsa.2023.01411110.

[23] B. Yigit Ozkan and M. Spruit, “Adaptable Security Maturity Assessment and Standardization for Digital SMEs,” Journal of Computer Information Systems, vol. 63, no. 4. Informa UK Limited, pp. 965–987, Sep. 28, 2022. doi: 10.1080/08874417.2022.2119442.

[24] L. Bamidele, L. B. Benjamin, A. Adegbola, P. Amajuoyi, M. D. Adegbola, and K. B. Adeusi, ‘Digital transformation in SMEs: Identifying cybersecurity risks and developing effective mitigation strategies’, Global Journal of Engineering and Technology Advances. 2024.

[25] N. Hossain and M. Hasan, “The Impacts of Cyberattack on SMEs in the USA and Way to Accelerate Cybersecurity,” Advances in Social Sciences Research Journal, vol. 11, no. 10. Scholar Publishing, pp. 197–203, Oct. 29, 2024. doi: 10.14738/assrj.1110.17724.

[26] T. Joswig and W. Kurz, “Empirical Analysis of NIS2 Adoption in EU SMEs: Challenges for Critical Infrastructure in Germany,” Journal of Next-Generation Research 5.0. Journal of Next-Generation Research 5.0, Mar. 12, 2025. doi: 10.70792/jngr5.0.v1i3.99.

[27] M. Neri, F. Niccolini, and R. Pugliese, “Assessing SMEs’ cybersecurity organizational readiness: Findings from an Italian survey,” Online Journal of Applied Knowledge Management, vol. 10, no. 2. International Institute for Applied Knowledge Management - IIAKM, pp. 1–22, Sep. 15, 2022. doi: 10.36965/ojakm.2022.10(2)1-22.

[28] A. Pikó, A. Bánáti, and E. Kail, “Supporting NIS 2 Directive with SOC,” 2024 IEEE 22nd Jubilee International Symposium on Intelligent Systems and Informatics (SISY). IEEE, pp. 533–540, Sep. 19, 2024. doi: 10.1109/sisy62279.2024.10737537.

[29] A. K. Daou, F. Li, and S. Shiaeles, “A Cost-Efficient Threat Intelligence Platform Powered by Crowdsourced OSINT,” 2023 IEEE International Conference on Cyber Security and Resilience (CSR). IEEE, pp. 48–53, Jul. 31, 2023. doi: 10.1109/csr57506.2023.10225008.

[30] A. Ramsdale, S. Shiaeles, and N. Kolokotronis, “A Comparative Analysis of Cyber-Threat Intelligence Sources, Formats and Languages,” Electronics, vol. 9, no. 5. MDPI AG, p. 824, May 16, 2020. doi: 10.3390/electronics9050824.

[31] A. D. Yudhistira and R. Harwahyu, “Implementation Strategy Analysis of Network Security using dalo RADIUS and Pi-hole DNS Server to enhance Computer Network Security, Case Study: XYZ as a Fintech Company,” Jurnal Indonesia Sosial Teknologi, vol. 5, no. 10. Publikasi Indonesia, pp. 4364–4379, Oct. 28, 2024. doi: 10.59141/jist.v5i10.5321.

[32] Y. Jin, M. Tomoishi, and N. Yamai, “Trigger-based Blocking Mechanism for Access to Email-derived Phishing URLs with User Alert,” 2023 International Conference on Electronics, Information, and Communication (ICEIC). IEEE, pp. 1–6, Feb. 05, 2023. doi: 10.1109/iceic57457.2023.10049906.

[33] S. Jinu, K. V. Krishnan, P. Yadav, M. Ramanan, and A. S. Revathy, “Blocking malicious domains: An experimental case study on DNS RPZ mechanism,” 2024 27th International Symposium on Wireless Personal Multimedia Communications (WPMC). IEEE, pp. 1–4, Nov. 17, 2024. doi: 10.1109/wpmc63271.2024.10863586.

[34] R. Sharma, N. Feamster, and A. Hounsel, “Measuring the Availability and Response Times of Public Encrypted DNS Resolvers,” 2022, arXiv. doi: 10.48550/ARXIV.2208.04999.

[35] R. Radu and M. Hausding, “Consolidation in the DNS resolver market – how much, how fast, how dangerous?,” Journal of Cyber Policy, vol. 5, no. 1. Informa UK Limited, pp. 46–64, Jan. 02, 2020. doi: 10.1080/23738871.2020.1722191.

[36] G. Hu and K. Fukuda, “Privacy Leakage of DNS over QUIC: Analysis and Countermeasure,” 2024 International Conference on Artificial Intelligence in Information and Communication (ICAIIC). IEEE, pp. 518–523, Feb. 19, 2024. doi: 10.1109/icaiic60209.2024.10463369.

[37] O. Starkova, K. Herasymenko, S. M. Korotin, V. Afanasiev, and A. Lisnyk, “Development of Recommendations for Ensuring Security in a Corporate Network,” 2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT). IEEE, Dec. 2019. doi: 10.1109/atit49449.2019.9030470.

[38] A. Jare, S. Kolte, S. Kadam, V. Babar, P. Tekade, and D. Salunke, “DefendNet: Harnessing AI/ML for Dynamic DNS Filtering and Network Security,” 2024 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS). IEEE, pp. 1–5, Oct. 17, 2024. doi: 10.1109/icbds61829.2024.10837330.

[39] A. Karlsson, R. Höglund, H. Wang, A. Iacovazzi, and S. Raza, “Enabling Cyber Threat Intelligence Sharing for Resource Constrained IoT,” 2024 IEEE International Conference on Cyber Security and Resilience (CSR). IEEE, pp. 82–89, Sep. 02, 2024. doi: 10.1109/csr61664.2024.10679511.

[40] C. Wagner, A. Dulaunoy, G. Wagener, and A. Iklody, “MISP,” Proceedings of the 2016 ACM on Workshop on Information Sharing and Collaborative Security. ACM, Oct. 24, 2016. doi: 10.1145/2994539.2994542.

[41] L. F. Ilca, O. P. Lucian, and T. C. Balan, “Enhancing Cyber-Resilience for Small and Medium-Sized Organizations with Prescriptive Malware Analysis, Detection and Response,” Sensors, vol. 23, no. 15. MDPI AG, p. 6757, Jul. 28, 2023. doi: 10.3390/s23156757.

[42] A. Iklody, G. Wagener, A. Dulaunoy, S. Mokaddem, and C. Wagner, ‘Decaying Indicators of Compromise’, ArXiv, vol. abs/1803.11052. 2018.

[43] S. Mokaddem, G. Wagener, A. Dulaunoy, and A. Iklody, ‘Taxonomy driven indicator scoring in MISP threat intelligence platforms’, ArXiv, vol. abs/1902.03914. 2019.

[44] A. Rüedlinger et al., “FeedMeter: Evaluating the Quality of Community-Driven Threat Intelligence,” Proceedings of the 10th International Conference on Information Systems Security and Privacy. SCITEPRESS - Science and Technology Publications, pp. 54–66, 2024. doi: 10.5220/0012357600003648.

[45] R. N. Dasmen, D. Darwin, I. Irham, and B. Riansyah, “Pi Hole on SOE Computer Network using Raspberry Pi 3 Model B+ to Optimize Bandwidth Management and Improve Employee Performance,” PIKSEL : Penelitian Ilmu Komputer Sistem Embedded and Logic, vol. 11, no. 1. Universitas Islam 45, pp. 11–22, Mar. 29, 2023. doi: 10.33558/piksel.v11i1.5911.

[46] A. M. Taib, “Securing Network Using Raspberry Pi by Implementing VPN, Pi-Hole, and IPS (VPiSec),” International Journal of Advanced Trends in Computer Science and Engineering, vol. 9, no. 1.3. The World Academy of Research in Science and Engineering, pp. 457–464, Jun. 25, 2020. doi: 10.30534/ijatcse/2020/7291.32020.

[47] T. Wunder, “AdGuard Home vs Pi-hole: Best Ad Blocker?” WunderTech, Mar. 4, 2023. [Online]. Available: https://www.wundertech.net/adguard-home-vs-pi-hole-best-ad-blocker/

[48] B. Marshall, “AdGuard Home vs Pi-hole – Best Adblocker,” Virtualization Howto, Mar. 3, 2023. [Online]. Available: https://www.virtualizationhowto.com/2023/03/adguard-home-vs-pihole-best-adblocker/

[49] A. Hacquard, “AdGuard vs Pi-hole: Taking Back Control of Your Internet with DNS Ad Blocking,” Enginyring, Oct. 12, 2023. [Online]. Available: https://www.enginyring.com/en/blog/adguard-vs-pi-hole-taking-back-control-of-your-internet-with-dns-ad-blocking

[50] N. D. Anggana, D. Hariyadi, R. Sahtyawan, and A. R. Jannah, ‘IMPLEMENTASI PI-HOLE UNTUK MEMBANGUN SISTEM PERTAHANAN JARINGAN DARI SERANGAN MALVERTISING’, Teknomatika: Jurnal Informatika dan Komputer. 2022.

[51] J. Magnusson, “Survey and Analysis of DNS Filtering Components,” 2024, arXiv. doi: 10.48550/ARXIV.2401.03864.

[52] M. Faiella, G. Gonzalez-Granadillo, I. Medeiros, R. Azevedo, and S. Gonzalez-Zarzosa, “Enriching Threat Intelligence Platforms Capabilities,” Proceedings of the 16th International Joint Conference on e-Business and Telecommunications. SCITEPRESS - Science and Technology Publications, pp. 37–48, 2019. doi: 10.5220/0007830400370048.

[53] K. T. A. U. Lakmal, L. M. C. Perera, S. P. K. Padmika, S. P. A. De Silva, D. Pandithage, and D. Siriwardana, “Email Armour: A Multi-Layered Email Defense Solution,” 2024 9th International Conference on Information Technology Research (ICITR). IEEE, pp. 1–6, Dec. 05, 2024. doi: 10.1109/icitr64794.2024.10857760.

[54] B. Stojkovski, G. Lenzini, V. Koenig, and S. Rivas, “What’s in a Cyber Threat Intelligence sharing platform?,” Annual Computer Security Applications Conference. ACM, pp. 385–398, Dec. 06, 2021. doi: 10.1145/3485832.3488030.

[55] M. van Haastrecht, I. Sarhan, A. Shojaifar, L. Baumgartner, W. Mallouli, and M. Spruit, “A Threat-Based Cybersecurity Risk Assessment Approach Addressing SME Needs,” Proceedings of the 16th International Conference on Availability, Reliability and Security. ACM, pp. 1–12, Aug. 17, 2021. doi: 10.1145/3465481.3469199.

[56] M. Meiessaar, MISP-to-Pi-Hole: Open-source CTI to DNS filtering integration for SMEs, GitHub repository, 2024. [Online]. Available: https://github.com/meieme/MISP-to-Pi-Hole

Appendix 1 – Non-exclusive Licence for Reproduction and Publication of a Graduation Thesis

I Mert Meiessaar

1. Grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis "How to Protect Small Businesses Using Public Cyber Threat Intelligence", supervised by Toomas Lepik and Hillar Põldmaa.

1.1. to be reproduced for the purposes of preservation and electronic publication of the graduation thesis, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright;

1.2. to be published via the web of Tallinn University of Technology, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.

2. I am aware that the author also retains the rights specified in clause 1 of the non-exclusive licence.

3. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

18.05.2025