Advancing Internet-Connected Devices Posture Analysis with a Meta-Search Engine: A Case Study in Energy Systems

Andrea Bernardini[1,*,†], Mario Lezoche[2,‡], Simone Angelini[1,†], Giovanna Dondossola[3,†] and Roberta Terruggia[3,†]

[1]*Fondazione Ugo Bordoni, Viale del Policlinico, 147, 00161, Rome, Italy*
[2]*University of Lorraine, CNRS, CRAN, Nancy, F-54000, France*
[3]*RSE S.p.A., Via Raffaele Rubattino, 54, 20134, Milan, Italy*

Abstract

In the contemporary digital ecosystem, Internet of Things Search Engines can be used for passive reconnaissance of Internet-connected devices, mapping possible attack surfaces without a direct interaction with the target devices or infrastructures. Each IoT search engine utilizes diverse scanning techniques and analytical methodologies, resulting in metadata with varying levels of coverage, accuracy, and relevance.

This research introduces an IoT meta-search engine prototype designed to aggregate and merge metadata from commercial IoT search engines (Shodan, Censys, Netlas, Zoomeye, Binaryedge, Fofa) complemented by Common Vulnerabilities and Exposures (CVE) and Common Weakness Enumeration (CWE) sources. By merging those data, a more comprehensive and detailed perspective of the interconnected device landscape can be provided. Our methodology leverages an ontological framework using Stanford's Protégé and Python, implementing zero-shot learning with a panel of three Large Language Models (LLMs) under human supervision to map IoT search engine taxonomic structures and quantitatively validate the generated Knowledge Base. The IoT meta-search engine is tested on photovoltaic (PV) energy production and monitoring systems, a domain essential to renewable energy grids. Vulnerabilities in PV systems can be exploited by hackers, causing energy disruptions, data breaches, or manipulation of grid operations. Although the findings are preliminary, they serve as a proof of concept to demonstrate the feasibility of the methodology to provide various types of overviews and insights associated with individual and multiple hosts for security posture evaluation.

Keywords

Search engine, vulnerability, security, ontology, Internet of Things, Internet Connected Device, LLM, energy systems,

1. Introduction

Internet-connected devices (ICDs) allow remote control and automation, transforming the human interaction with the environment in various fields, from homes to industries and critical infrastructures. This widespread reach, however, may raise serious issues regarding security and privacy [1] since successful cyber-attacks can lead to financial losses, operational disruptions, or reputational damage. Vulnerabilities in ICDs make them potentially subject to exploitation due to their poor patch management, presence of open ports and usage of default credentials. Moreover their reliance on open-source software, which, while due to code transparency, facilitates external audits and vulnerability remediation, simultaneously provides attackers with potential exploit pathways. Moreover, the dependence on open-source software may result in inconsistent maintenance unable to ensure timely updates and security patches.

An IoT search engine (IoTSE) is a specialized tool designed to discover, index, and retrieve information about ICDs connected to a network. Like web search engines, this tool performs automated network

Joint National Conference on Cybersecurity (ITASEC & SERICS 2025), February 03-8, 2025, Bologna, IT
*Corresponding author.
†These authors contributed equally.
abernardini@fub.it (A. Bernardini); mario.lezoche@univ-lorraine.fr (M. Lezoche); sangelini@fub.it (S. Angelini); giovanna.dondossola@rse-web.it (G. Dondossola); roberta.terruggia@rse-web.it (R. Terruggia)
0000-0002-3271-1742 (M. Lezoche)

indexing by systematically scanning for active devices, primarily through comprehensive port scans of individual IP addresses using an advanced internet crawler.

IoT search engines collect data by sending probe packets to network devices and analyzing their responses. This process extracts metadata such as open ports, services, operating systems, device types, and geographic locations. The information is then organized into "banners" textual metadata collections for each device. These banners can be further enriched with additional details about the type of service or product identified using specific attributes or tags. The IoTSEs differ in terms of the algorithms and methodologies used to identify devices that generate unique and distinctive outputs, helping to systematically map the digital footprints of systems, services, and infrastructures. However, their systematic use and integration into vulnerability detection and security assurance processes are limited by the lack of standards and the heterogeneity of responses from IoT engines. To face this criticality, ontologies, established methodologies for defining and representing information, which provide a structured view of a specific domain [2], can be used. These approaches facilitate efficient, machine-readable processing of large-scale information, significantly enhancing data integration and interoperability. The main goal of an ontology is to offer a formal and detailed representation of knowledge to facilitate the sharing, integration, and understanding of information within a given domain [3] [4]. As a drawback, the process of ontology engineering is a complex and time-consuming task since it often demands substantial input from human experts and the creation of specialized programmatic code. With the breakthrough of Large Language Models (LLMs), an artificial intelligence type that understands and creates natural language text, ontology engineering has received a significant boost. LLM may assist humans in performing tedious and complex tasks by leveraging their ability to extract semantics from texts and structured information, even when used by non-expert but semi-knowledgeable users. Despite their well-documented limitations [5] [6] such as occasional inaccuracies and the phenomenon of generating inexplicable responses, known as "*hallucinations*", LLMs offer significant potential to accelerate ontology development. While their contributions remain partial and require consistent human validation, LLMs represent a promising tool to enhance productivity in this field. Emerging evidence [7] shows that their evaluations often align with those of experts, and they have already demonstrated success in various domains [8] [9]. As their capabilities improve, LLMs are set to support humans in judgment and prediction tasks, creating new opportunities for effective collaboration.

The main contributions of this work are fourfold:

- Analyzing and comparing IoT search engine rankings and results to explore integration scenarios;
- Proposing a novel methodological approach for integrating search results from multiple IoT search engines by a data collection pipeline for gathering and storing results and by a data processing pipeline to populate an ontology with heterogeneous data sources into a Knowledge Base (KB);
- Investigating the usage of LLMs under human supervisory control (*human in the loop*) for performing: (1) semantic metadata alignment of IoT search engine outputs (ontology alignment) and (2) quantitative validation of the generated Knowledge Base;
- Validating the proposed methodology on an energy-related case study - photovoltaic production monitoring systems.

2. Related works

Research on identifying vulnerable ICDs is evolving through diversified methodological approaches utilizing specialized IoT search engines, vulnerability tools, and external data sources. While Shodan [10] remains a fundamental platform, emerging tools like Censys [11], Zoomeye [12], Fofa [13], Netlas [14], and Binaryedge [15] have expanded device reconnaissance possibilities while each platform offers unique indexing methodologies [16]. In [17], authors decode the Common Platform Enumeration (CPEs) found in the Shodan banners and compare them with hash tables containing the entire CPEs database to efficiently extract vulnerabilities from the National Vulnerability Database (NVD) [18]. A

similar approach, based on interfacing with Censys, is used by the Scout tool [19] [20], in which the basic functionalities are extended with metadata from NVD to identify additional vulnerabilities in specific services operating on publicly available IP addresses. The analysis then integrates the CPEs by associating their results with the CVEs. Results were then validated against active industry tools such as Nessus and OpenVAS. Similarly, in [21], Shodan was used in conjunction with Nessus for results evaluation. It is worth noting that neither of the tools mentioned above uses ontologies for data structuring. Regarding the comparison of IoT search engine results, [22] provides preliminary evaluations without delving into result integration. Other works compare methodologies and strategies among search engines like Censys and Shodan [23], and Zoomeye [24], highlighting the differences in vulnerability identification capabilities. In [25] the author provides a high-level model for merging results from IoT search engines, focusing on challenges such as identifying relevant devices, reusing results, and cross-referencing data with external sources, including multiple IoT search engines. [26] introduces a Dynamic Cybersecurity Ontology (DCO) designed to map dynamic scan results from services like Shodan and Censys. This approach aims to merge data and create a real-time correlation mechanism for identifying organizational vulnerabilities, laying the groundwork for a systematic cyber risk assessment process. A subsequent study [27] explores the use of ontologies to support the identification and mitigation of vulnerabilities in PLC components. Although preliminary, it demonstrates an interesting use case for CPE, NVD, NIST controls, and CERT reports. The work [28] underlines the need to transform IoT search engine data into more effective and structured formats (e.g., device name, manufacturer, and software version) to facilitate the identification of exposed devices but also enables vulnerability assessments and the implementation of appropriate security measures.

About the merging results [29] integrates Shodan and Binaryedge while [30] is particularly relevant to our study as it focuses on combining results from four search engines (Shodan, Censys, Zoomeye, and Fofa) using 23 keywords related to EV Charging Management Systems. However, rather than analyzing the complete set of available metadata, the authors limit their examination to banners associated with the results, followed by manual verification of their relevance to the search query. Traditional ontology development (engineering), is performed manually and is often constrained by the biases, expertise, and perspective of the developers, leading to incomplete or overly rigid structures that fail to adapt to evolving knowledge domains. It is indeed a very time-consuming activity. On the other hand, LLMs have shown significant potential in the development and maintenance of ontologies, addressing some limitations inherent in human-developed ontologies. LLMs are gradually often used for building [31], aligning [32] [33] or populating [34] ontologies. In SPIRES [34] LLM technology is used in a knowledge extraction framework to perform zero-shot learning and respond to queries by leveraging flexible prompts, ensuring the output is aligned to a user-defined knowledge schema. All these recent works suggest that a hybrid approach, combining human expertise with LLM capabilities, is needed to ensure the resulting ontologies are accurate and flexible, representing real-world knowledge.

3. Integration Scenarios for IoT search engines

The first evaluated scenario focuses on "*ranking fusion*", a methodology typical of information retrieval which aims to combine and aggregate results from multiple sources to improve the overall quality and relevance of the retrieved information. Each IoT search engine uses different scanning techniques and has access to distinct yet partially overlapping internet segments, resulting in coverage, accuracy, and relevance variations. Ranking fusion combines results from multiple engines to generate a single, more relevant ordered list, optimizing the overall relevance of the outcomes. A first sample of 23 queries was defined and submitted to the search engines to evaluate this scenario. For each engine, the top 500 results in terms of IP numbers were recorded. The IP number lists were then compared to identify any overlaps among the responses of the search engines. The overlap between search engine results was minimal, as shown in the Appendix, indicating that, unlike traditional search engines, the dataset in this context is highly sparse. This suggests significant differences in coverage, as well as in the indexing and retrieval of results.

The second scenario investigated focuses on "*metadata fusion*" associated with search engine results. Metadata fusion could combine results from different sources to produce a unified set of data, optimizing the consistency and completeness of the information. This process can drive a more accurate and comprehensive representation of the data. For example, only two of the six IoT search engines (Shodan and Netlas) in the available configurations reported vulnerabilities. Meanwhile, some engines provide additional information that can refine the understanding of an IP address, ownership, and location, which could be of use from a vulnerability assessment perspective. Zoomeye, for instance, indicates whether an IP address corresponds to a honeypot; Censys provides details about the source and vendor of a service; Binaryedge, Fofa, and Zoomeye report the identified banners as metadata itself. To automatically identify services, generate descriptive tags, assign product labels, and gather metadata about the hosting organization, various search engines use AI-based algorithms.

This observation inspired the definition of a preliminary methodology for expanding the knowledge base related to a service or exposed host by combining results from multiple search engines to generate richer and more comprehensive information by integrating different perspectives.

4. Proposed methodology for collecting and merging of IoT metadata

To get an overview of the scope and of the objectives the system will have to respond is helpful to create a set of competency questions (CQs) [35] [36] which describe what the system is expected to answer or facilitate. These questions help define the scope and requirements of the system, serving as a clear set of benchmarks that indicate what it should represent and how it should behave. Here, the goal is to create a meta-engine to gather, merge, and harmonize the results of various IoT search engines to provide analytical tools for users conducting vulnerability assessments. Below is a list of queries identified as relevant, along with an example of the corresponding answers that will be extracted later by the proposed meta-search engine, specifically tailored to an energy use case.

- **CQ 1: What vulnerabilities are associated with exposed hosts in a specific country?** Example: for queries targeting "*ABB solar inverters*" in Italy, the system identified three hosts with several CVEs, including some recent ones, such as CVE-2023-48795[1] and CVE-2023-38408[2], which have also been reported by the Italian CSIRT (*Computer Security Incident Response Team*). Such information could be essential for regional cybersecurity assessments.
- **CQ 2: Do the identified results expose critical vulnerabilities for which exploits have already been identified?** Example: by cross-referencing IoTSE data with the CVE database, the system flagged devices where exploits were actively discussed in security forums. For instance, one host associated with "*Modicon M340*" revealed multiple documentation and links (e.g., GitHub repository of exploit code) through the ontology enrichment module, demonstrating the system's capability to detect threats.
- **CQ 3: Given a set of IP addresses, which services and products are identified by the IoT engines and with which CPEs?** Example: 12 devices tagged as End of Life (EOL) by IoTSEs under the "*SmartPOWER Energy Management System*" may be at risk, as they will no longer receive updates. Additionally, multiple firmware versions, ranging from 3.11 to 4.0.3, for the Solarview product indicate potential vulnerability to known exploits.
- **CQ 4: What weaknesses are associated with each host that could be exploited by a malicious actor?** Example: For a device affected by CVE-2022-28615, the system identified CWE-190 (*Integer Overflow*) as a significant weakness. This insight highlights the possibility of attackers using arithmetic manipulations to disrupt the host's logic, causing resource exhaustion or service disruption.

[1] https://www.csirt.gov.it/contenuti/la-settimana-cibernetica-del-21-aprile-2024

[2] https://www.csirt.gov.it/contenuti/poc-pubblico-per-lo-sfruttamento-della-cve-2023-38408-relativa-a-openssh-al01-230720-csirt-ita

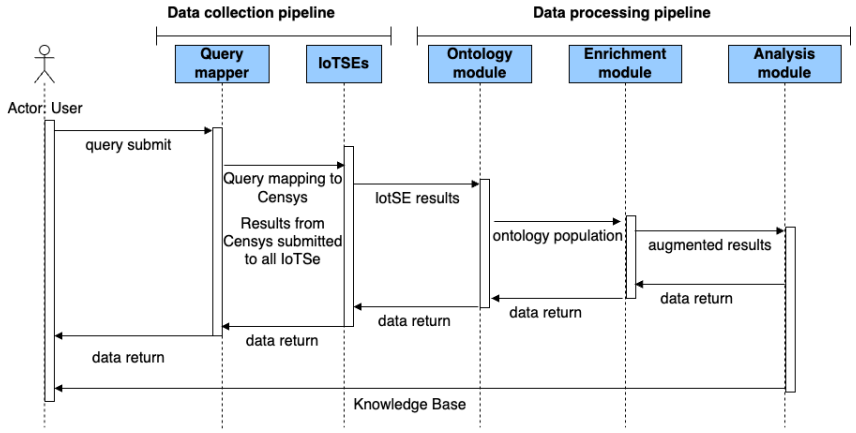

Figure 1: Sequence Diagram

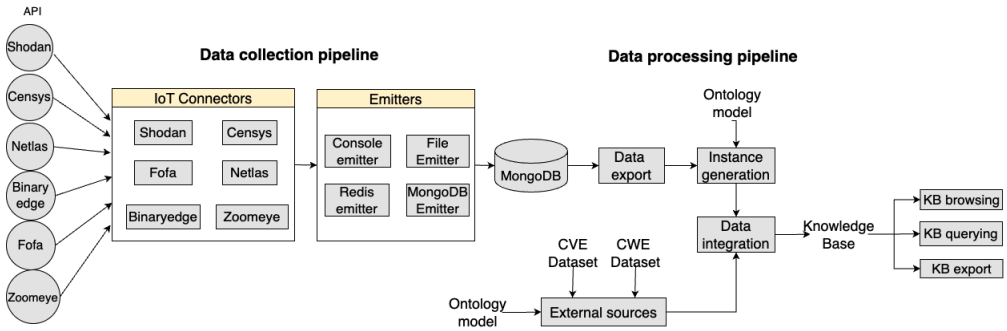

Figure 2: Architecture

After defining the system's objectives by the competency queries, the focus is on the information flow, which starts with a query sent to search engines and concludes with the aggregation of results in a format suitable for further vulnerability assessment. To illustrate this flow, we present a sequence diagram in Fig.1 to clearly represent the temporal flow of communications between the actors and entities, highlighting the order in which messages are exchanged and operations are carried out.

The diagram illustrates how a user query is progressively processed through the various components, adding value or transforming the data at each step. The process concludes with the population of the ontology and the return of results to the user through the system's various layers in textual or graphical form, addressing the competency questions. More in detail the "*User*", initiates the interaction with the system by submitting a query to the "*Query Mapper*", a component that receives the user's query and translates it into a format compatible with IoT search engines. Queries are sent to the "*IoTSEs*" component that processes the translated query to retrieve relevant results which are passed to a "*Merging Module*" that combines and organizes the results obtained from the "*IoTSEs*". The merged results are then augmented by an "*Enrichment module*" with additional information from external sources, such as CWE and CVE databases. Lastly, the "*Analysis Module*" is in charge of exporting the results in multiple formats, from data visualization to CSV exports.

The corresponding architecture is presented in Fig. 2 structured into two main components: (1) a data collection pipeline responsible for querying search engines and gathering and storing the results, and (2) a data processing pipeline tasked with transforming raw data into an ontology, further enriched with information from external data sources related to CVEs and CWEs.

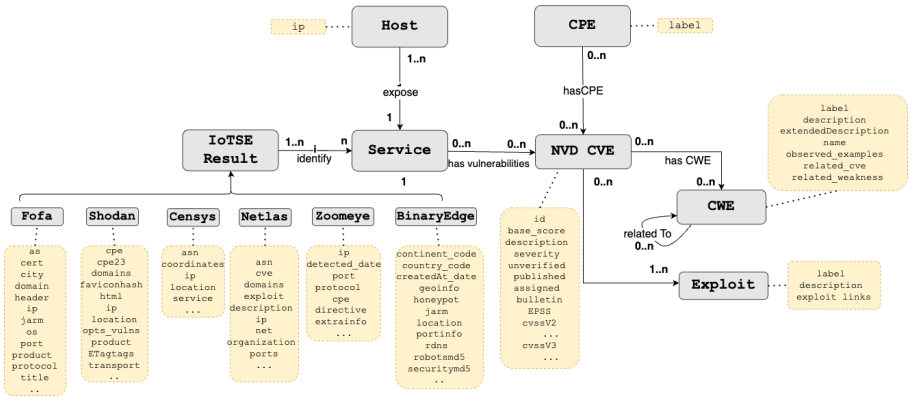

Figure 3: Conceptual model

4.1. Data collection pipeline

The data collection pipeline is organized in a partially automated process where (1) Censys is used for an initial textual query (e.g., to search for a component under investigation), and (2) the hosts identified by Censys are individually analyzed with all IoT engines. The selection of Censys as the primary engine is tied to the expressiveness of its query language, which enables the development of queries that allow for precise device fingerprinting [37]. Starting from a general query about a device identified as potentially vulnerable, a more specific query is then developed using distinctive features such as hashes, labels, titles, favicons, and other characteristics Each source contributes its stream of raw data to the central system. The middle component of the architecture hosts the IoT connector components, which establish connections using their respective APIs to ensure the format normalization and integrity of acquired data. The emission system manages the distribution of processed information by multiple output modalities: a console interface for operational monitoring, a file storage system for data persistence, Redis integration for cache management, and a MongoDB connection for definitive information storage.

4.2. Data merging pipeline

The subsequent step preliminary to the ontology deployment is the definition of a conceptual data model (Fig. 3) which includes entities, relationships, and attributes that will be used in the logical modeling process, particularly in defining the TBox (*Terminology Box*) and the ABox (*Assertion Box*). This conceptual distinction allows for the separation of general domain knowledge from specific information. To simplify the complex ontology engineering process we detail the merging and alignment process:

Ontology Conceptualization: A conceptual model defines core entities (e.g., *IoTSE Result, CVE, CWE*), their relationships (e.g., *hasCPE, hasCVE*), and attributes (e.g., *Exploit Link, Location*). Protégé [38] is employed to translate this model in Fig, 3 into an OWL ontology as in Fig. 4.

Data Harmonization and Semantic Mapping: Semantic heterogeneity across search engine results is addressed through alignment. Different IoT search engines may label the concept of a service using different terms, such as "app", "category" or "product".

Ontology Population: Data is imported into the ontology using a Jupyter Colab Notebook[3], where instances from IoTSE are merged with data generated from CVE and CWE databases. For instance, a Shodan search result is transformed into an OWL instance as in Fig. 5. This approach minimizes human intervention in aligning heterogeneous data, reducing manual effort while ensuring high accuracy.

The integration of results from different IoT search engines is complex due to their structural differences and varying levels of nesting, as well as the type of response, which may focus on a single port or multiple ports belonging to a host. This semantic fragmentation makes it difficult to achieve

[3]configured with baseline specifications: Python 3 Google Compute Engine backend, CPU, up to 12.7 GB of system RAM, 30 GB of disk storage without any performance guarantees.

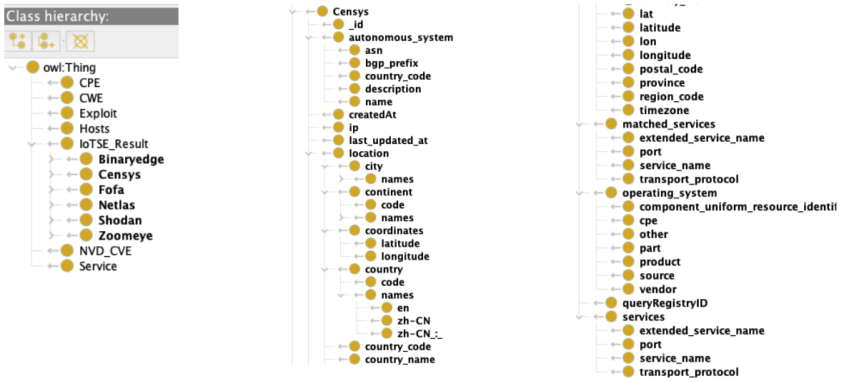

Figure 4: Class hierarchy on Protégé

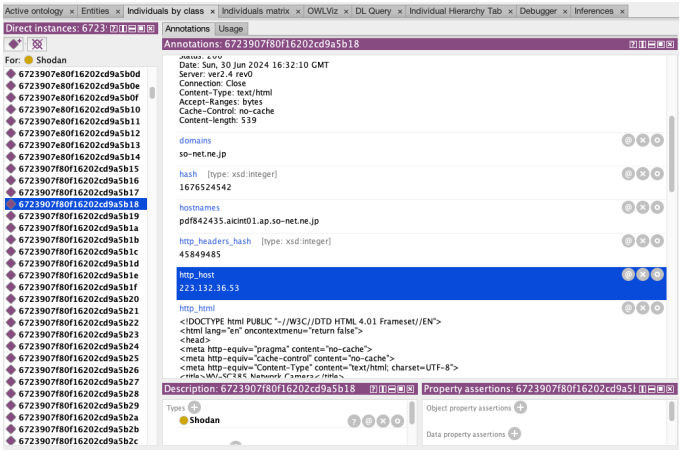

Figure 5: Protégé instance

the interoperability and effective aggregation of IoT data. During the ontology population, it was then decided: (1) to maintain this structural heterogeneity when storing instances related to the analyzed IPs. This aligns with the prototypical nature of the tool and allows for individual exploration of the results, as well as enabling future comparisons to assess the level of detail in the responses provided. (2) To address the heterogeneity of IoTSE results in terms of data nesting levels and the presence of multiple subclasses by using annotations in a flat, single-level structure for each class, rather than a complex hierarchy with multiple subclasses. This methodological approach simplified data management and loading, facilitating the integration and comparison of information from heterogeneous sources. The task of aligning the structures of the results was framed within the context of ontology alignment, an area focused on identifying and establishing semantic correspondences between equivalent concepts in different data structures to facilitate data integration. An experimental approach was used to extend the initial conceptual mappings between a sample of manually conducted results. Outside the pipelines, three LLMs[4] were prompted to identify analogous concepts across the results metadata, enabling the creation of a mapping table that converts and aggregates these concepts during the querying/visualization phase of the data. The choice to use three LLMs is motivated by the fact that even LLMs, as humans, can have different biases [39] vulnerable to exploits that can misguide the system [40] [41]. A possible approach to improve the robustness and the fairness of automated judging systems is to include multiple LLM models to form a sort of "*LLM-judging committee*". While LLMs provide significant support in mapping and aligning heterogeneous IoT metadata, their limitations must be acknowledged and mitigated to ensure the accuracy and reliability of the ontology.

[4]Claude - Haiku, Gemini - 1.5 Flash, and ChatGPT - GPT-4

5. Case study: Photovoltaic production monitoring systems

The use case for testing the methodology is the PV system, a sector where there has been a proliferation of devices and monitoring tools accessible via the internet and controlled by end users, third parties, and service companies, creating a vast attack surface vulnerable to threats at the device level to gain access to the power grid. From initial coordinated attacks on thousands of devices, scenarios of compromised energy distribution networks, from energy communities to critical infrastructures, can emerge. Eight components were identified by taking evidence from the literature and leveraging evidence from security forums: "*Solarview compact*", "*Enphase envoy*","*Altenergy*","*ABB solar inverter*","*Huawei smarterlog 2000 Power System Inspire*", "*SmartPOWER Energy Management System*","*Modicon M340*","*Victron energy*". For each component, a specific query using the Censys query language is then generated and submitted to the pipelines. The KB generated contains over 1.9 million triples (33% from IoTSE results, 66% from the CVE database, and less than 1% from the CWE database).

Several strategies with varying levels of granularity are available for accessing and visualizing the data. Data can be accessed by focusing on single instances and search engines as in Fig. 5 where a single result from Shodan is shown in Protégé. In the left panel, the list of loaded instances is presented, while the central-right panel displays the list of metadata among the other services and products identified with CPE, the Autonomous System Name (ASN) indication, and the date field. Data can be accessed through the Colab interface, allowing users to query both a single IP address and visualize content representing the fusion of information extracted from IoTSE, as shown by answering the competency questions in the Appendix and highlighting end-of-life product, old version of firmware of software and CVEs. Beyond the identification of vulnerabilities, what stands out is that by merging information generated by advanced proprietary methodologies of each engine, the identification of the product, the service, the operating system, or the possible associated CPE is enabled. All collected information potentially signals vulnerabilities, yet these indicators require meticulous verification, as they might represent server-side patches or false positives that demand careful and systematic investigation, This insight is useful as it can be paired and studied alongside banners, page HTML code, and other details (such as identifying whether it is a honeypot or determining its vendor) to build a digital footprint associated with single or multiple hosts.

In the Appendix, a qualitative evaluation of the proposed solution is presented. Answering the competency questions that guided the development of the ontology, a sample of the possible data extrapolations is shown using SPARQL queries and Python functionalities for aggregating and visualizing the results. For the quantitative evaluation of the ontology, three LLMs with a zero-shot prompting methodology without a specific previous training for this task are used, on the base of the ontology statistics produced by Protégé. The three evaluations redacted from LLMs describe an ontology that is "*large, data-driven, and focused on individual instances rather than complex terminological structures. The ontology features a significant number of instances, annotations, and axioms, which suggest a well-documented system. However, it remains relatively simple in logical complexity, emphasizing basic hierarchical structures and object/data properties over complex logical constructs. Despite its size, the class hierarchy is not overly intricate, with fewer equivalent or disjoint classes.*" LLMs have clearly and unequivocally identified the limitations and potential of this ontology, confirming some of the objectives for the continuation of this activity. These objectives pertain to reducing the amount of metadata associated with instances and improving organization through relationships.

6. Conclusions

IoT search engines are valuable tools for analyzing the attack surface of infrastructures or studying vulnerabilities associated with specific devices. This work proposes an ontological approach to developing a prototype IoT meta-search engine that merges results from multiple IoT search engines and external sources, such as CVE and CWE databases. It also integrates zero-shot prompting, supervised by experts, for ontology alignment and Knowledge Base (KB) validation. The objective is to obtain

"better" and potentially more useful information for each identified host for subsequent vulnerability assessment and risk mitigation stages. This work presents a unique perspective not previously explored in terms of the number of properties analyzed and involved in IoT search engines. The proposed IoT meta-search engine has been evaluated on a use case of devices for the production and monitoring of photovoltaic systems. The generated KB contains over 1.9 million triples, and each identified host can be characterized by up to 175 properties. This work has identified several indicators of potential vulnerabilities, unpatched systems, and discontinued products. However, our primary focus was evaluating the technical feasibility of incorporating these IoT findings into an ontology-driven solution. This will help establish a foundation for more comprehensive assessments in future studies. From the obtained results, in addition to specific analyses on a single host, more complex reasoning can be developed, such as examining the presence of a vulnerable device within a particular country, generating aggregated statistics for devices, and preliminarily evaluating the impact and the reliability of integrating a panel of LLMs in the process of ontology engineering.

7. Acknowledgments

The authors would like to express gratitude to Shodan, Censys, Zoomeye, Fofa, Binaryedge, Netlas, and Vulners for their collaboration and generosity in providing access to their search engines and vulnerability scanners. This work is original and has been partly supported by a collaboration between RSE S.p.A. and Fondazione Ugo Bordoni, financed by the Research Fund for the Italian Electrical System under the Three-Year Research Plan 2022-2024 (DM MITE n. 337, 15.09.2022), in compliance with the Decree of April 16th, 2018. The authors would like to thank our colleagues Claudio Carpineto and Gianni Romano from Fondazione Ugo Bordoni for their help in the preliminary activities related to the ranking merging scenario.

Declaration on Generative AI

During the preparation of this work, the authors used Grammarly in order to: Grammar and spelling check. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] S. A. Baho, J. Abawajy, Analysis of consumer iot device vulnerability quantification frameworks, Electronics 12 (2023) 1176.

[2] N. Guarino, D. Oberle, S. Staab, What is an ontology?, Handbook on ontologies (2009) 1–17.

[3] B. Chandrasekaran, J. R. Josephson, V. R. Benjamins, What are ontologies, and why do we need them?, IEEE Intelligent Systems and their applications 14 (1999) 20–26.

[4] M. Uschold, M. Gruninger, Ontologies: Principles, methods and applications, The knowledge engineering review 11 (1996) 93–136.

[5] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, et al., A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions, ACM Transactions on Information Systems (2023).

[6] Z. Xu, S. Jain, M. Kankanhalli, Hallucination is inevitable: An innate limitation of large language models, arXiv preprint arXiv:2401.11817 (2024).

[7] A. R. Doshi, J. J. Bell, E. Mirzayev, B. S. Vanneste, Generative artificial intelligence and evaluating strategic decisions, Strategic Management Journal (2024).

[8] X. Luo, A. Rechardt, G. Sun, K. K. Nejad, F. Yáñez, B. Yilmaz, K. Lee, A. O. Cohen, V. Borghesani, A. Pashkov, et al., Large language models surpass human experts in predicting neuroscience results, Nature Human Behaviour (2024) 1–11.

[9] J. Gu, X. Jiang, Z. Shi, H. Tan, X. Zhai, C. Xu, W. Li, Y. Shen, S. Ma, H. Liu, et al., A survey on llm-as-a-judge, arXiv preprint arXiv:2411.15594 (2024).

[10] J. Matherly, Shodan: The search engine for internet-connected devices, https://www.shodan.io, 2022. Online tool, retrieved on November 27, 2024.

[11] Z. Durumeric, D. Adrian, A. Mirian, M. Bailey, J. A. Halderman, A search engine backed by Internet-wide scanning, in: 22nd ACM Conference on Computer and Communications Security, 2015, p. "". Online tool, retrieved on November 27, 2024.

[12] nd, Zoomeye global internet asset data, https://www.zoomeye.hk, 2021. Online tool, retrieved on November 27, 2024.

[13] nd, Fofa search engine version 4.9.148, https://en.fofa.info/, nd. Online tool, retrieved on November 27, 2024.

[14] nd, Netlas, https://netlas.io/, nd. Online tool, retrieved on November 27, 2024.

[15] nd, binaryedge, https://app.binaryedge.io/login, nd. Online tool, retrieved on November 27, 2024.

[16] R. Li, M. Shen, H. Yu, C. Li, P. Duan, L. Zhu, A survey on cyberspace search engines, in: Cyber Security: 17th China Annual Conference, CNCERT 2020, Beijing, China, August 12, 2020, Revised Selected Papers 17, Springer Singapore, 2020, pp. 206–214.

[17] B. Genge, C. Enăchescu, Shovat: Shodan-based vulnerability assessment tool for internet-facing services, Security and communication networks 9 (2016) 2696–2714.

[18] H. Booth, D. Rike, G. A. Witte, The national vulnerability database (nvd): Overview, nd (2013).

[19] J. O'Hare, Scout: A contactless 'active'reconnaissance known vulnerability assessment tool, nd (2018).

[20] J. O'Hare, R. Macfarlane, O. Lo, Identifying vulnerabilities using internet-wide scanning data, in: 2019 IEEE 12th International Conference on Global Security, Safety and Sustainability (ICGS3), IEEE, 2019, pp. 1–10.

[21] R. Williams, E. McMahon, S. Samtani, M. Patton, H. Chen, Identifying vulnerabilities of consumer internet of things (iot) devices: A scalable approach, in: 2017 IEEE International Conference on Intelligence and Security Informatics (ISI), IEEE, 2017, pp. 179–181.

[22] F. Z. Fagroud, E. H. Ben Lahmar, H. Toumi, K. Achtaich, S. El Filali, Iot search engines: Study of data collection methods, in: Advances on Smart and Soft Computing: Proceedings of ICACIn 2020, Springer Singapore, 2021, pp. 261–272.

[23] C. Bennett, A. Abdou, P. C. van Oorschot, Empirical scanning analysis of censys and shodan, in: Workshop on Measurements, Attacks, and Defenses for the Web, 2021, p. "".

[24] M. Yu, J. Zhuge, M. Cao, Z. Shi, L. Jiang, A survey of security vulnerability analysis, discovery,

detection, and mitigation on iot devices, Future Internet 12 (2020) 27.
[25] M. Arnaert, Y. Bertrand, K. Boudaoud, Modeling vulnerable internet of things on shodan and censys: An ontology for cyber security, in: Proceedings of the Tenth International Conference on Emerging Security Information, Systems and Technologies (SECUREWARE 2016), 2016, pp. 299–302.
[26] J. Pastuszuk, P. Burek, B. Ksiezopolski, Cybersecurity ontology for dynamic analysis of it systems, Procedia Computer Science 192 (2021) 1011–1020.
[27] T. Heverin, M. Cordano, A. Zeyher, M. Lashner, S. Suresh, Exploring ontologies for mitigation selection of industrial control system vulnerabilities, in: International Conference on Cyber Warfare and Security, volume 17, 2022, pp. 72–80.
[28] M. Borhani, G. S. Gaba, J. Basaez, I. Avgouleas, A. Gurtov, A critical analysis of the industrial device scanners' potentials, risks, and preventives, Journal of Industrial Information Integration 100623 (2024).
[29] A. Daskevics, A. Nikiforova, Shobevodsdt: Shodan and binary edge based vulnerable open data sources detection tool or what internet of things search engines know about you, in: 2021 second international conference on intelligent data science technologies and applications (IDSTA), IEEE, 2021, pp. 38–45.
[30] T. Nasr, S. Torabi, E. Bou-Harb, C. Fachkha, C. Assi, Chargeprint: A framework for internet-scale discovery and security analysis of ev charging management systems, in: NDSS, 2023, p. "".
[31] S. Toro, A. V. Anagnostopoulos, S. Bello, K. Blumberg, R. Cameron, L. Carmody, A. D. Diehl, D. Dooley, W. Duncan, P. Fey, et al., Dynamic retrieval augmented generation of ontologies using artificial intelligence (dragon-ai), arXiv preprint arXiv:2312.10904 (2023).
[32] Y. He, Language models for ontology engineering, Ph.D. thesis, University of Oxford, 2024.
[33] R. Amini, S. S. Norouzi, P. Hitzler, R. Amini, Towards complex ontology alignment using large language models, arXiv preprint arXiv:2404.10329 (2024).
[34] J. H. Caufield, H. Hegde, V. Emonet, N. L. Harris, M. P. Joachimiak, N. Matentzoglu, H. Kim, S. Moxon, J. T. Reese, M. A. Haendel, et al., Structured prompt interrogation and recursive extraction of semantics (spires): A method for populating knowledge bases using zero-shot learning, Bioinformatics 40 (2024) btae104.
[35] M. Gruninger, Methodology for the design and evaluation of ontologies, in: Proc. IJCAI'95, Workshop on Basic Ontological Issues in Knowledge Sharing, 1995, p. "".
[36] N. F. Noy, D. L. McGuinness, et al., Ontology development 101: A guide to creating your first ontology, 2001.
[37] A. Bernardini, C. Carpineto, S. Angelini, G. Dondossola, R. Terruggia, Ask the right queries: Improving search engine retrieval of vulnerable internet-connected devices through interactive query reformulation, in: Proceedings of ITASEC 2024, 2024, p. "".
[38] M. A. Musen, The protégé project: a look back and a look forward, AI Matters 1 (2015) 4–12.
[39] P. Wang, L. Li, L. Chen, Z. Cai, D. Zhu, B. Lin, Y. Cao, Q. Liu, T. Liu, Z. Sui, Large language models are not fair evaluators, arXiv preprint arXiv:2305.17926 (2023).
[40] J. Rando, F. Croce, K. Mitka, S. Shabalin, M. Andriushchenko, N. Flammarion, F. Tramèr, Competition report: Finding universal jailbreak backdoors in aligned llms, arXiv preprint arXiv:2404.14461 (2024).
[41] A. Zou, Z. Wang, N. Carlini, M. Nasr, J. Z. Kolter, M. Fredrikson, Universal and transferable adversarial attacks on aligned language models, arXiv preprint arXiv:2307.15043 (2023) "".

Appendix

Ranking merging

To evaluate the overlapping of search engine results and understand their coverage and similarity we created 23 general search queries. We submitted the queries to IoTSEs, collecting and documenting the returned results. Table 1 show the results of the comparison between the five search engines, as well as between combinations of the engines. It should be noted that at the time of this analysis, an agreement had not yet been reached with Netlas for its use in the project activities, and therefore it was not included among the engines analyzed. The 'intersection' column indicates the IP addresses common to all five engines. For each engine, the number of IPs found is listed, with the number of distinct IPs for that engine in parentheses. For example, for query 18, Binaryedge reports 500 IPs, but only 50 are distinct. The intersection among all the engines is calculated only for the distinct IPs. The intersection of rankings obtained has yielded very few matches. Authors suppose it is primarily due to differences in evaluation criteria and result aggregation methods. Each engine may assign a different relevance score to devices depending on its search algorithm, the quality of available data, or the priorities set by its ranking system. Moreover, IoT search engines may operate on different network architectures and use varying methods for data collection, further influencing the consistency of the results.

Query-ID	Intersection	Shodan	Censys	Fofa	Binaryedge	ZoomEye
1	11	23(22)	74(74)	500(483)	101(61)	500(472)
2	52	445(429)	399(399)	500(477)	315(252)	500(483)
3	1	392(341)	465(465)	500(428)	279(177)	147(130)
4	3	18(14)	313(313)	500(388)	77(62)	500(436)
5	0	500(499)	500(500)	500(489)	500(26)	500(479)
6	0	500(478)	500(500)	500(450)	500(331)	500(440)
7	1	392(341)	465(465)	500(428)	279(177)	147(130)
8	57	485(464)	399(399)	500(477)	315(252)	500(483)
9	6	30(22)	48(48)	457(174)	24(14)	500(357)
10	0	500(499)	500(500)	500(343)	500(496)	500(244)
11	0	500(486)	500(500)	500(446)	500(362)	500(401)
12	0	500(480)	500(500)	500(445)	500(430)	500(477)
13	0	443(152)	0(0)	500(193)	500(154)	500(310)
14	0	0(0)	500(500)	500(478)	340(288)	500(489)
15	0	500(500)	408(408)	500(495)	0(0)	500(496)
16	0	500(440)	500(500)	500(476)	500(220)	500(366)
17	0	51(51)	9(9)	500(385)	1(1)	256(254)
18	0	500(265)	500(500)	500(463)	500(50)	500(234)
19	0	500(485)	500(500)	500(405)	500(258)	500(471)
20	0	500(500)	500(500)	500(495)	500(500)	500(493)
21	0	500(379)	500(500)	500(190)	500(79)	500(229)
22	0	3(3)	500(500)	500(339)	500(420)	500(491)
23	48	500(440)	493(493)	500(309)	500(376)	500(423)
TOTAL	179					

Table 1
Ranking overlaps

Qualitative Evaluation: Answers to Competency Questions

CQ 1: What vulnerabilities are associated with exposed hosts in a specific country?

In Fig. 6 shows a table listing the IP address, domain, ASN (*Autonomous System Name*) related to the "*ABB solar inverter*" in Italy. Regarding the identified CVEs, it is worth noting that CVEs were only identified on three out of seven hosts, exclusively by the Shodan engine. Moreover and it could topic for further investigation the three identified hosts show an identical distribution of vulnerabilities.

CQ 2: Do the identified results expose vulnerabilities considered dangerous for which exploits have already been identified?

To answer this question it is possible to correlate metadata as number of CVEs, presence of exploit and links to online resources explaining how to use such exploit as in Fig. 7 for "*Modicon M340*" query. The direct link to the exploit is a available in KB, but it falls outside the scope of this work.

CQ 3: Given a set of IP addresses, which services and products are identified by the IoT engines and with which CPEs?

The merged results combine the methodologies used by IoTSE to classify and describe devices, services, or systems detected during network scanning using tags. Analysis can be broader as in Fig.8 where all products identified for a query are shown. Alternatively, one can conduct more sophisticated analysis by examining specific characteristics as in Fig. 9 where 12 devices identified by searching for "*SmartPOWER Energy Management System*" (column 4 in the chart) are tagged as EOL (*End of Life*), indicating they will no longer receive updates and may be at risk. Another investigative approach involves examining firmware and software versions to identify older editions. Fig. 10 highlights partial details of CPEs (standard or enhanced 2.3 versions), showing firmware versions and occurrences for the query Solarview from firmware version 3.11 up to 4.0.3. These indicators serve as a starting point for deeper security analysis. However, such findings require rigorous validation, as they may have been patched on the server side through backporting of security updates.. Nonetheless, they offer a valuable external reconnaissance snapshot of an internet-connected device's potential vulnerabilities.

CQ 4: What weaknesses are associated to an IP address that could be exploited by a malicious actor?

This question can be addressed by exploring individual IoT results using the Protégé interface or querying the knowledge base (KB). The process involves extracting all CVEs associated with the specified IP address and using the ontology's hasCWE relationship to cross-reference and identify the corresponding weaknesses linked to each CVE as shown below:

- CVE-2006-3918 ['CWE-79']
- CVE-2022-28615 ['CWE-190']
- ...
- CVE-2022-31813 ['CWE-345', 'CWE-348']

	domains	autonomous_system_name	cve_Shodan_unverified
79.54.230.6	[host-79-54-230-6.retail.telecomitalia.it, tel...	ASN-IBSNAZ	NaN
79.42.134.168	[host-79-42-134-168.retail.telecomitalia.it, t...	ASN-IBSNAZ	NaN
94.33.229.190	tiscali.it	TISCALI-	['CVE-2008-3844', 'CVE-2019-16905', 'CVE-2016-...
87.17.208.175	[telecomitalia.it, host-87-17-208-175.retail.t...	ASN-IBSNAZ	NaN
79.37.64.254	[telecomitalia.it, host-79-37-64-254.retail.te...	ASN-IBSNAZ	['CVE-2008-3844', 'CVE-2019-16905', 'CVE-2016-...
79.42.0.18	[telecomitalia.it, host-79-42-0-18.retail.tele...	ASN-IBSNAZ	NaN
95.251.170.78	[host-95-251-170-78.retail.telecomitalia.it, t...	ASN-IBSNAZ	['CVE-2008-3844', 'CVE-2019-16905', 'CVE-2016-...

Figure 6: Results for ABB Solar Inverter

index	IP	Num_CVE	Exploit_Present	Num_Links
0	47.92.160.143	0	false	0
1	162.241.86.106	0	false	0
2	220.141.7.170	0	false	0
3	185.102.77.136	17	true	71
4	195.42.161.129	0	false	0
5	47.107.158.164	0	false	0
6	220.141.34.111	0	false	0
7	8.138.90.173	0	false	0

Figure 7: CVE, exploits and links to description

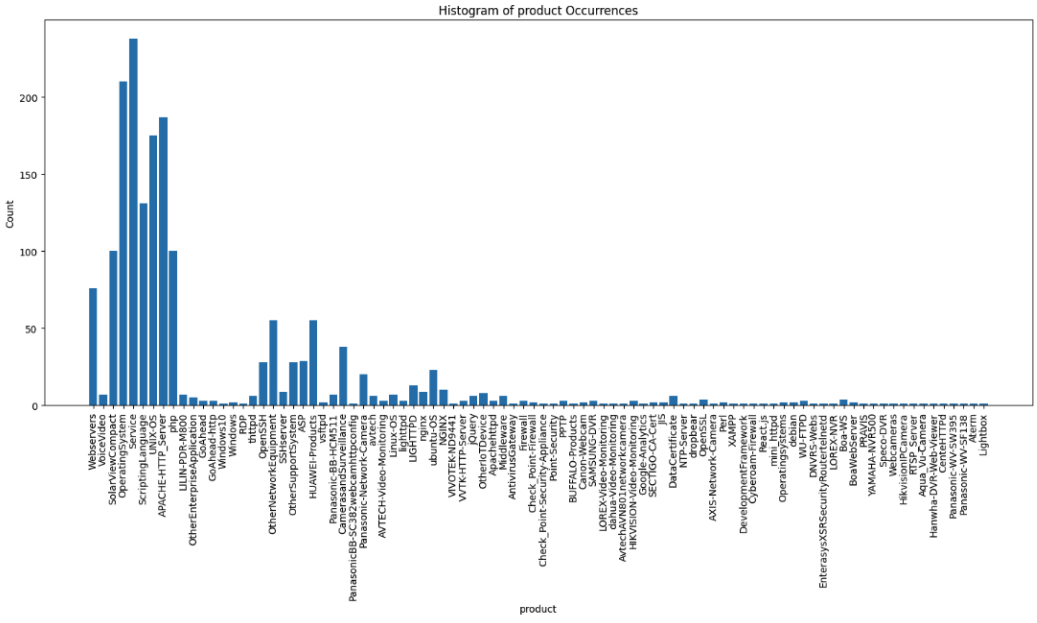

Figure 8: Detected products

The most frequent CWE identified (Fig.11) is associated is CWE-190 ("*Integer Overflow or Wraparound*"), which occurs when an arithmetic operation produces a result too large for the assigned integer variable to handle. An attacker using this weakness can manipulate application logic to generate scenarios where standard program flow is disrupted, potentially leading to infinite loops, system resource exhaustion, or complete application failure.

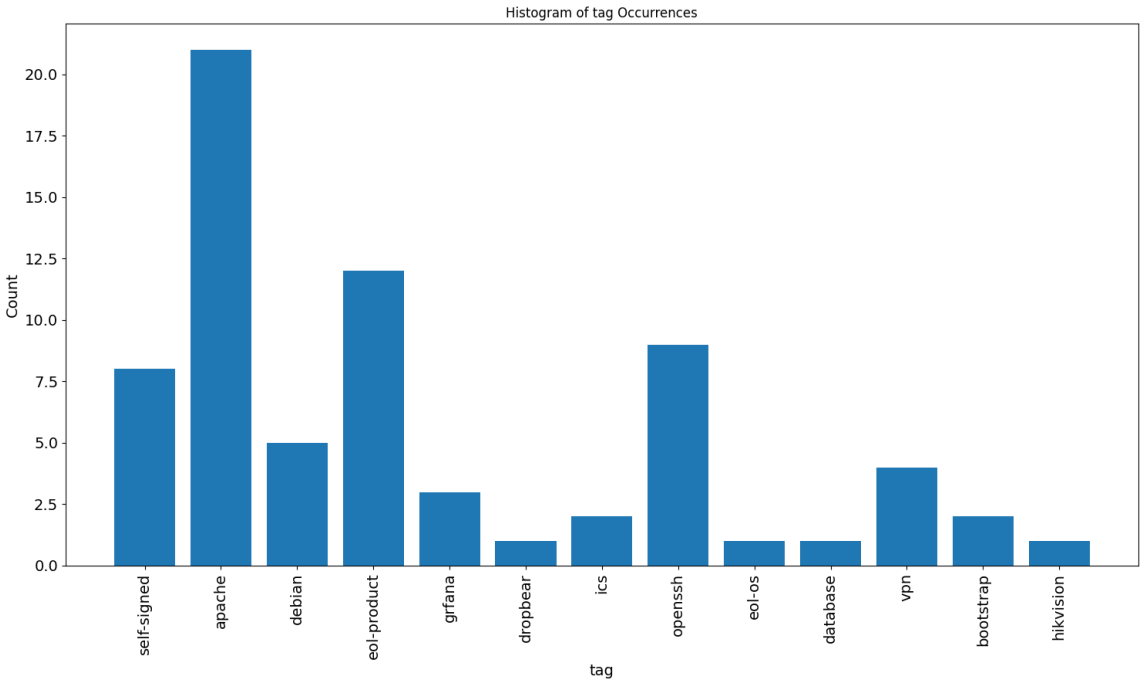

Figure 9: Assigned tags from IoT search engines

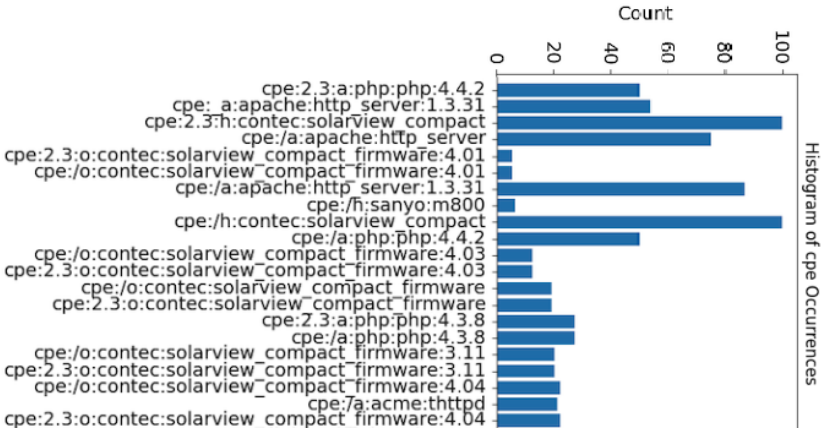

Figure 10: Detected CPEs

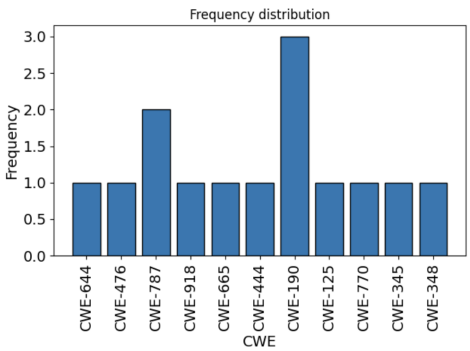

Figure 11: CWEs associated corresponding to a single ip