The Design and Application of a Unified Ontology for Cyber Security

Khandakar Ashrafi Akbar[1], Fariha Ishrat Rahman[1], Anoop Singhal[2], Latifur Khan[1], and Bhavani Thuraisingham[1]

[1] The University of Texas at Dallas
{khandakarashrafi.akbar, farihaishrat.rahman, lkhan, bxt043000}@utdallas.edu
[2] National Institute of Standards and Technology
anoop.singhal@nist.gov

Abstract. Ontology enables semantic interoperability, making it highly valuable for cyber threat hunting. Community-driven frameworks like MITRE ATT&CK, D3FEND, ENGAGE, CWE and CVE have been developed to combat cyber threats. However, manually navigating these independent data sources is time-consuming and impractical in high-stakes situations. By adopting an ontology-based approach, these cybersecurity resources can be unified, enabling a holistic view of the threat landscape. Additionally, leveraging semantic query languages empowers analysts to make the most of existing data sources. This paper explores how through the application of a semantic query language (SPARQL) on a unified cybersecurity ontology, analysts can effectively exploit the information contained within these resources to strengthen their defense strategies against cyber threats.

Keywords: Ontology· OWL · SPARQL · Cybersecurity

1 Introduction

In an era of unprecedented network expansion, the ever-growing scale of computer networks has paved the way for malicious entities to orchestrate large-scale attacks, posing a substantial risk to the individuals and organizations that rely on these interconnected systems. Compounding this threat is the relentless ingenuity of attackers, who seek out novel methods to infiltrate and compromise systems, requiring constant vigilance and robust defensive strategies to safeguard against these evolving cyber risks.

Advanced Persistent Threats (APTs) represent a category of highly sophisticated cyber threats carried out by known groups of actors, who have been identified by the tactics, techniques, and procedures (TTPs) they use [20]. While APTs pose significant challenges due to their evolutionary nature [10], the security community remains committed to its ongoing effort to improve defense against these attacks. Information sharing plays a crucial role in this effort; knowledge about the perpetrators, their methods, and targets is disclosed through various government and industry channels [20]. The frameworks and data sources, MITRE

ATT&CK, D3FEND, ENGAGE, CWE and CVE offer valuable insights into the tactics and techniques employed by threat actors, the weaknesses and vulnerabilities they exploit, and effective countermeasures. However, the information pertaining to APTs is scattered across these different resources, highlighting the need for consolidation to facilitate effective analysis and improve mitigation strategies.

Technical contextualization in cybersecurity is necessary for a prompt, successful cyber threat response. Particularly in post-compromise scenarios, the effectiveness of defending against adversarial behavior significantly improves when analysts can efficiently narrow down their focus, disregarding irrelevant information through analytics [21]. The key challenge faced by analysts is the overwhelming volume of information they need to access in order to effectively analyze incoming attacks. This information gap can significantly impede analysts when responding to time-sensitive threats. In our proposal, we introduce a tool that aims to bridge the information gap by providing analysts with relevant knowledge quickly and efficiently. This is achieved through the creation of an ontology, which streamlines the process of accessing pertinent information in just a few steps.

Ontologies are formal representations of knowledge about a domain, and they provide a structured way to represent the components and relationships of a network. By using ontologies, it is possible to gain a deeper understanding to aid in identifying and preventing the spread of attacks. We developed a knowledge base (KB) in the form of an ontology. This knowledge base (KB) serves to establish connections between various components within the cybersecurity domain. It links ATT&CK tactics and techniques, weaknesses documented in the Common Weakness Enumeration (CWE) database, vulnerabilities listed in the Common Vulnerabilities and Exposures (CVE) database, defensive solutions outlined in MITRE's D3FEND framework, and adversary engagement techniques from MITRE's ENGAGE framework.

By linking these different elements, the KB provides a comprehensive understanding of the relationships between attack techniques, weaknesses, vulnerabilities, defensive solutions, and adversary engagement techniques. This integrated information allows for a more holistic approach to cybersecurity, enabling organizations to identify potential threats, assess their impact, develop effective defensive strategies, and employ adversary engagement techniques to better understand and counteract adversaries.

Without proper inference capabilities provided by an existing KB, security analysts may struggle to extract the necessary information for pre-offensive and defensive tasks. Therefore, our ontology's inferential capability proves valuable in cybersecurity scenarios, such as detecting ongoing attack tactics or techniques and identifying the vulnerabilities that may have contributed to the situation. By utilizing the association information within our ontology, analysts can identify and remove the responsible application from other systems, mitigating potential repercussions.

To demonstrate the utility of our ontology, we can explore the following scenario. The APT kill chain unfolds in stages, and each stage can be linked to one or more tactics from the ATT&CK framework. Once the current stage of an APT campaign is identified, we can leverage SPARQL queries on our ontology to infer the potential tactics and techniques that might be utilized in the subsequent stages. This information enables us to make additional inferences and retrieve all the defensive countermeasures associated with these attack techniques. In the event that a specific attack technique is identified, we can retrieve the corresponding CVE tags, which can further assist in narrowing down the search for suitable countermeasures. Additionally, by retrieving a list of affected products that exhibit these vulnerabilities, we can proactively address and remedy the potential security issues.

Our ontology, coupled with the utilization of SPARQL queries and the integration of multiple data sources, facilitates a faster response to cybersecurity incidents.

Our contribution can be summarised as follows:

1. Construction of a Unified Ontology that serves as a comprehensive Knowledge Base encompassing APT tactics/techniques, weaknesses, vulnerabilities, adversary engagement techniques, and defense countermeasures.
2. Investigation of semantic queries (SPARQL) to extract relevant context and relationships from the Ontology, which can be utilized to draw meaningful inferences and accelerate the response process.

The rest of the paper is structured into six sections. Section 2 provides an introduction to semantic web technologies and the data sources utilized in building the ontology. In Section 3, the construction process of our ontology is explained. In Section 4, we demonstrate the practical use case of our ontology through example SPARQL queries that extract valuable insights. Section 5 discusses related work in the field, while Section 6 outlines potential areas for future research. Finally, Section 7 presents the conclusion of the study.

2 Background

In this section, we provide an introduction to the semantic web technologies employed for building and exploring our ontology. Subsequently, we present an overview of the data sources utilized in constructing the ontology.

2.1 Semantic Web Technologies

RDF [26], or the Resource Description Framework, is a versatile framework designed to represent interconnected data on the web. It provides a simple data model based on subject-predicate-object triples, allowing the description of relationships between resources. RDF's capacity to integrate data from diverse sources makes it a comprehensive proposition language, capable of unifying and consolidating heterogeneous data from multiple origins [23]. OWL [24] is an expressive language for creating ontologies. It extends RDF by providing additional constructs and vocabulary to define classes, properties, and relationships in a more structured and semantically rich manner. OWL allows for the specification of logical constraints, reasoning capabilities, and inference rules to enable automated reasoning and deduction over the ontology. SPARQL [25], a semantic query language for databases, is specifically designed to query and manipulate data stored in the Resource Description Framework (RDF) format. It has been employed to execute queries on the RDF data generated from the ontology.

2.2 Data Sources

The ontology is developed using cyber threat intelligence sourced from the following MITRE frameworks and datasources – ATT&CK (Adversarial Tactics, Techniques, and Common Knowledge), D3FEND (Detection, Denial, and Disruption Framework Empowering Network Defense), ENGAGE, CWE(Common Weakness Enumeration), and CVE(Common Vulnerabilities and Exposures).

ATT&CK [6] is a comprehensive knowledge base that focuses on adversary behavior and tactics observed in real-world cyber attacks. It categorizes various tactics, techniques, and sub-techniques used by threat actors, providing insights into their strategies and methodologies. The enterprise attack matrix has served as a foundational resource for constructing our ontology. This matrix provides an overview of 14 attack tactics, which are further categorized into 196 techniques and 411 sub-techniques. APT attacks follow a seven-stage kill chain [27], including Initial Compromise, Establish Foothold, Escalate Privileges, Internal Reconnaissance, Move Laterally, Maintain Presence, and Complete Mission. Attackers employ tactics from the MITRE ATT&CK framework throughout these stages. Understanding these stages and tactics helps organizations defend against APT attacks.

D3FEND [4] is designed to complement ATT&CK by focusing on defensive techniques and countermeasures. The D3FEND matrix describes 6 defensive tactics, which are further categorized into 22 techniques and 154 sub-techniques. These defensive tactics and techniques are directly linked to Digital Artifact Objects (DAOs), which, in turn, are connected to offensive techniques. The relationship between offensive techniques and defensive countermeasures is established through these DAOs.

ENGAGE [7] provides a framework that aligns defenders, vendors, and decision-makers by capturing real-world adversary behavior and guiding strategic cyber outcomes. The ENGAGE Matrix is composed of three main components: Goals, Approaches, and Activities. When adversaries exhibit specific behaviors or techniques from the ATT&CK framework, they inadvertently expose vulnerabilities or weaknesses. By understanding these weaknesses, we can devise engagement activities that exploit these vulnerabilities and enhance our defensive capabilities. Mapping the engagement activities in MITRE Engage to specific ATT&CK techniques ensures that each activity is directly informed by observed adversary behavior. For instance, if an adversary demonstrates the Remote System Discovery technique (T1018), they may be vulnerable to collecting, observing, or manipulating deceptive system artifacts or information. Armed with this knowledge, defenders can strategize and employ tactics such as using lures to elicit desired behaviors from the adversary, leveraging additional or advanced capabilities against the target, or influencing the adversary's dwell time within the compromised environment. [5]

CWE [3] is a collection of weaknesses found in software and hardware. These weaknesses can arise in various aspects such as architecture, design, code, or implementation, and can potentially lead to exploitable security vulnerabilities. The purpose of CWE is to provide a standardized language for describing these weaknesses, serving as a benchmark for security tools targeting these weaknesses, and establishing a common standard for identifying, mitigating, and preventing weaknesses. In essence, a weakness refers to a condition in a software, firmware, hardware, or service component that, in specific circumstances, could contribute to the introduction of vulnerabilities.

CVE [9] is a comprehensive list of publicly known vulnerabilities. Each entry in the Common Vulnerabilities and Exposures (CVE) database, includes an identification number, a description, and references to publicly known cybersecurity vulnerabilities. The entries may also provide additional information such as fixes, severity scores, impact ratings based on the Common Vulnerability Scoring System (CVSS), and links to exploit and advisory information. Weaknesses are errors that can lead to vulnerabilities [3], therefore, a connection can be established between CWE and CVE entries. This relationship between CVE and CWE implies that the Vulnerability is an example of the (type of) Weakness. [11]

3 Ontological Design

Our ontology builds upon the work presented by Akbar et al. [2] and extends its scope by incorporating the MITRE D3FEND and ENGAGE framework. Furthermore, in contrast to their approach of associating vulnerability information solely through the CVETag property, our work takes it a step further by integrating CWE weaknesses and CVE vulnerabilities as distinct classes. This expansion provides access to additional properties such as CVSS scores and information about specific products affected by the vulnerabilities. By incorporating these elements, our approach offers a more comprehensive representation of weaknesses, vulnerabilities, countermeasures and adversary engagements, enhancing the overall knowledge representation within the ontology.

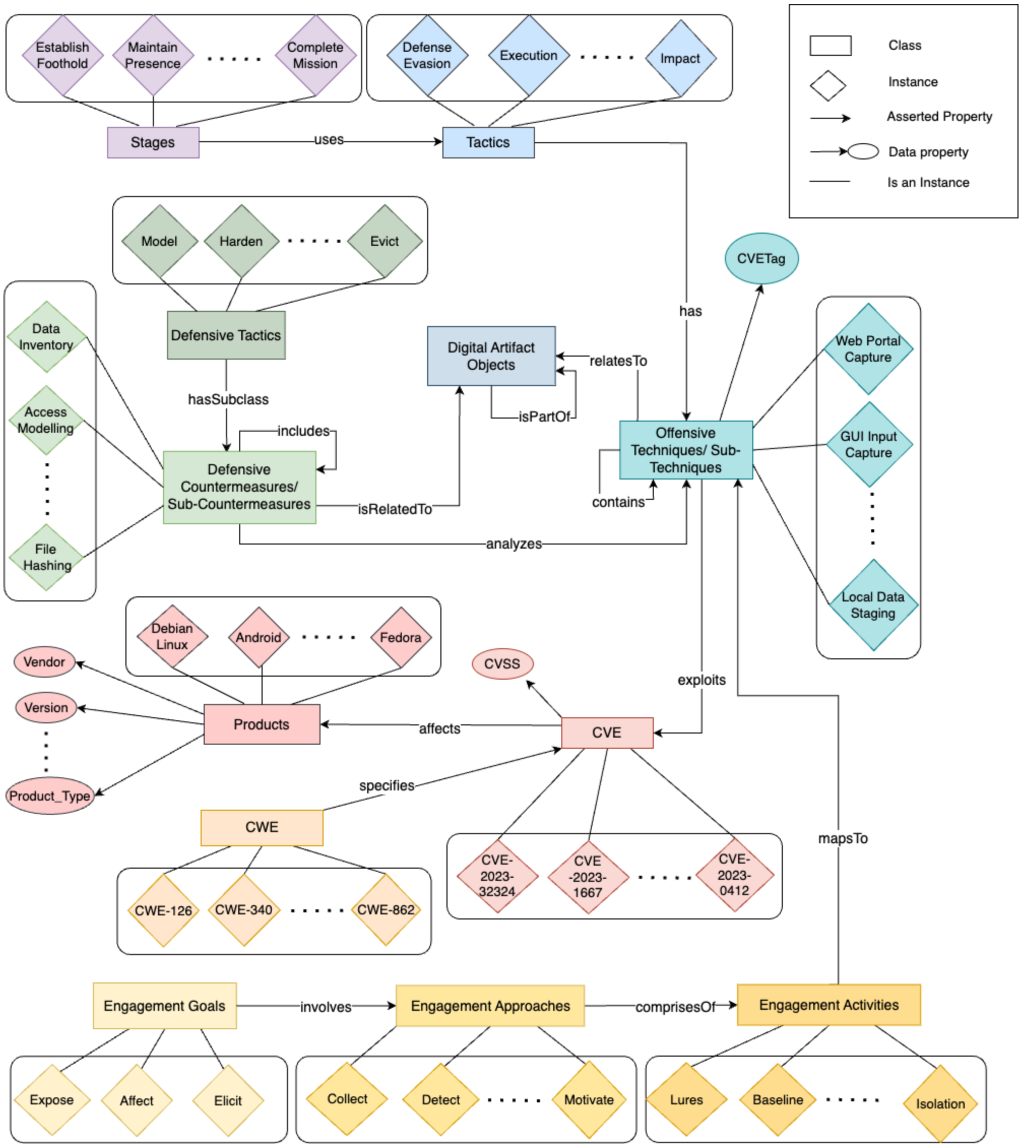


Fig. 1: Conceptual Representation of the Unified Ontology

3.1 WAVED: Unified Ontology

Fig 1 illustrates the conceptual representation of our proposed ontology – WAVED (**W**eakness, **A**tt&ck, **V**ulnerability, **E**ngage, **D**3fend). Our extended ontology encompasses a total of 14 classes – Stages, Tactics, Techniques, Sub-Techniques, CWE, CVE, Products, DAO (Digital Artifact Object), Defensive Tactics, Countermeasures, Sub Countermeasures, Engagement Goals, Engagement Approaches and Engagement Activities. These classes represent different entities within the ontology and are linked to each other. Table 1 provides an overview of the classes that were used to build the ontology and the number of instances for each class. Table 2 provides an overview of the object properties that links these classes together.

The relationships among the instances of these classes are defined below:

1. **uses:** APT attack employs attack tactics in different stages of the kill chain.
2. **has:** Each attack tactic can be accomplished via different attack techniques.
3. **contains:** Each attack technique may or may not contain sub-techniques.
4. **relatesTo:** This relationship links offensive techniques and sub-techniques to Digital Artifact Objects (DAOs).
5. **isRelatedTo:** Countermeasures and sub-countermeasures are linked to DAOs through this relationship.
6. **isPartOf:** Digital Artifact Objects (DAOs) form a hierarchical structure, with certain DAOs being part of other DAOs.
7. **analyzes:** This provides a direct link between defensive countermeasures/sub-countermeasures and offensive techniques/sub-techniques.
8. **hasSubclass:** Countermeasures are a subclass of defense tactics.
9. **includes:** Countermeasures may be further specified into sub-countermeasures.
10. **exploits:** Offensive techniques/sub-techniques exploit vulnerabilities documented in the CVE.
11. **affects:** A product is affected by one or more vulnerabilities listed in the CVE.
12. **specifies:** This relationship links CWE entries to CVE entries.
13. **mapsTo:** Engagement activities are mapped to attack techniques/sub-techniques.
14. **comprisesOf:** Engagement activities are organized under engagement approaches.
15. **involves:** Engagement approaches are organized under engagement goals.

Table 1: Overview of Ontology classes.

Class	Instances	Description
Stages	7	The seven stages of the APT kill chain: 1) Initial Compromise, 2) Establish Foothold, 3) Escalate Privileges, 4) Internal Reconnaissance, 5) Move Laterally, 6) Maintain Presence, and 7) Complete Mission.
Tactics	14	Refers to tactics from the attack matrix. An attack tactic refers to a high-level category or strategy used by threat actors to achieve their objectives. It represents a broad approach or methodology employed in a cyber attack. .
Techniques	196	The means by which threat actors achieve their tactical objectives. These techniques represent specific actions, methods, or tools used by adversaries to carry out their attacks. Each technique is associated with a particular tactic.
Sub-Techniques	411	ATT&CK techniques can be further categorized into sub-techniques.
Defensive Tactics	6	The elements of the D3FEND matrix are classified into six high-level categories: Model, Harden, Detect, Isolate, Deceive, and Evict. These categories serve as a framework for organizing and classifying the various techniques and countermeasures available to defenders.
Countermeasures	22	Each defensive tactic contains several techniques that describe how to implement appropriate strategies to counter cyber threats.
Sub-Countermeasures	154	Countermeasures can be further classified into sub-countermeasures.
DAO	521	The D3FEND matrix uses the concept of digital artifacts to establish connections between the defensive techniques with the offensive techniques from the ATT&CK framework.
CWE	1395*	List of publicly known weaknesses found in software and hardware.
CVE	206798*	List of publicly known cybersecurity vulnerabilities.
Products	37465*	List of products that are affected by one or more vulnerabilities documented in the CVE database.
Engagement Goals	3	ENGAGE matrix provides 3 engagement goals – Expose, Affect, and Elicit that describe the desired outcomes of adversary engagement operations.
Engagement Approaches	7	High-level methods or strategies used to engage the adversary.
Engagement Activities	23	Concrete techniques or actions used to implement the engagement approaches.

* To demonstrate the use case of our ontology, only a subset of these instances were utilized in its construction.

Table 2: Object Properties of the Ontology

Property	**Domain**	**Range**
uses	Stages	Tactics
has	Tactics	Techniques
contains	Techniques	Sub-techniques
relatesTo	Techniques, Sub-Techniques	DAO
isRelatedTo	Countermeasures, Sub-Countermeasures	DAO
isPartOf	DAO	DAO
hasSubclass	Defensive-Tactics	Countermeasures
includes	Countermeasures	Sub-Countermeasures
analyzes	Countermeasures, Sub-Countermeasures	Techniques, Sub-Techniques
exploits	Techniques, Sub-Techniques	CVE
affects	CVE	Products
specifies	CWE	CVE
mapsTo	Engagement-Activities	Techniques, Sub-Techniques
comprisesOf	Engagement-Approaches	Engagement-Activities
involves	Engagement-Goals	Engagement-Approaches

4 Querying Ontology for Security Insights

Semantic query language can be used to explore and make inferences from the ontology. In this section, we showcase examples of sparql queries that can be applied to our unified ontology to gain security insight and help analysts combat cyber threats.

SPARQL is a valuable tool for querying ontologies due to its capabilities in handling complex joins and relationships among entities and properties within the ontology. It enables the execution of analytic query operations, such as joins, sorting, aggregation, and filtering. SPARQL's flexibility and functionality make it well-suited for querying ontologies and extracting meaningful insights from the data they represent.

4.1 Simple Queries

In this section, we will explore some simple SPARQL queries that can be used to retrieve information from classes of a single datasource or to retrieve combined information by joining two datasources. An overview of these queries are presented in Table 3. The first column presents the queries expressed in English, while the second column presents the corresponding queries implemented in SPARQL.

Table 3: Examples of simple SPARQL queries

Query	**SPARQL**

Q1. Retrieve CVE tags for a certain adversarial technique (Command and Scripting Interpreter)	PREFIX WAVED: <ontologyURI>
	SELECT ?technique ?value
	WHERE {
	    ?technique WAVED:CVETag ?value.
	    FILTER(?technique=<ontologyURI
	            #Command_and_Scripting_Interpreter>).
	}

Q2. Retrieve products that might contain a particular vulnerability (CVE-2022-24663)	PREFIX WAVED: <ontologyURI>
	SELECT ?CVE ?Products
	WHERE {
	    ?CVE WAVED:affects ?Products.
	    FILTER(?CVE=<ontologyURI#CVE-2022-24663>).
	}

Q3. Retrieve specifics of a certain product (Chrome)	PREFIX WAVED: <ontologyURI>
	SELECT ?Products ?Product_Type ?Vendor ?Product
	        ?Edition ?Language ?Version ?Update
	WHERE {
	    ?Products WAVED:Product_Type ?Product_Type.
	    ?Products WAVED:Vendor ?Vendor.
	    ?Products WAVED:Product ?Product.
	    ?Products WAVED:Version ?Version.
	    ?Products WAVED:Update ?Update.
	    ?Products WAVED:Edition ?Edition.
	    ?Products WAVED:Language ?Language.
	    FILTER(?Products=<ontologyURI#Chrome>).
	}

Q4. Retrieve APT stage, tactic, technique/ sub-technique associated with a specific CVE Tag (CVE-2019-1943)	PREFIX WAVED: <ontologyURI>
	SELECT ?stage ?tactic ?technique ?sub-technique
	WHERE {
	    ?stage WAVED:uses ?tactic.
	    ?tactic WAVED:has ?technique.
	    ?technique WAVED:contains ?sub-technique.
	    ?technique WAVED:CVETag ”CVE-2019-1943”.
	}

Q5. Retrieve defensive countermeasures that are related to the same attack technique	PREFIX WAVED: <ontologyURI>
	SELECT ?countermeasure1 ?attack_technique1
	WHERE {
	    ?countermeasure1 WAVED:analyzes ?attack_technique1
	    {
	      SELECT ?countermeasure2 ?attack_technique2
	      WHERE {
	          ?countermeasure2 WAVED:analyzes
	                            ?attack_technique2.
	      }
	    }
	    FILTER(?attack_technique1= ?attack_technique2).
	}

Q6. Retrieve all defensive countermeasures for a certain attack technique (Boolkit)	PREFIX WAVED: <ontologyURI>
	SELECT ?countermeasures
	WHERE {
	        ?technique WAVED:analyzes ?attack_techniques.
	        FILTER(?attack_techniques =<ontologyURI
	                                #Boolkit>).
	}

Q7. Retrieve all connected attack techniques to a defensive countermeasure (Bootloader Authentication)	PREFIX WAVED: <ontologyURI>
	SELECT ?attack_techniques
	WHERE {
	      ?countermeasures WAVED:analyzes ?attack_techniques.
	      FILTER(?countermeasures =<ontologyURI
	                      #Bootloader_Authentication>).
	}

Q8. Associate defensive countermeasures with attack techniques through DAOs	PREFIX WAVED: <ontologyURI>
	SELECT ?countermeasure ?attack_technique
	WHERE {
	        ?countermeasure WAVED:isRelatedTo ?dao.
	        ?attack_technique WAVED:relatesTo ?dao.
	 }

Q9. Retrieve countermeasures when a specific attack technique is detected (Valid Accounts)	PREFIX WAVED: <ontologyURI>
	SELECT ?attack_techniques
	WHERE {
	     ?countermeasures WAVED:analyzes ?attack_techniques.
	     FILTER(?attack_techniques=<ontologyURI
	                            #Valid_Accounts>).
	}

Q10. Retrieve engagement activity mapped to a certain adversarial technique (Remote System Discovery)	PREFIX WAVED: <ontologyURI>
	SELECT ?technique ?engagement-activity
	WHERE {
	?technique WAVED:mapsTo ?engagement-activity.
	FILTER(?technique=<ontologyURI #Remote_System_Discovery>).
	}

Q11. Retrieve CWE entry associated to a specific CVE (CVE-2009-1699)	PREFIX WAVED: <ontologyURI>
	SELECT ?CWE ?CVE
	WHERE {
	?CWE WAVED:specifies ?CVE.
	FILTER(?CVE=<ontologyURI#CVE-2009-1699>).
	}

Q12. Retrieve all CVE entries associated to a specific CWE (CWE-611)	PREFIX WAVED: <ontologyURI>
	SELECT ?CWE ?CVE
	WHERE {
	?CWE WAVED:specifies ?CVE.
	FILTER(?CWE=<ontologyURI#CWE-611>).
	}

Q1 demonstrates how our ontology enables us to query and retrieve vulnerabilities that are susceptible to specific adversarial techniques. Conversely, we can also perform reverse queries to obtain information about adversarial stages, tactics, techniques, and sub-techniques linked to a particular CVE tag, as demonstrated in Q4 using the example "CVE-2019-1943". The result of this query is presented in Table 4. This highlights the bidirectional nature of our ontology in providing insights into the relationship between adversarial techniques and vulnerabilities.

Table 4: Result from SPARQL Query Q4

Stage	**Tactic**	**Technique**	**Sub-Technique**
Completed Mission	Impact	Data Manipulation	Transmitted Data Manipulation
Completed Mission	Impact	Data Manipulation	Stored Data Manipulation
Completed Mission	Impact	Data Manipulation	Runtime Data Manipulation
.....	.....	.....	.....

Our ontology also enables the identification of products that may be affected by specific vulnerabilities. Q2 exemplifies this by querying for such products. By delving deeper into the information, as shown in Q3, security analysts can obtain comprehensive details about the involved products. This knowledge empowers them to evaluate whether a specific version of a product is present in their system and take appropriate actions if any vulnerability associated with that version is detected.

Q5 provides a comprehensive list of defensive countermeasures organized by attack techniques. This query can be further refined to focus on specific attack techniques of interest. For example, Q6 retrieves the countermeasures associated with the "Boolkit" attack technique. By reviewing the list of defensive countermeasures linked to a particular attack technique, security analysts can explore alternative approaches if one countermeasure proves ineffective. This query empowers analysts to make informed decisions and adapt their defensive strategies based on the available options and their effectiveness in countering specific attack techniques. Conversely, Q7 focuses on retrieving all attack techniques connected to a specific defensive countermeasure, specifically those associated with "Bootloader Authentication". Table 5 presents a subset of the list of attack techniques linked to this defensive countermeasure, providing valuable insights into the potential threats that this countermeasure aims to mitigate.

Table 5: Result from SPARQL Query Q7

Attack Techniques
Software Packing
AppCert DLLs
Dynamic-Link Library Injection
Thread Execution Hijacking
File Deletion
Application Layer Protocol
....

The ontology leverages Digital Artifact Objects (DAOs) to establish connections between defensive countermeasures or sub-countermeasures and offensive techniques or sub-techniques. Q8 is designed to retrieve this mapping, showcasing the relationship between countermeasures and attack techniques. A snippet of the result from this query is provided in Table 6, offering a glimpse into the interconnectedness of defensive measures and offensive techniques within the ontology.

By correlating engagement activities in MITRE Engage with specific ATT&CK techniques, we can determine the appropriate engagement activity to exploit vulnerabilities demonstrated by adversaries. Q10 demonstrates this by retrieving

Table 6: Result from SPARQL Query Q8

Countermeasures	**Attack Techniques**
Credential Compromise Scope Analysis	OS Credential Dumping
Authentication Cache Invalidation	Additional Cloud Credentials
Credential Revoking	Unsecured Credentials
Decoy Session Token	Unsecured Credentials
.....	.....

the engagement activity associated with the use of the Remote System Discovery technique by adversaries.

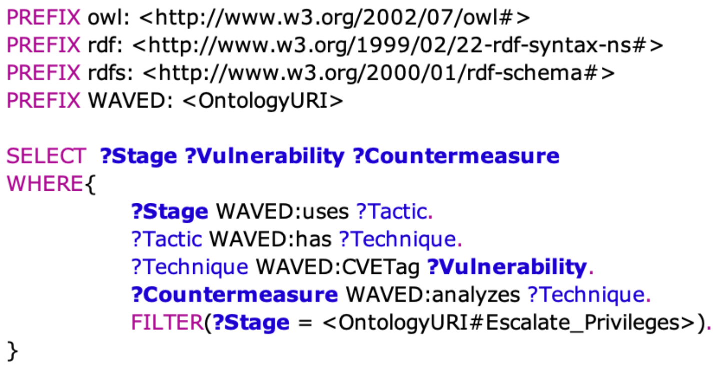
A CWE entry is associated with a collection of CVE vulnerability entries. This connection allows for the identification of similar vulnerabilities across different operating systems and applications, which can occur due to shared software development practices or coding styles. When anticipating an upcoming attack stage, it is crucial to not only focus on individual vulnerabilities but also understand the underlying weaknesses present in the system. By analyzing the weakness associated with a vulnerability, it becomes possible to proactively assess the potential presence of other vulnerabilities that may stem from the same weakness. This holistic approach, rather than addressing vulnerabilities in isolation, allows for a more comprehensive understanding of the system's security landscape. By querying the ontology, Q11 demonstrates how the weakness associated with a specific vulnerability can be retrieved, while Q12 showcases how all vulnerabilities organized under a particular weakness can be retrieved. These capabilities enable security analysts to take preemptive actions and mitigate potential risks before they can be exploited.

4.2 Advanced Queries

To fully unlock the potential of our ontology, having access to a complete and comprehensive context is vital. This underscores the importance of being able to execute a single query that can instantly retrieve information from multiple data sources. In this section, we explore complex queries that facilitate such capabilities.

The primary use case of our ontology is focused on detecting and mitigating Advanced Persistent Threat (APT) campaigns. When a specific stage of an APT campaign is detected in a system, it indicates that the attacker may advance to the next stage at any time. To effectively mitigate these threats, prompt identification and patching of vulnerabilities in the system is essential. It is not only crucial to identify the vulnerabilities that may be exploited but also to identify appropriate analysis tools to defend against or mitigate an imminent attack.

Fig 2 illustrates an example query in which the adversary has already established a foothold in the system. The next stage anticipated is "Escalate Privileges". By using this query, we can retrieve information about the vulnerabilities that the adversary is likely to exploit in the next stage. Additionally, the query allows us to identify the appropriate countermeasures that can be taken to defend against these vulnerabilities and hinder the adversary's progress.



```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX WAVED: <OntologyURI>

SELECT  ?Stage ?Vulnerability ?Countermeasure
WHERE{
            ?Stage WAVED:uses ?Tactic.
            ?Tactic WAVED:has ?Technique.
            ?Technique WAVED:CVETag ?Vulnerability.
            ?Countermeasure WAVED:analyzes ?Technique.
            FILTER(?Stage = <OntologyURI#Escalate_Privileges>).
}
```

Fig. 2: Retrieve Vulnerabilities and Countermeasures associated with the APT Stage "Escalate Privileges"

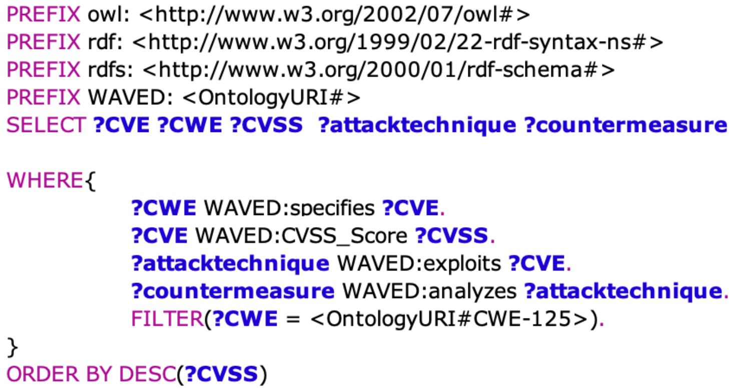
However, with the proliferation of technology across various domains, the number of vulnerabilities has significantly increased, posing a challenge in determining which vulnerabilities should be addressed without significant delays. Our ontology addresses this challenge by providing a list of associated vulnerabilities that require immediate attention. By extending the ontology to include severity scores of CVEs and information on software prone to these vulnerabilities, security analysts can prioritize and address the most critical vulnerabilities promptly.

Figure 3 demonstrates how when a weakness (CWE-125) is identified within a system, we can retrieve the vulnerability, attack technique and countermeasures associated with it in descending order of CVSS Score so that the most critical vulnerabilities may be tackled first.

In summary, our ontology plays a vital role in detecting APT campaigns, prioritizing vulnerability patching, identifying appropriate analysis tools, and selecting defensive countermeasures and engagement activities to effectively defend against sophisticated cyber threats. Its holistic approach to vulnerability management and threat mitigation enhances the effectiveness of cybersecurity efforts in addressing complex and evolving threats.

5 Related Work

Ontologies have proven to be effective and robust solutions for representing domain-specific knowledge, integrating data from diverse sources, and enabling



```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX WAVED: <OntologyURI#>
SELECT ?CVE ?CWE ?CVSS  ?attacktechnique ?countermeasure

WHERE{
            ?CWE WAVED:specifies ?CVE.
            ?CVE WAVED:CVSS_Score ?CVSS.
            ?attacktechnique WAVED:exploits ?CVE.
            ?countermeasure WAVED:analyzes ?attacktechnique.
            FILTER(?CWE = <OntologyURI#CWE-125>).
}
ORDER BY DESC(?CVSS)
```

Fig. 3: Retrieve CVE, Attack Techniques and Countermeasures for a given weakness (CWE-125) in descending order of CVSS Score

various semantic applications [19]. This is evident in various domains, including the Internet of Things (IoT) [16] and Information Selection [13, 14], where ontology-based approaches have been applied to enhance data integration, knowledge representation, and semantic reasoning.

According to the study conducted by JASON [8], constructing a common language and a set of basic concepts within the cybersecurity research community is vital for making significant progress in the field. As cybersecurity deals with adversaries, these concepts may evolve over time, but having a shared language and agreed-upon experimental protocols will facilitate hypothesis testing and concept validation.

Threat intelligence plays a crucial role in enhancing security operations by providing evidence-based knowledge about current and potential cyber threats. This leads to improved efficiency and effectiveness in detecting and preventing such threats. To effectively organize and represent this knowledge, tools like taxonomies, sharing standards, and ontologies are used. However, upon analyzing existing taxonomies, sharing standards, and ontologies, it becomes apparent that a comprehensive threat intelligence ontology is lacking [15]. This underscores the need for the development of a more encompassing ontology to address this gap and enable more cohesive and coherent cybersecurity research efforts.

In the domain of network security, several existing ontologies have been developed to capture domain-specific concepts and relationships. Obrst et al. [17] proposed a methodology for creating ontologies based on well-defined ones that can be used as modular sub-ontologies. They emphasized the usefulness of existing schemas, dictionaries, glossaries, and standards as a means of knowledge acquisition for defining an ontology.

Oltramari et al. [18] introduced a three-layer cyber security ontology called CRATELO with the goal of improving the situational awareness of security analysts and enabling optimal operational decisions through semantic representation. They built upon existing ontologies, extending them to include security-related middle-level ontology (SECCO) and low-level sub-ontology (OSCO) for capturing domain-specific scenarios related to threats, vulnerabilities, attacks, countermeasures, and assets.

STUCCO [12] is another notable example of a network security ontology that collects data from security systems and integrates it into a network security knowledge graph. It consolidates information from various structured data sources and establishes relationships among different entity types, such as software, vulnerabilities, and attacks.

The Unified Cybersecurity Ontology (UCO) [22], developed by Syed et al., focuses on integrating various cybersecurity ontologies, heterogeneous data schemes, and common cybersecurity standards to facilitate the sharing and exchange of cyber threat intelligence. UCO aims to unify the representation of threat and vulnerability data within knowledge graphs and ontologies.

Similarly, BRON [11] utilizes a single bidirectional graph to connect entries from different sources, ranging from tactics to vulnerable software. This relational approach enables the representation and analysis of various aspects of network security.

Our ontology improves upon existing implementations by integrating a more diverse and comprehensive range of data sources. It goes beyond just capturing attacks and vulnerabilities to include countermeasures and adversary engagement techniques, providing a broader scope for analysis. Additionally, our paper highlights how our ontology enriches the context and enhances inferential capabilities by leveraging semantic query language to explore the extensive and diverse data sources integrated within the ontology.

6 Limitations and Future Work

This paper primarily focuses on the modeling and querying of a cybersecurity ontology. However, there is room for future work beyond the scope of this paper that could explore semi-automating the construction process of the ontology. The manual effort required for establishing associations between classes within the ontology, such as defensive and offensive techniques, as well as linking attack techniques with CVE tags, does come with its limitations. It's worth noting that even existing frameworks like D3FEND rely on manual knowledge base generation, which demands significant human effort. The challenge becomes more evident when trying to associate new defensive techniques with existing attack techniques or zero-day attack methods. This underscores the pressing need for automation in the association process, where machine learning and data-driven approaches could offer substantial assistance [1].

Additionally, there may be gaps and missing links within the ontology that require attention. This presents an opportunity for future enhancement by incorporating natural language processing (NLP) techniques. The automation potential of NLP can prove invaluable in predicting and establishing these missing links within the ontology, significantly boosting its overall completeness and accuracy. Moreover, NLP can be leveraged to extract pertinent information from diverse sources such as reports, blogs, and threat report websites. By doing so, we can enrich the ontology with up-to-date insights and data. Through the synergistic integration of NLP and diverse data sources, the ontology can be expanded and improved upon.

To bolster the paper's contributions and provide tangible evidence of the ontology's effectiveness in real-world cybersecurity scenarios, empirical studies are imperative. At present, empirical validation is an aspect that remains unaddressed, yet it is absolutely vital for gauging the practical applicability of the proposed ontology. Looking ahead, we have concrete plans to take action in this regard. Specifically, we are committed to releasing an all-encompassing tool that incorporates our current ontology. This tool will be meticulously designed to assist security analysts in their day-to-day tasks. This practical tool will enable us to collect empirical evidence regarding the utility and real-world impact of our ontology.

7 Conclusion

The importance of curated knowledge in the cybersecurity field cannot be overstated. In the face of attack incidents that demand immediate and impactful actions, effective knowledge management plays a crucial role in providing guidance to security analysts. Whether it is an individual or an organization, minimizing the damage caused by cyber attacks hinges on the proper dissemination of information. Our ontology serves as a valuable tool for curating knowledge and assisting security analysts in effectively mitigating the continued spread of ongoing cyber attacks.

Disclaimer Certain equipment, instruments, software, or materials are identified in this paper in order to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement of any product or service by NIST, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

Acknowledgement The research reported herein was supported in part by NIST Award 60NANB23D007, NSF awards DMS-1737978, DGE-2039542, OAC-1828467, OAC- 1931541, and DGE-1906630, ONR awards N00014-17-1-2995 and N00014-20-1-2738.

References

1. Akbar, K.A., Halim, S.M., Hu, Y., Singhal, A., Khan, L., Thuraisingham, B.: Knowledge mining in cybersecurity: From attack to defense. In: IFIP Annual Conference on Data and Applications Security and Privacy. pp. 110–122. Springer (2022)

2. Akbar, K.A., Halim, S.M., Singhal, A., Abdeen, B., Khan, L., Thuraisingham, B.: The design of an ontology for att&ck and its application to cybersecurity. In: Proceedings of the Thirteenth ACM Conference on Data and Application Security and Privacy [Poster Presentation]. pp. 295–297 (2023)
3. Corporation, M.: Common weakness enumeration. `https://cwe.mitre.org/`
4. Corporation, M.: A knowledge graph of cybersecurity countermeasures. `https://d3fend.mitre.org/`
5. Corporation, M.: Mapping the engage matrix to mitre att&ck. `https://engage.mitre.org/wp-content/uploads/2022/05/Mapping-Engage-to-ATTCK.pdf`
6. Corporation, M.: Mitre att&ck. `https://attack.mitre.org/`
7. Corporation, M.: Mitre engage. `https://engage.mitre.org/`
8. Corporation, M.: Science of cyber-security. `https://irp.fas.org/agency/dod/jason/cyber.pdf`
9. Corporation, M.: The ultimate security vulnerability data source. `https://www.cvedetails.com`
10. CSRC, N.: Advanced persistent threat. `https://csrc.nist.gov/glossary/term/advanced_persistent_threat`
11. Hemberg, E., Kelly, J., Shlapentokh-Rothman, M., Reinstadler, B., Xu, K., Rutar, N., O'Reilly, U.M.: Linking threat tactics, techniques, and patterns with defensive weaknesses, vulnerabilities and affected platform configurations for cyber hunting. arXiv preprint arXiv:2010.00533 (2020)
12. Iannacone, M., Bohn, S., Nakamura, G., Gerth, J., Huffer, K., Bridges, R., Ferragut, E., Goodall, J.: Developing an ontology for cyber security knowledge graphs. In: Proceedings of the 10th Annual Cyber and Information Security Research Conference. pp. 1–4 (2015)
13. Khan, L., McLeod, D., Hovy, E.: Retrieval effectiveness of an ontology-based model for information selection. the VLDB Journal **13**, 71–85 (2004)
14. Luo, F.: Ontology construction for information selection. In: 14th IEEE International Conference on Tools with Artificial Intelligence, 2002.(ICTAI 2002). Proceedings. pp. 122–127. IEEE (2002)
15. Mavroeidis, V., Bromander, S.: Cyber threat intelligence model: an evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence. In: 2017 European Intelligence and Security Informatics Conference (EISIC). pp. 91–98. IEEE (2017)
16. Mozzaquatro, B.A., Agostinho, C., Goncalves, D., Martins, J., Jardim-Goncalves1, R.: "an ontology-based cybersecurity framework for the internet of things. Sensors (Basel, Switzerland) **18(9): 3053** (2017). https://doi.org/10.3390/s18093053
17. Obrst, L., Chase, P., Markeloff, R.: Developing an ontology of the cyber security domain. In: Semantic Technologies for Intelligence, Defense, and Security (STIDS). pp. 49–56 (2012)
18. Oltramari, A., Cranor, L.F., Walls, R.J., McDaniel, P.D.: Building an ontology of cyber security. In: Semantic Technologies for Intelligence, Defense, and Security (STIDS). pp. 54–61 (2014)
19. Salatino, A.A., Thanapalasingam, T., Mannocci, A., Birukou, A., Osborne, F., Motta, E.: The computer science ontology: A comprehensive automatically-generated taxonomy of research areas. Data Intelligence **2**(3), 379–416 (2020)
20. Shlapentokh-Rothman, M., Kelly, J., Baral, A., Hemberg, E., O'Reilly, U.M.: Coevolutionary modeling of cyber attack patterns and mitigations using public datasets. In: Proceedings of the Genetic and Evolutionary Computation Conference. pp. 714–722 (2021)

21. Strom, B.E., Battaglia, J.A., Kemmerer, M.S., Kupersanin, W., Miller, D.P., Wampler, C., Whitley, S.M., Wolf, R.D.: Finding cyber threats with att&ck-based analytics. The MITRE Corporation, Bedford, MA, Technical Report No. MTR170202 (2017)
22. Syed, Z., Padia, A., Finin, T., Mathews, L., Joshi, A.: Uco: A unified cybersecurity ontology. UMBC Student Collection (2016)
23. Tomaszuk, D., Hyland-Wood, D.: Rdf 1.1: Knowledge representation and data integration language for the web. Symmetry **12**(1), 84 (2020)
24. World Wide Web Consortium (W3C): Owl web ontology language guide. Tech. rep., World Wide Web Consortium (2004), `https://www.w3.org/TR/owl-guide/`
25. World Wide Web Consortium (W3C): Sparql query language for rdf. Tech. rep., World Wide Web Consortium (2008), `https://www.w3.org/TR/rdf-sparql-query/`
26. World Wide Web Consortium (W3C): Resource description framework (rdf). Tech. rep., World Wide Web Consortium (2014), `https://www.w3.org/RDF/`
27. Zou, Q., Sun, X., Liu, P., Singhal, A.: An approach for detection of advanced persistent threat attacks. Computer **53**(12), 92–96 (2020)