Exploring Ontologies for Mitigation Selection of Industrial Control System Vulnerabilities

Thomas Heverin[1], Michael Cordano[2], Andy Zeyher[1], Matthew Lashner[1] and Sanjana Suresh[1]
[1]Drexel University, Philadelphia, USA
[2]Gryphon Technologies, Philadelphia, USA
th424@drexel.edu
Michael.cordano@gryphontechnologies.com
az458@drexel.edu
mil28@drexel.edu
ss5264@drexel.edu

Abstract: Mitigating vulnerabilities in industrial control systems (ICSs) represents a highly complex task. ICSs may contain an abundance of device types, all with unique software and hardware components. Upon discovering vulnerabilities on ICS devices, cyber defenders must determine which mitigations to implement, and which mitigations can apply across multiple vulnerabilities. Cyber defenders need techniques to optimize mitigation selection. This exploratory research paper shows how ontologies, also known as linked-data models, can potentially be used to model ICS devices, vulnerabilities, and mitigations, as well as to identify mitigations that can remediate or mitigate multiple vulnerabilities. Ontologies can be used to reduce the complexity of a cyber defender's role by allowing for insights to be drawn, especially in the ICS domain. Data are modelled from the Common Platform Enumeration (CPE), the National Vulnerability Database (NVD), standardized list of controls from the National Institute of Standards and Technology (NIST), and ICS Cyber Emergency Response Team (CERT) advisories. Semantic queries provide the techniques for mitigation prioritization. A case study is described for a selected programmable logic controller (PLC), its known vulnerabilities from the NVD, and recommended mitigations from ICS CERT. Overall, this research shows how ontologies can be used to link together existing data sources, to run queries over the linked data, and to allow for new insights to be drawn for mitigation selection.

Keywords: ontology, industrial control systems, vulnerability management

1. Introduction

Industrial control systems (ICSs) support the real-time control of large-scale operations and industrial processes in entities such as electricity distribution systems, water treatment facilities, nuclear power plants, manufacturing plants, smart cities, and military weapons systems. The National Institute of Standards and Technology (NIST) (2018, p. 98) states that ICSs consist of "combinations of control components (e.g., electrical, mechanical, hydraulic, pneumatic) that act together to achieve an industrial objective (e.g., manufacturing, transportation of matter or energy."

The U.S. Cybersecurity and Infrastructure Security Agency (CISA) (2020) considers many types of ICSs to be "...so vital to the United States that their incapacitation or destruction would have a debilitating effect on security, national economic security, national public health or safety, or any combination thereof." As ICSs are vital to society, they have often been targeted by cyber attackers. One of the most well-known ICS cyber-attacks was Stuxnet. Stuxnet was the first weaponized cyber-attack against an ICS. The worm targeted Iranian nuclear plant centrifuges in 2010, causing substantial damage to the Iranian nuclear program (Knapp and Langhill, 2015). More recently, in the U.S there have been cyber-attacks on water treatment facilities, intrusions into dams, ransomware in meat processing plants, and breaches in car manufacturing plants. Cyber defenders of ICSs play a critical role. Cyber defenders must identify the assets (hardware and software) found in ICSs, determine vulnerabilities found on those assets, and select mitigations to protect the assets. Completing these steps can be quite time consuming and challenging. The goal of this research paper is to reduce the complexity of protecting ICSs.

2. Background

2.1 Complexity of ICS Vulnerability Management

ICSs are complex systems to defend for various reasons. Bristow (2020) stated that maintaining the inventory of all the assets found in an ICS represents a time-consuming and costly effort. With potentially hundreds of unique ICS devices on a network, along with traditional hardware and software assets, it can be a daunting task to keep

track of every asset on the network (Arnaert, Bertrand & Boudaoud, 2016). Veitch et al (2013) stated that ICSs lack built-in security features, lack mitigation technologies, and use protocols with many vulnerabilities.

Furthermore, with many devices included in an ICS, the attack surface of an ICS network can be considerably larger than a general company's network. The larger attack surface of an ICS network can make it easier for attackers to find openings to infiltrate the ICS. Additionally, ICS devices typically hold unique software and hardware which can be hard to maintain and upgrade. This means that in many cases, ICS devices are often unpatched leaving vulnerabilities exposed for years (Bristow, 2020).

In vulnerability management, cyber defenders must synthesize data from multiple sources to detect, respond and recover from attacks (D'Amico et al, 2005; Syed, 2020). To find and organize vulnerabilities, cyber defenders use specialized searches on the Internet; however, these searches can result in thousands of results that "...can be complex to sort, understand, and analyze." (Arnaert, Bertrand & Boudaoud, 2016 p. 299). With all the data that must be synthesized, cyber defenders face a cognitive overload which negatively impacts their work (D'Amico et al, 2005; Tyworth et al, 2012; Yen et al, 2010). Furthermore, cyber defenders work in high-stakes domains that require quick reaction time to mitigate vulnerabilities before attackers develop exploits (Joh and Malaiya, 2010). There are many demands on cyber defenders in protecting ICS networks, especially since cyber defenders must sift through many data sources. Ontologies can provide one way to help lessen these data-intensive demands. Ontologies provide the foundation for the discovery of new relationships within a domain, and across domains, especially when various data sources are linked together (World Wide Web Consortium, 2015).

2.2 Ontologies for Decision Making in Vulnerability Management

Ontologies have been identified as one way to integrate vulnerability data from multiple sources. Ontologies provide formalized, explicit knowledge representations of domains, including objects in those domains and relationships between those objects (Guariono, 1998; Syed, 2020). Ontologies can model relationships between data, integrate data, and allow for reasoning across the data (Wan, Guo & Camargo, 2010; World Wide Web Consortium, 2015). According to Syed (2020), although real-world vulnerability management involves integrating data from multiple vulnerability data sources, there is limited research in this area.

Wang and Guo (2009) developed an ontology for vulnerability management by modelling key entities and relationships such as vulnerabilities, attacks on the vulnerabilities, and countermeasures. It used the National Vulnerability Database (NVD) as its primary source. Fenz and Ekelhart (2009) created a general ontology of information security, with a small part focused on vulnerabilities. The ontology was modelled across various sources including NIST sources and others. Ontologies have been used to model vulnerabilities and requirements in the cybersecurity domain (Elahi, Yu, and Zannone, 2009). Vorobiev and Bekmamedova (2010) created several ontologies, one of which focuses on vulnerabilities of assets. They linked together cybersecurity policies with vulnerability mitigations.

Salini and Shenbagam (2015) included vulnerabilities in their ontology model to develop techniques for attack prediction. They used three models in their ontology: a threat model, a vulnerability model, and an attack model. Arnaert, Bertrand, and Boudaoud (2016) developed an ontology to model vulnerabilities of ICS objects to reduce the number of aggregated results, increase the relevance of results, and be usable for non-cybersecurity experts. They found that across various cybersecurity data sources, there lacks correlation between vulnerabilities and Computer Emergency Response Team (CERT) data elements. However, they only focused on one type of vulnerability (default credentials) in their research.

Syed (2020) developed a vulnerability management ontology to better inform users about impending vulnerabilities and countermeasures. Data sources included official cybersecurity data sources, along with social media intelligence. Although there have been multiple research papers authored about using ontologies for vulnerability management, Syed (2020) stated that critical items are missing from ontology-vulnerability models: the lack of context in vulnerability modelling (including CVSS scores) and a lack of linking to countermeasures to use against vulnerabilities. Our research aims to help fill these gaps in research on vulnerability management ontologies.

3. ICS Mitigation Ontology

3.1 Ontology Domain Concepts

In ontology development, it is critical for domain experts to identify domain concepts and relationships among the concepts (Sugumaran and Storey, 2002). The authors of this paper have over a decade of experience in the ICS cybersecurity domain, especially in the U.S. Navy ICS domain. In the vulnerability-management domain, main concepts include assets, vulnerabilities on those assets, and mitigations that can be used on those vulnerabilities (Syed, 2020). We used this common knowledge in vulnerability-management to explicitly link together ICS product, ICS vulnerability, and ICS mitigation data sources.

ICSs consist of assets including computers, human machine interfaces (HMIs), sensors, and programmable logic controllers (PLCs). PLCs represent critical ICS assets and play a central role in ICSs. ICS technicians configure PLCs to manage specialized inputs and outputs from devices such as actuators, valves, motors, engines, fans, pumps, and sensors (such as for geographic position, temperature, liquid levels etc). PLCs also consist of logic programs that automate the control and monitoring of industrial processes (Ghaleb et al, 2018). In the naval warfare domain, PLCs can be used to control many ship systems including steering, navigation, engineering, missile launch, radar, and gun systems. Given that PLCs play critical roles in ICSs, PLCs represent the focus of this study.

Vulnerabilities can be found on all types of assets. ICS assets, although critical assets, may have multiple vulnerabilities. According to NIST (2021), a vulnerability is "a weakness in the computational logic (e.g., code) found in software and hardware components that, when exploited, results in a negative impact to confidentiality, integrity, or availability." Many vulnerabilities have been found on PLCs across various vendors (such as Siemens and Allen-Bradley Rockwell Automation).

Mitigations are protections that cyber defenders can use to remediate, or mitigate the risk of, identified vulnerabilities. Example mitigations include using a virtual private network (VPN), updating software to a newer version, restricting port access, minimizing network exposure, and more.

3.2 Ontology Data Sources

We explicitly model these three concepts (assets, vulnerabilities, and mitigations), as well as other concepts, in a Mitigations Ontology. To best represent these concepts, we use real-world data sources including sources that help standardize or provide asset names, vulnerabilities, and mitigations.

Since many different companies may call the same hardware or software asset by different names, the Common Platform Enumeration (CPE) specification provides a standardized way to name all types of assets including ICS assets (MITRE, 2013). As an example, the CPE standardized format for a S7-300 PLC is this: cpe:2.3:h:siemens:simatic_s7-300. By using CPE as a naming method for ICS assets, we can link ICS assets to various other data sources that also rely on the CPE nomenclature.

The National Vulnerability Database (NVD) collects and stores publicly known vulnerabilities. The NVD is one of the largest publicly-available vulnerability databases (Blinokwski and Piotrowski, 2020). NVD vulnerabilities are stored as Common Vulnerability and Exposure (CVEs) entries. CVE entries include a description of the vulnerability or exposure, as well as the type of vulnerability. These entries also include the product vendor and affected firmware or software versions in CPE format. CVE entries contain metadata in terms of the confidentiality, integrity, and availability-impact metrics as well as exploitability metrics. These metrics provide values according to the Common Vulnerability Scoring System (CVSS). The NVD utilizes the CVSS standard to rate vulnerabilities based on their potential impact to the affected system or software (Forum of Incident Response and Security Teams, 2021).

An example CVE found on the S7-300 PLC is CVE-2016-9158. For CVE-2016-9158, remote attackers can cause the PLC to go into defect mode, requiring a cold start of the PLC. The CVSS severity is score 7.5 (out of 10), the impact subscore is 3.6 and the exploitability subscore is 3.9

Although the NVD provides explicit data and metrics/scores about vulnerabilities, it does not directly provide mitigations for those vulnerabilities. It may provide links to external resources, such as vendor-specific bulletins, for vulnerabilities. In the ICS domain, cyber defenders must research ICS CERT advisories to find mitigations. ICS CERT advisories provide information including executive summaries, affected products, vulnerability overview,

links to CVEs, and mitigations (Cybersecurity and Infrastructure Security Agency, 2021). The mitigations in ICS CERT advisories are provided in full-text sentences or paragraphs which poses a challenge for ICS cyber defenders as they must spend time reading long paragraphs to determine which mitigations to use. NIST 800.53 (revision 5.1) provides a standardized list of controls (mitigations) one can use to secure information systems. However, ICS CERT mitigations that are described in full-text are not mapped to NIST 800.53 controls. Our ontology lays the foundation to expand on this lack of mapping.

3.3 Ontology Structure

Ontologies consist of three main types of entities: classes, object properties, and data properties. We used Protégé, a common ontology development tool, for our modelling. Classes represent the main concepts to be modelled, and signify categories of objects. Figure 1 shows the main classes in our Mitigations Ontology. For example, the "Vulnerability" class consists of CVEs from the NVD like CVE-2016-9158 while the "ICSDevice" class consists of device names like the S7-300 PLC.

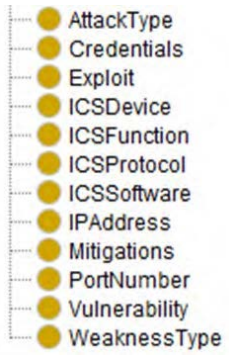

Figure 1: Classes in the Mitigations Ontology

The Mitigations class contains specific mitigations. As mentioned previously, ICS CERT advisories provide recommended mitigations in full-sentence and paragraph form. We manually developed a vocabulary of mitigations from analysing multiple ICS CERT advisories on PLCs. Then we mapped the sentences from ICS CERT advisories that described mitigations to our vocabulary of mitigations. Furthermore, we mapped our vocabulary to NIST 800.53 (revision 5.1) that provides a standardized list of controls. Two researchers independently developed vocabularies. Then those vocabularies were compared and finally synthesized into one final ICS Mitigation Vocabulary. The current ICS Mitigation Vocabulary is shown in Figure 2. The list in Figure 2 forms the objects of the Mitigations class.

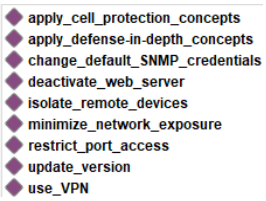

Figure 2: Objects Found in the Mitigations Class.

Object properties are used to formalize the links between classes and then to specify links between named objects within those classes. As an example, the object property *canMitigate* is the name of the object property that links together the classes "Mitigations" and "Vulnerability." One can read this link as: "Mitigation *canMitigate* Vulnerability." We can then "fill out" or specify this object property (*canMitigate*) at the object level. The following is an example of this:

- "deactivate_web_server" (a mitigation) *canMitigate* CVE-2016-9158 (a vulnerability).

When object properties are specified, they are considered to be triples. Triples consist of a subject, predicate, and object. When triples are formed, they are classified as being in the Resource Description Framework (RDF) format. RDF provides a formalized way to model with ontologies. Figure 3 shows examples of the object property *canMitigate* specified, linking specific mitigations with specific CVEs.

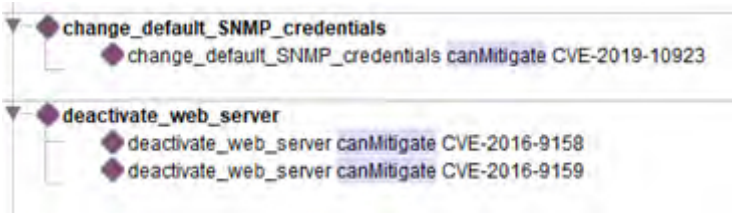

Figure 3: Examples of the CanMitigate Object Property in the Mitigations Ontology

With object properties we can create a chain of links. The below is an example showing a specific device linked with a specific vulnerability that is then linked to a specific mitigation:

- S7-300 PLC *hasVulnerability* CVE-2016-9158 and deactivate_web_server *canMitigate* this vulnerability.

Data properties provide metadata about objects in an ontology. They provide descriptive details about objects. As an example, Figure 4 shows key data properties for vulnerabilities that are drawn from the CVSS. For example, in regard to CVE-2016-9158, the CVSS base score is 7.5 (out of 10), the ImpactSubScore is 3.6 and the Exploitability Subscore is 3.9.

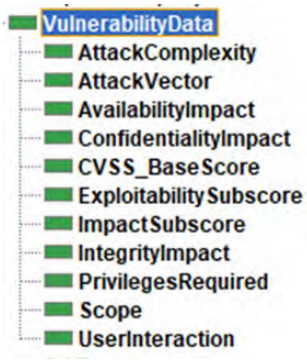

Figure 4: Vulnerability Data Properties Found in the Mitigations Ontology

We can also use data properties to provide names of objects in unique syntax such as CPE syntax. For example, we use a data property called "CPE" associated with ICS Devices. For the S7-300 this CPE data property is: cpe:2.3:h:siemens:simatic_s7-300. With Mitigation Ontology classes, object properties, and data properties specified, one can query over the ontology to draw insights and make decisions by using queries.

4. Ontology Queries and Results

4.1 SPARQL for Ontology Queries

SPARQL, a recursive acronym for SPARQL Protocol and RDF Query Language, is one common query language used to query RDF data (triples) in ontologies (World Wide Web Consortium, 2008). SPARQL holds many similarities with Structured Query Language (SQL). One advantage of using SPARQL is that, while being able to still query data out of relational databases like SQL, it is designed to combine vastly different data from several sources. In terms of syntax, SPARQL uses similar commands found in SQL, including SELECT, FROM, WHERE, UNION, GROUP BY, HAVING, and more. Furthermore, the results of SPARQL queries can be used with visualizations to allow for the development of insights. Protégé, provides OntoGraph, a visualization tool, to visualize relationships within ontologies. Examples of these are shown below.

4.2 SPARQL Queries for Vulnerability Management

In our queries, we first wanted to search across our Mitigations Ontology to identify all vulnerabilities associated with S7-300 PLC. The syntax is this SPARQL is provided below:

```
SELECT  ?Asset ?Vulnerability
WHERE  { Asset ics:hasVulnerability ?Vulnerability . FILTER (?Asset = ics:S7-300 ) .}
ORDERBY (?Vulnerability)
```

The above query produces 17 vulnerabilities (CVEs) that are associated with S7-300 PLCs. The results are visualized in Figure 5 via OntoGraph in Protégé. Each object in the Figure 5 can be examined by clicking on it to see additional object properties as well as data properties associated with that object.

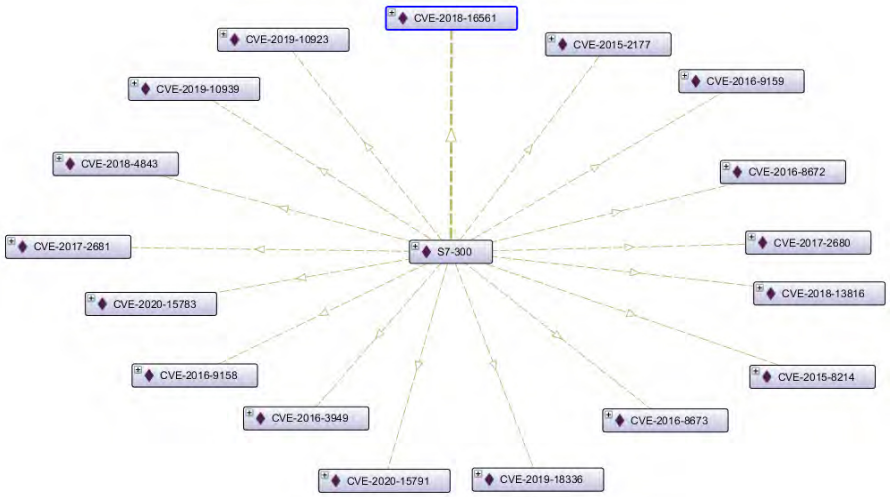

Figure 5: Visual of SPARQL Query Results that Show Vulnerabilities for S7-300 PLCs

When given a list of vulnerabilities, cyber defenders need prioritize which mitigations to use and determine, if possible, which mitigations could be used across multiple CVEs. Figure 6 shows how mitigations are mapped to S7-300 CVEs based off object property specifications ("canMitigate") within the Mitigations Ontology.

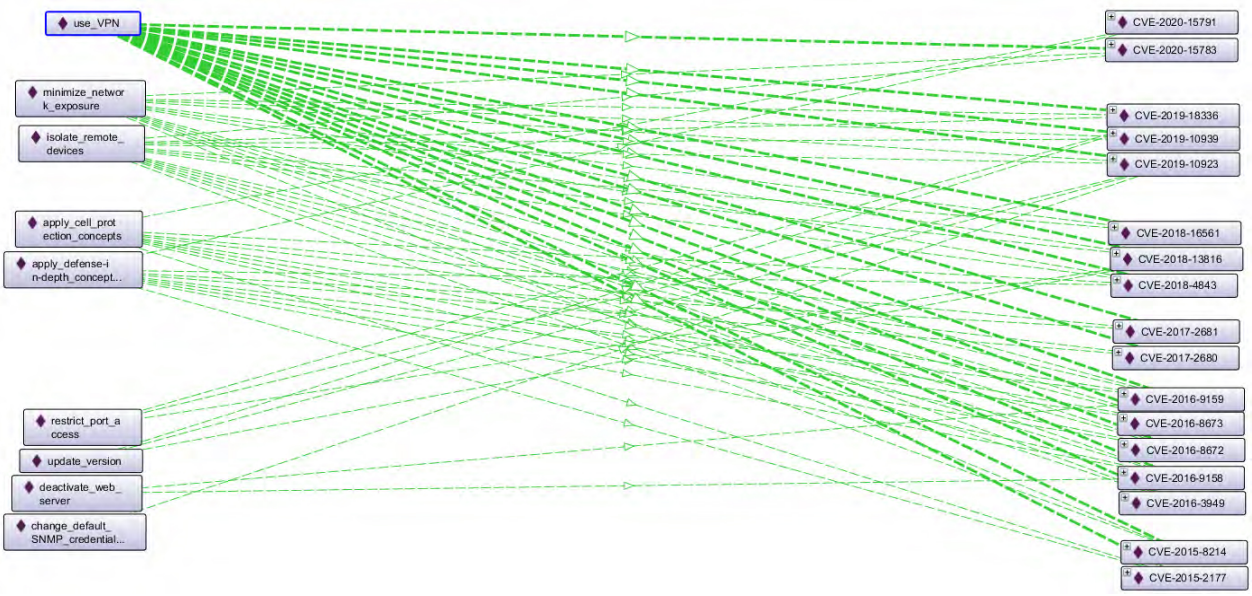

Figure 6: Visualization of Mitigations Mapped to S7-300 PLC CVEs

In Figure 6, on the left side, the mitigations are listed; on the right side, the associated CVEs are listed. The lines linking mitigations with the CVEs signify a specification of the object property "canMitigate." For example the line between "use_VPN" and CVE-2020-15783 can be read as "use_VPN *canMitigate* CVE-2020-15783." Various insights can be drawn from Figure 6. For example, one can see that the following mitigations map to a considerable number of CVEs: use_VPN, minimize_network_exposure, isolate_remote_devices, apply_cell_protection_concepts, and apply_defense_in_depth_concepts. The remaining mitigations only map to either one, two or three CVEs: restrict_port_access, update_version, deactivate_web_server, and change_default_SNMP_credential). Therefore, based on Figure 6, ICS cyber defenders can reason that using a mitigation from the top left will mitigate a considerable number of CVEs for the S7-300 PLC. This will help optimize the mitigation selection process.

Although ontology visualizations can allow for the drawing of insights, query results can provide empirical results to support the insights that are drawn. The next SPARQL query we developed, focused on figuring out which mitigations could be used across multiple S7-300 PLCs as well as the associated NIST 800.53 standardized controls. Also, each NIST 800.53 control falls within a control family or a general category of controls. Seeing the

control families can help cyber defenders determine what types of mitigation steps need to be taken. The syntax of this SPARQL query includes the following and the results are shown in Table 1:

```
SELECT ?Mitigation (COUNT(?Mitigation) as ?MitigationCount)
WHERE  { ?Mitigation ics:canMitigate ?Vulnerability.
?Mitigation ics:mapsToNISTControl ?Control.
?Control ics:mapsToControlFamily ?ControlFamily .)
GROUPBY ?Mitigation ORDERBY DESC(?MitigationCount)
```

Table 1: SPARQL Query Results of Mitigations Across Vulnerabilities

Mitigations for S7-300 PLCs	Number of S7-300 PLC CVEs Mitigated	NIST 800.53 Rev. 5.1 Controls	NIST SP 800-53 Rev 5.1 – Control Families
Use VPN	16	AC-17 (Remote Access)	Access Control
Minimize network exposure	13	SC-7(2) (Boundy Protection – Access Points)	Boundary Protection
Isolate remote devices	13	SC-7(21)(Boundary Protection – Isolation of System Components)	Boundary Protection
Apply cell protection concepts	9	SC-1 (System Policy and Procedures)	System and Communications Protection
Apply defense in depth_concepts	9	PL-8(1) (Security and Privacy Architectures – Defense in Depth)	Planning
Restrict port access	3	AC-3(5) (Access Enforcement – Security Relevant Information)	Access Control
Update version	3	SI-7 (Software, Firmware, and Information Integrity)	System and Information Integrity
Deactivate web server	2	SC-7(19) (Boundary Protection – Block Communication)	System and Communication Protection
Change default SNMP credentials	1	IA-5(5)(Authenticator Management – Change Authenticators)	Identification and Authentication

The SPARQL results in Table 1 show that using VPN can mitigate 16 out of 17 vulnerabilities (94% of the vulnerabilities) associated with the S7-300 PLC. The visual in Figure 5, allows for user interaction. For example, upon clicking on CVE-2020-15971 (the one CVE that "use_VPN" does not mitigate), two mitigations are available: "apply_cell_protection_concepts" and "apply_defense_in_depth_concepts." Selecting one of these mitigations along with "use_VPN" will mitigate 17 out of 17 vulnerabilities for the S7-300 PLC.

Additional insights can be drawn from Table 1. For example, we can see that, overall, "Access Control" and "Boundary Protection" mitigations can mitigate a considerable amount of S7-300 PLC vulnerabilities. Based on the real-world ICS cybersecurity experience of multiple authors of this paper, these two types of the controls are heavily used in the military ICS mitigations domain especially on in-service Navy ICSs. In other words, these types of mitigations can be used on operational ICSs. From Table 1, we also see that planning and architecture mitigations, "System Policy and Procedures" and "Planning", can mitigate 9 out of 16 CVEs. The specific mitigations that fall under these two general categories include: "apply_cell_protection_concepts" and apply_defense_in_depth_concepts." Based on the authors' ICS cybersecurity experience, policies and planning also play a key role in ICS cybersecurity especially when planning the design of ICSs. However, these controls may not be used immediately in a live ICS environment. Therefore, an ICS defender working in a live environment may select other controls to immediately address S7-300 PLC vulnerabilities.

Vulnerability management, in terms of mitigations selection, is complex due to sources not directly mapped to each other. More work is needed in this exploratory research to find ways to help reduce the complexity of mitigation selection especially when examining multiple ICS devices.

5. Conclusion and Future Work

Our exploratory research shows that ontology modelling can be used to link asset, vulnerability and mitigation sources as well as shows that SPARQL queries can be run to aid in vulnerability management. SPARQL queries focused on vulnerabilities and their mitigations demonstrate that a list can be generated for specific mitigations that can be used across the most vulnerabilities. These results can aid ICS cyber defenders in their decision making, since ICS defenders work in high-stakes environments that require optimal use of resources and time. This work can be expanded in the future with more automation of data ingestion into the ontology. For example, in the current research project, ontology modelling was completed manually. Future research will focus on automating information extraction from the NVD to pull in more ICS vulnerabilities along with vulnerability metadata. Future research will also focus on using natural language processing (NLP) to read ICS CERT advisory sentences and paragraphs in order to map those sentences and paragraphs to our ICS Mitigations Vocabulary as well as to NIST 800.53 (revision 5.1) controls. We also will use testing and evaluation techniques including a task evaluation to assess the correctness of results and evaluation of the structure of the ontology by domain experts (Brank, Grobelnik,and Mladenić, 2005).

References

Arnaert, M, Bertrand, Y, & Boudaoud, K 2016, 'Modeling vulnerable Internet of Things on SHODAN and CENSYS: An ontology for cyber security', *Tenth International Conference on Emerging Security Information, Systems and Technologies,* pp. 299-302.

Blinowski, GJ & Piotrowski, P 2020, 'CVE based classification of vulnerable IoT systems', *International Conference on Dependability and Complex Systems, 2020,* pp. 82-93.

Brank, J, Grobelnik, M & Mladenić D 2005, 'A survey of ontology evaluation techniques', *Conference on Data Mining and Data Warehouses*, *Ljubljana,Slovenia, 2005*.

Bristow, M 2020, *ICS asset identification: it's more than just security*. SANS Institute.

Cybersecurity and Infrastructure Security Agency 2020, *Critical infrastructure sectors*, viewed 12 October 2021, <https://www.cisa.gov/critical-infrastructure-sectors>.

Cybersecurity and Infrastructure Security Agency, 2021, *ICS-CERT advisories*, viewed 12 October 2021, <https://us-cert.cisa.gov/ics/advisories>.

D'Amico, A, Whitley, K, Tesone, D, O'Brien, B & Roth, E 2005, 'Achieving cyber defense situational awareness: a cognitive task analysis of information assurance analysts', *The Human Factors and Ergonomics Society Annual Meeting,* 2005, pp. 229-233.

Elahi, G, Yu, E & Zannone, N 2009, 'A modeling ontology for integrating vulnerabilities into security requirements conceptual foundations', *International Conference on Conceptual Modeling, Gramado, Brazil 2009,* pp. 99–114.

Fenz, S & Ekelhart A 2009, 'Formalizing information security knowledge', *The 4th International Symposium on Information, Computer, and Communications Security, Sydney, NSW, Australia*, *2009*, pp. 183–194.

Forum of Incident Response and Security Teams, 2021, *Common Vulnerability Scoring System Version 3.1: Specification Document,* viewed 12 October 2021, <https://www.first.org/cvss/specification-document>.

Ghaleb, A, Zhioua, A & Almulhem, A 2018, 'On PLC network security," *International Journal of Critical Infrastructure Protection*, vol. 22, pp. 62-69.

Guarino, N 1998, 'Formal ontology in information systems', *The International Conference on Formal Ontology in Information Systems, Trento, Italy, 1998,* pp. 3–15.

Joh, H & Malaiya Y 2010, 'A framework for software security risk evaluation using the vulnerability lifecycle and CVSS metrics', *International Workshop on Risk and Trust in Extended Enterprises, San Jose, CA, 2010*, pp. 430–434.

Knapp, ED, & Langill, JT 2015, *Industrial network security: securing critical infrastructure networks for smart grid, SCADA, and other industrial control systems*, Syngress, Walthham, MA.

MITRE 2013, Common Platform Enumeration, viewed 16 October 2021, < https://cpe.mitre.org/specification/>.

National Institute of Standards and Technology 2004, *Standards for security categorization of federal information and information systems.*

National Institute of Standards and Technology 2018, *Risk Management Framework for information systems and organizations*. Revision 2.

National Institute of Standards and Technology 2020, *SP 800-53 Release Search*, viewed 15 November 2021, <https://csrc.nist.gov/Projects/risk-management/sp800-53-controls/release-search#!/>.

National Institute of Standards and Technology 2021, *Vulnerabilities*, viewed 12 October 2021, <https://nvd.nist.gov/vuln>.

Salini, P & Shenbagam J 2015, 'Prediction and classification of web application attacks using vulnerability ontology' *International Journal of Computer Applications*, vol. 116, no. 21, pp. 42–47.

Sugumaran, V & Storey, V 2002, 'Ontologies for conceptual modeling: their creation, use, and management', *Data Knowledge Engineering*, vol. 42, no. 3, pp. 251–271.

Syed, R 2020, 'Cybersecurity vulnerability management: A conceptual ontology and cyber intelligence alert system', *Information & Management*, vol *57*, no. 6.

Tyworth, M, Giacobe, NA, Mancuso, V & Dancy, C 2012, 'The distributed nature of cyber situation awareness." *IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), 2012.*

Veitch, CK, Henry, JM, Richardson, BT, & Hart, DH 2013, *Microgrid cyber security reference architecture* (No. SAND2013-5472). Sandia National Lab, Albuquerque, NM.

Vorobiev, A & Bekmamedova N 2010, 'An ontology-driven approach applied to information security', *Journal of Research and Practice in Information Technology*, vol. 42, no. 1, pp. 61–76.

Wang, JA & Guo, M 2009, 'Security data mining in an ontology for vulnerability management', *The International Joint Conference on Bioinformatics,Systems Biology and Intelligent Computing, Washington, DC, 2009,* pp. 597–603.

Wang, JA, Guo, MM & Camargo J 2010, 'An ontological approach to computer system security', *Information Security Journal: A Global Perspective*, vol. 19, no. 2, pp. 61–73.

World Wide Web Consortium 2008, *SPARQL query language for RDF*, viewed 12 October 2021, <https://www.w3.org/TR/rdf-sparql-query/>.

World Wide Web Consortium 2015, *Ontology*, viewed 12 October 2021, <https://www.w3.org/standards/semanticweb/ontology>.

Yen, J, McNeese, M, Mullen, T, Hall, D, Fan, X & Liu, P 2010, 'RPD-based hypothesis reasoning for cyber situation awareness', in S Jajodia (ed), *Cyber Situational Awareness*, Springer, Boston, MA, pp. 39-49.