SIMON: Semantic Inference Model for Security in Cyber Physical Systems using Ontologies

Rohith Yanambaka Venkata, Rohan Maheshwari and Krishna Kavi
Department of Computer Science and Engineering
University of North Texas
Denton, Texas-76207
Email: {rohithyanambakavenkata,rohanmaheshwari}@my.unt.edu and krishna.kavi@unt.edu

***Abstract*—Cyber Physical Systems (CPS) are an integration of computational and physical processes, where embedded cyber systems monitor and control physical processes. Cyber attacks largely target components in the cyber domain with the intention of disrupting the functionality of the components in the physical domain. In this paper, we present SIMON, an Ontological design and verification framework that captures the intricate relationship(s) between cyber and physical components in CPS by leveraging standard specification Ontologies and extending the NIST CPS framework. We demonstrate the capabilities of SIMON using two vehicle to infrastructure (V2I) safety applications. In addition, we also investigate introducing resiliency measures that will ensure compliance of physical systems with their design specifications.**

Keywords–CPS Security; Ontology; CPS Privacy, CPS Resiliency.

I. INTRODUCTION

CPS systems can be considered as electronic or computer systems that control physical systems. These systems use sensors to collect information about the physical system and possibly other situational inputs, process these inputs to determine appropriate decisions and affect these decisions on the physical system via actuators. The data collection and transmission of actions may involve the use of communication networks. Thus, CPS systems contain sensors, actuators, electronic/processing components and communication networks, explosing CPS systems to cyber attacks. These cyber attacks will likely impact the physical operation of the system and may also impact the physical world these systems reside in. Thus, it is essential to understand the inter-relationships between the functions of the physical systems and the cyber (or electronic) systems and how an attack on one affects the other.

We advocate the use of Ontologies to model CPS systems and the relationships between their constituent subsystems. An Ontology is a formal description of knowledge as a set of concepts within a domain and the relationships that hold between them [1]. To enable such a description, we need to formally specify components such as individuals (instances of objects), classes, attributes, and relations as well as restrictions, rules, and axioms. Ontologies not only enable a shareable and reusable knowledge representation but, can also add new knowledge about a domain [1]. Our approach extends NIST CPS framework [2] by differentiating between an abstract realization and a concrete realization levels. The abstract level translates the conceptual requirements of CPS systems (such as functional, timing, trustworthiness requirements) into responsibilities and roles of system components (such as sensors, actuators, processing elements, communication systems, computational algorithms). The concrete realization defines specific products used to implement the abstract responsibilities and functionalities (such as selecting a specific IoT system, or a communication device). Our Ontologies allow for common vocabularies to describe concepts and properties of CPS systems at various levels of the design framework. This permits for adapting best design practices of one domain to the design of systems in another domain.

In this paper, we present our preliminary work in vulnerability assessment and design validation of CPS systems. Our prior work on using Ontologies in vulnerability assessment in cloud systems [3] [4] enables us to extend those Ontologies to address security concerns in CPS systems. Using the NIST CPS framework as a basis for SIMON allows for a broad and integrated view of CPS and positions trustworthiness among other aspects of CPS design. Furthermore, using standard Ontologies like SOSA will help streamline the process of secure CPS design by considering the properties of a CPS system like sensing and actuation.

The rest of the paper is organized as follows. Section III describes SIMON, our proposed CPS framework. This section also describes the various standard Ontologies, as well as some of our new Ontologies used in our framework. Section IV includes two case studies to show how SIMON can be used for the design and validation of CPS systems. We show some examples of cyber attacks and use reasoners to identify potential compromise of design goals associated with the physical system.

II. RELATED WORK

Extensive research has been done in applying Ontologies to either identify or validate the security posture of CPS or IoT systems. Mozzaquatro et al. [5] proposed a framework that employs a model-driven approach to designing secure CPS systems. While this may be prudential in some domains, it fails to account for concerns from various stakeholders in a CPS system. This is addressed by the NIST CPS framework.

Fenz et al. [6] and Settas et al. [7] proposed Ontological frameworks that are complemented by Bayesian networks to predict threat probabilities in cloud systems. The key competencies of these contributions is vulnerability assessment and threat modeling for cyber systems in the cloud.

SIMON aims to bridge the gap between design validation using cyber threat data from multiple sources. We believe that this approach will help in the design of secure CPS systems.

III. THE FRAMEWORK

The proposed framework combines (and extends) existing standard specification Ontologies such as Semantic Sensor Networks (SSN), and new ones as required by the domain of interest. Let us take a closer look at some of the Ontologies and frameworks used in our research.

 ISBN: 978-1-61208-752-8

A. NIST CPS Framework

National Institute of Standards and Technology (NIST) has developed a framework that provides guidance in designing, building and verifying complex CPS systems [2]. The framework captures generic functionalities that CPS provide, the activities and artifacts needed to support conceptualization, realization and assurance of CPS design [2]. Designing a CPS system involves:

- **Conceptualization -** Capturing all activities related to high-level goals, functional requirements and organization of CPS as they pertain to what the CPS is supposed to do. It provides a conceptual model of the CPS system under consideration.
- **Realization -** Capturing all activities surrounding the detailed engineering, design, production, implementation and operation of the desired systems. However, to facilitate comparing Ontological models of CPS systems, we propose bifurcating the overarching realization phase described in the NIST CPS framework into the following sub-phases.
  - **Abstract Realization -** In this phase, design goals are broken down into roles and responsibilities and delegated to subsystems and interfaces. For example, we may identify that the network communications needed in the system will be handled by a wireless data communication application but not provide details on either the specific hardware device or communication protocols. We use Ontologies to capture the Abstract Realization.
  - **Concrete Realization -** The roles and responsibilities identified during the abstract realization phase need to be implemented by specific products. For example, a Cisco ASR1002-10G-HA/K9 is selected as the wireless data communication application identified in the Abstract Realization phase. We use Ontologies to relate the products used for various functions and roles identified in the Abstract Realization.
- **Assurance -** The assurance phase deals with obtaining confidence that the system built in the realization phase satisfies the model developed in the conceptualization phase [2]. In our case, we use reasoners to infer and derive assurances (or violations) of the goals and functional requirements are met. We use additional Ontologies to capture cyber threat data so that vulnerabilities, cyber attacks and possible mitigations can be related to the products identified in Concrete Realization; we rely on NIST Common Platform Enumeration (CPE) identities with specific products for this purpose.

SIMON can be used to modify the CPS design at any of the various phases to address any design violations discovered by our reasoners.

Figure 1 describes an abstract view of our framework for the design and verification of CPS systems, focusing on security and trustworthiness. We use different Ontologies in our framework to describe the concepts, properties and restriction associated with CPS systems at each of the design phases described in the previous section.

B. Sensor-Observation-Sampling-Actuator Ontology (SOSA)

The Sensor-Observation-Sampling-Actuation Ontology (SOSA), a subset of the Semantic Sensor Network (SSN) Ontology presents a conceptualization of all entities, activities and properties that typically constitute a CPS. SOSA is a World Wide Web Consortium (W3C) standard specification.

The *core structure* of SOSA Ontology encompasses all of the three modeling perspectives; the activities of observing, sampling, and actuating [8]. Each activity targets a feature of interest by either changing its state or revealing its properties by following a designated procedure. All activities are carried out by an object, also called an agent.

SOSA aims to strike a balance between the expressivity of the underlying description logic, the ease of use of language features and the expectations of the target audience, while accommodating a broad range of domains and applications [8].

C. Cyber Threat Information Ontology

The activities of observing and sampling must be followed by communicating the data and processing to interpret the observations and making decisions on the actions. These actions are then used to control physical systems through actuation. The communication and processing subsystem, which is not directly included in the SOSA ontology can expose the cyber and physical components of the CPS to security attacks. Thus, SOSA must be extended to describe the processing and communication subsystems. This allows us to relate cyber threat data from multiple sources to obtain insights into the security posture of a CPS system under consideration. We have defined an Ontology that captures Cyber Threat Information (CTI) from three sources:

- **The National Vulnerability Database (NVD) -** A U.S. government repository of standards based vulnerability management data [9].
- **Exploit Database -** An archive of public exploits and corresponding vulnerable software, developed for use by penetration testers and vulnerability researchers [10].
- **Metasploit -** A framework for developing, testing and executing software exploits [11].

Our Ontology can easily be extended to capture CTI from ohter sources. The cyber threat Ontology is underpinned by the STIX structured language, that enables organizations to share, store and analyze CTI in a consistent manner, allowing security communities to better understand what computer-based attacks they are most likely to see and to anticipate and/or respond to those attacks more effectively [12]. The STIX Ontology utilizes twelve core concepts: Attack pattern, Campaign, Course of Action, Identity, Indicator, Intrusion Set, Malware, Observed Data, Report, Threat Actor, Tool and Vulnerability.

Attack Pattern describes ways that threat actors attempt to compromise targets and *Campaign* categorizes malicious activities that occur over a period of time by identifying their intended targets. *Vulnerability* describes a flaw in software (or hardware) that can be exploited by a *Threat Actor* to breach a target.

Our objective in defining the CTI Ontology is to unify information from three sources (described earlier in this section) and facilitate logical reasoning about the security of CPS using *Axioms*. Axioms are rules that are used by a reasoner to infer additional information that may be hard to define using a knowledge representation language. To provide a perspective

 ISBN: 978-1-61208-752-8

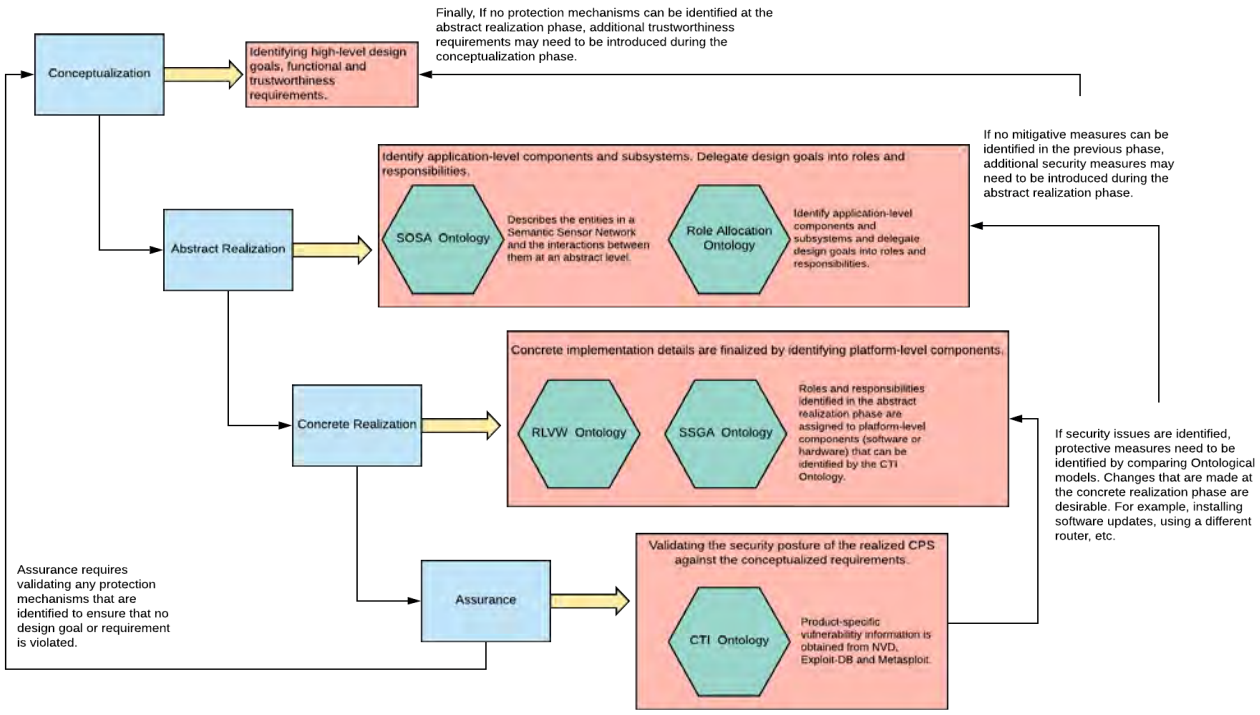

Figure 1. The SIMON Ontological Framework.

of the complexity of CTI Ontology, it includes 6657 axioms that describe CTI data. In addition to STIX, the CTI Ontology also inherits characteristics from two additional Ontologies:

- **Cyber Observable Expression (CybOX)** - A standardized language for encoding and communicating information about cyber observables [12]. Using CybOX language, relevant observable events or properties pertaining to an attack pattern can be captured.
- **Common Attack Pattern and Enumeration (CAPEC)** - Provides a dictionary of known patterns of attack employed by adversaries to exploit known weaknesses in cyber-enabled capabilities.[13].

Here is a brief look at some of the important characteristics of our CTI Ontology:

- **Attack:** This feature is mapped to the *Indicator*, *Observed Data* classes in the STIX Ontology and the *Observation, FeatureOfInterest and ObservableProperty* classes in the STIX Ontology. This characterizes a cyber attack by identifying a pattern, set of adversarial behaviors or information observed on a system in the network.
- **Exploit:** Mapped to the *Vulnerability* and *Intrusion set* classes in the STIX Ontology and the *Sensor, Actuator* and *Sample* classes in the SOSA Ontology, the Exploit feature enumerates a flaw in a platform (Software or Hardware with a CPE entry in the NVD) that can be leveraged by an adversary to compromise a CPS system.
- **Ramification:** Incident response teams often desire to know the consequences/objectives of potential adversaries to prioritize responses to cyber attacks. In a similar vein, threat modeling at the design phase of a CPS system will equip CPS designers to understand the outcome of cyber attacks and design more secure or resilient systems. At present, threat classification is based on the Spoofing, Tampering, Repudiation, Information disclosure, Denial of Service and Elevation of Privilege (STRIDE) classification model [14], where each type of threat is assigned its own class. The *Ramification* feature maps to a class in the STRIDE based on the nature of the threat. In addition, it also maps to the *ThreatActor, CourseOfAction* and *Vulnerability* classes in the STIX Ontology and the *Actuation, Observation, Procedure, FeatureOfInterest, Platform* and *ObservableProperty* classes in the SOSA Ontology.

Thus, our framework allows users to identify and enumerate cyber threats that affect a CPS system of interest. We rely on Ontologies because of the following benefits they offer:

- **Knowledge Representation:** The primary benefit of using an Ontology is it's ability to define a semantic model of data, within the context of an associated domain knowledge and this can be leveraged to achieve knowledge sharing and more importantly, knowledge reuse, which is discussed in the next section.
- **Modularity:** Our framework facilitates modularity by allowing CPS designers to use domain-specific properties (Ontologies like SOSA). Users have the option of using additional vocabulary, in addition to the W3C specification to model proprietary systems.
- **Extensibility:** CPS systems are constantly evolving. Advances in networking and embedded system technologies like system-on-chip (SoC) and wireless transceivers result in the emergence of new CPS applications. The structure of SIMON, coupled with its modular design supports integrating or modifying CPS characteristics, and

 ISBN: 978-1-61208-752-8

to reason about the security posture of a system.

IV. Vehicle To Infrastructure (V2I) Wireless Data Interface Ontology: A Case study

As a case study to show the use of our framework, we use the Red Light Violation Warning (RLVW) safety application as described in the US Department of Transportation document [15]. The Red Light Violation Warning (RLVW) application enables a connected vehicle approaching an instrumented signalized intersection to receive information from the infrastructure regarding the signal timing and the geometry of the intersection. The application in the vehicle uses its speed and acceleration profile, along with the signal timing and geometry information to determine if it appears likely that the vehicle will enter the intersection in violation of a traffic signal. If the violation seems likely to occur, a warning can be provided to the driver.

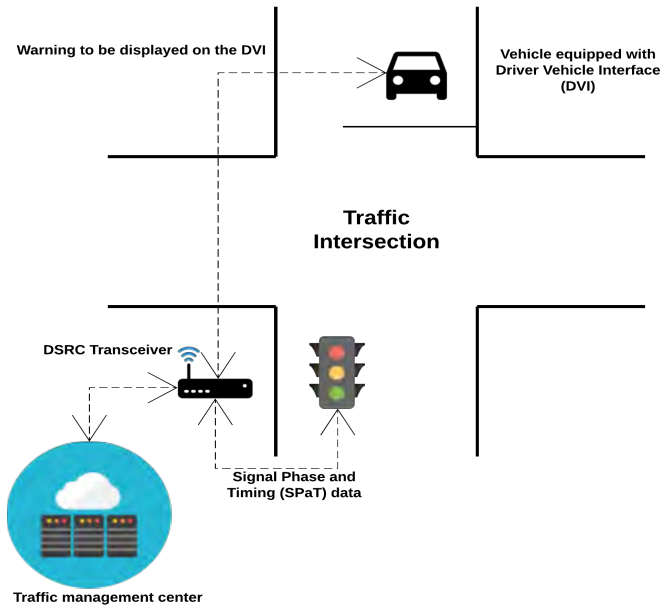

Figure 2. The RLVW system

Figure 2 depicts the RLVW system. To identify the most vulnerable areas in this system, it is vital to understand the flow and origin of data (i.e., sensing and observation aspects of the system). We have developed an Ontology for the Vehicle To Infrastructure or Infrastructure To Vehicle (V2I/I2V) Wireless Data Interface through which all vehicular and infrastructure data is exchanged. The ontology highlights the cyber and physical components comprising the wireless data interface portion of the V2I system and distinguishes between the physical components that produce the data and the cyber components that transmit the data. The flow of data in the ontology has revealed that the Infrastructure Wireless Data System (IWDS) and the Vehicle Wireless Data System (VWDS) which are connected through the V2I Wireless Data Interface are the most vulnerable regions of the entire V2I CPS because data flows on a completely open network when traversing through these cyber components. With the source and destination IP addresses of data packets unprotected, this can lead to numerous threats from any third party with a V2X communication handler. Before classifying the immediate threats that can occur with data flowing on this presumed 5G open network, the Ontology also describes the various types and origins of data to understand the impact of a cyber attack.

Accurately modeling a CPS system is vital in identifying and mitigating security issues. The Ontological framework described in Section III helps achieving this because it offers the following features.

- **CPS Framework:** Perhaps the most important characteristic that enables comparison between Ontologies is sharing a common underlying framework that ensures similarities in structure. The NIST CPS framework is an ideal candidate because it addresses cross-cutting concerns that are crucial to identifying design flaws or vulnerabilities that could be introduced due to interaction between cyber components. Ontologies needs to be modeled using the CPS framework to be compared.
- **Cyber Threat Ontology:** Using an Ontology that is optimized for identifying, obtaining and organizing cyber threat data for CPS systems is invaluable in identifying potential mitigation measures that will ensure compliance with design goals of a CPS system.
- **Domain-specific Properties:** Identifying and expressing domain-specific properties is imperative in accurately modeling CPS systems. This helps correctly identify aspects (such as Functional, Human, Timing, etc.) and concerns (such as Physical security, Predictability, Dissociability, etc.). These properties contribute to identifying design flaws/vulnerabilities that are unique to the CPS system under consideration. For example, the SOSA Ontology used in this framework helps identify design goals such as latency and timing requirements that are unique to sensor networks. This helps identify pertinent mitigation measures that ensure compliance with the design goals. If no such measure(s) can be found, a change in the design of a CPS system may be required.

Sharing a common underlying framework enables knowledge-reuse by providing a shared conceptualization of a domain of interest. Therefore, it stands to reason that Ontologies describing similar CPS systems, sharing the same semantic structure can be compared to investigate protection mechanisms that could protect against security threats. With this in mind, let us consider a CPS model that was developed using the framework described in Section III.

A. Infrastructure Data Types and Significance

Starting with the Infrastructure, its physical components consists of the signalized intersection sensor systems that capture two main types of data [15].

1) SPaT: SPaT data (Signal Phase and Timing) contains information about the behavior of the traffic controllers regarding the state of the signal (viz., red, green or yellow), how long that state will remain, and time until next phase change.

2) Driving Conditions: The physical component of the infrastructure also produces data that characterizes the environmental conditions approaching vehicles may face. This data consists of weather data, visibility data and road conditions for the vehicle to incorporate in its decision making computations to improve precision in judgement as approaching the intersection.

 ISBN: 978-1-61208-752-8

B. Vehicle Data Types and Significance

The vehicle's physical components consists of the position and stability systems, actuators, and telematic sensors that transmit Differential GPS (DGPS) and Dynamic Telematic Data (DTD) [15].

1) Differential GPS: DGPS data contains map data of the vehicle's position relative to the approaching signalized intersection. The vehicle data systems transmit DGPS to the infrastructure in order to alert the traffic controllers of the instantaneous distance the vehicle is from the intersection.

2) Dynamic Telematic Data: DTD consists of information regarding the vehicle's speed, position and reveals how the vehicle is behaving internally. This data is combined with DGPS, and incoming SPaT data for the vehicles to calculate using DVI equations and algorithms in order to make precise judgement of whether the driver should increase or decrease speed to avoid traffic violations and or accidents at the intersection.

C. Stop Sign Gap Assist (SSGA)

As a reference for investigating the reuse of protection mechanisms that are currently in place, we use the Stop Sign Gap Assist (SSGA) system in the Department of Transportation V2I specifications [15]. The SSGA Infrastructure Application component delivers roadside advisory, alert, and warning messages to the driver, based upon infrastructure-based sensor systems placed on the major roadway that detect the speed and location of approaching remote vehicles. It is intended to improve safety at non-signalized intersections where only the minor road has posted stop signs [16]. This application includes both on-board (for connected vehicles) and roadside signage warning systems (for non-equipped vehicles) [16]. The application will help drivers on a minor road stopped at an intersection understand the state of activities associated with that intersection by providing a warning of unsafe gaps on the major road. The SSGA application collects all available sensor information (major road, minor road, and median sensors) data and computes the dynamic state of the intersection in order to issue appropriate warnings and alerts [16].

Intuitively, it is easy to recognize the similarities in the design goals of the RLVW and SSGA applications, the distinction being that signalized intersections are replaced by a stop sign in the SSGA system.

The CPS framework ensures that concepts of two Ontologies being compared are aligned. Comparing the relationship of each aligned concept with its neighbors in the Ontologies being compared yields the differences in interpretation. The abstract realization phase of the framework deals with identifying, defining and delegating design goals identified in the conceptualization phase into roles and responsibilities for system components and interfaces at an abstract level. This provides a good basis to determine if the conceptualized concepts and their relationships are aligned.

D. Identifying Security Threats and Protection Mechanisms

In this section, let us consider a few vulnerabilities in the RLVW system that can be addressed by reusing mitigation measures employed in a distinctly different CPS system, viz., the SSGA application, by comparing their Ontological models. Now that the baseline for the V2I WDI region is set, we can analyze the proposed ontology to classify potential threats in the flow of data.

1) V2X Remote DSRC Interjection Threat: The IWDS and VWDS communicate through the V2I WDI over a bidirectional DSRC network [15]. While DSRC provides a robust and low latency connection for short distance communication [17], its security protocol only prevents Distributed Denial of Service (DDoS) attacks from a short distance. Therefore, a third party with V2X communication handlers can interject data transmission remotely through Internet Protocol and Domain Name Service (IP/DNS) Spoofing attacks to reroute outgoing Differential GPS (DGPS) data and Dynamic Telematic Data (DTD) from the vehicle. With this data in their possession, unauthorized V2X handlers can track drivers and read into vehicle logs which creates privacy issues for the victim. The NIST Vulnerability Database highlights a similar issue with the configuration *cpe:2.3:a:cisco:application-policy-infrastructure- controller:8.31s6:*:*:*:*:*:*:** [9]. Existence of this vulnerability suggests that this simple attack is highly probable if correct mitigation is not in place. A potential start for resolving this issue may involve ITS developers implementing a SSL certificate with outgoing data which requires V2X handlers to have a certain cryptographic key in order to access the contents of the data packets [18].

The RLVW and SSGA systems share some design goals. Furthermore, comparing their abstract realization phases reveals that they share the same WDI. This is further evidenced by comparing their concrete realization phases, which reveals that they use the same DSRC transceiver and network communication subsystem. It may be worthwhile to compare the two Ontologies to determine if protection mechanisms employed in the SSGA application can be reused in the RLVW system.

```
The Ontology is consistent
Road side equipment CPE : cpe:2.3:a:cisco:application-policy-infrastructure-controller:8.31s6:*:*:*:*:*:*:*
Adversary may leverage CVE-2017-12352 to gian elevated privileges
Adversary may tamper with the RLVW warning data that is broadcast to vehicles
(Warning) Potential violation of functional requirement 1.1.5 of the Road side equipment
(Inferred) Potential violation of functional requirement 1.2.4.7 of the RLVW system
(Inferred) Potential violation of functional requirement 1.2.5.2 of the RLVW system
(Inferred) Potential violation of functional requirement 1.1 of the Driver Interface system
```

Figure 3. RLVW Inference.

The CTI Ontology obtains vulnerability information for components identified in the concrete realization phase using NIST CPE (Common Platform Enumeration) identifications. In this example, let us consider one vulnerability that can be exploited for a privilege escalation with NIST Common Vulnerability Enumeration(CVE) identification, *CVE 2017-12352*, associated with the CISCO router with *cpe:2.3:a:cisco:application-policy-infrastructure-controller:8.31s6:*:*:*:*:*:*:** [9]. An adversary can exploit this vulnerability in certain system script files on Cisco Application Policy Infrastructure Controllers to gain elevated privileges and execute arbitrary commands with root privileges on an affected host operating system [19]. The vulnerability is due to insufficient validation of user-controlled input that is supplied to script files of an affected system [19]. A simple fix would be to install a software update for the application policy infrastructure controller. However, to demonstrate the capabilities of Ontological modeling and reasoning, we will assume that no software patches are available for this component.

Figure 3 shows how the CTI Ontology uses semantic reasoning to link vulnerabilities to the design goals identified during the conceptualization phase. While an elevation of privilege attack can lead to catastrophic failure of the affected system, we will focus on adversaries potentially spoofing their

 ISBN: 978-1-61208-752-8

identities in this example.

The SSGA system uses Extensible Authentication Protocol (EAP), a certificate-based authentication scheme to validate the V2X handler that issues requests for DGPS and DTD data. This prevents most spoofing attacks.

```
The Ontology is consistent
Distinction identified between SSGA and RLVW VWDS
(Asserted) SSGA uses wireless message authentication scheme
(Inferred) EAP introduces latency
(Inferred) Potential violation of requirement 1.3.1 of the Driver Interface System
(Inferred) Potential violation of requirement 1.5.2.2 of the RLVE system
```

Figure 4. Comparing the Ontologies

Figure 4 illustrates how the message authentication scheme used in SSGA is capable of preventing the spoofing attack identified by the CTI Ontology. However, this scheme introduces latency, which may impact the timing requirement listed in the conceptualization phase of RLVW. Let us investigate if message authentication scheme is a viable solution for RLVW.

```
(Asserted) Timing Requirements 1.3.4.1 needs to be met
(Inferred) RLVW zone needs to be extended to 100 meters
(Asserted) DSRC radio has a maximum range of 120 meters
(Inferred) Additional requirements need to be added at abstract realization
```

Figure 5. Testing compliance

As evidenced from Figure 5, the Ontology determines that the RLVW requirement to warn drivers well in advance of a red light violation, to provide ample stopping distance may be violated by the latency that is introduced by the authentication scheme. Furthermore, the Ontology also infers that the components used in this system are capable of supporting the timing requirement as the DSRC transceiver has a range of 120 meters. To address this, the Ontology recommends that the warning zone be increased from 80 meters before the intersection to 100 meters, which should provide ample time for EAP to authenticate the communication. A requirement needs to be added in the abstract realization phase to include an authentication scheme that also includes fail-safe measure if authentication is inconclusive. A domain expert needs to be consulted to ensure that all design goals are accurately captured in the SIMON framework.

2) V2X Handler Elevation of Priviledge Threat: Unfortunately, DSRC communication between V2I WDI and VWDS is not the only insecurity of the WDI region. The performance requirements set by the DoT do not mention any form of security over the functionality of the IWDS and VWDS [15]. In this section, we investigate the possibility of improving the resiliency of a CPS system against privilege escalation attacks by implementing a fail-safe mechanism. The proposed ontology outlines the path of data through the Infrastructure Application component (IAC) and platform (IAP) that reveals no form of encryption on data produced by the physical components or verification when that data is transmitted through the cyber components. Therefore, V2X Handlers with identical communication functionality and IP address can replace the role of the IWDS in the TCP handshake and give false acknowledgement to the IAP. V2X Handlers can then tamper with outbound SPaT and road data which results in the vehicle application component producing false metrics. These metrics may result in a red light traffic violation or even roadside accidents. A similar vulnerability issue is noted with the configuration *cpe:2.3:o:cisco:ios-xe:16.10.1:*:*:*:*:*:*:*:** in the NIST Vulnerability Database [9], thus, indicating the possibility of this threat occurring roadside. A general solution to this vulnerability can involve ITS developers implementing an ingress filtering protocol that requires the VWDS to check incoming data packets for their source headers to ensure it matches the one of the origin and to reject the packet if it does not [18].

The SSGA application uses Public Key Infrastructure (PKI) encryption for communication between the components. This requires a Certifying Agency (CA) to generate and assign a public key to each component in the system. The CA is maintained by the DoT. The messages are authenticated using Message Authentication Code (MAC). PKI is a comprehensive security and authentication scheme requiring all entities to ensure confidentiality, integrity, non-repudiation and end-to-end monitoring and key life cycle management.

The CTI identifies the configuration of the V2X handler and maps it to *cpe:2.3:o:cisco:ios-xe:16.10.1:*:*:*:*:*:*:*:**. It is able to identify vulnerability *CVE 2019-1756* that can be leveraged by adversaries to launch an elevation of privilege attack to breach the communication channel between the IAC and IAP. A vulnerability in Cisco IOS XE Software could allow an authenticated, remote attacker to execute commands on the underlying Linux shell of an affected device with root privileges [20]. The vulnerability occurs because the affected software improperly sanitizes user-supplied input. An attacker who has valid administrator access to an affected device could exploit this vulnerability by supplying a username with a malicious payload in the web UI and subsequently making a request to a specific endpoint in the web UI. A successful exploit could allow the attacker to run arbitrary commands as the root user, allowing complete compromise of the system [20].

```
The Ontology is consistent
Road side equipment CPE : cpe:2.3:o:cisco:ios-xe:16.10.1:*:*:*:*:*:*:*:*
Adversary may leverage CVE-2019-1756 to gian elevated privileges by code injection
Adversary may tamper with SPaT data
(Asserted) Potential violation of functional requirement 1.1.5 of the Road side equipment
(Inferred) Potential violation of functional requirement 1.2.4.7 of the RLVW system
(Inferred) Potential violation of functional requirement 1.2.5.2 of the RLVW system
(Inferred) Potential violation of functional requirement 1.1 of the Driver Interface system
(Inferred) Potential violation of functional requirement 1.1.6 of the RLVW system
```

Figure 6. Elevation of Privilege Threat Inference

The potential impact of this vulnerability being exploited is shown in Figure 6. The framework is able to infer that the primary design goals of the RLVW application and the roadside equipment may be violated as a direct result of this vulnerability.

As discussed in the previous example, SSGA uses message authentication and EAP. The same measures can be used in this example to protect the RLVW system. However, we are interested in identifying possible resiliency measures that can be employed by the RLVW system to protect against privilege escalation attack. To identify activities that can be used in the vehicle to detect spurious data from the infrastructure, let us consider an autonomous vehicle that is capable of perceiving the world around it.

We have defined a simple Ontology that models approximately 3118 attributes of an autonomous vehicle that includes driving actions like stop and go, a collision warning system, a lane change detection system and so on The insights provided

 ISBN: 978-1-61208-752-8

by this Ontology can be used to prevent attacks like those discussed above by introducing resiliency into the design of the CPS system. The inference engine compares the RLVW system against three principles of a fully autonomous vehicle.

- **Sensing the world -** It is imperative for autonomous vehicles to possess the ability to perceive the world around them.
- **Conveying intent -** Assuming that other autonomous vehicles are present in the immediate vicinity, conveying intent such as lane change or impending change in driving action to other vehicles (and possibly pedestrians) is required.
- **Situational awareness -** Assigning a context to the information obtained by sensing the world is essential in making an informed decision. Comprehending events in the environment with respect to time and space is crucial.

```
1. Ensure vehicle meets requirement for sensing the world (1.2.1.1)
                    - Cameras in the front windshield to detect traffic lights (Refer to requirement 1.3.3.2)
```

Figure 7. Measure to introduce resiliency into the RLVW system

The Ontology limits the inference to the design principle of sensing the world for the RLVW system as the other principles do not apply to it. Applying all three principles will negate the role of the infrastructure elements in this V2I system. To that end, the insights provided by the Ontology are shown in Figure 7.

While this is only a preliminary design of a specific region of the V2I CPS, the potential of an Ontology-based model is shown through the vulnerabilities it can classify. By describing various components through their roles, data types, and functionality, the Ontology can reason about new threats or vulnerabilities upon the addition of an unknown component to the system. If the properties of the unknown component, which in this case study is a V2X handler, become known, the ontology can use reasoners to infer where this new component may interject by comparing properties of the new component with existing components in the CPS. When a match is found, the ontology will classify the new component in a certain instance of the CPS. This knowledge can be used to implement new levels of security and mitigation in existing components to make it difficult for V2X handlers to either interject the CPS, or play the role of a component in the CPS [21].

V. Conclusion and Future Work

In this paper, we have presented an argument for modeling CPS using Ontologies. We also presented SIMON, a framework that is based on the NIST CPS framework, but extends it in several ways. We use Ontologies during each design phase of the framework to check for compliance and provide recommendations by reusing knowledge. Increased traction in CPS adoption, their growing complexity, and heterogeneous nature necessitates accuracy in capturing the relationship between various components in a CPS. Reasoning about a CPS realization and validating that the realization does not violate functional as well trustworthiness goals is essential in improving the security posture of a CPS system. The SIMON framework can aid in this process. We have only described the framework at a very high level and we plan to integrate various Ontologies and reasoning engines in the near future. Although Ontologies are used extensively for knowledge representation in domains such as healthcare and bioinformatics, we aim to leverage their capabilities to define a domain agnostic framework that can be extended to various CPS domains by attributing domain-specific properties (like SOSA). We are also developing tools for automatically (or semi-automatically) convert CPS designs using NIST framework to SIMON framwork.

Acknowledgement

This research is supported in part by the NSF Net-centric Industry-University Cooperative Research Center at UNT and the industrial members of the Center.

References

[1] "What are ontologies?." URL: https://ontotext.com/knowledgehub/fundamentals/what-are-ontologies/ [accessed: 2019-06-11] .

[2] D. A. Wollman, M. A. Weiss, Y. Li-Baboud, E. R. Griffor, and M. J. Burns, "Framework for cyber-physical systems," *Special Publication (NIST SP) - 1500-203*, 2017.

[3] P. Kamongi, M. Gomathisankaran, and K. Kavi, "Nemesis: Automated architecture for threat modeling and risk assessment for cloud computing," 12 2014.

[4] P. Kamongi, S. Kotikela, K. Kavi, M. Gomathisankaran, and A. Singhal, "Vulcan: Vulnerability assessment framework for cloud computing," in *Proceedings of the 2013 IEEE 7th International Conference on Software Security and Reliability*, SERE '13, (Washington, DC, USA), pp. 218–226, IEEE Computer Society, 2013.

[5] B. Mozzaquatro, C. Agostinho, D. Goncalves, J. Martins, and R. Jardim-Goncalves, "An ontology-based cybersecurity framework for the internet of things," *Sensors*, vol. 18, p. 3053, Sep 2018.

[6] S. Fenz, "An ontology- and bayesian-based approach for determining threat probabilities," in *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, ASIACCS '11, (New York, NY, USA), pp. 344–354, ACM, 2011.

[7] D. Settas, A. Cerone, and S. Fenz, "Enhancing ontology-based antipattern detection using bayesian networks," *Expert Systems with Applications*, vol. 39, no. 10, pp. 9041 – 9053, 2012.

[8] K. Janowicz, A. Haller, S. J. D. Cox, D. L. Phuoc, and M. Lefrançois, "SOSA: A lightweight ontology for sensors, observations, samples, and actuators," *CoRR*, vol. abs/1805.09979, 2018.

[9] "National Vulnerability Database." URL: https://nvd.nist.gov/ [accessed: 2019-06-11].

[10] "Exploit-DB." URL: https://www.exploit-db.com [accessed: 2019-06-20].

[11] "Metasploit-penetration testing framework." URL: https://www.metasploit.com/ [accessed: 2019-06-20].

[12] S. Barnum, "Standardizing cyber threat intelligence information with the structured threat information expression (stix)," *MITRE*, 2014.

[13] "Common Attack Pattern Enumeration and Classification (CAPEC)." URL: https://capec.mitre.org/ [accessed: 2019-07-02].

[14] Microsoft Corporation, "The STRIDE threat model." URL: https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20) [accessed: 2019-07-08].

[15] Department of Transportation, "Performance Requirements, Vol. 3, Red Light Violation Warning (RLVW)," *Vehicle-to-Infrastructure (V2I) Safety Applications*, pp. 1–68, 2015.

[16] D. Stephens, J. Schroeder, and R. Klein, "Vehicle-to-infrastructure (v2i) safety applications - stop sign gap assist (ssga)," *FHWA-JPO-16-254*, vol. 7, 2015.

[17] "Dedicated Short Range Communications (DSRC) Service," 2019. URL: https://www.fcc.gov/wireless/bureau-divisions/mobility-division/dedicated-short-range-communications-dsrc-service [accessed: 2019-06-11].

[18] "DDoS Glossary," 2019. URL: https://www.cloudflare.com/learning/ddos/ glossary/ [accessed: 2019-06-11].

 ISBN: 978-1-61208-752-8

[19] Cisco, “Cisco application policy infrastructure controller local command injection and privilege escalation vulnerability,” 2017. URL: = https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-20171129-apic [accessed: 2019-07-22].

[20] Cisco, “Cisco ios xe software command injection vulnerability,” *Cisco security advisory*, 2019. URL: = https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-20190327-iosxe-cmdinject [accessed: 2019-07-22].

[21] R. Y. Venkata and K. Kavi, “An Ontology-Driven Framework for Security and Resiliency in Cyber Physical Systems,” *The Thirteenth International Conference on Software Engineering Advances*, pp. 4–6, 2018.

 ISBN: 978-1-61208-752-8