I've Seen This Before: Sharing Cyber-Physical Incident Knowledge

Faeq Alrimawi
Lero-The Irish Software Research Centre
Limerick, Ireland
faeq.alrimawi@lero.ie

Liliana Pasquale
University College Dublin & Lero
Dublin, Ireland
liliana.pasquale@ucd.ie

Deepak Mehta
United Technologies Research Centre-UTRC
Cork, Ireland
mehtad@utrc.utc.com

Bashar Nuseibeh
The Open University & Lero
UK/Ireland
bashar.nuseibeh@lero.ie

ABSTRACT

An increasing number of security incidents in cyber-physical systems (CPSs) arise from the exploitation of cyber and physical components of such systems. Knowledge about how such incidents arose is rarely captured and used systematically to enhance security and support future incident investigations. In this paper, we propose an approach to represent and share incidents knowledge. Our approach captures *incident patterns* – common aspects of incidents occurring in different CPSs. Our approach then allows incident patterns to be instantiated for different systems to assess if and how such patterns can manifest again. To support our approach, we provide two meta-models that represent, respectively, incident patterns and the cyber-physical systems themselves. The incident meta-model captures the characteristics of incidents, such as assets and activities. The system meta-model captures cyber and physical components and their interactions, which may be exploited during an incident. We demonstrate the feasibility of our approach in the application domain of smart buildings, by tailoring the system meta-model to represent components and interactions in this domain.

CCS CONCEPTS

• **Software and its engineering → Model-driven software engineering**

KEYWORDS

Cyber-Physical Systems, Incident Pattern, Smart Buildings, Meta-model

ACM Reference format:

F. Alrimawi, L. Pasquale, D. Mehta, and B. Nuseibeh. 2018. I've Seen This Before: Sharing Cyber-Physical Incident Knowledge. In *Proceedings of IEEE/ACM 1st International Workshop on Security Awareness from Design to Deployment, Gothenburg, Sweden, May, 2018 (SEAD '18)*, 8 pages.
DOI: 10.1145/3194707.3194714

SEAD'18, May 27, 2018, Gothenburg, Sweden

ACM ISBN 978-1-4503-5727-2/18/05…$15.00
https://doi.org/10.1145/3194707.3194714

1 INTRODUCTION

A Cyber-Physical System (CPS) combines computation, communication, and physical processes to produce systems that are more adaptive, collaborative, and autonomous [1]. Applications of CPS [2] can be found in various domains including industrial control, transportation, and smart buildings. This combination of processes enables interactions between cyber and physical components, in which an event caused by a cyber component can have an impact on physical ones, and vice-versa. For example, in a smart building, a rise in the measured temperature of a room can trigger a digital process to issue a command to an air conditioner to start cooling the room.

Interactions between cyber and physical components are giving more opportunities to malicious individuals to cause harm [3]. For example, in the Ukrainian power grid incident [4], offenders used spear phishing to gain a foothold in the distribution companies computer networks. Then, they gained access to the power grid network, where they infected some devices (e.g., workstations, serial-to-Ethernet) that control electricity distribution with malware. Subsequently, they disabled infected devices. This caused a disruption of the normal operation of the grid. Previously, in the German steel-mill incident [5], offenders used spear phishing to gain a foothold in the corporate network. Then, they gained access to the plant's network, where they infected programmable logic controllers with malware. Subsequently, they caused damage to various components such as the blast furnace and the alarm system. Consequently, the normal operation of the plant was interrupted.

Incidents often have similar characteristics. For example, in the Ukrainian power grid and the German steel-mill incidents, an offender infiltrated into a private network using spear phishing. Although commonalities between these incidents can be observed, these have not been captured and used systematically to enhance security and support incident investigations [6]. Current attack modeling techniques (e.g., attack graphs [7]) focus on representing how a traditional cyber attack (e.g., denial of service) can occur. As these techniques do not account for the interactions between cyber and physical components, they are not suitable to represent cyber-physical incidents [8]. Moreover, they focus on representing the actions of an attack, while underrepresenting other characteristics such as resources and intent, which can be useful in a digital forensic investigation. Other work [9] focuses on modeling specific attacks (e.g., switching attacks) that can occur in certain application domains, such as smart grids. Thus,

this modeling technique cannot be applied to represent different types of attacks that can also happen in other application domains. Moreover, resources for capturing and sharing incidents commonalities, such as the Common Attack Pattern Enumeration and Classification (CAPEC) catalog [10], only focus on cyber attacks. Incident knowledge is represented using natural language, making it difficult to use the CAPEC catalog in an automated fashion.

In this paper, we propose an approach to represent and share incident knowledge. Our approach captures *incident patterns* – common aspects of incidents occurring in different CPSs. Our approach then allows incident patterns to be instantiated for different systems to assess if and how such patterns can manifest again. To support our technique, we provide two meta-models to represent, respectively, incident patterns and cyber-physical systems themselves. The incident pattern meta-model captures CPS incidents characteristics, such as activities, assets, actors, resources, goals, and motives. The system meta-model captures cyber and physical components and their interactions, which may be exploited during an incident. We demonstrate the feasibility of our approach using smart buildings as an application domain, by tailoring the system meta-model to represent components and interactions in this domain. Our ultimate objective is to use knowledge of previous incidents to enhance security, for example, by enabling security measures to prevent incidents conforming to some of the discovered patterns. Incident knowledge can also be leveraged to improve forensic readiness in CPSs [11]. For example, it is possible to identify data proactively that may be relevant to an incident (i.e. potential evidence) in order to support future digital investigations. Identifying potential evidence is considered a challenge and the first step towards forensic readiness [12].

The remainder of this paper is organized as follows. In Section 2 we discuss a motivating example for sharing incidents knowledge among different smart buildings. In Section 3 we describe our approach. In Sections 4 and 5 we illustrate, respectively, the cyber-physical system meta-model and the incident pattern meta-model. In Section 6 we apply our approach to our example. Finally, in Section 7 we conclude and discuss future work.

2 MOTIVATING EXAMPLE

We present an example to motivate why representing and sharing knowledge about incidents in cyber-physical systems is important. As depicted in Fig. 1, our scenario is centered on the *ACME Company* that operates across three different smart buildings: a *Research Center*, a *Warehouse*, and a *Manufacturing Plant*. The plan of the 2nd floor of the *Research Center* consists of a *Server Room*, a *Control Room*, and a *Toilet*. The *Server Room* has a *Fire Alarm*, an air conditioning unit *(HVAC)*, and some *Servers*. The *Control Room* has a *Workstation*. The whole building is equipped with smart lights. The listed devices, including the smart lights, are connected to the *Internal IP network.*

One day the security administrators of the *ACME Company* discovered that an incident occurred in the *Research Center*. An

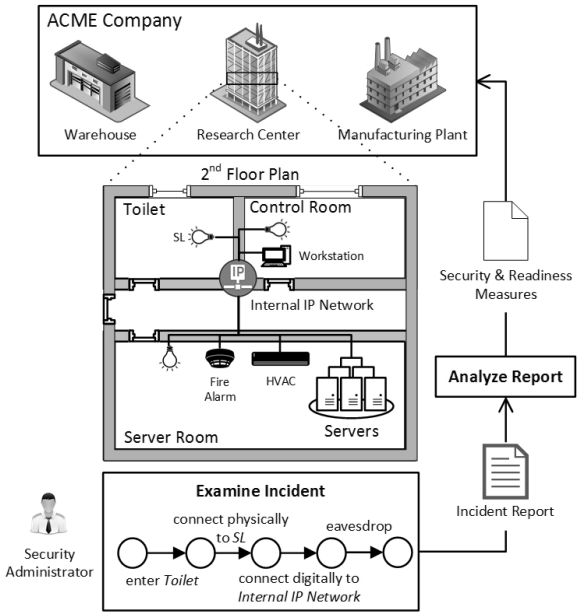

Figure 1 Motivating scenario for sharing incidents knowledge.

offender reached the 2nd floor, entered the *Toilet*, and connected to the smart light (*SL*) using a laptop. After that, s/he obtained access to the *internal IP Network* and was able to eavesdrop data transmitted over the network (e.g., data exchanged between the *Workstation* and the *Servers*). The incident actions are listed at the bottom of Fig. 1.

Upon the discovery of the incident, security administrators wrote a report describing how the incident occurred. They needed to assess whether similar incidents can take place also in the other smart buildings and in what ways. This would allow security administrators to enhance security in the smart buildings because they can enact security measures able to prevent similar incidents from happening. Moreover, this would allow identifying data indicating that similar incidents are occurring in the smart buildings. Monitoring this data proactively can support investigating these incidents, shall they occur.

To assess whether similar incidents characteristics can manifest also in the other buildings, security administrators have to examine the physical structure of each building, as well as the software and network configurations of the digital devices within the buildings in order to identify existing vulnerabilities brought by cyber and physical components. After that, security administrators can analyze what activities of an incident can reoccur because they can exploit discovered vulnerabilities. This methodology brings the following challenges:

- How should information about an incident be represented? Incident reports are usually written in natural language and may not be structured. Therefore, it can be arduous to analyze these documents manually. This can increase the effort required to assess whether certain incident characteristics can manifest again in other systems. Moreover, incident reports can contain information that is

too specific (e.g., access to the 2nd floor of the *Research Center* or to the *Toilet*) making it difficult to generalize incident knowledge to other domains. Incident reports also may contain sensitive information about the company (e.g., the internal network structure of the *Research Center*) hindering the possibility to share incident information with other companies.

- What information should be shared about an incident? Incident reports often focus on representing malicious actions. However, to perpetrate an incident an offender can perform both legitimate and malicious actions. Thus, representing only malicious actions might lead to overlooking some legitimate actions that are relevant (e.g., physical accessibility to smart lights in the *Toilet*). Consequently, data related to these actions might not be collected and stored proactively, hence, any future investigations of similar incidents might be more difficult because some relevant evidence is missing. Moreover, identifying vulnerabilities in a system can be difficult, since a system as a smart building can contain several hundreds of components with various vulnerabilities that can be exploited [13]. Some vulnerabilities can be thus overlooked due to human errors.
- Are available resources sufficient? Current resources such as the Common Vulnerabilities & Exposures (CVE) dictionary [14] focus on cybersecurity vulnerabilities, which are used to assess cybersecurity of a system. However, for incidents in cyber-physical systems, this is not sufficient due to the interactions between cyber and physical components that are often exploited (e.g., physical reachability to smart light to connect to the digital network) [15]. Moreover, to support digital investigations, it is also necessary to represent other incident characteristics, such as actors, resources adopted and assets targeted by an action. These characteristics have been neglected in existing incident representations.

To address the aforementioned challenges, we introduce our approach in the next section.

3 REPRESENTING & SHARING INCIDENTS KNOWLEDGE

Our approach aims to share incident knowledge across different cyber-physical systems. Incident knowledge is represented as *incident patterns* indicating common characteristics, such as activities, assets, resources, locations, and motives, among incidents that occurred in different systems. As shown in Fig. 2, our approach includes two main activities: 1) *Incident Pattern Extraction* and 2) *Incident Pattern Instantiation*. During incident patterns extraction, patterns are identified from incidents that occurred and are then stored in an *Incident Pattern Repository* shared across different systems. In our incident example, the actions "enter Toilet" and "connect to Internal IP Network using SL" can be expressed in a more abstract form into an incident pattern such as "enter Location" and "connect to IP network" activities. A *Visitor* who is inside the *Location* can perform both activities and can exploit a co-located *SmartDevice* to connect to

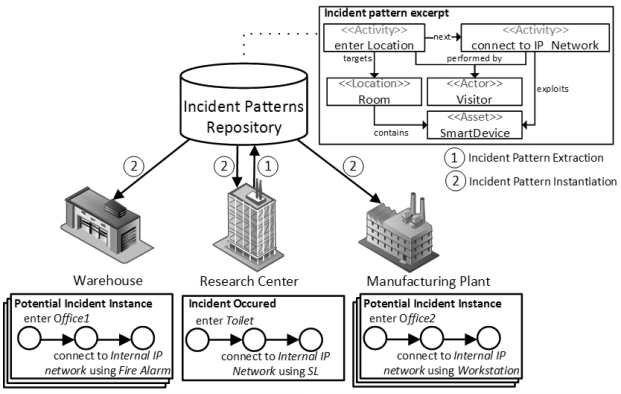

Figure 2 Application of our approach on the example.

the IP network. The extracted pattern can then be added to the repository. During incident patterns instantiation, patterns are mapped to different systems to identify potential incident instances i.e. to identify whether and how such patterns can manifest again. For example, the extracted pattern, shown in Fig. 2, can be instantiated to the *Warehouse,* the *Research Center* and the *Manufacturing Plant*. In the *Warehouse*, the pattern activities can be mapped, for example, to the actions "enter Office1" and "connect to IP network using Fire Alarm". In the *Manufacturing Plant*, the pattern activities can be mapped to the actions "enter Office2" and "connect to Internal IP Network using Workstation".

The I*ncident Pattern Extraction* activity is carried out as illustrated in Fig. 3. First, a security administrator models the incident that occurred. Incident modeling requires, as input, a system representation that specifies the system components and their potential interactions. In addition, incident modeling is assisted by an incident pattern meta-model and a system meta-model, which act as templates. The two meta-models are discussed later in the paper. Subsequently, the *Pattern Extraction* activity extracts a pattern from the incident model. This activity is assisted by the incident pattern meta-model and the system meta-model, which can be used to identify possible levels of abstractions that could be used to make the actions and the

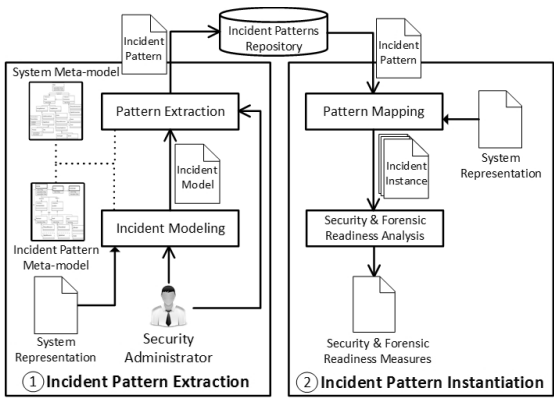

Figure 3 Incident Pattern Extraction (1) and Instantiation (2).

incident characteristics more general and re-usable across different systems. Several incident patterns with different levels of abstraction may be extracted, which can then be reviewed by a security administrator. Afterwards, the extracted pattern is sent to the repository and could be merged with other existing patterns, if necessary.

Furthermore, Fig. 3 shows the *Incident Pattern Instantiation* activity, which is executed as follows. During *Pattern Mapping,* an incident pattern is fetched from the repository and then mapped against a system representation to identify potential incident instances. Subsequently, during *Security & Forensic Readiness Analysis*, incident instances are analyzed to determine which security and forensic readiness measures should be applied to, respectively, prevent and investigate generated incident instances. Security measures can prevent some actions in the generated incident instances from occurring, while forensic readiness measures can identify incident-relevant data that should be collected proactively because they may constitute an evidence during future digital investigations.

4 REPRESENTING CYBER-PHYSICAL SYSTEMS

We describe a meta-model to represent cyber and physical components and their interactions. We tailor our system meta-model to represent smart buildings as an application area of CPS since we have a research interest in it.

A simplified version of the smart building meta-model is shown in Fig. 4. The meta-model includes the following entities. An *Asset* is an abstract entity that represents a component in a smart building such as a server. Each *Asset* instance is identified by its *name*. An *Asset* can be physical or digital. *PhysicalAsset*s represent any physical component in a smart building, such as *Actor*, *PhysicalStructure*, and *ComputingDevice*. *Actor* can be a person in the smart building such as a *Visitor* or an *Employee*. For example, in the research center incident example, the offender is represented as a *Visitor*. *PhysicalStructure* represents part of the smart building physical layout, which includes *Room* and *Floor*. For example, the *Toilet* in the research center can be defined as a *Room*. *ComputingDevice* represents any computing device such as *FireAlarm*, *SmartLight*, *Server, HVAC,* and *Workstation*. *DigitalAssets* can be any data or software that is created, stored, manipulated, run, or communicated in digital form such as *File* and *Processes*. *Process* has an attribute *status* that defines its current state (e.g., RUNNING or STOPPED).

Moreover, the meta-model allows representing containment and connectivity relations between system components. *Containment* is represented through relations *containedAssets* and *cotnainedDigitalAssets*. The *containedAssets* relation denotes the *Asset*(s) contained by a *TanigbleAsset*. For example, the *Server Room* in the research center can be defined as a *Room* containing *SmartLight*, *FireAlarm, HVAC,* and *Server*. The *containedDigitalAssets* relation denotes the *DigitalAsset*(s) contained by a *DigitalAsset*. For example, a *SmartLight* can contain a *Process* managing communication with other *ComputingDevices*. A *Connection* has attributes *asset1* and

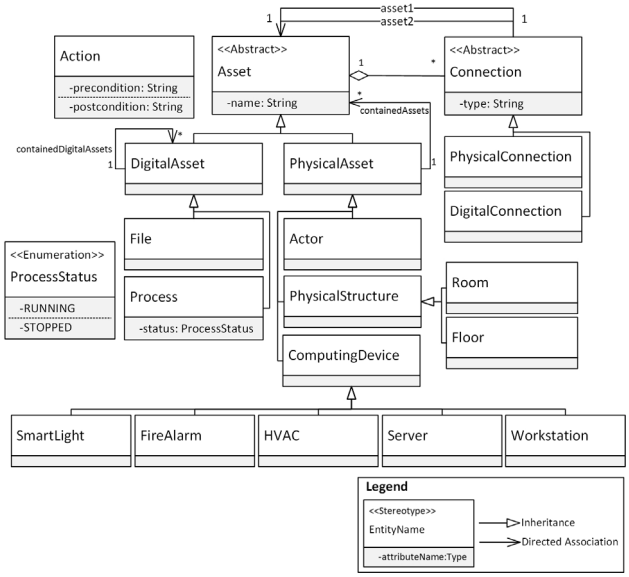

Figure 4 Smart building meta-model (simplified).

asset2, which represent both ends of a connection, and a *type* specifying the connection type such as "wired". *Connection* is extended to *DigitalConnection*, which represents connections between *DigitalAssets* and/or *PhysicalAssets*, and *PhysicalConnection*, which represents connections between *PhysicalAssets*. For example, a *DigitalConnection* can be defined between the *SmartLight* in the *Toilet* and the *Servers* in the *ServerRoom*, which has type *IP_network*. A *PhysicalConnection* between two *ComputingDevices* (e.g., *Fire Alarm* and *HVAC*) can be defined with the *type* wired. The meta-model also includes the entity *Action*, which specifies the dynamics of a system. For example, the *Research Center* can include actions such as "enter a Room" and "connect to a ComputingDevice". An *Action* may have a *precondition* and *postcondition* that describe, respectively, the required system state before the action is performed and the system state after the action is performed. For example, the *precondition* of action "enter a Room" is that the *Actor* performing the action is inside a *Room* that is physically connected to the *Room* to be accessed, for example, through a door. Although not addressed here, contextual constraints over entities' properties and actions could also be represented as state properties of an entity, such as *Context*. For example, a contextual constraint named *WorkingHours* can be attached to action "enter a Room", which indicates that accessing a room is permitted only during working hours. The *postcondition* of action "enter a Room" is that the accessed *Room* contains the *Actor* who performed the action.

The meta-model was implemented as an Eclipse plugin that is publicly available[1].

[1]https://tinyurl.com/yb2kkuvl

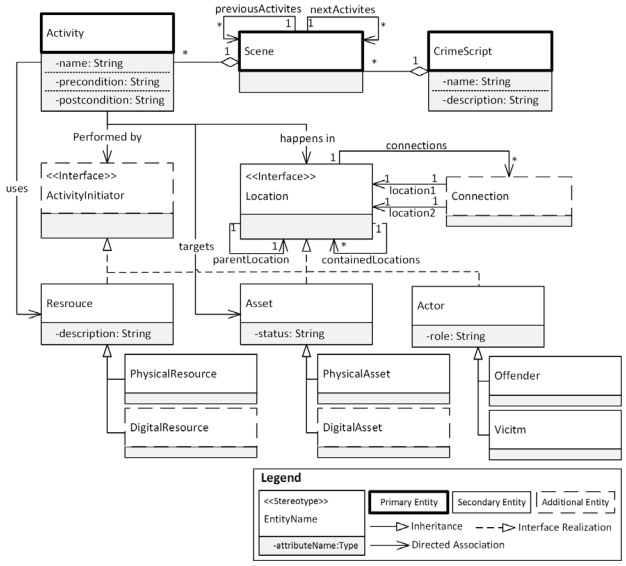

Figure 5 Incident pattern meta-model (simplified).

5 REPRESENTING INCIDENT PATTERNS

We represent incident patterns using a meta-model that is based on the concept of crime script [16]. A crime script is used in criminology to describe the sequence of activities of a physical incident in order to improve the understanding of the incident-commission process and the identification of incident prevention techniques [17]. However, there is a lack of a unified model to represent the entities and relationships found in a crime script. Moreover, crime scripts focus on physical incidents only, while neglecting cyber incidents and cyber-physical incidents.

Our meta-model captures the primary and secondary entities found in crime scripts. Primary entities are those represented in all crime script models published in the literature, while secondary entities are those mentioned, implicitly or explicitly, in most of the models published in the literature [17][18]. The meta-model also includes additional entities such as *DigitalAsset* to represent cyber components. A simplified version of the meta-model is shown in Fig. 5. The meta-model was implemented as an Eclipse plugin that is publicly available[2].

Fig. 5 shows the incident pattern meta-model. A primary entity is the crime script itself; it is characterized by a *name* and a *description. CrimeScript* entity includes a set of *Scenes*, which are the phases in which certain activities take place (e.g., preparation scene). Each scene, in turn, includes a set of activities that an entity performs during the incident. An *Activity* is characterized by a *name*, a *precondition* that represents the system state required to perform the activity, and a *postcondition* that represents the system state after executing the activity. An *Activity* also defines, as relations, its *nextActivities*, and *previousActivities*. An *Activity* corresponds to an *Action* entity in the system meta-model. The pre-/post-conditions of an *Activity* can be abstracted from the pre-/post-conditions of *Action*(s) defined in a system representation. For example, the *Action* "enter Toilet*"* can be abstracted to the *Activity* "enter Room*"*. Secondary entities are used to relate an activity to the entity performing it (e.g., victim or offender). Additional entities, such as *Asset*, *Resource,* and *Location* can better characterize an activity.

In the incident pattern meta-model, an *Asset* is an entity that can be harmed during an incident. The *status* of an *Asset* can be defined as an attribute. For example, a *Workstation* defined as an *Asset* can have on/off as *status*. An *Asset* can be further extended by the entities *DigitalAsset* and *PhysicalAsset*. An *Asset* can have a direct mapping to the *Asset* entity presented in the system meta-model. As shown in the previous section, *Assets* can be further extended in the system meta-model to represent more concrete entities such as *Room and ComputingDevice*.

An *Actor* represents a group or an individual who performs an activity and can be an *Offender* or a *Victim*. A *Resource* represents a tool needed to perform an activity. *PhysicalResource* refers to a physical tool used by an offender in an incident (e.g., laptop). *DigitalResource* represents a software tool that an offender can use to perform certain activities in an incident (e.g., malware). An *Actor* and a *Resource* could be extended by entities *Actor* and *Asset* represented in the system meta-model. A *Location* represents a place where an activity or a sequence of activities of a scene is performed. *Location* in the meta-model is an interface that is implemented by *Asset*, *Resource*, and *Actor*. A location can be physical or digital. A *PhysicalLocation* represents a place in the physical space (e.g., a room) where an activity or a sequence of activities takes place. A digital location represents a place in the cyberspace such as an IP address or a digital folder. For each *Location*, contained locations can be defined via the relation *cotnainedLocations,* and also its parent location via the relation *parentLocation*.

Connections can be defined between a *Location* and other entities (e.g., digital connection between two *Workstations*). A *Connection* has a direct mapping to a more concrete *Connection* entity that is defined in the system meta-model. For example, a *DigitalConnection* in an incident pattern can be an abstraction of a more concrete *Connection* (e.g., *WiFiConnection*) defined in a system representation. The *ActivityInitiator* is an interface that defines the entity that performs an activity. *ActivityInitiator* is implemented by entities *Actor*, *Asset*, and *Resource*. This implies that our meta-model allows, not only an *Actor* to perform activities, but also *Asset* and *Resource*. For example, an activity may be performed by a malware, which can be considered as a *Resource*.

The incident pattern meta-model has the potential to provide a systematic and rich representation of incidents since it encompasses not only the activities of an incident but also related entities and relationships (e.g., location, assets, and actors). Moreover, the possibility to extend the meta-model entities with domain-specific entities identified from a system representation makes our meta-model extensible and general enough to be applied to different types of systems.

[2]https://tinyurl.com/y796ouyq

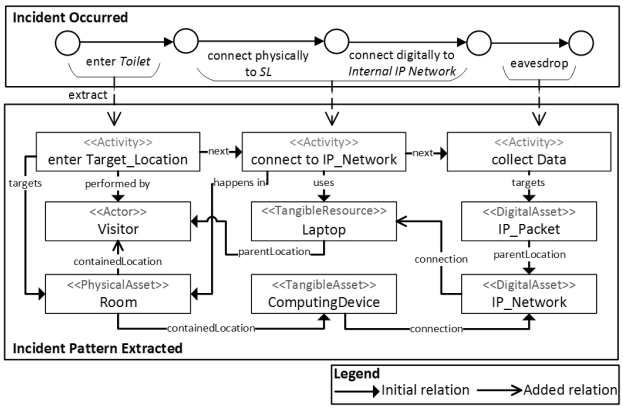

Figure 6 Incident pattern extraction.

Fig. 6 shows the incident pattern extracted from the incident example described in Section 2. Action "enter Toilet" can be abstracted to the activity "enter Target_Location" and is performed by a *Visitor.* A *precondition* for this activity is that the *Target Location* should be a *Room* that contains a *SmartDevice* connected to a *DigitalAsset* of type *IP_Network*. A *postcondition* for the activity is that *Target Location* contains *Visitor*.

Actions "connect physically to SL" and "connect digitally to internal IP network", can be abstracted to activity the "connect to IP_Network". This activity is associated with entities *Room*, *IP_Network*, and *Laptop*. A *precondition* for this activity is that the *Visitor* contains *Laptop* i.e. the visitor carries a laptop. This activity requires activity "enter Target_Location" to have occurred, and results in the creation of a new *connection* between the *Laptop* and the *IP_Network*.

To further illustrate the mapping from an incident to a pattern, we suggest the following guidelines. The first aspect to determine is what sequence of actions can be mapped to a single activity. This depends on how closely related are these actions, for example, whether they share many of their entities and relations.

Second aspect to consider is identifying entities in an incident that need to be extracted. Each entity associated with an action can be used, targeted, exploited, initiator, or can denote a location. Determining the role of an entity in an action can determine the entity type to use from the incident pattern meta-model. For example, the action "enter Toilet" has the entities *Visitor*, which represents the ActivityInitiator as an *Actor*, and *Toilet* which represents a *PhysicalAsset* that is a targeted *Location*. The relationship between the *Visitor* and the *Toilet* is that the *Toilet* should *contain* the *Visitor* after executing the action.

Third aspect is to determine what level of abstraction and properties is appropriate for the pattern. This will heavily depend on the level of details needed. In our approach, we use a system meta-model to determine possible abstraction levels. For example, in the system meta-model, a *SmartLight* can be abstracted to *ComputingDevice*, *PhysicalAsset*, and *Asset*, ranging from the least abstract to the most abstract entity. When a more abstract entity is adopted (e.g., *PhysicalAsset*) the incident pattern can be applied to a wider set of systems compared to when a more specific entity is considered such as *SmartLight*. The choice of a suitable level of abstraction requires the intervention of a security administrator. For example, if the objective is to investigate ways in which a computing device (smart light or other) can be exploited to connect to a network, then using *SmartLight* will not be sufficient, so *ComputingDevice* would be a more suitable abstraction.

Finally, a sequence of actions of an incident may be abstracted by reusing existing activities of incident patterns that have already been stored in the repository. For example, if an offender exploited the fire alarm in an office to gain access to an internal IP network, an incident pattern could be created using activities "enter Location" then "connect to IP network using DigitalAsset". These activities can be mapped to the actions of the incident that occurred in the research center.

6 USING INCIDENT PATTERNS

In this section, we demonstrate how our approach uses incident patterns to assess how such patterns can manifest in the other smart buildings i.e. the *Warehouse* and the *Manufacturing Plant.*

As shown at the bottom of Fig. 7, the *Warehouse* has three rooms (*Office1*, *Office2*, and *Toilet*), and a *Storage Area*. *Office1* contains a *Fire Alarm* and a *Smart Light* (*SL1*). *Office2* contains a *Smart Light* (*SL2*). The *Toilet* contains a *Smart Light* (*SL3*). *Fire Alarm*, *SL1*, and *SL2* are connected to the *Internal IP Network*. Based on this building configuration, the incident pattern can be mapped to 3 potential incident instances as shown in Fig. 7. Activity "enter Target_Location" can be mapped to actions "enter Office1" or "enter Office2". This is because both offices contain smart devices (e.g., *Fire Alarm, SL2*) that are in turn connected to the *Internal IP Network*. This satisfies the precondition of activity "enter Target_Location" requiring that the entered room contains a *SmartDevice* connected to *DigitalAsset* of type *IP_Network*. "enter Toilet" is not a possible action since *SL3* in the *Toilet* is not connected to the *Internal IP Network*, hence, it does not satisfy the activity precondition. The next activity "connect to IP_Network" can be mapped to six different actions that depend

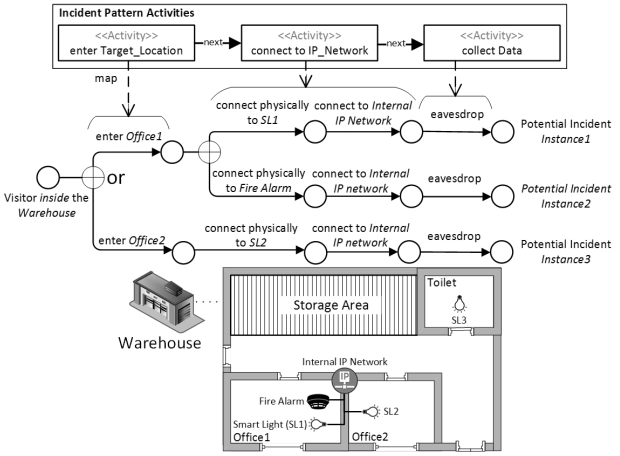

Figure 7 Map of incident pattern to the Warehouse.

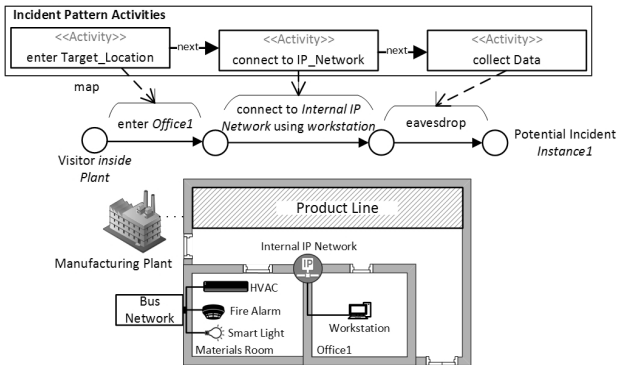

Figure 8 Map of incident pattern to the Manufacturing Plant.

on which smart device is exploited i.e. *SL1, SL2, or Fire Alarm*. For example, if the *Visitor* is in *Office1*, then the activity may be mapped to the two actions "connect physically to SL1" and "connect to Internal IP Network". Similar actions can be identified for the *Fire Alarm* and *SL2*. The final activity "collect data" can be mapped to one action "eavesdrop" in this case, however, more actions can be chained to satisfy the activity if more details are provided, such as what type of data can be eavesdropped from the IP network.

A similar approach can be adopted to map the incident pattern to the representation of the *Manufacturing Plant* shown at the bottom of Fig. 8. The *Manufacturing Plant* contains two rooms (*Office1* and *Materials Room*) and a *Product Line*. *Office1* contains a *Workstation* that is connected to the *Internal IP Network*. *Materials Room* contains a *Smart Light*, *Fire Alarm*, and *HVAC* all connected to a separate network that is the *Bus Network*. According to this configuration, one potential incident instance can be identified, as shown in Fig. 8. Activity "enter Target_Location" is mapped to action "enter Office1", since it satisfies the precondition of the activity. The action "enter Materials Room" is not a possible action since none of the devices in the *Materials Room* are connected to the *Internal IP Network*. Assuming that the *Visitor* is in *Office1*, activity "connect to IP_Network" can be mapped to action "connect to Internal IP Network using Workstation". Finally, the activity "collect data" can be mapped to the action "eavesdrop".

Different configurations of the cyber and physical components in smart buildings may lead to different manifestations of the same incident pattern, as shown earlier. These manifestations can be further reasoned about to identify adequate measures to improve security and forensic readiness of a system. For example, a security measure for the *Warehouse* is to ensure that smart devices are firmly installed to prevent physical manipulation.

7 RELATED WORK

The literature shows little work has been done to represent and share incidents knowledge in cyber-physical systems. Current Attack Modeling Techniques (e.g., attack graphs [7]) focus on representing how a traditional cyber-attack (e.g., denial of service attack) can occur. Resources available are also focusing on sharing information about cyber attacks.

A close work representing CPS incidents is proposed by Hawrylak et al. [19]. In this work, the authors develop Hybrid Attack Graph (HAG) to model cyber-physical attacks. HAG is a formalism that produces a graph of all possible ways a set of exploit patterns can be applied to a system. However, the approach focuses on representing malicious actions that exploit vulnerabilities found in some devices and does not consider other non-malicious interactions between cyber and physical components that can lead to undesired state. Additionally, the work focuses on representing actions, while neglecting other incident characteristics (e.g., intent, resources) that can be useful during digital forensic investigations. Chen et al. [20] use Petri nets as a modeling formalism to represent cyber-physical attacks on a smart grid. The approach represents concurrent physical and digital events to represent coordinated attacks performed by multiple attackers working in parallel. For example, an offender hacks the access control system [digital event], while another enters a prohibited location [physical event]. However, this approach still focuses on events and does not explicitly model other aspects of an incident that can be relevant for an investigation such as actors, and assets.

Yampolskiy et al. [8] propose a cyber-physical attack description language (CP-ADL) that is based on a six-dimensional taxonomy of attacks on CPS. In their work, an incident is represented as a causal chain, which contains a set of atomic attacks. An atomic attack consists of a set of actions (includes attack means, and preconditions), cause (includes attack element, and changes), and effect (includes influenced element and impact on it). However, their approach does not consider other aspects of incidents such as locations of elements. Clausing et al. [21] provide a general attack modeling approach for industrial facilities. Their approach is based on designing a shared architecture view for Industrial Control Systems (ICS), which consists of several elements: entity, interface, carrier, protocol, humans, and data. Their focus is on modeling the system components, then adding steps of an attack to it. However, the approach is specific to ICS.

Resources for sharing CPS incidents knowledge are limited. Currently, available resources provide information about cyber attacks. For example, the Common Vulnerabilities & Exposures (CVE) [14] is a publically available dictionary of known cybersecurity vulnerabilities in software and devices. Moreover, The US National Vulnerability Database (NVD) [22] is a database, based on CVE, of cybersecurity vulnerabilities that includes various metrics such as severity, impact on environment, and interactions required from users. The Common Attack Pattern Enumeration and Classification (CAPEC) [10] catalog provides a textual description of various attacks against "cyber-enabled capabilities". Repository of Industrial Security Incidents (RISI) [23] is a private resource that provides reports about incidents that occurred in ICS. However, both CAPEC and RISI provide information expressed in natural language about incidents. Therefore, incident information cannot be processed automatically.

8 CONCLUSIONS & FUTURE WORK

We proposed an approach to share incidents knowledge using *incident patterns*. Incident patterns capture commons aspects of incidents occurring in different systems. To support our approach, we provided two meta-models that represent, respectively, incident patterns and the cyber-physical systems themselves. We described a meta-model to represent components and interactions in smart buildings. We also discussed the incident pattern meta-model and gave an example of an incident pattern and some guidelines to create them. We demonstrated how our approach could be used to create an incident pattern, and map such pattern to different systems to identify how similar incidents may reoccur.

In future work, we intend to evaluate expressivity of our incident pattern meta-model by using it to represent different incidents that can occur in CPS. We will try to model synthetic incidents extracted from the literature as well as real incidents. In addition, we intend to develop a technique to automate the process of extraction of incident patterns and instantiation of such patterns to cyber-physical systems. To instantiate incident patterns, we intend to use a modeling formalism to reason about system dynamics. Bigraphical Reactive Systems (BRS) [24] are a strong candidate to reason about system dynamics since they provide reaction rules to express system evolution. BRS use Bigraphs to represent the system state. Bigraphs allow representing the configuration of cyber and physical components as well as their interactions. This eliminates constraints (e.g., limited connectivity) imposed by other formalisms such as action calculi, which are suitable to represent interactions only between physical or cyber components. Moreover, BRS have been used to reason about CPS for various applications domains such as adaptive security systems [25]. Finally, we plan to develop a technique to analyze incidents that are generated from mapping a pattern to a system. Our analysis will aim at identifying potential evidence (e.g., assets, actions), which can be collected and stored proactively for the purpose of supporting future investigations. We will apply our techniques, once developed, to several scenarios to evaluate them against some metrics such as correctness, performance and scalability.

ACKNOWLEDGMENTS

This work was partially supported by ERC Advanced Grant no. 291652 (ASAP) and Science Foundation Ireland grants 10/CE/I1855, 13/RC/2094 and 15/SIRG/3501.

REFERENCES

[1] E. A. Lee, "CPS foundations," in *Proceedings of the 47th Design Automation Conference*, 2010, pp. 737–742.

[2] V. Gunes, S. Peter, T. Givargis, and F. Vahid, "A survey on concepts, applications, and challenges in cyber-physical systems," *KSII Trans. Internet Inf. Syst.*, vol. 8, no. 12, pp. 4242–4268, 2014.

[3] G. Loukas, *Cyber-Physical Attacks: A Growing Invisible Threat.* Butterworth-Heinemann, 2015.

[4] R. M. Lee, M. J. Assante, and T. Conway, "Analysis of the Cyber Attack on the Ukrainian Power Grid," 2016.

[5] R. M. Lee, M. J. Assante, and T. Conway, "German Steel Mill Cyber Attack," *Ind. Control Syst.*, pp. 1–15, 2014.

[6] A. Cardenas, S. Amin, B. Sinopoli, A. Giani, A. Perrig, and S. Sastry, "Challenges for Securing Cyber Physical Systems," in *Workshop on future directions in cyber-physical systems security*, 2009, p. 5.

[7] H. S. Lallie, K. Debattista, and J. Bal, "An Empirical Evaluation of the Effectiveness of Attack Graphs and Fault Trees in Cyber-Attack Perception," *IEEE Trans. Inf. Forensics Secur.*, pp. 1–1, 2017.

[8] M. Yampolskiy, P. Horvath, X. D. Koutsoukos, Y. Xue, and J. Sztipanovits, "A language for describing attacks on cyber-physical systems," *Int. J. Crit. Infrastruct. Prot.*, vol. 8, pp. 40–52, 2015.

[9] S. Liu, S. Mashayekh, D. Kundur, T. Zourntos, and K. Butler-Purry, "A framework for modeling cyber-physical switching attacks in smart grid," *IEEE Trans. Emerg. Top. Comput.*, vol. 1, no. 2, pp. 273–285, 2013.

[10] MITRE Corporation, "Common Attack Pattern Enumeration & Classification." [Online]. Available: http://capec.mitre.org/. [Accessed: 18-Jan-2018].

[11] R. Rowlingson, "A ten step process for forensic readiness," *Int. J. Digit. Evid.*, vol. 2, no. 3, pp. 1–28, 2004.

[12] F. Alrimawi, L. Pasquale, and B. Nuseibeh, "Software Engineering Challenges For Investigating Cyber-Physical Incidents," in *Proceedings of the 3rd International Workshop on Software Engineering for Smart Cyber-Physical Systems*, 2017, pp. 34–40.

[13] T. Mundt and P. Wickboldt, "Security in building automation systems-a first analysis," in *2016 International Conference On Cyber Security And Protection Of Digital Services (Cyber Security)*, 2016, pp. 1–8.

[14] MITRE Corporation, "Common Vulnerabilities & Exposures (CVE)." [Online]. Available: https://cve.mitre.org/. [Accessed: 15-Nov-2017].

[15] Z. DeSmit, A. E. Elhabashy, L. J. Wells, and J. A. Camelio, "An approach to cyber-physical vulnerability assessment for intelligent manufacturing systems," *J. Manuf. Syst.*, vol. 43, pp. 339–351, Apr. 2017.

[16] D. Cornish, "Crimes as scripts," in *Proceedings of the inter. seminar on env. criminology and crime analysis*, 1994, pp. 30–45.

[17] D. Cornish, "The procedural analysis of offending and its relevance for situational prevention," *Crime Prev. Stud.*, vol. 3, pp. 151–196, 1994.

[18] H. Brayley, E. Cockbain, and G. Laycock, "The Value of Crime Scripting: Deconstructing Internal Child Sex Trafficking," *Policing*, vol. 5, no. 2, pp. 132–143, 2011.

[19] P. J. Hawrylak, M. Haney, M. Papa, and J. Hale, "Using hybrid attack graphs to model cyber-physical attacks in the Smart Grid," in *5th ISRCS*, 2012, pp. 161–164.

[20] T. M. Chen, J. C. Sanchez-Aarnoutse, and J. Buford, "Petri net modeling of cyber-physical attacks on smart grid," *IEEE Trans. Smart Grid*, vol. 2, no. 4, pp. 741–749, 2011.

[21] R. Clausing, R. Fischer, J. Dittmann, and Y. Ding, "Your Industrial Facility and Its IP Address: A First Approach for Cyber-Physical Attack Modeling," Springer, Cham, 2016, pp. 201–212.

[22] NVD, "The US National Vulnerability Database (NVD)." [Online]. Available: https://nvd.nist.gov/. [Accessed: 15-Nov-2017].

[23] RISI, "The Repository of Industrial Security Incidents (RISI)." [Online]. Available: http://www.risidata.com/. [Accessed: 15-Nov-2017].

[24] R. Milner, "Bigraphical Reactive Systems," *CONCUR 2001 --- Concurr. Theory*, vol. 2154, pp. 16–35, 2001.

[25] C. Tsigkanos, L. Pasquale, C. Ghezzi, and B. Nuseibeh, "On the Interplay Between Cyber and Physical Spaces for Adaptive Security," *IEEE Trans. Dependable Secur. Comput.*, 2016.