A Semantic Framework With Humans in the Loop for Vulnerability-Assessment in Cyber-Physical Production Systems *

Yuning Jiang[1][0000−0003−4791−8452], Yacine Atif[1][0000−0002−7312−9089], Jianguo Ding[1][0000−0002−8927−0968], and Wei Wang[2][0000−0003−1781−2753]

[1] School of Informatics, University of Skövde, Sweden
[2] School of Engineering Science, University of Skövde, Sweden
{firstname.lastname}@his.se

Abstract. Critical manufacturing processes in smart networked systems such as Cyber-Physical Production Systems (CPPSs) typically require guaranteed quality-of-service performances, which is supported by cyber-security management. Currently, most existing vulnerability-assessment techniques mostly rely on only the security department due to limited communication between different working groups. This poses a limitation to the security management of CPPSs, as malicious operations may use new exploits that occur between successive analysis milestones or across departmental managerial boundaries. Thus, it is important to study and analyse CPPS networks' security, in terms of vulnerability analysis that accounts for humans in the production process loop, to prevent potential threats to infiltrate through cross-layer gaps and to reduce the magnitude of their impact. We propose a semantic framework that supports the collaboration between different actors in the production process, to improve situation awareness for cyberthreats prevention. Stakeholders with different expertise are contributing to vulnerability assessment, which can be further combined with attack-scenario analysis to provide more practical analysis. In doing so, we show through a case study evaluation how our proposed framework leverages crucial relationships between vulnerabilities, threats and attacks, in order to narrow further the risk-window induced by discoverable vulnerabilities.

Keywords: Cyber-Physical Production System Security · Human-in-the-Loop · Vulnerability Assessment · Semantic Model · Reference Model.

1 Introduction

Industry 4.0 is the current trend of automation in manufacturing sector. Cyber-physical systems (CPSs) are the main driver of the fourth industrial revolution trend, which is evolving as an interaction between ICT systems and control elements used to operate a physical process in order to achieve production

* This research has been supported in part by the EU ISF Project A431.678/2016 ELVIRA (Threat modeling and resilience of critical infrastructures), coordinated by Polismyndigheten/Sweden.

objectives. A cyber-physical production system (CPPS) is supported by CPS controls to respond to changing conditions, and anticipate changes in physical processes [1]. However, the increasing connectivity facilitated by communication links within and across CPS networks, could prompt an adversary to exploit vulnerabilities along those communication links to create cyber-attacks. For instance, in the year of 2017, the "WannaCry" ransomware attack occurred in several manufacturing plants and caused production to stop [2], incurring substantial business losses.

Meanwhile, dynamic and complex production processes involve multi-domain enterprise management procedures, which may result in communication gaps throughout interconnected application-specific sub-systems of the overall production fabric [1]. Consider Numerical Controllers or NC machining part as an example, which describes a production system typically across four stages, namely part-design that defines its Product and Manufacturing Information (PMI) data, process planning that creates the detailed NC machining process data, part machining that runs this process data on Computer Numerical Controller or CNC machines and the tool condition data to monitor the production process, as well as quality-inspection that involves quality-assessment data. Groups of application-specific staff are responsible for design, machining, and inspection activities within the production process, such as designer, process planner, CNC machine operator and quality inspector. Software administrators are mainly responsible for operational and maintenance tasks to ensure capabilities of Software-as-a-Service (SaaS) within the cloud-based environment, to enable the services of software programs such as computer aided design (CAD) and computer-aided manufacturing (CAM) programs. Therefore, a concrete model should be based on multiple sources of heterogeneous data [3] which needs to be transformed into a common semantic representation [4], and in a machine readable format, to improve a common view of situation awareness.

However, current vulnerability instance response mechanisms in complex CPPS are faced with challenges to bridge the knowledge gap between cybersecurity techniques, industrial control system (ICS) expertise, and socio-technical management procedures, that involve human actors in the production lifecycle. A successful attack that propagates without notice could result in severe impact, due to lack of communication through manufacturing networked-layers and related operators such as network administrators, application-specific engineers and security managers. However, in current manufacturing management structures, the communication between different groups is limited due to inherent differences in working contents. Therefore, it appears vital to set up a common framework to provide a unified understanding from different views, in order to prevent potential threats to infiltrate through cross-layer gaps and to reduce the magnitude of their impact.

In this paper, a semantic framework of vulnerability-assessment with human-in-the-loop is proposed to facilitate a greater level of automation in vulnerability assessment and support a greater level of communication between vulnerability-handling stakeholders. The rest of this paper is organised as follows: In Section

II, we explore the state of the art and compare current research approaches against our method. In Section III, we introduce the background of CPPS, with emphasis on topological structures and functional dependencies. In Section IV, we propose our vulnerability-assessment framework to set up semantic mapping between threat, vulnerability and attack instances, and leverage use cases via actors involved in the production process through which such infiltration and evolution may occur. In Section V, we provide a case study to present and evaluate an application of our vulnerability-assessment framework in CPPS from different perspectives. In Section VI, we provide some concluding remarks and discuss some future research directions.

2 Related Works

Qualitative Vulnerability-assessment models primarily address relationships among vulnerability and risk, to express vulnerability observations based on qualitative data, such as those employed in security-risk management frameworks like SECTEC [8]. In these works, an ontology is used for system implementation as a vocabulary basis consisting of facts (both abstract facts and entity facts), constraints, types and attributes of the system, in order to adopt a common semantic information for knowledge base construction, and to support modelling of security applications. However, these frameworks focus on risk-management fragment on an abstract level, and do not address the details of other security elements, such as how a vulnerability could be exploited by a threat and further materialise into an attack. Therefore, these frameworks suffer from being vague and ultimately subjective. Some works take into consideration the complex and dynamic attributes of CPS [7]. For instance, Quality Control (QC)-based taxonomies are setup by a taxonomy of attack types on CPPSs to improve quality control [10]. Still, these works mostly concentrate on risk-management and/or vulnerability-management from a management perspective, and do not consider the complex and dynamic attack behaviours and exploit patterns.

Some other related works concentrate on attack modelling, such as the metamodel based architecture pwnPr3d (referring to an attack-graph-driven probabilistic threat-modelling approach) [9], which highlights the need for automatic attack-graphs generation to mitigate cyber-attacks. However, these models focus on attack-steps and corresponding prerequisites in vulnerabilities instead of the vulnerable nature of the system. That is they focus on defining attack vectors that model attack-patterns, more than exploit vectors that represent vulnerability patterns. Both patterns are used to model cyberthreat patterns though. The complex structure of CPPSs and the limited computing capability embedded sensing devices lower the protection degree to withstand these cyberthreats, and thus contribute to increasing their vulnerability degree. Such vulnerabilities emerge from specific features of CPPS that are not well addressed in previous works. Furthermore, these works mostly focus on attack modelling and mitigation techniques, while neither works contribute to bridging security controls across CPPS management perspectives.

One missing point in the previous approaches is to take into account the stakeholders to contribute to vulnerability assessment, which can be further combined with attack-scenario analysis to provide more practical security-assessment. Stakeholders with knowledge about CPPS can provide valuable information about vulnerability-exploitability with varying degrees of impact-severity, and are important to be involved in the assessment cycle to support analytics-based decision-making processes to protect critical infrastructures. Our approach proposes a semantic framework to support communication between risk managers, cybersecurity engineers, and also domain-specific operators. This approach also illustrates the connections between different vulnerability instances, threat instances, and attack instances in CPPS, to extract threat, vulnerability and attack (TVA) patterns that support CPPS vulnerability evaluation.

3 Cyber-Physical Production System Security and Human Actors

Conceptually, a cyber-physical system includes a cyber, a control and a physical process layer [5]. In smart manufacturing or CPPS, the control layer includes a network of microprocessor-controlled physical objects, such as programmable logic controllers (PLCs), which interface with physical process sensors. The physical process consists of a production-flow regulated by workstation machines and other manufacturing equipments. Thus, the control layer relays measurements from sensors that interact with field devices such as milling machines and drilling machines, to remote control centres, as illustrated in Fig. 1.

In the physical layer, engineers or operators could locally maintain workstations using local stations or Human Machine Interfaces (HMIs). In the control-layer, Operating Technology or OT administrators, engineers or operators could remotely maintain workstations using remote stations or through a virtual private network (VPN), to optimise production operations. Then, engineers use control and command servers to process these data to support operational production decisions, and to synchronise their operations. The top cyber-layer expresses decision-support analytics to manage the underlying control-system operations, in an enterprise platform of application servers and datastore. The application-servers provide various application services, containing CAD server, software-update server, operation-system server, time-unit server, web server, etc,. The datastore includes process-data server, historian-database, and domain-controller. Specifically, design engineers use CAD program to design product through application servers and store the corresponding 3D-model files in data-store servers. Software administrators use software-update server to update out-dated firmware or system software, with the support from historian-database. The process-data server stores and transmits design-, process- and manufacturing-data from production-flow, which supports file transfer between data analysers. The historian-database stores historical-data from application servers, which is queried by operators to monitor production processes. The domain-controller

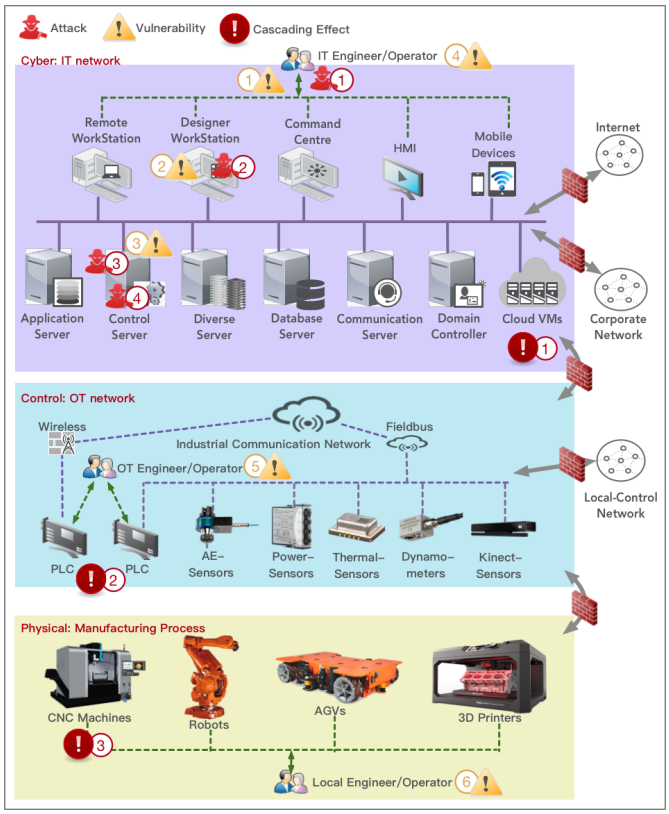

Fig. 1. Cyber-Physical Production System and Involved Human Actors

reserves user-information, and supports corresponding authorisation maintained by administrators.

An advanced persistent threat (APT) might be materialised by an attacker using vulnerability-chain[3], whereby exploits on one particular component may give access to another exploit on another set of components, as illustrated in Fig. 1. *V-x* refers to different vulnerabilities, while *A-x* refers to different attacks. *V-1(default password setting)* of an application-engineer's account might be exploited by an attacker to compromise this account in Attack *A-1*, which might be followed by another Attack *A-2* by trying to compromise a correlated designer workstation using compromised accounts. If the designer-workstation has *V-2(weak authentication management)*, then *A-2* might have a higher success probability. Furthermore, another attack *A-3* might be triggered by the threat agent to gain access to a database server through a compromised designer-workstation. A potential vulnerability *V-3(weak access control)* might allow *A-3*

[3] http://cwe.mitre.org/documents/glossary

to happen, which might let the threat agent further trigger another attack *A-4* to manipulate certain geometry CAM programmes in the control server.

The previous successful attacks of the CAM programme code manipulation may stay unnoticed due to *V-4(insufficient communication between CAM-engineers and security officers)*, which gives time to the attacker to compromise the whole IT network. Normally, IT department and OT department are distributed in different locations and may have *V-5 and V-6 (communication gaps between IT-personnel, OT-personnel and local operators)*. In this case, the manipulated CAM file might reach PLC without any correction, and further triggers wrong movement track in NC or CNC machines. This can result in a severe consequence in the production process due to communication gaps.

4 Cyber-Security Conceptual Framework

To prevent threat-induced anomalies or intrusion attempts, vulnerabilities need to be rooted out from CPPS infrastructure and assessed to enumerate and rank their severity. This assessment involves modelling vulnerability to account for salient features, as well as identifying critical-component of CPPS infrastructure to weigh vulnerabilities, while accounting for CPPS actors who may be contributed to threat vectors. The component-model in Fig. 2 shows our proposed concept unified modelling language (UML) framework of CPPS cybersecurity taxonomic links across *System*-Objects, *CPPS*-Objects, as well as *Actor*-Objects. Ontological method is adopted to tie eliminate ambiguity and support consistency checking. This semantic framework could support vulnerability-driven cybersecurity analysis across CPPS environments. We also provide an user-interaction model in Fig. 3 that reveals related semantic relationships of contributing stakeholders, based on which we involve different actors into the vulnerability assessment process. The proposed vulnerability assessment methods are later evaluated in the context of CPPS case study.

A) Component Model for System-Induced Vulnerability Analysis

The component model represents two concepts and related information, *SecurityObject*, and *CPPSObject*. A vulnerability could be regarded as an emergent property of an asset within CPPS. Differentiating intrinsic-properties and emergent-properties of an asset could support the detection of abnormal behaviours. Confidentiality, integrity and availability are intrinsic metrics of impact property, while confidentiality-weakness, integrity-weakness, and availability-weakness measure emergent-properties. Vulnerability-assessment evaluates a potential *asset* disruption prospect. Combining *Criticality* of vulnerable asset, and *Severity* of all the emerged *Vulnerability* in this vulnerable asset, we could further compute *Vulnerability Index* at the asset level.

SecurityObject accumulates TVA (Threat, Vulnerability and Attack) information and related relationships. Vulnerabilities may be exploited by an *Attacker* (i.e. threat agent) in different ways using *Exploits*. An *Attacker* may further

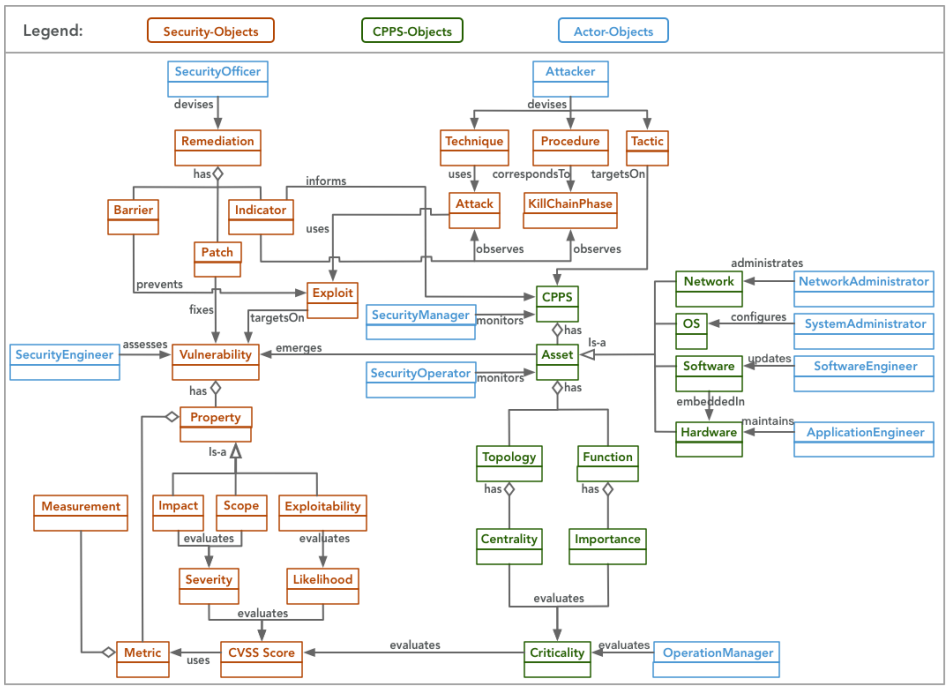

Fig. 2. Vulnerability-Driven CPPS Security Component Model

trigger an attack in different ways. Each vulnerability instance has specific impact, scope and exploitability properties. Each *Exploit* may be used to disrupt a targeted component in *CPPS* infrastructure in different ways. A vulnerability-assessment process mainly involves the identification, definition and evaluation of how exploits trigger attacks, and the magnitude of those attacks i.e. *Impact*. Furthermore, our framework uses cybersecurity-attributes in the form of value-pairs, and could provide a multidimensional view of vulnerabilities, to further support a quantitative analysis using scoring mechanisms such as the industrial standard Common Vulnerability Scoring System (CVSS)[4] to evaluate the severity of each identified vulnerability, and to support risk analysis.

CPPSObject aggregates CPPS assets and related components. Systems and softwares components of various assets in the digitalised industry are interconnected. Vulnerabilities emerge due to these interconnections. Considering the nature of CPPS, we define an asset component to be either a software (e.g. a CAD program), a hardware (e.g. a milling machine), an Operating System (OS), or a network (e.g. a TCP/IP protocol). A software is embedded in a hardware, to form the asset that drives a physical process, for instance electricity supply. Vulnerabilities could be initially categorised based on corresponding types of assets. Different vulnerabilities might contribute to threats that bring different

[4] https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator

levels of impact, measured through levels of losses in confidentiality, integrity and availability (CIA) triad as well as in view, control and/or communication of the physical process. For example, a CPPS hardware may be vulnerable by having no physical-access protection, which might be used by an attacker to gain unauthorised physical access through USB, that could be unknowingly introduced by other legitimate CPPS actors.

B) User Interaction Model for Human-Induced Vulnerability Management

Dynamic and cooperating vulnerability analysis allows mitigations to occur within the time interval that span the discovery and disclosure of vulnerabilities, and giving time for vulnerability patches to become available and deployed before the time whereby exploits are made public. In our user-interaction model as illustrated in Fig. 3, *ActorObject* includes *CPPS-Staff*, and *Attacker*. Each instance of *ActorObject* may further have a profile including identity information (i.e. role and label) and character (i.e. grouping and sophistication). *Attacker* has specific *Technique*, *Procedure* and *Tactic* used to trigger attack instances.

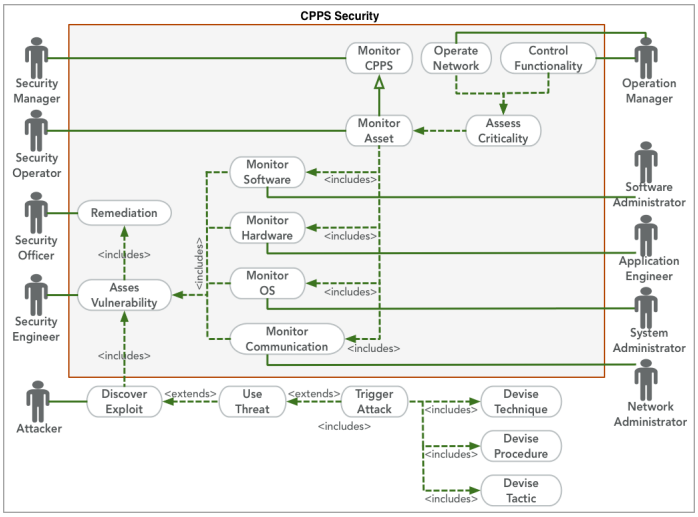

Fig. 3. CPPS Security User Interaction Model

CPPS-Staff instantiates *SecurityOfficer*, *SecurityManager*, *SecurityOperator*, *SoftwareAdministrator*, *ApplicationEngineer*, *NetworkAdministrator*, etc,. textitSecurityOperator monitors corresponding *Assets*. *SecurityManager* monitors CPPS *Indicators*. Conceptually, *SecurityOfficer* needs to gauge budget investments through adopting an expert-system interpretation of numerical vulnerability-indicators to carry out mitigation decisions like vulnerability-patching. *SecurityEngineer* rank assets by using following vulnerability-indicators : a)their

Criticality evaluated by *OperationManager*, b)their vulnerability exploitability *Likelihood* identified by *SystemAdministrator*, *SoftwareAdministrator*, and *NetworkAdministrator*, as well as c)the impact *Severity* of threats defined by *ApplicationEngineer* [6]. By distributing dynamic vulnerability-management tasks throughout CPPS organisation, we argue that it improves the level of communication between vulnerability-handling stakeholders.

5 Vulnerability Analysis with Human-In-The-Loop

In the following sections, we provide a thorough case study that involves actor roles in our simplified manufacturing workstation, to illustrate instances of vulnerability-management cooperation in a common industrial-production environment, followed by experiments to evaluate our immersive-analysis framework. In the experiments section, we retrieve CPPS vulnerability reports from cross-linked online vulnerability-repositories to analyse existing vulnerability instances, and produce a qualitative evaluation at CPPS-asset vulnerability level. We also reveal experiment results from our streamlined approach that involves stakeholders to quantitatively assess vulnerability scores. These scores provide statistical information for practical attack-graph generation, which delivers valuable information for security management.

5.1 Case Study of Human-Induced Vulnerability Management

We interviewed industrial-production professionals and operators of a vehicles manufacturing company, to collect information about manufacturing networks structure. This step ensures that the topological factors and other settings of the network structure proposed in this case study could reflect an actual scenario of industrial manufacturing processes. Our proposed model contains around 700 components, 1000 both topological and functional dependencies, as well as around 100 data-flows exchanged across network applications and around 90 involved actors. In this study, we report a simplified account of the manufacturing structure as a reference model that focuses on key functionalities, to emphasise some key functional connections and data-flows, which are illustrated in Fig. 4. Different layered blocks illustrate interdependencies across CPPS smart manufacturing networks, namely cyber-layer, control-layer, and physical-layer. Each layer incorporates different functional sections or zones of CPPS networks. There are four types of connections, namely network connections in blue lines, physical connections in brown lines, data connections in dashed blue lines, and user interaction in dashed orange lines.

We demonstrate how to enhance CPPS security through cross-organisation cooperation involving multilayered-CPPS stakeholders. Usually, groups of application-specific staff are collaborating to complete the production process, while having separate roles. For a simplified smart manufacturing that contains only one CNC machine, one robot, and one conveyor, 63 user-roles are needed to attend the whole process, as illustrated in Fig. 4. These 63 technical personnel

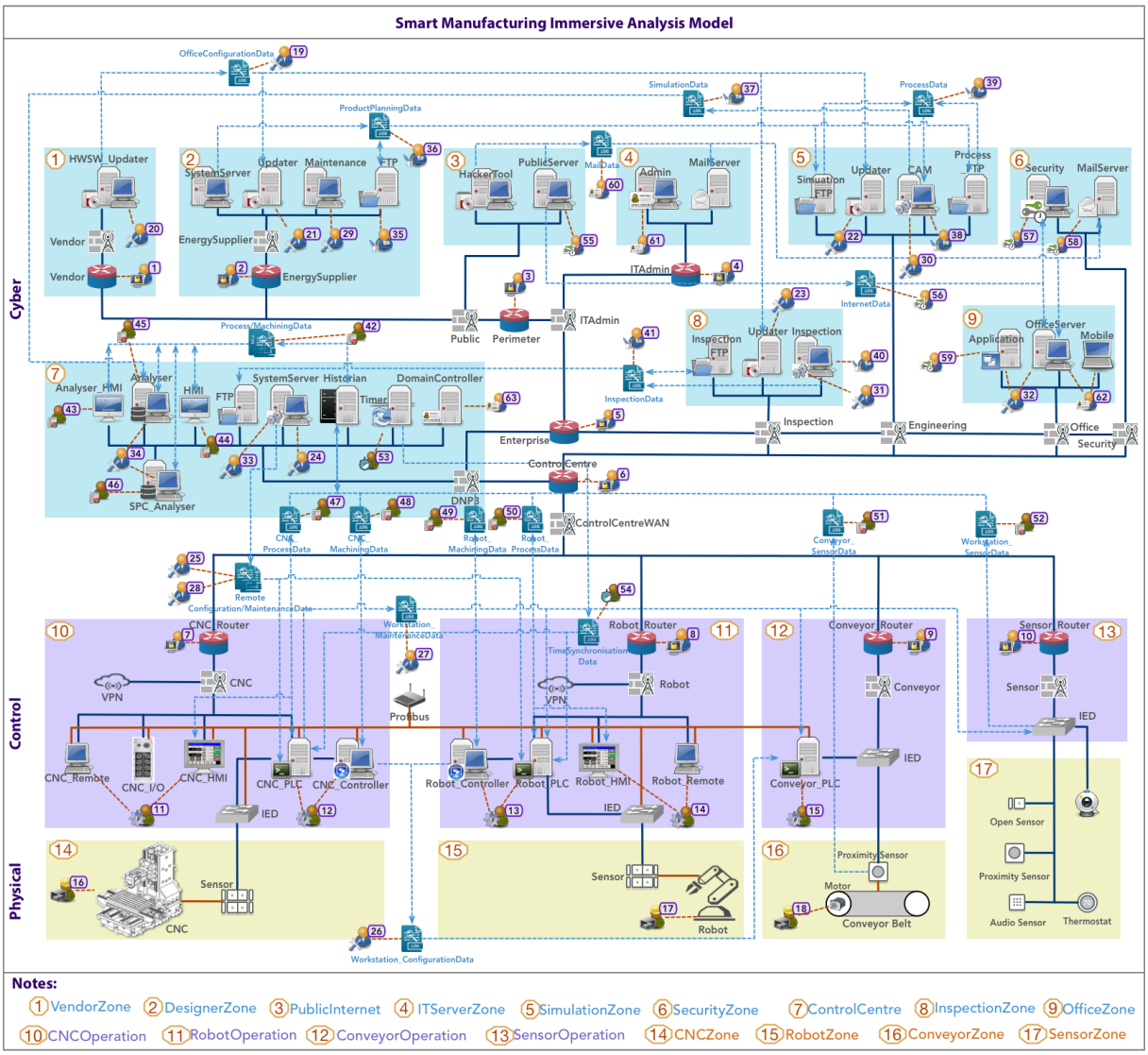

Fig. 4. Manufacturing Network Reference Model

include 10 network-administrators (labelled 1 to 10 in the figure), 5 CNC or PLC programmers (labeled 11 to 15), 3 application-engineers (labelled 16 to 18), 8 software-administrators (labelled 19 to 26), 8 system-administrators (labelled 27 to 34), 7 manufacturing engineers (labelled 35 to 41), 10 data engineers or operators(labelled 42 to 52), 2 operation managers (labelled 53 to 54), and 9 security engineers or domain-control officers (labelled 55 to 63). Here each staff's role refers to one user account, which might be owned by the same user or

different users. Therefore, access-control management is vital to ensure different users have different privileges, in order to prevent privilege escalation threats.

The physical layer contains a CNCZone (or CNC used to automate machining tools), a RobotZone (to program robots performing specific tasks in a production process), a ConveyorZone (to route material across machineries) and distributed SensorZones (to capture various measurements used to achieve basic production purposes). On each manufacturing equipment such as a CNC machine or a Robot, sensors are embedded to collect machining data. While some specific sensors and cameras are also distributed across the shop floor and used by workstations to collect motion and environmental data. In the physical layer, application-engineers are needed to ensure basic workstation maintenance.

In the control layer, typically both CNC machines and industrial Robots consist of drives, Numerical Control or NC kernels, remote input and out (I/O) and personal computer (PC) based controllers. These equipments are connected with each other, and can be reached through PLCs through an internal communication bus such as the process field bus Profibus. Normally CNC and PLC controllers communicate through a master-slave mechanism, namely a master device that initiates queries, and slave devices that respond with requested data to complete transactions. In our scenario, the CNC_PLC is selected as the master device. Meanwhile, CNC_PLC, Robot_PLC and some other intelligent devices are connected to routers through a local communication network such as Modbus. However, CNC_Controller, PLC_Controller, and intelligent electronic devices (IEDs) that are directly connected to the physical equipments, are usually not directly joined in the control network. Certain user-roles are needed optimise the efficiency and to monitor the security of the physical process operations, including network-administrators for local area network (LAN) administration, software-administrators and system-administrators for workstation system configuration update, operation-manager for synchronisation maintenance, CNC-operator and Robot-operator for remote machining and monitoring the process on human-machine interfaces (HMIs), and PLC-programmers that code optimal production-flow operational instructions to field devices.

The cyber layer basically contains a ControlCentre for machining-process monitoring, a SimulationZone for model simulation, an InspectionZone for production inspection, an OfficeZone for operations management, an ITAdminZone for network domain administration, a DesignerZone for model design, a VendorZone for software support, and a SecurityZone for security operation. Generally, CAD-designers in the DesignerZone create CAD files and transfer them as ProductPlanningData to the SimulationZone. CAM-engineers receive the CAD files and conduct simulations, and further creates G-code or M-code files (i.e programming instructions that tell machines what to do) out of the Process-Data. Data-engineers query the SimulationDatastore from the ControlCentre, and then further divide queried Process/MachiningData into CNCMachining-Data and CNCProcessData to CNC machines for production purpose, as well as RobotMachiningData and RobotProcessData for process-control purpose. Meanwhile, ProcessData and MachiningData from the workstation are sent back by

CNC_PLC and CNC_Controller separately, under the request of data-operators in ControlCentre. These data would be under quality-inspection by inspection-engineers and statistic process control (SPC) data-engineers. In addition, time synchronisation is a vital part of production processes, and is taken care of by operation-managers. Besides application specific technicians, administrators and other operators are in charge of security maintenance of the all system. For example, software-administrators are mainly responsible for update or maintenance of embedded application software. System-administrators are mainly responsible for configurations or maintenance of embedded OS. Network-administrators are responsible for domain-control administrative tasks of local area networks (LANs). Domain-control operators are responsible for corresponding authorisation such as mail service and user-management. Security engineers or officers are responsible for daily security monitoring such as penetration testing.

Dynamic vulnerability-management involves cooperation between IT-operators, application-specific engineers and managers throughout production processes. Different working groups contribute their professional expertise to vulnerability-management from operational, management and executive levels.

5.2 Experiments on CPPS Vulnerability Patterns and Attack Patterns

The following sessions include three parts, namely a) vulnerability pattern and threat pattern analysis based on statistical data retrieved from multiple online vulnerability repositories, b) vulnerability quantification involving stakeholders' knowledge, and c) attach pattern analysis with prerequisite vulnerability setting.

a) Statistical CPPS Vulnerability Patterns Analysis

We demonstrate our system-induced vulnerability analysis method through an analysis of Human-Machine Interface (HMI) in Supervisory Control and Data Acquisition (SCADA) control and monitoring system. HMI is a key CPPS asset where programmers transmit optimal production-flow operational instructions to actuators to implement these changes onto field devices, or where engineers monitor process/machining data.

By querying the vulnerability repository National Vulnerability Database (NVD) that discloses Common Vulnerability and Exposures (CVE) reports, we obtained 141 reported vulnerability instances related to HMI. According to the documentation of CVSS, we mapped each CVSS version 2 base-score of HMI vulnerability instances to the qualitative severity rating scale, and concluded that more than half (73 out of 141) of these vulnerability instances are evaluated as *High* severity of vulnerability instances, as illustrated in Part (a) of Fig. 5. In order to investigate on the corresponding threat patterns, we also correlated CVE reports against the threat categorisation provided in www.cvedetails.com, and found out the most typical threats that target HMIs are *Code Execution* and *Overflow*, as shown in Part(b) of Fig. 5. Both threat patterns and vulnerability patterns can provide statistic likelihood value for attack-occurrence and attack success rates.

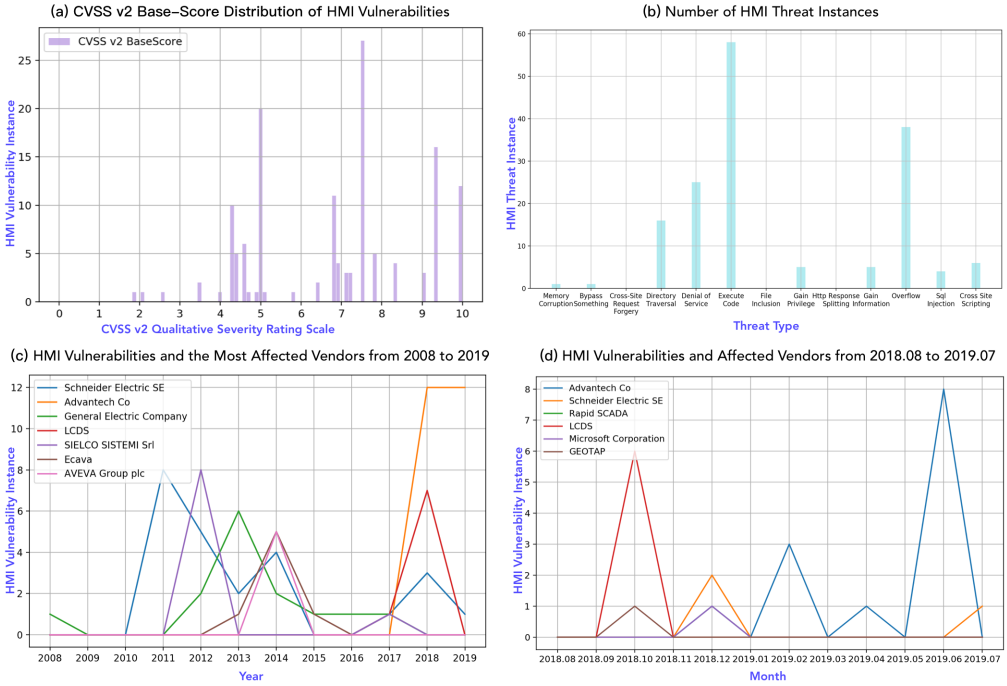

Fig. 5. Vulnerability Instance Amounts of Affect Human-Machine-Interface Vendors

Considering CPPS asset configurations, we retrieved detailed component information for each reported vulnerability through crosschecking our CVE findings with Common Platform Enumeration (CPE) repository. Based on CPE naming specification, we further acquire detailed information of vulnerable components, such as component type, vendor, component name, component version, etc. We sum up all the vulnerable instances and get 276 vulnerable application software instances, 9 vulnerable operating system instances, and 5 vulnerable hardware instances in HMIs. Based on retrieved results, we build a specific dictionary to store HMI vulnerable component versions. We further expand the dictionary by relating each CVE report to the disclosed time and the affected vendors. The most affected HMI vendors and the amount of reported vulnerability instances per year from 2008 till 2019 for each vendor are illustrated in Part(c) of Fig. 5. We also show the short-term situation from Aug 2018 to Jul 2019 in Part(d) of Fig. 5. The vendor-related information is valuable when an operation manager needs to update configuration settings or to choose from products from different vendors.

b) Quantitative Assessment Involving Stakeholders

Each vulnerability instance is associated to one or more specific properties, which could be measured across a range of values using vulnerability property-

related metrics by security engineers. We adopt CVSS Version 2 to calculate scores for vulnerability instances. For instance, the exploitability property of a vulnerability is based on the possibility, difficulty and complexity of exploiting a vulnerable asset, and could be quantitatively evaluated by security operators using inputs from software administrators, application engineer and network administrator. The impact property of a vulnerability measures the consequences resulting from exploiting the vulnerability, which could cause losses in CIA-triad of the corresponding asset. The impact property of vulnerability could be evaluated by application engineers, as well as network and software administrators.

SecurityOperator, *SecurityManager* and *OperationManager* are also involved to measure temporal or environmental metrics. For example, vulnerability instance *CVE-2015-0997* has a CVSS v2.0 Base-Score of 3.3. According to the v2 documentation[5], relevant actors are involved in the analysis process, as shown in Fig. 6. Taken into consideration of the given temporal and environmental measurements, a final CVSS v2 score of 3.2 is assigned. Application specialists can also provide valuable information such as potential costs for system recovering once success attacks happen.

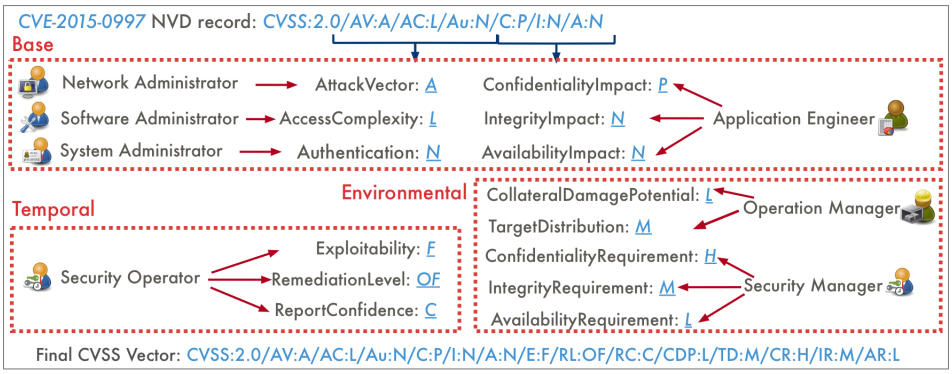

Fig. 6. Quantitative Vulnerability Assessment Involving Stakeholders

c) CPPS Attack-Graph Based on Prerequisite Vulnerability Setting Based on the manufacturing network reference model in and organisational staff management in Fig. 4, as well as the statistical vulnerability patterns and threat patterns for each asset collected beforehand, we generate attack graphs for different attack instances. In current stage, our model is built using securiCAD[6] which is based on probability Bayesian Network [9] to simulate attack-occurrence and propagation instances. Vulnerability may impact multiple asset instances, and vulnerable assets can be impacted by more than one vulnerability instance.

[5] https://www.first.org/cvss/v2/guide
[6] https://www.foreseeti.com/securicad/

In CPPS, hardware, software, OS and network assets are assembled and used in different ways within CPPS fabrics, which might create various binaries with potential backdoors. Meanwhile, human errors or mistakes could expose new vulnerabilities in both access-control and transferred data. The propagation of an attack is reasoned by statements that check asset dependencies. Here we illustrate an attack scenario of DesignerZone network being compromised, and how this successful attack can trigger SimulationZone network to be compromised as well, then finally end up compromising the ControlServer in ControlCentre, as discussed before in in Fig. 1. The generated attack graph in Fig. 7 shows how the vulnerabilities in CPPS and how compromised user-accounts contribute to the attack propagation from DesignerZone to the ControlCentre. The thickness of arrows represent the likelihood values of this attack success rate, which is given by the results in previous experiments.

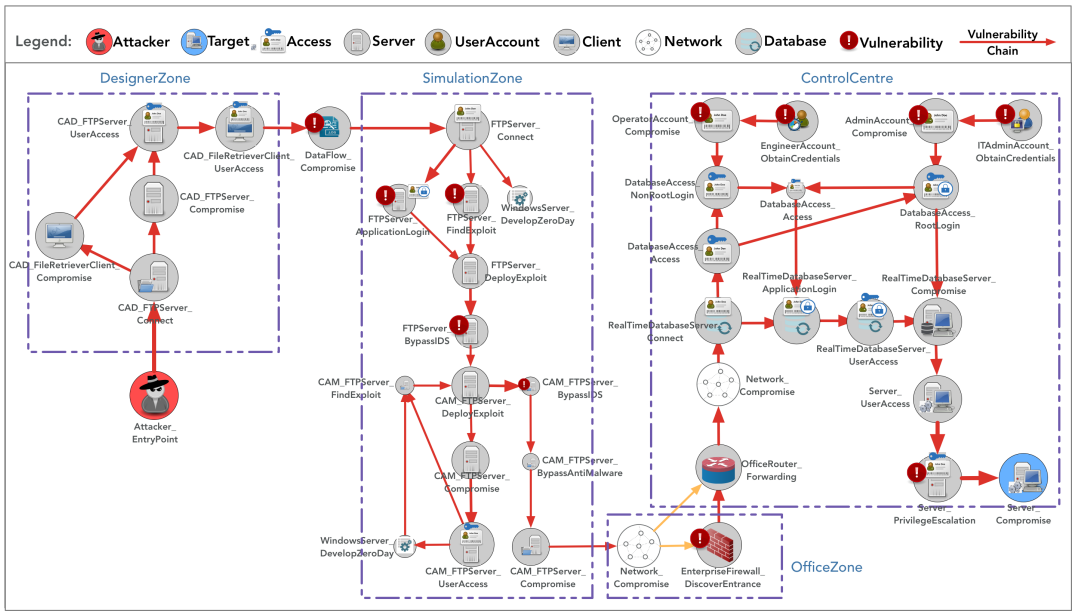

Fig. 7. Attack Scenario Illustration in Manufacturing Network

6 Conclusion and Future Works

CPPS networks generate a growing traffic of data, which is highly sensitive to cyberthreat vulnerabilities. Understanding and pinpointing vulnerabilities in such networks are difficult tasks, yet vital for cybersecurity purposes of production processes. In this paper, we proposed a framework to map CPPS components

against Threat, Vulnerability and Attack instances, which can reduce the effect of their occurrences. Threats are triggered through vulnerability exploits from malicious agents' behaviour that could result into a cyber-attack instance which can lead to disruption in production infrastructure operations. Our proposed framework also bridges the connection from CPPS system to different actors in the production process through collaborations between security and operation personnel stakeholders, to enable situation awareness that supports vulnerability intelligence, in order to stay on top of potential threats. We also provided a detailed evaluation of our framework through a case study through a manufacturing network reference model, which demonstrates organisational access control and cooperation. We also illustrate the processes of vulnerability information retrieval and analysis through experimental analysis, to show how we combine statistical pattern analysis with quantitative assessment and attack-graph generation for practical security assessment. Further on, we plan to extend our framework into an automatic vulnerability assessment system for CPPSs as part of our future work.

References

1. Wu, D., Ren, A., Zhang, W., Fan, F., Liu, P., Fu, X. and Terpenny, J. Cybersecurity for digital manufacturing. Journal of manufacturing systems, 48, pp.3-12, 2018.
2. Mohurle, Savita, and Manisha Patil. "A brief study of wannacry threat: Ransomware attack 2017." International Journal of Advanced Research in Computer Science 8, no. 5, 2017.
3. M. Välja, R. Lagerström, U. Franke, and G. Ericsson, "A framework for automatic it architecture modeling: applying truth discovery," 2018.
4. E. Rahm and P. A. Bernstein, "A survey of approaches to automatic schema matching," the VLDB Journal, vol. 10, no. 4, pp. 334-350, 2001.
5. A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-physical systems security-a survey," IEEE Internet of Things Journal, vol. 4, no. 6, pp. 1802-1831, 2017.
6. H. Kure, S. Islam, and M. Razzaque, "An integrated cyber security risk management approach for a cyber-physical system," Applied Sciences, vol. 8, no. 6, p. 898, 2018.
7. Jiang, Y., Jeusfeld, M., Atif, Y., Ding, J., Brax, C., & Nero, E. A Language and Repository for Cyber Security of Smart Grids. In 2018 IEEE 22nd International Enterprise Distributed Object Computing Conference (EDOC), pp. 164-170, IEEE, 2018.
8. Hafner, M., Breu, R., Agreiter, B. and Nowak, A. SECTET: an extensible framework for the realization of secure inter-organizational workflows. Internet Research, 16(5), pp.491-506, 2006.
9. P. Johnson, A. Vernotte, D. Gorton, M. Ekstedt, and R. Lagerström, "Quantitative information security risk estimation using probabilistic attack graphs", in International Workshop on Risk Assessment and Risk-driven Testing. Springer, pp. 37-52, 2016.
10. Elhabashy, A.E., Wells, L.J., Camelio, J.A. and Woodall, W.H. A cyber-physical attack taxonomy for production systems: a quality control perspective. Journal of Intelligent Manufacturing, pp.1-16, 2018.