Comprehensive Security Solution for an Industry 4.0 Garment Manufacturing System

Kalhara H. H. D
dept.Computer Systems Engineering
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
dasunpriyakalhara@gmail.com

Udara De Alwis
dept.Computer Systems Engineering
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
utdealwis@gmail.com

Anuka Jinadasa
dept.Computer Systems Engineering
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
anukajindasa@yahoo.com

Dinuwan Randunu
dept.Computer Systems Engineering Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
ridmadinu1@gmail.com

Wellalage Sasini Nuwanthika
dept.Computer Systems Engineering
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
sasini.w@sliit.lk

Pradeep Abeygunawardhana
dept.Computer Systems Engineering Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
pradeep.a@sliit.lk

***Abstract*— Industry 4.0 digital transformation towards Internet of Things (IoT) focuses on productivity rather than security, therefore often face cybersecurity challenges. To overcome such, a centralized security solution focusing on four major areas 1) standardization, 2) security configurations with update management, 3) authentication and physical access control and 4) intrusion detection was applied to a Cyber Physical System (CPS) based garment manufacturing system to overcome the security gap. Security policies based on ISO 27001:2013 and IEC 62443 security standards were applied to the security configuration management system in the design stage to achieve 'security by design' concept. Security and audit configurations were applied using ansible for the automated tool that can be customized based on requirements. A Raspberry Pi 4 was utilized as IDS (Intrusion Detection System) to evaluate performance. The physical access control prevents unauthorized access and access monitoring was done using logs. The proposed system enhances security, reaching for a cost-effective, efficient, reusable solution, and provides a comprehensive security solution for potential challenges of current and future smart manufacturing. The system is secured in terms of strategy, design, and operations rather than securing the system after deployment.**

Keywords—Industry 4.0, Internet of Things (IoT), cybersecurity, Cyber Physical System (CPS), Intrusion Detection System (IDS)

I. INTRODUCTION

Novel changes in Industry 4.0 lead to a passion for manufacturing plants with productivity, customization features, flexibility, operational efficiency [1] and SAM/SMV (Standard Allowed Minute/Standard Minute Value) rather than security. The integration of complex smart manufacturing technologies lead to increased intercommunication and data density, thus massively expand the scope of attacks pointing at industrial espionage and sabotage [1], because Industrial 4.0 are implemented targeting the functionality than security [2].

CPS are used to gain higher productivity in manufacturing [3], and to usher innovation due to their potential to integrate technology from different sectors for the implementation of real world processes [1][2]. Manufacturing automation is becoming a part of critical infrastructure in the present and future context. CPS is designed and implemented to be easily extendable and scalable as the structure includes heterogeneous communication technologies. The integration of IoT devices combined with various technologies is a complex task and implementing security for those systems is more complex task. Therefore, cybersecurity has evolved into a major concern.

Objective is to design and implement an automated system for garment manufacturing system focusing on four major cyber security aspects for garment manufacturing systems including:

- Frameworks and standardization
- Centralized security configurations with update management
- Authentication and Physical Access Control
- Intrusion detection

A comprehensive, cost effective and efficient security solution is proposed with a centralized security approach to overcome the potential challenges of the smart system.

The reminder of the paper presents:

- Literature based analysis of the four aspects mentioned above in the context of integration of IoT, CPS and Network devices and their challenges
- The methodology and approach for the centralized security architecture for the system.
- Results of the overall research, discussion, and future work
- Finally, the conclusion of the overall research project.

II. CURRENT SECURITY THRATS AND SECURITY SOLUTIONS IN INDUSTRY 4.0

The combination of novel technologies has enabled the transformation of conventional to intelligent manufacturing [3]. Cyber-attacks cause data breaches, system collapse and other disasters [4]. CPS is extendable and scalable, which leads to technical issues, such as system verifications, frequent software updates, network and data interoperability, synchronization, privacy, and security issues [5]. CPS must

be operated in a well-defined, predictable way with safety as a major priority. Most literature primarily focus on functionality of industrial 4.0 systems and security has been considered as a subordinate characteristic.[6] Therefore, the threat surface for industrial 4.0 automation systems are broad which affects to our research.

Security is the major issue, out of the CPS challenges listed in figure 1. and mostly originated in cyberspace but impacted on the physical system. Security analysis frameworks to develop security qualification matrices to measure security in CPS can be found under past literature [7] which is helpful to express the quantitative level of security.

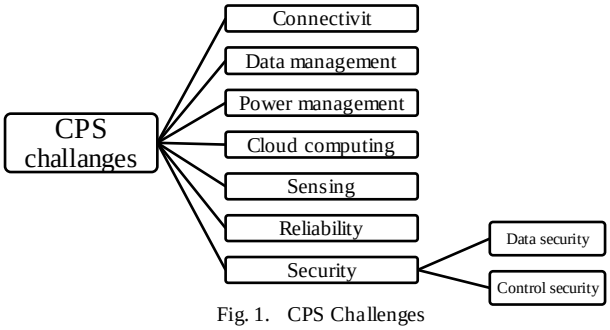

Fig. 1. CPS Challenges

CPS Vulnerabilities can occur due to poor system design and lack of security rules. Protection against threats and vulnerabilities is necessary for IoT to achieve its full capabilities [2]. Following are some utmost cases of CPS incidents. In 2010, a worm known as Stuxnet attacked nuclear facilities in Iran [8], An externally originated virus attacked Saudi Aramco Oil corporation by infecting 30,000 workstations [9], Mirai Botnet infected IoT devices and controlled them as a Zombie network for malicious activities [10]. Further, literature analysis is presented below according to major components of the research.

A. IoT Security Frameworks

The absence of effectual standards and regulations and weak governance can lead to downward trend in IoT security. Nevertheless, various information security frameworks exist to cover IoT concepts and deployments in various plumbs, classified according to the area of concern [6]. Research is conducted to identify current security frameworks, but there are few initiatives that have been developed to be used in CPS. Systematic reviews are done according to sources such as secure communication, embedded firewall & intrusion detection security, Authentication & Authorization controls, integration with security management systems. In an occurrence of security non-compliance, the security framework, must address security risk assessment and controls to mitigate the risk while abiding with standards and needs [11].

Cisco's proposed security implications for IoT/M2M (Machine to Machine) constructions are extensive, necessitating the deconstruction of a workable IoT/M2M security framework that may be used to execute security in IoT contexts. ISA (International Society of Automation) /IEC (International Electrotechnical Commission) 62443 cybersecurity standards for Industrial Automation and Control Systems (IACS) compliance can be obtained by Floodgate Security Framework. Constrained Application Protocol (CoAP) Frameworks to handle security and trust issues of IoT environments, OSCAR security framework, explores a novel approach to the problem of E2E (End to End) security in IoT [11]. There is literature on aiming to minimize development and testing time in industrial environment using a framework for rapid integration.

B. Security Configurations

Open Web Application Security Project (OWASP) [12] mentions that most manufacturers ship IoT devices with hardcoded passwords or insecure default settings. Due to insecure configurations devices lack authentication [13] and encryption leads to several security threats [14] including authentication attacks, brute force attacks, Man in the Middle attacks, and eavesdropping [15]. Due to large numbers of CPS and IoT devices mitigating security vulnerabilities manually reduces the efficiency of the production system, hence a configuration management system can be used to automate security configurations of the devices [16]. Chung's On-demand security configuration for IoT devices [17] transmits using device images. Problem with this method is regenerating device images is time-consuming and complex. Configuration management tools provide the below advantages to overcome these challenges. Configuration management tools such as Chef [18], Puppet [19], Ansible [20], and SaltStack [21] provide customization, reusability, scalability, and portability [22] to increase the efficiency of the system. Considering previously mentioned configuration management tools, Chef and Puppet rely on an agent on the client devices to establish a connection using mutual authentication or Secure Sockets Layer (SSL) protocol. However, Ansible and SaltStack do not need agents to stablish connection between client devices and use Secure Shell Protocol (SSH). The tool should be agentless to save storage and connections should rely on lightweight ciphers [23] to save processing power and reduce energy consumption.

C. Authentication and Access Control

CPS, despite its numerous advantages, is exposed to a variety of physical security and cyber security threats, such as side channel attacks due to its diverse nature, reliance on private and sensitive data, and large-scale implementation [24]. On most CNC (Computer Numerical Control) units, whether the HMI (Human Machine Interface) has soft keys, key switches, or conventional keyboards, these units can be exploit because they are open to everyone. On some models, only the physical key is used to control the physical access. There are no access monitoring solutions in CPS devices [25]. As a result, intentional or accidental exposures of these systems might have disastrous consequences, forcing the implementation of comprehensive security measures. Considering these requirements, allowing these physical security systems to monitor a person's every activity must be accompanied with the presumption that this information will be used only for the purpose intended and will be secured against malicious use or unauthorized access, as well as to prevent network cyberattacks and to build strengthen security [26].

D. Intrusion Detection

Even though the benefits outweigh the disadvantages, Industry 4.0 security and privacy challenges cannot be ignored. Even with proper security configurations it is crucial to have a proper Firewall and Intrusion detection and prevention system. Whitepaper written by Michael Brennan,

SysAdmin, Audit, Network, and Security (SANS) Institute, explains organizations with little to no network monitoring, opting for an expensive solution is not plausible. This paper focuses on the need for an Intrusion Detection System (IDS) for companies that cannot afford more quality commercial Intrusion Detection Systems available in the market [27]. In current market entry level network-based IDS solutions such as McAfee IDS will cost more than 10 000 USD [28].

Many research projects have been done to evaluate open-source Intrusion Detection Systems on single board computers specially on Raspberry Pi 2. These tests emphasize on the capability of Raspberry Pi 2 of running IDS and can it handle all the traffic without loss. Raspberry Pi 2 was able to run IDS and detect attacks to handle a considerable number of packets. However, these tests were done with old hardware but still showed the potential of single board computers [29][30]. Our focus is to integrate Raspberry Pi 4b as an IDS to a medium size network that contains IoT devices and evaluate the effectiveness.

III. Methodology

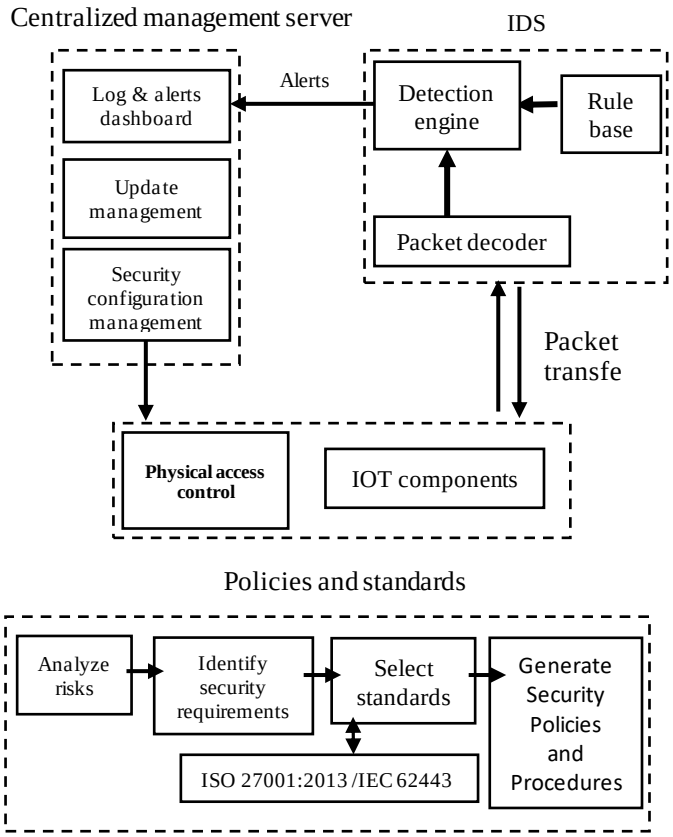

Fig. 2. Overall System Diagram

According to the literature analysis, security architectures such as isolated execution environment, trusted execution environment, and layer-based architecture have been proposed. Nevertheless, our approach for a centralized security architecture will increase efficiency in security management according to the details below. Figure 2 conveys how the aspects of the proposed solution are integrated.

A. Security Standardization and Policies

Security standards are chosen according to the results of Octave risk assessment, and specific standards are evaluated. The comparison of chosen security standards is done according to the security requirements of the research. Identification of frameworks, guidelines and procedures are subjected according to standardization. There are few IoT security frameworks in use and in various stages of development. European Telecommunications Standards Institute (ETSI) EN 303 645, IoT Security Compliance Framework, OWASP IoT Security Verification Standard (ISVS), European Network and Information Security Agency (ENISA) standard and National Institute of Standards and Technology (NIST) standard gives IoT security standards for different components. According to project requirements for the manufacturing system, two standards were chosen to create policies. International Organization for Standardization (ISO) 27001:2013 provides requirements for establishing, implementing, maintaining, and continually improving an Information Security Management System (ISMS) and IEC 62443 developed to secure industrial automation and control systems (IACS) throughout their lifecycle. Next step is to identify and document important policies to use them in implementation. IoT security standards should be chosen carefully according to the requirements, as there are many IoT security standards addressing different areas. The best solution is to customize a framework based on proper standardization according to the requirements of the system.

B. Centralized Security Configuration Management

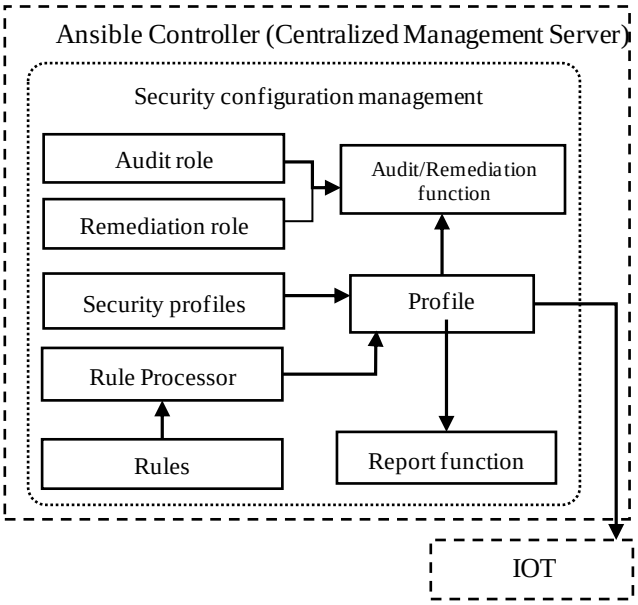

Fig. 3. Security Configuration Management Tool

As shown in Figure 3 the security configuration management tool uses a Linux server named ‘ansible controller’ to provide centralized security management. The security configuration management tool is installed in the ansible controller. The tool is written in Python and utilizes Ansible to deliver configurations because Ansible has a rich variety of modules and roles [20] that can be utilized for security configuration. The tool has the following components. Previously created security policies are converted into rules and these rules are stored in a file. The file contains information to help the system administrators to understand about rules applied on the devices. Due to the variety of device models, some rules may not applicable to every device. Therefore, a security profile is

used to select the necessary rules to be applied. These profiles can be customized based on the level of security required.

Rules are converted into ansible tasks, these tasks are included in the audit role and the remediation role. The audit role used for audit current security configurations on devices and send results back to the ansible controller. Then the tool processes the received results and creates detailed reports for each device. The remediation role used for remediating security configurations on devices.

C. Authentication and Access Control

The Access Control System is used to identify, authenticate, and authorize to access into the premises or devices, ensure the system's security and offering comprehensive protection. Authentication is provided secure access to services and monitor and filter user behavior on the system to avoid unauthorized access and venomous network assaults. To achieve the main goal, first background research was done to acquire information concerning authentication and access monitoring. It provides security by allowing for flexible control about who is allowed to access.

The Physical Security System was implemented using the Arduino platform. An Arduino board was used with a fingerprint sensor module since it is dependable biometric. Arduino IDE (Integrated Development Environment) was used for the write functions and C and C++ languages were used. First, the fingerprints were enrolled and stored in the sensor. In the verification process, the fingerprint was verified and collected logs locally and sent them to the local server or cloud-based server. At any time, administrators can search and visualize, analyze the access logs, error logs, and generate the report.

D. Intrusion Detection

The IDS sensor was based on Raspberry Pi 4B 4GB RAM version with Snort IDS software, initially Raspberry OS was installed and configured as a wireless access point by using hostapd and dnsmasq, although the Raspberry Pi 4B inbuilt Wi-Fi is on the weak side it is more than adequate for testing purposes.

Next, step was to install and configure Snort IDS software, Barnyard2 for store alerts in a Database and Pulledpork module to automate the Snort updates. When it comes to Snort rules, a combination of community & registered rule sets provided by Snort were used.

As the last step, iptables firewall was configured and automated the new sensor deployment process with security configurations using Ansible playbooks.

All IDS sensors will collect logs locally and send them to a central management server, this can be achieved by using a local server or a cloud-based server, in this case a local Linux server running elastic stack, where administrators can search, visualize, and analyze alerts in real time. As shown in Figure 2, IDS allows all IOT devices to connect to the Raspberry Pi access point and IDS will perform real time monitoring in the network, if there are any malicious packets detected IDS will act according to the rules.

IV. RESULTS

The research is conducted in the perspective of centralized management, unlike other research which uses a decentralized or layered approach. IoT devices are more resilient to attacks because of configurations applied by the security configuration management tool. These configurations reduce threats mentioned in literature. The tool enforces security policies as technical controls in the devices.

Performed testbed scenarios have ten IoT systems and the ansible controller connected through the same network. Ten IoT systems include five devices run on Raspberry Pi Operating System (OS), two devices run Robot Operating System 2 (ROS), and three devices run on Jetson Linux. SSH access and administrative access for IoT systems are given to the security configuration management tool.

TABLE. I. PERFORMED TESTBED SCENARIOS

Scenario	Starting Device Configuration	Security Configuration Method
1	10 insecure default installations	The security configuration management tool
2	1 secure Raspberry Pi OS + IDS deployed 2 secure Raspberry Pi OS 7 insecure default installations	
3	10 insecure default installations	Shell scripts

The results from performed testbed scenarios in TABLE I have shown that remediation using the tool takes t_α amount of time per device in the system that has N numbers of IoT devices. The audit takes t_β amount of time per device. The total downtime of the system is calculated as,

$$Total\ down\ time = t_\alpha + t_\beta$$

The tool uses a centralized management server to simultaneously remediate all the devices. Based on scenario 3, time loss due to human error is ω. the total down time is calculated as follow,

$$Total\ down\ time = N(t_\alpha + t_\beta) + \ \omega$$

TABLE. II. RESULTS FROM PERFORMED TESTBED SCENARIOS

Scenario	t_α	t_β	Total Downtime
1	20 minutes	10 minutes	30 minutes
2	20 minutes	10 minutes	30 minutes
3	25 minutes	15 minutes	400 minutes + 30 minutes

The mentioned times in TABLE II may change depending on network and device performance. In addition to results in TABLE II following results were identified from scenario 1 and 2. The tool was able to identify the securely configured devices and apply changes on the insecure devices without affecting the preconfigured secure devices. The results have shown that the tool skipped over already configured rules and only changed insecure configurations founded in the audit. Running remediation multiple times on the devices has not affected the services used by the IDS. In scenario 3, extra 30 minutes were spent on reverting uncompleted configurations that happened due to human error.

The efficiency of centralized system compared to decentralized system can be shown as follow,

$$Efficiency = \left(\frac{N(t_\alpha + t_\beta) + \omega}{t_\alpha + t_\beta}\right) * 100\%$$

The open-source libraries reduced implementation cost of the tool. The update management system provides trusted packages for the connected devices through the local network.

Authentication and access control is required to provide comprehensive data control and permissions management across all organizations that collaborate along the production life cycle because of the variety of sites of attacks. A fingerprint smart door lock lowers the risk of break-ins by replacing traditional locks with keys that can be stolen or misplaced. A physical smart lock that fuses fingerprint technologies for better identification accuracy and security. It is a reliable solution for all CPS and IoT devices. The level of confidence indicates how well the current fingerprint matches that of the sensor database. So, this sensor database can store 127 fingerprints and it has a false acceptance rate of less than 0.001%, making it quite safe. This physical smart lock system has an alert function that will notify user if there is a failed access attempt and will monitor access and error records.

Considering IDS, the objective was to find the balance between performance/connectivity and security. Kali Linux system with TCPReplay 4.3.3 was utilized to evaluate the performance of IDS, to represent a wide range of network behavior multiple types of packet capture files from internet sources were used [31]. These test scenarios were conducted using 3000, 6000 and 9000 lines of rules and packet files were categorized according to packet size. Raspberry Pi resource usage was continuously monitored using a python script.

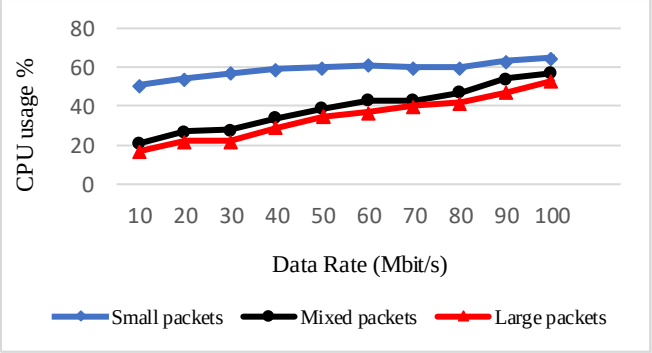

Fig. 4. Recource usage

As shown in Figure 3, larger data packets cause CPU usage to gradually increase, when it comes to smaller packets CPU usage reaches above 50% even in 10 Mbit/s speeds, the main cause for this is smaller packet sized need more higher packets per second (PPS) rate to reach the same throughput as larger packets, regardless of speed. High PPS rate generate more CPU interrupts causes CPU usage to up. Moreover, tests reveal the number of rules have no considerable impact on CPU usage, however rules directly impact the RAM usage, the main cause for this is Snort IDS loads all the rules into the memory. Even with 9000 lines of rules Snort IDS demands did not reach 100% RAM usage and the packet loss were recoded as 0.05%.

V. Discussion and Future Work

The proposed cyber security solution provides small-scale or medium-scale manufacturers with four aspects to achieve security in connected IoT devices. If left unchecked, vulnerabilities in IoT devices could pose a risk to the entire system.

According to Statista, there are around 21.5 billion interconnected devices in the world in 2021. IoT security standards should be selected carefully according to the requirements, as there are many IoT security standards addressing different areas. Various information security frameworks exist to cover IoT concepts and deployments in various verticals, classified according to the area of concern. The chosen standards and security frameworks should be aligned with organization's strategies, vision and mission or security requirements of the system. Therefore, the best solution is to customize a framework according to the requirements of the system to gain hardware root of trust and privacy by design.

The security configuration management tool narrows down the knowledge gap between system administrators and security experts. As shown in results centralized architecture of the tool reduces total down time of the system for remediation and reduces workload of the system administrators compared to decentralized management system. The tool reduces system down time due to human error while remediation. The ansible controller must be secured to avoid single point of failure. Because it has administrative access to all the IoT devices.

Rule management is a major part of Snort IDS, loading a high number of rules is not recommended since it could influence high resource usage and packet drops. Our test results show Raspberry Pi 4 can handle 6000 to 9000 lines of rules without any significant performance compromises, however, weak Wi-Fi range in Raspberry Pi 4b could cause bottlenecks in the network. This can be easily remedied by using an external Wi-Fi adaptor or range extender.

Hereafter, further development of this solution towards a standard security framework is planned, along with a cloud-based management server deployment.

VI. Conclusion

Due to the high number of interconnected devices, there are plenty of IoT security standards and various information security frameworks exist to cover IoT concepts and deployments in various verticals for different areas, therefore, choosing the correct IoT security standards according to project requirements and organizational strategies is a challenge. The best solution is to customize a framework according to the requirements of the system to gain hardware root of trust and privacy by design.

Even though security should be prioritized, organizations neglect security due to budget constraints. The security configuration management tool and proposed IDS are ideal cost-effective solution for small to medium Industry 4.0 environments. The mentioned scenarios prove that the security configuration management tool can use customized security profiles based on organization requirements. The tool

is ideal for growing Industry 4.0 manufacturers because of its scalability, reusability, and centralized security management which increase the efficiency of security audits and remediations. Considering the IDS, test results show Raspberry Pi 4 can run Snort IDS and act as a Wi-Fi access point without any compromises. When it comes to physical access control proposed solution is a sustainable option for the purposes of identification and authentication. For improved security, the system may be used to replace existing systems. An added feature is the report generation to analyze. The security solution is ideal for future direction of security in smart manufacturing.

VII. Acknowledgment

We wish to acknowledge the help provided by Dr. Asela Kulatunga, Head of Department of Manufacturing and industrial engineering, University of Peradeniya, Sri Lanka for giving us the opportunity to visit Peradeniya University to study CNC machine and robotic applications workflow which was a good initialization point for our research project.

We wish to show our appreciation for our external Supervisor Chartered Eng. P.A. Gamini De Alwis for guiding us in the correct direction, throughout our research project by providing experience the manufacturing field., Prof. Darshi De Saram for sharing knowledge and experience and providing comments to make our project a success.

References

[1] K. Zhou, Taigang Liu, and Lifeng Zhou, "Industry 4.0: Towards future industrial opportunities and challenges," in 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Zhangjiajie, China, Aug. 2015, pp. 2147–2152, doi: 10.1109/FSKD.2015.7382284.

[2] H. Xu, W. Yu, D. Griffith, and N. Golmie, "A Survey on Industrial Internet of Things: A Cyber-Physical Systems Perspective," IEEE Access, vol. 6, pp. 78238–78259, 2018, doi: 10.1109/ACCESS.2018.2884906.

[3] A. Bhattacharjee, S. Badsha, and S. Sengupta, "Blockchain-based Secure and Reliable Manufacturing System," IEEE Green Computing and Communications (GreenCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics), Rhodes, Greece, Nov. 2020, pp. 228–233. doi: 10.1109/iThings-GreenCom-CPSCom-SmartData-Cybermatics50389.2020.00052.

[4] H. Xu, W. Yu, D. Griffith, and N. Golmie, "A Survey on Industrial Internet of Things: A Cyber-Physical Systems Perspective," IEEE Access, vol. 6, pp. 78238–78259, 2018, doi: 10.1109/ACCESS.2018.2884906.

[5] N. Zainuddin, M. Daud, S. Ahmad, M. Maslizan, and S. A. L. Abdullah, "A Study on Privacy Issues in Internet of Things (IoT)," in 2021 IEEE 5th International Conference on Cryptography, Security and Privacy (CSP), Zhuhai, China, Jan. 2021, pp. 96–100. doi: 10.1109/CSP51677.2021.9357592.

[6]]N. Tuptuk and S. Hailes, "Security of smart manufacturing systems," Journal of Manufacturing Systems, vol. 47, pp. 93–106, Apr. 2018, doi: 10.1016/j.jmsy.2018.04.007.

[7] A. Aigner and A. Khelil, "A Security Qualification Matrix to Efficiently Measure Security in Cyber-Physical Systems," in *2020 32nd International Conference on Microelectronics (ICM)*, Aqaba, Jordan, Dec. 2020, pp. 1–4. doi: 10.1109/ICM50269.2020.9331797.

[8] Kamble, S., Gunasekaran, A. and Gawankar, S., 2020. Sustainable Industry 4.0 Framework: A Systematic Literature Review Identifying The Current Trends And Future Perspectives.

[9] Lins, Theo & Rabelo, Ricardo & Correia, Luiz & Sá Silva, Jorge. (2018). Industry 4.0 Retrofitting. 8 -15. 10.1109/SBESC.2018.00011. [6]Moktadir, M., Ali, S., Kusi-Sarpong, S. and Shaikh, M., 2019.

[10] N. Gupta, V. Naik and S. Sengupta, "A firewall for Internet of Things," *2017 9th International Conference on Communication Systems and Networks (COMSNETS)*, Bengaluru, India, 2017, pp. 411-412, doi: 10.1109/COMSNETS.2017.7945418.

[11] M. Irshad, "A Systematic Review of Information Security Frameworks in the Internet of Things (IoT)," in *2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, Sydney, Australia, Dec. 2016, pp. 1270–1275. doi: 10.1109/HPCC-SmartCity-DSS.2016.0180.

[12] "OWASP Internet of Things Project - OWASP." https://wiki.owasp.org/index.php/OWASP_Internet_of_Things_Project (accessed Jul. 21, 2021).

[13] M. El-hajj, M. Chamoun, A. Fadlallah, and A. Serhrouchni, "Analysis of authentication techniques in Internet of Things (IoT)," in 2017 1st Cyber Security in Networking Conference (CSNet), Rio de Janeiro, Oct. 2017, pp. 1–3. doi: 10.1109/CSNET.2017.8242006.

[14] A. C. Panchal, V. M. Khadse, and P. N. Mahalle, "Security Issues in IIoT: A Comprehensive Survey of Attacks on IIoT and Its Countermeasures," in 2018 IEEE Global Conference on Wireless Computing and Networking (GCWCN), Lonavala, India, Nov. 2018, pp. 124–130. doi: 10.1109/GCWCN.2018.8668630.

[15] M. Nawir, A. Amir, N. Yaakob, and O. B. Lynn, "Internet of Things (IoT): Taxonomy of security attacks," in 2016 3rd International Conference on Electronic Design (ICED), Phuket, Thailand, Aug. 2016, pp. 321–326. doi: 10.1109/ICED.2016.7804660.

[16] C. Lueninghoener, Getting started with configuration management. USENIX; login 36(2), 12–17 (2011)

[17] B. Chung, J. Kim, and Y. Jeon, "On-demand security configuration for IoT devices," in 2016 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Oct. 2016, pp. 1082–1084. doi: 10.1109/ICTC.2016.7763373.

[18] "An Overview of Chef Infra." https://docs.chef.io/chef_overview/

[19] P. Webteam, "Docs | Puppet." https://puppet.com/docs/

[20] "IT Automation with Ansible." https://www.ansible.com/overview/it-automation .

[21] "Salt Project Documentation." https://docs.saltproject.io/en/latest/

[22] R. Ranjan, K. Mitra, P. Prakash Jayaraman, L. Wang, and A. Y. Zomaya, Eds., Cyber Physical Systems and Internet of Things. Cham: Springer International Publishing, 2020. doi: 10.1007/978-3-030-43795-4.

[23] G. Sittampalam and N. Ratnarajah, "Enhanced Symmetric Cryptography for IoT using Novel Random Secret Key Approach," in 2020 2nd International Conference on Advancements in Computing (ICAC), Malabe, Sri Lanka, Dec. 2020, pp. 398–403. doi: 10.1109/ICAC51239.2020.9357316.

[24] Yaacoub, Jean-Paul & Salman, Ola & Noura, Hassan & Kaaniche, Nesrine & Chehab, Ali & Malli, Mohammad. (2020). Cyber-Physical Systems Security: Limitations, Issues and Future Trends. Microprocessors and Microsystems. 10.1016/j.micpro.2020.103201.

[25] Lopez, Javier & Rubio, Juan. (2018). Access control for cyber-physical systems interconnected to the cloud. Computer Networks. 134. 46-54. 10.1016/j.comnet.2018.01.037.

[26] Fink, Glenn & Edgar, Thomas & Rice, Theora & MacDonald, Douglas & Crawford, Cary. (2017). Overview of Security and Privacy in Cyber Physical Systems: Foundations, Principles and Applications. 10.1002/9781119226079.ch1.

[27] Michael P. Brennan "Using Snort For a Distributed Intrusion Detection System", version 1.3 in SANS Institute 2020, January 29, 2002

[28] Mcafee.com. 2021. Intrusion Prevention System – Network Security Platform | McAfee Products.Availableat:<https://www.mcafee.com/enterprise/en-us/products/network-security-platform.html>.

[29] A. K. Kyaw, Yuzhu Chen and J. Joseph, "Pi-IDS: evaluation of open-source intrusion detection systems on Raspberry Pi 2," 2015 Second International Conference on Information Security and Cyber Forensics (InfoSec), 2015, pp. 165-170.

[30] A. Sforzin, F. G. Mármol, M. Conti and J. Bohli, "RPiDS: Raspberry Pi IDS — A Fruitful Intrusion Detection System for IoT," 2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress

[31] Netresec. 2021. Public PCAP files for download. Available at: <https://www.netresec.com>.