A safety and security-aware industrial robotic system implementation: ROKOS

Muhammet Saral[1], *Ali Kemal Ayar*[1], *Burak Büyükyüksel*[1], *Ömer Şahabaş1, Gürol Çokünlü*[1], *Oğuzhan Urhan*[2]

[1]OTOKAR Otomotiv ve Savunma Sanayi A.Ş., Sakarya
msaral@otokar.com.tr
aayar@otokar.com.tr
bbuyukyuksel@otokar.com.tr
osahabas@otokar.com.tr
gcokunlu@otokar.com.tr

Kocaeli Üniversitesi Elektronik ve Haberleşme Mühendisliği Bölümü, Kocaeli
urhano@kocaeli.edu.tr

Özetçe

Gelişen teknolojiler birçok alanda olduğu gibi sanayide de kullanılmaktadır. İnsanla gerçekleştirmenin tehlikeli veya zor olduğu, insan performansının düşük kaldığı, zaman ve maliyet iyileştirmelerinin söz konusu olduğu durumlarda gelişen teknolojilerin çözüm olması beklenmektedir. Otokar'da üretilen gövdenin kalite kontrol işlemi, araç gövdesinin büyük, gövdedeki bileşen sayısının 2500-3000 civarında ve üretilen araç gövdesinin müşteri talepleri doğrultusunda sürekli değişiyor olması nedeniyle insan performansıyla hatasız ya da yüksek doğrulukla gerçekleştirmesi zor bir uygulamadır.

Otokar firması bu uygulamayı otomatik olarak gerçekleştirmek için robotik bir sistem geliştirmiştir. Değişen araç modellerine göre otomatik robot yörüngesi oluşturan bu sistemin emniyet, siber güvenlik ve gizlilik açısından doğrulaması ve geçerlenmesi gerekmektedir. Ayrıca sistem geliştirildiğinde iç ve dış saldırılara karşı güvenli olması için kapalı çevrim çalışacak şekilde kurgulanmıştır. Ancak günümüz koşullarında sistemden veri toplayabilmek ve sistemi diğer sistemlerle haberleştirip optimizasyon çalışmaları yapabilmek için sistemin İnternet ağına açılması istenmektedir.

Bu çalışmada geliştirilen endüstriyel robotik sistemin emniyetinin, siber güvenliğinin ve gizliliğinin doğrulanması ve geçerlenmesi için AB destekli Valu3s projesi kapsamında yapılan çalışmalar ve geliştirilen araçlar anlatılmaktadır. Penetrasyon testleri ve geliştirilen ROKOS (RObotlu Kalite kOntrol Sistemi) platformu ile sistemin çalışmasına ihtiyaç duymadan kötü niyetli saldırılar, hata enjeksiyonları karşısında ve otomatik robot yörüngesi oluşturma sürecinde sistemin performansının gerçeklenebilmesi konusunda bilgi verilmektedir.

Abstract

Developing technologies are used in industry as well as in many areas. Developing technologies are expected to be the solution in situations where it is dangerous or difficult to do with people, human performance remains low, and time and cost improvements are in question. The quality control process of the body produced at Otokar is a difficult practice to perform with human performance without errors or with high accuracy, since the vehicle body is large, the number of components in the body is around 2500-3000, and the body of the vehicle produced is constantly changing in line with customer demands.

Otokar has developed a robotic system to perform this application automatically. This system, which creates an automatic robot trajectory according to changing vehicle models, needs to be verified and validated in terms of safety, cybersecurity, and privacy. In addition, when the system was developed, it was designed to operate in a closed loop to be secure against internal and external attacks. However, in today's conditions, it is required to open the system to the internet network in order to collect data from the system, to connect the system with other systems and to carry out optimization studies.

This work describes the studies and tools developed within the scope of EU supported Valu3s project for the verification and validation of the safety, cybersecurity and privacy of the developed industrial robotic system. With the penetration tests and the developed ROKOS (Robotic Quality Control System) platform, realization performance of the system information is provided in the face of malicious attacks, error injections and in the process of creating an automatic robot trajectory without the need for the system to work.

1. Introduction

Otokar is producing autobuses and depending on the demand from the customer Otokar makes customizations in the vehicle body. A typical vehicle body consists of 2500-3000 parts, existence of which were controlled by quality operators. As there are too many parts and vehicle bodies are always

changing, sometimes absence of a part cannot be detected by operators in bodyshop control cell. In order to ensure that all the body parts are present as per design, Otokar developed an automatic robotic inspection cell [1] (Figure 1). In this industrial robotic cell, there are two 5-axis robots developed by Otokar and two cameras on each robot.

A digital twin of the robotic cell was developed to generate the safe trajectory of the robots to perform quality control [2]. The software starts by finding the window and door openings of the vehicle to decide the vehicle entry points. Hereafter, the software positions a virtual camera to the potential trajectory points inside and outside the vehicle to evaluate which part in the Bill of Materials (BOM) and what percentage of these parts can be viewed from these points. It then decides the robot trajectory, where all the parts in the BOM can be detected with minimum trajectory points in minimum time.

When the virtual camera is positioned in the digital twin built with CAD data, a synthetic 2D image showing the parts of the vehicle, which are in the view frame of the virtual camera, is created. The software, which performs these tasks in the virtual environment, sends a signal to the drivers controlling the axis motors of the robot via PLC to move the robots in the real production environment to the same location, where the virtual camera is positioned, to take 2D images of the vehicle with real cameras. Comparing the synthetic 2D images with the real 2D images part presence-absence check is performed. The method explained here, which is 2D image generation from 3D and then 2D-2D comparison is a method improvement when compared with classical 2D-2D and 3D-3D comparison techniques [3-4]. As a result, a report containing the location and part code of the missing parts is viewed in operator's screen.

However, this system was connected to a local network and had no connections to the networks which are open to the Internet. The main reason to keep this industrial robotic system in a local network was to protect the system from malicious attacks from outside. Moreover, the system was also not verified and validated for the attacks, which may occur inside. On the other hand, as the trajectory needs change depending on the diversity of the produced models, Otokar needed tools to verify and validate safety of the robot trajectory.

Participating in EU supported VALU3S project, Otokar aimed to develop and integrate the tools to the industrial robotic cell to improve the system performance by improving safety, cybersecurity and privacy (SCP) of the system.

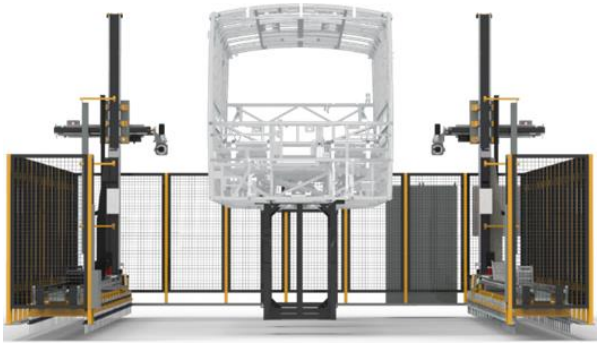

Figure 1: Robotic inspection cell for quality control

2. VALU3S: Verification and Validation of Automated Systems' Safety and Security

Automated system manufacturers and the component manufacturers of these systems spent enormous time on research and development activities. While doing these their primary goal is to make them work. However, switching from a prototype with newly developed features to the final product running on the field brings additional concerns such as security and safety. Hence, before being introduced into the market, automated systems need to be verified and validated considering the requirements of safety, cybersecurity, and privacy [5-6]. VALU3S project is focusing on improving methods and tools to provide a multi-dimensional framework, which can reduce the time and cost of verification and validation (V&V) process.

The framework consists of 8 dimensions which are: (i) the evaluation environment, (ii) the evaluation type, (iii) the type of component/system under evaluation, (iv) the type of V&V tools used, (v) the V&V stage in which the evaluation is conducted in, (vi) the purpose category where the component/system is best represented in, (vii) the evaluated quality attributes/requirements, (viii) the obtained evaluation performance indicators (Figure 2). The dimensions and their layers demonstrate how the V&V of automated systems are carried out for the six different use case domains, which are automotive, industrial robotics/automation, aerospace, agriculture, healthcare and railway, by mapping methods used in VALU3S to the Use Cases, test cases and evaluation scenarios stored in the repository.

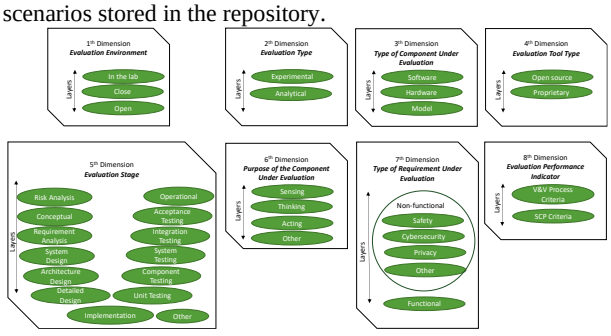

Figure 2: VALU3S multi-dimensional layered framework

3. Penetration Testing

In order to evaluate the system robustness in the case of user authentication, sensor data manipulation and evaluate effects of data manipulation in communication between server and PLC, a penetration test fulfilling ISO/IEC 27002:2013 was performed at the beginning of the project. Several attack types such as Man in the Middle (MitM), Denial of Service (DoS) and Address Resolution Protocol (ARP) Poisoning were executed [7-10]. During the tests open-source tools like ARP Spoofing, Wireshark, Ettercap, Greenbone, Metasploit were used [11-14]. As the system was designed to run in a closed loop to be secure initially, some vulnerabilities were identified during the penetration tests of the system, which is now open to Internet. Afterwards, a roadmap was determined that includes the measures to be taken to fix the system's vulnerabilities:

3.1) Network's topology has been improved

While creating the network topology during the design phase of the ROKOS system, the Purdue Model (Figure 3) was taken as a basis. In this model, a firewall is set up between the Control Zone and the Enterprise Zone and access to the OT (operational technology) level is limited.

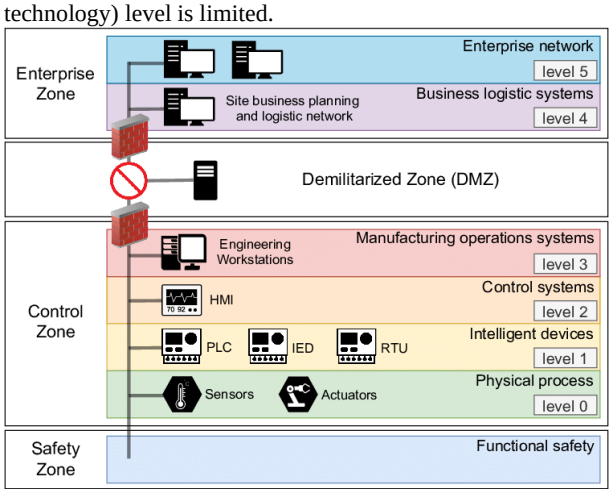

Figure 3: Purdue Model of ICS [15]

In the first topology (Figure 4), an offline cycle was designed, in which OT level had no network connection with the outside. System PC had connection only to a Network PC to receive or send data. System PC was transmitting data received from Network PC to automation system via OPC Server.

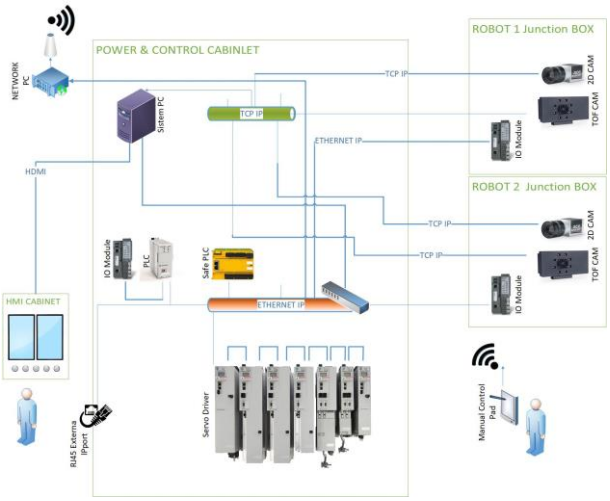

Figure 4: First Topology of the system

In the new and secure topology (Figure-5), an HSM (Hardware Secure Module) module on the server side which uses a random key generator for encrypting all data has been utilized. Afterwards a secure gateway which decodes all data and sends it to the client's services has been implemented. As a result of this, all the data which has been transferred between the server and the client was protected against attacks. If someone accesses the data transferred through network, this data will not be meaningful without the encryption key. In another case, if someone changes the content of the data without knowing the encryption key, data will not be processed by secure gateways and it will be understood that the data flow is corrupted.

Local safe zones, which have been encrypted by Secure Gateways, were established to maintain secure communication. While communication within the zone is provided to its own standards, communication with another zone or equipment is provided securely over Secure Gateways. This topology enabled all local zones to communicate securely among themselves.

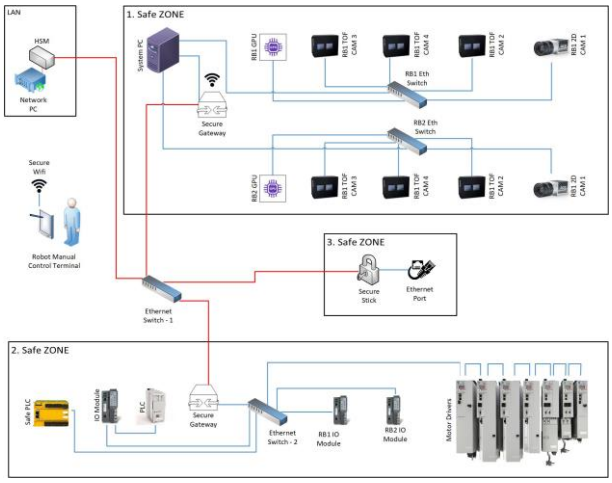

Figure 5: New and secure system topology

3.2) Network's security has been increased

In addition to topology improvements, the OT and IT network has been separated in two different VLAN (each has its own security settings). Physical firewall has been implemented between VLANs. The aim is to control every OT network access and prevent all malware injections. Furthermore, smart switch with the NAC protocol has been used for identifying the user's access privilege and managing the level of security of the users.

3.3) Corrupted data identification

Our own algorithm which is based on hash algorithms has been developed to identify the corrupted data transmitted to the secure zone. When ROKOS tool generates new robot trajectories and synthetic 2D images, our algorithm runs and reference data is created. The hash code codes of reference data is encrypted and saved in secure database. These hash codes are used as ground truth to determine corrupted data.

A time-based control method has been implemented for linking all data's hash codes. Every data in the series uses the previous data's hash code to generate its own hash code. With this method if any data produced is modified or deleted completely it can be detected and recovered.

During the demonstration phase all of these steps will be integrated in the quality control cell to have a TRL6 system. Then the system will be tested using the V&V criteria defined in Valu3s project.

4. ROKOS Tool

4.1. Trajectory Planning and Synthetic Image Generation

Robots need trajectory to be able to perform the planed task. The trajectory programming for the robots is usually carried out manually by the operators or it is done manually in offline programming tools. This programming effort is needed for every new product that will move into the production line. In the production areas where there is high diversity the need for programming increases time to market and when it is done manually there is a high risk of human error. Trajectory programming has been made human-independent by performing CAD model analysis with the help of digital twin software developed by Otokar, which will be explained in detail in Section 4.15.

4.1.1. Designing a virtual camera in a digital twin

Data such as internal and external parameters of the camera used, horizontal and vertical viewpoints of the used optical setup were obtained from the equipment manufacturers, and a virtual camera was designed in the digital twin software, which is equivalent to the physical camera. The area marked in red in Figure 6 shows the designed virtual camera.

Figure 6: Designed virtual camera in the digital twin

By positioning this virtual camera in CAD space, 2D synthetic images showing the view in the frame of the virtual camera are generated from 3D model in the digital twin.

4.1.2. Determination of the bounding-box of the vehicle

After the vehicle data is imported into the simulation software, the bounding-box of the vehicle must be determined in order to process the STL data. The bounding box determination process was carried out using the vertex points in the STL data.

By using the minimum and maximum values of the vertex points in the x, y and z planes, the bounding-box was determined and the volume covered by the vehicle was determined in space. The regions marked in red in Figure 7 correspond to the bounding-box coordinates.

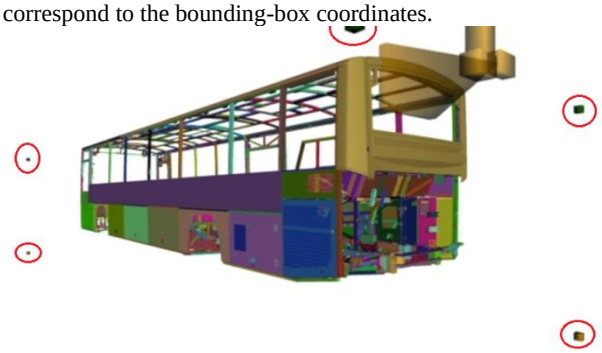

Figure 7: Bounding-box coordinates of the vehicle

4.1.3. Obtaining vehicle entry points

There are two camera systems, each of which are mounted to a robot, in our system. Thus, simultaneous operations can be performed on each side walls of the vehicle. The area occupied by the side walls of the vehicle in spatial dimension is known after the bounding-box determination. In order to prevent the robot from colliding with the chassis, it is necessary to determine the entry points of the vehicle, such as windows or doors.

By voxelization of both side wall bounding-box spatial areas of the vehicle, 3D pixel boxes were created. These boxes were moved towards the vehicle from both sidewalls unless there was a collision with the vehicle chassis. A virtual depth map of the vehicle was created by recording the movement distances at the collision points. The steps of obtaining vehicle entry points are shown in Figure 8.

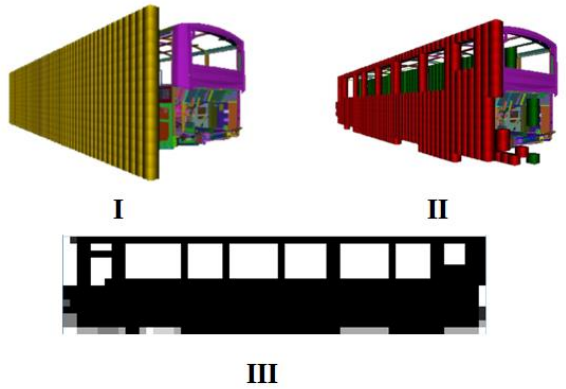

Figure 8: Steps of obtaining vehicle entry points

This depth map was analyzed with image processing techniques and the entrance areas such as windows and doors of the vehicle were determined.

After the determination of the entrance regions of the vehicle, trajectory nodes, which are potential trajectory points, are created. These nodes belonging to different section such as the top, bottom, side walls, rear and front sections of the vehicle, interior of the window and interior of the door in space were stored with the region labels where they were created. The reason for adding this tag information will be explained later.

4.1.4. Creation and scoring of trajectory nodes
Digital twin software scans and scores all potential trajectory points with different camera orientations. Figure 9 shows the scoring process of the potential trajectory points using the virtual camera.

Figure 9: Scoring process of potential trajectory points

This scoring process is carried out taking into account the following factors.

- Total number of visible parts in the frame of virtual camera at that coordinate and orientation
- The sum of the ratios of the visible area of each part in the current image to the area it occupies in space

The averages of the above numerical values produce the potential trajectory point score.

4.1.5. Optimization of trajectory points
Using the scored potential trajectory points safe robot trajectory, which can detect all parts with minimum points and minimum time, should be generated. Trajectory point selection is carried out according to the following criteria:

1. A structure is created that keeps the visibility of all chassis parts in space at selected nodes.
2. For the trajectory point selection process, the potential trajectory points are first ranked according to their scores.
3. The most valuable node is chosen as the master.
4. All parts that the master node can detect are marked if the visibility rate in the structure in step-1 is above 75%.
5. Parts marked with a visibility higher than 75% are deleted from the score list of next points as they are already detected
6. It is repeated from step-2 until there is no potential trajectory point to evaluate.

Using tags on the trajectory points, the trajectory information was written directly to the database without adding any trajectory point between the two trajectory points for the transition from the outer region to the outer region (i.e., transition from side wall point to bottom point). However, by considering the collision with the chassis during the movement from the inner region to the outer region or from the outer region to the inner region, intermediate trajectory points have been added to reset the camera positions between these transition trajectory points. By means of resetting the camera positions, the collision with the chassis is prevented. After the above processes, chassis specific trajectory generation was achieved.

4.1.6. Trajectory navigation and generation of synthetic images in the digital twin
The trajectory is run offline in the digital twin to perform collision check and generation of synthetic images to use as a ground truth. Thus, while the reliability of the trajectory was tested, it was also ensured that synthetic virtual images were created in advance to speed up the part existence control in field. Examples of synthetic images obtained are given in Figure 10.

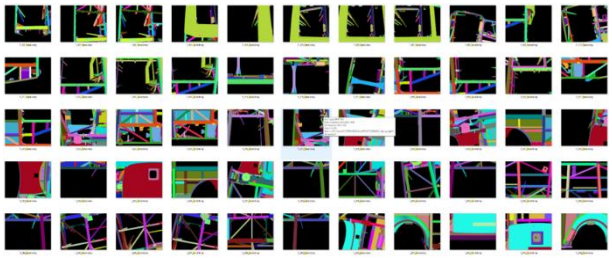

Figure 10: Examples of synthetic images generated

4.2. ROKOS & IM-FIT Tool Integration

As explained in the previous section ROKOS tool generates the safe robot trajectory and for each trajectory point 2D synthetic images are generated. While the system is running 2D images captured form the field and the 2D synthetic images are used to perform part existence check.

IM-FIT tool has been developed for testing the performance of ROKOS system with corrupted images to simulate the situation if someone hacks the cameras. IM-FIT uses the images captured form the field and stored in database. Then adds some noise to them. Afterwards it sends fault injected (manipulated) images to ROKOS tool. Part existence checks are performed using the

fault injected images and 2D synthetic images. At the end, system performance after injection of a fault or combination faults is reported.

As a result of this study, effects of different faults on the industrial robotic system can be verified and validated in a Simulation Based Test environment in a short time, without human effort and running the system on the field.

5. Evaluation Scenarios, Criteria and Test Cases

After implementation of ROKOS and other tools and improvements in hardware and software level and network topology, industrial robotic system's SCP will be evaluated according to some Evaluation Scenarios, Test Cases and Criteria, which are as follows**.**

5.1. Evaluation Scenarios

Manipulation of Sensor Data: Manipulation/corruption of sensor data stream at camera and safety sensors

Server and PLC Communication: Manipulation/corruption of PLC data stream

Safety Trajectory Optimization: Creating robot trajectory points automatically covering the safety of the robot and its apparatus as well as static objects in the workspace

Anomaly Detection at Component and System Level: Observation of the inspection process flow data to detect anomalies in production phases and component parameter data by utilizing ML and/or deep learning-based techniques

Server Ethernet Network Security: Monitoring and inspection of unexpected network data activity. In case of unexpected network data activity, the system will be shut down with safety protocols and will start working with a back-up server.

5.2. Test Cases

The test cases implemented are described below by giving preconditions, inputs and expected results.

Robot Trajectory Test

- **Preconditions:** Digital twin of the system should be modelled in Gazebo environment.
- **Input conditions / steps:** Trajectory created in digital twin which can control vehicle parts without collision and virtual images (2D) should be present.
- **Expected results:** Existence control of minimum %95 parts of vehicle in less than 25 minutes and in each trajectory point minimum %15 of each part must be visible.

Man in the Middle Attack Test

- **Preconditions:** Enable access to local network over a cable (like Cat 6, etc)
- **Input conditions / steps:** Try to sniff network communication
- **Expected results:** Cannot sniff network communication

Denial of Server Attack Test

- **Preconditions:** Enable access to local network over a cable (like Cat 6, etc)
- **Input conditions / steps:** Try to saturate local network by sending huge amount of communication packets.
- **Expected results:** Detection and isolation of attacker.

Address Resolution Protocol Poisoning Test

- **Preconditions:** Enable access to local network over a cable (like Cat 6, etc)
- **Input conditions / steps:** Try to positioning with fake ARP requesting packets.
- **Expected results:** Local switch infrastructure protected by security systems.

User Authentication Protocol

- **Preconditions:** Enable access to local network connection over a cable (like Cat 6, etc.).
- **Input conditions / steps:** Try to grow up the basic privilege to supervisor privilege.
- **Expected results:** Unable to access to the high level privilege.

Penetration Test (Firewall, Router etc.)

- **Preconditions:** Enable access to local network and external IP address
- **Input conditions / steps:** Try to bypass firewall and router systems.
- **Expected results:** cannot bypass firewall and router systems.

Task Safety in Faulty Situation

- **Preconditions:** body inspection tasks are provided to operation stack.
- **Input conditions / steps:** Mutating operations with faulty situations.
- **Expected results:** System do not cause any unsafe movement or behavior.

Fault Injection to Robotic System

- **Preconditions:** System should be operating normally without any fault.
- **Input conditions / steps:** Injecting faults and mutating source code of the operating system.
- **Expected results:** Robot system will handle errors and continue operating in normal mode.

Task Safety in Faulty Situation

- **Preconditions:** body inspection tasks are provided to operation stack.
- **Input conditions / steps:** Manipulating data transfer nodes
- **Expected results:** System do not cause any unsafe movement or behavior in cell.

Fault injection to robotic system

- **Preconditions:** System should be operating normally without any fault.
- **Input conditions / steps:** Injecting faults in source code which manipulates virtual pictures
- **Expected results:** Robot system will detect error and report situation.

Server and PLC Communication Test

- **Preconditions:** HSM and Secure Gateway adding to the system network.
- **Input conditions / steps:** Fault injection to robot trajectory on the both server and local system (PC&PLC) side.
- **Expected results:** Inspection of data manipulation and working properly with uncorrupted trajectory data from safe back up.

SCP Criteria

Eval_SCP_1: This criterion refers to the case in which the impact of a fault or an attack is covered e.g., due to the inherent robustness of the system under test or as a result of using mechanisms such as error handling or intrusion handling. This is to make sure that faults and attacks do not affect the output of the system and results in a deviation to the nominal output. The deviation, if significant, could have severe implications whereas if not significant could be classified as benign.

Eval_SCP_3: Simulate malicious attacks and faults in the system and check the quality of the detection. Percentage of correctly identified attacks and faults including confusion matrix [31] and/or sensitivity, specificity, accuracy, and precision.

Eval_SCP_9: Verifying that safety mechanisms prevent faults from leading to an accident.

Eval_SCP_12: Model-based software testing including fault injection to ensure fault-tolerant use case activity. Compliance through measurement and verification results.

Eval_SCP_13: Testing on a simulation-level of system under test with fault injection plug-in for system robustness assessment.

6. Conclusion & Future Work

Since in many factory models produced are constantly changing, the robot trajectory is prone to error when left to human initiative. However, when robot trajectory generation is given to automation systems, automation systems need to be verified and validated. With the development of the ROKOS tool and the V&V of the safe robot trajectory, an important know-how has been obtained in the Simulation Based Testing method. In addition, the ROKOS tool enabled the V&V process to be carried out in a short time and at less cost. Furthermore, failure to provide cyber security against attacks on industrial systems also creates safety risks. After the Penetration Test studies, standards have been established for the safe opening of an industrial robotic system to the Internet network.

The ROKOS tool is designed as a platform and is open to integrating other tools developed by the consortium. After the integration of the IM-FIT tool, fault injections were made into the system and it was observed that the system was affected differently by different errors. It has been decided to start a new initiative to develop an artificial intelligence algorithm that will determine whether the images obtained from the environment contain errors and if so, what type of error they contain.

Acknowledgement

Also, the research leading to this paper has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 876852. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Austria, Czech Republic, Germany, Ireland, Italy, Portugal, Spain, Sweden, Turkey. The views expressed in this document are the sole responsibility of the authors and do not necessarily reject the views or position of the European Commission.

References

[1] Brogårdh, “T.,Present and future robot control development—An industrial perspective”, Annual Reviews in Control, Volume 31, Issue 1, Pages 69-79, ISSN 1367-5788, 2007.

[2] N. Kousi, C. Gkournelos, S. Aivaliotis, C. Giannoulis, G. Michalos, S. Makris, "Digital twin for adaptation of robots’ behavior in flexible robotic assembly lines", Procedia Manufacturing, pp. 121-126, 2019

[3] J. Ma, X. Jiang, A. Fan, J. Jiang, J. Yan, “Image Matching from Handcrafted to Deep Features: A Survey”, vol. 129, p. 23-79, 2021.

[4] L. Li, R. Wang, X. Zhang, “A Tutorial Review on Point Cloud Registrations: Principle, Classification, Comparison, and Technology Challenges", Article ID 9953910, 2021.

[5] Leitner, A., Holzinger, J., Schneider, H., Paulweber, M., Marko, N.: “Seamless tool chain for the verification, validation and homologation of automated driving. In:

Validation and Verification of Automated Systems", pp. 165–176. Springer, 2020
[6] U. Ozguner, C. Stiller, and K. Redmill, "Systems for safety and autonomous behavior in cars: The DARPA grand challenge experience," Proc. of the IEEE, vol. 95, no. 2, pp. 397–412, Feb 2007.
[7] Yang, Y., McLaughlin, K., Littler, T., Sezer, S., Im, E. G., Yao, Z. Q., & Wang, H. F. (2012).: Man-in-the-middle attack test-bed investigating cyber-security vulnerabilities in smart grid SCADA systems.
[8] Bechtsoudis, A., & Sklavos, N. (2012). Aiming at higher network security through extensive penetration tests. IEEE latin america transactions, 10(3), 1752-1756
[9] Denial-of-service attack. https://en.wikipedia.org/wiki/Denial-of-service_attack
[10] Denis, M., Zena, C., & Hayajneh, T. (2016, April). Penetration testing: Concepts, attack methods, and defense strategies. In 2016: IEEE Long Island Systems, Applications and Technology Conference (LISAT) (pp. 1-6). IEEE.
[11] https://www.wireshark.org/
[12] https://github.com/Ettercap/ettercap
[13] https://www.openvas.org/
[14] https://github.com/rapid7/metasploit-framework
[15] https://www.researchgate.net/figure/ICS-Purdue-Model-architecture_fig1_349195440