InSecTT Technologies for the Enhancement of Industrial Security and Safety

Sasikumar Punnekkat, Tijana Markovic, Miguel León, Björn Leander, Alireza Dehlaghi-Ghadim, and Per Erik Strandberg

Abstract The recent advances in digitalization, improved connectivity and cloud based services are making a huge revolution in manufacturing domain. In spite of the huge potential benefits in productivity, these trends also bring in some concerns related to safety and security to the traditionally closed industrial operation scenarios. This paper presents a high-level view of some of the research results and technological contributions of the InSecTT Project for meeting safety/security goals. These technology contributions are expected to support both the design and operational phases in the production life cycle. Specifically, our contributions spans (a) enforcing stricter but flexible access control, (b) evaluation of machine learning techniques for intrusion detection, (c) generation of realistic process control and network oriented datasets with injected anomalies and (d) performing safety and security analysis on automated guided vehicle platoons.

1 Introduction

Industry 4.0 is aiming towards convergence between industrial systems and IT infrastructures to enable higher levels of productivity through information sharing among all stakeholders. Digitalisation, automation, autonomy, artificial intelligence (AI), cloud computing, higher connectivity are regarded as key drivers the next industrial revolution. Breaking the traditional 5-level automation pyramid (“Purdue”) architecture to enable interoperability, autonomy and seemless data transfers however comes

S. Punnekkat (✉) · T. Markovic · M. León
Mälardalen University, Västerås, Sweden
e-mail: sasikumar.punnekkat@mdu.se

B. Leander
ABB AB, Västerås, Sweden

A. Dehlaghi-Ghadim
RISE, Västerås, Sweden

P. E. Strandberg
Westermo Network Technologies AB, Västerås, Sweden

M. Karner et al. (eds.), *Intelligent Secure Trustable Things*, Studies in Computational Intelligence 1147, https://doi.org/10.1007/978-3-031-54049-3_5

with some concerns, especially with respect to safety and security in the traditionally rigid industrial segments.

As the future factories are envisioned to be flexible, adaptable and collaborative endeavours involving man and machines ("autonomous robots") forming complex system of systems, their emergent behaviours are quite hard to fully characterise at design time. Majority of the factories also can be termed as safety critical systems since failures can often lead to adverse impacts not only on productivity but also on humans, infrastructure and environment. Here comes the mandatory and often legal safety requirements set forth by various generic/domain specific standards and machine directives.

Security is one of the major focus aspects of the EU-funded InSecTT project (https://www.insectt.eu/). The ever-increasing landscape of cyber security threats, together with higher levels of connectivity and opening up of traditionally closed factories into Internet, pose many potential risks to productivity, safety of products and processes, as well as industrial repute. It becomes paramount to perform detailed hazard and risk analysis and careful planning of mitigation mechanisms to meet security requirements applicable to the targeted industrial domains.

2 Background

One of the key building blocks of the EU-funded InSecTT project is denoted as "BB3.1: Methodologies, concepts system solutions for enabling safety and security". It focuses on cross-layer security analysis, concepts, and system solutions including the essential steps of revealing security requirements and performing a threat analysis, defining suitable security methodologies, planning mitigation and resilience strategies (on system level), and finally looking at defining (cyber-)security concepts and solutions. The aim was to provide applied solutions as well more generic schemes, which can be used for a wider range of applications (e.g., cybersecurity in IoT system). Enabling safe system operation is of utmost priority for this task.

The task participants were an excellent mix of industrial partners (ABB, INDRA, ISS RFID, Kaitotek, LDO, LCC, Nurd, Philips Research, TietoEvry and Westermo) and academic partners (Mälardalen University (MDU), CINI, RISE, UCC, UTwente, UPM), who together provided 50+ specific requirements. These requirements can broadly be classified into the following four themes addressing cross-layer system level concepts and solutions as indicated in Fig. 1:

- Security requirements and threat analysis: High level requirement analysis and analysis of various vulnerabilities and effects of cyber-attacks on them.
- Security methodologies, concepts and system solutions: Focus on design of system/ application-level approaches and methods for identification of various security vulnerabilities as well as proactive measures for avoiding them.

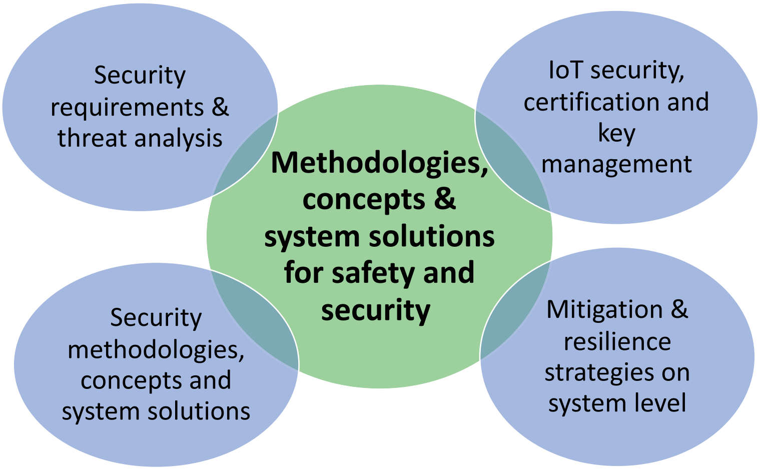

Security requirements & threat analysis
IoT security, certification and key management
Methodologies, concepts & system solutions for safety and security
Security methodologies, concepts and system solutions
Mitigation & resilience strategies on system level

Fig. 1 Cross-layer security analysis, concepts and system solutions

- IoT security, certification and key management: Device/edge level mechanisms for enabling secure infrastructures and architectures access control and for enforcement of privacy.
- Mitigation and resilience strategies: Identification and implementation of adequate mitigation strategies to assure required levels of systems' reliance and robustness.

During the first half of the InSecTT project, a deep discussion on the individual partner contributions took place, elaborating on the various use cases, their alignments and possible synergies, resulting in the following six sub building blocks:

(A) Access control and authentication infrastructure;
(B) Intrusion detection systems;
(C) IoT privacy and security mechanisms;
(D) Secure IoT applications;
(E) Security guidelines;
(F) Tools and simulators.

The grouping is mainly done to find synergies and collaboration than on unification. In this chapter, we mainly present some of the research/technologies, relating to A, B, E and F, developed by a set of Swedish partners working closely in the realisation of two use cases related to smart collaborative manufacturing and secure network communications. Main critical aspects we focused on were ensuring safety/security of the industrial automation and control systems and security and privacy of the network infrastructure.

2.1 Industrial Automation and Control Systems

Industrial Automation and Control Systems (IACS) are used for operating a wide range of industrial applications, including critical infrastructure, such as power plants and clean water supplies [62]. The safe and secure operations of these systems are of utmost importance, for system owners from a business perspective, for private persons relying on reliable services and safe products, and for the society as a whole for supply of critical resources and as a basis of economical stability.

There is a trend in Industrial Control System (ICS) architectures, transitioning from a hierarchical controller-centric model as described by the Purdue Enterprise Reference Architecture (PERA) [67], towards a network-centric design strategy using a common network back-bone [2, 32]. Driving forces behind this trend are technical advances as well as novel business models and market expectations related to flexibility and customization, etc. The trend is connected to the Industry 4.0 [21, 31, 42] paradigm which is currently shaping the future of IACS, implying huge changes both from a business and technological perspective.

The named developments have a fundamental impact on the technical level for how IACS are constructed, implying increased connectivity, higher diversity and complexity, and more classes of stakeholders taking part in the system. This makes cybersecurity a major concern.

3 Selected InSecTT Technologies Targeting Security and Safety

3.1 Access Control and Authentication Infrastructure

Access control [58] is a crucial aspect of enhancing security in industrial systems [22], but it is still relatively underdeveloped compared to modern IT systems.

At the beginning of the InSecTT project, a survey was conducted among cybersecurity practitioners within industrial organizations and companies in Sweden. The study was designed to sample which techniques and principles are used related to access control, what was the foreseen challenges in the area, and, possibly, to see if the technical maturity in access control usage was related to used cybersecurity standards. The study [36] focuses on two essential cybersecurity requirements: identification and authentication control.

The purpose of performing the study was to increase knowledge on state-of-the practice and establish a starting point for further exploration, thereby bridging the gap between the current state and the requirements of emerging systems with regards to access control.

Cybersecurity is an area were practitioners typically are reluctant in sharing potentially sensitive information, which made recruiting respondents to the study difficult. The surveying activity received enough responses to analyze and give a broad picture of “what is out there”, but not enough to claim any statistical significance. Questionnaire was sent out to 350 organizations, which resulted in 40 respondents, some of which dropped out before completing the survey.

In the part of the survey related to challenges, the respondents could answer in a free-text form. The answers were then analyzed, resulting in seven different themes:

T1 Cost related to inclusion of secure HW components.
T2 Cost of account management.
T3 Increasing system complexity.
T4 Lack of technical support and standardization.
T5 Improper use of methods.
T6 Regulations related to open market making implemented methods ineffective.
T7 Increasing amount of cyber-attacks.

Analyzing the perceived challenges indicated by the respondents, it becomes clear that they see increasing costs related to components (theme **T1**) as well as management effort (**T2**, **T3**) in relation to identification and account management. This may be an effect of increased system complexity driven by the Industry 4.0 evolution, but also requirements related to evolving best practices. As an example, the cost for changing from shared user accounts to unique user accounts puts a significant additional burden on the account management process.

The heterogeneity of the future industrial systems is seen as a big challenge (theme **T4**), with different component manufacturers choosing incompatible technical solutions. A lack of standardization is mentioned by several respondents as an issue hampering effective account management in industrial systems.

Three themes imply direct threats to the integrity of the industrial systems. Theme **T5** indicates a lack of technical maturity leading to improper usage of the available methods. Theme **T6** indicates that “right to repair”-regulations may force manufacturers to include mechanisms which could make authentication less secure. The theme **T7** related to cybersecurity attacks on industrial systems are possibly worsened by the previous two, as the likelihood of a successful attack will increase with improperly configured systems or inherently vulnerable mechanisms. Cybersecurity attacks and information leakages in other seemingly unrelated systems may have collateral impact also on industrial systems using unique user identifications, as password re-use over several platforms is a common issue.

The perceived challenges illustrate the on-going technical shift from isolated to increasingly interconnected systems, with a resulting complexity and heterogeneity that currently used solutions cannot handle, requiring investments both related to technical components and system solutions for account management. The fear is that lack of standardization and improper usage of technical solutions may lead to more vulnerable systems, consequently increasing the likelihood of successful cybersecurity attacks.

It is clear that detailed access control used correctly will improve security characteristics of a system, but there are risks of complexity and heterogeneity making management efforts too costly and difficult. An advanced security mechanism which is poorly configured can be worse than a very simple one used correctly. In the light of these challenges, we wanted to develop approaches and methods for handling access control in support of the emerging IACS characteristics, which are practically useful with regards to management effort and adherence available industrial standards.

Dynamic manufacturing is a well-established development of industrial automation and control systems. Manufacturing environments have, to a large extent, been optimized for high-volume production to a low per-item cost. This has led to highly specialized and optimized factories with a high complexity. These factories are prone of being difficult and expensive to retro-fit for changing demands or requirements.

Smart manufacturing [12, 47] and modular automation [30, 70] are design strategies optimized for being adaptable and customizable, in order to easily ramp up or down production, adapt to new innovations or specific customer requirements, etc. The resulting systems are dynamic manufacturing environments, which exhibits different levels of dynamicity, e.g., for modular automation, as follows.

1. Dynamic system composition–available processing modules and how they are interconnected change over time, due to changing high-level requirements.
2. Dynamic production schemes–available and active recipes describing the production workflow change on a daily basis, based on business requirements.
3. Dynamic operations–during recipe execution, different steps of the recipe-workflow are activated, implying different processing operations being executed.

In order to follow the principle of least privilege, being one of the fundamental practices within the access control theory [56], the rules of access control should adapt to the current system state. One difficult challenge arises on how to formulate access control policies to be sufficiently close to the least-privilege principle, while keeping the engineering effort related to policy formulation on a manageable level.

To investigate this challenge for dynamic manufacturing systems, we performed a study looking at five access control strategies, where three are currently being used, and two progressively aim towards a thought ideal.

All the strategies were implemented and evaluated in a simulation experiment with a number of attack scenarios, clearly showing each strategies relative effectiveness toward different attacks. As part of the study we also developed a method to automatically generate access control policies based on already available engineering data, targeting the challenge of minimizing the management effort of upholding policies close to the least-privilege principle. Details on the strategy evaluation and developed methods are available in [35].

Policy models [57] are focusing on the primitives and logic used for describing access control rules. As important are policy enforcement models, which describe the components needed, and their interactions, in order to ensure that the formulated policies are followed.

Our study [37] looked into how an access control enforcement architecture apt for dynamically changing access control scenarios of dynamic manufacturing systems could be constructed. Dynamic access control is not widely used in IACS, but it is highly relevant for the evolving system types which are inherently dynamic. Four different enforcement architecture models are investigated and evaluated based on three important metrics: resource server workload, network, load, and flexibility. The two most promising models required policy delegation mechanisms using access tokens. Four different variations on how to encode policy decisions into access tokens are provided and discussed with regards to available support in the Open Process Communication Unified Automation (OPC UA) standard [49]. Finally an implementation is performed, using a combination of one of enforcement models and delegation mechanisms, e.g., detailing the authorization protocol, access token encoding logic, and policy decision logic in the resource server. Separate studies [38, 52] are performed evaluating quality metrics on different aspects of the proposed OPC UA authorization protocol.

3.2 Intrusion Detection Systems

One of the biggest challenges in the network security research area is identifying malicious activities on time and mitigating them promptly. The process of analyzing network traffic to identify signs of malicious activity is called intrusion detection [41] and a system that automates this process is called the Intrusion Detection System (IDS) [8]. There are two common methodologies that IDSs use to identify threats: signature-based and anomaly-based [59]. A signature-based IDS monitors network packets and searches for patterns that correspond to known network attack types. Anomaly-based IDS learns the general behavior of normal network traffic and raises an alarm when significant deviations are detected.

In recent years, Machine Learning (ML) has become a popular and effective method for developing new anomaly-based IDS [9, 17, 46, 60, 65].

ML is the part of AI where algorithms learn patterns from datasets without explicit instructions [55]. It can be divided into the following areas:

- Supervised Learning (SL): algorithms within the SL category use input-output pairs to learn a function that maps from inputs to outputs;
- Unsupervised Learning (UL): algorithms within the UL category learn patterns within the input data without any output information given in the training phase;
- Reinforcement Learning (RL): algorithms within the RL category learn by trial and error. A specific "reward" or "punishment" is given depending on their actions and consequences.

Various ML algorithms were applied to existing datasets either to separate normal traffic for the malicious one (binary classification problems) or to detect specific attack types (multiclass classification problems). Buczak et al. [10] did a focused literature survey of ML methods used in IDSs and recognized some of the most commonly used methods such as Random Forest (RF), Decision Trees, density-based clustering algorithms (e.g., DBSCAN), Support Vector Machine (SVM), Artificial Neural Networks (ANN), Naive Bayes (NB), association rules, etc. Many studies in this area analyze the accuracy of different ML algorithms on different benchmark datasets. Revathi et al. [53] presented an evaluation of supervised ML algorithms (RF, J48, SVM, Classification and regression trees and NB) for multiclass classification on one dataset (NSL-KDD) and derived the conclusion that RF has the highest accuracy compared to all other algorithms. Abedin et al. [4] worked on the same problem by applying NB, J48, NBTree, Multilayer Perceptron (MLP), and RF and their findings were that J48 and RF had the best performance. Tuan et al. [66] evaluated SVM, ANN, NB, Decision Tree, and unsupervised ML on the UNBS-NB 15 and KDD99 datasets. This paper considered only the Distributed Denial of Service (DDoS) attacks and unsupervised ML was the best at differentiating between DDoS and normal network traffic, but it was not specified which unsupervised ML algorithms were used. There are several papers that evaluate a single ML algorithm on one or more benchmark datasets, such as different types of neural networks [9, 24, 28, 54, 68], RF [16], SVM [50], K-means [29], etc.

Most of existing works focus on evaluating ML algorithms on a single dataset or on evaluating single ML algorithm on multiple datasets. Also, most of the papers focused only on binary or multiclass classification.

Our research efforts in the EU-funded InSecTT project with respect to intrusion detection had the following overall objectives:

- Apply multiple methods on multiple datasets and compare their performances.
- Extend the SOTA anomaly classification problem.
- Extend the SOTA and SOP for the realization of federated learning in industrial contexts with resource constraints.
- Study the pros and cons of existing datasets and design new more realistic datasets and simulators to suit the manufacturing and networking domains.

In this section, we briefly present our results on the first three objectives, while the fourth one is addressed in Sect. 3.3.

As previously mentioned, AI is used as an IDS by many authors on different datasets, but the test results are usually limited in terms of algorithms, datasets or problem that was solved (anomaly detection or anomaly classification).

In [39], we made a more complete comparison including a total of 5 supervised learning algorithms (ANN, SVM, KNN, LDA, RF) and 3 unsupervised learning algorithms (K-means, mean-shift and DBSCAN) tested for anomaly detection and anomaly classification on 4 datasets (KDD99, NSL-KDD, UNSW-NB15 and CIC-IDS-2017). The results showed that RF, KNN, and SVM were the algorithms that performed the best in terms of accuracy. If in addition training and testing time are considered, RF emerges as the best option.

Now we know which ML method is more suitable for the desired problem. However, we believe that more layers of security are needed. For this reason, we proposed a Federated Learning (FL) framework that increases the security of data [44]. The framework is used on different clients (i.e., routers) that receive packages. On each client, a different RF is implemented and trained on the edge. RF is selected because of the various reasons that are proved by the experiments presented in the previous paragraph: the best performance and a reasonable time to be implemented in a real-time scenario. On top of that, RF is the algorithm with a high degree of explainability.

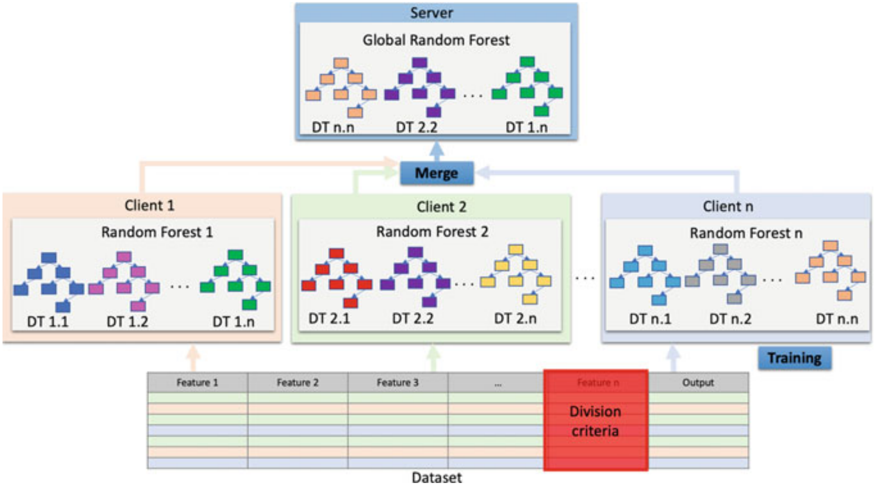


Fig. 2 Architecture of the Federated Learning Framework based on RF [44]

After training the models on different clients, these are sent to the server where a new super RF is created as a combination of them. Then, this final model is sent back to the clients to perform anomaly detection and classification. An overview of the framework is presented in Fig. 2. The results of the experiments clearly showed that combining the different RF algorithms is beneficial for the algorithm performance, increasing the detection rate. With this framework, we avoid distributing the data through the network to create the model in the server with all the data available, which protects the data in a sense, since after training the models, these data can be deleted and the risk of an intruder accessing the data is eliminated.

Subsequently, we proposed a second framework in which data, in this case, could be shared between different entities, because the data are encoded [40]. This framework uses an autoencoder (type of ANN [19]) to encode the data used by the ML algorithm to detect or classify anomalies. The novelty lies in the use of an optimization algorithm called Differential Evolution (DE) [61] to train the autoencoder. It uses two objectives to find the best model: (a) the error of the autoencoder when trying to decode the encoded features, and (b) the accuracy of a ML algorithm. The results show that for the algorithms that obtained the best performance in our first comparison [39], the performance is reduced by a small amount. On the other hand for the algorithms that could not perform high-quality anomaly detection or classification, performance increased significantly. Furthermore, the method is compared to the principal component analysis [3] obtaining better results for anomaly detection. An overview of the presented framework is given in Fig. 3. With this method, we can send the encoded data through the network with the certainty that no intruder will be able to decode the data since the model is needed for decoding. An additional benefit of this method is that by using autoencoder, we are able to recover the original data, which is not possible by the majority of the well-known encoding methods.

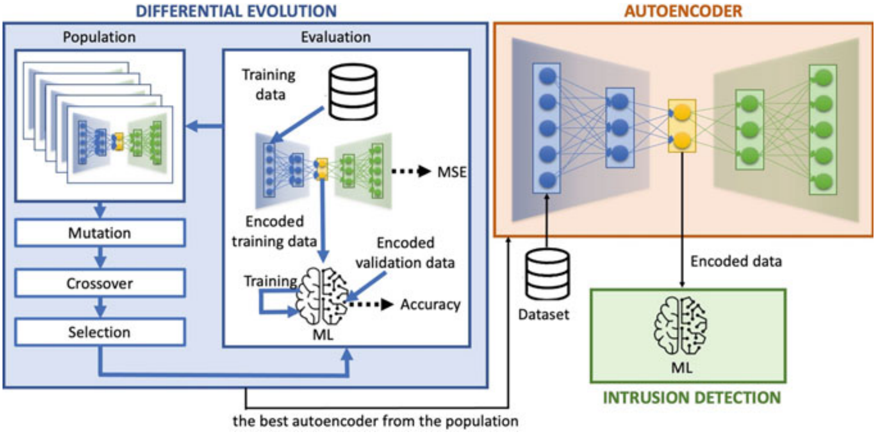


Fig. 3 Proposed framework for feature encoding [40]

3.3 *Tools, Simulators and Datasets*

The majority of experiments in the area of intrusion detection were conducted using one or more benchmark datasets [18]. Some of the well-known IDS datasets are: KDD99, NSL-KDD, UNSW-NB 15, CIDDS-001, CICIDS2017, CSE-CIC-IDS2018, etc. All of those datasets consist of a combination of normal and malicious traffic. Data about network packets were preprocessed to create the features, and every entry was labeled as normal activity, or as some type of network attack.

The impossibility of testing in operational industrial systems due to security issues leaves a gap in the environment for testing and measuring the impact of cyber threats and the development of defense systems. There are some real ICS testbeds in the world, such as the national SCADA testbed or Swat, a small-scale water treatment center. However, these testbeds are not accessible by all researchers. Building such an environment is also a pretty time-consuming and expensive process. Besides, scientists should deal with a vast range of unrelated technical problems that needs HW knowledge. These barriers have led many security researchers, especially those who want to use AI and ML methods for attack detection, to use available datasets for their experiments. Intrusion detection using prepared datasets prevents researchers to define customized test conditions or change the type of attack on the industrial systems. Therefore, a tool to create a virtual industrial control system cable to perform cybersecurity research will be a great asset for researchers.

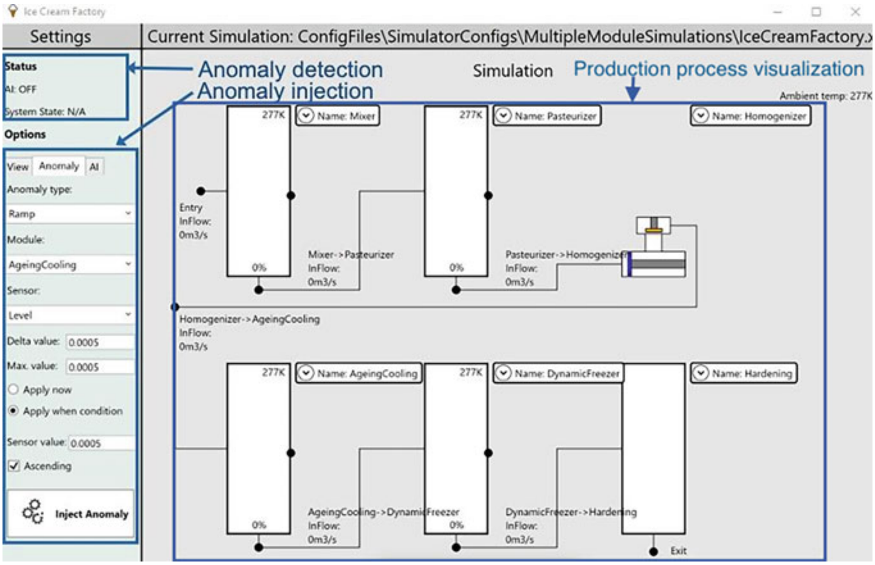
3.3.1 ICSSIM–A Framework for Building Industrial Control Systems Security Testbeds

The importance of studying cyberattacks, testing ICSs, and creating defense mechanisms cannot be overstated. However, due to safety concerns, conducting these studies on operational ICSs is often not allowed. One solution is to use a small-scale pilot ICS as a test environment, but these testbeds are not widely accessible, can be time-consuming and expensive to build, and require hardware knowledge to overcome various technical issues [20], as mentioned in Sect. 3.3. As a result, many security researchers, particularly those using machine learning methods for attack detection, resort to using existing datasets for their experiments. But using these datasets limits the ability to customize test conditions or change the attack type on industrial systems. Thus, a tool to create a test environment for ICS security would be valuable for researchers and practitioners alike.

After thorough analysis, we have reviewed numerous publications introducing various testbeds and simulation tools for ICS [6, 11]. Based on this review, we have compiled a list of essential features for a testbed. As a significant contribution to the EU-funded InSecTT project, we introduce ICSSIM [13], a framework designed to facilitate the creation of virtual testbeds for in-depth exploration of diverse cyber threats and network attacks within ICSs. ICSSIM has a set of base classes for modeling ICS components and communication. Notably, this framework allows deploying its simulated components onto hardware such as Raspberry Pi and containerized platforms like Docker. Furthermore, ICSSIM offers comprehensive support for physical process modeling, incorporating both software and hardware-in-the-loop simulation techniques.

The primary objective of ICSSIM is to expedite the development of ICS components, resulting in the creation of versatile, reproducible, and cost-effective ICS testbeds that capture real-world details. The efficacy of ICSSIM becomes readily apparent through the practical demonstration of its capabilities. In this context, we have leveraged ICSSIM to construct a testbed, showcasing its versatility in simulating various cyberattacks. We have implemented several attack scenarios within this environment, including Man in the Middle attack (MITM), DDoS attack, reconnaissance attack, false data injection using MITM, replay attack, and command injection by considering various attack scenarios.

We also published the ‘ICS-Flow’ dataset [15], created through sample security experiments in this environment. We presented the ICS-Flow dataset for ML-based IDS evaluation through supervised and unsupervised methods. The dataset was generated using the ICSSIM simulator, which emulates the ICS of a bottle-filling factory in a ‘Hardware in the Loop’ simulation and utilizes realistic industrial protocols such as Modbus. Network data and process state variable logs were recorded during normal operations and during four common cyberattacks. The ICS-Flow dataset includes raw network data, network flow data, process variable logs, and attack logs for ML-based anomalous record detection and sequence detection. We demonstrate the effectiveness of the ICS-Flow dataset by applying decision tree, random forest, and artificial neural network models for anomaly and attack detection, showing that the dataset can be effectively utilized for training ML models for intrusion detection.


Current Simulation: ConfigFiles\SimulatorConfigs\MultipleModuleSimulations\IceCreamFactory.x

Status
AI: OFF
System State: N/A

Options

Anomaly type: Ramp
Module: AgeingCooling
Sensor: Level
Delta value: 0.0005
Max. value: 0.0005
○ Apply now
● Apply when condition
Sensor value: 0.0005
☑ Ascending

Fig. 4 User interface of simulation environment for modular ice cream factory example, including: production process visualisation, anomaly injection and anomaly detection [33]

3.3.2 Modular Ice Cream Factory Simulator and Anomaly Injections

ABB and MDU developed a simulation environment to represent a modular manufacturing system [33, 34]. This environment is composed of simulated sensors and actuators and was built using the modular automation design strategy [30, 48, 69]. It allows easy configuration and combination of simple modules into complex production processes. Sensor and actuator signals are exchanged with controllers using the Message Queue Telemetry Transfer (MQTT) protocol [1]. Synchronization of the overall process is performed using high-level recipe orchestration, utilizing OPC UA [49] client/server communication. The simulation environment is presented in detail in [34]. Visualisation of the current state of the simulated process is provided as a simple Graphical User Interface (GUI) that contains visual representations of modules, their interconnections, and current values of the parameters. Additionally, there is functionality that enables users to manually inject different types of anomalies into analog sensors during the production process.

An example of the use of the simulation environment is a modular ice cream factory, in which, as shown in Fig. 4, the simulation engine is configured to simulate the behavior of six separate modules: a mixer, a pasteurizer, a homogenizator, an ageing and cooling module, a dynamic freezer, and a packaging module.

This setup was used to create an open dataset named Modular ice cream factory Dataset on Anomalies in Sensors (MIDAS) [45] that contains various anomalies in analog sensors and can be used for ML research in modular manufacturing systems. The anomalies were injected using a script that automatises the anomaly injection process, injecting anomalies to modify values of different sensors during different stages of the simulated process, with different values of parameters for a specific anomaly that is injected. The anomaly injection occurs in a randomly selected moment, either with an increasing or a decreasing trend, when the sensor value changes. The dataset contains a separate CSV file for each of 1000 runs, where 258 runs represent normal behaviour and 742 runs contains anomalies, with three different types of anomalies (Freeze, Step, and Ramp). It has 36,124,859 instances, where 49.67% instances that represent normal behaviour, and 50.33% instances contain anomalies. The distribution of instances between normal behaviour and each anomaly were Normal (50%), Freeze (15%), Ramp (17%), and Step (18%).

The generated dataset was used to evaluate different supervised ML algorithms (Logistic Regression, Decision Tree, Random Forest, and MLP, as well as a time-series ML algorithm (Long-Short Term Memory–LSTM)) for two different problems: anomaly detection and anomaly classification. Experiments showed that using the temporal information into LSTM network performed better than the non-temporal ML algorithms [43]. We decided to integrate the LSTM model into the demonstrator to provide reliable anomaly detection functionality.

3.3.3 Virtualization and Emulation of Industrial Network Topology–Westermo

Many of Westermo's products–switches and routers for harsh industrial settings–run the Westermo operating system (WeOS). In order to verify that WeOS is operating as expected after code changes and extensions, a significant effort has been invested in automated software testing of the devices [63].

In the ideal case, software has a low fail rate and high reliability once it is developed, see blue curve in Fig. 5. However, in reality, there are typically updates during the useful life of the software (red curve) [5, 51]. There is thus a strong need for quality assurance, software testing and preferably automated software testing.

When testing embedded systems, at some point one has to run it on physical hardware to verify timing and other non-functional characteristics related to hardware and the software-hardware integration. For this purpose, several physical test systems have been constructed at Westermo. There is a significant amount of physical test equipment, it weighs more than a tonne, requires redundant air conditioning and fills several large rooms. Since some years, many of the pure software parts of WeOS can run in a virtual environment, which enables testing of significant parts of WeOS without any hardware. There are thus a number of test systems constructed where the physical devices have been replaced with purely virtual digital twins, QEMU has been an important enabler.

A challenge that may occur when developing new software is to get access to hardware that supports it, in particular when new hardware models are developed in parallel with the software development. In Fig. 6 the timing of one software development sub-project from Westermo is illustrated. In blue, we see the trend of failing tests when running WeOS on virtual test systems, and in red the same trend on

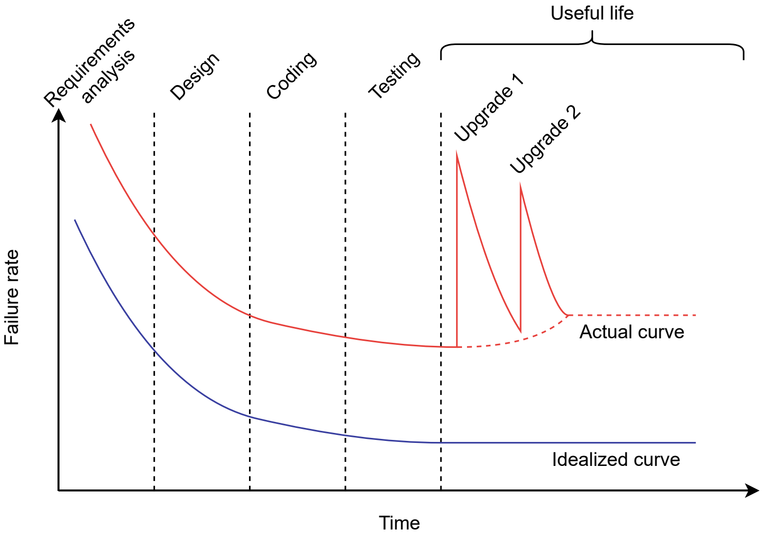


Fig. 5 Typical curve for software reliability (image from [5], used with permission)

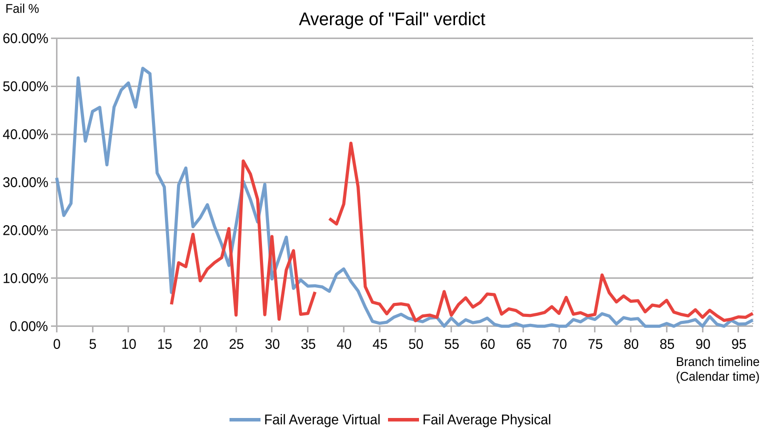


Fig. 6 Ratio of tests that fail, from an experimental development branch, over more than three months. Blue curve illustrates virtual test systems, and red physical test systems (image from [5], used with permission)

test systems with physical devices. There are two relevant observations to be made. First, testing on physical hardware is postponed, in this case only by two weeks–anecdotally, we know that this can be much more. Second, some bugs in the software are only visible on physical hardware (red peak at about 40 days). For this reason, we wished to explore hybrid test systems, where some if not most parts of a test system were virtualized whereas one or a few devices were physical. An overview of how this could be implemented is illustrated in Fig. 7.

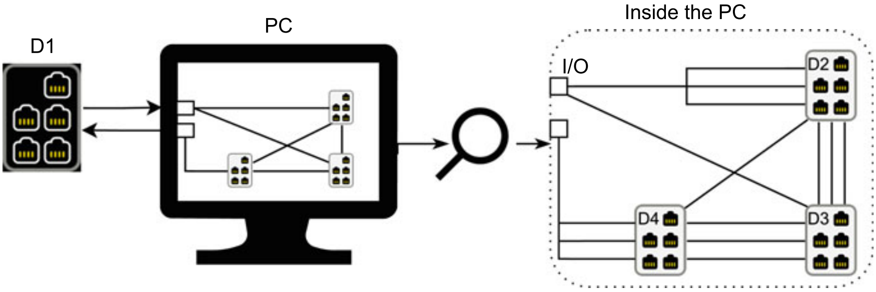


Fig. 7 Example of how a hybrid test system could be created (image from [5], used with permission)

During the InSecTT project, such hybrid test systems were explored with different activities followed by a standardisation and refinement phase. The results indicate that testing software modifications could start earlier, that certain software issues could be reproduced more reliably, when compared with testing in a purely virtual environment, in particular with respect to the reality gap and timing issues. Furthermore, some of the challenges with pure hardware environments are reduced (e.g., the need for hardware is reduced, which has become a problem due to the chip crisis). On the other hand, one can expect test systems with only hardware to more reliably mimic customer settings, therefore, hybrid systems can be expected to be less reliable. Furthermore, industry practitioners expressed concern that virtual-only test systems may lead to false positives when testing, but how more reliable hybrid test systems are when compared to virtual test systems remains to be explored [5].

A second technology Westermo have worked on during the EU-funded InSecTT project, is to run an AI inside a container in a Westermo router, see Fig. 8. For this work, Westermo developed a container feature in some WeOS versions on some hardware products (see A in Fig. 8). Thanks to InSecTT project activities, Westermo now has a rather mature container support based on cgroups which is a common container technology (though not as well known as Docker). Towards the end of InSecTT, we started implementing an AI for our containers (B in Fig. 8). The first step was to explore how well this worked in practice, and the limitations of the resources in the hardware. Preliminary results were mixed, hardware restrictions were acceptable, and many but not all anomalies were detected [23]. In future work, we could explore distributed or federated AI (C), or if a fog or cloud-based AI is better for this industrial context (D).

A third track of work from Westermo in InSecTT is collecting a realistic dataset for supporting AI research. To achieve this, Westermo teamed up with partners (MDU, RISE and TietoEVRY), to define and implement a data collection scenario. In our previous experience, when releasing a dataset [64] from the parallel research project AIDOaRt, we set up information security risk workshops. This practice was also used in InSecTT, and the network traffic dataset has now been published for the general public on GitHub [64].

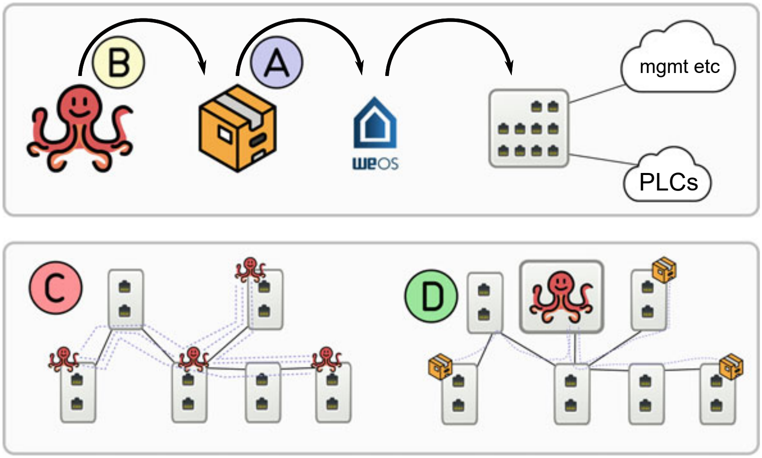


Fig. 8 Overview of AI in container implementation (top row), and possible extensions (bottom)

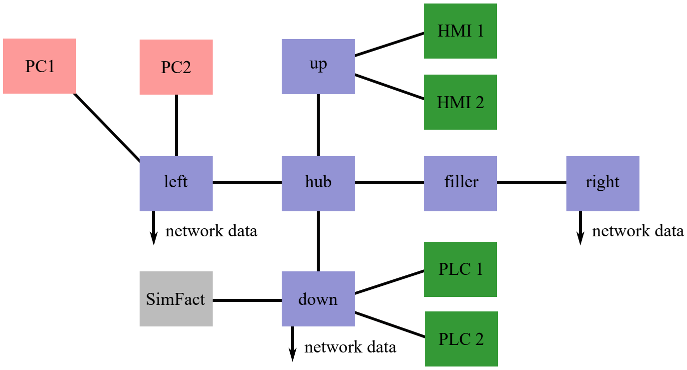


Fig. 9 Network topology used during data collection [64]

For the data collection, a test system with six Westermo devices, five Raspberry Pi devices and two laptops has been built. On the Raspberry Pi devices, we ran ICSSIM (see Sect. 3.3.1), two had the role of HMIs, two were PLCs, and the fifth simulated the physical world, see Fig. 9. Using port forwarding we redirected all network traffic from three Westermo devices to a PC where it was collected with tcpdump. During the data collection we conducted seven types of changes, misconfigurations or attacks, intermixed with periods where the network was not disturbed. The disturbances were:

1. Misconfigured IP: a random WeOS device has its IP address changed. Instead of a correct one, like 192.168.0.1, we would swap the second and third octet, into 192.0.168.1. After some time, the address would be corrected.
2. Duplicated IP: a random WeOS device is given the same IP address as another WeOS device. After some time, the address is corrected.

3. Acceptable SSH traffic: Using SSH with a correct username and password, we log in and check the contents of a log file, and log out.
4. Password guessing: By using usernames and passwords based on the Mirai Malware[1] we generate many parallel attempts to log in over SSH.
5. Port scan: Using nmap, we scan the ports of one or several devices in the network.
6. MITM: Using the attack toolbox of ICSSIM, we launch a MITM attack and rewrite values in modbus packets.

By including both human misconfiguration, attacks, as well as benign disturbances, we aim at supporting work on distributed AI and anomaly detection with this dataset. We speculate that it can also support work on where an anomaly detection system ought to be placed in the network (e.g., close to PLCs).

The Westermo network traffic dataset was used to evaluate ML algorithms with centralized, local, and federated approach for anomaly detection in network data [14]. We used different supervised ML algorithms in local and centralized approach, including: Logistic Regression, SVM, ANN, KNN, Decision Tree and RF. RF and ANN exhibited superior performance and were implemented in a federated setup. The experiments showed that federated version outperforms the local models, and achieves comparable or even superior results compared to the centralized model, while ensuring data privacy and confidentiality of sensitive information.

3.4 Safety and Security Analysis for AGV Platooning

There is an increasing trend of Automated Guided Vehicles (AGVs) and platooning. AGVs are an integral part of the Industry 4.0 [26]. Their platooning tends to improve overall safety, security and operational efficiency of production site. The published studies on platooning focus mainly on the design of technical solutions of automotive domain, but not considered the AGV platooning in production sites and Industry 4.0. We presented a platooning strategy which not only provides a means to control overall traffic flow at production site and reduce resource usage but also manage transportation risks in a dynamic manner. However, the automation, digitalisation and connectivity of AGVs with each other or with the infrastructure significantly pose the safety and security issues [25, 26]. A security attack or a single failure in one AGV could lead to unsafe behaviour of whole platoon that can potentially harm humans (injuries or even deaths) or create damages to machines, property or the environment. The safety-critical systems can only be regarded as safe if they are also secure. The literature highlights a dearth of comprehensive research on different aspects of vehicle platooning including safety and security analyses [7]. There is a need for comprehensive studies to deal with the situations such as joining and leaving platoon in production sites, connectivity with fog and cloud servers, system or component failures, security attacks, and influencing environmental factors.

[1] https://github.com/jgamblin/Mirai-Source-Code.

Established safety and security analyses methods such as the hazard and threat analyses are performed during design and development phase by using the Hazard and Operability (HAZOP) and Threat and Operability (THROP) techniques, respectively. We considered the interactions of collaborative autonomous systems with one another, and to the fog controller that, in-turn, interact with the cloud infrastructure. To perform the HAZOP and THROP analyses we establish a list of guide words (e.g., no/not, false/fake, incorrect, increase/exceed, unavailable, unintended, exploit, other than, etc.) and systems parameters/functions, such as sensors, actuators, communication and connectivity (e.g., WiFi, 4G/5G, IoT devices, fog) and type of messages (e.g., request, response, and command). As the collaborative autonomous systems underlie the need for dynamic risk management, the data is gathered to monitor systems operations, identify unexpected or incorrect behaviour, evaluate the potential implications and trigger control actions to resolve them.

We presented the overall approach for a fault- and threat tolerant platooning for materials transportation in production environments with detailed analysis in [27].

4 Novelty and Applicability of Proposed Technologies

Adaptable access control rule inference and enforcement are based on industrial standards. The technologies used for the enforcement architecture uses available standardized components, e.g., OPC UA for the communication stack, and JWT for access tokens, which makes the suggested solution applicable in any domain utilizing these standards. The publications both present novel material, and some of the enforcement architecture mechanisms developed are currently being evaluated for potential IP protection.

We have proposed a structured approach for generation of datasets on sensor anomalies in manufacturing context (both manual and automatic injection of anomalies supported). The architecture of the use case and the various simulation modules are following the modular automation principles, thus allowing easy evolution and adaption of the systems and related validation efforts. This also helps in quick security analysis through focused testing efforts. Our ML algorithms comparisons are based on a larger set of algorithms applied on multiple well-known anomaly datasets.

The ICSSIM is built based on container technology, which means that ICS components run on isolated operating system kernels. Moreover, simulated containerized components such as PLCs, HMIs, or HW in a loop (HIL) processes could be run on simulation engines such as GNS3 or emulation environments such as Docker containers, or they could totally be replaced with physical entities. It also has a stub for SW simulation of HIL to simulate the control process. Moreover, ICSSIM has interfaces to communicate with the HW through the file to use real HW for the process. ICSSIM can be used for simulation of any ICS.

This creation of a virtual network topology by Westermo is enhanced by addition of their test framework that can be used to test any functionality in an industrial network topology, not limited to ICSs. This can also run automatically with follow up of test results. A true first step of a digital twin (DT) that can be valuable for ensuring network security in diverse manufacturing setups is to enable even online detections and mitigations.

5 Conclusions and Future Perspectives

In this chapter we have presented some of the research outputs and technology contributions realised as part of the EU-funded InSecTT project. The presented works show fruits of extended industry-academia collaboration to solve important challenges related to safety and security in manufacturing environments. The industrial partners are keen on exploiting the above results as evidenced by patent applications and tools being inducted. We are currently continuing the work in several directions such as studies to include wider coverage of ML models, integrating federated learning model into resource-constrained network switches and further demonstrations in other contexts such as smart-cities.

References

1. MQTT Version 5.0. OASIS Standard, March 2019. Edited by Andrew Banks, Ed Briggs, Ken Borgendale, and Rahul Gupta
2. O-PAS Standard, Version 2.0: Part 1-Technical Architecture Overview. Open Group Preliminary Standard (P201-1), The Open Group (Feb. 2020)
3. Abdi, H., Williams, L.J.: Principal component analysis. In: Wiley Interdisciplinary Reviews: Computational Statistics, vol. 2, no. 4, pp. 433–459 (2010)
4. Abedin, M., Alam Siddiquee, K.N.E., Bhuyan, M.S., Karim, R., Hossain, M.S., Andersson, K., et al.: Performance analysis of anomaly based network intrusion detection systems. In: 43nd IEEE Conference on Local Computer Networks Workshops (LCN Workshops), Chicago, 1–4 Oct. 2018, pp. 1–7. IEEE Computer Society (2018)
5. Alhasan, W.: Evaluating Challenges, Benefits, and Dependability of Virtual and Physical Testing of Embedded Systems Software. Master's thesis, Mälardalen University (2022)
6. Ani, U.P.D., Watson, J.M., Green, B., Craggs, B., Nurse, J.R.C.: Design considerations for building credible security testbeds: perspectives from industrial control system use cases. J. Cyber Secur. Technol. **5**(2) (2021)
7. Axelsson, J.: Safety in vehicle platooning: a systematic literature review. IEEE Trans. Intell. Transp. Syst. **18**(5), 1033–1045 (2017)
8. Bace, R., Mell, P.: Intrusion detection systems. National Institute of Standards and Technology (NIST), Technical Report 800-31 (2001)
9. Behera, S., Pradhan, A., Dash, R.: Deep neural network architecture for anomaly based intrusion detection system. In: 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN), pp. 270–274. IEEE (2018)
10. Buczak, A.L., Guven, E.: A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun. Surv. Tutor. **18**(2) (2015)
11. Conti, M., Donadel, D., Turrin, F.: A survey on industrial control system testbeds and datasets for security research (2021). arXiv:2102.05631
12. Davis, J., Edgar, T., Porter, J., Bernaden, J., Sarli, M.: Smart manufacturing, manufacturing intelligence and demand-dynamic performance. Comput. Chem. Eng. **47**, 145–156 (2012)

13. Dehlaghi-Ghadim, A., Balador, A., Moghadam, M.H., Hansson, H., Conti, M.: Icssim-a framework for building industrial control systems security testbeds. Comput. Ind. **148**, 103906 (2023)
14. Dehlaghi-Ghadim, A., Markovic, T., Leon, M., Söderman, D., Strandberg, P.E.: Federated learning for network anomaly detection in a distributed industrial environment. In: 2023 22nd IEEE International Conference on Machine Learning and Applications (ICMLA). IEEE (2023)
15. Dehlaghi-Ghadim, A., Moghadam, M.H., Balador, A., Hansson, H.: Anomaly detection dataset for industrial control systems (2023). arXiv:2305.09678
16. Farnaaz, N., Jabbar, M.A.: Random forest modeling for network intrusion detection system. Proc. Comput. Sci. **89**, 213–217 (2016)
17. Fu, Y., Lou, F., Meng, F., Tian, Z., Zhang, H., Jiang, F.: An intelligent network attack detection method based on rnn. In: 2018 IEEE Third International Conference on Data Science in Cyberspace (DSC), pp. 483–489. IEEE (2018)
18. Ghurab, M., Gaphari, G., Alshami, F., Alshamy, R., Othman, S.: A detailed analysis of benchmark datasets for network intrusion detection system. Asian J. Res. Comput. Sci. **7**(4), 14–33 (2021)
19. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016). http://www.deeplearningbook.org
20. Green, B., Lee, A., Antrobus, R., Roedig, U., Hutchison, D., Rashid, A.: Pains, gains and PLCs: ten lessons from building an industrial control systems testbed for security research. In: 10th {USENIX} Workshop on Cyber Security Experimentation and Test {CSET}, vol. 17 (2017)
21. Hermann, M., Pentek, T., Otto, B.: Design principles for industrie 4.0 scenarios. In: Proceedings of the Hawaii International Conference on System Sciences, vol. 2016, pp. 3928–3937. IEEE (Mar. 2016)
22. IEC 62443 security for industrial automation and control systems. Standard, International Electrotechnical Commission, Geneva, CH, 2009-2018
23. Ingletto, G., Lidholm, P.: Anomaly Detection for Network Traffic in a Resource Constrained Environment. Master's thesis, Mälardalen University (2023)
24. Ingre, B., Yadav, A.: Performance analysis of NSL-KDD dataset using ANN. In 2015 International Conference on Signal Processing and Communication Engineering Systems, pp. 92–96. IEEE (2015)
25. Jaradat, O., Sljivo, I., Habli, I., Hawkins, R.: Challenges of safety assurance for industry 4.0. In: 13th European Dependable Computing Conference, EDCC Geneva, Switzerland (2017)
26. Javed, M.A., Muram, F.U., Hansson, H., Punnekkat, S., Thane, H.: Towards dynamic safety assurance for Industry 4.0. J. Syst. Archit. **114**, 101914 (2021)
27. Javed, M.A., Muram, F.U., Hansson, H., Punnekkat, S., Hansson, H.: Safe and secure platooning of automated guided vehicles in industry 4.0. J. Syst. Archit. **121**, 102309 (2021)
28. Kim, J., Kim, J., Kim, H., Shim, M., Choi, E.: CNN-based network intrusion detection against denial-of-service attacks. Electronics **9**(6), 916 (2020)
29. Kumar, V., Chauhan, H., Panwar, D.: K-means clustering approach to analyze NSL-KDD intrusion detection dataset. Int. J. Soft Comput. Eng. (IJSCE) ISSN, 2231–2307 (2013)
30. Ladiges, J., et al.: Integration of modular process units into process control systems. IEEE Trans. Ind. Appl. **54**(2), 1870–1880 (2018)
31. Lasi, H., Fettke, P., Kemper, H.-G., Feld, T., Hoffmann, M.: Industry 4.0. Bus. Inf. Syst. Eng. **6**(4), 239–242 (2014)
32. Leander, B., Johansson, B., Lindström, T., Holmström, O., Nolte, T., Papadopoulos, A.V.: Dependability and security aspects of network-centric control. In: 28th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). IEEE (2023)
33. Leander, B., Markovic, T., Leon, M.: Enhanced simulation environment to support research in modular manufacturing systems. In: IECON, pp. 1–6. IEEE (2023)
34. Leander, B., Marković, T., Čaušević, A., Lindström, T., Hansson, H., Punnekkat, S.: Simulation environment for modular automation systems. In: IECON (2022)
35. Leander, B., Čaušević, A., Hansson, H., Lindström, T.: Toward an ideal access control strategy for industry 4.0 manufacturing systems. IEEE Access **9** (2021)

36. Leander, B., Čaušević, A., Lindström, T., Hansson, H.: A questionnaire study on the use of access control in industrial systems. In: IEEE International Conference on Emerging Technologies and Factory Automation (ETFA) (2021)
37. Leander, B., Čaušević, A., Lindström, T., Hansson, H.: Access control enforcement architectures for dynamic manufacturing systems. In: 2023 IEEE 20th International Conference on Software Architecture (ICSA), pp. 82–92 (2023)
38. Leander, B., Čaušević, A., Lindström, T., Hansson, H.: Evaluation of an OPC UA-based access control enforcement architecture. In: ESORICS 2023 International Workshops: CyberICPS (2023)
39. Leon, M., Markovic, T., Punnekkat, S.: Comparative evaluation of machine learning algorithms for network intrusion detection and attack classification. In: 2022 International Joint Conference on Neural Networks (IJCNN), pp. 01–08. IEEE (2022)
40. Leon, M., Markovic, T., Punnekkat, S.: Feature encoding with autoencoder and differential evolution for network intrusion detection using machine learning. In: Proceedings of the Genetic and Evolutionary Computation Conference Companion (2022)
41. Liao, H.-J., Richard Lin, C.-H., Lin, Y.-C., Tung, K.-Y.: Intrusion detection system: a comprehensive review. J. Netw. Comput. Appl. **36**(1), 16–24 (2013)
42. Lu, Y.: Industry 4.0: a survey on technologies, applications and open research issues. J. Ind. Inf. Integr. **6**, 1–10 (2017)
43. Markovic, T., Dehlaghi-Ghadim, A., Leon, M., Balador, A., Punnekkat, S.: Time-series anomaly detection and classification with long short-term memory network on industrial manufacturing systems. In: 18th Conference on Computer Science and Intelligence Systems FedCSIS. IEEE (2023)
44. Markovic, T., Leon, M., Buffoni, D., Punnekkat, S.: Random forest based on federated learning for intrusion detection. In: Artificial Intelligence Applications and Innovations: 18th IFIP WG 12.5 International Conference, AIAI 2022, Hersonissos, Crete, Greece, June 17–20, 2022, Proceedings, Part I, pp. 132–144. Springer (2022)
45. Markovic, T., Leon, M., Leander, B., Punnekkat, S.: A modular ice cream factory dataset on anomalies in sensors to support machine learning research in manufacturing systems. IEEE Access **11**, 29744–29758 (2023)
46. Mazhar Rathore, M., Ahmad, A., Paul, A.: Real time intrusion detection system for ultra-high-speed big data environments. J. Supercomput. **72**(9) (2016)
47. Mittal, S., Khan, M.A., Wuest, T.: Smart manufacturing: characteristics and technologies. In: Harik, R., Rivest, L., Bernard, A., Eynard, B., Bouras, A. (eds.) Product Lifecycle Management for Digital Transformation of Industries, pp. 539–548, Cham, 2016. Springer International Publishing (2016)
48. NAMUR Working Group 1.12. NE 148 Automation Requirements relating to Modularisation of Process Plants. NAMUR-recommendation (2013)
49. OPC unified architecture: Standard, IEC, Geneva, CH (2016)
50. Pervez, M.S., Farid, D.M.: Feature selection and intrusion classification in NSL-KDD cup 99 dataset employing SVMs. In: International Conference on Software, Knowledge, Information Management and Applications (SKIMA), pp. 1–6. IEEE (2014)
51. Quyoum, A., Dar, M.-U.-D., Quadri, S.M.K.: Improving software reliability using software engineering approach-a review. Int. J. Comput. Appl. **10**(5), 41–47 (2010)
52. Radonjić, I., Bašić, E., Leander, B., Marković, T.: An authorization service supporting dynamic access control in manufacturing systems. In: IEEE 9th World Forum on Internet of Things (2023)
53. Revathi, S., Malathi, A.: A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection. Int. J. Eng. Res. Technol. (IJERT) **2**(12), 1848–1853 (2013)
54. Roy, B, Cheung, H.: A deep learning approach for intrusion detection in internet of things using bi-directional long short-term memory recurrent neural network. In: International Telecommunication Networks and Applications Conference (ITNAC), pp. 1–6. IEEE (2018)

55. Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach. Pearson Education Limited, Malaysia (2016)
56. Saltzer, J., Schroeder, M.: The protection of information in computer systems. Proc. IEEE **63**, 1278–1308 (1975)
57. Sandhu, R., Ranganathan, K., Zhang, X.: Secure information sharing enabled by trusted computing and PEI models. In: Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security, ASIACCS'06, vol. 2006, pp. 2–12 (2006)
58. Sandhu, R.S., Samarati, P.: Access control: principle and practice. IEEE Commun. Mag. **32**(9), 40–48 (1994)
59. Scarfone, K., Mell, P., et al.: Guide to intrusion detection and prevention systems (idps). NIST Special Publication, (800-94) (2007)
60. Shrivas, A.K., Dewangan, A.K.: An ensemble model for classification of attacks with feature selection based on KDD99 and NSL-KDD data set. Int. J. Comput. Appl. **99**(15), 8–13 (2014)
61. Storn, R., Price, K.: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J. Global Optim. **11**(4) (1997)
62. Stouffer, K., Pillitteri, V., Lightman, S., Abrams, M., Hahn, A.: Guide to Industrial Control Systems (ICS) Security NIST Special Publication 800-82 Revision 2. NIST Special Publication 800-82 rev 2, pp. 1–157 (2015)
63. Strandberg, P.E.: Automated System-Level Software Testing of Industrial Networked Embedded Systems. Ph.D. thesis, Mälardalen University (2021)
64. Strandberg, P.E., Söderman, D., Dehlaghi-Ghadim, A., Leon, M., Markovic, T., Punnekkat, S., Moghadam, M.H., Buffoni, D.: The Westermo network traffic data set. Data in Brief **50**, 109512 (2023)
65. Survey, A., Wang, S., Fernando Balarezo, J., Kandeepan, S., Al-Hourani, A., Gomez Chavez, K., Rubinstein, B.: Machine learning in network anomaly detection. IEEE Access **9**, 152379–152396 (2021)
66. Tuan, T.A., Long, H.V., Son, L.H., Kumar, R., Priyadarshini, I., Son, N.T.K.: Performance evaluation of Botnet DDoS attack detection using machine learning. Evolut. Intell. **13**(2), 283–294 (2020)
67. Williams, T.J.: The Purdue enterprise reference architecture. Comput. Ind. **24**(2), 141–158 (1994)
68. Yin, C., Zhu, Y., Fei, J., He, X.: A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access **5**, 21954–21961 (2017)
69. ZVEI-German Electrical and Electronic Manufacturers' Association. Module-based production in the process industry-Effects on automation in the "Industrie 4.0" environment. White Paper (Mar. 2015)
70. ZVEI-German Electrical and Electronic Manufacturers' Association. Process INDUSTRIE 4.0: The Age of Modular Production. White Paper, Frankfurt (2019)