Final Report: Framework for Forensic Analysis of Cyber-Physical Attacks: Correlating Digital Traces with Physical Outcomes

School of Computer Science and Informatics

Author: Jake Palmer

Moderator: Liam Turner

Supervisor: Eirini Anthi

Table of Contents

Abstract

The integration of Cyber-Physical Systems in many different industries has greatly improved efficiency and productivity across various sectors and has allowed automated processes to be streamlined in fields such as manufacturing, healthcare, transportation, and agriculture. Unfortunately, this combination of the physical and digital domains also poses new and unexplored challenges for both cyber-physical system security and digital forensic investigation processes. The increased prevalence of attacks on these systems necessitates a need for more robust security measures and investigation methodologies. While research on these in the context of cyber-physical systems does exist, limited work focuses on correlating the unique characteristics of the digital and physical evidence that can be collected when a breach occurs. This project aims to address this by proposing a comprehensive framework for guiding the forensic analysis of these systems and then outlining techniques for correlating the digital traces of the attack with its physical consequences. The framework is designed to be adaptable to all cyber-physical systems, and it can help to identify the attack vectors of a system breach, its scope, and its impact on the surrounding environment. For the completion of this framework, I outlined methods for evidence collection and preservation, the identification of physical outcomes, correlation analysis, and comprehensive forensic reporting, while adhering to ethical and legal standards. I then tested this methodology using data collected from both gas and water pipeline systems, incorporated machine learning techniques for categorising new attacks, and recorded the results.

1. Introduction

Cyber-Physical Systems (CPS) have been a catalyst for redefining societal functionality through the innovations of Industry 4.0 [22]. These are distributed systems that aim to seamlessly integrate both the physical and digital domains promising to improve the precision and efficiency of everyday processes, including energy production, manufacturing, and the advent of autonomous driving [24]. They encompass a variety of components including sensors and actuators which enable them to gather real-time data from the physical environment. This ability to collect and process physical data facilitates high process optimisation and proactively identifies and resolves potential issues such as predicting equipment failure or anticipating environmental anomalies [12]. Unfortunately, like all digital systems, and through this interconnection between the physical and digital domains CPS also introduces a large spectrum of security concerns, notably a significant risk of data breaches and other cyber-attacks. Additionally, these are difficult to detect until they substantially impact the CPS and its environment, causing significant damage not only to the system itself but also to buildings and even potential fatalities to personnel in industrial settings [9].

To further complicate this problem, it is also more challenging to perform forensic investigations on these systems due to the increased number of sensors and actuators on CPS and the integration of the digital and physical environments [7]. This increased complexity stems from the correlation of physical and digital information and the sheer volume of data collected and produced by CPS. A better approach to forensic investigations of CPS is required for their continued operation within crucial sectors of modern society. Traditionally, digital forensics have predominantly focused on just the digital evidence that has been collected and analysed from devices such as network logs and data stored within volatile memory, without exploring the physical impact of the damages. However, due to the interconnected nature of cyber-physical systems, a more comprehensive approach must be developed for cyber-physical systems to better understand the physical consequences of the attack. Investigating a cyber-physical system breach would expand this approach to include the interactions of the sensors and actuators and the physical consequences of the attack based on this data.

This framework aims to address this problem by providing an approach for conducting forensic analysis of these systems. It aims to identify the scope of the breach quickly and accurately by emphasising the evidence that is directly correlated to an attack's digital and physical consequences. This approach to addressing attacks on cyber-physical systems offers a holistic perspective of the overall impact caused by the breach. It can therefore better inform forensic investigators about the severity of the attack and how to mitigate any damage caused. Additionally, this framework also aims to incorporate traditional digital forensic methodologies, while recognising the need for an evolved approach due to the complexities of cyber-physical systems. Throughout this report, I will discuss my rationale behind producing the framework and evaluate its performance through the use of data taken from existing cyber-physical systems, adapting the approach if needed based on the results.

2. Background

2.1 Cyber-Physical Systems

Cyber-physical systems (CPS) constitute an entire subset of embedded systems that integrate physical and computational processes [25]. These are designed as part of a network of devices including sensors and actuators, which allow the system to interact with and monitor the surrounding physical environment using feedback loops [14]. This integration between the physical and digital worlds means that they can automate and control physical processes, based on data collected by the sensors. CPS have far-reaching applications across many sectors and are used in everyday systems such as transportation, manufacturing, and energy generation [5]. They are particularly useful in applications

such as autonomous driving, where the system needs to quickly adapt to the surrounding environment to avoid obstacles and pedestrians. This is also useful to increase the fault tolerance of essential systems; by analysing the data from the sensors these systems can utilise advanced decision-making techniques to prolong the lifespan of its hardware [14].

In terms of their architecture, Cyber-Physical Systems typically consist of three main layers: the physical layer used for sensing and actuation, a decision-making layer for deciding in advance what the system needs to do to control the environment, and the cyber layer used for processing the data to and from sensors and actuators and allowing the system to communicate with itself and other devices. This design of these systems ensures that it can adapt to changing physical conditions and allows it to be largely scalable.

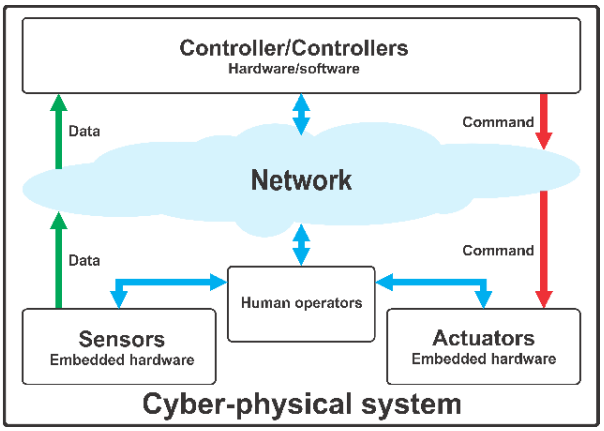


Figure 1: A diagram illustrating the architecture of cyber-physical systems. [25]

The main component of Cyber-Physical Systems beyond the sensors and actuators is the PLC (Programmable Logic Controller), particularly in the context of Automation, Transport and Industrial Control Systems [1] which require robust control and coordination of various processes, as well as real-time access to this information. These typically use a paradigm known as ladder logic, which mimics regular logical circuits except for being specifically designed for implementing sequential functions that allow the PLC to respond to real-time inputs from sensors and actuators. These are then usually managed by a SCADA (Supervisory Control and Acquisition) system, which provides a centralised interface for monitoring and controlling the PLC.

2.2 Types of Cyber-Physical Systems

2.2.1 Industrial Control Systems (ICS)

Industrial Control Systems (ICS) is a type of Cyber-Physical System that are used to monitor and control various industrial processes in many sectors including manufacturing and power delivery systems [34]. These typically consist of three major components:

- Supervisory Control and Data Acquisition (SCADA) Systems: SCADA Systems monitor and control industrial processes by collecting data from sensors and sending different commands to actuators. These often include Human Machine Interfaces (HMIs) which allow the operators to interact with and control processes directly, enabling them to intervene if necessary.
- Distributed Control Systems (DCS): DCS is used in manufacturing and process control industries to coordinate and automate complex processes across multiple locations within the ICS. These consist of networked controllers to communicate with the sensors and actuators and help to regulate processes within the industrial control system.
- Programmable Logic Controllers (PLCs): PLCs are programmable computers that are used to control machinery and automate industrial processes using the paradigm of ladder logic, which executes commands sequentially within the system, allowing for simple automation.

2.2.2 Smart Grids

Smart Grids are advanced electrical power and monitoring systems that integrate cyber-physical system technologies to enhance the overall efficiency and reliability of energy storage and distribution networks [5]. These use cyber-physical technologies to allow real-time monitoring of energy distribution. These typically consist of the following components:

- Advanced Metering Infrastructure (AMI): Smart meters including those installed at consumer premises and infrastructure facilitate two-way communication between energy providers and distributors, allowing the accurate monitoring of energy consumption over the Internet.
- Distributed Energy Resources (DERs): This is the integration of renewable energy sources such as solar panels and wind turbines into the grid, which enables decentralised energy generation and makes the smart grid more flexible.
- Grid Automation and Control: This is completed by combining sensors and actuators (mainly switches) and smart grid analytics to enable the dynamic management of fault detection, improving reliability.
- Demand Response Systems (DR): These allow consumers to adjust their electricity usage in response to changes in price signals or grid conditions, allowing them to save money while grid usage is lower.

2.2.3 Cyber-Physical Healthcare Systems

These integrate digital and physical components to enhance healthcare delivery and allow for greater accuracy and precision when monitoring patients [21]. These systems combine medical devices and connect them using a network of sensors and actuators to enable real-time vital sign monitoring and more personalised treatment that can be accessed and controlled remotely, facilitating improved disease management, containment, and analysis. Common features of these systems include:

- Medical Devices and Sensors: When equipped with CPS capabilities these allow the real-time centralised remote monitoring of patient vital signs which can aid investigations into health conditions and can facilitate better disease management and containment.
- Health Information Systems: These contain the data collected from the medical sensors and can also include electronic copies of patient records, and medical imaging records and facilitate the efficient retrieval of any healthcare data for a given patient, provided it is all stored using that system.
- Telemedicine and Remote Healthcare: These systems allow for remote diagnostics and virtual consultations using healthcare data.

2.3 Digital Forensics

Digital forensics is a rapidly developing field and is the discipline of collecting, processing, and analysing digital evidence for use in digital forensic investigations [20]. This is usually the examination of devices such as desktop computers, laptops, and mobile phones to retrieve data correlated to criminal activities and using specialised software to analyse this data to build a criminal case. As cyber-attacks are becoming more prevalent and complex, new approaches to digital forensics have become increasingly vital, not only for uncovering evidence but also for understanding and mitigating damage posed by cyber-attacks [3, 4]. According to [3], a digital forensics incident response consists of five main requirements: identification, collection, examination, analysis, and presentation. This is a standard operating procedure for any forensic investigation and each stage of this process must remain forensically sound and must preserve the integrity of any evidence collected during the investigation. Investigating criminal activity through digital forensics has become increasingly complex due to several factors. The prevalence of more intricate encryption methods and new security measures has the potential to create barriers between the investigator and the digital evidence,

requiring specialised expertise and software to navigate through the device [7, 20]. Furthermore, without proper data retention practices highly important artifacts stored in volatile memory may be lost forever as they have been overwritten during the investigative process.

In the context of Cyber-Physical Systems, during a digital forensics investigation, it is imperative to not only consider the digital evidence (i.e. network logs, data artifacts in volatile memory, etc) which is traditionally collected during the investigation of other devices but also the physical data concerning the attack such as data received by sensors and actuators as well as any evidence and effects found on the physical environment. The approach to forensics on Cyber-Physical Systems also requires a holistic understanding of how the digital and physical components interact, and how their functionality may be affected during a breach [9]. For instance, in the aftermath of a cyber-attack targeting an industrial control system, network traffic and system logs must be explored as well as the potential impact this may have on physical machinery and processes. A distinctive approach is therefore needed for the forensic analysis of cyber-physical systems to effectively counter and mitigate the damage caused by cyber-attacks due to their increased complexity and the difficulty in detecting breaches within these systems.

2.4 Legal and Ethical Considerations

Upholding all legal and ethical responsibilities throughout any digital forensics investigation is paramount to ensure that all evidence collected is admissible, accurate, and proven to be untampered with during all legal proceedings. Compliance with regulations that govern the collection and storage of evidence is essential to maintain the credibility of the evidence. Any evidence collected during an investigation must also comply with individual and corporate privacy rights and data protection laws, to protect the private data of those involved, and those indirectly involved with the investigation [10].

Furthermore, a fully documented chain of custody must be maintained during the investigation, which needs to be complete with timestamps, the identification number of the evidence item, and verified signatures of authorised personnel who are allowed to process the evidence. This meticulous record-keeping ensures that there is no unauthorised access or evidence tampering, which preserves the admissibility of the evidence. As such, every second of the investigation must be accounted for in a chain of custody document. The use of forensic tools and software is also regulated by legal requirements and ethical standards, ensuring that the acquisition of any data collected is lawful and the use of any open and closed-source software is outlined explicitly.

2.5 Types of Attacks on Cyber-Physical Systems

Cyber-physical systems (CPS) face a variety of threats that are not only common to general internet-connected devices but are also uniquely impactful due to their interconnection with physical processes. This includes attacks like denial-of-service (DoS), which overloads the system with requests making it unable to operate for the end-user. There are also manipulation attacks that exert control over sensor data or controls that aim to deceive the system and make it perform actions that are unsafe or that it wouldn’t do otherwise given the true environmental conditions. This is particularly consequential with autonomous driving, where decisions involving traffic are determined based on the sensor data, if this is incorrect then the system could cause an accident to occur. Additionally, spoofing attacks can also affect cyber-physical systems. These are similar to manipulation attacks but instead of changing the data inputted by the sensors, new data is simply inserted which impersonates the sensor/device.

3.0 Review of Current Approaches and Vulnerabilities

3.1 Literature Review

There have been many other approaches that are tailored to cyber-physical systems forensic investigations and significantly many more approaches to creating more general frameworks for digital forensic investigations which are all aimed at enhancing the detection, analysis, and response to security incidents. Most notable is an approach for forensically analysing and assessing the challenges posed by cloud-based cyber-physical systems [19]. This project introduces a purely conceptual by-design framework that advocates the incorporation of forensic principles during the development phase of cloud-based cyber-physical systems. This, however, does use many ideologies that I aim to use in producing my framework. It aims to streamline the collection, preservation, and analysis of data that may be useful during a forensic investigation and notes that it is unrealistic to expect large organisations to be able to identify and act on all threats to a cyber-physical system, therefore it uses the assumed breach approach which prioritises currently identified vulnerabilities. I will use this to direct my framework towards analysing the most common attacks for cyber-physical systems and then diversify this as and when needed. This approach also accounts for all events that are derived from physical sensors within the cyber-physical system.

A more generalised approach to digital forensics [27] stipulates a guide for digital forensics incident response (DFIR) in the context of IoT (Internet of Things) systems. This approach explores both digital and physical evidence during an investigation and recognises the importance of preserving the physical crime scene before any digital evidence is found, including attempting to reconstruct the full sequence of events, which is useful when exploring both digital and physical evidence in Cyber-Physical Systems. This approach was created in 2004 and therefore is largely outdated for use in newer cyber-physical systems. Nonetheless, its emphasis on preserving both physical and digital evidence concurrently to prevent data loss, particularly in volatile memory, remains highly relevant in modern digital forensic practices. Furthermore, this approach of initiating the analysis with physical evidence can facilitate a more efficient exploration of any evidence collected, as understanding the extent of physical effects can further inform and guide the subsequent digital forensic analysis, which is what my framework will intend to do as well.

Additionally, another approach for a framework that focuses on automotive digital forensics [15] identifies some of the potential information sources that may be used during a digital forensics investigation of automotive vehicles, such as the event data recorder (EDR, or black box) and the electronic control units (ECUs). This framework utilises a blockchain approach that collects all the data from each party involved in the investigation, such as the driver, vehicle manufacturer, maintenance, and law enforcement. This data and events extracted from the EDU are instantly submitted to the blockchain during an accident to both preserve the data and avoid tampering, which ensures evidential integrity throughout the investigation and instantly creates a timeline of events ready for analysis. This paper highlights the fact that the most important data to recover during an investigation on a vehicle is data stored in the control area network bus (CAN bus), which holds data from every sensor within the vehicle (tire pressure, braking, etc) which makes it an invaluable investigation source for reconstructing the events leading up to an accident. In a forensic investigation, this data in conjunction with the physical evidence from the surrounding environment would prove invaluable when determining how the accident occurred as it can be used to fully reconstruct the crime scene, which I will aim to do in my framework, by first analysing the digital data from the sensors, then by exploring the physical evidence.

Another framework that specialises in the investigation of Internet of Things (IoT) systems [27] is designed to periodically back up data that may be useful during a forensic investigation. While this

would be an efficient way to perform an investigation for IoT systems, this section of this approach does not apply to cyber-physical systems. Due to the scale and high volume of data that these systems store, it is largely infeasible to store and process this information indefinitely. However, this approach could be partially applicable to cyber-physical systems in some aspects, for instance, it performs a risk assessment on the system before the investigative process begins, something that I also intend to implement in my approach, and it aligns with best practices for ensuring the thoroughness of the investigation. Additionally, the inclusion of incident detection systems, although not universally applicable to all cyber-physical systems, holds promise for enhancing security and proactive threat mitigation. However, immediate isolation of affected components, as proposed in the framework, may pose challenges in certain cyber-physical environments, particularly those with stringent uptime requirements such as energy distribution and industrial control systems. I plan to integrate certain elements of this framework however, due to the unique characteristics of cyber-physical systems, not every aspect of this approach will be applicable.

Furthermore, an approach for a more general digital forensic framework [29] places a strong emphasis on delineating stages within each of the four pre-defined phases of the investigation: planning, incident response, investigation, and analysis. It achieves this by establishing both high-level control objectives and more detailed control objectives for each stage within the examination. This ensures that each stage of the investigation is comprehensively addressed, using these sub-goals to guide the process and to keep an account of progress within the investigation and facilitates collaboration with several investigators, which would also be useful in large-scale cyber-physical systems and ensures that no critical part of the investigation is overlooked. This framework's emphasis on defining sub-goals for each stage of the investigation could also help to analyse both physical and digital evidence methodically. This approach to digital forensic investigation could also be applied to Cyber-Physical Systems, due to the interconnected nature of these systems it could be modified to include protocols for capturing and preserving both digital data (system log files, etc) and physical evidence sources such as sensor readings.

The final forensic framework I opted to review whilst examining current approaches to this problem involves investigating complex attacks on cyber-physical systems by understanding the goals of the attack [16]. This approach implores that the cyber-physical system being investigated is a closed system (one that is entirely self-contained within its own network) and proposes the analogy that an attack can be seen as an 'epidemic'. The theory is that devices within the system can be sorted into three categories (susceptible, infected, and recovered), and this will determine the spread of the attack once it has been detected. I aim to utilise and expand upon this strategy by stipulating that the investigator must analyse every component within a cyber-physical system, cataloguing the most common vulnerabilities applicable to the components which makes it possible to prioritise not only the investigation into the breach but also makes the recovery of the system quicker once all the evidence has been collected of the attack. One potential limitation while using this goal-oriented approach for investigating an attack on a cyber-physical system is that it may overlook possible sources of evidence due to being solely focused on the goal of the attack, therefore in my approach, I will attempt to address this by finding a balance between using a goal-driven approach and a process-driven approach to ensure that all the evidence is uncovered. Additionally, I also plan to integrate machine learning techniques and algorithms into the analysis of these attacks due to the large volume of data provided and the adaptability of these techniques for when new threats emerge. This is a unique approach for digital forensic investigations, most current approaches are more reactive, only responding to a breach once it has occurred, whereas this approach is a more proactive strategy that aims to understand the attacker's motive instead, allowing the investigator to decipher how and why the system is compromised quickly.

3.2 Identifying Common Vulnerabilities in Various Cyber-Physical Systems

In this section, I aim to explore the current research surrounding cyber-physical systems to document current vulnerabilities and security threats that are exploited in most attacks on cyber-physical systems. By exploring the existing research on this, I aim to identify key weaknesses that apply to all cyber-physical systems to target better the framework towards finding these quicker, to allow for faster detection and mitigation of these types of breaches. Furthermore, to ensure that I cover every aspect of cyber-physical systems I will divide these into the three separate layers of cyber-physical systems: physical, decision, and cyber. These are summarised below.

3.2.1 Threats Affecting the Physical Layer

These vulnerabilities are primarily attacks that revolve around attacking physical components of the system, which can also result in damage to sensors and actuators:

Type of Attack	Explanation and How to Discover/Mitigate in an Investigation
Physical Damage/Tampering of Sensors/Actuators	Causing physical damage to the sensors may render the device inoperable—Analyse data and system logs to discover these and explore the impact.
Replay attack on sensor data	The attacker keeps resending legitimate data either to gain the trust of the system or to cause a denial of service (DoS) attack. To discover, analyse message logs, and verify message timestamps.
Electromagnetic Interference (EMI)	Involve the generation of electromagnetic signals that interfere with the operation of components within the cyber-physical system. To discover this, it is vital to periodically analyse the signal quality of sensors and ensure that the CPS is properly grounded.
Power Supply/Distribution Attacks	It can target components or the system as a whole due to improper voltage regulation which can affect crucial components. A secondary, uninterruptable power supply can be used to prevent this, and continuous monitoring of this infrastructure would help mitigate this.

Environmental Attacks	These exploit vulnerabilities in the surrounding physical environment, such as humidity/temperature fluctuation to degrade the performance of the sensors of the CPS. Redundant power sources would also help to mitigate these as well as separate HVAC systems (Heating, ventilation, and air conditioning).

3.2.2 Threats Affecting the Decision Layer

Threats that affect the decision layer of cyber-physical systems typically introduce vulnerabilities in the decision-making processes, compromising the reliability of system operations, and potentially causing equipment damage/failure:

Type of Attack	Explanation and How to Discover/Mitigate
Malicious Code Injection	Injection of malicious code that can manipulate data/decisions that determine the behaviour of the system. Input validation can be used to prevent this, and this can be detected through event management software.
Data Manipulation	These include threats such as relay attacks, which can manipulate the data taken in by the sensors. To prevent and mitigate against these data integrity checks can be performed to ensure the validity of data that is sent by the sensors, and then checked that this is the same at the root of the system.
Supply Chain Attacks	These can affect the decision layer by compromising the integrity of software and components within the CPS through third-party suppliers. This can be detected using continuous monitoring and event management tools.

Software Vulnerabilities	Exploiting software vulnerabilities can allow an attacker to execute arbitrary code which may bypass the ordinary decision-making process of the system. To mitigate this, it is crucial to establish robust patch management throughout the CPS to patch software as soon as updates become available.
Algorithmic Bias and Misconfiguration	Algorithmic bias refers to a decision-making practice that is incorrect due to training data used to train the algorithm while misconfiguration refers to the improper deployment of either software or network components within the CPS. To mitigate these, configuration and data audits must take place periodically and during an investigation.

3.2.3 Threats Affecting the Cyber Layer

These threats largely involve the attacker infiltrating the network and communication components within the system, which can disrupt critical processes, jeopardising the operational safety of components:

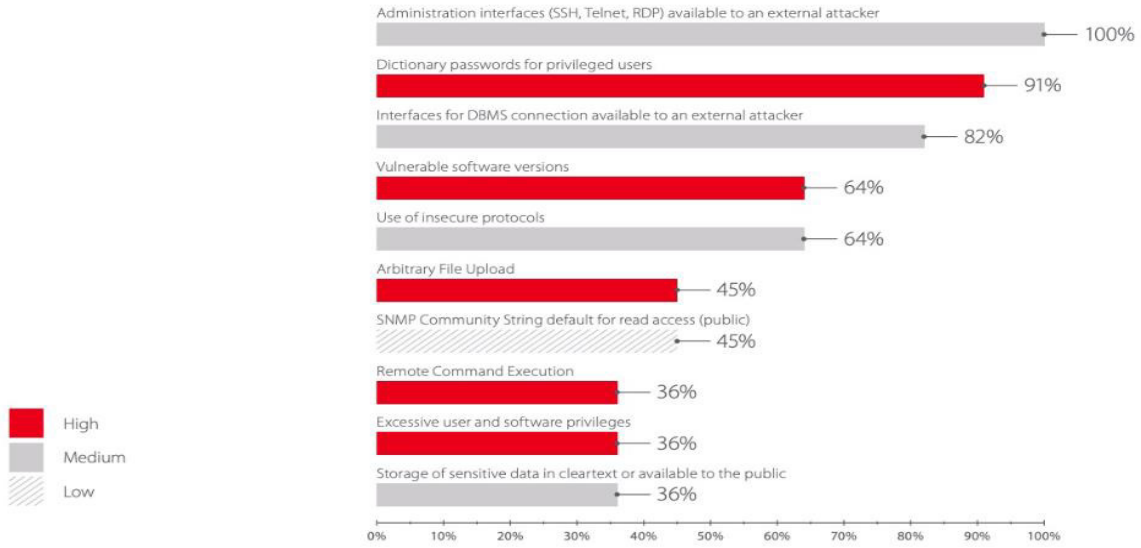
Type of Attack	Explanation and How to Discover/Mitigate
Credential Theft	This allows unauthorised access to the system either through networked components or administrative interfaces within the CPS. These can be prevented by using stronger password policies and deploying intrusion detection systems.
Network Intrusions	These involve unauthorised access to the network and/or firewalls. These can be prevented by constantly monitoring network traffic.

Brute Force Attacks	These attempt to guess or crack passwords and encryption keys within the CPS by trying all possible combinations until a correct one is found. These attacks can be mitigated by using multi-factor authentication and monitored using intrusion detection and security logging software.
Data Breaches and Information Leaks	These compromise the integrity of data stored by the CPS, resulting in the unauthorised access and distribution of data collected by the CPS. The damage of these can be mitigated using data anonymisation and isolating all networked systems within the CPS.
Man-in-the-Middle Attacks (MitM)	Man-in-the-middle attacks intercept communications between devices/sensors and allow the attacker to eavesdrop and manipulate data exchanges. These can be mitigated by using more secure communication protocols like SSH and can be detected using network monitoring.

3.3 Attack Vector Analysis of Cyber-Physical Systems

3.3.1 Industrial Control Systems (ICS)

According to the dataset provided below taken from an article studying the different attack vectors on industrial control systems [23], the most significant type of attack that can affect industrial control systems are attacks that involve the exploitation of the administration interfaces within the system that are accessible to an external attacker, which has an occurrence rate of 100% among the companies that were surveyed to create this data (11 companies total). These attacks grant the attacker permissions that allow them to access critical interfaces within the CPS and therefore a vulnerability with a very high severity for the overall functionality of the cyber-physical system. The next type of attack vector that is common to the industrial control systems surveyed is the use of dictionary attacks to guess the passwords for users of the system that have elevated user permissions. This data suggests that these are risks that are ubiquitous across all the companies surveyed, and therefore can apply to the whole field of Cyber-Physical Systems. As such I aim to target the framework better towards prioritising the discovery and mitigation of these types of attacks sooner. The best way of discovering and mitigating these during a digital forensics investigation would be by analysing the network traffic and system log files and correlating these log files with timestamps of known physical incidents within the system.



Top 10 vulnerabilities on the corporate information system perimeter of industrial companies (percentage of client companies, by severity level)

Figure 2: Taken from Passive Technologies Studying ICS Security [23].

3.3.2 Autonomous Vehicles (AVs)

For autonomous vehicles, the attack vectors appear to be vastly different when compared to industrial control systems (ICS). From the table below [8], attacks on cloud servers have been the most common issue affecting automotive vehicles between 2010 and 2021. The unauthorised use of keyless entry fobs is the second most prevalent breach affecting automotive systems (26.3% between 2010 and 2021), and these can be investigated from both a physical and digital perspective depending on the approach that the attacker used. For example, if an attacker used signal interception or cloned the fob, then these could be analysed from both perspectives. These attacks present unique challenges for a digital forensics' investigation, as the key fobs use wireless communication to enable access and ignition control to the vehicle, which is harder to prevent, but can be analysed by exploring the network/signal traffic.

The data regarding attacks on the ECU-TCU gateway (Electronic Control Unit, Telematics Control Unit) also shows an increase in attacks in recent years, from 2.6% between 2010-2018 to 12.2% between 2010-2021. These components of automotive vehicles control vital functions related to engine management, access, and the telematics used by the vehicle. This data suggests that attackers are breaching these systems to gain access to the central vehicle control systems, which can have a large impact on passenger safety and vehicle operation, allowing the attacker to compromise the brakes or steering for example. This physical evidence, when an accident occurs involving this type of attack, can be correlated to the analysis of the firmware and software running on both the ECU and TCU. From this data, most of the other types of attacks seem to be decreasing recently for these vehicles, so this framework will be more targeted towards exploring the previously mentioned attacks instead, as these are more likely to have occurred.

Table 2: Automotive Attack Vectors				
Hardware or software	**Share: 2010–2018**	**Share: 2010–2019**	**Share: 2010–2020**	**Share: 2010–2021**
Cloud servers	21.4%	27.2%	32.9%	41.1%
Keyless entry key fob	18.8%	29.6%	25.3%	26.3%
ECU-TCU* gateway	2.6%	5.0%	4.3%	12.2%
Mobile app	7.4%	12.7%	9.9%	7.3%
Infotainment system	7.4%	7.7%	7.0%	5.7%
OBD** port	10.4%	10.4%	8.4%	5.4%
IT system/network	N/A	N/A	7.0%	5.1%
Sensors	3.5%	5.3%	4.8%	3.3%
In-vehicle network	N/A	3.3%	3.8%	2.9%
Wi-Fi network	4.4%	5.3%	3.8%	2.9%
Bluetooth	3.1%	4.4%	3.6%	2.7%
OBD dongle	1.8%	3.6%	3.1%	N/A
Cellular network	4.8%	4.1%	2.4%	N/A
USB or SD port	3.1%	N/A	2.1%	N/A

(Data sourced from Upstream Security's 2019, 2020, 2021, and 2022 Cybersecurity Reports)
*Electronic control unit to telematics control unit **On-board diagnostics

Figure 3: Taken from EETimes Studying Automotive Cyber Security [8].

3.3.3 Building Automation Systems (BAS)

The final category of CPS that I aim to explore the most common attack vectors for is Building Automation Systems (BAS). According to an article written by Kaspersky in 2022 [31] studying threats on operational technology, particularly focusing on building automation threats because out of the systems that use Kaspersky for their security solution, "nearly half of the computers (42%) faced cyber threats". This graph shows data taken from the first half (H1) of 2021 and 2022 and the second half (H2) of 2021 from the Kaspersky article (figure 4).

From the data below the most significant threat that affects building automation systems (BAS) are the use of malicious scripts and phishing pages which use JavaScript and HTML. These saw a substantial increase recently from 9.5% in H2 2021 to 12.9% in H1 2022. This indicates that these attacks are more prevalent than the others and as such it is a good idea to target the framework towards this type of threat. From my table discussing threats that affect the decision layer of cyber-physical systems, this threat falls under that category and can be detected by analysing system log files. This also correlates with the threats that affect the previously mentioned systems (ICS and Autonomous Vehicles), which shows that analysing the system logs is a vital piece of information when investigating cyber-attacks on CPS. The framework will therefore begin by performing this analysis as it seems applicable to most systems, while simultaneously preserving the physical evidence to correlate to the data collected from these logs.

The next significant threat that affects building automation systems is the unauthorised access to blacklisted or restricted internet resources. These attack the system by exploiting vulnerabilities in software (such as executing malicious scripts) and may also involve consuming system and network resources, potentially causing a Denial of Service (DoS) attack which can degrade HVAC Systems and intrusion detection. This threat has decreased in H1 2022, from 14% in H1 2021 to just 9.5% in H2 2022. However, from this data, this is still the second most significant threat that affects building automation systems and as such it would be beneficial to explore these sooner in the framework as well. This can be completed by analysing the access logs and network traffic, these must be analysed before examining the physical evidence in this case in my opinion, however, it is still crucial that this evidence is preserved as soon as possible.

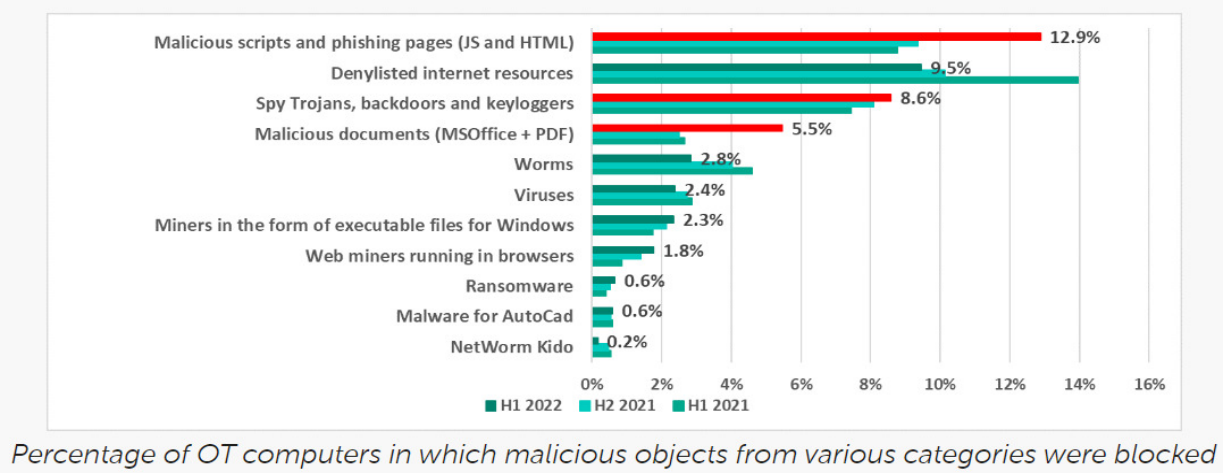


Figure 4: Taken from Kaspersky Discussing the Attack Vectors on BAS [31].

4.0 Methodology

4.1 Theoretical Framework

In this section, I aim to outline the process of identifying the most effective procedures for analysing and preserving data within the framework for Cyber-Physical System (CPS) forensics and specifying the order that this should be completed during an investigation. This involves determining the best methods and techniques for collecting, processing, and analysing both digital and physical evidence obtained from Cyber-Physical System environments. The goal of this is to establish a structured approach that ensures the thorough examination of data while maintaining accuracy, integrity, and relevance throughout the forensic investigation process as well as ensuring that a fully documented chain of custody is consistently maintained. This framework will have five total phases to follow throughout the course of an investigation as follows:

1. ***Evidence Capture and Data Collection:*** This needs to be completed as soon as possible and involves collecting evidence from various sources within the system (defined in the next section) and must be preserved using forensic techniques and best practises to maintain integrity and admissibility.
2. ***Data Analysis***: The data collected from the system, as well as any physical artifacts, must then be analysed to identify correlations, patterns, or discrepancies that may be indicative of a security breach. These can then be reconstructed if needed and a timeline of events must be established.
3. ***Root Cause Determination***: The root cause of the breach must then be determined and rectified before the system can become operational once again.
4. ***Reporting***: All findings throughout the investigation must be reported, including a full summary of the incident, how the evidence was analysed, a full chain of custody, and any recommendations to prevent the same breach from occurring in the future. This report must be fully compliant with all legal and regulatory requirements regarding data privacy, evidence handling, and incident reporting.
5. ***Remediation Phase***: Based on the findings from the investigation any corrective actions must be implemented to address vulnerabilities found during the investigation to mitigate risks to the overall security of the system.

4.1.1 Data Collection and Preservation

When conducting a digital forensics investigation, it is imperative to prioritise the preservation of the most volatile data at the beginning of the examination of the system to ensure that no data is lost or overwritten either by the continued use of the system or if the system or its components loses power at any point. This also encompasses the physical data that must be collected from the surrounding environment, as this may alter due to ecological changes over time therefore must be collected and preserved at the examiner's earliest convenience. Additionally, this must include every type of data that can be collected from the CPS, and the priority of these may change when analysing an active system compared to an inactive system. The data that must be collected and the procedures for completing this (in order) are as follows, including a summary of potential ways that this could be explored:

1. Physical Evidence (Volatile evidence available in the surrounding environment):
   - *Procedure:* Conduct a thorough visual inspection of the surrounding area, buildings, and any components of the system as soon as possible that may be damaged or have been tampered with (physical artifacts), cataloguing these with a timestamp of when these

have been discovered, evidence ID, and a short description of the evidence item. Can also take physical samples using externally calibrated equipment ensuring that the chain of custody is complete. This should be completed first to prevent ecological changes from affecting the evidence.

- *Documentation:* Where possible, take photographs or data readings of this evidence, which can also be done with sketches however these are generally less accurate, and therefore photographic evidence would be preferred. Additionally, write a more detailed description of the data to be catalogued along with the evidence.

- *Analysis:* Use a variety of tools to analyse this data/evidence, such as data visualisation software like Microsoft Power BI [32] or Geospatial Analysis software like ArcGIS [2] which allows the investigator to overlay physical data to visualise geographical patterns.

- *Preservation:* Preserve any collectible physical evidence using secure, sealable packaging to avoid contamination, degradation, and tampering. Additionally, meticulously document any findings through photographs and detailed descriptions, and labelling and store the evidence in a secure facility that has intrusion detection and full access control.

- *Typical Data Format:* Photographs/Sketches (.jpg, .png, .gif), Physical Samples (.pdf, .docx).

2. Core System Logs and Memory Dumps (taken from either live system or inactive, collection procedure will be different, in both cases system should be isolated immediately from wider internet and any other unaffected systems as it isn't yet known if a malware/worm is responsible for the breach, includes system logs and config files):
   - *Procedure (Live System):* Identify the type of core system that needs to be analysed (Central Control System, or ECU in autonomous vehicles, etc) and access the system causing as little disruption as possible. Then use logging tools/commands to extract the system logs capturing information such as power states, configuration messages, authentication attempts, and error messages. If possible (system dependent) access through syslog servers instead as this will cause minimal disruption to the system.

   - *Procedure (Inactive System):* In the case that the system is disabled/rendered inactive, a bit-for-bit copy of the system storage (including all partitions) and volatile memory must be created and stored on an external drive using forensic imaging tools/software, (dd in Linux to create an '*image.dd*' file, or software like FTK Imager [6]) From this the system logs can be accessed. A hardware write blocker must also be used to prevent the drive from being written during the collection process.

   - *Documentation:* The date and time of the log collection must be recorded, including the type and path of the log file collected/accessed, the version, serial number, and name of the system at the time of the collection, the method and tools used for collection, the name and details of the person accessing it and the hash of the file before the file is accessed for analysis. This must also be maintained throughout the investigation, so each time the file is accessed the chain of custody must be fully documented.

- *Analysis:* The analysis of the system logs involves looking for patterns, anomalies, and indicators of suspicious activity within the CPS. This ideally would be performed before the analysis of the physical evidence, to avoid tunnel vision during the investigation. This means that the physical evidence should be correlated to the digital evidence after it is preserved.

- *Preservation:* Logs must be stored using secure, tamper-proof external storage media with verifiable MD5 hashes and access control to ensure integrity calculated from the time of collection, maintaining the chain of custody document and regular back-ups of this data throughout the investigation.

- *Typical Data Format:* System Logs (.txt, .log, .csv), Memory Dumps (.dmp, .bin, .img).

3. Network Traffic Logs (taken from within the CPS during/after the incident):
   - *Procedure:* Deploy network monitoring software or packet capture software such as Wireshark [30] to capture network traffic in real time at strategic points within the CPS and analyse network logs stored on the system using this software. These can then be filtered to capture/analyse only relevant network traffic configuring rules based on protocols, IP addresses, and ports.

   - *Documentation:* These need to be timestamped (most software used to capture traffic will do this anyway) and will need to include appropriate metadata to provide better context for analysis. Will also again need to maintain a chain of custody and be externally stored and verified. Details of how the data was captured must also be included, including the duration and what filter rules were applied.

   - *Analysis:* These logs must be analysed to look for anomalies/suspicious IP addresses found within the network. The protocols used in these packets must also be examined to better understand the nature of communications and what the attacker might have done when they accessed the system. Patterns correlating the timestamps of the network traffic in comparison to when the incident occurred/data alterations within the environment can help to reconstruct the series of events.

   - *Preservation:* Must be stored in secure, tamper-proof environment such as encrypted or write-protected disk images. Detailed CoC must be kept logging all access, and data must be periodically backed up and stored in multiple locations to ensure that the data isn't lost.

   - *Typical Data Format:* Packet Capture (.pcap, .pcapng), Log Files (.log, .txt, .csv).

4. User Activity Logs:
   - *Procedure:* Obtain authorised access to the system operating system and any security software deployed and retrieve them ensuring that they are collected in their original format (calculate MD5 hash upon retrieval).

- *Documentation:* Document the source of the log, the time and date that it was collected, the method used for retrieval, and the format of the logs (i.e. structure and metadata). Furthermore, a detailed chain of custody must also be kept with times and dates.

- *Analysis:* These logs must be reviewed to detect any unauthorised access to the system and other actions which may be relevant to the investigation. This can include anomalous access logs that differ from regular operating hours and attempts outside of the network. To do this, the regular activity of the CPS should be analysed first, so it is easier to detect when a suspicious log-in attempt is made.

- *Preservation:* These logs need to be preserved using tamper-proof storage, like encrypted disks or write-protected storage. Backup copies of these logs can also be made and stored in multiple secure locations to prevent corruption. Detailed CoC must be kept logging every access to the logs/storage device and a retention period must be determined, as these logs can contain sensitive information regarding the CPS.

- *Typical Data Format:* Log Files/Audit Trails (.log, .txt, .csv).

5. Sensor Data:
   - *Procedure:* All the sensors within the CPS will need to be identified including the types of data that they collect and in what format (e.g. temperature sensors either being in degrees Celsius or Fahrenheit). Authorised access to this information must then be granted to collect the data, this can be done by either physically connecting to the sensors using appropriate cables or accessing it over the network or management hub for the CPS. Timestamps of this data must also be collected to correlate the digital and physical evidence.

   - *Documentation:* A full catalogue of every system within the CPS must be recorded, and all recent sensor data (up to two weeks before the breach) must be kept, with this increasing depending on when the attacker first accessed the system. All details of the data acquisition process must also be recorded, and the format of the data collected.

   - *Analysis:* This will involve validating the data that has been collected from the sensors, by cross-referencing this with data collected by similar sensors within the same environment. This can also be completed after all the data has been retrieved from the system if it is still usable by collecting new readings and externally validating these. The sensor data should then be analysed for any anomalous readings which may indicate equipment malfunctions or security incidents, therefore the data before the breach may need to be analysed to detect data that differs from regular operating procedures.

   - *Preservation:* The integrity of the sensor data must be preserved by storing it in a secure, tamper-proof format such as encrypted disks or write-protected storage. A full chain of custody must be kept with timestamps, the person who accessed the data, the acquisition process, and any other relevant information regarding the integrity of the data. A retention period should also be defined for this data if it is sensitive to the system or if it is relevant to any other investigations.

- *Typical Data Format:* Time-series data/Senor Readings (.csv, .txt)

6. Alert Logs
   - *Procedure:* Identify the alerting systems/tools within the CPS such as any intrusion detection systems (IDS) or event management systems and obtain access permissions to retrieve these logs, collecting them in their original format without any modification. Filtering criteria can also be developed to only collect the logs that are relevant to the investigation such as policy violations and system errors.

   - *Documentation:* The sources of these alert logs must be recorded including names, versions, and configurations of the alerting systems. Acquisition methods must also be logged including timestamps, the name of the investigator, the type of the alert, and any other relevant information. Detailed chain of custody must also be kept throughout.

   - *Analysis:* These logs must be reviewed to identify any relevant alerts within the CPS environment during the investigation period. These can be correlated with other evidence items previously gathered, such as system logs, network traffic, or physical evidence like sensor malfunction/equipment failure to reconstruct the sequence of events and investigate the impact that this has had on the system. These alerts can be prioritised in terms of severity, which will lessen the downtime of the system.

   - *Preservation:* The integrity of the alert logs must be preserved by storing them in secure, tamper-proof storage facilities or write-protected storage. Additionally, backup copies of these can be created and these can be stored in multiple locations. Access controls can also be implemented to protect against the unauthorised access of these logs during the investigation. A full chain of custody must also be maintained throughout the investigation, with timestamps and the signature of the person analysing the data.

   - *Typical Data Format:* Log Files: (.log, .txt, .csv), Alert Notifications (.txt, .csv).

7. System State Snapshots and Firmware:
   - *Procedure:* Snapshots must be taken of the current system state of the CPS, including system files, registry, and configuration settings, running processes, and open network connections. This can be done using system restore points. Firmware images can also be extracted from various components within the CPS, this can be done by investigating the firmware version of the device and downloading it from the manufacturer's website, if available.

   - *Documentation:* Document the contents of the snapshot, including files, directories, and configuration settings captured in each snapshot, paying particular attention to any variations between snapshots from different times. Also document details of extracted firmware images, including the component they belong to, manufacturer, version number, and any relevant release notes.

   - *Analysis:* Compare system state snapshots taken at different points in time to identify changes within the CPS environment, correlating these to system/access logs. Analyse the

firmware images to understand their functionality, and research these to identify any known vulnerabilities or backdoors to the system.

- *Preservation:* Preserve the integrity of system state snapshots and firmware images by storing them using tamper-proof or write-protected storage and creating backup copies of these. A full chain of custody must also be recorded, which includes the calculated MD5 hash of the image.

- *Typical Data Formats:* System Snapshots: (.iso, .vmdk, .vhdx), Firmware Images: (.bin, .img, .hex).

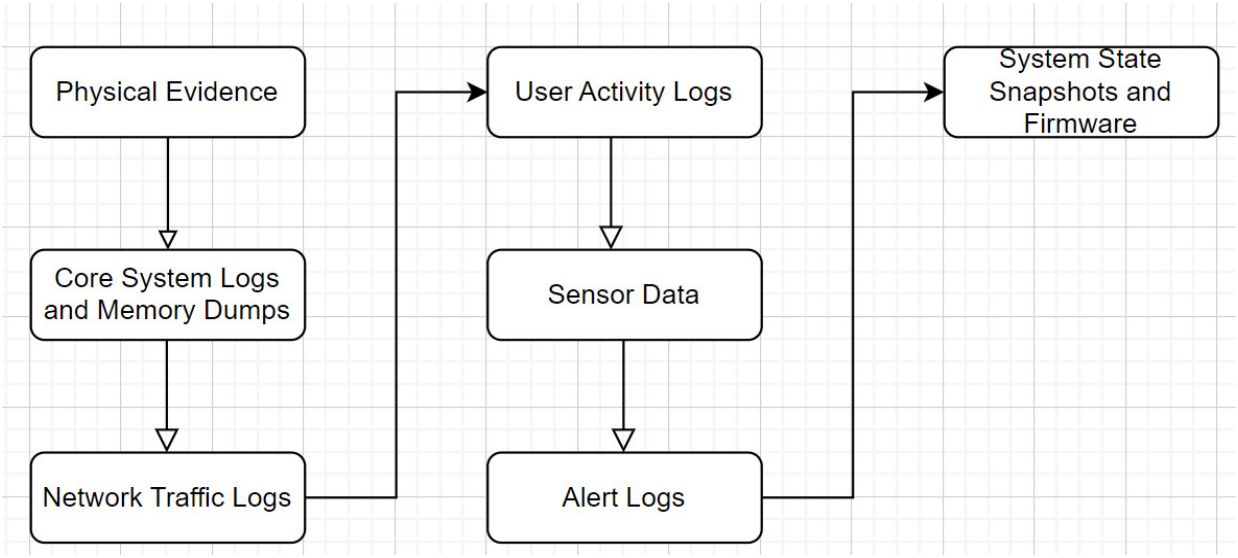


Figure 5: Flowchart of the Data Acquisition Process

Once each piece of evidence has been fully documented, it can then be categorized using the table below to attempt to correlate the determined type of attack to its physical consequence, therefore showing the impact that this has had on the system:

Attack Type	*Physical Consequence*	*Digital Trace*
Malware	Damage to physical components (e.g. Sensors, Actuators, Pipes, Conveyors)	Evidence of data breach (access logs) or malicious software execution (incorrect values)
Denial of Service (DoS)	Disruption of critical operations, leading to safety hazards or equipment failure.	Severe network/system traffic and service downtime, large amount of time to execute operations.
Man-in-the-middle	Tampering with control signals causing unsafe actions.	Data interception, unfamiliar network traffic, data manipulation.
Ransomware	Locking or disabling critical system functions, compromising personnel and equipment safety	Extortion, financial losses, data loss, and full equipment failure/shutdown.

Spoofing	False data leading to equipment damage/improper operations executed	Unauthorised access from logs and falsified sensor/actuator data.
Phishing	Deception of personnel leads to unauthorised access which compromises the system.	Credential theft, unauthorised access, social engineering evidence.
Supply Chain Attacks	Compromise of third-party components or services crucial to the system.	Backdoor access, evidence of data leaks.

4.2 Identifying Optimum Methods of Data Analysis Using Pre-Existing Data Sets

To identify optimum procedures for analysing data from Cyber-Physical Systems (CPS), a selection of data that has been acquired from several simulated systems using a testbed will be used. I will be using the datasets compiled by Tommy Morris which detail cyber-attacks on industrial control systems [18]. Specifically, I will be using the third dataset provided which is compiled of data that has been collected during simulations of cyber-attacks occurring on both a gas pipeline and a water storage tank. Notably this data has been found to contain unintended patterns that can skew results from some machine learning patterns, leading to the inaccurate identification of attacks that were incorrectly associated with specific parameter values; however, efforts have been made to rectify these issues, and as this project intends on exploring various methods of reviewing this data anyway, this shouldn't skew the results too much once the data has been pre-processed. This data is taken from systems using Supervisory Control and Acquisition Systems (SCADA). These allow the real-time acquisition and processing of data within the system and rely on a communication infrastructure within the system to relay this information.

Like most Cyber-Physical Systems, these pipelines rely on the MODBUS protocol to communicate with other devices on its network. It uses a master-slave approach, where the master device initiates the communication by sending requests to each slave device which then replies, giving the data that the master device requested. It is well suited to Cyber-Physical Systems as it supports various data formats such as ASCII, binary, and hexadecimal and ensures that data is efficiently coordinated across the whole system. Each frame sent by this protocol is segmented into four primary fields including device address, function code, data, and error-checking fields, this allows for the analysis of error rates as seen in the dataset and the protocol allows the system to continuously poll for new data. The general structure of a MODBUS packet is shown below:

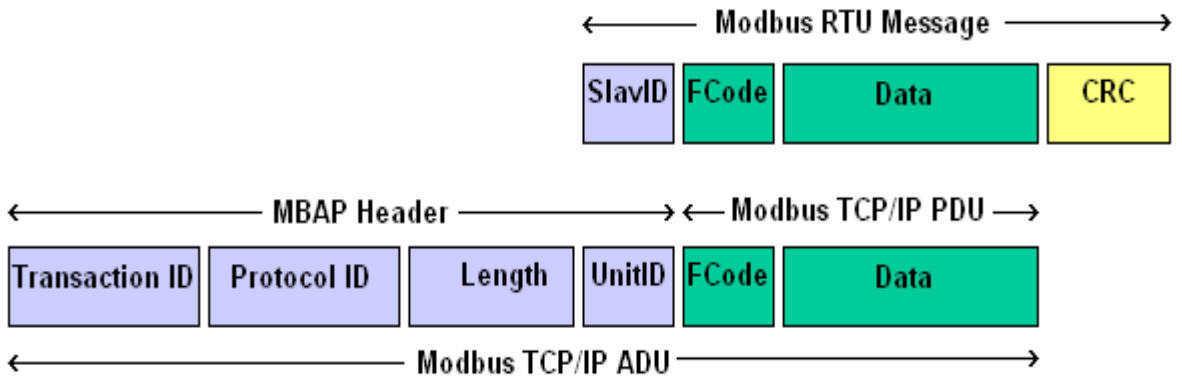


Figure 6: General Structure of a MODBUS Packet [33]

To create this data, several types of attacks have been used. Below is a summary of these as well as their expected effect on the data in each dataset (Note: This only includes the data that applies to other types of CPS as well, as they are network features using MODBUS):

Type of Attack	*Description*	*Potential Effect*	*What data does this affect in the dataset?*
Reconnaissance Attacks	This type of attack aims to gather crucial information about the system such as its network and device architecture, which allows an attacker to plan more harmful attacks in the future.	This could result in the identification of components throughout the system, their relationships within the system, and vulnerability discovery (through Firmware revisions)	Command address, response address, command memory, response memory, command memory count, command length, response length, time interval.
Response Injection Attacks	These manipulate data sent from the central server to its clients, providing false system state information.	Could deceive the system/operators into performing incorrect actions based on falsified data. This can lead to unintended changes to system behaviour.	Response length, time interval, and potentially CRC error rate.
Command Injection Attacks	This involves injecting unauthorised commands into the system to manipulate system/component behaviour.	This could lead to unauthorised control of critical components within the system and allow an attacker to alter the configurations of the system or components.	Response length, time interval, CRC error rate.
Denial of Service (DoS)	Disrupts the availability of the system by overwhelming system resources, rendering the system unresponsive to genuine users.	This can result in a loss of access and control to the system and an inability to respond to critical events leading to operational failure.	Time interval, CRC error rate.

To ensure a proper forensic investigation the data should be collected in the order specified in 4.1.1 and then it should be pre-processed to remove any unnecessary data that may disrupt the results of the analysis. This would involve removing any lines of data with missing values (unless significant, as this could indicate a problem with a particular component) and compiling/formatting the data into a suitable format (.csv, for example) ready for analysis. The features above then need to be selected for analysis including any other features that are specific to that system (e.g. pump rate in the water system) The additional features contained within this dataset for both the gas pipeline and the water pump are as follows:

Gas Pipeline	*Water Pump*
Gain	HH – High Alarm Setpoint
Reset – How long the system corrects for long-term variations	H - High Setpoint
Deadband – When no action is taken by the CPS	L – Low Setpoint
Cycletime – Interval between measurements	LL – Low Alarm Setpoint
Rate – Rate of change	Control_mode
Setpoint – Desired pressure	Control_scheme
Control_mode – Operational mode such as PID or manual control	Pump – State of the pump
Control_scheme – Overall scheme of regulating pressure management	Measurement – Water level
Pump – State of the pump	
Solenoid – Activates the relief valve	
Measurement – Pressure Level	

4.2.1 Analysing Gas and Water Pipelines

This approach will begin with pre-processing the data to address any obvious inconsistencies within either dataset. Once this is accomplished it will then be essential to explore diverse analytical techniques that will also apply to other systems such as the use of anomaly detection methods and statistical analysis to uncover patterns within the data that specifically correlate to the timestamps of the simulated attacks. These methods will then be evaluated based on their accuracy, ensuring that they correlate to the attack's physical consequences as well. Additionally, correlation analysis could be performed between altering variables within the dataset which could reveal relationships between these that may be indicative of a cyber-attack occurring. The analysis of this data will begin by initially converting the raw data .txt files to .csv format, then using Python (version 3.11.3) modules such as:

- ***Pandas:*** Pandas is a library that allows the visualisation and analysis of data using data frames created from various file formats and allows the cleansing of anomalous data from CSV files.

- ***Matplotlib:*** Matplotlib is a library for creating both static and animated visualisations, allowing the user to create a wide range of plots and charts from the data formatted by Pandas.

- ***Scikit*-learn (sklearn):** Scikit-learn is a machine-learning library that allows for data mining and data analysis, using methods like classification and clustering to further analyse the data. Can also be used in conjunction with NumPy.

- *Standard Scaler (sklearn):* This is a preprocessing technique used on the dataset that standardises the features within the dataset before the anomaly detection and result estimation.

- *Random Forest Classifier (sklearn):* Random Forest Classifier constructs multiple decision trees during training and outputs the most frequent prediction of the individual trees, this is especially useful with data like this as it is robust against overfitting, helping to combat some of the issues with machine learning detailed on the website for the dataset [18].

- *Random Forest Regressor (sklearn):* Similar to random forest classifier but used in this case to adjust the anomaly scores to further ensure accuracy.

- *Cross_val_score (sklearn):* Used to evaluate the model's performance by splitting the dataset into k consecutive folds during the estimation of the result column for both datasets.

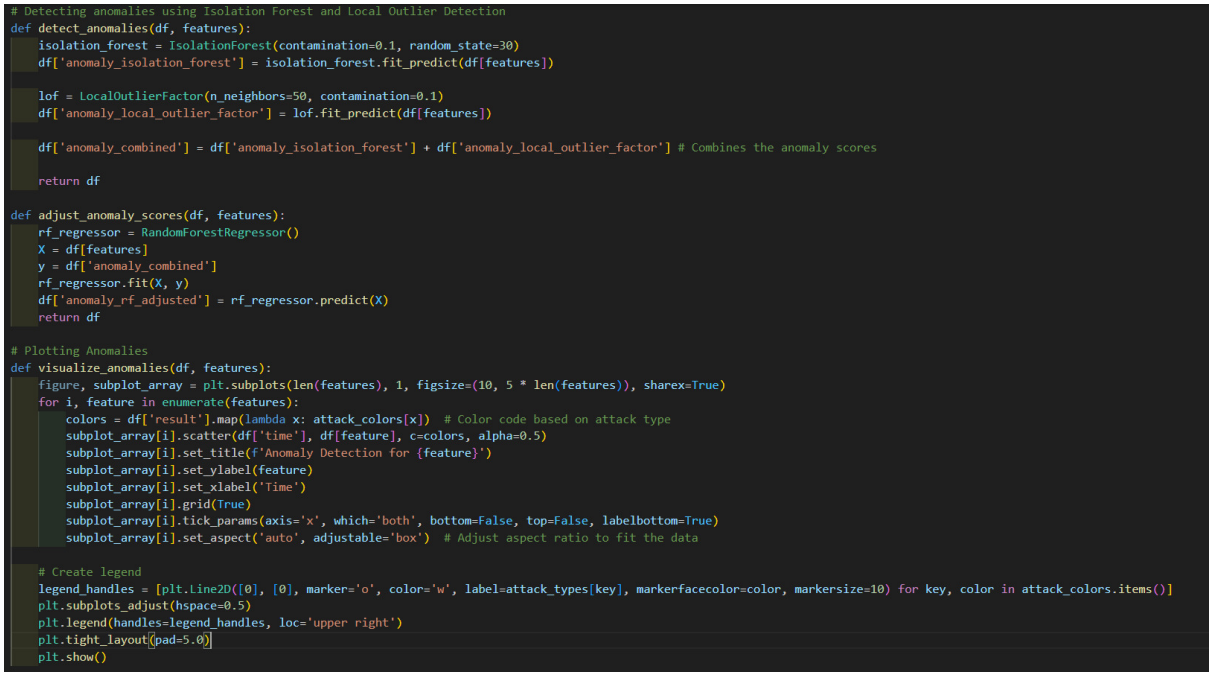
The code that I've written for the analysis of these datasets conducts anomaly detection and result estimation on both the gas and water pipeline datasets curated by Tommy Morris. Firstly, dictionaries are initialised specifying the attack types for the dataset and these are mapped to their corresponding colours to visualise anomalies as follows, including the code for pre-processing the dataset to remove any obvious inconsistencies is as follows:

```
# Define attack type descriptions
attack_types = {
    0: 'Normal - Not part of an attack',
    1: 'NMRI - Naive Malicious Response Injection Attack',
    2: 'CMRI - Complex Malicious Response Injection Attack',
    3: 'MSCI - Malicious State Command Injection Attack',
    4: 'MPCI - Malicious Parameter Command Injection Attack',
    5: 'MFCI - Malicious Function Command Injection Attack',
    6: 'DoS - Denial of Service',
    7: 'Reconnaissance - Probe for System Information'
}

# Define custom color map for attack types
attack_colors = {
    0: 'blue',      # Normal
    1: 'green',     # NMRI
    2: 'red',       # CMRI
    3: 'orange',    # MSCI
    4: 'purple',    # MPCI
    5: 'yellow',    # MFCI
    6: 'cyan',      # DoS
    7: 'magenta'    # Reconnaissance
}

# Loading the dataset
def load_data(file_path, column_names, skiprows):
    df = pd.read_csv(file_path, encoding='utf-8', names=column_names, header=None, skiprows=skiprows)
    df.drop_duplicates(inplace=True)
    df.dropna(inplace=True)

    return df

# Standardising the Features
def preprocess_data(df, features):
    scaler = StandardScaler()
    df[features] = scaler.fit_transform(df[features])
    return df
```

These are the attack types with their corresponding colour:

Attack Type	*Description*	*Colour*
0	Normal - Not part of an attack	Blue
1	NMRI – Naive Malicious Response Injection Attack	Green
2	CMRI – Complex Malicious Response Injection Attack	Red
3	MSCI – Malicious State Command Injection Attack	Orange
4	MPCI – Malicious Parameter Command Injection Attack	Purple
5	MFCI – Malicious Function Command Injection Attack	Yellow
6	DoS – Denial of Service	Cyan
7	Reconnaissance – Probe for System Information	Magenta

Standard Scaler is applied to standardise the data, ensuring uniformity across all dataset features, I did this to attempt to mitigate the issues highlighted in the dataset summaries. Then, anomaly detection is performed using Isolation Forest and Local Outlier Factor algorithms, assigning scores to each data point within each column of the data frame (every column within the .csv file). These scores are further refined using Random Forest Regression and combined to generate a comprehensive anomaly score for each data point. The code then employs a Random Forest Classifier to predict the result column, splitting the data into training and testing sets for training and evaluation, respectively, and writing the true and estimated results of these to .csv files allowing for further analysis of their accuracy and physical consequences. This process aids in accurately identifying potential cyber-attacks, thus proving invaluable in digital forensic investigations involving cyber-physical systems for promptly categorizing new attacks as they occur. The code to do this section is as follows:



```
# Detecting anomalies using Isolation Forest and Local Outlier Detection
def detect_anomalies(df, features):
    isolation_forest = IsolationForest(contamination=0.1, random_state=30)
    df['anomaly_isolation_forest'] = isolation_forest.fit_predict(df[features])

    lof = LocalOutlierFactor(n_neighbors=50, contamination=0.1)
    df['anomaly_local_outlier_factor'] = lof.fit_predict(df[features])

    df['anomaly_combined'] = df['anomaly_isolation_forest'] + df['anomaly_local_outlier_factor'] # Combines the anomaly scores

    return df

def adjust_anomaly_scores(df, features):
    rf_regressor = RandomForestRegressor()
    X = df[features]
    y = df['anomaly_combined']
    rf_regressor.fit(X, y)
    df['anomaly_rf_adjusted'] = rf_regressor.predict(X)
    return df

# Plotting Anomalies
def visualize_anomalies(df, features):
    figure, subplot_array = plt.subplots(len(features), 1, figsize=(10, 5 * len(features)), sharex=True)
    for i, feature in enumerate(features):
        colors = df['result'].map(lambda x: attack_colors[x])  # Color code based on attack type
        subplot_array[i].scatter(df['time'], df[feature], c=colors, alpha=0.5)
        subplot_array[i].set_title(f'Anomaly Detection for {feature}')
        subplot_array[i].set_ylabel(feature)
        subplot_array[i].set_xlabel('Time')
        subplot_array[i].grid(True)
        subplot_array[i].tick_params(axis='x', which='both', bottom=False, top=False, labelbottom=True)
        subplot_array[i].set_aspect('auto', adjustable='box')  # Adjust aspect ratio to fit the data

    # Create legend
    legend_handles = [plt.Line2D([0], [0], marker='o', color='w', label=attack_types[key], markerfacecolor=color, markersize=10) for key, color in attack_colors.items()]
    plt.subplots_adjust(hspace=0.5)
    plt.legend(handles=legend_handles, loc='upper right')
    plt.tight_layout(pad=5.0)
    plt.show()
```

For the analysis of these datasets, I decided to choose the following features for anomaly detection:

Gas Pipeline	*Water Pipeline*
command_memory	command_memory
response_address	response_address
command_length	command_length
pump	pump
solenoid	crc_rate
crc_rate	measurement
measurement	time

The selection of features for anomaly detection in both datasets was based on their relevance to the functionality and control of these systems, as they all have crucial roles in monitoring and controlling the pipelines' operations. For instance, command memory and response address are fundamental for tracking the communication between different components of the pipeline systems, while command length and CRC Rates provide insights into the structure and integrity of the commands being sent. Pump and solenoid statuses are also essential for understanding the activation of key components within the pipelines. Additionally, the measurement data allows for monitoring the pipeline's performance, while time stamps facilitate the temporal analysis of events, which is vital for identifying anomalies and potential cyber-attacks. These specific features were also chosen as they are relevant to this framework while being diverse enough so that the steps provided in this section are also crucial to all other types of cyber-physical systems, just using different data instead, although it is notable that more relevant features could be selected. While I managed to achieve considerably high accuracy results using these selected features, and it does make it easier to compare the results with both datasets, including additional features would undoubtedly be beneficial during digital forensics investigations.

To complete the attack prediction section for both the gas pipeline and the water pipeline dataset several machine learning techniques were used to ensure a thorough analysis of how these techniques could be used to analyse this data and which one is best. The techniques used include:

- *Random Forest:* This is an ensemble learning technique based on decision trees. During training it constructs multiple decision trees, and the predictions are made by combining the results from these trees, which is useful for this data as this prevents overfitting. It is also highly resilient to noisy data and irrelevant features, making it a good choice for large datasets such as those found in investigations of Cyber-Physical Systems.

- *Decision Tree:* This method divides the dataset into smaller subsets based on its features and makes decisions by traversing the tree from root to leaf nodes, where each internal node represents a decision based upon these features, and each leaf node represents the outcome of the prediction. This technique is good for numerical and categorical data; however, it can also be prone to overfitting when using large amounts of data.

- *Support Vector Machine:* This is a supervised machine learning algorithm that finds the optimal hyperplane that best separates data points between different classes. This is good when working with data from Cyber-Physical Systems as it is robust to overfitting, however, it can be memory-intensive for larger datasets.

- ***K-Nearest Neighbours (KNN):*** This algorithm predicts new data by finding the K closest data points in the training set to the given data point that needs to be classified. The majority class among these K Nearest Neighbours is then assigned to the data point. This is well-suited to Cyber-Physical System data as it can easily handle both numerical and categorical data. It is however computationally expensive, so may not be suitable for extremely large datasets.

- ***Gaussian Naïve Bayes:*** This algorithm classifies data using Bayes' theorem to calculate the conditional probability of a class label given a set of features and assumes that each value is independent of the next (useful for high-dimensional datasets). This, however, may be suitable for complex Cyber-Physical Systems such as gas and water pipelines, as each feature may be dependent on another column, skewing the results.

- ***Logistic Regression:*** Logistic Regression can be useful for its simplicity and interpretability, allowing domain experts to understand the model's decision-making process of predicting the probability that a given instance belongs to a particular class. However, Logistic Regression assumes a linear relationship between the features and the log odds of the response, which may not always hold in complex Cyber-Physical Systems with nonlinear dependencies. Additionally, it may struggle with datasets that have imbalanced classes or non-linear decision boundaries, which are common challenges in cybersecurity applications.

The code to complete this prediction section is as follows:

```
def estimate_result(df, features):
    # Split the dataset into training and testing sets
    X_train, X_test, y_train, y_test = train_test_split(df[df['anomaly_combined'] == 0][features],
                                                        df[df['anomaly_combined'] == 0]['result'],
                                                        test_size=0.2, random_state=42)

    classifiers = { # More can be added if needed.
        "Random Forest": RandomForestClassifier(random_state=42),
        "Decision Tree": DecisionTreeClassifier(random_state=42),
        "Support Vector Machine": SVC(random_state=42),
        "K-Nearest Neighbors": KNeighborsClassifier(),
        "Gaussian Naive Bayes": GaussianNB(),
        "Logistic Regression": LogisticRegression()
    }

    results = {}

    for name, classifier in classifiers.items():
        # Train the classifier
        classifier.fit(X_train, y_train)

        # Predict 'result' column on the entire dataset
        df[f'estimated_result_{name}'] = classifier.predict(df[features])

        # Calculate evaluation metrics
        accuracy = accuracy_score(df[df['anomaly_combined'] == 0]['result'], df[df['anomaly_combined'] == 0][f'estimated_result_{name}'])
        precision = precision_score(df[df['anomaly_combined'] == 0]['result'], df[df['anomaly_combined'] == 0][f'estimated_result_{name}'], average='weighted')
        recall = recall_score(df[df['anomaly_combined'] == 0]['result'], df[df['anomaly_combined'] == 0][f'estimated_result_{name}'], average='weighted')
        f1 = f1_score(df[df['anomaly_combined'] == 0]['result'], df[df['anomaly_combined'] == 0][f'estimated_result_{name}'], average='weighted')

        results[name] = {
            "accuracy": accuracy,
            "precision": precision,
            "recall": recall,
            "f1_score": f1
        }

    return df, results

def evaluate_result_estimation(df):
    # Evaluate accuracy of 'estimated_result' compared to actual 'result'
    accuracy = accuracy_score(df[df['anomaly_combined'] == 0]['result'], df[df['anomaly_combined'] == 0]['estimated_result'])
    return accuracy
```

Once this has been performed it is then essential to correlate the effects that the anomalous data has on the system with its physical consequences. For example, if there are anomalies in the pump rate or control scheme in either system, then this could be further clarified by measuring the pressure level of

the system to determine the impact of these readings. Furthermore, after completing this the investigator could conduct simulations to validate further the correlations between the anomalous data and their corresponding physical effects, and then use techniques like the Pearson correlation coefficient [26] to quantify the strength of the relationship between the anomalous reading and what changing it does to the system. The visualisation of this data allows the investigator to view the attacks that have been performed using the filtered data to show the anomalies, including the type of the attack which allows them to quickly pinpoint the component of the system that is malfunctioning, allowing them to resolve this quickly and accurately. A summary of the data collected is shown below, including the anomalous values detected in the dataset which could indicate a cyber-attack for the selected features, and a sample of the data that is written to the CSV file predicting the attacks of each dataset:

4.2.2 Gas Pipeline Anomalies

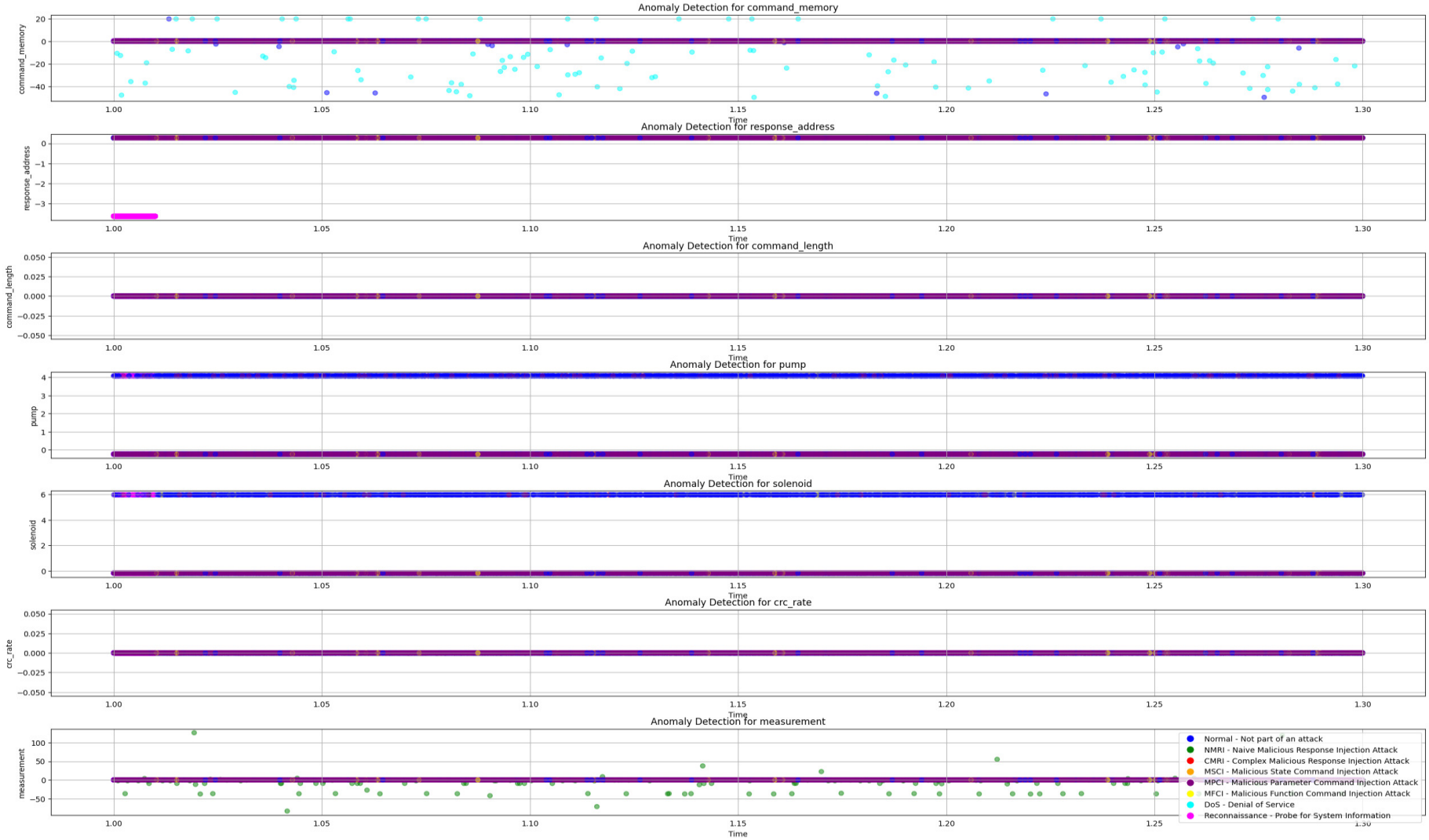


Figure 7: Gas Pipeline Anomaly Detection Data

4.2.3 Water Pipeline Anomalies

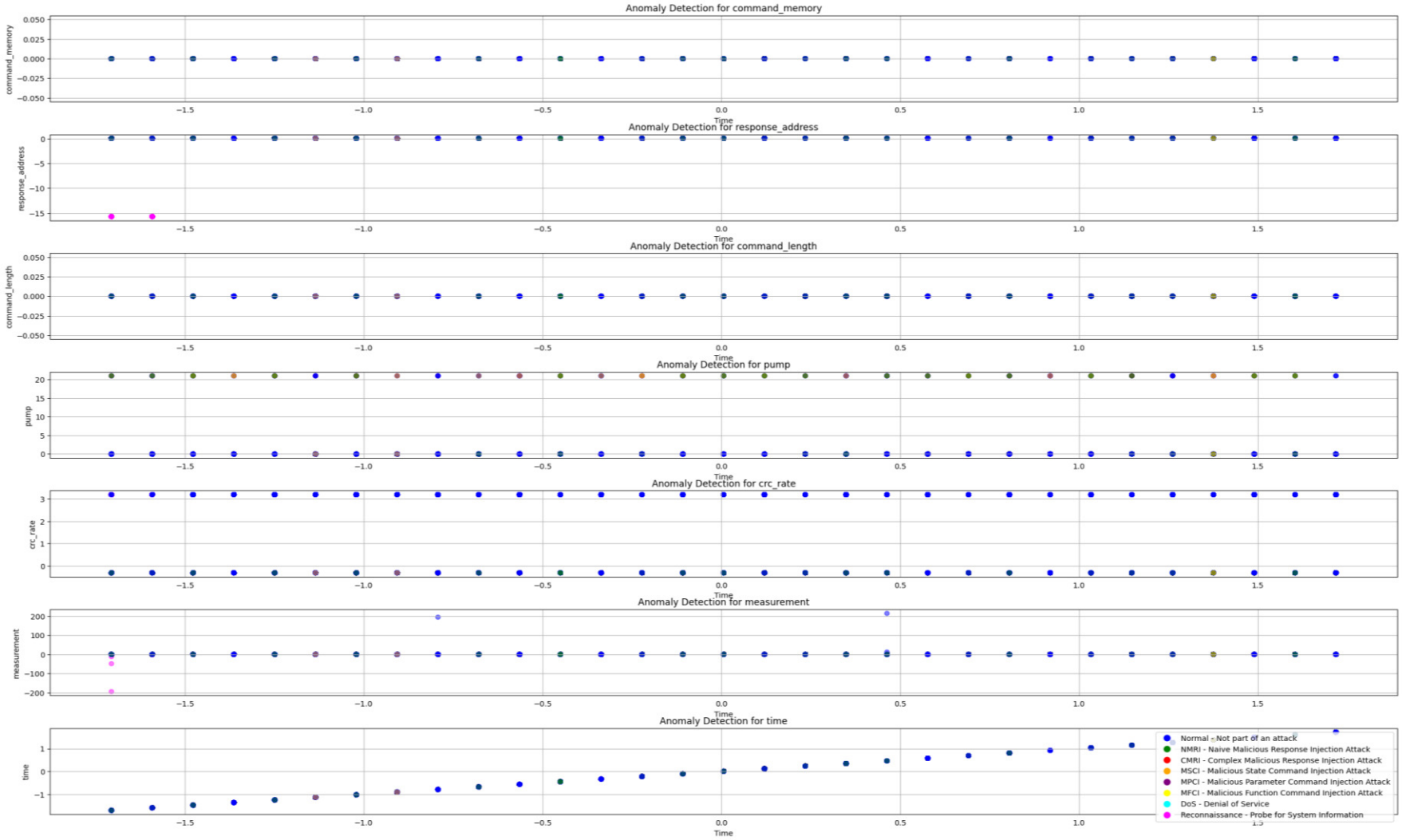


Figure 8: Water Pipeline Anomaly Detection Data

4.2.4 Sample of Result Prediction for Gas Pipeline

result	anomaly_isolation_forest	anomaly_local_outlier_factor	anomaly_combined	anomaly_rf_adjusted	estimated_result_Random Forest
0	1	1	2	1.9807912079078458	1
0	1	1	2	1.9807912079078458	1
0	1	1	2	1.9807912079078458	1
0	1	1	2	1.9807912079078458	1
0	1	1	2	1.9807912079078458	1
0	1	1	2	1.9807912079078458	1
6	1	1	2	1.9807912079078458	1
0	1	1	2	1.9807912079078458	1
6	1	1	2	1.9807912079078458	1
0	1	1	2	1.9807912079078458	1
0	1	1	2	1.9807912079078458	1
6	1	1	2	1.9807912079078458	1
6	1	1	2	1.9807912079078458	1
6	1	1	2	1.9807912079078458	1
0	1	1	2	1.9807912079078458	1
6	1	1	2	1.9807912079078458	1
0	1	1	2	1.9807912079078458	1
6	1	1	2	1.9807912079078458	1
6	1	1	2	1.9807912079078458	1

Figure 9: Gas Pipeline Result Prediction Sample

4.2.5 Sample of Result Prediction for Water Pipeline

result	anomaly_isolation_forest	anomaly_local_outlier_factor	anomaly_combined	anomaly_rf_adjusted	estimated_result_Random Forest
0	-1	-1	-2	-2.0	0
7	-1	-1	-2	-2.0	7
0	-1	-1	-2	-2.0	0
0	-1	-1	-2	-2.0	0
0	-1	-1	-2	-0.029518958993669537	0
0	-1	-1	-2	-0.02375399308745655	0
7	-1	-1	-2	-0.30885600633770094	7
7	-1	-1	-2	-2.0	7
0	-1	-1	-2	-2.0	0
7	-1	-1	-2	-2.0	7
0	-1	-1	-2	-0.031368176085928054	0
7	-1	-1	-2	-2.0	7
7	-1	-1	-2	-0.22447828292747599	7
7	-1	-1	-2	-2.0	7
0	-1	-1	-2	-2.0	0
0	-1	-1	-2	-2.0	0
0	-1	-1	-2	-0.011592237101103138	0
0	-1	-1	-2	-2.0	0
7	-1	-1	-2	-2.0	7

Figure 10: Water Pipeline Result Prediction Sample

4.3 Exploring the Relationships between Physical and Digital Consequences

The primary objective of this analysis phase of the framework is to correlate the relationships between the digital and physical evidence found during the investigation. This allows for the rapid detection of failed components within the system and can allow organisations to faster resolve any issues within the system after a breach has taken place. This can be completed using the Tommy Morris dataset above by identifying the components that were affected by the breach using the command address feature of the dataset, which can indicate how the surrounding environment was affected, coupled with the data readings at that timestamp which is indicative of a cyber-attack (temporal correlation).

To identify the components, the command address column can be used, which specifies the address of the component that the command was sent to from the SCADA System. As this dataset does not include any physical evidence this will instead be a theoretical investigation into what physical consequences may potentially occur based on the data and components within each system. This same process can be completed for any cyber-physical system however and should be completed when the system is installed regardless. The tables below in conjunction with the command address of the affected component, as well as the collection of physical evidence should be more than enough to be able to correlate the physical and digital consequences of a breach on either of these systems.

4.3.1 Physical Consequences of Water Pipeline System Breach

The potential physical consequences of the data points within the dataset above on each component can then be determined using the table below:

Component	*Potential Physical Consequences*	*Affected Column in the Dataset*
Primary Storage Tank	Leakages or ruptures leading to water loss, flooding, or structural damage to surrounding areas.	command_address, response_address, command_memory, response_memory, command_memory_count, response_memory_count, comm_read_function, comm_write_fun, resp_read_fun, resp_write_fun, sub_function, command_length, resp_length
	Corrosion causing the degradation of tank material and the potential for contamination of stored water.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, crc_rate, measurement, time
	Overfilling due to malfunctioning level sensors, leading to overflow and water wastage.	HH, H
	Underfilling results in inadequate water supply.	L, LL
	Tank collapses due to structural weaknesses or overpressure, causing significant damage and potential injuries.	control_mode control_scheme pump

Secondary Storage Tank	Leaks or malfunctions causing water loss, overflow, or inability to maintain proper water levels in the primary storage tank.	HH, H, L, LL
	Corrosion leading to tank deterioration and possible contamination of stored water.	command_address, response_address, command_memory, response_memory, command_memory_count, response_memory_count, comm_read_function, comm_write_fun, resp_read_fun, resp_write_fun, sub_function, command_length, resp_length
	Overfilling resulting in overflow and wastage of water.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, crc_rate, measurement, time
	Underfilling leading to insufficient water supply for replenishing the primary tank or other purposes.	HH, H, L, LL
Pump	Pump failure leading to inadequate water transfer between tanks, causing imbalance or insufficient water levels.	command_address, response_address, command_memory, response_memory, command_memory_count, response_memory_count, comm_read_function, comm_write_fun, resp_read_fun, resp_write_fun, sub_function, command_length, resp_length
	Motor or mechanical failures resulting in pump shutdown and interruption of water flow.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, crc_rate, measurement, time
	Overheating of pump components due to prolonged operation or blockages, leading to equipment damage or fire hazards.	HH, H, L, LL
	Pump cavitation/seal failures causing reduced pump efficiency or contamination.	HH, H, L, LL
Relief Valve	Malfunctions or blockages causing improper water flow between tanks, leading to pressure build-up or system overflows.	command_address, response_address, command_memory, response_memory, command_memory_count, response_memory_count,

		comm_read_function, comm_write_fun, resp_read_fun, resp_write_fun, sub_function, command_length, resp_length
	Failure to relieve excess pressure, resulting in tank rupture or damage to associated piping.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, crc_rate, measurement, time
	Inability to close valve properly, leading to either water loss or overflowing.	HH, H, L, LL
	Corrosion or degradation of valve components, affecting performance and reliability.	HH, H, L, LL
Water Level Sensor	Faulty or inaccurate readings lead to incorrect monitoring of water levels, causing overflows or underflows.	command_address, response_address, command_memory, response_memory, command_memory_count, response_memory_count, comm_read_function, comm_write_fun, resp_read_fun, resp_write_fun, sub_function, command_length, resp_length
	Sensor calibration errors resulting in inaccurate level measurements	HH, H, L, LL
	Sensor obstruction leads to incorrect readings or malfunctions.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, crc_rate, measurement, time
	Electrical failures cause the sensor to shut down or perform erratically, leading to unreliable level monitoring.	HH, H, L, LL
	Environmental factors such as temperature variations affect sensor performance and accuracy.	HH, H, L, LL

4.3.2 Physical Consequences of Gas Pipeline System Breach

Similarly, for the gas pipeline:

Component	*Potential Physical Consequences*	*Affected Columns within the Dataset*
Closed Loop Pipeline	Corrosion, damage, or faulty welds within the pipeline can result in gas leaks, posing safety	command_address, response_address, command_memory, response_memory, command_memory_count, response_memory_count,

	hazards for personnel and environmental risks.	comm_read_function, comm_write_fun, resp_read_fun, resp_write_fun, sub_function, command_length, resp_length
	Cracks, ruptures, or buckling of the pipeline due to incorrect pressure monitoring (overpressure) can result in catastrophic total system failures, gas leaks, and explosions.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, solenoid, crc_rate, measurement, time
	Gas leakage may also occur due to inaccurate pressure readings, which can lead to environmental contamination, safety hazards, and economic losses.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, solenoid, crc_rate, measurement, time
Air Pump	Mechanical failure due to incorrect readings misleading the control system, causing the air pump to operate beyond safe limits, resulting in equipment damage/failure.	command_address, response_address, command_memory, response_memory, command_memory_count, response_memory_count, comm_read_function, comm_write_fun, resp_read_fun, resp_write_fun, sub_function, command_length, resp_length
	Overheating may be caused by inaccurate sensor data which fails to detect excessive heat build-up in the air pump, leading to reduced efficiency, energy waste, and/or thermal damage.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, solenoid, crc_rate, measurement, time
Manual Release Valve	Incorrect readings may cause the valve to inaccurately open or close, leading to unintended blockages, flow disruptions, or pressure imbalances within the pipeline.	command_address, response_address, command_memory, response_memory, command_memory_count, response_memory_count, comm_read_function, comm_write_fun, resp_read_fun, resp_write_fun, sub_function, command_length, resp_length
	Faulty sensor data may mask actual safety hazards or vulnerabilities, increasing the risk of accidents, injuries, or catastrophic events during manual valve operations.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, solenoid, crc_rate, measurement, time
	Failure to release pressure due to sensor inaccuracies may result in environmental contamination or regulatory violations, posing a risk to public health and ecosystem integrity.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, solenoid, crc_rate, measurement, time
Solenoid Release Valve	Incorrect readings may result in the solenoid valve failing to fully	command_address, response_address, command_memory, response_memory,

	open/close, resulting in flow restrictions, pressure fluctuations, and inefficiencies within the system.	command_memory_count, response_memory_count, comm_read_function, comm_write_fun, resp_read_fun, resp_write_fun, sub_function, command_length, resp_length
	Inaccurate pressure data may prevent the solenoid valve from activating when needed, resulting in system unavailability to maintain normal operating conditions.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, solenoid, crc_rate, measurement, time
	Failure to release pressure may also result in environmental contamination.	gain, reset, deadband, cycle time, rate, setpoint, control_mode, control_scheme, pump, solenoid, crc_rate, measurement, time

4.3.3 Anomalous Values Determined by Attack Type and Deviation from Baseline

The columns with anomalous values for each attack type were also able to be determined using Python. This can give more of an indication of the type of attack used and therefore how components within the pipeline systems have been affected. This, coupled with the physical evidence gathered during the data acquisition phase of the framework should allow investigators to identify the full consequences of the attack. To do this all the columns in both the gas and water pipeline dataset were filtered by their attack type (excluding 0, as no attack is present) and then compared with the mean of all the rows where the result column is 0 for that feature. This was completed using Python and the attacks with their differences were outputted to a .txt file. The columns with anomalies depending on their attack type are shown in the tables below:

Gas Pipeline:

Attack Type	Columns with Significant Differences
1 - NMRI - Naive Malicious Response Injection Attack	resp_read_fun control_mode pump solenoid measurement result
2 - CMRI - Complex Malicious Response Injection Attack	resp_read_fun control_mode pump solenoid measurement result
3 - MSCI - Malicious State Command Injection Attack	resp_read_fun control_mode control_scheme pump solenoid

	measurement result
4 - MPCI - Malicious Parameter Command Injection Attack	resp_read_fun setpoint control_mode pump solenoid measurement result
5 - MFCI - Malicious Function Command Injection Attack	resp_read_fun sub_function setpoint control_mode pump solenoid measurement result
6 - DoS - Denial of Service	comm_read_function resp_read_fun control_mode pump solenoid measurement result
7 - Reconnaissance - Probe for System Information	response_memory response_memory_count resp_read_fun resp_write_fun resp_length control_mode pump solenoid measurement result

Water Pipeline:

Attack Type	Columns with Significant Differences
1- NMRI- Naive Malicious Response Injection Attack	control_mode pump crc_rate measurement result
2- CMRI- Complex Malicious Response Injection Attack	control_mode pump crc_rate measurement

	result
3- MSCI- Malicious State Command Injection Attack	HH H LL control_mode pump crc_rate measurement result
4- MPCI- Malicious Parameter Command Injection Attack	HH H L LL control_mode pump crc_rate measurement result
5- MFCI- Malicious Function Command Injection Attack	sub_function control_mode pump crc_rate measurement result
6- DoS- Denial of Service	control_mode pump crc_rate measurement result
7- Reconnaissance- Probe for System Information	response_memory response_memory_count resp_write_fun resp_length control_mode pump crc_rate measurement result

In the broader context of Cyber-Physical Systems beyond just the pipelines used in this analysis, the approach outlined can be adapted to any system where digital attacks can have physical consequences. When the specific attack type is unknown, establishing a baseline of measurements during normal operations is paramount. Instead of relying on the mean of readings with the result column of 0, averaging known good readings when the system is operating under standard conditions provides a suitable reference point when the attack type isn't known and is largely better as more data can be used to do this than is in this dataset. This baseline serves as a benchmark against which deviations can be detected, regardless of the nature of the attack. This has been completed using the following code:

```
"""
C21040310 - Jake Palmer
Attack Category Correlation to Correlate Digital and Physical Consequences
"""
import pandas as pd

def calculate_differences(normal_readings, anomalous_readings):
    # Align the columns of normal and anomalous readings
    common_columns = normal_readings.columns.intersection(anomalous_readings.columns)
    normal_aligned = normal_readings[common_columns]
    anomalous_aligned = anomalous_readings[common_columns]

    # Calculate the differences
    differences = anomalous_aligned.sub(normal_aligned.mean())
    return differences

def get_column_names_diff(differences, normal_readings):
    normal_mean = normal_readings.mean()
    return differences.columns[differences.abs().mean() > (normal_mean * 0.1)]  # Adjust the threshold as needed

def filter_by_attack_type(df, attack_type):
    return df[df['result'] == attack_type]

def save_differences_to_txt(output_data, pipeline_name):
    filename = f'differences_{pipeline_name}.txt'
    with open(filename, 'w') as txtfile:
        for line in output_data:
            txtfile.write('\n'.join(map(str, line)) + '\n')
```

To correlate the digital effects of the attack with its physical consequences, it is possible to map the above-affected columns to the physical components that they represent within each pipeline, also having the ability to trace the command and response addresses for these to identify the affected components. This digital evidence, along with any other evidence gathered during the physical evidence acquisition phase of the investigation (e.g. damage to the pipeline components, damage to the surrounding environment, etc) should be more than enough to rapidly identify and address any remaining issues present within either of these systems. At this point during the investigation, it should also be possible to identify the root cause of the breach, as it is now known how it propagated through the system, which commands were executed, and how the surrounding physical environment was affected by the breach.

4.4 Reporting the Findings from the Investigation

The reporting phase of the investigation must follow strict protocols and guidelines and must always adhere to sufficient legal standards. It is also crucial for the owner/manufacturers of the system that the evidence acquisition and analysis stages of the investigation are well documented to aid any future investigations, as well as any revisions needed to the systems or its components (such as firmware upgrades). This section of the framework will therefore provide some guidelines for reporting the findings of the investigations in a structured and comprehensive manner and will outline what specific features of components within the system need to be reported.

4.4.1 Digital Forensics Report Structure

A general structure of a report of the investigation should be as follows:

1. ***Summary of the Investigation:*** This should be a concise overview of the investigation findings, which highlights the key points, implications, and recommendations based on the evidence analysed during the investigation. This should serve as a quick reference for someone who may need a quick summary of what happened to the system, how this was resolved, and what steps were implemented to protect it from reproducing in the future.

2. ***Introduction:*** This should include a background of the investigation, including what occurred in the system, the name(s) of the investigator, and the dates and times of these events. This

should also briefly summarise the methodologies and techniques used during the analysis phase of the investigation.

3. *Components of the System:* This section should contain a detailed inventory of every component within the cyber-physical system, including serial numbers, firmware versions, the component's location, and a detailed description of that component's role within the system.

4. *Methodology:* This should be the main portion of the report, which includes the tools and techniques used during the investigation for data acquisition/analysis (Python libraries/other software used for analysis) and how the root cause of the breach was determined. All findings from the investigation, including anomalous values discovered and any other evidence acquired should be presented within this section.

5. *Digital and Physical Consequences:* This should detail the correlations between the digital and physical consequences that were observed and identified during the investigations, including any component failures, leaks, overflows, or any other environmental hazards.

6. *Recommendations/Conclusions:* These should be actionable recommendations at the end of the document specifying how the security of the system can be enhanced based on the key findings of the investigation, what preventative measures need to be implemented for the system to prevent future attacks and what monitoring techniques have been put into place for detecting a similar attack in the future.

4.4.2 Reporting Evidence Items in Chain of Custody

Following the strict protocols and guidelines governing forensic investigations, the chain of custody documentation for evidence collected during the investigation of the Cyber-Physical System must adhere to meticulous standards to ensure its legal validity and integrity. This document should meticulously detail the custody, handling, and transfer of each evidence item found from the moment it is discovered until its final disposition. For digital evidence, the chain of custody should include information such as the date and time of acquisition, the identity of the individual who collected the evidence, the storage media used, and any relevant metadata such as the file's MD5 hash, or version numbers. Similarly, for physical evidence, the chain of custody should record details such as the location and circumstances of discovery, packaging, and labelling procedures, and every transfer of custody, including signatures or acknowledgments from each custodian. An example chain of custody document for network files found during an investigation is as follows:

Chain of Custody Document	Case Number: 01
Evidence Item	Network Packet Capture (.pcap) Files
Description	Digital network traffic packet capture files obtained from the cyber-physical system.
Date and Timestamp of Acquisition	April 11th, 2024 2:06 pm
Location	Facility
Collector (with signature)	Jake Palmer
Storage Medium	External Hard Drive (Model 01234)

Transfers from:	Transfers to:
1	1
2	2
3	3
Metadata:	File size: 2GB MD5 Hash: f75e6b2f1e7f1a23e9fbac82c02c8378 Capture Duration: 24 Hours

4.4.3 Reporting Correlation with Physical Outcomes

Reporting the correlation between the physical and digital evidence that has been discovered during the investigation is crucial to this framework and it allows the person reading the report to fully assess the physical impact and the surrounding environment, this section should be relatively concise but detailed enough so that the cause of these effects is clearly outlined and the full impact of the breach on the system is identified. The structure that can be used for this section of the report is as follows:

1. *Identification of Physical Effects:* It is important to begin by summarising the observed physical effects on the system and any effects on its environment that resulted from the cyber-attack.

2. *Correlation Analysis:* Discuss the correlation between the physical effects of the breach and its digital traces found during the investigation. This should be backed up with the evidence found through the analysis phase, and all findings relating to this must be presented at this stage.

3. *Interpretation and Conclusions:* It is then important to offer insights into any broader contextual factors that may have exacerbated the effects of the breach, how the breach has been rectified, and what steps can be taken to ensure that the same attack cannot happen again.

Adhering to this reporting framework will ensure that all findings during the investigation into the breach are thoroughly reported and justified and will again ensure that crucial evidence items haven't been overlooked, and their chain of custody has been fully documented and ready for any legal proceedings that may occur.

5.0 Evaluation

The evaluation of any framework designed for analysing Cyber-Physical System data is crucial to ensuring its effectiveness, reliability, and practicality in real-world attack investigation scenarios. Therefore, a comprehensive evaluation plan is necessary to thoroughly assess the framework's theoretical performance. In this section, I have outlined the criteria used for evaluating the proposed framework and provide a detailed discussion of each criterion and given data on how the framework addresses each of these.

5.1.1 Criteria for Evaluation

The criteria that needs to be met for the evaluation of this framework is as follows:

- *Accuracy of Attack Prediction:* The accuracy of attack is a fundamental aspect of measuring the framework's performance, gauging its ability and accuracy to predict cyber-attacks based on selected data features. To evaluate this criterion, the framework's predictive capabilities with known attack data will be analysed. The accuracy of the predictions will be assessed and compared against the actual types of attacks observed in the dataset. Additionally, metrics such as precision, recall, and F1 score will be considered to provide a comprehensive evaluation of the prediction accuracy.

- *Correlation with Physical Consequences:* This criterion evaluates the framework's ability to correlate digital anomalies identified through data analysis with their corresponding physical consequences. The effectiveness of this correlation is essential for understanding the real-world impact of cyber-attacks on physical systems. To assess this criterion, it is crucial to analyse how well the framework identifies and links digital anomalies to potential physical outcomes, such as equipment failure, system downtime, or environmental damage identified using the attack types given by the Tommy Morris dataset [18].

- *Scalability to all Cyber-Physical Systems:* The scalability of the framework to different cyber-physical systems is essential to ensure its broad applicability and relevance beyond specific datasets or scenarios. This criterion evaluates whether the framework can be adapted and applied to various cyber-physical system environments, including industrial control systems (ICS), smart grids, autonomous vehicles (AVs), and healthcare systems. To evaluate scalability, we will assess how easily the framework can accommodate different types of data, alternate system architectures, and attack scenarios.

- *Further Research:* This criterion focuses on identifying any limitations or drawbacks of the framework and proposing areas for further research and improvement. This will critically evaluate the strengths and weaknesses of the framework, considering factors such as its robustness to noise and uncertainty, and usability in real-world settings. Based on this evaluation, potential avenues for future research will be determined, such as incorporating additional features or data sources or conducting field experiments to validate the framework's performance under different conditions.

5.1.2 Accuracy Levels of Attack Predictions in Gas Pipeline Dataset

By splitting the data for each pipeline into sets of training and testing and outputting this data to the '**predicted_data_gas_pipeline.csv'** and the '**predicted_data_water_pipeline.csv'** files it was possible to compare the accuracy of each machine learning method to predict attacks based on known data. This data is shown below for each pipeline and the corresponding machine-learning prediction algorithms:

Gas Pipeline:

Machine Learning Technique	*Accuracy (%)*	*Precision*	*Recall*	*F1-Score*
Random Forest	92.4	0.871	0.924	0.895
Decision Tree	92.4	0.871	0.924	0.895
Support Vector Machine	92.3	0.870	0.923	0.895
K-Nearest Neighbours	93.2	0.879	0.931	0.903
Gaussian Naïve Bayes	20.1	0.173	0.206	0.176
Logistic Regression	91.5	0.861	0.915	0.883

Amongst the machine learning techniques that were applied to the gas pipeline dataset, Random Forest, Decision Tree, and Support Vector Machine demonstrate high accuracy rates varying around 92.4%. These techniques indicate that all of these algorithms are well suited to predicting attacks in cyber-physical systems, boasting precision, recall, and F1-score values consistently surpassing 0.87. These results show Random Forest, Decision Tree, and Support Vector Machine are all highly accurate when handling the intricate nature of gas pipeline data, exhibiting resilience to noise and extraneous features present in this dataset. Notably, K-Nearest Neighbours achieved the highest accuracy score of 93.2%, proving that anomalies amongst certain features within the dataset are indicative of a particular cyber-attack, and therefore consequently a corresponding physical effect as well.

On the other hand, Gaussian Naïve Bayes presents a different picture with notably lower accuracy at 20.1% and subpar precision, recall, and F1-score values. Its struggle to classify instances effectively suggests that it's not well-equipped to handle the complexities of the gas pipeline dataset, and therefore cyber-physical system datasets altogether. This can be attributed to Gaussian Naïve Bayes' reliance on the assumption of feature independence and normal distribution, which proves inadequate for capturing the intricate relationships within the dataset. Similarly, Logistic Regression, while still achieving a respectable accuracy of 91.5%, lags slightly behind its counterparts in precision, recall, and F1-score metrics. Furthermore, despite its ability to model linear relationships between features and outcomes, Logistic Regression also appears to struggle with capturing the nuanced intricacies present in the dataset, indicating potential limitations in its predictive capabilities when working with this type of data. Based on their high accuracy rates and robust performance metrics when working with this data, a selection of Random Forest, Decision Tree, Support Vector Machine, and K-Nearest Neighbours can be seen as the preferred algorithms for predicting cyber-attacks in Cyber-Physical System datasets.

5.1.3 Accuracy Levels of Attack Predictions in Water Pipeline Dataset

The accuracy metrics of the prediction techniques when using the water pipeline dataset are as follows:

Water Pipeline:

Machine Learning Technique	*Accuracy (%)*	*Precision*	*Recall*	*F1-Score*
Random Forest	97.2	0.945	0.972	0.958
Decision Tree	97.2	0.945	0.972	0.958
Support Vector Machine	97.2	0.944	0.972	0.958
K-Nearest Neighbours	0.972	0.948	0.972	0.959
Guassian Naïve Bayes	0.810	0.973	0.810	0.881
Logistic Regression	0.972	0.944	0.972	0.958

In this analysis of the water pipeline dataset, Random Forest, Decision Tree, and Support Vector Machine again exhibit notably high accuracy rates, hovering around 97.2%. These techniques demonstrate their effectiveness in predicting attacks within Cyber-Physical Systems, showcasing precision, recall, and F1-score values consistently exceeding 0.94. The results indicate that Random Forest, Decision Tree, and Support Vector Machine are well-suited for handling the complexities of water pipeline data as well as gas pipeline data, meaning that these results and the algorithm characteristics show that they are well-suited for the analysis of cyber-physical systems data. Additionally, K-Nearest Neighbours achieves a high accuracy score of 97.2%, and as this is the highest accuracy level for both datasets it is clear that this algorithm is best used when working with cyber-physical system data.

Conversely, Gaussian Naïve Bayes again presents a relatively poor performance with notably lower accuracy of 81.0% and subpar precision, recall, and F1-score values. Its struggle to effectively classify instances suggests its inadequacy in navigating the intricacies of the water pipeline dataset and, by extension, cyber-physical system datasets. Logistic Regression also faces similar challenges in capturing the correlations present in the water pipeline dataset between the result and the selected features, indicating potential limitations in its predictive capabilities for this type of data.

5.1.4 Correlation with Physical Consequences and Scalability

By using the Tommy Morris Dataset [18] for this project, it is clear that the true physical consequences of the attack aren't given due to it being a simulation, with this data not provided in this dataset. Instead, for this section, I aim to discuss how well the framework performs when correlating the digital traces of the attack with the potential consequences of the attack, as these aren't known with this dataset. Firstly, the framework does identify a clear investigative structure to ensure all evidence, both physical and digital are found in an order that allows the most volatile evidence to be preserved first. This is a crucial requirement for any digital forensics investigation and ensures that important pieces of evidence are gathered systematically to ensure that the investigation doesn't miss any artifacts that may be overlooked and provides a method for analysing each one, regardless of the type of system to make it applicable to any Cyber-Physical System, not just the water and gas pipelines used for analysis.

Furthermore, using the techniques I outlined it is possible to correlate the digital and physical consequences despite the absence of true physical data within the Tommy Morris dataset I used. Furthermore, by utilising a combination of anomaly detection, machine learning prediction algorithms, and pattern analysis, the framework successfully identifies potential data points (and by using the command/response address, and their corresponding components) that are indicative of a cyber-attack. These are then mapped to hypothetical physical relationships with potential effects on components of the pipeline or the environment, based on established cause-and-effect relationships within pipeline systems.

Additionally, the framework's ability to categorise digital anomalies based on known attack types is crucial during a forensic investigation. By predicting the classification of anomalous behaviour into distinct categories, such as reconnaissance attacks, command injections, or denial-of-service attacks, investigators can infer the likely objectives and methods of the perpetrators, allowing them to rectify potential issues faster. This categorisation facilitates the correlation of digital traces with potential consequences by identifying the specific mechanisms through which cyber-attacks may manifest in the physical domain.

Furthermore, it is essential when applying this framework to ensure that the full context of the system is considered. For example, while some of the communication protocols (MODBUS) are present in most autonomous vehicles similar to the pipelines, their sensors and actuators are completely different, and therefore require different data acquisition and analysis methods. This framework ensures that the data acquisition, analysis, and reporting are consistent with any cyber-physical system. This framework also highlights the importance of contextual analysis - the fact that while the components and features of water and gas pipelines are used in this example analysis, the same theoretical principles may be applied in a forensic investigation of any other Cyber-Physical System.

5.2 Further Research and Improvements

The framework proposed within this project aims to explore cyber-attacks within Cyber-Physical Systems, aiming to establish correlations between digital intrusion events and their impacts on the physical infrastructure of the system and its surrounding environment. The approach theoretically achieves this by exploring the Tommy Morris Dataset as a foundational resource, utilizing its insights to speculate on potential real-world consequences. However, for comprehensive validation and robustness, the framework should be applied to a broader and more adaptable dataset, ideally through the simulation of cyber-physical systems. Originally, the initial plan for this project included this crucial step of testing the framework with a larger, more diverse dataset and simulating cyber-physical system scenarios. However, practical limitations and time constraints within accessing and using the system obstructed the completion of this aspect within the project's timeline. This next section will detail the plan for this so that future work on this can be completed at a later date.

5.2.1 Breach Simulation Plan

The system that will be used to simulate a breach in this example is the Industrial Control Trainer produced by LJ Create [13], which offers a platform specifically designed for practical learning surrounding Industrial Control Systems (ICS). It comprises various components such as sensors, actuators, programmable logic controllers (PLCs), and Human Machine Interfaces (HMIs). This model offers the ability to interact with each component individually in a controlled manner, which is crucial for the investigation. The investigation of this system will provide the following data:

- ***Event Logs:*** Generated by sensors or the PLC, which document any significant system events.

- ***System Logs:*** Including system configuration settings of the PLC, components, and networked devices.

- ***Performance Metrics:*** This includes response times, throughput, and resource utilisation, allowing the overall impact of the security incident to be measured.

- ***Network Traffic Analysis:*** The network traffic of the device can reveal patterns indicative of normal and abnormal behaviour, as seen in the Tommy Morris dataset [18].

- ***Simulation Environment Configuration:*** The configuration settings of the simulation environment will also be documented to ensure that the breach is fully reproducible using the exact settings, maintaining reliability and accuracy for any subsequent simulations.

A summary of each component within the industrial work cell is shown below:

Component	*Description*
Manual Override Switch and Controls (including LEDs and sensor indicators)	Allows the operator to manually control the system's functions, override automated processes, and provide visual feedback by activating LEDs.
Conveyor Belt	Transports materials and parts from one stage to another within the automation system.
Pressurised Pistons	Used to store correctly sorted parts produced by the system.
Storage Box	Used to hold parts or materials before or after they have been processed within the system.

Programmable Logic Controller (PLC)	Handles input, output and monitors the system. This uses ladder logic to control the connected devices and is similar to common Siemens PLCs in large-scale industrial applications.

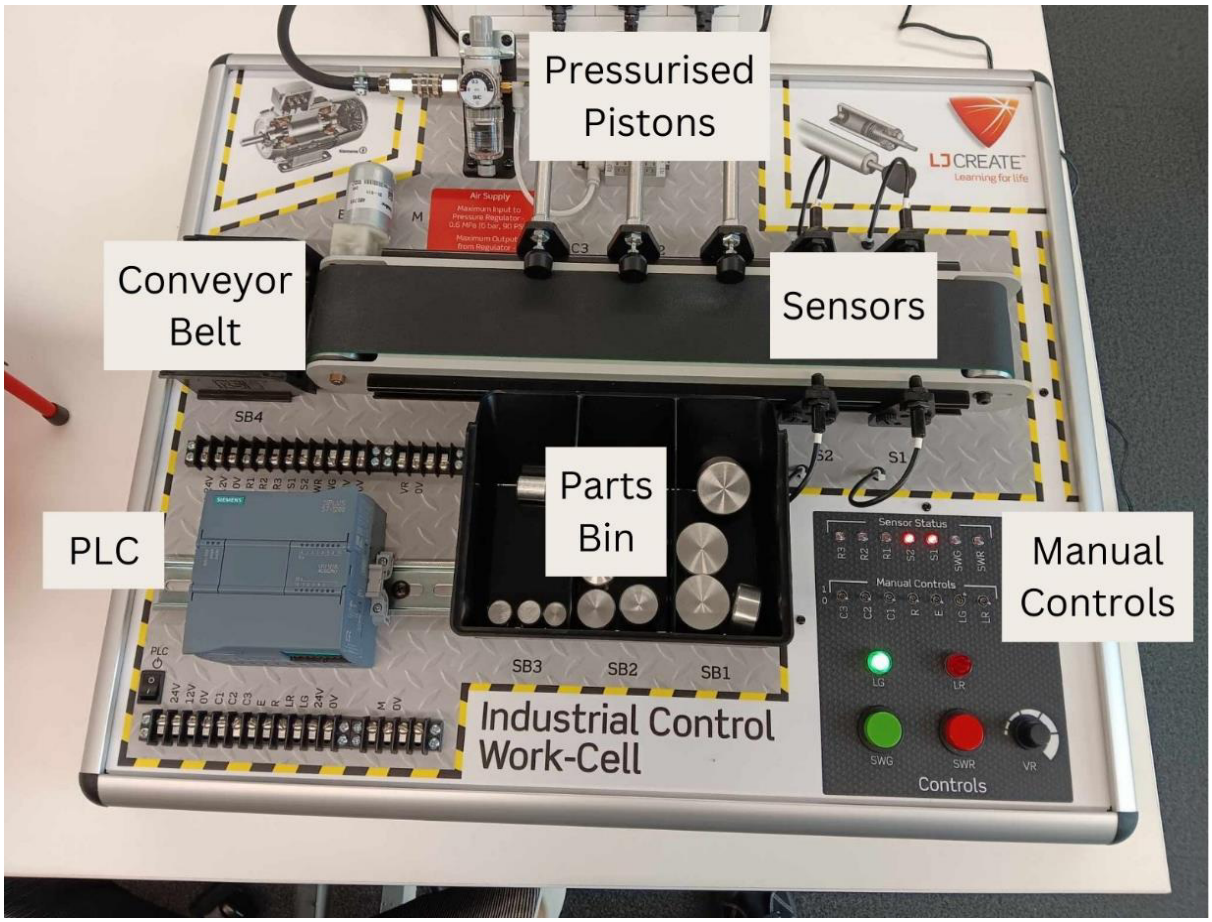


Figure 11: Industrial-Control Work Cell by LJ Create [13].

5.2.2 Attack Vectors for Industrial Work Cell

Many attacks could be performed on an industrial work cell such as this one, such as DoS, MitM, Malware Injection, Firmware Tampering or physically interfering with the system. The one that I wanted to perform was a denial-of-service attack targeting the operation of the PLC. This aims to overwhelm the PLC with traffic and prevent it from executing its designated control tasks, causing the system to effectively shut down and potentially causing a cascade of failures throughout the system. This is a good attack to use in this scenario as it directly disrupts the centre of the work cells' operations without any need for physical access, which is crucial as the most common cyber-attacks affecting these systems (previously mentioned) occur without the need for physical access to the system. Furthermore, this will affect most of the data that is collected by the PLC, meaning that it will be readily available for analysis quickly, and is easier to log than most other attacks.

Other attacks such as Man-in-the-Middle (MitM), Malware Injection, Firmware Tampering, or physical interference with the system were also considered; however, these methods typically require a higher level of access privileges to execute successfully. A Denial-of-Service attack, on the other hand, can be quite straightforward and can be launched remotely, making it a more accessible and common form of disruption for attackers. Additionally, the impact of a DoS attack can be immediate and widespread, affecting not just the targeted PLC but also the entire operation dependent on it. This kind of attack can also be sustained over a long period, causing prolonged downtime to the system. It's also an attack that can be difficult to trace back to its source, adding an extra layer of challenge for the security teams trying to mitigate its effects. Therefore, in this scenario, targeting the PLC with a DoS attack could be seen as a highly effective strategy for disrupting the industrial work cell's operations and allowing the attack to be reverse-engineered when forensically analysing the data taken from the system to create this framework.

Several attack vectors could be used to perform a denial-of-service (DoS) attack targeting the operation of the PLC in the industrial work cell. One approach might involve flooding the network with an overwhelming amount of traffic directed at the PLC, which could render it unable to process legitimate control commands from the operators. This could be achieved by exploiting network vulnerabilities or by using a botnet to generate excessive traffic within the system. Another method might be to send rapid, continuous requests to the PLC from within the network, potentially by compromising a connected device, such as a sensor or a human-machine interface. This internal DoS attack could disrupt the PLC's normal operations without the need for external traffic. Additionally, an attacker could target the PLC with specially crafted packets that exploit vulnerabilities in the PLC's communication protocol, causing it to crash or become unresponsive.

References

1. Alphonsus, E.R. and Abdullah, M.O. (2016). A review of the applications of programmable logic controllers (PLCs). Renewable and Sustainable Energy Reviews, [online] 60, pp.1185–1205. doi: https://doi.org/10.1016/j.rser.2016.01.025
2. ArcGIS (2023). ArcGIS Online. Arcgis.com. Available at: https://www.arcgis.com/index.html
3. Årnes, A. (2017). *Digital Forensics*. *Google Books*. John Wiley & Sons. Available at: https://books.google.co.uk/books?id=xqNaDwAAQBAJ&lpg=PR15&ots=q78ObjjYdz&dq=digital%20forensics&lr&pg=PR1#v=onepage&q=digital%20forensics&f=false
4. Baryamureeba, V. and Tushabe, F. (2004). *The Enhanced Digital Investigation Process Model*. Available at: https://dfrws.org/wp-content/uploads/2019/06/2004_USA_paper-the_enhanced_digital_investigation_process_model.pdf
5. Borlase, S. (2017). Smart Grids: Infrastructure, Technology, and Solutions. [online] Google Books. CRC Press. Available at: https://books.google.co.uk/books?hl=en&lr=&id=UTbNBQAAQBAJ&oi=fnd&pg=PP1&dq=what+are+smart+grids%3F&ots=eLOXl1PSOv&sig=7TR00yPXkdNVtAPW5TtH4uAm2Ss&redir_esc=y#v=onepage&q=what%20are%20smart%20grids%3F&f=false
6. Exterro. (n.d.). FTK Imager Version 4.7.1. Available at: https://www.exterro.com/ftk-product-downloads/ftk-imager-version-4-7-1
7. G. Mohay, "Technical challenges and directions for digital forensics," First International Workshop on Systematic Approaches to Digital Forensic Engineering, Taipei, Taiwan, 2005, pp. 155-161, doi: https://doi.org/10.1109/SADFE.2005.24
8. Juliussen, E. (2022). *Automotive Cybersecurity: More Than In-Vehicle and Cloud*. [online] EE Times Europe. Available at: https://www.eetimes.eu/automotive-cybersecurity-more-than-in-vehicle-and-cloud/
9. K. Huang, C. Zhou, Y.-C. Tian, S. Yang and Y. Qin. (2018). "Assessing the Physical Impact of Cyberattacks on Industrial Cyber-Physical Systems," IEEE *Transactions on Industrial Electronics*, vol. 65, no. 10, pp. 8153-8162, doi: https://doi.org/10.1109/TIE.2018.2798605.
10. K. Nance and D. J. Ryan, "Legal Aspects of Digital Forensics: A Research Agenda," *2011 44th Hawaii International Conference on System Sciences*, Kauai, HI, USA, 2011, pp. 1-6, doi: https://doi.org/10.1109/HICSS.2011.282
11. Lee, E.A. (2008). Cyber Physical Systems: Design Challenges. *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*. [online] doi: https://doi.org/10.1109/isorc.2008.25
12. Lee, J., Jin, C. and Bagheri, B. (2017). Cyber physical systems for predictive production systems. *Production Engineering*, 11(2), pp.155–165. doi: https://doi.org/10.1007/s11740-017-0729-4
13. LJ Create. (n.d.). Industrial Control Trainer. [online] Available at: https://ljcreate.com/uk/engineering/industrial-control-trainer/

14. Lucia, S., Kögel, M., Zometa, P., Quevedo, D.E. and Findeisen, R. (2016). Predictive control, embedded cyberphysical systems and systems of systems – A perspective. *Annual Reviews in Control*, 41, pp.193–207. doi: https://doi.org/10.1016/j.arcontrol.2016.04.002
15. M. Cebe, E. Erdin, K. Akkaya, H. Aksu and S. Uluagac, "Block4Forensic: An Integrated Lightweight Blockchain Framework for Forensics Applications of Connected Vehicles," in *IEEE Communications Magazine*, vol. 56, no. 10, pp. 50-57, OCTOBER 2018, doi: https://doi.org/10.1109/MCOM.2018.1800137
16. Mishra, S., 2019. Forensic Investigation Framework for Complex Cyber Attack on Cyber Physical System by Using Goals/Sub-goals of an Attack and Epidemics of Malware in a System. In Recent Trends in Communication, Computing, and Electronics: Select Proceedings of IC3E 2018 pp. 491-504. doi: https://doi.org/10.1007/978-981-13-2685-1_47
17. Mohamed, N., Al-Jaroodi, J. and Jawhar, I. (2020). Cyber–Physical Systems Forensics: Today and Tomorrow. *Journal of Sensor and Actuator Networks*, 9(3), p.37. doi: https://doi.org/10.3390/jsan9030037
18. Morris, T. (2015). Industrial Control System (ICS) Cyber Attack Datasets - Tommy Morris. Available at: https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets
19. N. H. Ab Rahman, W. B. Glisson, Y. Yang, and K.-K. R. Choo, "Forensic-by-Design Framework for Cyber-Physical Cloud Systems," in *IEEE Cloud Computing*, vol. 3, no. 1, pp. 50-59, Jan.-Feb. 2016, doi: https://doi.org/10.1109/MCC.2016.5
20. N. Kumari and A. K. Mohapatra, "An insight into digital forensics branches and tools," 2016 International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT), New Delhi, India, 2016, pp. 243-250, doi: https://doi.org/10.1109/ICCTICT.2016.7514586
21. Nair, M.M., Tyagi, A.K. and Goyal, R. (2019). Medical Cyber Physical Systems and Its Issues. Procedia Computer Science, 165, pp.647–655. doi: https://doi.org/10.1016/j.procs.2020.01.059
22. Pivoto, D.G.S., de Almeida, L.F.F., da Rosa Righi, R., Rodrigues, J.J.P.C., Lugli, A.B. and Alberti, A.M. (2021). Cyber-physical systems architectures for industrial internet of things applications in Industry 4.0: A literature review. *Journal of Manufacturing Systems*, 58, pp.176–192. doi: https://doi.org/10.1016/j.jmsy.2020.11.017
23. Positive Technologies (2018). *Industrial companies: attack vectors*. Ptsecurity.com. Available at: https://www.ptsecurity.com/ww-en/analytics/ics-attacks-2018/
24. Rho, S., Vasilakos, A.V. and Chen, W. (2016). Cyber physical systems technologies and applications. *Future Generation Computer Systems*, [online] 56, pp.436–437. doi: https://doi.org/10.1016/j.future.2015.10.019
25. Sanislav, T., Sherali Zeadally, Mois, G. and Hacène Fouchal (2017). Multi-agent architecture for reliable Cyber-Physical Systems (CPS). doi: https://doi.org/10.1109/iscc.2017.8024524.
26. Turney, S. (2022). Pearson Correlation Coefficient (r) | Guide & Examples. [online] Scribbr. Available at: https://www.scribbr.com/statistics/pearson-correlation-coefficient/#:~:text=The%20Pearson%20correlation%20coefficient%20(r

27. V. R. Kebande and I. Ray, "A Generic Digital Forensic Investigation Framework for Internet of Things (IoT)," 2016 IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud), Vienna, Austria, 2016, pp. 356-362, doi: https://doi.org/10.1109/FiCloud.2016.57
28. Vincze, E.A. (2016). Challenges in digital forensics. *Police Practice and Research*, pp.183–194. doi: https://doi.org/10.1080/15614263.2015.1128163
29. von Solms, S., Louwrens, C., Reekie, C., Grobler, T. (2006). A Control Framework for Digital Forensics. In: Olivier, M.S., Shenoi, S. (eds) Advances in Digital Forensics II. DigitalForensics 2006. IFIP Advances in Information and Communication, vol 222. Springer, Boston, MA. https://doi.org/10.1007/0-387-36891-4_27
30. Wireshark (2017). Wireshark · Go Deep. Wireshark.org. Available at: https://www.wireshark.org
31. www.kaspersky.com. (2022). Spyware on the rise, 'restless' building automation: OT threats in 2022. Available at: https://www.kaspersky.com/about/press-releases/2022_spyware-on-the-rise-restless-building-automation-ot-threats-in-2022
32. www.microsoft.com. (n.d.). Power BI - Data Visualization | Microsoft Power Platform. [online] Available at: https://www.microsoft.com/en-us/power-platform/products/power-bi
33. www.simplymodbus.ca. (n.d.). About Modbus TCP | Simply Modbus Software. [online] Available at: https://www.simplymodbus.ca/TCP.htm
34. Zhang, P. (2010). Advanced Industrial Control Technology. [online] Google Books. William Andrew. Available at: https://books.google.co.uk/books?hl=en&lr=&id=OPlhfkU05DMC&oi=fnd&pg=PP2&dq=what+are+industrial+control+systems+&ots=o0BpUBPgWk&sig=tkEd5btVK9TQ6RtlDB4BWnEvYYU&redir_esc=y#v=onepage&q=what%20are%20industrial%20control%20systems&f=false