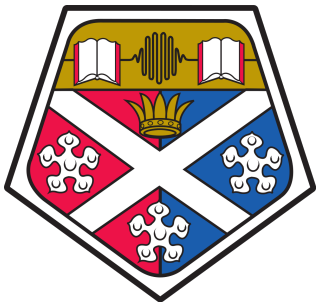


Security of Robotic Workflows

Ryan Karim Shah

A thesis presented for the degree of

Doctor of Philosophy

Department of Computer and Information Sciences

University of Strathclyde

Declaration

This thesis is the result of the author's original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree.

Signed:  Date: 20th October 2022

Acknowledgements

I would like to first thank my supervisors Shishir Nagaraja, Chuadhry Mujeeb Ahmed and Paul Duncan, for their continuous support and guidance throughout my PhD. I would also like to further thank Crawford Revie for his support as my first supervisor in Shishir's absence and helping me to the finish line. It has been a pleasure working alongside all of you and I look forward to collaborations in the future. I am also grateful to EPSRC for funding this PhD through their iCASE studentship program (EP/11288S170484-102), as well as the National Physical Laboratory (NPL) Scotland providing funding, invaluable information and support through the Data Science program. I would like to thank my parents, Sarah and Fazal, for their continued support and love which helped me achieve my goals. Furthermore, I am also grateful for the support and encouragement from Claire, Liam and the rest of my immediate family, who have powered me through numerous breakdowns and other low points throughout. Finally, I would like to dedicate an acknowledgement to my Poppy (Ahmed), Nana (Laura) and Granny (Penny), who have given me strength and wisdom in the best and worst of times, and the courage to pursue and achieve my dreams. I love you all.

Abstract

Recent advances in computer science, artificial intelligence and engineering has pioneered the field of robotics, bringing guarantees of higher levels of accuracy and lowered complications in a wide array of environments such as the automotive, manufacturing and healthcare industries. These environments we interact with on a daily basis are becoming increasingly connected, leaving many of these robotic systems vulnerable to a new set of threats and attacks from both a physical and cyber standpoint. Upon review of the robotics security landscape, the focus of the thesis is split into two parts.

The first part of this thesis looks at the capabilities of a passive attacker in both the cyber and physical domain. Existing literature focuses on active attackers with little attention paid to passive attackers. If an attacker is able to passively gather information about robot behaviours, such as how it moves, they could use this information to reconstruct entire operational workflows. For example, in surgical settings, if movement information was captured, then entire surgical procedures could be reconstructed. Combined with other information sources, such as patient admission and exit times, patient privacy could be compromised. Upon review of teleoperated robot architectures, three side channel attacks are investigated. The first side channel is traffic analysis in the cyber domain, wherein an attacker eavesdrops on the encrypted communication link between a robot and its controller, using traffic features to fingerprint robot movements and workflows. The second side channel leverages unintentional acoustic emanations in the physical domain as a robot moves and acoustic char-

acteristics are exploited for fingerprinting. The third and final side channel explored is radio frequency, where unintentional emissions of radio frequencies from microprocessors and motors are captured and analysed to fingerprint movements and workflows. Upon evaluation of all three side channels, radio frequency is the most successful with at least 96% accuracy. The acoustic and traffic analysis side channel, while also useful to an attacker, show lowered accuracy in comparison.

The second part of this thesis pertains to securing calibration for robotic systems. The calibration ecosystem intends to shift to a digital environment to keep up with technological advances. However, existing processes require immediate change in order to scale and remain robust to an evolved threat landscape. Specifically, little attention has been paid to the security of robot calibration and several inadequacies need to be addressed, including: efficiency, availability, integrity and tamper-resistance, confidentiality and managing conflicts between interacting parties in the calibration ecosystem. To address these challenges, two solutions are explored. First, blockchains adequately meet these required system properties and significantly outperform the current state-of-the-art in calibration traceability. While these properties are met by the proposed blockchain solution, the enforcement of some of these security properties – namely integrity, confidentiality and managing conflicts of interest – come with a set of information flows that present an interesting access control challenge. Specifically, it is important to verify that an individual verifying the calibration of a device, or even calibrating a device, has the appropriate rights to do so. The second solution explored demonstrates that existing models cannot adequately manage the unique information flows, ultimately requiring a novel unification of three existing models that outperforms

traditional models and can scale well with robots and IoT.

Ultimately, this thesis provides a review on the robotics threat landscape and identifies open challenges, to which several passive attacks and solutions are explored in both the cyber and physical domains. Further, this thesis also provides the first insights into a completely novel aspect of robotics security that needs careful consideration – securing the calibration of robots.

Contents

Securing Calibration in Robotics Environments

Appendices

List of Figures

List of Tables

Published Work

Some research presented within this thesis has appeared previously in the following publications.

Conference Papers

- Shah, R. and Nagaraja, S. **A Unified Access Control Model for Calibration Traceability in Safety-critical IoT**. – *In International Conference on Information Systems Security*, pages 3-22. December, 2020. Springer, Cham.

Pre-prints

- Shah, R., McIntee, M., Nagaraja, S., Bhandary, S., Arote, P. and Kuri, J. **Secure Calibration in High-Assurance IoT: Traceability for Safety Resilience**. – *arXiv*. 2019.

- Shah, R., Ahmed, C.M., Nagaraja, S. **Can You Still See Me?: Reconstructing Robot Operations Over End-to-End Encrypted Channels**. – *arXiv*. 2022.

Posters

- Shah, R. and Nagaraja, S. **Unified Access Control for Surgical Robotics**. – *In Proceedings of the 24th ACM Symposium on Access Control Models and Technologies*, pages 231-233. May, 2019.

- Shah, R., Ahmed, C.M., Nagaraja, S. **Can You Still See Me?: Reconstructing Robot Operations Over End-to-End Encrypted Channels**. – *In Proceedings of the 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 298-300. May, 2022.

1 — Introduction

The field of robotics was envisioned many years ago, with ancient Greek engineers pioneering this space with the likes of water clocks with automatons. The term robot was later coined and used by writers such as Capek [1] and Lang [2]. However, it was not until the early 1940's where Asimov introduced the term *robotics* in his book "Runaround" [3] which introduces the laws of robotics to ensure they can operate safely with humans. It was around this period where the use of robotics was truly thought about with regard to an idealised world of automation and intelligence, bringing promises of benefits to society and industry. With a steady progression and increased interest in robotics, many early systems were developed ranging from a variety of robotic arms [4–6], to mobile robotic systems [7, 8]. However, with advancements in computer science, engineering and manufacturing, the advent of modern robotics came to fruition.

Modern robotic systems have seen an increase in adoption in a wide array of application areas, including industrial, surgical and automotive, among many others [9]. The adoption of such systems stems from the peremptory need for higher accuracy, precision and efficiency. For example, in surgical environments, higher accuracy and precision for even a simple scalpel incision could mean the difference between life and death of a patient [10]. In the case of industrial settings such as manufacturing, aside from accuracy the aim is to also promote higher levels of efficiency to increase the output of production lines within supply chains.

With regard to concerns surrounding error and resulting liabilities,

these were most prominent for early robots in the physical domain as many were not connected systems. Simply, the most common attack on these robots was physical compromise. This includes tampering with sensors or actuators to cause operational workflows to halt or deteriorate operational ability to hinder any benefits they would otherwise bring. A robotic workflow in this thesis corresponds to a single operation (e.g. picking up from a conveyor belt and placing it down on another) made up one or a series of individual robot movements making use of at least one degree-of-freedom. With advancements in engineering and computer science, we now see the use of sensing equipment and artificial intelligence to mitigate or prevent physical compromise in the form of safety mechanisms. For example, sensors may be used to detect when humans or objects in the robot's operating environment are too close in physical proximity and may result in triggering the stopping mechanism or avoidance of the obstacle where possible. However, as new defence strategies are employed, new attacks arise, such as the use of jammers or lasers to disrupt sensing equipment [11, 12], or manipulating training data in machine learning models resulting in incorrect behaviours [13, 14]. Aside from just the physical domain, becoming Internet-connected significantly expands the threat landscape and exposes robotic systems to attack and compromise in the cyber domain, resulting in safety becoming an increasing concern from a cybersecurity viewpoint.

Existing work in the area of robotics security primarily focuses on attacks in the cyber domain involving an active adversary. These attacks range from modifying control messages to the robot and feedback sent to controllers (integrity compromise) [15], to eavesdropping on [16,17] and disrupting robot networks (denial-of-service) hindering

normal operation (availability compromise) [18–20]. Unfortunately, however, little attention has been paid to the capability of a passive adversary where reconnaisance can lead to consequences that are just as devastating. While various techniques can be used to eavesdrop on a robot, the focus in this thesis is on passive side channel attacks and the impact information leakage from mounting such attacks can have in a robotics environment. Specifically, three side channels are explored which aim to exploit information leakages to compromise robotic workflows. If an attacker is able to learn about what movements a robot is carrying out, entire workflows can be revealed. For example, in surgical settings, movement patterns may correlate with known surgical procedures and in combination with other metadata, such as patient admission and exit times, this could result in a compromise of patient privacy. From an industrial perspective, the leakage of industrial workflows could be used in a malicious manner (e.g. selling on to competing companies) and can result in the compromise of operational confidentiality of targeted organisations [21, 22].

The first side channel explored is traffic analysis, which aims to eavesdrop on the communication link between the robot and controller. By monitoring traffic patterns and using a shallow neural network architecture, an attacker can fingerprint (classify) robot movements with at least 60% accuracy even when traffic is encrypted under TLS, with more fine grained information leakage (e.g. speed and distance of movement) *fingerprinted* with similar accuracy. Furthermore, using this architecture and capturing traffic patterns corresponding to entire warehousing workflows in an industrial setting, this accuracy increases to at least 85%.

The second passive side channel attack explored is the acoustic side

channel. Naturally, objects produce sound when they vibrate [23, 24]. In the case of a robot, vibrations are present while motors are activated and the robot moves. Due to these fundamental acoustic emanations during robot operation, the aim is to determine whether movements and workflows can be fingerprinted, as with the other passive side channels explored in this thesis, solely from sound. Using a smartphone to advocate for a passive insider attacker, robot movements and warehousing workflows were recorded and acoustic characteristics were extracted into a feature set for fingerprinting. Upon evaluation, robot movements were fingerprinted with at least 75% accuracy as a baseline and entire warehousing workflows could be reconstructed with 64% accuracy. It is clear that a passive insider adversary has the potential not only to reveal what a robot is doing but take the resulting liabilities of such an attack to an extreme that impacts even the organisations that employ them. As well as this, in certain robotics environments, such as in surgical settings, procedures may be streamed and/or recorded for viewing, education or research [25–27]. Therefore, it is important to question how VoIP impacts the audio samples for movements and workflows and, ultimately, the success of the attack. Using the Opus codec – a common choice for most modern VoIP applications – fingerprinting accuracy was 90% for baseline speed and distance, which is nearly 15% more accurate than the baseline without the Opus codec employed. This presents new research questions regarding side channel attacks via VoIP communication networks which target robotic systems.

The third and final side channel explored in the first part of this thesis is radio frequency. Many robotic systems make use of stepper motors and microprocessors which have been shown to generate un-

intentional radio frequencies [28]. Using a novel fingerprinting attack strategy which converts signals, corresponding to robot movements and warehousing workflows, into a feature set that can be fingerprinted efficiently, movement fingerprints can be classified with at least 96% accuracy. This increases to near perfect accuracy when reconstructing entire warehousing workflows.

Aside from attacks in the physical and cyber domains, there exists an ecosystem which, at the heart of robotic systems, underpins operational accuracy and safety – calibration. All devices are calibrated to ensure that they operate and/or measure at the highest possible level of accuracy and precision, with the lowest margins of error (measurement uncertainty). To ensure this, devices that are calibrated must have measurements that can be verifiably *traced* to national standards (SI units) [29,30]. While existing calibration processes are coping well, the ubiquitous nature and scale of IoT leaves less time before there will be a need for an immediate move to a digitised ecosystem to allow for smarter behaviour, increased efficiency and automation, among other ideal properties. Unfortunately, while discussions and some progress has been made with regard to digitisation [31, 32], little attention has been paid to the security of the ecosystem in digital environments. Upon review of the calibration ecosystem, there are several inadequacies and concerns that need to be addressed in order to scale in digital environments. First, the calibration process is entirely manual and paper-based, meaning any related processes are time-consuming and inefficient. Second, there is no standardisation for calibration records across vendors meaning there is an inherent lack of collaboration which is required. Third, calibration records are usually held in centralised storage at the organisation where the respective devices are calibrated

and must be requested requiring access rights and potentially financial incursion. Fourth, because of this lack of collaboration and centralised storage, there is no *complete*, auditable record of calibration (e.g. who performed and recorded the calibration of a device and when this was carried out). Finally, there are varying security requirements of integrity and confidentiality among interacting parties, some of whom may additionally share an adversarial relationship (in conflict).

Upon discussion with key stakeholders in the domain of calibration, a new solution to help this progression to a digitised ecosystem should have the following properties. First, it should be highly available, distributed and tamper-resistant, to maintain the integrity of calibration records but also mitigate the potential for denial-of-service threats to calibration infrastructure. Second, it should allow for forensics such that parties who performed calibration, for example, can be identified in the event of failures. Third, the new solution should enforce calibration *hygiene* by being *smart* and automated. Fourth, it should be both cross-platform and cross-vendor in an efficient and timely manner, to allow for more fluid collaboration among interacting parties. Finally, the security of calibration (integrity, confidentiality and conflicts of interest) should be enforced. From these properties, a novel solution to this problem is proposed which makes use of the Ethereum public blockchain and is shown to adequately meet these properties and significantly outperform the current state-of-the-art in calibration traceability. While these properties are achieved by the blockchain solution, the enforcement of the calibration security properties – namely integrity, confidentiality and managing conflicts of interest between providers of calibration services – come with a set of information flows that presents an interesting access control challenge.

Specifically, it is important to verify that an individual verifying the calibration of a device, or even calibrating a device, has the appropriate rights to do so. With respect to integrity, it is important to manage who can write calibration records (during calibration of a device). If an attacker manages to write fake records for an illegitimate device, then information flowing in a chain of calibration to/from this device could be leaked. For example, an attacker would be able to observe who is verifying traceability of their devices which would involve this fake record in its traceability chain. With respect to confidentiality, it is important to protect *what-is-being-calibrated* and *how-often-it-is-calibrated* which could be leaked through continuous monitoring of calibration verification checks. As well as this, managing conflicts of interest between vendors of calibration is vital, as sensitive information flowing to competitors is a less than ideal scenario. Ultimately, existing access control models were not enough to manage the unique information flow constraints present with calibration traceability and requires a novel unification of three existing models: BIBA [33] (integrity), BLP [34] (confidentiality) and Chinese Walls [35] (conflicts of interest). Upon evaluation, the proposed unified model significantly outperforms a simple conjunction of existing models and is able to authorise calibration traceability verification checks efficiently. The latter part of this thesis provides the initial insights into the need to secure calibration and sets up an open challenge space pertaining to calibration and the importance of this consideration to the security of robotic systems.

1.1 Contributions

This thesis provides an in-depth look into the security of robotics workflows from two novel standpoints. First, the need for exploration on the impact of passive insider attacks on robotics systems is key. Given that passive attacks aim to uncover information from regular system behaviours and the fact they are hard to detect in nature, it is important to understand what threats can target robotic systems in this domain and ensure preventative measures can be understood and put in place before they occur in the real-world. Second, given that traceable calibration underpins the accuracy, precision and error of measurements from most devices, this also includes robotic systems. Moving to a digitised, connected infrastructure, a simple transition is not as straight-forward as one would hope, coming with the need for a completely revamped threat landscape. This thesis sets out the security requirements for which traceable calibration must conform to, as well as two solutions pertaining to secure record keeping and access control as the first steps in achieving this.

In summary, the thesis presents a series of works that address the following research questions:

Research Question 1. Can an attacker leverage and mount passive information leakage attacks which target robotic systems? To what extent (granularity) can an attacker observe information leakages that can lead to the compromise of operational confidentiality of robotic workflows?

Research Question 2. Assuming a robot is secure, there exists another ecosystem which ultimately underpins the accuracy, precision and error of measurements output from most devices (e.g. sensors).

With the need to move to a digitised ecosystem, how does this impact the security of robotic workflows?

1.2 Organisation

This thesis is organised as follows. Background information leading into a reviewed threat landscape for robotics systems is described in Chapter 2, which identifies novel problem spaces in passive side channel attacks, as well as a new attack vector pertaining to calibration. The thesis is then divided into two parts. The first relates to chapters on passive side channels used to mount information leakage attacks targeting operational confidentiality. Specifically, three side channel attacks are considered, namely: traffic analysis, acoustic and radio frequency, which can be found in Chapters 3, 4, and 5 respectively. The traffic analysis chapter looks into mounting a passive information leakage attack in the cyber domain, while the acoustic and radio frequency side channels show that the same attack goal can be achieved in the physical domain. The second part of this thesis focuses on secure calibration pertaining to robotic systems and respective chapters can be found in Chapters 6, 7 and 8. Finally, the thesis concludes in Chapter 9.

2 — Background

In the past, robotic installations and integrations within existing environments was too expensive and considered a higher risk implementation that may not have seen any benefit. In the past, robots were not as commonplace, as integrating them into safety-critical environments such as surgical theatres was expensive and usage benefits did not outweigh potential risk. Fortunately, with advances in engineering and computer science, the integration of robotics systems is now highly adaptable, flexible and scalable to many industries. This now provides a significant change in economic outcomes, whilst enabling higher levels of accuracy, precision and efficiency compared to historical operations (i.e. via humans).

The use of robotic systems has been embedded into a wide array of application areas, including: autonomous and unmanned vehicles such as passenger vehicles and submarine vehicles; medical robots such as those used for surgery; and industrial robots such as those used in warehouses and manufacturing. As these are important in the context of this thesis, each of these application areas are outlined in more detail below.

2.1 Autonomous and Unmanned Vehicles

Research into Connected and Autonomous Vehicles (CAVs) has shown rapid progression, with the market for autonomous vehicles expected to reach a value of $40 billion (USD) by 2025 [36]. While many of these

devices are teleoperated, particularly submarinal and aerial vehicles, a subset of them are fully autonomous with the consumer population in some domains willing to pay premiums for full autonomy. The use of CAVs is fairly widespread, with their usage seen in both public and private transport, surveying and mapping, submarinal and aerial applications.

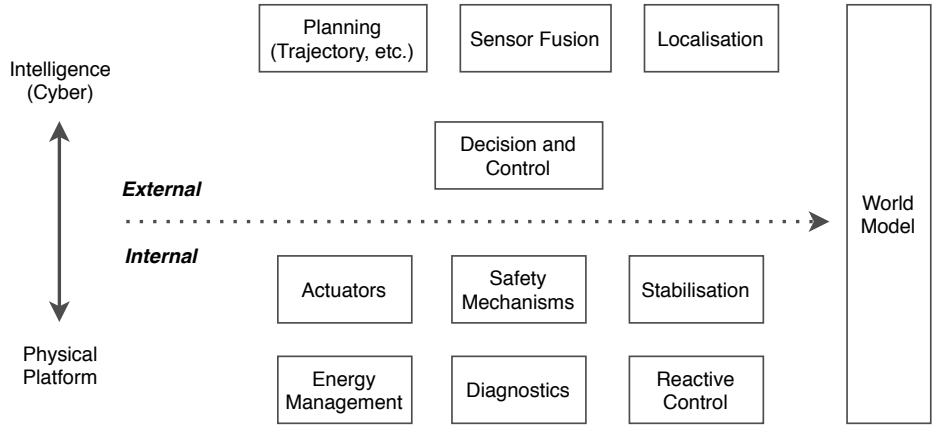


Figure 2.1: Overview of CAV Architecture

Although the architectures of CAVs differ among the classes (autonomous cars, aircraft, UAVs, USVs), the majority of them utilise common components, such as cameras, sensing devices, GPS, etc. which are heavily relied on to aid with actions such as path planning and real-time manoeuvering. An overview of CAV architecture can be viewed in Figure 2.1. Within this architecture, data flows through the various functional components where the outputs of sensing components (external) influence decision and control, linked with the electrical control units (internal), with both internal and external components interacting with the world model [37]. Most sensors in autonomous vehicles are usually fixed, however with added mobility, coordination and cognition that comes with full autonomy, it is common that physical motion and configuration changes (such as to field of view, zoom of lens, etc.) may be needed. Further, calibration

changes to the sensing equipment may be needed at runtime (such as due to changing exposures during the day and re-calibration if, for example, physical damage is suspected). Further, keeping with the overview architecture, these functional components are managed by electrical control units (ECUs) inter-connected by the Controller Area Network (CAN) protocol [38]. Initially designed in the 1980s, it is still the most widely used in-vehicle bus to date, mainly due to cost efficiency [39,40] and its support for multiple subsystem architectures.

2.2 Surgical Robotics

Early surgical robots were typically configured based on pre-planned operations, such as bone-milling robots [41]. Prior to their introduction, there was an innate need to address problems of accuracy and precision in surgical settings, such as optimal sizing and fitting of implants. As a result, these pre-operative planning robotic systems were introduced. These robots typically consist of a planning workstation and the robot itself (i.e. a robot arm) equipped with an interchangeable instrument (end effector) [42]. As a result of these robots being installed, good feasibility was demonstrated coupled with higher success rates and a clear reduction in post-surgical complications.

As technological advances progressed, surgery did too with the introduction of teleoperation. In the surgical domain, these include the Zeus surgical system [43], da Vinci surgical suite [44] and the RAVEN II surgical robot [45]. Commonly, these surgical robots are arm robots that are operated at some distance (wired or wirelessly) via a human surgeon through a surgeon's console (controller). The role of the controller is to translate human movements, via mediums

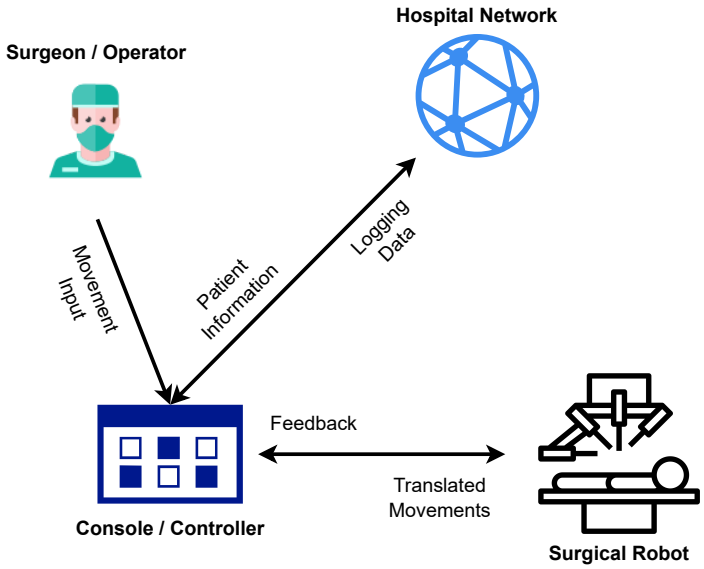


Figure 2.2: Overview of Surgical Robot Architecture

such as finger controllers and foot pedals, into instructions the robot can translate to local space and execute. Furthermore, it may also provide feedback to the surgeon, such as a 3D view of the surgical site and force-feedback through movement mediums. This shared common architecture among these systems is depicted in Figure 2.2. A key safety-critical component of this is the link between the controller and the robot's main electronic control system, where feedback from and inputs to the robot traverse.

2.3 Industrial Robotics

The addition of robotic systems in industrial settings accompany a large portion of robot installations, and range in expertise from collection and packing of products in warehouses [46–48] to the automotive industry [49, 50]. In this sector, as with medical surgery, robotic arms are most prominent – either teleoperated or autonomous. In either case, the system involves the robot paired with a teach pendant (controller) which allows teleoperation or pre-programmed operations to

be executed autonomously. The main electronic control system links together the controller, robot and other inter-connected components, and the network in which the robot operates.

Interestingly, an experimental analysis of industrial robots [51] showed that most industrial robot architectures are extremely similar to surgical robot architectures in terms of the key critical components. Therefore, teleoperated architectures can be generalised in terms of these components, as shown in Figure 2.3. Given this, and the fact that teleoperated robots contribute to a large portion of the robotics market [52], the primary focus of this thesis will be on the security of teleoperated robots and the workflows in which they operate.

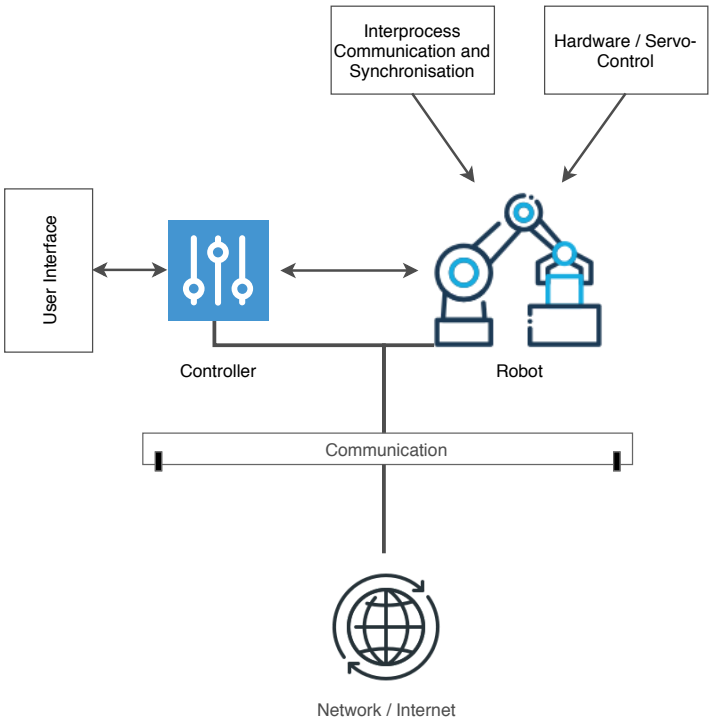


Figure 2.3: Generalised Teleoperated Robot Architecture

2.4 Security Challenges in Robotics

While the ubiquitous nature of safety-critical robotics systems leads to varying applications in which they operate, many share similar critical components and thus provoke common security challenges. Due to the cyber-physical nature of these systems, the security challanges can be classified into both the cyber and physical domain, where attacks or compromise originate from either the Internet or other networks, or from the physical environment respectively.

2.4.1 Cyber Domain Challenges

Attacks in the cyber domain become present when a device is connected (i.e. to the Internet). By observing how information flows through teleoperated robotic systems, several security challenges can be identified.

First, one challenge associated with this information is **integrity compromise**. Control or feedback messages could be tampered with in order to manipulate the system. Bonaci et al. [15] identified that this type of attack can be classified in two key categories: *intention modification* and *intention manipulation*. The first involves impacting the intended actions of the operator, such as modifying messages in-flight after the controller has transmitted them and the operator has no control over this. The second involves the modification of feedback from the robot to the controller, where in this case messages are assumed to be authentic and valid and consequently causes the operator's subsequent actions to end up causing harm. Further, this harm may even be extreme enough to result in liabilities in the physical domain, such as damage to the environment or nearby humans. Notably,

this is also a challenge relating to **authenticity**, which corresponds to a third classification of attack on operators described by Bonaci – *hijacking*. While feedback authenticity is a problem, a hijacking attack can lead to messages from the controller to the robot to be completely ignored and instead perform other actions (e.g. from messages sent by an adversary). The result of this challenge may not be revealed until after any malicious actions, either intentional or unintentional, have been executed. For example, in the context of drones on a battlefield, not verifying the authenticity of messages could allow an (enemy) adversary to not only collect information on the drone's location but perhaps even hijack the drone for their own reconnaisance purposes. Aside from just message authentication, it is also important to give consideration to authentication in relation to access control. McClean et al. [16] describe an assessment of the security of the Robot Operating System (ROS) [53] and discovered a large number of ROS masters open to the public Internet via the default master port. In total, they uncovered 15 instances with a number being legitimate robots capable of being remotely accessed and teleoperated. This demonstrates that it is possible for an adversary to, without any authentication, eavesdrop on data flowing between interacting robot components and even send messages to them which could allow remote operation. McClean et al. [17] also found that it was possible to analyse the traffic flowing through these systems and gather ROS message headers to spoof and replay messages.

As well as integrity and authenticity challenges, those pertaining to confidentiality are also important. McClean et al. [17] found that ROS communication is done in plaintext and no encryption is carried out. This leaves any passive eavesdropping adversary readily able to un-

derstand what information is being sent in the robot system, as well as other systems which communicate with the robot in its network. Furthermore, while single messages may not necessarily leak much information, such as a message to change the position of an actuator, the use of continuous monitoring in this case could be used to string messages together which could lead to the compromise of operational confidentiality. For example, consider a surgical robot operating on a patient. If an adversary was able to uncover individual surgical operations, they could piece them together to reconstruct known surgical procedures. In combination with other meta-data, such as patient admission and exit times, this could lead to the compromise of patient privacy, and ultimately violating HIPAA legislation [54]. While relatively trivial in modern times, it is vital to ensure the use of strong channel security measures to protect the confidentiality of transmitted information.

2.4.2 Physical Domain Challenges

While there are several challenges which compromise various security properties in the cyber domain, it is also important to give consideration to the potential threats and attacks present in the physical domain.

First, one must consider physical compromise of a robotic system. Similar to the integrity of information flowing through the system and networks, maintaining the integrity of the physical components of the system is vital. For this, there are two main possibilities: system trauma (indirect) and direct tampering. Pertaining to trauma, the impact of physical shocks in the environment can disrupt the proper functioning of the device [55, 56]. This may be temporary or cause a

permanent shift in placement, both of which could result in incorrect data capture or movement outputs [57]. Merely shifting devices or small shocks could be enough to disturb the systems physical integrity. In terms of direct tampering, this involves actions such as intentionally swapping out devices (e.g. by malicious technicians or operators), or replacing damaged equipment with newer ones that also have faults (e.g. incorrectly calibrated) [58].

Second, it is possible that an adversary may employ a jamming attack. A jamming attack is one that aims at directly interfering with one or more components to disrupt regular operation. For example, radio interference can be *passively* directed to disrupt wireless networks by increasing signal noise at the receiver [59, 60]. This can result in a denial-of-service (DoS) attack and halt communication entirely. As well as this, a more active adversary could direct a laser at sensing equipment to induce perturbations to input data [61]. This may lead to misclassification of objects in the robot's perceived environment [62].

Third, risks such as collisions or jerky movements can occur at any time during a robot's operation, and it is important to mitigate and prevent this from happening. Some robots may make use of sensing equipment as a form of safety mechanism, such as to halt operation if an obstacle is present, however these can be expensive. While cheaper, off-the-shelf components can be used as a cheaper option, component lifespan and accuracy may not be guaranteed [63]. Thus, physical safety mechanisms such as an emergency stop trigger can be used to provoke an immediate halt response. From an attacker's perspective, there are two possible mechanisms for emergency stops: a physical switch or a software-based mechanism [64]. While a physical switch

may be present, triggering this maliciously is noticeable, but in the case of software-based emergency stop mechanisms, a trigger may be a result of system compromise.

Overall, while there are many potential threats to consider in the physical domain, many are active attacks and mostly pose a threat to the **availability** of robotic systems.

2.4.3 Secure Calibration

While many threats in the cyber and physical domains may impact system safety by targeting the accuracy of the system, there is one key factor which ultimately encapsulates the accuracy and precision of system components – calibration. The calibration of any equipment, such as sensor devices, is carried out to ensure that any outputs (measurements) from devices remain accurate and have low margin of error (measurement uncertainty). Interestingly, even the cables and resistors used within these devices are also calibrated. To put it simply, calibration involves a set of processes that underpin operational safety.

After its manufacturing lifecycle, a component would typically undergo some form of calibration *in-house* and after a specified amount of time (typically a year) it would be recalibrated to reassess its accuracy and uncertainty and adjusted if necessary. Interestingly, a distributed calibration infrastructure can be observed. A root of trust is established at the national level by National Measurement Institutes (NMIs) who maintain the *gold standard*, such as NPL in the UK or NIST in the USA. These organisations then provide calibration for reference devices at some intermediary level organisation, who then perform calibration for manufacturers or system-level organisations.

When a device is calibrated, a traceable chain of these reference calibrations is established to the root-of-trust, so that when the accuracy and uncertainty of device outputs need to be validated, the calibration status can be derived and verified at each step.

The importance of securing the calibration process and calibration status of all robots and their components is vital. For example, Quarta et al. [51] point out that calibration parameters, which are used for determining precise positioning of actuators or white-balancing sensors, are an essential construct. By manipulating these parameters through a means of integrity compromise, they were able to impact the servo motor in a robot resulting in highly erratic movements. While erratic movements in a laboratory setting may not seem disastrous, consider a surgical context. Even an offset of just a few millimeters could result in the difference between life and death. While this was explored in the form of disruption, it could be thought of as a form of denial-of-service attack which is hindering the ability to provide a continually accurate operation. Furthermore, while calibration parameters may be stored at the system-level to allow for quick verification, for example during a secure boot (startup), what would happen if the integrity of calibration was compromised during the traceable chain of calibration at any level up to the root-of-trust? Could the calibration status of robots dependent on these upper levels be trusted? Furthermore, many of these parties – particularly at the intermediary level – may share an adversarial relationship (conflict of interest). Therefore, any potential for the leakage of confidential information between potential competitors should be prevented.

2.4.4 An Overview of Security Challenges in Robotic Workflows

Overall, a number of challenges can be established which arise in relation to the security of robotic workflows, ranging from cyber and physical domain challenges, to further refined challenges with respect to the calibration of interconnected components within robotic systems.

Integrity, Authenticity and Confidentiality of Teleoperated Data Transmission

The first challenge pertains to end-to-end **integrity** of teleoperated data, such as the input captured by sensors or the commands sent between the controller and the robot itself. If the integrity was compromised, the validity of such data would be questionable. Can we trust that the input is sanitised and will not result in the robot causing harm to itself, people or environment? A simple solution one could consider is the use of Message Authentication Codes (MAC) or hash to ensure data integrity [65, 66]. In either case, a key is shared between the communicating parties which is used along with a message into a cryptographic function. The message is sent by one party along with the hash or MAC, which is then re-computed at the receiver end and verified that the result matches the one that was sent with the message. If this matches exactly, the message was not tampered with in-transit. While such techniques can ensure data integrity in-transit, there are two key issues. First, how can the receiver ensure that the data being sent can be trusted (or ensure that the sender itself can be trusted)? Second, while the data is protected from tampering in-transit, how can one trust that the data is not being captured whilst

in-transit? These two challenges pertain to the authenticity of the data and data source, and the confidentiality of the data in-transit, respectively.

With respect to the **authenticity**, a trusted mechanism here is the use of a digital signature scheme (DSS) [67–69]. In a DSS, the sender of a message will have a public and private keypair. The sender will sign a message using their private key and the receiver will verify the message using the sender's public key. This gives the receiver high confidence that the sender is indeed who they say they are, whilst also providing data integrity as tampering the message would invalidate the digital signature. While this covers the authenticity of data transmission, it is entirely possible that a malicious component could be communicating false data, with a legitimate identity, to the rest of the system. In this case, authentication schemes to prove component identity could be achieved through a challenge-response protocol, such as one involving Physical Unclonable Functions (PUFs) [58, 70]. In this case, a challenge issued from a device is inherently unique to only that device. For example sending a signal through a PUF circuit generates a signature that is unique to the device, given the unique timing properties intrinsic to slight manufacturing variations within the circuits. Thus, a list of challenge-response pairs can be maintained such that if the device was to be swapped out for a malicious device, this would be detected on startup and thus communication with that device can be flagged and blocked temporarily.

Third, while the authenticity and integrity of the data and data source can be maintained, it is entirely possible that network traffic can be monitored to allow an adversary to eavesdrop on potentially *confidential* communication. For example, if control commands to be sent to

the robot were captured in-transit, continous monitoring and traffic analysis techniques could reveal entire robotic workflows and potentially compromise the **operational confidentiality** of the organisation of which the robot operates. This information could be used to bribe the company or used by competitors to gain a competitive advantage. While it is relatively trivial to prevent the use of plain text data transmission, in some robotics environments (i.e. industrial settings) there is no mandate or standard for ensuring data confidentiality. Interestingly, in medical contexts the Medical Device Regulation (UK) [71,72] and HIPAA (USA) [54] mandate the use of TLS for legal compliance. Specifically, as of August 2019, they require the use of TLS1.2 with FIPS-based cipher suites, with support for TLS 1.3 by January 2024 [73]. In any case, the use of appropriate channel-security technology such as TLS can help ensure that data is encrypted and communicating parties establish a secure, authenticated connection between one another. One question remains, however. Is it possible, even with the use of channel-security and end-to-end encryption, to still breach operational confidentiality?

Physical Plane Security

While a robot may be secured in the cyber domain, once it is ready to be used how can we ensure it is *physically secure*? It is vital to guarantee that robots can maintain relatively normal operation, perhaps within some tolerance, even while under adversarial threat or attack. If a jamming or signal saturation attack [11] is used to disrupt communications of a robotic system, the impact of these attacks must be mitigated as best as possible to ensure it can continue operating in its usual fashion with minimal or no disruption. On top of this, all inter-

connected components of the system should be confirmed as authentic and not physically tampered with. Further, components installed at the time of manufacturing, or legitimately installed otherwise, should be recorded securely such that events resulting in liability (e.g. injury to operators) can involve an audit of the robotic system. This will not only prevent the potential for faulty components to be used, but also prevent adversaries from modifying and/or installing components to act maliciously. As well as this, another difficult challenge pertains to the locality of many teleoperated robots. Given that they may be in a remote location, such as teleoperation being conducted from a different room or facility, secure maintenance may be difficult to perform – particularly on a live robot. A part of maintenance is also establishing that there may be a problem that warrants maintenance during operation. Therefore, is it possible to monitor the robot in the physical domain for abnormalities or erroneous behaviour? This is explored in Chapters 4 and 5.

Security of Robot Calibration

While there exist many security challenges in both the physical and cyber domains that threaten the accuracy and error tolerances of robot systems, even if the system is theoretically secure calibration ultimately encompasses these properties. The question here is, how can aspects of the calibration ecosystem that pertain to robotic systems be secured before, during and after the robot is manufactured and installed. After a component is manufactured, it should be legitimately calibrated to guarantee it operates accurately within a tolerated output range. For example, a temperature sensor may be calibrated to record temperatures between 1–100°C, with an error tolerance of

±1°C. As well as this, a calibration report is produced detailing this information, as well as information about the organisation(s) and reference (parent) device(s) involved in the calibration process. After some period of time, typically annually, the component is calibrated to make sure it conforms to the tolerances specified in the report(s) from its previous calibration. Ultimately, the requirement is to securely maintain these records and ensure that the calibration of components is kept up-to-date and performed correctly and legitimately recorded in accordance with professional standards. In a real-world scenario, assuming a sensor was out-of-calibration, the robot may perform less efficiently or incorrectly classify sensed objects; and as such the question of whether the robot should still be in operation is highly questionable.

Summary

Previous events in the real-world involving robotic systems have shown the extent to which threats to robotic systems, for example misconfigurations, have resulted in human deaths and only deemed to be a result of malfunctions [10,21]. However, are these truly malfunctions? Could these liabilities have risen from improper calibration of one or more components in the robotic system? In a connected environment, could these be the result of an attacker? Ultimately, it is clear that there are many challenges in all domains pertaining to robotic systems, with many defences and countermeasures proposed to address them.

Interestingly, however, little attention has been paid to the capabilities of a passive adversary in both the cyber and physical domains. While end-to-end encryption and channel security is introduced, pre-

vious research has shown it is possible to have success in inferring data from encrypted workflows, such as classifying types of Internet traffic [74–76] and fingerprinting IoT devices [77–79]. Thus, we question whether the legislative mandates of simplying using TLS or other channel security technology is enough to protect the operational confidentiality of organisations that employ robots. Specifically, can an adversary use traffic analysis to infer information about robotic workflows and reveal information about an organisation and the operating environment? With regard to the physical domain, again most attacks are active and involve a form of physical compromise. However, can an adversary also gain information about robotic workflows as a cyber-approach could do, but in the physical environment? It has been shown previously that physical side channels, such as electro-magnetic fields, power and acoustics, can be captured and analysed to gather information about device operation. Thus, is it possible to compromise operational confidentiality as a passive attacker, by exploiting side-channel information leakage in the physical domain? Finally, while there exists a calibration ecosystem, a progressive shift to a digitised ecosystem and the increasing installation of safety-critical robotic systems present several key challenges to be addressed, particularly with respect to calibration verification.

3 — Passive Reconnaissance of Robotic Workflows via Encrypted Traffic Analysis

The Internet-connected nature of modern robotic systems further expands the threat landscape, exposing them to new threats in the cyber domain. Previous research has identified several of these challenges, such as active targeted attacks tampering with control commands and other transmitted data [64]. However, there has been little focus on reconnaissance aspects, such as eavesdropping and fingerprinting. Even though there is less focus on the capabilities of a passive adversary, it is still highly important to investigate. If an adversary was to passively monitor communications between a robot and its controller, it is entirely possible that they could reconstruct operational workflows. These workflows could correspond with highly confidential operations such as warehousing workflows in industrial settings, or surgical procedures in medical settings, which could lead to a breach of operational or patient confidentiality respectively. Simply put, the liabilities that could arise as a result of passive reconnaissance techniques have the potential to wreak havoc on organisations. In this case, it is a relatively trivial assumption that organisations would employ strict security requirements that are to be conformed to, in order to mantain confidentiality. For example, legislation such as HIPAA (USA) [54,73]

and the Medical Device Regulation (UK) [72] mandate the use of TLS (1.2) for connected medical devices, which includes robotic systems, to protect private patient information that may flow between them. However, there is an important question that needs to be addressed. Prior art has shown that it is possible to perform identification and classification of IoT devices [77, 80, 81], websites and users [82, 83], by simply performing passive reconnaissance to observe and analyse encrypted traffic. From the main research questions presented in Chapter 1.1, this chapter aims to address research question 1 and whether it is possible for an adversary to infer what operations a robot is currently carrying out, solely using the network traffic between the robot and its controller, even if the communication is protected through the use of TLS.

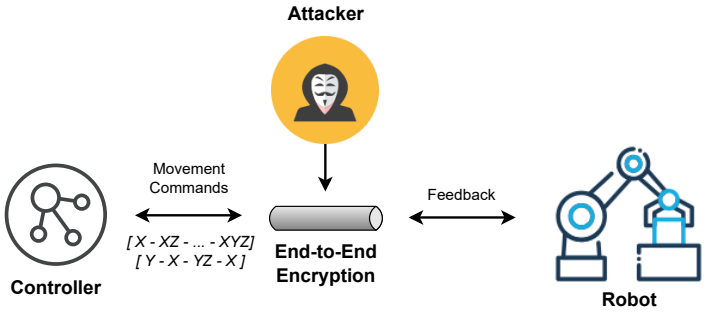


Figure 3.1: Robot and Traffic Analysis Setup

3.1 Threat Model

In this context, the adversary is a passive, stealthy attacker who is able to eavesdrop on the communication channel between the key component of modern teleoperated robotic systems, the controller, and the robot itself (Figure 3.1). First, consider an insider adversary such as the technical staff who can access the internal organisational or robot network. By capturing the traffic between the robot and the

controller, and combining this with other data sources, information leakages about the operational environment exposes another degree of detail. By identifying potentially confidential workflows, such as warehousing operations, this information could be sold on to competitors or be used as bribery in an attempt to discredit the operator or organisation. Second, it is entirely possible that a passive outsider adversary could gain access to the network through some attack vector (e.g. application exploits). Competing organisations could use leaked operational workflows to compare performance and improve their own operations, or uncover problems that could be used to discredit the company (i.e. to cause an audit to fail if an organisation was found to be *cooking-the-books*).

3.2 System Design

The focus of this chapter is to investigate whether an adversary can mount an information leakage attack using the traffic analysis side channel to fingerprint robot movements, by monitoring encrypted traffic flows between a robot and controller in a teleoperated architecture. An overview of the setup for this attack can be seen in Figure 3.1.

In this work, the robot used is uFactory's uARM Swift Pro [84] which is operated by an Arduino Mega 2560 running MicroPython. The robot is connected to a controller on a Windows 10 laptop via the uARM Python (3.8.X) SDK [85]. To mimick a teleoperated communications architecture, a client-server communication topology is followed, in which an asynchronous TCP/IP socket is established for communication between the robot and controller. The message structure is adapted from the work presented by Zeng et al. [86] and is

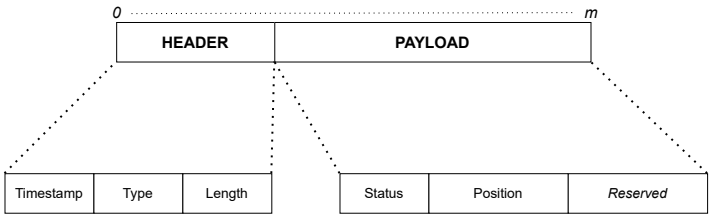


Figure 3.2: Robot Message Structure

depicted in Figure 3.2. First, in the header, the timestamp is an unsigned 32-bit integer representing the time at which the message was generated before transmission. The type field is a 2-byte ASCII code representing the type of the message being sent. For example, this could represent whether it is a feedback or control message. The length field is a two-byte representation of the length of the message, excluding the header. Second, in the payload, the status field is a byte field in which system status codes are present. The position is an n-byte field where each byte represents the position of each of the n axes to be positioned. Finally, the reserved field is a set of k bytes which are held for extensions to the message. For example, sensor controls or medical images (in the case of surgical robots) may use this field. In the work presented by Zeng et al. [86], other fields such as session management data is also a part of the message, however, this is handled by TLS session establishment in this work.

To enable TLS-encrypted communications between the robot and controller, the traffic is routed through a software-defined network (SDN) using Mininet 2.3.0, running a TLSv1.2 client-server network to allow for realistic simulation and evaluation of various network conditions. This representation of teleoperated robot communications can be generalised due to similarities shared between them [51]. The key generalisation in packet structure refers to the movement data, where additional input (i.e. system information and sensor data) do not

directly contribute to this attack in terms of movement inference in non-autonomous systems, beside any influence on human operators to make decisions based on this additional input. Finally, Wireshark [87] was used to capture the encrypted traffic flows between the robot and controller, imitating a passive adversary. Wireshark was set up as another host in the network and eavesdrops on the communication link between the robot and controller (Figure 3.1).

3.2.1 Emulation of Network Link Characteristics

For measuring the impacts of link characteristics such as packet loss and delay/jitter, a suitable network emulation environment is required. Several options were considered, including: PlanetLab [88], DETER [89, 90], Emulab [91], NS3 [92] and Mininet [93, 94]. Mininet was chosen as it is best suited for modelling arrivals as a Poisson process – a similar behaviour as seen with teleoperated robots [95]. The justification for not using other approaches is as follows. PlanetLab [88] is a global testbed for network systems research with nodes spread across the earth. The main challenge associated with this is that results are not reproducible as network conditions can vary over time [96]. Other real-world emulation testbeds such as DETER [89,90] and Emulab [91] face similar challenges, where resources are shared among many researchers which can skew results. NS3 [92] on the other-hand is designed as a discrete-event network simulator and thus, is unsuitable for this work as the network layer will be simulated even if the robot-edge runs live. While Mininet is not without limitations – primarily scalability (due to single threaded components in the core) – in this case with a small network containing a few nodes and switches, these limitations do not manifest. Ultimately, it would be interesting

to observe how other arrival models for delay and loss would impact movement classification [97, 98], and is a point for future work.

3.3 Challenges in Applying Traffic Analysis Approaches

While there exist a number of context-dependent challenges when applying traffic analysis approaches, several challenges can be identified with respect to the compromise of operational confidentiality in robotic workflows:

(C_1) Can individual robot movements on each axis be fingerprinted?

(C_2) Can permutations of robot movements be fingerprinted?

(C_3) How is the identification of movements affected by the distance and speed in which the robot moves?

(C_4) How do realistic network conditions, such as network link delay or packet loss, affect the identification of robot movements?

(C_5) Is it possible to reconstruct operational workflows that correspond to a set pattern of movements?

In order to understand the challenges and how to approach a solution, it is important to conduct a preliminary analysis of the problem setting. To recap, the focus of this attack is on industrial teleoperated robots. In a typical warehouse that employs teleoperated robots, the most common type is a single-arm robot with at least 3 degrees of freedom (axes). Given that the majority of single-arm robots make use of the same principal components as the one used in evaluating

this attack, it is a suitable replication candidate for movement fingerprinting. At the core, the key data for movement inference is the transmission of control commands, which would affect timing patterns of operations and the payload in control messages, for example. The data from other components, such as the input from sensing devices, would be indistinguishable from those provided by actuators and motors which are operated via control commands.

3.3.1 Robot Traffic Generation

The first step in the preliminary analysis stage is generating appropriate traffic traces. Here, an emulation-based approach is used to generate various traffic behaviours to address each of the challenges. For $C1$, the uARM robot was programmed to carry out movement operations along the X, Y and Z axes. $C2$ involves generated traffic traces of permutations of the movements along these axes (i.e. X and Y simultaneously). For $C3$, movements were programmed with varying distances and speeds of movement for more fine grained inference. Specifically, distances range from 1–50mm and speeds range from 12.5–100mm/s. For $C4$, the aim is to determine whether this attack performs well under realistic network conditions. Here, the controller was programmed to send control commands to the robot under various network link delays, ranging from 10ms–1s, and undesirable packet losses of 10–50%. Finally, for $C5$, the robot was programmed to carry out various operational workflows. The context of this work surrounds warehousing workflows in which industrial arm robots operate, and thus, the robot was specifically programmed to carry out four common warehousing operations, including: push, pull, pick-and-place and packing. The dataset used for evaluating

this attack contains around 150,000 samples, with traffic features in the packets collected and represented in a tabular-like dataset. One sample corresponds to a row in the dataset, where each column corresponds to the data contained within each feature of the collected packet.

3.3.2 Analysing Traffic Features

The first step in the analysis stage was to determine how time-frequency representations of traffic features may offer clues as to what movement or operation the robot is carrying out. In Figure 3.3, it can be seen that the robot traffic, among all movements, produces an average of 2 packets transmitted per second between the robot and controller. The second interval between the robot and controller corresponds to the programmed movement interval in order to effectively capture individual movement traffic. While there is an observable increase in packets transmitted at the start of each operation, this is due to the initialisation of the communication between the robot and controller (i.e. TLS session initialisation) and thus discounted for further analysis. Ultimately, it is clear that time-frequency representations do not provide any useful information outside of the initialisation stage of each data collection step.

The next logical step after this is to observe variations among movements using information contained within the packets themselves. Among all available features in the traffic, the most prominent were the following: *Packet Time*, *Frame Length*, *Frame Capture Length* (frame length stored in capture file), *IP Length*, *TCP Length*, *Bytes in Flight*, *Push Bytes Sent* (bytes sent since last PSH flag), *ACK Round-Trip-Time (RTT)*, and the *TLS Record Length*. A box-plot analysis

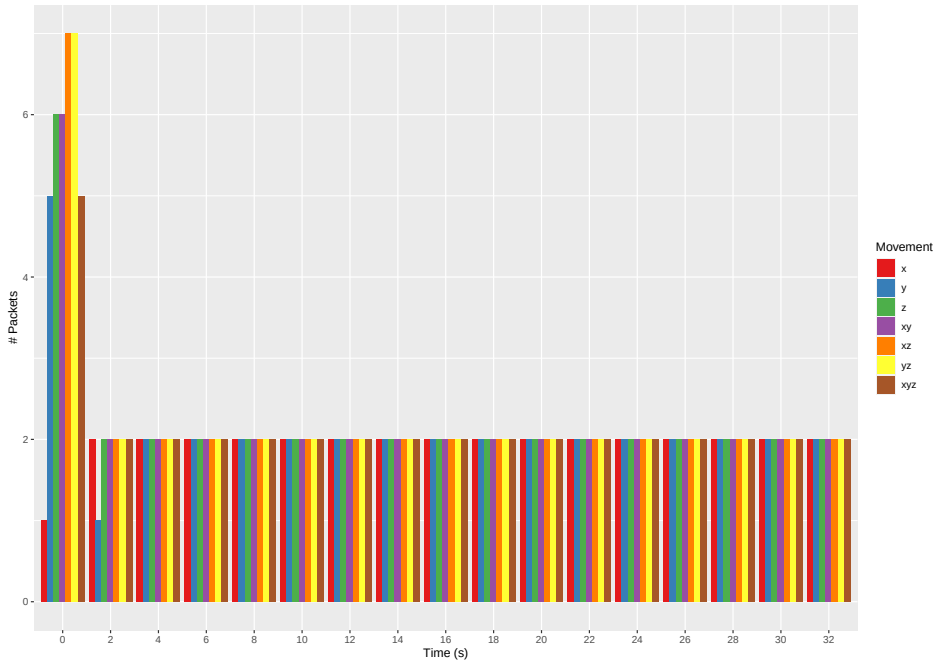


Figure 3.3: Packets per Second for Robot Movements
The flow of packets per second across movements show that time-series analysis alone does not provide enough basis for fingerprinting movements

of the variation among these features can be seen in Figure 3.4, where the x and y axes represent the movement and size of the traffic feature, respectively. Initially, apart from the *ACK RTT* and *Packet Time*, a similar pattern can be observed, with the XYZ movement showing a clearly identifiable variation in comparison. Both the X and Z movements appear to be more similar, also seen with the XY and YZ movements, however the median values in each case are at the opposite ends of the inter-quartile range – the same for both sets of movements, respectively. Interestingly, the *ACK RTT* and *Packet Time* are relatively similar with little difference in the case of outlying values. Ultimately, upon observation, it is clear that simple approaches such as *eye-balling* the dataset or *basic frequency analysis* are not enough to answer the stated challenges.

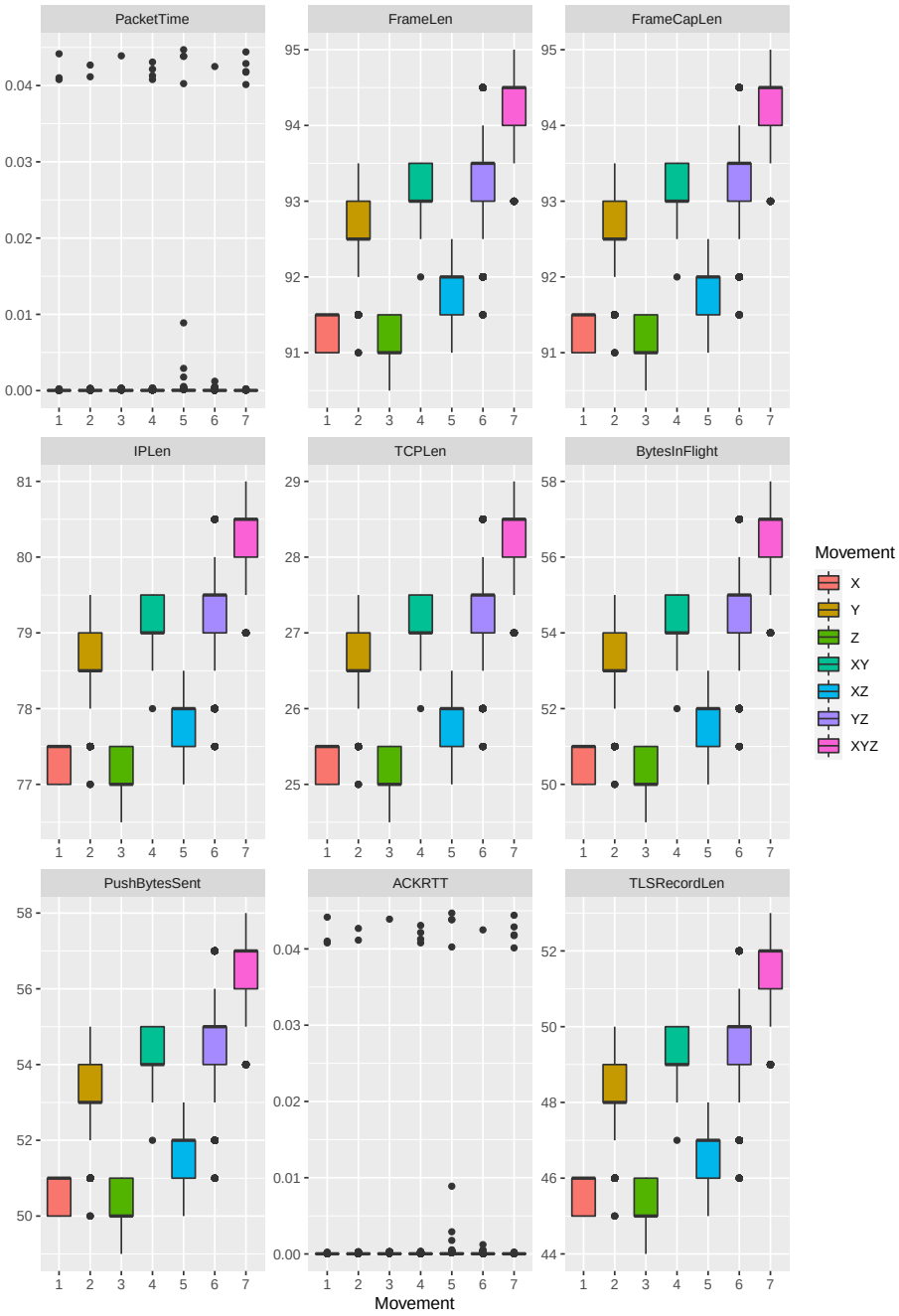


Figure 3.4: Traffic Features Across Movements
A closer look shows there are more subtle variations across movements aside from only time variations

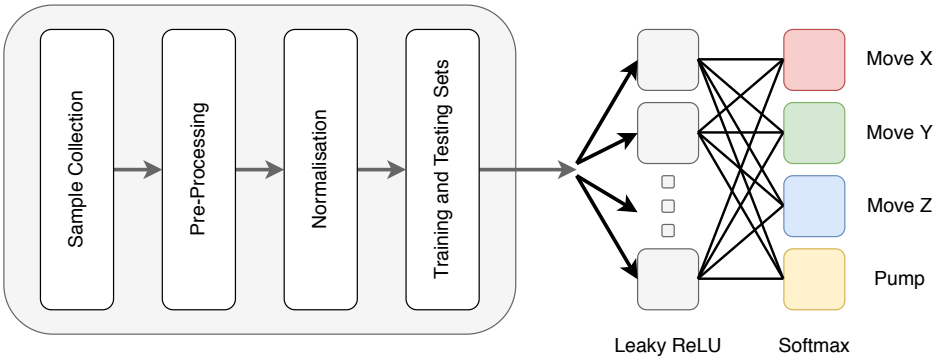


Figure 3.5: Traffic Analysis Approach

3.3.3 Attack Methodology

For this attack, given the inadequacy of simpler approaches, the next step is to take a machine-learning approach to recognise traffic feature patterns for fingerprinting robot movements and, ultimately, reconstruct workflows. The effectiveness of applying machine learning techniques on encrypted traffic has shown success in various other applications, such as VoIP [99–101], mobile [102] and IoT [77, 81]. The reconstruction approach involves several stages (Figure 3.5).

Dataset Pre-Processing

The first step after sample collection was to pre-process the dataset to allow for a machine-learning approach. The goal of this step is to reduce *noise* and *redundancy* in the dataset by removing fields with constant values that have no impact on classification output, such as TCP flags that had no variance among all samples, as well as handling any null values. After this, the dataset was normalised using the *scikit-learn* [103] MinMaxScaler, which normalises values in each feature in the training set to a real number in the range $(0, 1)$. Furthermore, given that the range of distance of Y and YZ movements is larger than the other axes, there is a larger sample size. To even the

sample set among all movement classes, the dataset was also stratified and weighted. Finally, the dataset was split *randomly* into training, validation and testing sets, with 60% of samples used for training and the remaining 40% split evenly among testing and validation.

Corrective Statement. During the viva, it was identified that the normalisation in this chapter was done incorrectly. Specifically, the normalisation should have been done after splitting the dataset into training, testing and validation groups. This is important as the normalisation is on the full dataset which can leak information about the test or validation sets into the training set which is meant to represent real-world data.

Neural Network Architecture

After pre-processing the samples into training, testing and validation sets, the next stage was to construct the neural network to classify (fingerprint) movements and, ultimately, answer the aforementioned challenges. To create the neural network, a sequential model was used which groups layers of neurons together in a linear (feed-forward) fashion. This was achieved using the Keras [104] API library. The model used for this attack is a shallow neural network with three layers. The choice of the parameters of the neural network architecture (e.g. number of layers, neurons, activation functions) were selected using a 3-fold cross-validated grid search [105, 106]. The first layer, the input layer, consisted of 16 neurons corresponding to each of the 16 traffic features in the dataset. The second layer was a hidden, *Dense* layer with 108 neurons and makes use of the Rectified Linear Unit (ReLU) activation function [107]. A dense layer is a deeply connected layer wherein each neuron is connected to each of the input

and output neurons (to the next layer), and the output is a result of matrix multiplication with an activation function and weights matrix. The choice for the number of neurons for this layer is calculated using the following formula [108], to keep it below some value N to prevent over-fitting in most cases:

$$N_h = \frac{N_s}{(\alpha * (N_i + N_o))}$$

where:

N_i = Number of input neurons

N_o = Number of output neurons

N_s = Number of samples in training dataset

α = Arbitrary scaling factor

The alpha (α) value is the effective branching factor (number of non-zero) weights for each neuron, which was given a value of 2. The value of N_s is 4968, the number of samples in the training set for the baseline samples, such that an effective comparison can be made against the other experimental parameters (i.e. movement distance). As a result, the optimal number of neurons, N_h is 108. While this formula provides an estimated optimal number of neurons, the cross-validated grid search explained earlier found this was most optimal. Finally, the output layer uses the SoftMax activation function [109] to have the output in the range of $[0, 1]$ for use as predicted probabilities, with categorical cross-entropy [110] as the loss function. The optimiser used is Adam [111] with a learning rate of 0.00001. This learning rate was chosen as others, such as those with higher learning rates, resulted in lowered accuracy scores. Given the use of SoftMax and categorical cross-entropy, the output layer has 7 neurons, with each neuron corresponding to one of the 7 movement classes.

Choice of Activation and Optimisation Functions

The ReLU activation function was chosen over other activation functions, as the reduced likelihood of vanishing gradient allows for a constant gradient resulting in faster learning. Further, the sparsity of representations are shown to be more beneficial than dense representations, as seen in other activations such as sigmoids [112–114]. The softmax activation function, combined with categorical cross-entropy [110] for the loss function, was chosen due to the fact that this is a multi-class classification problem. Simply, a sample can belong to one of the 7 classes, with each class corresponding to one of the robot movements. For optimisation, Adam was an ideal candidate. It is an extension to the Stochastic Gradient Descent (SGD) method, based on adaptive estimation of first- and second-order moments [111]. Specifically, it allows for the updating of network weights iteratively based on the training data, and fits best with the weighted sample sets in opposition to other tried methods such as standard SGD, RMSProp and SGD + Nesterov Momentum.

3.4 Attack Evaluation

3.4.1 Baseline

In accordance with the challenges set out in Section 3.3, the first two (C_1 and C_2) aim to determine whether it is possible to infer individual robot movements on each of the axes, as well as permutations of these movements. Each movement corresponds to the robot arm moving from some reference point to a destination in a specified direction and speed of movement. The first set of experiments are hereafter

Movement	Precision	Recall
X	70%	85%
Y	69%	54%
Z	80%	63%
XY	21%	60%
XZ	68%	92%
YZ	81%	31%
XYZ	72%	97%

Table 3.1: Baseline Classification Results
The baseline samples contain only samples with a single distance unit and lowest movement speed, with no other varying parameters

referred to as the *baseline*, to enable comparisons against other parameters, such as evaluating how the distance of movement impacts fingerprinting accuracy. In the baseline, the robot is programmed to move with minimum distance (1mm) and speed (12mm/s), with no network parameters in effect.

The results of running the neural network on the baseline samples can be viewed in Table 3.1. The precision metric is the ratio of correctly predicted positive movements to the total predicted positive movements. In the case of the baseline, relatively good precision for most classes can be observed, averaging around 65%, with the exception of the XY movement. Interestingly, the confusion matrix (Appendix A.1) shows that some movements are being incorrectly classed as XY or YZ movements, with Y-based movements showing the lowest accuracy compared to others. Notably, the Z and YZ movements show the highest precision of around 80%, yet the recall of the Y-based movements are very poor in comparison to outstanding recall for the X, XZ and XYZ movements. From Figure 3.4, it is clear that this is likely due to the similarities in traffic features across these movements. In the context of this attack, good or perfect recall is desirable but not the most important if certain movements may be missed. The recall

across most movements is good and thus, the combination of certain movements – even with some missing – can still lead to the identification of entire workflows through the likes of continous monitoring and regular pattern matching.

3.4.2 Impact of Experimental Parameters on Movement Classification

In the next set of experiments, the goal is to determine the impact of robot parameters (distance and speed – C_3) and network parameters (jitter and packet loss – C_4) on the classification accuracy. In real-world networks the channel between the controller and robot may experience jitter and packet loss. Jitter and packet loss in real-world networks are known to follow a Poisson distribution [115–119], particularly in the case of single-link communication (such as the case of teleoperated robots) whose interaction can be abstracted to the likes of an M/M/1 queue for example [120, 121]. To realistically emulate link characteristics, varying link parameters in the SDN were used for delay and packet loss between the robot and controller. Notably, Mininet follows a Poisson process for packet arrival in a single link system, which meets the expectations for real-world network emulation. Further, in the Mininet SDN, a link speed of 100Mb/s is used where it has been previously shown that emulating link properties of delay and packet loss at this rate can be done realistically [122].

Movement Distance

The first parameter evaluated was the distance the robot arm moved in a particular direction. The results of this experiment can be seen in Table 3.2. At 2 distance units (mm), a decrease in precision among

D = Distance (mm), P = Precision, R = Recall										
	D=2		D=5		D=10		D=25		D=50	
	P	R	P	R	P	R	P	R	P	R
X	60%	59%	62%	84%	38%	63%	36%	57%	52%	38%
Y	76%	49%	69%	45%	75%	45%	54%	37%	67%	8%
Z	72%	78%	74%	100%	30%	62%	31%	82%	41%	86%
XY	28%	58%	38%	74%	68%	94%	67%	14%	26%	55%
XZ	43%	74%	33%	35%	0%	0%	0%	0%	38%	65%
YZ	67%	48%	80%	47%	72%	28%	96%	71%	84%	62%
XYZ	78%	88%	61%	97%	36%	94%	54%	100%	53%	77%

Table 3.2: Classification Results With Distance Parameter
Examining movement distance results in more fine-grained movement inference,with classification accuracy increasing with movement distance in most cases compared to the baseline

the X, Z, XZ and YZ movements can be observed compared to the baseline (Table 3.1). The X and Z movements and XZ samples seem to be incorrectly classified as each other, with most of the XZ samples being predicted actually being Y samples. Interestingly, the low precision of XY is due to incorrect classifications of either Y or YZ, perhaps due to the similarities in packet features between them, which is also present in the baseline evaluation. In most cases, there is an increase in recall. For the baseline, Y-involved movements had lower recall in comparison with other movements, but with this first increase in distance there is an increase in their precision. At 5 distance units, the results are fairly similar, with the XY movement precision increasing but with some incorrect predictions of Y and YZ movements as seen with 2 distance units. Notably, the Z movement has perfect recall. With the XZ movement, the precision and recall decrease further again, with most samples here being incorrectly as X or Z. At 10 distance units, there are decreases in precision and recall across all movements aside from the Y and XY movements. At 25 units, results are similar to those seen at 10 units, with the exception of YZ which has a precision of 96% and XYZ movements having perfect recall. Finally, at 50 distance units, the XZ movement is classified

S = Speed (mm/s), P = Precision, R = Recall								
	S=25		S=50		S=75		S=100	
	P	R	P	R	P	R	P	R
X	66%	89%	100%	100%	67%	88%	66%	85%
Y	62%	46%	61%	28%	65%	34%	56%	45%
Z	82%	60%	90%	98%	80%	62%	75%	65%
XY	24%	70%	26%	76%	20%	46%	23%	41%
XZ	61%	86%	37%	83%	34%	70%	42%	68%
YZ	79%	28%	69%	24%	74%	45%	66%	39%
XYZ	68%	95%	66%	97%	86%	97%	73%	97%

Table 3.3: Classification Results With Speed Parameter
Examining the speed of movement provides an attacker with a more fine-grained movement fingerprintwith classification accuracy showing similar increases across some iterations as with distance

correctly but not successfully, as seen at lower distance units. Overall, not only does the distance of movement provide more granularity to movement fingerprinting, but does also influence an adversary's ability to fingerprint them. While an increase in distance does reduce the accuracy of classifying some movements, changes in the payload (i.e. larger integer for distance in plaintext) and round trip time better showcase trends on the Y-axis as distance increases. Further, among some movement classes, increasing the distance parameter does correlate with incorrect predictions among similar classes (i.e. XZ with both the X and Z movements).

Movement Speed

For the next experiment, the aim is to determine how the speed of the movement affects the classification accuracy, with the results shown in Table 3.3. The justification for the range of S is primarily due to the scale and limitation of the robot used in this study. In terms of generalisability, with regard to scale, the packet features would remain the same however the speed field in the raw data would simply contain a larger number in accordance with the packet structure used (see 3.2). At $S = 25$, there is a slight decrease in precision for most movement

classes, compared to the baseline (Table 3.1), aside from the Z and XY movements which show a slight increase. Similar to the distance parameter, most XY predictions are actually Y and YZ movement samples, which may be due to close similarities in traffic features. However, this is lower than the baseline leading to the slight increase in precision. Similarly, there is an observable decrease in recall in the same cases, aside from the X and XY movements. At $S = 50$, there are further decreases in precision and recall for the Y, XZ, YZ and XYZ movements. Notably, the X movement has perfect precision and recall and the Z and XY movements show both improved precision and recall compared to lower speed iterations. At $S = 75$, the precision and recall for the X movement drop, compared to the movements at $S = 25$. Similarly, among the Y, YZ and XYZ movements, there is an increase in precision with others decreasing slightly compared to lower speeds. The recall in most cases show a decrease, aside from the YZ and XYZ movements which have the highest recall respectively among all speeds. Finally, at $S = 100$, the results show to be similar to the $S = 50$ results but with lowered recall. Overall, in this experiment, it is clear that movement speed improves movement fingerprinting compared to the baseline, specifically at $S = 50$. As well as this, slightly better results for fingerprinting movements are found with the speed parameter compared to distance.

Network Link Delay

Aside from features inherent to the robot itself, it is also important to investigate the impact of network characteristics (C_4), due to the nature of network traffic in the real world. The first of two parameters in this context is network link delay. In Mininet, this parameter

L = Link Delay, P = Precision, R = Recall								
	L=10ms		L=50ms		L=100ms		L=1s	
	P	R	P	R	P	R	P	R
X	64%	89%	99%	100%	100%	100%	98%	100%
Y	100%	100%	100%	99%	100%	100%	100%	99%
Z	80%	63%	100%	100%	100%	100%	100%	100%
XY	99%	90%	100%	100%	100%	100%	100%	100%
XZ	89%	83%	76%	100%	100%	100%	100%	100%
YZ	100%	99%	100%	85%	100%	100%	100%	100%
XYZ	100%	99%	89%	100%	100%	100%	100%	100%

Table 3.4: Classification Results With Network Link Delay
There is a significant improvement when a low link delay is introduced, with the precision, recall and accuracy reaching perfect as the delay increases

corresponds to the packet delay time over the link – in this case between the robot and controller. While it is possible to emulate a series of random delays throughout movement transmissions (as delays may typically be unpredictable in terms of magnitude in a realistic setting), the goal is to observe the impact of delays over a range of values that may be considered reasonable for continued robot operation. The results for this experiment are found in Table 3.4. In comparison with the baseline results shown in Table 3.1, there is a significant increase in both precision and recall in all cases, with the majority of movements having perfect precision and recall. The X movement initially has the poorest precision and Z with the poorest recall. In this case, a proportion of the X movements are incorrectly classified as Z and vice-versa – a similar trend seen in previous experiments. However, overall, as the delay increases there are significant improvements to accuracy. In this set of experiments, it is clear that introducing a low link delay significantly improves the precision and recall of all movements. As the delay increases over the robot-controller link, the results show that an adversary can infer robot movements with some degree of link delay almost perfectly where there are *acceptable* delays, and even better with larger delays. Notably, this significant increase

L = Packet Loss, P = Precision, R = Recall						
	L=10%		L=25%		L=50%	
	P	R	P	R	P	R
X	86%	100%	100%	100%	92%	100%
Y	100%	100%	100%	88%	100%	100%
Z	100%	100%	100%	100%	84%	100%
XY	100%	100%	100%	100%	100%	90%
XZ	91%	83%	73%	100%	89%	83%
YZ	99%	97%	100%	100%	100%	97%
XYZ	100%	99%	100%	100%	100%	100%

Table 3.5: Classification Results With Network Packet Loss
Near perfect accuracy is observed in most cases, with the precision and recall increasing as the loss increases

may be due to differences in round-trip time and packet inter-arrival times for each movement, increasing the variation among them collectively, unlike distance and speed which seem to only affect the payload of the collected traffic.

Network Packet Loss

The second of the network experiments looks at the effect of network packet loss on movement fingerprinting. Realistically, failures or inefficiencies of network components that carry the data, such as a faulty router or weak wireless signal, can cause lost or dropped packets and should be accounted for. In a TLS connection, *TCP flow control* detects packet losses and attempts to retransmit these messages for reliable communication. This results in decreased throughput which may have influence on the time-series data gathered from this attack. In a realistic scenario, the question surrounding acceptable packet losses are important. In many applications, quality of service (QoS) considerations are given based on the type of data sent. For a safety-critical IoT system, such as industrial robots, even the loss of a small amount of packets could result in delays that could result in serious harm. It has been noted that losses between 5% and 10% of the total packet

stream seriously impacts the quality of service [123]. For completeness, the starting point is a 10% loss and moves up to a 50% loss. While this may be rare and potentially unavoidable if this is the case, it is still useful to determine the feasibility of the attack. Even with higher packet loss, movement data may still be present with (spuriously) retransmitted packets if the network was deemed unfit for a robot to continue reliably performing operations over. In Mininet, the packet loss is the rate of packets lost (% of random packets per second) over a given link. The results for this experiment can be seen in Table 3.5. In comparison with the baseline in Table 3.1, results for packet loss are similar to those for network link delay as shown in Table 3.4 with a significant increase in precision and recall across all movements. At 10% loss, there is near perfect accuracy with only slight drops in precision, most notably for the X movement. This is due to some X samples being mispredicted as XZ movements. At 25% packet loss, there is much more of an improvement, with most classes having perfect precision and recall. The X movement here improves, however, the Z movement precision decreases in comparison to the precision at 10% loss, with Y movements incorrectly predicted as XZ. Finally, at 50% loss, a similar trend can be observed. However, the precision for the X, Z and XZ movements decreases but still significantly better than the baseline. As with link delay, introducing a percentage packet loss over the robot-controller link also results in greater precision, recall and overall classification accuracy for all robot's movements. Given the use of TLS as the secure channel technology for the robot-controller link in the emulated network, drops in packet arrivals will result in transmissions with increased interarrival times. Notably, some work highlights a possible correlation between

U = # Unknowns, P = Precision, R = Recall										
	U=2		U=3		U=4		U=5		U=6	
	P	R	P	R	P	R	P	R	P	R
X	68%	84%	73%	86%	71%	87%	52%	99%	54%	99%
Y	68%	55%	71%	65%	62%	79%	61%	82%		
Z	79%	63%	79%	71%	79%	65%				
XY	23%	67%	25%	74%						
XZ	67%	90%								
Unknown	95%	54%	97%	62%	88%	74%	90%	65%	99%	92%

Table 3.6: Open-World Classification Results
These results show the impact on classification accuracy when decreasing the number of known movements while increasing the number of unknown movements. There is a correlation regarding accuracy, where it decreases as the number of known movements increase

packet loss and higher link utilisation which can increase packet interarrival times [124].

3.4.3 Open-World Evaluation

In the main set of experiments, closed-set testing was used where it is assumed the attacker can detect a known, strict set of movements. However, realistically, the attacker may only know a subset. Thus, this naturally provokes the need to understand the impact on classification when only some movements are known. In this case, a comprehensive approach for the dataset is labelling each movement progressively and leaving the rest of the movements unlabelled, to observe the impact of increasing numbers of unlabelled classes. As shown in Table 3.6, an open-set approach is taken, which involves labelling each class progressively, leaving the rest unlabelled. The aim of this is to observe the impact of U-unknown classes. Within this set of experiments, 1 unknown class is discounted as this would be the same as the baseline. First, for 2 unknown classes ($U = 2$), fairly similar outcomes are oberved when compared to the baseline results shown in Table 3.1. For 3 unknowns, there are improvements in precision and recall for all

remaining movements. At 4 unknowns, there is only a slight improvement for the Z movement and the recall of the X and Y movements, but a reduction in precision for X and Y compared to 3 unknowns. For 5 unknowns, precision and recall for the Y movement stay relatively consistent, but the precision for X drops greatly with the recall increasing. A greater percentage of X movements are incorrectly predicted where they should fall into the unknown class. Finally at 6 unknowns, the results are similar to those of 5 unknown classes but the recall is near perfect for the X movement. Overall, the precision and recall for unknown movements increases and this was expected. Furthermore, the increase in recall in most cases for movements still known is also expected given that the subtle feature differences are more present given a larger variance in feature values for the unknown set.

3.4.4 Workflow Reconstruction

Finally, for the last set of experiments in the evaluation of this attack, the goal was to investigate whether it is possible for an adversary to use the inferred movements to reconstruct operational workflows corresponding to a set pattern of movements (C_5). For this experiment, the focus was on a subset of common warehousing workflows that involve robotic arms. This includes: pick-and-place, push, pull and packing operations (Figure 3.6). These were chosen, as they represent those that are unique and common operations to a realistic warehouse which makes use of these robots [125–128]. Quantifying the accuracy of recovering these workflows demonstrates that an adversary could reveal daily operating environments within logistics supply chains that could later be used for ransom, or to gain an operational advantage,

among other motives.

For these workflows, inspiration was taken from existing industrial robot datasets, such as the *Forward Dynamics Dataset Using KUKA LWR and Baxter* [129] for pick and place and the *Inverse Dynamics Dataset Using KUKA* [130] for push/pull. At the heart of these workflows is the actual dynamic movements themselves which may be aided by additional input (i.e. from sensors). Ultimately, given that movement patterns are the primary factor which establishes these workflows, it is reasonable to conduct this experiment on reconstructing workflows from traffic patterns solely using position data. In total, there are over 100 sets of test samples for each workflow with varying speeds of movement, distances and directions to evaluate the effectiveness of the attack in this context.

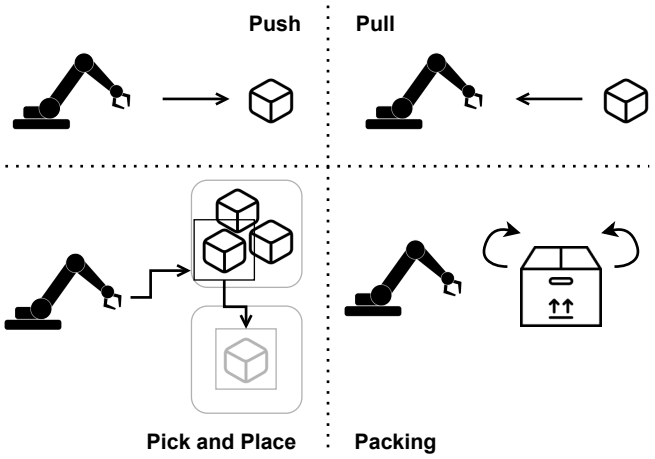


Figure 3.6: Warehousing Workflows

As shown in Table 3.7, it can be observed that, on average, the manufacturing workflows can be recovered much better than individual movements, averaging around 90% accuracy. This is an important result as it demonstrates that continuous monitoring of movement patterns can reveal potentially confidential workflows and could be given to competing facilities. Further, this information can even combined

Operation	Recovery Rate	Pos Changes
Push	97%	2–3
Pull	97%	2–3
Pick-and-Place	84%	7–9
Packing	88%	6–9

Table 3.7: Workflow Reconstruction Results
Warehousing workflows can be recovered at a much higher rate compared to individual movement fingerprints and a pattern-matching approach

with other side channels, such as the acoustic side channel, which may help increase the accuracy of the attack.

3.5 Discussion

In this chapter, it was investigated whether an adversary can indeed still fingerprint robot movements even when the traffic is protected by some *mandatory* channel-security technology such as TLS, ultimately compromising operational confidentiality.

From a preliminary analysis, it can be seen that simply observing variations in traffic features, such as via eye-balling or basic frequency analysis, an adversary could potentially identify some movements but with difficulty. Furthermore, given variations on a smaller scale among combinations of movements, it is much harder to identify and fingerprint them in this manner (Figure 3.4). This motivated the need for a machine learning approach to identify the robot operations in encrypted traffic.

Upon evaluation, it is clear that a passive adversary can reconstruct workflows even when the traffic is encrypted, with high accuracy. In the baseline, most movements can be fingerprinted with at least 65% accuracy and precision. In realistic settings, however, this may be less than ideal. For example, let's consider MIT and Boston Dynamic's

Dr. Spot [131, 132], which makes use of a teleoperated quadruped robot for measuring patient vital signs. This work demonstrates that parameters such as distance, where this correlates with how far Dr. Spot is away from the patient, could lead to leakage of what vitals are being measured. In combination with other sources of information, such as GPS devices, this may lead to further unintentional information leakages, such as what triage zones were visited and why. Based on examples such as this, the evaluation was taken further to explore higher levels of granularity pertaining to information leakage – the distance and speed of robot movements. Exploring this shows that these parameters are in fact very meaningful to an attacker, where accuracy and precision increased by at least 10%. Furthermore, in realistic cases, such as Dr. Spot, teleoperation would be conducted over a wireless network where factors such as packet loss and delay/latency come into play. Upon evaluation, it is clear that realistic network conditions provide an attacker with much better accuracy for fingerprinting robot movements, with almost perfect precision and recall in all cases. Finally, the attack was taken a step further, investigating whether the attacker can reconstruct entire workflows. Specifically, the investigation in this context related to manufacturing workflows. It was found that an adversary could reconstruct such workflows with at least 85% accuracy. From here, it could be possible to compromise operational confidentiality by combining the workflow fingerprints inferred from this side channel attack with other sources of information, such as delivery times, package metadata and even information from other side channels.

3.5.1 Implications

While this work explores the efficacy of the attack in a warehousing context regarding teleoperated industrial robots, it is important to recognise that teleoperated robots are also prominent in other application areas. Notably, a rise in these systems are found in surgical settings, such as the Zeus and da Vinci surgical systems [43, 133, 134]. Given the similarities of such systems in comparison with surgical robots, for example, it is important to explore how this attack may indeed impact the parties involved in these contexts. Specifically, given that end-to-end encryption is not enough to mitigate attacks on confidentiality, it is key to understand the implications the attack has on legislative and professional standards.

Legislative Standards

Legislative standards are those which are set by governing bodies (i.e. national governments) to establish technical detail, such as the safety of a system, which should be conformed to in order to achieve long-term objectives. Given the similarities in architecture for both industrial and surgical robots, both will be discussed to not only cover the context used for demonstrating the efficacy of the attack, but other robotic systems as well.

In industrial contexts, there are two legislative standards that are in effect. First, the IEC 62443-4-2:2019 – Security for Industrial Automation and Control Systems - Part 4-2: Technical Security Requirements for IACS Components [135]. This international standard states that data confidentiality is one of the foundational requirements for control systems and the current advice is to maintain data confidentiality. Second, the CAPSS2021 Security Characteristic 1.1 [136] states in

Dev.406 that appropriate cryptographic algorithms such as TLS/IPsec must be used to protect information in transit over untrusted links to counter interception.

In surgical contexts, there are two key legislative standards from both the USA and UK. First, the Health Insurance Portability and Accountability Act (HIPAA) states that end-to-end encryption measures such as TLS should be used to prevent unintentional disclosure of Patient Health Information (PHI), and ensure any transformation of data results in a low probability of assigning meaning [54]. Furthermore, HIPAA privacy rules state that medical devices which transmit, receive or record PHI should be HIPAA compliant. In this case, one could argue that given surgical robots, such as da Vinci, make use of PHI and therefore should also comply with HIPAA. Second, according to the UK Medical Devices Regulation (MDR) 2017 [71], surgical robotics are medical devices and classified as class III (highest level of control) [72]. The MDR does not explicitly describe what should measures to take to protect PHI but only indicates that protection is required. The EU Guidance on Cybersecurity for Medical Devices [137] does mention encryption [138] but nothing more substantial. Ultimately, this attack showcases that employing TLS is not enough to mitigate or prevent information leakage relating to robot operations. Furthermore, the evaluation also shows that workflows can be reconstructed and in combination with other information, such as admission and exit times in a surgical context, could lead to confidentiality breaches (e.g. PHI). Simply put, this leaves robotic systems to be non-compliant with legislative standards that mandate TLS.

Overall, it is clear that legislative standards require investigation to protect against unintentional information leakages from side channels,

and update existing mitigation and defence requirements.

Professional Standards

Professional standards refer to practices and behaviours which entities must adhere to. In the context of this attack, the likes of hospitals and warehouses are examples of professional entities to which these standards apply. Taking the surgical context as an example, the NHS (UK) the code of practice, which governs information security management [139], describes standards that must be adhered to in order to maintain the protection of PHI, among other confidential records. This information is detailed as the *lifeblood* of the health service (2.13) and is covered under the Data Protection Act 1998 [140]. Further, other policies such as the Clinical Commissioning Policy for Robot-Assisted Surgery (NHS England) [141] also define professional standards for surgical robots. Unfortunately, in both cases data confidentiality is ignored and only touch upon this matter within an access control and authentication context.

3.5.2 Limitations

The attack shows significant accuracy in terms of workflow reconstruction, yet there are some limitations. Specifically, two key limitations arise regarding both a deviation from robot path and procedure carried out, and ageing effects.

Disconnect Between Tool Motion and Workflow

The first limitation corresponds between the path of the arm and/or tool, and the workflow being carried out. When tools are deployed, such as scalpels or pumps/grippers, they may have the same traffic

pattern for many operations. For instance, during a scalpel incision operation, the thickness may result in different cuts for the same position or direction. Furthermore, the starting position of the workflow can have a significant impact on planning, as the tools used must account for environmental characteristics in which the workflow is taking place (i.e. physical height). Another example includes the rate of workflows, such as the cutting rate in the case of incisions or opening of packages, which are unlikely to be recovered from a traffic analysis approach. Therefore, appropriate compensation techniques, such as information from other side channels, may provide useful mitigation to these effects but ultimately, these parameters are not derivable from a traffic analysis approach alone.

Ageing Effects

In the evaluation of this attack, a single robot is used. One question which arises in this case, is that if the classifier is to be trained on samples from a new, *fresh* robot, would it be possible to determine whether there may be a deterioration in accuracy over time as the robot ages? In the case of this traffic analysis attack, this should be mostly unaffected with the payload information carrying key inference properties which stay consistent throughout a robot's lifecycle. However, network component ageing would result in wear-and-tear that can physically affect device function [142] and ultimately result in an impact, to some degree, on factors such as link delays. To address the concept of component ageing in a robotic system, the exploration of other side channels, such as acoustics or power analysis, could be done to observe potential impacts. The reason for this is due to the fact that side channels based on physical processes are subject to ageing,

such as with belt-driven robots developing slack over time which could induce drift in physical characteristics such as noise. As well as this, the impact of device calibration over time could results in variations of robot movements. For example, improperly calibrated motors could result in jerky movements.

3.6 Countermeasures

While the attack shows success for fingerprinting robot movements while traffic is encrypted with TLS, as well as reconstructing of operational workflows, it is important to investigate potential defences an organisation can employ to mitigate or entirely protect against the impact of the attack.

3.6.1 The Onion Router (Tor)

One first potential countermeasure is The Onion Router (Tor). Tor is a low-latency, circuit-based overlay network, which enables anonymous communication by allowing different streams to overlap each other. This enables traffic volumes to remain hidden, achieving perfect forward secrecy (assurance of non-compromisable session keys even for long-term secrets). Specifically, after a Tor circuit has been demolished (route from start to end point nodes), traffic is unreadable. Given that Tor has been successful in other applications as a defence, such as website fingerprinting [82], and the fact it is application transparent, it is important to establish how Tor can perform as a countermeasure to this attack against robotic systems.

Movement	Precision	Recall
X	38 %	57 %
Y	53 %	88 %
Z	76 %	8 %
XY	35 %	45 %
XZ	86 %	49 %
YZ	44 %	27 %
XYZ	60 %	80 %

Table 3.8: Tor Classification Results
The Tor samples make use of samples with the same traffic characteristics as the baseline, showing a classification accuracy of ˜49% – a decrease by at least a factor of 2 in comparison

Tor Setup

In order to evaluate Tor as a defence, the first step was to establish a Tor hidden service which receives the control commands sent by the controller over HTTPS to the hidden service. Tor hidden services provide two-way anonmyity so it is not possible to distinguish the IP addresses of inter-communicating components. While HTTPS is typically not used within a robot system, the focus is on the message protocol being the main carrier, with the payload of the HTTP message left to encapsulate the protocol with minimal overhead. Using Wireshark, incoming traffic to the hidden service machine (the control commands) was monitored and common flows corresponding to the robot traffic were captured. The traffic in this experiment was routed through multiple autonomous systems (AS) over around 20 circuits. This allowed for a more realistic approximation of the results.

Findings

From the results in Table 3.8, in comparison with the baseline results shown in Table 3.1, the precision across most movements decreases slightly (averaging around a 20% decrease), with the exception of the

Operation	Recovery Rate	Pos Changes
Push	45%	2–3
Pull	47%	2–3
Pick-and-Place	51%	7–9
Packing	48%	6–9

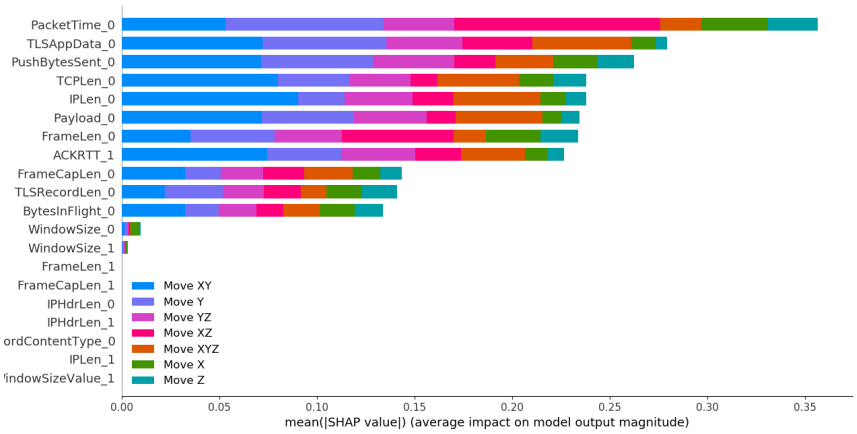
Table 3.9: Tor Workflow Reconstruction Results
The results show that workflow reconstruction accuracy is hindered when Tor is employed, in comparison with those shown in Table 3.7

XY and XZ movements which show an increase in precision of 14% and 18% respectively. In the case of recall, more decreases across the movements are seen, with an increase of 34% only present for the Y movement. Notably, there is a substantial drop in recall for the Z and XZ movements with a decrease of 55% and 43% respectively. Further, looking into performance impact of Tor, the latency does not (overall) present as a big problem for many cases with packet times fairly sporadic but under 1s. However, in critical cases this wait time for control commands to be received may be less than desirable.

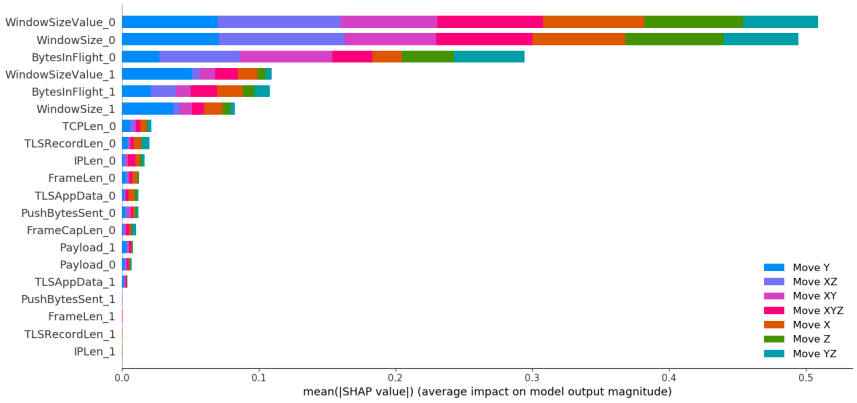
The next step was to investigate the success of workflow reconstruction when Tor is employed as a countermeasure. The results are found in Table 3.9. In comparison with the reconstruction results in Table 3.7, the use of Tor as a countermeasure reduces the recovery rate by at least a factor of 2, which is a significant drop. However, this accuracy in comparison with baseline invidual movements (non-patterns) is relatively similar.

While Tor does show some strength as a countermeasure, it is unclear as to why this is the case by simply observing the result output alone. To this, SHAP analysis [143] was employed to analyse the most prominent features in both the TLS and Tor datasets. SHAP values allow for the evaluation of the impact of features in the dataset, on

the predictions made by the neural network model. The SHAP values can be seen in Figure 3.7. From these values, it is clear that without Tor employed, packet time remains the most important feature, with the TCP packet features (i.e. application data and payload) considered highly important as well. However, under Tor, it is interesting to observe that neither of these features are of much importance for classification. Instead, features such as window size and bytes-in-flight are better. In the case of TLS packet-size related features, Tor connections make use of padding cells sent in both directions at varying transmission intervals depending on consensus parameters. This leads to the payload of TCP packets to be tranmitted in fixed-size cells of 514 bytes, or if the payload is smaller then the cell is zero-padded. While there may be some similarities between movement patterns, with regard to existing TCP features, a lack of variability leads to lowered reliance by the neural network. Interestingly, Tor does not affect the bytes in flight or the window size. The BytesInFlight feature is defined by Wireshark as an indicator for the amount of unacknowledged data that the robot controller has transmitted. The shorter the *distance* for receiving ACKs (faster time) results in lower bytes in-flight, and ultimately a lowered window size needed for optimal performance. The window size is an advertisement from the receiving robot of how many bytes of data it can receive at some point in time to control data flow, which may be dependent on the movement(s) being carried out. In Tor, the data is in equal-sized cells which leave the window size to be constant as multiple Tor circuits are multiplexed through the same TCP connection [144–146]. In the evaluation of Tor, different circuits were used for each movement over multiple runs to minimise this. However, even with this additional step, window size



(a) SHAP – TLS



(b) SHAP – Tor

Figure 3.7: SHAP Values for TLS/Tor Datasets

seems to remain an important feature.

Overall, while Tor does reduce the accuracy of classification in most cases, some of these reductions are only slight. Further, there is also a presence of increase in some movements as well. Therefore, it still may be possible for an attacker to successfully fingerprint movements over longer monitoring periods. In terms of workflow reconstruction, however, the rate of reconstruction is greatly reduced and leaves little success left for an attacker to feasibly carry out the attack.

3.6.2 Other Countermeasures

While Tor may be useful as a protection measure against workflow reconstruction, it is still not infeasible for an adversary to carry out the attack entirely. Therefore, it is important to realise the potential for other countermeasures which may perform better alone, or be used in conjunction with Tor. As a part of future work, two other potential countermeasures can be considered, including: padding robot traffic, and mixing robot traffic with other background traffic.

Padding

While there exists a number of padding techniques, padding an encrypted message fundamentally aims to make traffic analysis harder by obscuring the true size of messages (i.e. payload). The choice of this length may be randomised or constant-rate. Constant-rate padding techniques, such as those based on the perfect secrecy theory proposed by Shannon [147] (i.e. inserting dummy traces to created padded traffic) have been shown to not prove as effective against statistical techniques. To this, Fu et al. [148] propose a variable rate traffic padding countermeasure which can defend against attacks by leveraging sample variance and entropy to exploit correlations between traffic rate and packet inter-arrival times of padded traffic. Other approaches such as BuFLO [149] which removes side channel information by sending packets of fixed-length at fixed intervals also show promise for areas such as HTTP traffic analysis. However, in the case of many safety-critical robotics systems which are time-critical, this may be less than ideal. In any case, it would be interesting to determine the impact of a countermeasure that pads traffic using variable inter-arrival times, which have shown to be effective regardless of sample statistics col-

lected by an adversary when sample distributions of the inter-arrival times for the robot traffic are analysed, to produce a design guideline for a VIT-based approach [148, 150].

Mixing

The second potential countermeasure is mixing, which involves mixing in background traffic with the robot traffic. Existing approaches show that regularising traffic (e.g. constant-rate padding) may induce a higher overhead, or induce delays in general traffic, which question the suitability of this as a defence for robotic systems. Instead, a more lightweight approach which does not require additional infrastructure would be ideal. Thus, approaches to mixing in background traffic, such as GLUE [83] which adds dummy traces to have DNS traffic appear as a longer consecutive trace, could be modified and applied to robot communication protocols. This would make it harder to identify end points with much lower overhead. This is shown to be more successful than existing defenses in the area of website fingerprinting, as many existing attacks rely on single traces.

3.7 Related Work

In this work, the goal was to determine whether robot movements could be fingerprinted from traffic characteristics, when the channel is protected by TLS. The problem of detection and prediction of applications using encrypted traffic traces has been investigated from a variety of different angles. Early approaches to identification and prediction focused on identifying application traffic such as for firewalls, websites and quality of service mechanisms and identifying

actions (such as device user actions) [75, 151]. Many of these early approaches focus on payload- and signature-based detection, which do not work as well when traffic is encrypted [152]. Further, such approaches took either a graphical approach – through understanding of social networks and motifs to understand patterns of communication and relationships between features – or a simple statistical approach through probability density functions of traffic features and port-based classification [75, 153]. From these limitations, the advent of machine learning approaches have shown to be advantageous in relieving such limitations by combining statistical and graphical approaches to build patterns which can associate traffic with application protocols [154]. However, machine learning approaches mainly apply to labeled data and require to be "taught" when results are incorrect, to which deep neural networks do not require human intervention to learn from mistakes (machine learning approaches almost always require structured data).

With the ubiquitous nature of network traffic, the prominence of deep learning techniques increased significantly in areas such as: traffic classification for website fingerprinting [155]; device/user [156, 157] fingerprinting; and distinguishing between VPN and non-VPN traffic [99], among others [76, 158, 159]. While many of these techniques have shown success, there has been little on identification of safety-critical IoT systems such as robotics which is the focus of this work. Oh et al. [155] describe the use of deep neural networks for website fingerprinting and show that in comparison to other state-of-the-art fingerprinting techniques, the deep learning approach demonstrated a significant increase in classification accuracy. Furthermore, the promise of deep learning approaches is backed up by similar successes in identifi-

cation over TLS-encrypted traffic [76, 158, 159]. Further, these similar approaches describe that the additional features supplied to traffic with the adoption of TLS can bring higher accuracy compared to standard packet features used in earlier approaches, such as packet size and timing features.

3.8 Summary

In summary, it is clear that it is possible for a passive adversary to still identify robot movements, even when the traffic between a robot and controller in a teleoperated architecture is encrypted under TLS. Further, the attack evaluation showcases that even more fine-grained movements (identifying movements of specific speeds and distances) and entire workflows can also be inferred in an effort to compromise the operational confidentiality of organisations. While the attack is successful, there are countermeasures which hinder its feasibility. Therefore, this provokes a natural response for an adversary to explore other attack vectors in the domain of side channels to achieve better success in passive reconnaisance.

4 — Passive Reconnaissance of Robotic Workflows via Acoustic Side Channel

In the previous chapter, the focus was on mounting an information leakage attack to compromise the operational confidentiality of robotic workflows via traffic analysis in the cyber domain. However, while in general there is little focus on passive attacks, in the physical domain this is less so. With a similar goal to the traffic analysis attack, one can question whether there are other side channels that can be used to mount the same information leakage attack with the aim of achieving better success.

In the physical domain, side channels on similar computer systems have been studied [160–162]. For example, 3D printers have been shown to leak senstive data relating to intellectual property that could be reconstructed from unintentional information leakage in the power side channel [163]. While older robots could be compared to CNC machines like 3D printers, modern robotic systems are much more dynamic and unpredictable compared to more static behaviours seen in other systems, leaving movement inference a much more difficult task for an attacker to achieve.

From the main research questions presented in Chapter 1.1, this chapter focuses on an extension to question 1, examining whether the same

information leakage attack presented in Chapter 3 can be mounted to compromise operational confidentiality via the acoustic side channel in the physical domain. This attempts to investigate whether fingerprinting accuracy and workflow reconstruction are more successful than traffic analysis in the cyber domain.

4.1 Background

Robot movements which involve electromechanical components produce vibrations [24, 164]. Higher vibration amplitudes lead to the production of sound waves. Further, if these resonate at a frequency within the human hearing range, audible sounds will emanate. In the design of modern robotic systems, aspects such as design and motion are key considerations, but aspects belonging to noise and vibration are also important given that noise is inherently generated as they move. Trovato et al. [165] describe that sounds generated as a consequence of the vibrations of robot components can convey meaning. While this is described as a means of communication, such as tonal sounds (e.g. high energy continuous *beeps* at a pure tone) being used to indicate alarms, using this notion one may question whether such meaning can indicate behaviours even when this sound is unintentional. Specifically, given that stepper motors and other components will emit sound during operation, could this sound be used to mount an information leakage attack which targets the operational confidentiality of organisations?

4.1.1 Threat Model

Many previous attacks focus on an active attacker, which can involve the tampering of messages [15] or replaying attacks between the robot or controller [17]. In this chapter the primary attacker is a passive insider, such as a malicious technician or operator near. Being an insider near the robot would allow them to record the acoustic emanations during the robot's normal operations using a smartphone, which they may have on them and use covertly [166]. As well as this, it is also possible than an insider attacker is able to covertly *plant* a microphone which could transmit recorded audio to the attacker remotely or be retrieved at a later time. In either case, if an attacker is able to mount an information leakage attack to fingerprint robot movement patterns from acoustic emanations, this could lead to the revelation of sensitive workflows (i.e. in a warehouse) and ultimately compromise the operational confidentiality of the organisation. For example, this information could be given to competitors to gain an advantage or use it maliciously.

A second possible threat comes from a telemonitoring perspective. While telemonitoring is less common in industrial settings, in surgical settings the use of medical recording devices, such as medical data recorders or intraoperative video recorders, are used for post-surgical review or teaching (alongside patient consent) to learn from suboptimal scenarios and improve performance [167–169]. While privacy laws and medicolegal requirements govern the use of such devices, data from them is not typically required as evidence in court so long as patient confidentiality is maintained [170]. However, acoustic emanations captured by such recordings could reveal the operations the robot is carrying out, and ultimately piece together surgical proce-

dures. Combined with other metadata (e.g. patient admission and exit times), patient confidentiality could be breached. In any case, for telemonitoring, VoIP technologies may be employed [25, 26] and thus looking at the same side channel attack but leveraging VoIP call audio is a novel threat that also requires investigation. Furthermore, it is also entirely possible that someone on the factory floor can make use of VoIP to carry out the attack from an insider's perspective, either intentionally or unintentionally. In the case of intentional attack, someone can make a VoIP call to a remote attacker or computer system running the VoIP receiver. In the unintentional cases, an insider could be making a legitimate call but unintentionally also record the robot operating and leak information through this call. Further, it is also possible for such an insider to have malware on their mobile device and this malware could record the robot movements and/or transmit them.

Ultimately, reviewing the nature of acoustic emanations in robotic systems, as well as the proposed threat model, the aim of this chapter pertains to the investigation of whether an advesary can record the acoustic emanations from a robot during its normal operation. and make use of distinct characteristics of the recorded audio to fingerprint robot movements and workflows. Several hypothetical factors will come into play which could influence the potential success of this attack. First, the type of operations being carried out by the robot can vary in terms of speed and distance of movement, and so the attack should be robust enough to fingerprint between these parameters. Second, the distance at which the microphone is situated away from the robot will also have an impact on the success of the attack, for example examining the inverse square law [171] where sound intensity

decreases with distance away from source, and should be investigated. Finally, given that in some cases VoIP technology will be employed, such as for recording purposes or to livestream medical procedures with surgical robots, the impact of VoIP audio on the attack should be evaluated. Ultimately, the following research questions are proposed:

(R_1) Can an attacker fingerprint individual robot movements on each axes, as well as permutations of them?

(R_2) How is movement fingerprinting affected by:

(i) The speed and distance of movements?

(ii) The distance the recording device (i.e. smartphone) is away from the robot?

(R_3) Can entire robot workflows be reconstructed from acoustic emanations?

(R_4) How do VoIP codecs influence the success of the attack?

4.2 Attack Methodology

Now that the goals of the study have been set out, the next step is to set up the robot environment, parameters of the study and the feature extraction process for movement and workflow fingerprinting.

4.2.1 Robot Environment

The context of this study surrounds teleoperated robots used in warehouses, whose typical architecture can be viewed (at a high level) as a pairing between the robotic system itself and its controller (teach

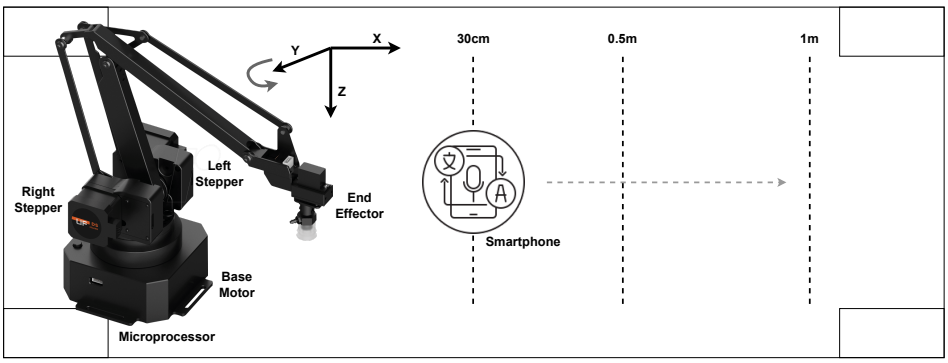


Figure 4.1: Robot Environment for Acoustic Side Channel

pendant). For this work, uFactory's uARM Swift Pro is used which runs on an Arduino Mega 2560 with MicroPython installed. The controller is run on a Windows 10 laptop using the uARM Python (3.8.X) SDK to enable controller instructions to be written in Python which are then translated into instructions understood by the robot. An overview of the robot environment used in this study is depicted in Figure 4.1. For capturing the acoustic emanations which arise when the robot operates, the robot is positioned on a rectangular table with the smartphone placed in varying distances away from the robot. The recordings were made while the robot carried out its various operations within a moderately sized computer lab that can suitably hold around 15 people.

4.2.2 Experiment Parameters

With the robot setup for evaluating the acoustic side-channel attack, the parameters of the study can now be outlined.

Speed and Distance

In addition to capturing the acoustic emanations which arise during operation along the X, Y and Z axes and permutations of them (R_1), it is important to evaluate more fine-grained movements. To this end,

the robot is programmed to do movements with varying distances (in millimetres) as well as varying speeds of movement (mm/s) ($R_{2(i)}$). This is because in realistic cases, a surgical robot for example would not move in each direction with constant distance and speed. Therefore, it is vital to understand whether an adversary can also fingerprint this meta-information as well as just the movements themselves to provide more granularity to movement inference from an attacker's perspective.

Recording Distance

As well as speed and distance of robot movements, the distance at which the smartphone or recording device is placed away from the robot will play a key role in attack success ($R_{2(ii)}$). The effect of the inverse-square law [171, 172] leads to sounds softening with distance and, trivially, the further the microphone is away from the robot, the lower the amplitude of the captured sound. It can be hypothesised that as the distance of recording decreases, the accuracy of the attack will also decrease, but it is important to evaluate to distinguish whether this may be the case, and how much of an impact it has with regard to the scale of the robot. The microphone is initially set to record robot movements at 30cm away, but for completeness distances of 50cm (0.5m) and 100cm (1m) away from the robot will also be evaluated with respect to the scale of the robot.

VoIP

The final parameter for this study is to evaluate the impact VoIP has on the success of the attack (R_4). For this study, the codec employed by the majority of VoIP applications is Opus [173,174]. The first step

is to observe how the codec performs, but also how packet loss will also affect audio quality and the success of the attack.

4.2.3 Acoustic Characteristics

The first step to determining an appropriate attack strategy is to understand the different characteristics of acoustic emanations and what may be most useful from an attack perspective.

Root Mean Square Energy

Root Mean Square (RMS) energy [175] is a measure of the amplitude based on all samples in a frame of audio and can be thought of as an indicator of loudness of the audio signal [176]. This may be useful in the context of this attack given that combinations of movements (i.e. simultaneous movements along two or more axes) may produce a louder sound given the use of multiple stepper motors, for example. As well as this, as the robot components cross over the microphone, the sound may be louder and thus this feature may help provide further information to the discrimination between movements in different positions.

Zero-Crossing Rate

Zero-Crossing Rate (ZCR) is a measure of the number of times a signal crosses the horizontal time axis and can help identify pitch variations in monophonic tones (sound emitted from one location) [176]. Given the robot is stationary in this case, the ZCR may be a useful feature candidate.

Spectral Centroid

The spectral centroid provides information corresponding to frequency bands that contain most of the energy, where lower centroid (energy) values are linked with *duller* sounds and higher centroid values for *brighter* sounds [177]. In a robotic system, smaller movement distances and speeds will naturally require less energy and appear more dull sounding to the human ear, whereas faster and longer movements have better tonality, and may ultimately provide useful for distinguishing between different movements of the same source.

Spectral Bandwidth

Spectral bandwidth is defined as the *full* width of band of light (wavelength interval) at half the peak maximum [178,179]. Acoustic signals oscillate about a point and the bandwidth for each time interval in a signal is the sum of the maximum deviation on both sides of this point. The point of the centroid of the signal may vary for different robot movements and may be an important feature for fingerprinting.

Spectral Rolloff

Spectral rolloff is the fraction of frequency bins under a cutoff point where the total energy of the spectrum is contained and can help distinguish between noisy sounds and more harmonic sounds (below the roll-off point) [180]. This feature may provide useful to this attack as it can *roll off* frequencies that may fall outside of the *useful* range of frequencies where the energy of the sound of movements is contained.

Spectral Contrast

Spectral contrast is the measure of energy of frequencies in windows of time [181] and can help identify strong spectral peaks to reflect the distribution between harmonic and non-harmonic components of the acoustic emanations. As a robot moves, the frequency contents may have energy that changes with time and capturing the spectral contrast can help measure this energy variation.

Chroma Feature

Chroma feature, sometimes referred to as a chromagram, profiles a sound into 12 *pitch class profiles* [182]. In music analysis, the attempt is to capture the harmonic and melodic characteristics of a song where pitches can be categorised to one of the scales in the equally-tempered set $\{C, C\#, D, D\#, E, F, F\#, G, G\#, A, A\#, B\}$ [183, 184]. While recorded robot movements are not akin to songs that are analysed in this fashion, the pitch of sound may correlate with the speed and distance of movement and may provide useful as a mid-level feature for fingerprinting movements.

Mel-Cepstrum Frequency Coefficients

The Mel scale is a scale of pitches that is felt to be equal in distance from one another. For example, in audible acoustics listened by a human, differences in frequency content can be observed if the source of acoustic emanations are in the same distance and atmosphere [185, 186]. The short-term power spectrum of acoustic emanations can be represented by the Mel frequency cepstral (MFC) and a combination of coefficients (MFCCs) make up the MFC. The MFC equally distributes frequency bands to approximate human au-

ditory response. While observing the robot operate in person, it was clear that there are some variations present among the sounds (e.g. pitch change) while listing to the robot in the lab, which provoked the thought of including this as a feature. If variations in robot movements can be inferred from audible sound, then looking at MFCC coefficients (the list of amplitudes of the spectrum in the mel scale) will provide useful information to the attack.

4.2.4 Acoustic Dataset

After determining the appropriate acoustic features to extract from the captured sounds, the next step was to create the dataset. Before the robot is programmed to move, the recording is started and then the robot can execute the programmed movements. After this is finished, the recording is then stopped. In this dataset, there are 2 subsets. Within both subsets, there are samples pertaining to both individual and permutations of movements with varying speeds and distances of movement, the microphone distance, and robotic warehousing workflows. These workflows are the same as those found in the radio frequency side channel chapter (Chapter 5), and include workflows such as pick-and-place, packing, push and pull operations, which were replicated from those found in existing industrial robot datasets such as the *Forward Dynamics Dataset Using KUKA LWR and Baxter* [129] and the *Inverse Dynamics Dataset Using KUKA* [130]. For these workflows, movements were slightly perturbed to account for a small degree of entropy that may be present in real-world operations (i.e. those that may arise due to drift in equipment calibration or wear-and-tear). In contrast to the first subset, the second subset contains the same samples but are passed through the Opus codec to

evaluate the impact of VoIP on recorded audio in this attack. This was achived using the Opus codec tool suite. Specifically, while all samples are passed through the Opus codec, they are further split by packet loss. Packet loss has been shown to negatively impact call quality in VoIP communications [187, 188] as they cause impact in the form of dropped calls or parts of speech, slow rate of speech (latency) or noise/interference. Because of this, these further subsets are divided by packet loss values of 1%, 5%, 10%, 25% and 50%. As a whole, the first subset contains 27.2K samples for individual movements and 658 samples for warehousing workflows, with each subset using 60% of the total samples for training, 20% for validation and another 20% for testing. The second contains the same amount of samples for each of the packet losses evaluated.

Pre-Processing

The features in the dataset, as listed above, are computed using the *librosa* [189] Python library. For each feature, the mean value of each feature across each signal sample is taken and computed from a Short-Time Fourier Transform (STFT) with a Hann window and FFT length of 8192. For the MFCCs, 14 coefficients are used. Typically, 8–13 are used with the zeroth excluded given it only represents the average log-energy of the input signal [190].

4.2.5 Neural Network

Before an evaluation can take place, the next step is to construct an appropriate neural network architecture for fingerprinting movements and ensuring a successful attack. To create the neural network, a sequential model was used where neurons are grouped in a linear

fashion. This was created using the Keras API [104]. The parameters and structure for the layers in the neural network were evaluated on the dataset using a cross-validated grid search [105, 106] to find the most optimal number of neurons, layers, activation function and dropouts if necessary. The input for the maximum number of neurons to be tested was calculated using the formula proposed by Demuth et al. [108], also used to construct the neural network in Chapter 3, with an alpha branching factor of 2. Using the grid search with 3 cross validations, the most optimal neural network architecture for this feature set consists of 6 layers. First, the input layer containing 21 neurons for each of the input features. Next, there are 4 hidden layers. The first is a *Dense* layer with 290 neurons and uses the ReLU activation function [107]. The next hidden layer is a *Dropout* layer which is used to randomly set input units to 0 at a rate of 0.05 at each step during training to prevent overfitting. The next layer is another *Dense* layer of 350 neurons with ReLU activation, followed by another Dropout with a rate of 0.05 to prevent overfitting. The number of neurons, while suggested by the formula by Demuth et al. [108], were selected using a grid search with 3 cross validations. Finally, the last layer is a *Dense* output layer of 7 neurons, one for each of the movement classes, and uses the SoftMax activation function [109] to have the output in the range of $[0, 1]$ for use as predicted probabilities. Sparse categorical cross-entropy is used as labels are integers and not one-hot encoded, for which categorical cross-entropy would be used [110]. The optimiser used is Adam [111] with a learning rate of 0.001. This learning rate was chosen as others, such as those with higher learning rates, resulted in lowered accuracy scores. The choice for the activation and optimisation functions, while decided using a cross-validated

grid search, are further detailed in Chapter 3.3.3. The model was fitted with a batch size of 32 and was run for 1000 epochs.

4.3 Evaluation

After setting up the robot environment and capturing the acoustic emanations during various stages of operations, the next step is to evaluate the success of the attack. As per the research questions listed in Section 4.1.1, the evaluation of this attack and related results will be organised in this order.

4.3.1 Individual Movement Fingerprints

The first research question (R_1) aims to investigate whether an attacker can infer individual movements (on each axis) and permutations of these movements from the recorded audio. To compare this against other parameters, this experiment is considered as a baseline where the speed and distance of movement are the lowest possible values (1mm and 12.5mm/s respectively), and no VoIP codec used. As seen in Table 4.1, an average accuracy of around 75% can be observed across all movements, with the YZ movement having the highest precision among the movements. In comparison with the traffic analysis side channel, there is an increase in accuracy of around 10%. Interestingly, Y-involved movements are better recovered than other movements overall. This may be due to the Y-axis moving across the microphone range. Looking at the Z-involved movements, these are among the lowest. This may be due to the Z axis involving a vertical movement only and not moving nearer the microphone for better recording (Figure 4.1). Interestingly, looking at the confusion matrix

Movement	Precision	Recall
X	76%	81%
Y	77%	78%
Z	61%	71%
XY	78%	80%
XZ	68%	65%
YZ	85%	78%
XYZ	72%	67%
Accuracy	75%	

Table 4.1: Baseline Classification Results
As a whole, the baseline accuracy is 75% which is fairly good fingerprinting accuracy for an attacker and outperforms the traffic analysis side channel

seen in Appendix C.1, it is clear that the Z-involved movements are incorrectly predicted as one another.

4.3.2 Impact of Movement Distance

For the next research question (R_2), the evaluation will look into how the distance and speed ($R_{2(i)}$) of robot movements impact the success of fingerprinting movements from the acoustic side channel. First, as a robot moves, there is likely to be more sound that can be recovered as the distance of movement increases. As seen in Table 4.2, an increase by a single distance unit increases the model accuracy by 1%, improving Y-involved movement precision by around 10%. Furthermore, the Z movement also gains a slight increase in precision. Unfortunately, this results in lowered accuracy for the other movements. This increase in distance results in the sound of movement being held for longer and may either provide useful for distinguishing variance between movements or even reduce this variance. To explore this, larger distances of movements are explored. The confusion matrices for distance experiments can be seen in Appendix C.2. At 5mm,

there is a drop in accuracy of around 4%, with X-involved movements having much higher accuracy. At 10mm, the accuracy of the model overall decreases significantly to 57%. Y-involved movements in this case are much poorly fingerprinted, yet X-involved movements have a further increase in precision. For the Z movement at this stage, there is unfortunately a further drop in precision but the recall remains relatively similar. At 25mm, the accuracy starts to improve by 7% with the X movement having similar precision and recall to 10mm, and most other movements have an increase in both precision and recall. Finally, at 50mm, the accuracy nears that of the baseline and 2mm, however X-involved movement accuracy is significantly improved. Notably, one may question why distance stops at 50mm. This is simply because the range of movement, specifically for the X and Z axes, are limited and will result in only three or less movement variations above this. While a large number of repetitions may provide better success for these distances, this is an influence of bias as this will result in an imbalanced dataset [191]. In other robots, such as larger scale industrial robots, larger distances can be explored and this is a point of future work.

4.3.3 Impact of Movement Speed

After looking at movement distance, the next parameter for robot movements is the speed at which the robot is moving along each of the axes ($R_{2(i)}$). As seen in Table 4.3, the speed parameter is less accurately fingerprinted by the attack compared to the distance parameter by at least 10% on average. Interestingly, a similar pattern is observed regarding X-involved movements, with accuracy increasing with speed, except from the XYZ movement. While there are slight

D = Distance (mm), P = Precision, R = Recall												
	D=1		D=2		D=5		D=10		D=25		D=50	
	P	R	P	R	P	R	P	R	P	R	P	R
X	76%	81%	69%	71%	77%	87%	85%	58%	83%	70%	86%	84%
Y	77%	78%	88%	77%	77%	79%	80%	54%	66%	43%	90%	88%
Z	61%	71%	65%	83%	64%	79%	51%	81%	64%	71%	83%	66%
XY	78%	80%	68%	60%	67%	63%	63%	53%	57%	65%	79%	81%
XZ	68%	65%	62%	57%	83%	47%	67%	49%	60%	61%	64%	79%
YZ	85%	78%	94%	94%	66%	83%	37%	54%	55%	57%	56%	59%
XYZ	72%	67%	69%	81%	76%	68%	45%	52%	63%	84%	64%	58%
Accuracy	75%		76%		72%		57%		64%		74%	

Table 4.2: Classification Results With Distance Parameter
At a slight increase in distance, the accuracy remains similar to the baseline, but further increases in distances lead to a reduction in fingerprinting accuracy. Notably, unlike the baseline, X-involved movement are better fingerprinted at distance

S = Speed (mm/s), P = Precision, R = Recall										
	S=12.5		S=25		S=50		S=75		S=100	
	P	R	P	R	P	R	P	R	P	R
X	76%	81%	57%	81%	54%	74%	78%	53%	72%	81%
Y	77%	78%	79%	76%	61%	42%	59%	45%	72%	69%
Z	61%	71%	50%	56%	52%	58%	62%	84%	77%	75%
XY	78%	80%	73%	72%	46%	40%	67%	57%	57%	70%
XZ	68%	65%	79%	57%	75%	79%	57%	60%	60%	56%
YZ	85%	78%	67%	59%	66%	45%	53%	65%	66%	69%
XYZ	72%	67%	65%	63%	51%	66%	54%	57%	62%	47%
Accuracy	75%		66%		58%		60%		66%	

Table 4.3: Classification Results With Speed Parameter
The speed parameter performs worse than the distance parameter in the acoustic side channel

drops in accuracy, the precision and recall across most movements remains similar as speed increases. This is interesting, as the initial hypothesis was that a higher speed would result in higher pitched acoustic emanations, however the results seem to contradict this. In any case, perhaps the perceptual characteristics for human audio, while a clear pitch change is present listening to the robot in the lab, used by the feature algorithms regarding pitch (i.e. chroma feature) may not pick up on this for robot sounds. The confusion matrices for speed experiments can be seen in Appendix C.3.

4.3.4 Microphone Distance

While observing more fine-grained information leakage is useful to an attacker, one problem that may impact the success of the attack is the distance the recording device is away from the robot ($R_{2(ii)}$) – in this case, the smartphone. Naturally, due to the inverse square law [172, 192], the intensity of sound decreases over distances and one would hypothesise that, because of this, the accuracy may be significantly impacted as the distance of recording increases. Specifically, the power contained within the audio sample is inversely proportional to the square of the distance from the robot emitting the sound. Thus, as the distance the microphone is away from an operational robot, the intensity (i.e. loudness) of the sound is four times less. In this experiment, two other microphone distances (50cm and 100cm) are also tested in addition to the baseline recorded at 30cm. While these are not large recording distances, given the small scale of the robot used for the evaluation of the attack, these are relatively suitable candidates to be tested. As seen in Table 4.4, as the distance the microphone is away the robot is increased, the accuracy of the attack compared to the baseline decreases by around 10% at each recording distance step. Notably, this is much more significant for Z-based movements which were previously described to have poorer fingerprinting accuracy due to the limited range of motion that does not cross the recording device (remains stationary and moves vertically). In this case, a point a future work may be to evaluate the impact on position of the smartphone around the robot, aside from facing in front. Collectively, recordings from multiple angles may provide better fingerprinting accuracy in all cases.

MD = Microphone Distance (cm), P = Precision, R = Recall						
	MD=30		MD=50		MD=100	
	P	R	P	R	P	R
X	76%	81%	57%	79%	75%	76%
Y	77%	78%	67%	74%	67%	68%
Z	61%	71%	48%	66%	45%	63%
XY	78%	80%	88%	91%	64%	68%
XZ	68%	65%	61%	52%	51%	40%
YZ	85%	78%	83%	54%	47%	35%
XYZ	72%	67%	52%	39%	33%	33%
Accuracy	75%		65%		54%	

Table 4.4: Classification Results With Microphone Distance
As the microphone distance increases away from the robot being recorded, on average the accuracy decreases around 10% at each step compared to the baseline - more significantly for Z-based movements

4.3.5 Workflow Reconstruction

The next step in the evaluation looks at whether entire warehousing workflows can be reconstructed through the acoustic side channel (R_3). While a pattern matching approach can be successful using individual movement fingerprints, the ability to reconstruct entire workflows may be useful from an auditing perspective, for example, where offsets in *normal* movement signals can be flagged and investigated further. As seen in Table 4.5, the explored warehousing workflows can be recovered on average with around 62% accuracy. Notably, the pick-and-place and packing workflows are recovered with much higher success than the push and pull workflows. Simply, the former have much more variation in the pattern of movements and thus the variance helps with fingerprinting. In the case of push and pull movements, both are highly similar and it can be hypothesised that only the direction of movement away from the microphone (i.e. pull is a reverse of push) provides at least some degree of accuracy between the two. Looking into the confusion matrices found in Appendix C.6, this appears to confirm this hypothesis.

Workflow	Precision	Recall
Push	37%	16%
Pull	31%	59%
Pick-and-Place	100%	96%
Packing	97%	100%
Accuracy	64%	

Table 4.5: Workflow Reconstruction Results
Common warehousing workflows can be reconstructed in their entirety and are better recovered through the acoustic side channel if they are more complex and varied. Push and pull operations are less accurate due to the fact they are very similar movements

4.3.6 Impact of VoIP

In certain robotics environments, such as in surgical settings, procedures may be streamed and/or recorded for viewing, education or research [25–27]. Therefore, it is important to question how VoIP impacts the audio samples for movements and workflows and, ultimately, the success of the attack (R_4). In many modern VoIP applications, the Opus codec is the preferred choice [173, 174] given its standardisation and rank of higher quality compared to other audio formats for a variety of bitrates. To explore this, the open-source nature of Opus allows for easy implementation to encode and decode the audio samples and, during decoding, investigate various packet losses. In VoIP applications, Packet Loss Concealment (PLC) is used as a decoder feature for receiving data from an unreliable source, which masks the effects of packet loss in VoIP communications. In realistic settings, packets may arrive late, be dropped or be corrupted, which may result in not only a lowered audio quality but in the worst case, dropped parts of the audio or the entire audio sample entirely. Given that in VoIP applications, a 1% packet loss is considered an acceptable rate for VoIP to minimise impact on call quality [193, 194], however in the event of

network failures or availability attacks this may be higher. For completeness, 5 packet losses of 1%, 5%, 10%, 25% and 50% are evaluated. Furthermore, as it was shown that constant bitrate quality does not perform as well as variable bitrate quality [195], the audio samples are encoded and decoded with variable bitrate. This experiment used the same model as the previous experiments, but fitted with a batch size of 256 and 100 epochs of training. As seen in Table 4.6, the results for the baseline speed and distance of movement (12.5mm/s and 1mm respectively) under various packet losses via the Opus codec can be seen. Interestingly, at low packet loss, the classification accuracy is around 90% and increases by around 15% compared to the baseline without VoIP employed. Further, X movements are more accurately fingerprinted across all packet losses compared to the baseline without VoIP. As the packet loss reaches more undesirable amounts of 25% and 50%, the accuracy slightly decreases but the accuracy still remains much higher than the baseline without VoIP. For reference, the confusion matrices for the VoIP experiments can be found in Appendix C.5. The reason for these results may be due to the PLC algorithm switching between CELT or SILK mode and variable bit rate. Specifically, frames that are deemed important are re-encoded at a lower bitrate and allows for partial recovery for important lost packets. This may be targeting the movement audio within the sample thus leading to higher variance among classes. Another reason for the higher accuracy may also be due to the dynamic jitter buffer. As frames arrive after the length of the jitter buffer has been exceeded, they are discarded [101]. The Opus codec adapts to lossy conditions by not only switching between modes but also by embedding packet information into subsequent packets for better reconstruction rates in

L = Loss (%), P = Precision, R = Recall										
	L=1		L=5		L=10		L=25		L=50	
	P	R	P	R	P	R	P	R	P	R
X	99%	99%	99%	100%	100%	100%	100%	100%	99%	100%
Y	90%	94%	87%	96%	90%	92%	82%	97%	86%	97%
Z	86%	72%	88%	68%	91%	73%	86%	72%	90%	74%
XY	88%	91%	91%	88%	88%	91%	94%	81%	90%	81%
XZ	89%	93%	82%	83%	82%	82%	80%	85%	80%	85%
YZ	86%	89%	87%	88%	85%	89%	91%	80%	91%	85%
XYZ	93%	98%	94%	97%	92%	96%	89%	97%	90%	97%
Accuracy	90%		90%		90%		88%		89%	

Table 4.6: Classification Results (Baseline) With Opus Codec and Packet Loss

Interestingly, the precision and recall remains relatively similar across packet losses, with a slightly drop in accuracy for undesirable large packet losses. Notably, there is an increase in accuracy of around 15% compared to the baseline without the Opus codec employed

CELT mode compared to SILK. Even if some sample packets are lost, the data sampling rate is low enough that there are still enough samples for fingerprint recovery. Looking at Figure 4.2, the spectrograms clearly show that when the codec and loss are introduced, the sound corresponding to the robot movement is more prominent and earlier in the sound (prioritised) and there is clearly less noise impact on the audio sample when the codec is employed. Again, this may be due to the dynamic jitter buffer and the rate at which frames arrive within the time series [101]. Further, as the loss increases, there is some more noise present at the beginning of the sample, appearing at 25% loss from $0 - 0.05$s.

4.4 Discussion

The acoustic side channel attack showcases potential for another passive side channel attack which can compromise the operational confidentiality of organisations, but in the physical domain.

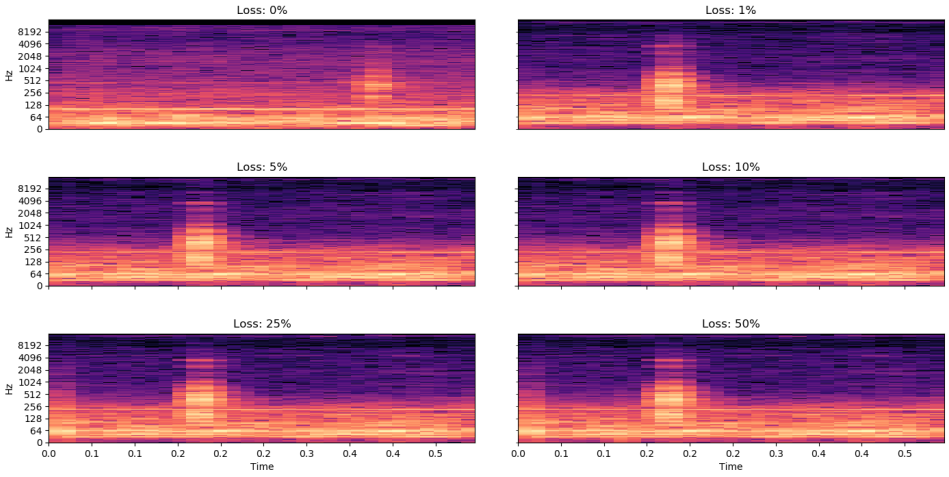


Figure 4.2: Spectrograms Demonstrating Impact of Packet Loss and Codec

4.4.1 Influence of Noise

During the recording of acoustic samples for robot movements, there is likely some degree of background noise that should be accounted for. Given the recordings were made while the robot carried out its various operations, within a moderately sized computer lab that can suitably hold around 15 people, background noise effects may include the likes of light chatter, keyboard tapping and rolling chairs, among others. While relatively good accuracy is observed even with the background noise, it is important to also look into techniques to eliminate such noise to determine whether this results in better fingerprinting accuracy.

In the human auditory system, sound waves contain the relative signal of the oscillations due to density and pressure of air in the ear. In digital audio, sound waves are encoded in digital form as numerical samples in a continuous sequence (time series). The recordings taken in this attack are recorded at a sampling rate of 44.1KHz with 16-bit depth and thus there are $65,536$ possible values the signal can take in the sequence.

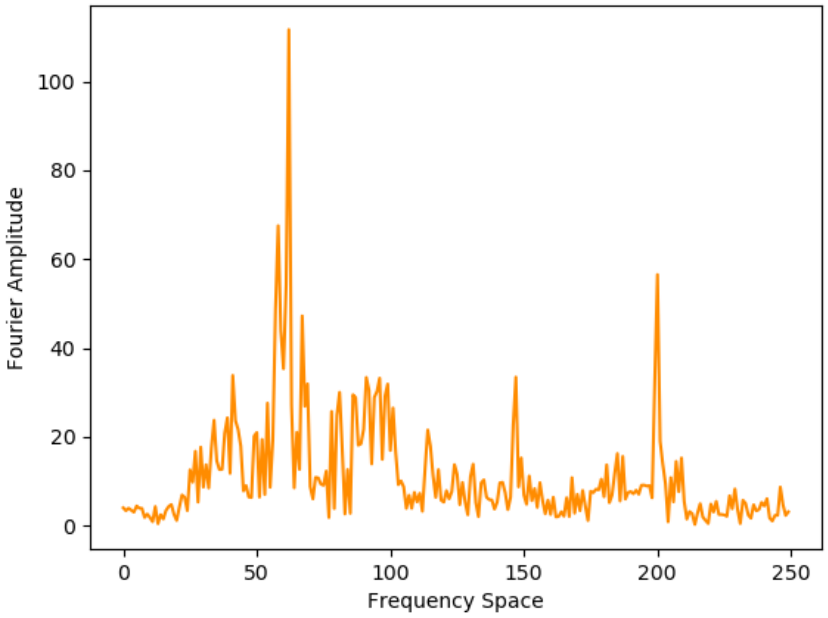


Figure 4.3: FFT of Acoustic Signal
Peaks can be observed at 60Hz corresponding to electric hum, with other peaks at 150Hz and 200Hz (among others) which may correlate with robot movement

As shown in Figure 4.3, the amplitude of the frequency content of the acoustic signal can be observed using the Fast Fourier Transform (FFT). In this attack, the techniques are originally applied to human acoustics, but given that the robot movements produce sound that is audible to the human ear as well, they may also be applied. Looking at the frequency content, notable amplitude was not found past 1KHz, so the scope of frequency content is narrowed further to 250Hz as signal oscillations appear more random past this point. There is a notable spike around 60Hz, which is the frequency standard common to alternating current and is an effect known as *electric hum* due to electrical noise getting into an acoustic recording medium. The next largest peaks can be observed at around 150Hz and 200Hz which may correspond with the robot movements. Further investigation into

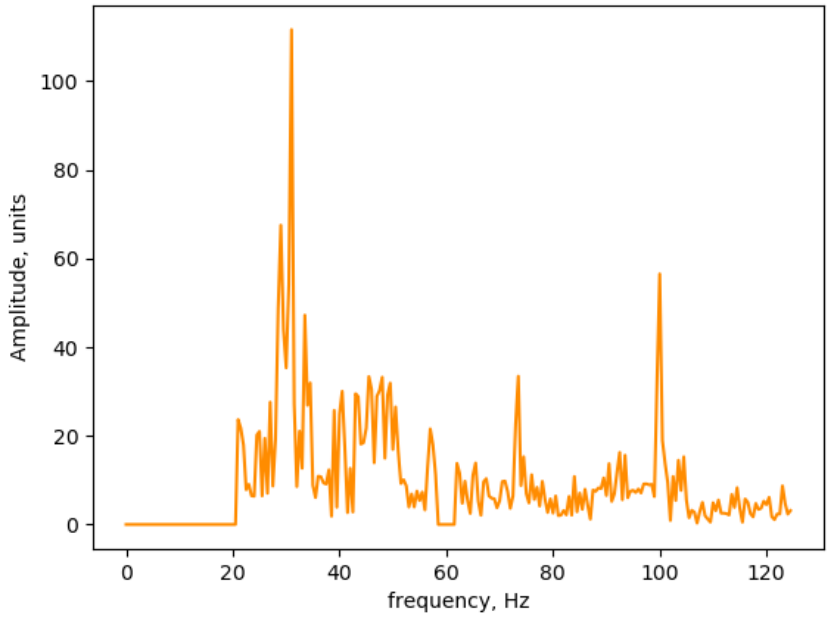


Figure 4.4: FFT of Acoustic Signal (Filtered)
The amplitude at points correlating with electric hum or those outwidth the human hearing range are set to 0 (filtered out)

duty cycles and PWM frequencies used may be useful to investigate the likely frequency spectrum and location of frequency peaks. As a first step to noise reduction/filtering, one technique is amplitude filtering, where the amplitudes of FFT values to be filtered can be set to 0Hz, to which the original signal can be recreated using an inverse FFT. Doing this filters out frequencies where there is high signal concentration related to noise. In this experiment, the electric hum, as well as frequencies outwidth the human hearing range ($>$ 20Hz and $>$ 20KHz) are filtered by dropping the amplitude of these ranges. The smartphone used in this study could not record past the upper bound. A depiction of the amplitude drop can be seen in Figure 4.4. Looking at Table 4.7, the accuracy of baseline movement fingerprints can be observed with amplitude filtering in place. While the accuracy overall decreases by 1% compared to the baseline without

Movement	Precision	Recall
X	72%	81%
Y	75%	73%
Z	68%	72%
XY	86%	71%
XZ	69%	68%
YZ	76%	83%
XYZ	72%	70%
Accuracy	74%	

Table 4.7: Amplitude Filtering Classification Results
While the accuracy is slightly reduced compared to the baseline with no filtering, the precision for some movements increases further, with better recall seen in most cases

amplitude filtering, the precision for Y and XY movements increase. This may be due to unfortunate noise events present in these samples that the filter has rectified. However, there is still a reduction in overall accuracy, which may mean that electric hum and other peaks may not be the best indicators of noise to remove when recording a robotics system. In this case, as a point of future work other noise reduction techniques that have shown to be successful in other areas, such as stationary or non-stationary spectral gating [196, 197] which reduce noise in time-domain signals by estimating noise thresholds for the frequency bands in a signal to gate (mask) noise below the threshold, are worth exploring in the hope that attack accuracy improves.

4.4.2 Other VoIP Codecs

Opus is the primary choice for many VoIP applications due to its royalty free and open source nature, alongside the benefits of higher quality and low-bandwidth streaming, in comparison with other codecs such as Speex [198] or SILK [199] (Opus' predecessor). While it may be interesting to evaluate other codecs, Opus is the main choice for

the majority of modern applications, such as Zoom, Teams and Discord [200, 201] and has taken over previously dominating codecs.

4.4.3 Defences

While the attack is successful, and even more so when the attack targets VoIP communiations, a natural question pertains to countermeasures and defences against the acoustic side channel attack. In this work, acoustic emanations result in unintentional information leakages about robot behaviours and can ultimately lead to the compromise of operational confidentiality.

One defence that could be considered is to make use of vibration- or sound-reduction mechanisms to hinder the effect of the attack. As seen in Section 4.3.4, as the microphone distance increases the accuracy of fingerprinting also decreases. While this is due to the inverse-square law [171, 172] that is naturally at play with regard to sound intensity (i.e. loudness), a reduction in this from other means may result in the same outcome of reduced success of fingerprinting. Techniques in this space include the likes of using vibration isolation pads [202] or damping to reduce vibration [203, 204] for the robot as a whole. In the case of noise reduction for robot components such as stepper motors, potential defences include using a clean damper [205] or higher resolution stepper motors.

Another potential defence is to make use of a masking noise to interfere with attack inference, by distorting the signal related to information leakage in the acoustic side channel [206–208]. Adding a masking signal has shown success, but two challenges need to be addressed. First, the mask must be similar to the signal requiring masking to ensure difficult separation. Second, the masking noise should not cause any

degrading effect on usability of the robotic system. For example, if the masking noise is to cover up other sound such as those used for emergencies or other operator feedback, then this will be much less than ideal and potentially lead to catastrophic liabilities.

4.4.4 Limitations

One limitation of this work is the robot used. In terms of direct replicability for a real-world industrial robot arm, this robot is much smaller than one typically seen on the factory floor. While the attack may not provide the same level of movement inference in this case as with the robot used in this study, a larger robot may employ more motors that require more power to operate and ultimately may vibrate more emanating more acoustic sound. Given that the attack is successful here, perhaps larger vibrations (and ultimately louder and clearer sounds) would lead to the attack simply requiring extra exploratory analysis in terms of tuning the parameters and FFT length in feature extraction. As well as this, it would be interesting to see how this attack compares for different robotic systems. The relates to another limitation regarding noise itself. Some robotic systems which do not require sound for alerts, etc. may make use of certain materials to reduce the vibrations and noises emitted. This may be particularly prominent in more safety-critical contexts such as nuclear power plants. In either case, this would require an analysis of multiple different robotic systems which may or may not employ noise reduction methods and is a point of future work.

A second limitation one can consider is a disconnect between the tool equipped at the end-effector and the motion being carried out. The robot arm used in this study is equipped with an integrated pump

with a maximum pressure of 33kPa and maximum lifting weight of 1kg. The goal with the use of the tool was to determine whether an attacker could identify the contents (by weight inference) of packages being lifted. Unfortunately, there was no difference in sound from the pump being on and lifting with no object, with the lifting of different objects. However, given that the pressure of the pump required for lifting may correlate with power consumption, this is a further side channel that could be explored and is another open challenge in the space of future work.

4.5 Related Work

While acoustic side channel attacks have not been explored for robotic systems, enhancing the novelty of this work, there has been previous research in the area of acoustic side channels. In a similar respect to robotics, the exploration of information leakage in the acoustic side channel has been explored for 3D printers [206] – some of which making use of smartphones to carry out the attack [160, 209] – and additive manufacturing systems [161]. However, these attacks focus on IP theft. The acoustic side channel attack presented in this work focus solely on the movement of the robot arm and the compromise of operational confidentiality, which when looking at the bigger picture is much more valuable to an attacker. Furthermore, the reconstruction of G-code is an unnecessary extra step as movements which correspond to these can be inferred from individual movement fingerprinting under the assumption the robot is operated by an Arduino. Furthermore, while the robot in this work is operated by an Arduino, the focus is on reconstructing movements from the acoustic emanations, irrespec-

tive of the microcontroller used and thus applies to robotic systems in general and not those restricted to being operated by an Arduino.

4.6 Summary

In summary, it is clear that even acoustic emanations provide a high level of accuracy for fingerprinting movements and showcases a possible passive side channel attack in the physical domain. Similarly, the acoustic side channel can result in the same compromise of operational confidentiality but with much higher accuracy than traffic analysis (Chapter 3). However, the accuracy could be considered infeasible to some degree with consideration given to risk versus reward. This provokes the need to look at other passive side channels in the physical domain and see if the performance of the attack is much better.

5 — Passive Reconnaissance of Robotic Workflows via Radio Frequency Side Channel

In the previous chapter, the first passive side channel in the physical domain was explored, exploiting unintentional emanations of acoustic sound while a robot moves and attempting to fingerprint robot movements and workflows. While this attack was slightly better than traffic analysis, with further increases in accuracy found in VoIP settings (cyber domain), the accuracy may still not be considered feasible from the perspective of risk vs. reward for an attacker. Because of this, it is interesting to observe how other side channels in the physical domain may compare. From the main research questions presented in Chapter 1.1, this chapter focuses research question 1 and whether the same information leakage attack presented in Chapters 3 and 4 can be mounted to compromise operational confidentiality via the radio frequency (RF) side channel in the physical domain.

5.1 Background

Radio Frequency (RF) is used in a wide array of applications, ranging from TV and radio, to wireless and satellite communications. Specifically, it is a measurement which represents the oscillation rate of

electromagnetic waves, whose frequencies are within range of around 3kHz–300GHz. While the term radio frequency refers to the range of 3kHz–300GHz, this is actually the radio spectrum in which several frequency bands are defined in which different transmission systems operate. For example, AM radio operates within 600kHz–1.6MHz. Electronic circuits emit some degree of electromagnetic emissions while they operate. While these emissions can theoretically span the entire spectrum, the focus of this study is on emissions that fall within the radio frequency spectrum. While there may be many significant sources of unintentional RF emissions, most modern robotic systems will make use of microprocessors and stepper motors. The former has been shown to emit RF due to switching activities of transistors alternating varying current flows [28]. Similarly, in the case of the stepper motors, digital pulses and phase shifts in voltage may also contribute to RF emissions.

In this chapter, a robotic arm is used (the same as Chapters 3 and 4). This robot consists of three stepper motors for each of the three axes (Figure 5.1). The base motor controls the Y axis while the left and right control the X and Z axes. In some cases, the microprocessor may be located in another physical component between the robot and controller, however in others it may be situated in the robot itself. In this robot, the microprocessor is located in the base structure alongside the stepper motor. For the proposed attack, this robot is a suitable replication candidate as it compares to other single-arm robots used in typical industrial settings such as warehouses, which have at least 3 degrees-of-freedom, and also consists of the same principal components for movement fingerprinting (microprocessor and stepper motors). With regard to potential emanations of unintentional RF,

Figure 5.1: Robot Components

the primary component of interest is the microprocessor in the base of the arm robot. However, given that stepper motors may also contribute to sources of RF, any unintentional emissions from them may also be useful to an attacker.

5.1.1 Threat Model

The goal of this attack is to conduct a *stealthy* approach to fingerprint the movements a teleoperated industrial robot via the radio-frequency side channel. If successful, robot movement fingerprints can be used to reconstruct operational workflows. The context in which this attack is evaluated is a logistics warehouse, where products are packaged and stored or moved around the warehouses (i.e. along conveyor belts) to then progress to the next stage in a supply chain.

The primary adversary considered in the scope of this attack is an insider, such as technical staff who operate or maintain the robot on the warehouse floor. To capture unintentional RF emissions, an RF receiver is used. These have varying ranges for frequency capture but also can vary in physical size as well. A passive insider here should be stealthy to avoid possible detection and thus the question pertaining to this is, does there exist a small enough receiver to be used to capture the RF emanations from the robot to minimise the possibility of detection? Aside from the ability to conduct the attack, another important consideration relates to the opportunities that arise from the successful collection of movement fingerprints. By collecting movement fingrprints, it would be possible to correlate a series of movements with known patterns that correspond to warehousing workflows, such as picking and placing products between two conveyor belts. Furthermore, it may also be possible to reconstruct workflows directly by capturing the unintentional RF emanations at the time of these workflows and later performing direct workflow fingerprinting (as opposed to workflow identification from a collection of movement patterns). In either case, the information leakage of these operational workflows can expose a degree of operational-level detail that may not otherwise be available to an attacker. For example, operational confidentiality in terms of claimed efficiency of warehousing procedures or what operations are carried out in the warehouse could be collected and be used as bribery, leaked to competitors, or be used to discredit an operator or organisation at the expense of compromising operational confidentiality. To this threat model, the following research questions arise:

(R_1) Can an adversary identify teleoperated robot movements and

permutations of these movements via unintentional RF emissions?

(R_2) Can these robot movements be fingerprinted with more granularity? Specifically, does the speed or distance of movement impact the accuracy of fingerprinting?

(R_3) Does the distance at which the antenna (RF receiver) is placed away from the robot impact classification accuracy?

(R_4) Can *higher-level* warehousing workflows be reconstructed from unintentional RF emissions?

5.2 Attack Methodology

An overview of the methodology for this attack can be seen in Figure 5.2. First, a Butterworth bandpass filter is applied to eliminate undesired frequency content. Next, the Short-Time Fourier Transform (STFT) of the signal is computed to represent the frequency content over time, which is then used to compute the Mean Frequency Profile (MFP) to observe the width to frequency peaks and the cadence of robot movements. Principal Component Analysis (PCA) is then used as a technique for dimensionality reduction resulting in a smaller feature set for each signal, which is then normalised before fingerprinting. Further detail to each of the steps in the methodology are subsequently described below.

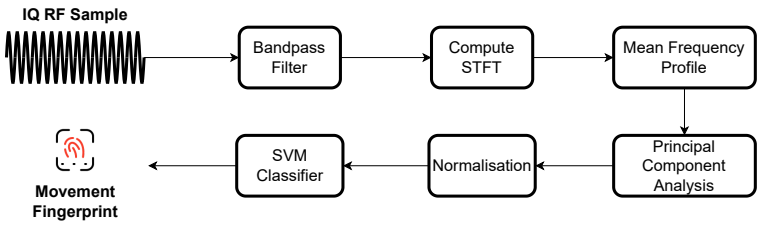


Figure 5.2: Attack Methodology

5.2.1 Experimental Setup

Robot Environment

The first stage of the experimental setup is to establish a realistic and replicable robot environment. In this environment, a robot (consisting of sensors, actuators, etc.) is paired with a controller (i.e teach pendant) that is used to send commands or execute pre-programmed actions which the robot can interpret and execute on the factory floor. In this work, this is replicated on a smaller scale. For the robot, uFactory's uARM Swift Pro was used, operated by an Arduino Mega 2560 running MicroPython. The controller is run on a Windows 10 laptop running the uARM Python (3.8.X) SDK. To capture RF emissions, a Mini-Whip Medium-Shortwave Active Antenna was used, which was placed near the base of the robot arm. The antenna amplifies the unintentional RF signal (emitted during robot operations) and transmits this to an RSPdx RTL-SDR receiver to capture raw IQ data of the RF signals via a shielded coaxial cable. This antenna is suitable for the scale of the robot in this study, with an operating frequency of 10kHz–30MHz and a small physical size of $113 * 32 * 7mm$. The small physical size aims to demonstrate that this attack can be conducted stealthily, by*hiding* near the robot, as well as economically given its cost. The RF emissions were captured as IQ files using SDRuno at a sampling rate of 2MHz. An overview of the robot and antenna setup

can be seen in Figure 5.3, including the robot, RSPdx SDR, antenna and pointer towards the location of the controller.

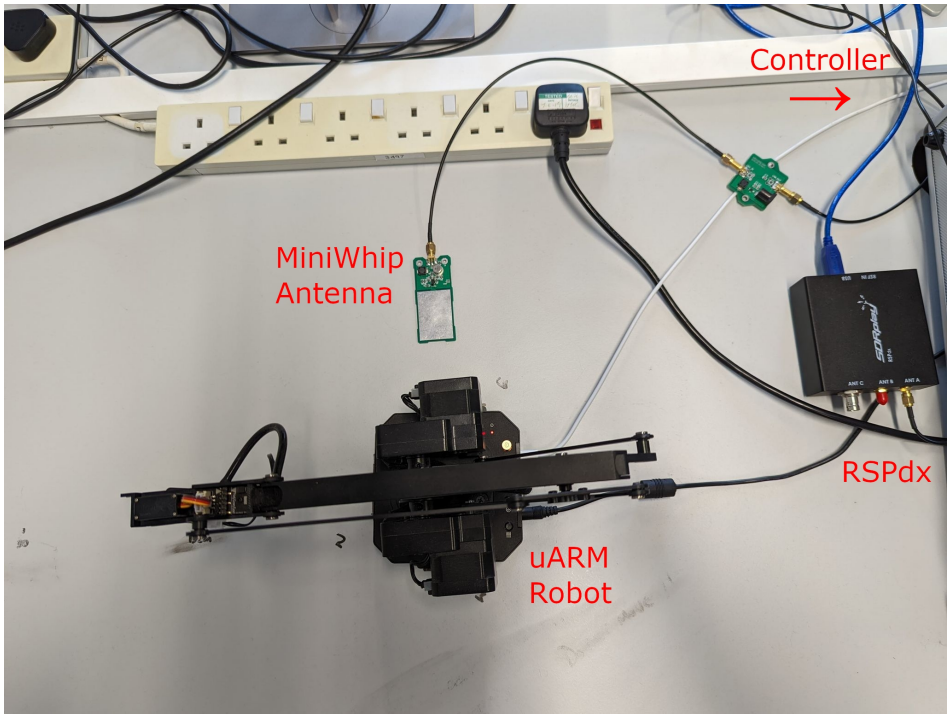


Figure 5.3: Robot Environment
The attack environment consists of the robot, MiniWhip antenna, RSPdx SDR and path to controller

Movement Dataset

The next stage of the experimental setup was to create the movement dataset. In this dataset, sets of samples exist that correlate with the objectives of the attack described above (R_{1--4}) and is split into two subsets. The first subset contains samples pertaining to all permutations of X, Y, and Z with varying speeds of movement (12.5–100mm/s) and distances (1–50mm) ($R_1 - R_2$). The robot is programmed to perform these movements at 2s intervals and the RF signal is captured spanning multiple iterations over the range of movement on the axes. The signal is then sliced at this interval correlating with the duration of the movement, which is then pre-processed as described below.

The second contains samples of warehousing workflows such as pick-and-place, push and pull operations which were replicated from those found in existing industrial robot datasets such as the *Forward Dynamics Dataset Using KUKA LWR and Baxter* [129] for pick and place and the *Inverse Dynamics Dataset Using KUKA* [130] for push/pull. Each workflow sample includes the whole duration of the workflow, as opposed to a sequence of samples. A depiction of these workflows can be seen in Figure 5.4. The core information which details these workflows are the dynamic movements being carried out, which are potentially influenced by additional input (i.e. from sensors). Further, also within this second subset, existing data was perturbated (e.g. adding minor jerks to workflows) to form additional samples to account for a small degree of entropy that may be present in real-world operations (e.g. those that may arise due to drift in equipment calibration or wear-and-tear) ($R_3 - R_4$). As a whole, the first subset contains around 7.8K samples for individual movements and the second containing around 400 samples for warehousing workflows, with each using 20% of total samples for testing.

5.2.2 Feature Extraction and Movement Fingerprinting

After establishing the movement dataset, the next step in the attack methodology was to extract features from each of the signal samples which would later be used for fingerprinting. The goal here is to ensure that each signal sample for a movement can be easily distinguished from other movements.

First, the goal was to observe whether peaks are present in the captured RF emissions corresponding to robot movements. To do this,

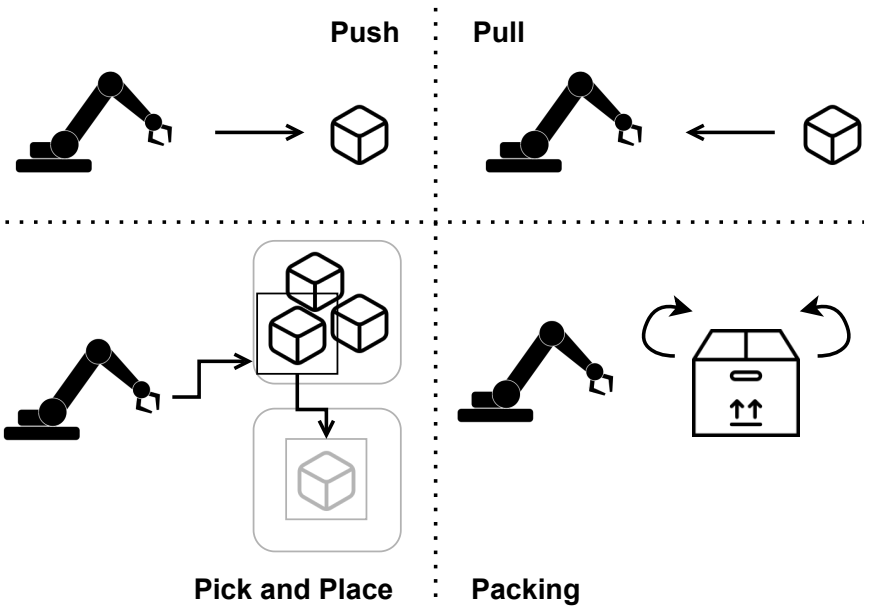


Figure 5.4: Depiction of Common Warehousing Workflows
The dataset contains common warehousing workflows such as pushing, pulling, packing and moving objects

the Short-Time Fourier Transform (STFT) of a set of signal samples is taken and the log-spectra (spectrogram) was computed using a segment length of 8192 and a Hann window as a default, first observation. The STFT allows for the observation of information which concerns variations in frequency content of the signal over time. For this, the robot was programmed to perform an X movement at 2s intervals over a period of 10 seconds. As illustrated in Figure 5.5, peaks can be observed at 2s intervals, which correspond with the robot moving as programmed. However, it is clear that variations present within and between movements are not easily distinguishable through a visual approach such as those used in audio classification via spectrograms [210, 211]. From this, feature extraction is carried out for each sample which will be used as input for movement fingerprinting. The techniques applied for feature extraction are a variation of those presented in the work by Zabalza et al. [212] which demonstrates good

classification accuracy for identifying moving targets via higher frequencies captured by radar systems.

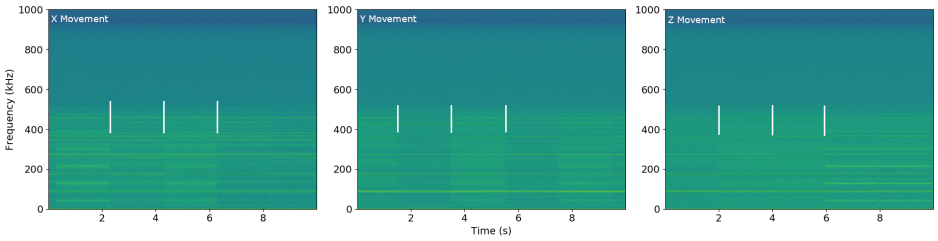


Figure 5.5: Observing Frequency Peaks for X, Y and Z Movements
Observing log-spectra for movements shows movement peaks and demonstrates a visual approach may not be suitable for distinguishing variance within and between movements. The white bars show the spikes which indicate the start of a movement

First, as shown in Figure 5.5, frequencies seem to drop off after around 500kHz. To confirm this, looking at the power spectral density shows that they indeed drop off at around 500kHz and also fall outside a lower limit of 10kHz. To limit the focus to just the information within these upper and lower bounds, the choice was to use a Butterworth band-pass filter (Figure 5.6) to filter frequencies out of this range. Interestingly, the filter closely resembles one of decimation due to an observable "flat top" where bins are not scaled relative to one another as would be observed in Gaussian approaches. The Butterworth filter is chosen over other filters given that there is a quicker roll-off at the cut-off frequencies with no rippling and thus robustly preserving frequency content compared to other linear filters.

After applying the band-pass filter, the STFT is computed to obtain the frequency content of the signal samples over time. The next step is to compute the Mean Frequency Profile (MFP) – the mean of the absolute value of each frequency over time – from the STFT. By computing this, one can observe both the location of frequency peaks (variation in amplitude across movements), as well as the width to

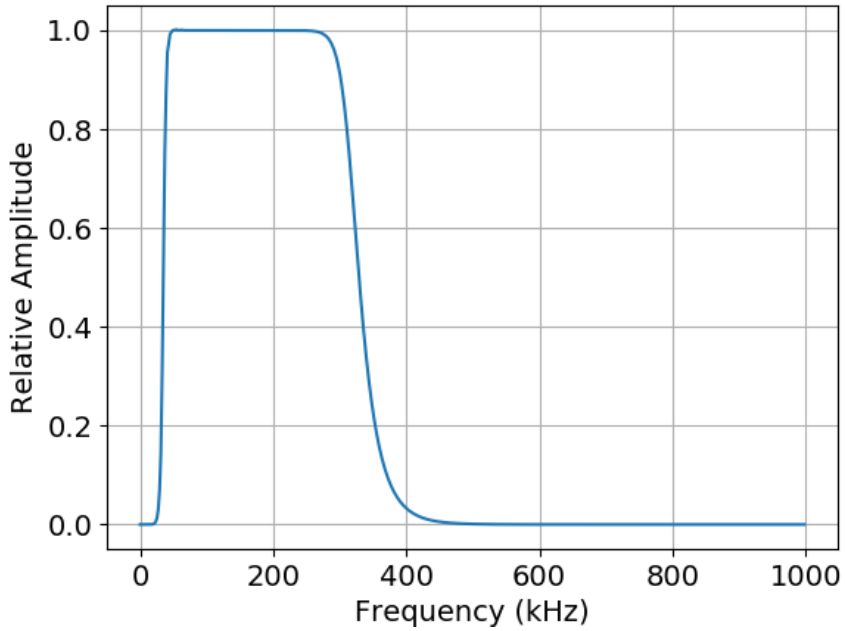


Figure 5.6: Butterworth Band-Pass Filter Amplitude Response
The butterworth bandpass filter is used to eliminate frequencies that fall outside of the range of important frequency content (10–500kHz)

the frequency peak (different movements may have different velocities for each moving component). The MFP is computed as follows:

$$MFP(v) = \frac{1}{M}\sum_{m=1}^{M}|STFT(v,m)| \quad MFP(v) \in \mathbb{R}^{L} \tag{5.1}$$

where M is the number of time instants of STFT and L is the number of discrete points in the Fourier transform. In the case of fingerprinting individual movements, it is clear to assume a constant cadence between target robot movements over windows of time, as they are programmed to be sampled at 2s intervals. By averaging the frequency bins over time, it is possible that some resolution is lost regarding movements of specific components of the robot arm. However, given that the aim is to discriminate between robot movements themselves, rather than parts of the arm, this information is acceptable to discard in this context.

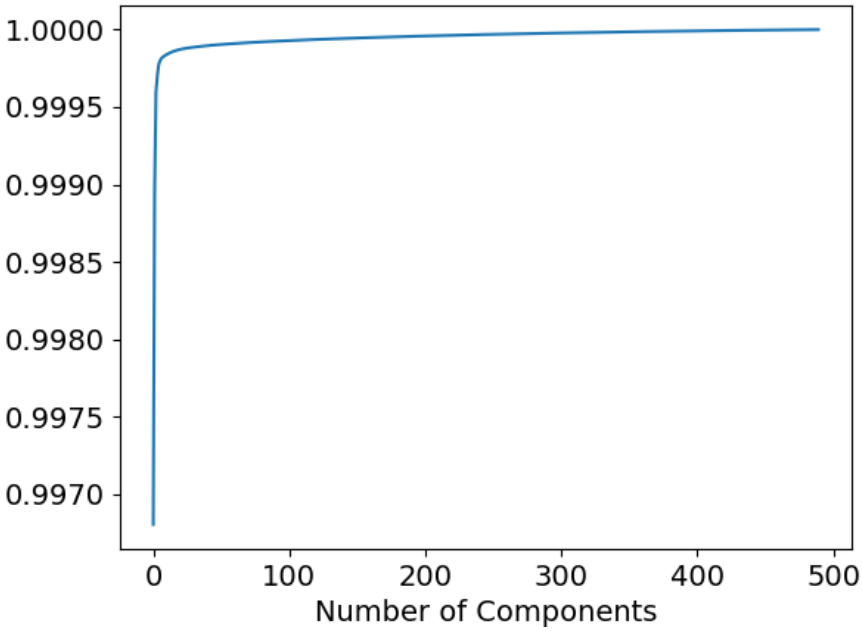


Figure 5.7: Cumulative Explained Variance for Movements
The cumulative explained variance shows that 14 components is enough to represent 99.999% of the overall variation among movements

Even by extracting the MFP from the STFT, the resulting feature vector for each movement sample is still fairly large and would induce a large amount of strain on computing power for fingerprinting. To this, Principle Component Analysis (PCA) is used as a dimensionality reduction technique [213] to decorrelate components of the MFP to a smaller subset that still retains a high level of discrimination among features in the feature vector. This allows most of the information to be represented and analysed to produce the same results with a much smaller feature vector, increasing the efficiency of computing movement fingerprints. In Figure 5.7, the cumulative explained variance among feature vectors for all movements can be observed – an accumulation of variance for each principal component. Overall, it was found that 14 components was enough to represent at least 99.999% of the overall variation among features.

For the last step in the feature extraction stage, normalisation is applied to the resulting feature vectors in the form of zero mean and standard unit variance to produce a scaled feature set to optimise the performance in the classification/fingerprinting stage.

Once a collection of the resulting feature vectors for the movements are put together in the dataset, movement classification (fingerprinting) is done using a C-Support Vector SVM [214, 215] classifier. An SVM was used for this attack, as other approaches such as deep neural networks require a larger sample set and one of the goals of this attack is to conduct it stealthily and efficiently with a small sample set, given potentially limited windows of opportunity for an attacker to make use of the physical RF receiver. Furthermore, the use of an SVM allows for more efficient computation of the movement fingerprints and are easier to train given small datasets. The hyperparameters for the SVM were selected using Grid Search Cross-Validation, which prompted the use of a linear kernel with most optimal kernel parameters of $\gamma = 1.0e-3$ and $C = 1.0e3$. In the case of larger sample sizes, other approaches such as SVM trained with Stochastic Gradient Descent (SGD) may be more desirable in terms of computational efficiency [216].

	Blackman			**Hamming**			**Hann**		
	8192	16384	32768	8192	16384	32768	8192	16384	32768
Accuracy	93%	93%	94%	92%	95%	89%	94%	96%	95%
Time (ms)	507	581	740	506	563	646	524	565	655

Table 5.1: Comparison of STFT Parameters
The Hann window with FFT length of 16384 is the most optimal in terms of accuracy but also efficiency in computing the movement fingerprint

Choice of STFT Parameters

Before evaluating the attack, the first step is to evaluate a choice of the parameters used for the STFT step in feature extraction. Specifically,

the FFT length and STFT windowing function were looked at. Given that the RF movement samples are recorded at 2MHz sampling rate, the signals are recorded at 2,000,000 samples per second. According to the Nyquist-Shannon Sampling Theorem [217], this means the signals can contain frequency content up to 1MHz. Given that the STFT provides time-localised frequency content, there is to be a trade-off between temporal and frequency resolution. Simply, a narrow window results in better temporal resolution but poorer frequency resolution, and vice-versa. Given that a discrimination between both what movements are being carried out, as well as full operational workflows is required, it is important to ensure a balance between both temporal and frequency resolutions to allow for success in both cases. Aside from the accuracy as a choice of this first parameter, another consideration is the time taken to compute it. As with most signal processing applications, a relatively fast computation time is desirable, and inherently a longer FFT takes more time to compute. Finally, the last determining factor for the choice of STFT is the windowing function. The use of the (sliding) window function allows for the overlapping of disjointed parts of the input signal. This aims to decrease the amount of spectral leakage and minimise effects such as rippling, by determining the amplitude of side lobes to distribute spectral leakage. A typical window suitable to many applications is the Hann window, which is a form of generalised cosine window, with other popular choices including Hamming and Blackman windows [218]. For a choice of suitable windowing functions, the choice included the Hann, Hamming and Blackman windows – three generalised cosine windows – due to demonstrable success in pulse shaping/filtering [219] and many other applications. Interestingly, given a pre-computed window (as is pos-

sible with the relatively constant cadence across movement samples), the window function should not have any impact on computation time, but this is evaluated for completeness.

In Table 5.1, the accuracy of a baseline set of results (distance of 1 and lowest speed of 12.5mm/s) can be with varying window functions, FFT lengths and the time taken to compute the STFT averaged over 100 runs. With respect to the overall accuracy corresponding to window choice, it is clear the Hann and Hamming windows both outperformed the Blackman window, with the Hann window having slightly higher accuracy overall. Notably, the accuracy increases as the FFT length increases due to higher spectral resolution. Unfortunately, however, this trades off with a longer computation time. Because of this, a key consideration is the computation time for movement fingerprints. With respect to the computation time, the most reasonable window and FFT length combination was a Hann window with an FFT length of 16384. While the window function has no measurable impact on computation time assuming a pre-computed window, as was the case for this attack, erroneous times are accounted for due to measurement noise. In this case, more bespoke devices such as a GPU or FPGA will speed up processing, which given more data may end up providing higher accuracy for robot movement fingerprints. In either case, with regard to FFT length, a longer window requires a more expensive FFT and ultimately results in a non-linear increase of computation time with length. For this attack, a Hann window with FFT length of 16384 is the best choice, as a compromise of 2ms computation time is reasonable for the accuracy increase in comparison with the same FFT length and a Hamming window.

5.3 Evaluation

As per the research questions listed above, the evaluation of this attack and related results will be set out in that order.

5.3.1 Individual Movement Fingerprints

As per the first research question (R_1), the first step was to determine whether the proposed insider attacker could fingerprint *individual* robot movements via unintentional RF emissions. Simply, this will formulate a baseline set of results, consisting of a baseline (smallest) distance of 1mm and speed of 12.5mm/s and RF antenna situated at the base of the robot, to which other parameters such as movement distance can be compared against. As shown in Table 5.2, an average accuracy of 96% can be observed for the baseline ($D = 1$). While most movements show relatively consistent accuracy of at least 90%, the XYZ movement was the lowest among them with 89% accuracy where a small set of samples are mistaken for YZ (Appendix B.1). Interestingly, the Y and Z movements show the most success in terms of fingerprinting accuracy. Overall, it is clear that Y-involved movements show good precision and recall, unlike the traffic analysis side channel where Y-involved movements were among the worst fingerprinted. Notably, the movement permutations (i.e. YZ, XYZ) have the lowest precision and recalls. This may be due to the use of multiple motors resulting in a mixing of similar frequencies that are unintentionally emitted and thus, may be interfering with one another causing a lack in varation among discriminating features in the signal.

D = Distance (mm), P = Precision, R = Recall												
	D=1		D=2		D=5		D=10		D=25		D=50	
	P	R	P	R	P	R	P	R	P	R	P	R
X	91%	97%	90%	100%	100%	90%	100%	100%	100%	100%	100%	100%
Y	100%	100%	100%	92%	90%	90%	94%	83%	100%	85%	94%	85%
Z	100%	100%	91%	100%	100%	100%	100%	100%	100%	100%	100%	100%
XY	96%	100%	92%	100%	68%	100%	92%	100%	94%	100%	88%	96%
XZ	100%	96%	94%	94%	93%	87%	100%	100%	100%	83%	100%	94%
YZ	94%	91%	65%	88%	69%	55%	87%	91%	86%	100%	92%	96%
XYZ	89%	84%	100%	38%	56%	56%	90%	90%	100%	100%	96%	100%
Accuracy	96%		88%		81%		94%		96%		95%	

Table 5.2: Impact of Movement Distance on Classification Accuracy
Movement distance provides more fine-grained information leakage and can be fingerprinted with similar accuracy to the baseline. Slightly increased distances from the baseline show reduced accuracy, while larger distances show similar accuracy to the baseline

5.3.2 Impact of Movement Distance on Fingerprinting

The second question (R_2) concerns a higher level of granularity for movement fingerprints. Specifically, can an adversary infer how far or how fast a movement is being carried out? The first of these two parameters that is evaluated is the distance of movement and how it impacts classification accuracy. The results of this parameter can be seen in Table 5.2. An overview of these results suggest that fingerprinting decreases as the distance increases by only a few millimetres, but remains at a similar level as the baseline for larger distances. At 2 distance units, there is a decrease in precision for most movements, with the exception of Z movement. The accuracy compared to the baseline drops by around 8%. Most notably, the YZ movement has a significant drop in precision, due to some XYZ samples incorrectly predicted as YZ movements (Appendix B.2). Ultimately, the lowered accuracy may be due to lowered variation among principle components between 1 and 2 distance units. At 5 distance units, the accuracy drops by a similar amount as that between 1 and 2 distance

S = Speed (mm/s), P = Precision, R = Recall										
	S=12.5		S=25		S=50		S=75		S=100	
	P	R	P	R	P	R	P	R	P	R
X	91%	97%	100%	100%	90%	100%	96%	100%	100%	100%
Y	100%	100%	79%	90%	70%	75%	67%	65%	65%	35%
Z	100%	100%	94%	100%	92%	100%	93%	96%	87%	96%
XY	96%	100%	65%	68%	64%	64%	65%	68%	46%	46%
XZ	100%	96%	97%	100%	100%	94%	100%	100%	97%	100%
YZ	94%	91%	73%	75%	86%	61%	55%	59%	40%	45%
XYZ	89%	84%	83%	59%	69%	76%	55%	46%	45%	56%
Accuracy	96%		87%		82%		76%		68%	

Table 5.3: Impact of Movement Speed on Classification Accuracy
Movement speed provides lowered accuracy as the speed increases for movement fingerprints, particularly among Y-based movements, which may be linked to the physical infrastructure

units, with permutations of axis movements showing further reductions in precision and recall, aside from the XZ movement which decreases only slightly compared to 2 distance units. The YZ movement in this case is incorrectly predicted as other Y-involved movements, with most being the YZ and XYZ movements. At 10 distance units, perfect precision is observed for X, Z and XZ movements. The Y movment has lowered precision but perfect recall, and other movements show improved accuracy compared to 5 distance units. At 25 distance units, there is perfect precision and recall for most movements, however, both precision and recall is lower for Y-involved movements. Finally, at 50 distance units, the precision increases back to 100% for XZ with recall also improving by around 10%. The precision and recall for Y-involved movements improves compared to 25 units, with the exception of the XY movements with slightly lowered precision.

5.3.3 Impact of Movement Speed on Fingerprinting

As well as movement distance, the speed at which the movement is being carried out may also provide a higher level of granularity to

movement fingerprints. The results for the speed parameter can be seen in Table 5.3. Overall, it is clear that as the speed of movement increases, the variation in the RF feature set reduces resulting in lowered classification accuracy. Better fingerprinting accuracy is observed for the X and XZ movement in the speed parameter compared to the distance parameter, with the Z movement remaining above 90% accuracy but with lowered precision and recall compared to the baseline ($S = 12.5$) as speed increases. Taking all movements into account, the precision for the Y-involved movements are among the poorest as the speed increases, with many incorrectly predicted as either Y or YZ movements (Appendix B.3) perhaps due to a lack in variation between them. This may be due to the design of the uARM robot in which Y movements making primary use of the base motor (with the X and Z axes primarily using the right and left motors as their respective primary motors). The distance parameter is more accurately fingerprintable than the speed parameter, and thus would be more useful to track. While speed may be a useful candidate still, keeping track of the distance is a more reliable metric in most cases. For example, the speed at which an operator performs an operation can vary depending on the context or object being moved, but the distance may be relatively consistent for these specific avenues. In an industrial setting, for fingerprinting alone speed may not be as useful, however in the case of an audit in which one might wish to determine whether the robot was behaving in an erratic fashion, speed data might prove useful through continuous monitoring.

A = Antenna Distance (cm), P = Precision, R = Recall								
	A=0		A=25		A=50		A=100	
	P	R	P	R	P	R	P	R
X	91%	97%	93%	100%	77%	71%	76%	51%
Y	100%	100%	98%	98%	70%	76%	42%	36%
Z	100%	100%	98%	93%	91%	94%	38%	39%
XY	96%	100%	84%	97%	71%	76%	47%	53%
XZ	100%	96%	95%	72%	55%	72%	43%	52%
YZ	94%	91%	71%	83%	50%	38%	45%	50%
XYZ	89%	84%	79%	59%	64%	58%	31%	33%
Accuracy	96%		88%		69%		45%	

Table 5.4: Impact of Antenna Distance on Classification Accuracy The further the antenna is situated away from the robot, the lower the classification accuracy. This suggests that useful frequency content captured by the antenna drops as distance increases, related to the inverse-square law which states that as distance doubles, the power of the RF signal is four times less. This is likely due to the scale of the robot

5.3.4 Impact of Antenna Distance on Fingerprinting

The next experiment in this evaluation looks at whether the distance the RF receiver antenna is placed away from the robot under attack has an impact on fingerprinting accuracy. Naturally, as defined by the inverse-square law [172, 192], the power (intensity) of RF waves is inversely proportional to the square of the distance from an emitting source. Simply, as the distance increases, the intensity of unintentional RF emanations will also decrease. If the distance doubles, the intensity is four times less. In this experiment, given the scale of the robot, the initial distances tested range from the base of the robot (0cm) to 100cm away from the robot. The reason for these distances is simple – if the intensity of unintentional RF emanations significantly reduces the accuracy of fingerprinting robot movements at the worst case distance, then it demonstrates that this is due to the scale of the robot used in this evaluation. The results for this experiment can be seen in Table 5.4. First, increasing the antenna distance by around

25cm leads to a slight increase in precision and recall for th X movement, but a reduction in precision and recall for all other movements by around 8% on average. At 50cm away, the intensity of the signal is weaker and results in further drops to precision and recall across all movements by around 20%. Interestingly, the Z movement drops only slightly with this increase in antenna distance. Finally, at 1 meter away, the overall accuracy drops by around a factor of 2 compared to the baseline. Ultimately, this demonstrates that due to the scale of the robot, any unintentional radio frequencies that are emitted during its operation are not strong enough to be captured at a distance. While other antennas were considered, many antennas were unable to capture such low frequencies, as well as financial limitations being another aspect. A spectrum analyser could be used but an attacker may not see this as an ideal *stealthy* candidate. As a point of future work, however, looking into larger scale robots such as large robotic arms used in manufacturing may result in a more intense signal that can be captured further away. The confusion matrices for this experiment can be seen in Appendix B.4.

	Recovery Rate		
Operation	1	2	3
Push	100%	100%	100%
Pull	100%	100%	100%
Pick-and-Place	100%	100%	100%
Packing	100%	57%	57%
Accuracy	100%	88%	88%

Table 5.5: Workflow Reconstruction Results
Common warehousing workflows can be reconstructed in their entirety with much higher accuracy, compared to an approach which involves pattern-matching using individual movement fingerprints

5.3.5 Workflow Reconstruction

Aside from exploring the efficacy of the attack on individual and combinations of movements, which can be used in pattern-based reconstruction, it is also interesting to determine whether higher-level warehousing workflows can be reconstructed via the RF side channel. For this experiment, the second dataset containing warehousing workflow samples was used, which is detailed in 5.2.1. The results for the experiment on reconstructing warehousing workflows can be seen in Table 5.5. In this experiment, three different sets of each workflow were captured and analysed using the same attack strategy as individual movements. Interestingly, looking at the cumulative explained variance of the principal components for workflows, a very similar amount of variance as with individual movements can also be captured for entire workflows with the same number of principal components, and thus the attack parameters remain the same. Using the same SVM parameters, the first set of workflows achieves perfect reconstruction accuracy. While this is a notable result, the second and third sets of these workflows achieves similar results, with the packing operation being the exception with 57% accuracy in both cases. In the case of the second set, some of the pull operations are incorrectly predicted as packing, and for the third set, some samples of pick and place are incorrectly predicted as packing (Appendix B.5). This may be due to the fact that the packing operation may have similarities to these movements in the sets explored resulting in the lowered accuracy. Overall, it is clear that using this attack approach, an adversary can very successfully reconstruct entire operational workflows. Through the use of continuous monitoring, entire daily operations and their performance (i.e. by also capturing timing information) can be leaked

to competitors for a potentially malicious advantage.

5.4 Discussion

The proposed side channel attack using radio frequency (RF) demonstrates a feasible approach for a insider attacker, such as a malicious technician/operator, to place an easily disguised and economical antenna on the factory floor. By doing so, they can fingerprint robot movements and reconstruct entire warehousing workflows with very high accuracy. Furthermore, the evaluation shows that such an attacker can also infer more fine-grained information such as the speed or distance in which a movement is being carried out.

5.4.1 Defences

Given that the radio frequencies that are captured in this attack are emitted from an unintentional radiator source, in this case the robot, the radiated fields require a form of mitigation (suppression) to prevent information leakage. To recap, the robot used in this study is likely to emit unintentional RF from its microprocessor and stepper motors, with the majority from the microprocessor. A best defence here comes in the form of physical layer security measures such as shielding critical portions of the microprocessor layout or robot enclosure. Shielding against electromagnetic fields (EMF) such as RF makes use of a barrier made of conductive or magnetic materials to isolate minimise interference but also act as a sort of Faraday cage. The materials typically used include copper, silver or brass, with copper being the most common for RF shielding. In the case of the robot used, the enclosure is made of aluminium. While this is not

as conductive as copper (60%), it is usually a second choice due to other properties such as electrical conductivity, strength-to-weight ratio, cost and malleability. With respect to the enclosure, a suitable defence strategy to explore is to evaluate whether a thicker aluminium casing would provide better protection against the proposed RF side channel attack. Furthermore, it would also be interesting to observe whether other casing materials such as copper or brass might perform as well or better with their innate contrasts in protective properties (i.e. conductance).

5.4.2 Impact

While the impact of this attack is demonstrable from a business perspective, in the form of operational compromise of robotic workflows that could leak sensitive business information it is important to look at the impact of the attack in other areas, including government/international specification, regulations and international standards.

The first impact to review is government specification. In the United States of America, the government and NATO specification TEMPEST is used to cover methods for eavesdropping on and protecting (shielding) against information leakages from unintentional emanations such as RF or acoustic [220,221]. While many specifics of TEMPEST are classified, in the public domain three levels of protection requirements are set out: NATO SDIP-27 LEVEL A, B and C. Level A is the strictest standard which assumes the attacker has almost immediate access to the environment or devices. Level B assumes the attacker cannot be within 20 metres and is more relaxed, and Level C assumes a distance of 100 metres. Unfortunately, TEMPEST mainly

addresses nation-state level equipment and facilities. The guidance states that in general, devices such as these robots are typically not qualified under TEMPEST as most of these devices, including commercial off-the-shelf components of robots which are assumed to conform to Level C without any modification. Further, it is not clear on specific key requirements of shielding against unintentional emanations. Given the impact on businesses in any respect, their confidentiality is key – particularly against those which they are in conflict (competition) with – and thus, the potential of insider threat should ultimately be a cause to improve the standards of robot protection in non-military settings.

The next impact of this attack pertains to regulation and regulatory compliance. Guidance from regulators, such as Ofcom [222] in the UK, issue rules for compliance for all uses of RF, whether emanations occur under normal conditions or from unintentional radiators such as the context of this work. However, such guidance does not cover robotics systems but only a subset of typical components (i.e. power input or cables). In the case of Ofcom, the regulation only details where an EMF record is not required but does not include where robotics systems or unintentional RF emanations apply. Clearly some modifications and additions to existing regulation is needed to cover attacks like this on robotic systems. As well as existing regulation, one key question that may arise is how can one audit a robot? While continuous monitoring and audits of software and hardware components can provide some guarantees, the risk of malware or invalid device calibration that disrupts operational accuracy also needs attention. Interestingly, this is observed from the high recovery rates of industrial warehousing workflows as detailed in the evaluation. From

this, the use of the RF side channel could potentially act as a defence that can provide information to auditing procedures. By monitoring typically correct robot workflows over time, any drift in accuracy from either a malicious or unintentional *attack* vector could be recognised by the system, for example through the use of LSTM networks and RF time-series information. This is a point of future work to be explored. The final considerations pertain to international standards. The key international standard that applies to the impact of this attack is CISPR 11 [223] for governing EMF emissions from industrial, scientific or medical (ISM) equipment, among others, which can use the ISM license free bands like 2.4 GHz. The ISM bands are defined by international telecommunication union (ITU) radio regulations, which have a variety of allocated ranges within the band of 6.76MHz–256GHz. In the context of this attack, the smaller robot used falls outside of ISM bands where such standards and regulation typically apply. However, for larger industrial robots, an evaluation of this attack would be worth observing to truly understand the impact in this case. Interestingly, an investigation into the RF emissions related the size and general load/power requirements of different robotic systems may also reveal that some may also fall out of the ISM bands and may in future provoke a discussion on updating existing standards.

5.4.3 Limitations

One limitation of this work is the robot used. In terms of direct replicability for a real-world industrial robot arm, this robot is much smaller than one typically seen on the factory floor. While the attack may not provide the same level of movement inference in this case as with the robot used in this study, a larger robot may employ more

motors that require more power to operate. Given that the motors and integrated circuits still emit unintentional RF from the nature of their operation [28], the attack would simply require extra exploratory analysis in terms of the frequency range of the RF emissions to tune the parameters. As well as this, it would be interesting to perform an exploratory analysis of unintentional RF emissions from a range of different robotic systems involved in various different workflows. Interestingly, while this antenna is noticeable if placed behind the smaller robot used in this study, if the same antenna could be used to mount the same attack on a larger robot then it would result in a much more stealthy attack.

In addition to the first limitation, a second limitation is that only one robot was used in this study. While this work showcases that a single arm robot can indeed be fingerprinted via the RF side channel, in industrial settings, for example, there may be more than one robot employed in the same workspace. Thus, any unintentional RF emissions that arise from other robots may act as a form of noise mask, potentially hindering an attack on a single robot. As well as this noise from other robots, some may also question whether other sources of radio frequency may also be picked up by the RF antenna, such as Wi-Fi or unintentional RF from other computing devices. While this has been thought of, Wi-Fi in particular falls out of the range of frequencies captured in this attack, and other devices do not spike at the same intensity or frequencies as the robot. The feature extraction process allows for the identification of significance in peaks and movement characteristics (i.e. cadence) which is successful even without noise reduction techniques. In any case, a point of future work examining noise reduction techniques would be interesting to see potential

improvements in the attack – which would particularly useful in the context of multiple robots operating in the same space. In addition to this, not only would evaluating this attack on various types (i.e. brands or style) of robots be of use to demonstrate wider application, but also the impact of mulitple robots in the same space.

A third limitation one can consider is a disconnect between the tool equipped at the end-effector and the motion being carried out. The robot arm used in this study is equipped with an integrated pump with a maximum pressure of 33kPa and maximum lifting weight of 1kg. The goal with the use of the tool was to determine whether an attacker could identify the contents (by weight inference) of packages being lifted. Unfortunately, there was no measurable unintentional RF emissions between weights being lifted, nor to distinguish whether the pump was on or off. For a larger robot however, a pump end-effector with a larger lifting weight that requires more power may change this result. As well as this, it would be interesting to see the impact of additional power requirements for lifting weights with different tools (i.e. grippers) has on RF emissions.

5.5 Related Work

While there have not been any side channel literature pertaining directly to robotics systems, the use of side channels have shown success relating to components of robotics systems, as well as similar architectures such as 3D printers. There are many devices which emanate unintentional RF, including microprocessors and motors, among others. Graham et al. [224] demonstrate that RF emanations can be identified by correlating RF emissions with bit flips that produce de-

tectable electrical pulses. This is analogous to the switching activities in transistors as pointed out by Cobb et al. [28]. In similar work, deep learning approaches have shown success operating on raw waveforms [225,226], such as using convolutional neural networks operating on time series data [227,228] or residual neural networks [229], to fingerprint IoT devices and processes running on them. However, while these approaches are successful, this attack requires a much smaller sample set to show similar accuracy – a benefit of many SVM-based approaches.

Aside from the radio-frequency side channel, other side channels such as power and acoustics also show some success. For example, Sami et al. [162] describe an attack via the acoustic side channel that can extract sound traces from the vibrations reflected to lidar sensors. Several authors [161,230] propose an acoustic side channel attack on 3D printers wherein acoustic emanations are used to reconstruct G-code used by 3D printers which may correspond to potentially confidential (patented) designs. Related to this, Song et al. [160] also describe a similar end goal but enhancing the acoustic side channel using the magnetic side channel by exploiting the conductivity of a stepper motor. Compared to this attack, the extraction of G-code corresponds to Arduino-based 3D printers. Given that this robot is also operated by an Arduino, this attack focuses solely on the movement of the robot arm and thus reconstruction of G-code to then compromise operational confidentiality is an unnecessary extra step. While earlier approaches make use of regression models, more recent work make use of neural networks that require a large labelled sample set to which this approach provides better opportunities to an passive, insider adversary. As well as this, given that individual movements can also be

reconstructed, pattern matching individual or permutations of these movements could also lead to the leakage of confidential intellectual property.

5.6 Summary

In summary, it is clear that it is possible for a passive adversary to make use of other side channels for robotic systems, aside from those present in the cyber domain. This attack showcases that even in the physical domain, compromise of operational confidentiality can still be compromised in a stealthy (hard to notice) manner with much higher success than both the traffic analysis and acoustic side channels, as described in Chapters 3 and 4 respectively. With the success of the attack, as well as a discussion into the inadequacies of current regulation and specifications, the implications of the presented results showcase the need for a full review of existing standards. Unfortunately, while the same level of granularity of movement inference can be gathered from this attack (i.e. speed and distance), the contents (weights) of objects moved could not be inferred from the RF side channel. While this thesis showcases three passive side channels, for physical domain side channels there is now a clear set of open challenges in the space of defences pertaining to these attacks, which remain as a point for future work.

6 — Calibration and Robotic Systems

Everything can be assigned a value. For example, an age old riddle by John Marciano questions *"What weighs more, a pound of feathers or a pound of gold?"*. Some may say gold because it is heavier, others say they are both a pound and thus weigh the same. A pound of either *must* be the same weight. Measurements like this are used in our day-to-day lives and we inherently trust them. But, how do we know that these values are in fact correct? Can we simply just take every scale, thermometer or other measuring device at face value? Can we trust that they will always give the same readings if they measure the same respective quantity?

There are many activities which depend on having accurate and reliable measurements, with a tolerable margin of error (uncertainty). For example, in industrial manufacturing plants the dimensions of components must be specifically defined and manufactured to ensure that outputs as a whole meet the strict specifications in which they are to be used. In the pharmaceutical sector, manufactured medicines must only contain a strictly specified quantity and quality of the substances it is made up of. In these areas, any slight hindrance to the accuracy and reliability of measurements can lead to disastrous consequences. For example, there have been cases of poorly manufactured medicines shipped out to consumers which have ultimately lead to healthcare failures such as antibiotic resistant bacteria, disease spreading or even death [231–233]. So how can these be disastrous consequences be

prevented at the measurement level?

The science of measurement, Metrology, documents everything from the design of measurements, to carrying out and analysing measurements and calculating relative measurement uncertainties. This spans a wide array of areas, from the organisation and development of measurement standards and maintaining them at the highest level, to ensuring the accuracy of measurements and the adequate functioning of measuring devices at a consumer level (i.e. in production or testing).

In this thesis, the focus is on calibration, a part of metrology which governs a set of operations whose purposes are to detect, report and eliminate measurement errors for devices. The majority of devices are ideally calibrated to ensure that it operates with the highest accuracy and lowest margin of error, ranging from sensor devices (i.e. temperature sensors) to even the cables and resistors embedded within these devices. In the case of a robotic system, if the calibration was to be incorrect or the device required reconfiguration to ensure it conforms to its ideal calibrated state, what would happen? In surgical contexts, for example, even a slight offset in accuracy during a scalpel incision could result in the difference between life and death. Similarly, for industrial robots, if the measurements recorded were incorrect and used by a safety mechanism to prevent injury to human operators nearby (i.e. collision detection), the result may be just as catastrophic. Ultimately, before attacks in both the physical and cyber domains can be considered, calibration is a key factor which underpins operational safety of robotic systems.

6.1 What is Calibration?

"I have been struck again and again by how important measurement is to improving the human condition."

– Bill Gates

Calibration is a set of operations which govern accurate and reliable measurements. Simply put, in the context of IoT components in robots, we can consider the process of calibration as a comparison between output measurements from a component (e.g. an infrared thermometer sensor) and another from a more accurate reference device.

The output of the calibration process for some Device Under Test (DUT) is a calibration report. This report, mainly used to report the results of the calibration process, details essential information, including (but not limited to):

- date(s) of calibration,
- environmental conditions at the time of calibration,
- calibration standards adhered to,
- organisation-related information (i.e. contact information), and
- **a traceability statement**

Given that a device after calibration can (in-effect) be trusted for a given time until its next calibration, a consumer would then simply trust the (accredited) calibration provider that the process was carried out to standard [234, 235]. However, aside from accreditation and other factors which related to consumer trust, how do we know that the reference device used in a device's calibration is more accurate? To answer this, the (parent) reference device is itself calibrated by

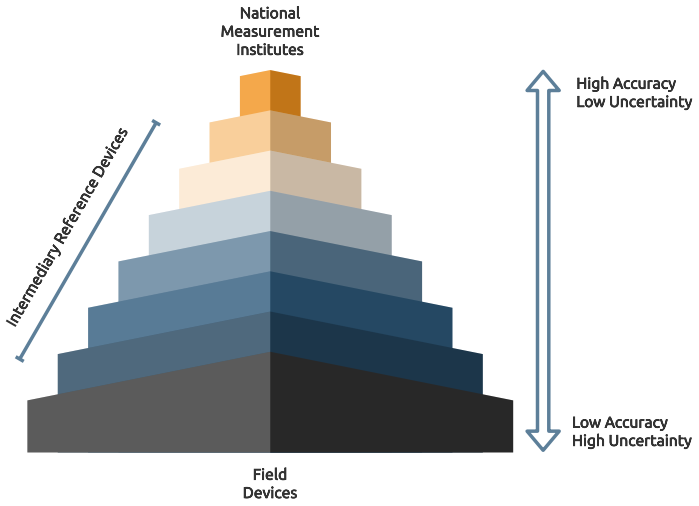


Figure 6.1: High-Level Traceability Hierarchy
Calibration moves from the SI units maintained at National Measurement Institutes of highest accuracy and lowest error margins, to intermediate units and ultimately robot components at the field level

some other more accurate reference, and so forth. The limit is set at national standards where National Measurement Institutes (NMIs), such as NPL in the United Kingdom and NIST in the United States, hold a master-level device which is calibrated against the SI units. Specifically, they coincide with governing bodies at an international level, assuring that their calibrations compare with each other. Finally, these international-level institutes base their measurements on realisations of the International System of Units (SI units) [236]. This chain of calibration is referred to as a traceability chain [235] (Figure 6.1).

6.1.1 Measurement Uncertainty

Any results obtained by a measuring device should provide some degree of confidence to its consumer(s). This confidence is quantified by measurement uncertainties associated with the result. However, alone these uncertainties are not enough to ensure the credibility and con-

fidence one can have in measurement results. Instead, this credibility is provided by the traceability of the measurement results, which also requires an accompanying statement of its respective measurement uncertainties. Through an expression of a result's measurement uncertainty, the reliability of the result can be *reasonably* confirmed and maintained.

The ISO Guide to the expression of Uncertainty in Measurement (GUM) [237] defines measurement uncertainty as variable relative to a measurement result that characterises the dispersion of possible error sources which could be reasonably attributed to the measurement result. Measurement uncertainties are present, as realistically measurements are never made under perfect conditions. There can be several sources of uncertainty present in calibration, which can come from: the measuring instrument (i.e. due to drift, wear-and-tear, noise, etc.); the item being measured; the environment; and the measurement process itself, among others. These sources can be either random (repetitions produce varying results) or systematic (the same influence affects results for each repetition). In either case, a spread of a set of uncertainty values take the form of a probability distribution. This is typically a normal (Gaussian) or uniform distribution, but in rarer cases may take the form of others such as triangular or M-shaped distributions.

While uncertainties are typically expressed to consumers in a statement consisting of the result and uncertainty figure, such as "Temperature measured was $60°C \pm 1°C$", this is not enough in the context of calibration (i.e. verifying correctness). Instead, a statement of the type of uncertainty (such as standard uncertainty) and the level of confidence – along with how the uncertainty was estimated – is required

to fully express the information so that it is usable. Within calibration reports, this is typically summarised as an **uncertainty budget**. Ultimately, this uncertainty budget depicts a structured approach for calculating measurement uncertainty, which provides a formal record of analysis whilst also satisfying ISO 17025 [235] requirements. Examples of uncertainty budgets can be seen in [238].

6.1.2 Metrological Traceability

Having metrological traceability (a complete, unbroken chain of calibration) can allow one to verify the validity of calibration of a device. This is due to the fact that measurement uncertainty (and ultimately, measurement accuracy) is derived from the process of calibration for each of its parents [239]. The higher a device is in the chain, its measurement uncertainty is smaller and its accuracy, higher (and vice-versa).

So, what is the importance of metrological traceability? First, without verifiable traceability, a calibration provider can effectively claim anything they want in a calibration report that may be taken at face value. Further, consumers may fall victim to fraud in contexts where a calibration provider does not include a device's traceability. For example, consider a hospital, which employs surgical robots, that makes use of a third-party calibration provider to calibrate the robot's components (e.g. sensors). A lack of traceability for one or more measuring instruments means that one cannot verify the measurement uncertainties, nor trace the calibration back to the established primary standard. This leaves any measurements suspect to unknown error and drastically decreases the trust one can have in the robot. As a patient under the knife, even slight inaccuracies of sensor measurements dur-

ing the operation that are not uncovered could result in the difference between life and death. As an organisation who employs such third-party facilities and owns the robots used, the resulting liabilities could be catastrophic.

Unfortunately, not all measurements are traceable. Typically, this is due to some device in a traceability chain being *missing*, for example if it needs to be recalibrated as part of an annual cycle or if it has been decommissioned. If this point in the chain succumbs to such conditions, all measurements produced by devices below that point have no traceability and are suspect to unknown error (questionable accuracy). Ultimately, in order for a *device's measurement* to be traceable, it must satisfy **all** of the following conditions [235].

Documented Calibration

Every level in a traceability chain must be documented and no point in the chain should be missing (complete and unbroken). As a minimum, this means documenting the results of a device's calibration in its calibration report. This allows one to audit whether a device's calibration was carried out correctly, adhering to standards (e.g. ISO 17025 [235]) and in accordance within the calibration provider's quality specifications. In current practices, these reports are stored internally within the calibration provider's storage infrastructure. For verifying metrological traceability, these are requested by an authorised person such as the device operator. Ideally, calibration reports for a device (and any previous reports) should be held in some immutable record. Finally, within a calibration report, a traceability statement must be supplied that is typically trusted at face value. An example statement may appear as follows:

"This calibration is traceable to the International System of Units (SI), through National Metrology Institutes (NPL, NIST, etc.), ratiometric techniques, or natural physical constants."

Timely Calibration

Calibration of a device must be carried out at regular intervals. Re-calibration is important as what may once have been accurate measurements can drift over time (i.e. through wear-and-tear). In the current state-of-the-art of calibration, this is typically an annual cycle. Ultimately, when the calibration expires and is no longer valid, the traceability chain can also be considered as expired. This is because any child devices to which this device is a reference in its calibration, can now not be guaranteed to produce reliable measurements and ultimately do not satisfy calibration traceability requirements [235]. Subsequently, in all cases, a note of the validity period (date of calibration and expiry) should also be included within a device's calibration report.

Stated Measurement Uncertainty

It is vital that every level in a traceability chain has any measurment uncertainties documented. The reason this is an important requirement is that one may end up calibrating an already accurate device with a reference that is less accurate, or vice-versa. In another case, if the calibration procedure being carried out results in large or varying uncertainties, the calibration is poor and not traceable. Ultimately, if there are no stated measurement uncertainties, one cannot claim a measurement is traceable.

Reference Indicator

It is important to also realise what reference is being used in a device's calibration. This is typically also detailed in the device's calibration report along with the reference's traceability.

Calibration via Accreditation

Finally, it is important to ensure calibration was carried out by trained and competent technicians, who are trusted to conduct calibration processes by an accredited calibration provider. In some cases, such as if calibration is carried out internally at a single facility, calibration-related activities are not accredited and thus, cannot produce an accredited calibration report. In most cases, it may not be reasonable or necessary to obtain accreditation. Specifically, if internal activities follow a quality system that adheres to a known quality standard, such as the ISO 9001 quality standard [234]. However, in regulated industries or where critical measurements are required, such as in automotive manufacturing or surgical robotics, it is important to ensure calibration is carried out properly and to the highest quality. If calibration is carried out in-house, it is important to conduct regular internal audits to ensure that the internal laboratory is traceable (i.e. proper quality assurance system(s), documented traceability, correct and valid uncertainty calculation, etc.). Ultimately, in such cases, it may be better to make use of a trusted, accredited external calibration provider.

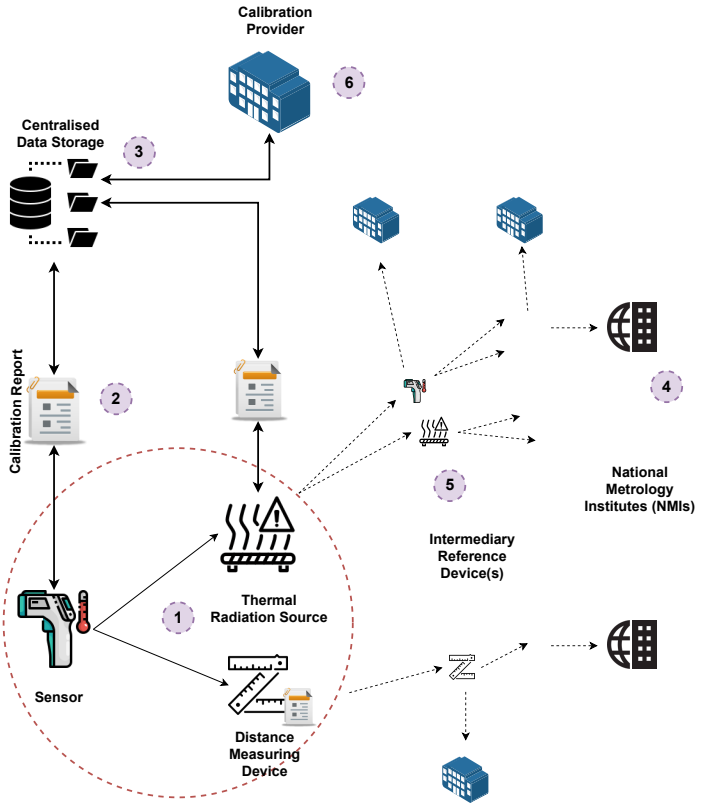


Figure 6.2: Calibration Ecosystem for a Single Robot Component

6.2 The Calibration Ecosystem

The calibration ecosystem involves a number of actors, including: system and device operators at the field level; original equipment manufacturers (OEMs); (third-party) calibration providers; system and provider auditing organisations and NMIs. Each party plays a core responsibility to ensuring the calibration lifecycle can continue. Furthermore, a subset of these interacting parties may share an adversarial relationship. For example, two third-party calibration providers may compete against each other.

Taking a robotics context, for example a surgical robot, the actors involved would include: the surgeons operating the robot and nursing staff in the field; system technicians that maintain the robot and managerial staff within the hospital. At the calibration level related

to the robot, there may exist one or more calibration technicians that may maintain the calibration of the robot in-house. Given that a robot consists of many components, whether that be sensing equipment (e.g. infrared thermometer sensors) or even the cables used to connect various components together, each of these components require calibration and each will likely make use of different reference (parent) devices. While it is unlikely that such a robot will be recalibrated to ensure its optimal accuracy before every surgery, calibration parameters may be retuned or tested to ensure that everything is within tolerance (e.g. actuators are level and sensors are white-balanced) [51]. However, there are factors which come into play that would require a recalibration of one or more components of the entire robotic system outwidth a typical annual cycle of recalibration. To explain this, an overview of a subset of the calibration ecosystem pertaining to a single robot component – an infrared thermometer sensor – can be seen in Figure 6.2. Such a tool would likely be used to ensure the correct temperature for a tool used in electrocauterization of a wound. Ultimately, however, while the lifecycle is described for a single robot component, a typical robot may consist of hundreds of calibrated components and the scale and complexity of a robot's calibration lifecycle can be envisioned.

6.2.1 The Calibration Lifecycle

In order to calibrate an infrared thermometer sensor, a calibration technician requires: a thermal radiation source, a transfer standard of higher accuracy used for comparison in calibration, an ambient temperature thermometer, and a distance measuring device [240]. In some cases, for example where an aperture may be part of a device's calibration, additional equipment may also be required. To describe the

calibration lifecycle, we will assume that the calibration of this measurement device is conducted at some third-party calibration provider, which is also responsible for the other reference and transfer standards used in the sensor's calibration. Upon review, we note there are three key stages in the calibration lifecycle: *initial calibration*, *traceability verification* and *recalibration*.

Initial Calibration

The initial calibration of a measuring device can be viewed as its induction, or *birth*, into the calibration ecosystem and commences the start of its calibration lifecycle. For all measuring devices, there is always an initial calibration step. After a device has been manufactured at some OEM, there are three common ways in which the initial calibration is carried out, which are: in-house calibration (i.e. by the manufacturer before being shipped to consumers, or at the deployment level); calibration at a third-party intermediary calibration provider; or calibration at an NMI. Typically, the latter is only made use of by third-party calibration providers who calibrate their own internal *master-level* reference devices with NMIs, which can then be used to calibrate consumer devices in-house. During calibration, the calibration technician who is carrying out the process for the certifying provider is responsible and entrusted with carrying it out correctly and to standards [234, 235]. Upon successful calibration, a calibration report is produced and signed by the technician, which details the uncertainty budget, organisation and device information, etc. related to the calibration process. Further, the calibration report will specify when the next calibration (recalibration) is required – typically after a year has passed, but otherwise within the time period agreed upon

between the provider and device owner.

Traceability Verification

During initial calibration and before recalibrations, the traceability of a measurement produced by the device during the process needs to be verified. Specifically, the technician will ensure the criteria stated in Section 6.1.2 is adhered to.

Recalibration

The third key stage in a device's calibration lifecycle is recalibration. In some cases, recalibration is referred to as the process of adjustment, where a device under test (before its specified recalibration period has been reached) may be adjusted dependent on potential changes in output to ensure outputs agree with values from the applied standard within a specified tolerance (accuracy) range. The process of recalibration is required where one or more of the following criteria are true:

- The recalibration period (calibration expiry) detailed on a device's calibration report has been reached or exceeded;
- A critical measurement is taken;
- The device has been repaired or modified;
- The device has been moved (i.e. changes in ambient temperature can potentially affect the output of a thermometer);
- The device has been exposed to a critical event, such as shock, vibration or physical damage which is thought to may have an impact on the integrity of its calibration;

- The accuracy (or uncertainty) of device output has noticeably degraded or drifted before the expiry period; or

- Any parent (reference) unit in the device's traceability chain to national standards is declared to have invalid calibration or has been mmrecalibrated.

6.3 Calibration in the Digital Age

The metrological activity of realising the modern International System of Units (SI), which was resolved in the 1960s and redefined in 2019, was based on measuring devices that were not Internet-connected. Global communication networks were only proposed at around the same time. As technology and the Internet evolved, among other computer networks, so did the calibration ecosystem. However, given the nature of computing during this period, existing calibration processes followed manual, paper-based approaches, with only some avenues moving towards a digital approach more recently (such as calibration report storage) to some degree.

Many modern IoT systems, such as industrial and surgical robots, are employed to address requirements for automation, higher accuracy and precision. For example, the use of surgical robots like Robodoc [42] have shown to lead to a decrease in post-operative complications compared to traditional surgery. In the context of autonomous vehicles, a large variety of sensing and measurement devices are required to provide assistance for making highly accurate, real-time decisions. In most cases, safety is paramount (i.e. to provide minimal risk to patients or passengers). However, with these systems now being connected to the Internet, safety now also becomes a concern for

security.

While the security and privacy of IoT systems, which make use of measuring devices, has received a lot of attention with a variety of solutions proposed for different application areas [241–244], the security of the calibration ecosystem which ultimately underpins reliable and correct device operation has received little attention. Recently, there have been incidents in which IoT systems have failed with regard to operational safety, such as autonomous vehicles incorrectly classifying hazards [245, 246], or surgical robots burning patients [10]. Existing literature has shown that from an IoT security standpoint, cases such as attacks on machine learning classifiers using this sensed information or tampering with communications to and from these systems can be detected and defended against [51, 64, 247], but what if catastrophic failures resulting from these attacks, are indeed not a result of attacks that target the system itself? In a case of law, will these be simply be judged as a random mechanical or electrical failure that becomes solely the responsibility of the manufacturer or consumer? Or could the nature of device and system calibration perhaps play a pivotal role in the cause of safety failures?

Often, calibration providers think that simply connecting measurement systems to the Internet is key to solving challenges pertaining to data management, such as increasing granularity, scale and frequency of measurement, as well as ease-of-use. This typically stems from the need to support consumer requirements of higher quality, actionable data that is available to usable, efficient and timely processes. However, simply connecting existing measurement systems to the Internet does not bring any evolution with regard to the fact that Internet-connected devices come with a larger attack surface, and will require

frequent monitoring and mitigation against potential threats/attacks. The current calibration ecosystem assumes that all actors will behave themselves and there is little to no consideration for security. Unfortunately, in a rapidly growing Internet-connected world with modern (IoT) systems, simply *"digitising"* a small subset of processes (such as storage of calibration reports) within the calibration ecosystem is not enough to meet evolving consumer requirements such as efficiency, safety, high accuracy and availability, as well as remaining secure in a constantly evolving threat landscape. It is with considerations and requirements such as these, where several inadequacies with current state-of-the-art calibration practices present themselves. These inadequacies would not only hinder the progression towards digitising the calibration ecosystem, but also leave it vulnerable to threats and attacks from a security standpoint which threatens existing safety claims from correct calibration. Ultimately, the calibration ecosystem will need to evolve, as there will likely be an increase in adversarial pressure when exposed to the cyber domain. The first step towards resilient evolution to a digitised ecosystem is a sound threat model.

6.4 Threat Model

A foundational challenge to high-assurance, safety-critical robotic systems is record-keeping. While the solution space for such a challenge has been investigated [248–250], the challenges surrounding record-keeping which encapsulates calibration activities has been paid little attention. It can be observed that the calibration ecosystem is highly data-centric. Calibration data is the primary and permanent asset that determines what devices can measure, the accuracy and reliabil-

ity of what is being measured and ultimately underpins the correctness of IoT outputs. Calibration processes for different devices, adjustment procedures to ensure devices adhere to correct measurements, calibration-related business practices and even business relationships and possible competitive behaviours between interacting parties in the calibration ecosystem, are all inextricably tied to this data. The need for digitalisation in a data-centric ecosystem comes with increasing reliance and trust with having correct calibration data in a new threat landscape.

What new security concerns arise when a typical factory can be considered to eventually make use of hundreds of robotic system which all employ thousands of sensing devices, and thousands of these factories to share several hundred calibration providers? Will existing calibration processes be able to support the ubiquitous nature of IoT and robotic systems at scale, and manage a large number of interacting parties where some may share an adversarial relationship? A key part of a good threat model is first understanding the inadequacies of the current state-of-the-art and then uncover what threats may arise with progression to a digital environment.

6.4.1 Inadequacies of Calibration in a Digitised Environment

Integrity of Calibration Reports

Given that calibration reports are paper-based, the possibility of integrity compromise is of real concern. A relatively trivial assumption for physical records is that they can be lost or damaged, which means vital information is irretrievable or open to misinterpretation. In the

case of reports that are stored in a digital format, the potential for tampering with the records is also a concern. In the case of an outsider adversary, it is possible that measurement uncertainties and parent reference devices can be modified. As well as this, the potential for forgery of paper-based reports is also of real concern. In either case, this could lead to traceability chains that allow calibration information to flow to the attacker, or cause IoT systems to blindly trust potentially incorrect or unreliable measurements [251]. In the case of an insider adversary, the complete control over calibration records leaves room for providers to *cook-the-books* and modify details about the calibration process for their own benefit, such as in the event of an audit.

Freshness of Calibration Information

Another concern regarding the use of paper-based assets for calibration records is with freshness – how up-to-date the calibration records are. Specifically, aside from integrity concerns which can impact the freshness (i.e. an adversary modifying the date and time of calibration and expiry), maintaining the freshness of calibration records is an entirely manual process that could be left unnoticed. It is up to calibration providers to ensure that devices are recalibrated through notifying system and device owners. Updates to calibration records, and notifications for recalibration are infrequent – typically performed annually based on the common recalibration cycle for a device – which is not suitable for critical IoT environments. Consider a manufacturing line for integrated circuits. If the freshness of reports is not maintained, any problems regarding calibration will not be uncovered until the next point of verification (the annual cycle). By this point,

incorrect measurements used in the manufacturing cycle will be left unnoticed and potentially thousands of units will have been shipped to consumers. In critical IoT environments, it is vital to have the most up-to-date accurate and reliable calibration information available at any time.

Revoking Invalid Calibration

In the current state-of-the-art in calibration, revocation of calibration reports is practically non-existent. When some reference device has reached the point where after years of service even adjustments to the device cannot satisfy a valid recalibration result it is simply decommissioned without notification. Without notification, the rest of the downstream devices calibrated from this point will blindly trust their measurements as being accurate under the assumption this parent reference device in the chain is still in the ecosystem. Furthermore, outside of the scope of decommissioned devices, when a parent device is deemed to have invalid calibration, its calibration report should be revoked and downstream devices notified to state that the accuracy and reliability of measurements are indeed questionable and devices should be recalibrated. In the current state-of-the-art, this type of notification does not always occur and paper-based reports cannot be revoked remotely. The lack of revocation can ultimately compromise the integrity of entire data supply chains.

Availability Compromise

The centralised nature of calibration record-keeping leaves the calibration ecosystem susceptible to availability compromise due to single points of failure at the responsibility of calibration providers. If a

calibration provider's storage infrastructure is the focus of a targeted Denial-of-Service attack (DoS), this could prevent access to calibration reports used in traceability verification. If we consider the need to verify the traceability of measurements produced by components of a surgical robot prior to carrying out emergency surgery, the verification of their traceability would fail as a requirement is a complete, unbroken chain to national standards. This would mean that any measurements produced are unreliable and further use of the robotic system could result in serious harm to patients.

Lack of Transparency

Given that calibration information in the ecosystem flows through multiple providers in different domains, there is an inherent lack of transparency among them. This lack of transparency among providers in traceability chains hinders the ability for consumers to challenge and verify the calibration statuses of devices claimed by providers and verify traceability chains are unbroken/complete and correct. Simply put, a lack of transparency prevents a need for public verifiability at any time, which is a vital requirement for safety-critical IoT systems. Not only may this affect consumers but the lack of transparency may also impact other interacting stakeholders in the calibration ecosystem such as manufacturers, regulators and other calibration agencies, who wish to collaborate efficiently. Notably, this lack of transparency may also stem from the competitive nature between calibration providers. For example, a calibration technician who calibrates devices for one vendor, should not be allowed to do so for another in order to prevent potential leakage of information about devices and organisation processes.

Inefficient Verification

Verifying the traceability of measurements is highly inefficient. These paper-based records are requested from the calibration provider where they are stored, by a subject (verifier) such as a system/device owner. Once authorised this information is provided to the verifier, who must then repeat this process for each of the parent reference devices listed in the device's report, up to national standards to verify a complete, unbroken chain of traceable calibration. Furthermore, another step in this verification is to also ensure measurement uncertainties are indeed correct at each stage, to verify the calibration at each stage was in fact carried out properly. Outside of a robotics contexts, a wait of potentially hours to several days to retrieve this information manually may be considered a reasonable wait. However, in robotics contexts – especially in cases where time is critical – this is far from reasonable. Specifically, consider this same time-inefficient process needed to be used for tens to hundreds of calibrated components in an entire system. The verification time would take too much time, leaving one to question whether the system should simply be allowed to continue carefully while waiting. Looking back at the context of a surgical robot needing to carry out emergency surgery, simply an assumption of extra-vigilant operation is not a guarantee, where even slight offsets caused by incorrect calibration that cannot be verified could mean the difference between life and death for even a simple scalpel incision.

Confidentiality of Calibration

So far, the inadequacies described assume that all calibration information can be viewed at any time without any conditions pertaining

to the access of this information. In the real-world, this is simply not true. Consider a system-level organisation who employs hundreds of robotics systems which make use of calibrated measurement devices. While these devices may be calibrated in-house, in some cases an external third-party calibration provider may be employed who may wish to remain confidential. As well as this, the organisation which employs them may also not want to reveal the fact they outsource their calibration responsibilities (i.e. to stakeholders) to protect business relationships. In either case, confidentiality among participants (*who-calibrates-for-whom*), would be compromised if this information was revealed. In another case, allowing a calibration technician to calibrate devices for a provider in conflict with one in competition with the first, the potential for the collection and leakage of sensitive business secrets is also of concern. Furthermore, while the confidentiality of calibration information itself could be compromised, it may be possible to compromise the operational confidentiality of IoT deployments. Specifically, monitoring calibration processes such as frequent traceability verification checks (which is required for IoT) and collecting the information (*what-is-being-calibrated* and *how-often-it-is-calibrated*), one could potentially reveal how these devices are used and ultimately piece together what the system as a whole is doing.

6.4.2 Summary of Threats

Upon review of the inadequacies in the current state-of-the-art in calibration, at least five key threats can be identified.

Insider Threat

The first threat to consider is an insider adversary. It is possible that a calibration technician could fabricate *idealised* results for a device's calibration to *cook-the-books*. Calibration providers are typically audited every two years to determine the quality and correctness of the provider's calibration processes. In this audit, the goal is to receive a Calibration and Measurement Capability (CMC) certification, which is determined by verifying: all instruments are traceable to standards; calibration models and uncertainty calculations are correct; and that there are procedures in place for maintaining the calibration facility [237]. The CMC ensures that the calibration provider can calibrate with the best achievable measurement uncertainty for an almost ideal measuring device under normal operating conditions. While the CMC would show consumers that a provider can properly calibrate devices within the uncertainty values, verified by the CMC, it does not prevent the provider from fabricating results and thus may go unnoticed until the next audit. As well as fabricating results, it is possible for a rogue or malicious technician to fabricate reports for measuring devices further up in a traceability chain. As the calibration of downstream devices is dependent on upper levels, the integrity of calibration would be compromised. It is also important to consider that some consumers may calibrate their devices with a technician and/or provider that may not be certified, for example to minimise costs. In this case, a technician could fabricate calibration reports for potentially non-existent devices, possibly establishing fake traceability chains which a consumer would blindly trust. This would result in invalid calibration due to having no true traceability, but also potentially very inaccurate measurements which could result in catastrophic

results during device operation.

Large-Scale Compromise

An intentional attack carried out by state actors, or state-sponsored groups, could uncover systemic weaknesses that could lead to the compromise of large fractions of the calibration ecosystem. These vulnerabilities could be exploited by a capable outsider adversary resulting in seeding significant confusion in the best case. In a worst-case scenario, it is entirely possible that entire production cycles can be compromised. For example, consider a supply chain for integrated circuits (IC) used in robotic systems. Any compromise at the calibration level could lead to incorrect measurements, resulting in faulty ICs to be manufactured. By the time any inaccuracies in calibration itself are uncovered, for example after the yearly recalibration cycle in the current state-of-the-art in calibration, potentially thousands of faulty units would have made it to consumers.

Behavioural Economics

While the current calibration ecosystem can be viewed as a set of hierarchical trees of some degree, the considerations of the scale and ubiquitous nature of IoT would substantially increase this with potentially millions of participants. In this case, selfish or malicious behaviours from calibration providers may manifest which can lead cost optimisation at the expense of the rest of the calibration ecosystem. A competing calibration provider, for example, could deploy ransomware to the report storage of another provider, which prevents any calibration records from being accessed until a fee has been paid. Not only would this potentially allow the competitor to become a bet-

ter option in the eyes of consumers requiring calibration (whilst also destroying business relationships for the target), traceability verification and other processes would be significantly delayed or fail, leading to an inability to verify whether an IoT device or system can provide correct measurements.

Flying Debris

In the event of attack, there may be a secondary impact that damages the calibration infrastructure even if other targets were the intended focus. For example, a DoS attack may result in failure to verify if the network is shared with other systems. In the current state-of-the-art in calibration, a centralised storage infrastructure is used. An attack on the availability of the storage would result in significant delays in traceability verification. This may leave robotic systems in an unreliable state, potentially resulting in uncontrolled and/or inaccurate movements and ultimately lead to a precautionary or immediate shutdown of operations.

Inability to Repudiate

Finally, as well as constraining the behaviour of field devices, and close-to-field devices, a discussion of mitigating potential compromise is important. Specifically, ensuring that devices are held accountable for the data they collect and transmit, such as measurements taken from a sensor device. The data should be recorded such that incorrect measurements in traceability verification, for example, can be traced directly back to the device itself, aiding in isolation and mitigation measures to take place in the event of compromise.

7 — Securing Calibration Record Keeping in Digital Environments

Given the proposed threat model and discussion around protection requirements in Section 6 – primarily those of integrity, availability, non-repudiation and tamper-resistance – the key question is what a suitable design would be for a solution that supports calibration traceability and forensics in digital environments, where robots are employed. A summary of the protection requirements that correspond to calibration record keeping are as follows:

(R_1) Allow for the collaboration between interacting parties in the calibration ecosystem, including but not limited to: device operators, manufacturers, regulators and calibration agencies; via storing calibration records in a tamper-evident and fully-decentralised manner (calibration integrity);

(R_2) Enable efficient and timely record-keeping (storage and retrieval), that is also highly available in the event of attack/compromise;

(R_3) Allow for the transparent verification and proving of calibration status prior to device and system operation.

Solutions such as decentralised, cryptographic hash chains [252, 253], where each hash could be associated with a calibration report, show some success. Unfortunately, existing challenges such as ensuring synchronicity across networks and selecting correct hashes remain a

problem. However, using blockchains as a fully decentralised storage mechanism satisfies these problems [254].

Naturally a candidate solution, blockchains provide an innate tamper-proofing mechanism which can be used to track transactions, in this case those that involve operations using calibration reports, which is key to auditing and verifying calibration traceability. Specifically, this can be achieved by applying a transaction-based state machine, or smart contract, on top of the blockchain storage infrastructure that allows to support application for traceability verification and forensics in a non-repudiable manner. Furthermore, a blockchain is also fully decentralised and highly available, which can support not only the integrity and availability requirements (i.e. via replication), but also requires no party to blindly trust others within calibration workflows [255, 256].

Meeting the protection requirements at the scale of the Internet is a challenge, even more so with considerations given to safety-critical IoT systems such as robots. Specifically, such systems require a highly distributed infrastructure that does not rely on centralised components such as centralised authorities, where a single point of failure could hinder the availability and integrity of a system. Instead, the use of a blockchain allows for a peer-to-peer (P2P) transactional network of nodes to independently maintain storage of calibration records, with the use of strong cryptographic links for ensuring data integrity and enforcing non-repudiation. Keeping a record of all transactions, such as creating calibration reports (a result of calibrating devices) or verification checks, ensures that calibrated devices at all levels in the calibration ecosystem cannot deny any operations carried out and thus, be held accountable for these actions.

In a blockchain network, each node receives transactions in a different order. All transactions that are received within a certain time period are aggregated into a new *block*. To ensure that data in all blocks remain consistent across all peer nodes in the network, a voting (*mining*) scheme is employed. Instead of a centralised vote, a proof-of-work (PoW) or proof-of-stake (PoS) model is used to determine which node wins the vote to determine the next accepted block [257,258]. To provide an incentive for good behaviour, an incentive system is used such that each node processes the transactions it receives into the next block and a transaction fee is paid to the mining node of the block including these transactions. Therefore, each node is incentivised to solve as many blocks as possible. Ultimately, each transaction consists of performing certain fixed operations defined by a smart contract, and finally modifying the persistent data on the blockchain.

7.1 System Design

With regard to the design of an appropriate blockchain-based solution for calibration record keeping, there are several points to consider. First, what will be stored on the blockchain that is necessary for calibration traceability verification? Second, how can completeness be achieved in traceability verification from an auditing/forensics standpoint, such that no pertinent information is missing? Third, how can this new solution provide efficient, yet secure, traceability verification over the data stored on the blockchain platform?

What is Stored on the Blockchain?

The first consideration to what will be stored on the blockchain relates to both the data that will aid in verifying calibration traceability, but also data which provides input to tracing measurements back to IoT devices (i.e. robot components) at the field level. From the calibration hierarchy (Figure 6.1), it is clear that all devices are associated with a calibration report. This report outlines information about reference (parent) calibration devices and operating ranges that were tested in calibration with a measurement uncertainty, among others. Further, the report also details the technician and organisation responsible for the device's calibration. An example calibration report can be seen in Figure 7.1. Given that calibration reports in the current state-of-the-art are paper-based, with a digital form taking form of a simple PDF replication or scan, the concern is how to best design a digital calibration report. While it is relatively simple to store PDF files and maintain integrity and regulate access to them, it is much more difficult to perform efficient operations on the data contained within the reports. There has been work in this aspect, with the idea being to ensure digital calibration reports can be universally exchanged and understood, for example using the eXtensible Markup Language (XML) [31]. This serves as an approach for digitising reports and suggesting measures for authentication, encryption and signed transmission of calibration reports. In the proposed digital calibration report, there are 4 subjects within the data, corresponding to: administrative data (i.e. parent references, organisational information, etc.); results of calibration; comments and a human-readable document. Notably, while the human-readable document may be useful, in a blockchain implementation it may be more useful to make use of a smart con-

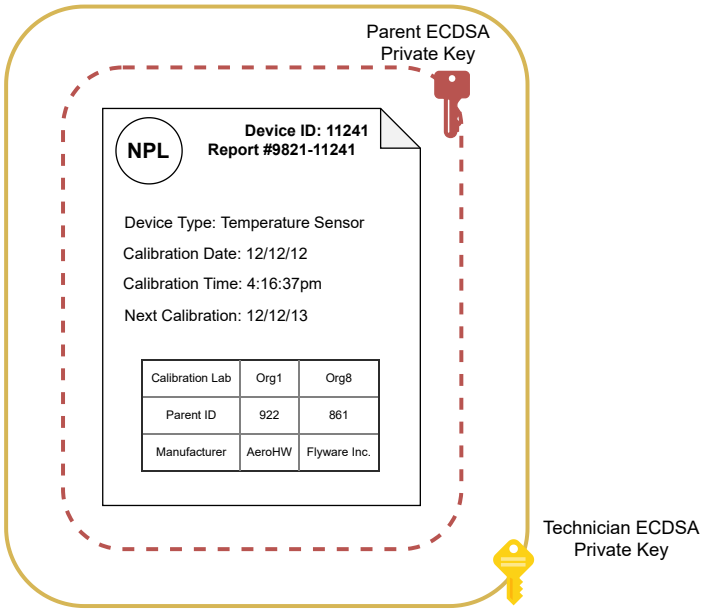


Figure 7.1: Example (Simplified) Calibration Report

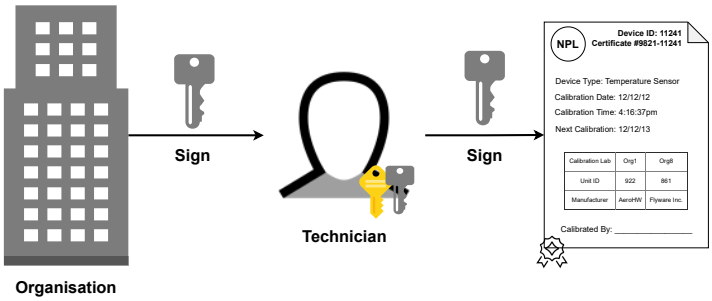


Figure 7.2: Signing Calibration Certificates

tract to collect the necessary information and a different process to present the data, ultimately reducing storage and economic costs. In this work, as well as storing reports, information about the technicians and organisations will also be stored on the blockchain. By storing the reports, the information about reference devices can be used by a smart contract to verify traceability by looking up the associated parents and verifying their calibration, tracing upwards to the master calibration devices maintained at the root (national) level.

Using the Blockchain for Traceability Verification

The next step after deciding what is stored on the blockchain, is to understand how traceability verification can be conducted. Popular blockchain implementations, such as Ethereum [259], make use of smart contracts to execute code and interact directly with the blockchain. They are typically used to automate the execution of an agreement on a transaction such that any participants involved in the transaction are immediately certain of the outcome without the need for an intermediary party [260,261]. For traceability verification, the smart contract can be used to automate the verification check and verify whether there is a complete, unbroken chain of valid device calibration at each stage up to the root of trust (NMI). In a secure boot for a robotic system, for example, the smart contract can be executed such that the system can only be allowed to begin operation under the assumption each of its components have valid calibration. An overview of the smart contract interaction for a sensor component in a robotic system can be seen in Figure 7.4. The contract will take the input of a device ID and execute a function to verify the traceability of a measurement from the sensor. The algorithm used for the verification check can be found in Appendix D.1, which verifies all calibration reports to the root of trust and checks if the root organisation is an NMI. If, and only if, the chain of traceable calibration is valid and unbroken, a *verified* result is returned. While this alone allows for one to verify whether traceability is complete as part of a calibration validity check, it does not prevent an adversary from interfering with a device's chain pretending to be a certified technician. To this, the use of ECDSA (Elliptic-Curve Digital Signature Algorithm) signatures are used to prevent the forgery or creation of fraudulent

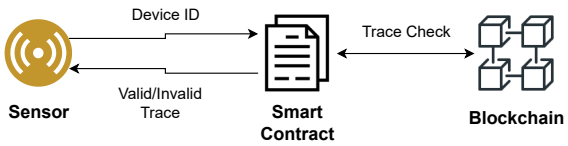


Figure 7.3

Figure 7.4: Sensor Traceability Verification using a Smart Contract
A sensor's device ID is passed into the smart contract which is used to verify its calibration traceability by checking against the appropriate records in the blockchain

calibration reports in the blockchain [262, 263]. Upon calibration, the device's public key will be signed by the certified technician, whose own key is signed by the certifying organisation, and in turn finally signed by the root of trust NMIs. As part of the traceability verification check, the signatures are also verified by the smart contract, which adds a further level of security in establishing a chain of trust.

7.2 Evaluation

The third and final consideration for adopting a new solution to this problem, is to evaluate whether or not a real-world implementation is practical – aside from being theoretically sound from a security standpoint.

7.2.1 Blockchain Environment

For the blockchain implementation, the Ethereum [259] blockchain is used. Ethereum is a Turing-complete, decentralised *value-transfer* environment which facilitates the use of smart contracts to interact with the underlying blockchain. Transactions in Ethereum are those which modify the persistent memory of the state machine, and needs to be run by all nodes to ensure synchronicity across the network.

Smart Contracts

In Ethereum, smart contracts are written in a programming language called Solidity [264], which is implemented as a set of 140 opcodes which all nodes execute deterministically. The opcodes condense to form a bytecode string which can be published on the network in the form of a smart contract. During deployment, a transaction is created by a user who deploys the contract and the contract is given its own unique address. When the transaction is accepted, the smart contract can persist. The smart contract can also have many functions and can also allocate persistent memory on the network, and any user that wishes to interact with it uses the address to call its functions. Given that each transaction function requires computational resources to be executed, every opcode is assigned a fixed cost that is charged for executing the contract.

In this implementation, the purpose of the smart contract is to define the functions set out in the system design. First, functions for establishing the calibration hierarchy were written, which includes creating technicians, organisations and calibration reports. Next, a function was written which enables a user to verify the traceability of a device. The entire list of functions and descriptions can be viewed in Table 7.1. The smart contract was deployed and tested on a private Ethereum blockchain (Ganache) [265], as well as a public test network (Ropsten Testnet [266]) with aid from the Truffle testing framework [267].

Ropsten Test Network

In order to understand how the new solution performs practically in a real-world environment, it was deployed on the Ropsten test network [266]. Also known as the Ethereum Testnet, Ropsten is the

Function	Description
createOrganisation	Accepts an ID and a name, and creates an organisation on the blockchain
createTechnician	Requests an Ethereum address and an organisation id, and will create a technician on the blockchain
createReport	Accepts a number of parameters, such as the device id and technician id, and creates a calibration report object on the blockchain
TraceCal_READ	Accepts a device ID and returns a root calibration-report if it has one, else returns the calibration-report of the device itself.
getParentReport	Accepts a device and returns its direct parent's calibration report or NULL.
getOrgName	Accepts an organisation ID and returns the name of the organisation
getTechnicianOrganisation	Returns the organisation ID of the organisation who certified the technician

Table 7.1: List of Smart Contract Functions

largest Ethereum test network[1] and runs the same PoW protocol as Ethereum. Unlike classic Ethereum, Ropsten is designed specifically for testing a smart contract before it is deployed on the main Ethereum network. In terms of cost, Ropsten uses a form of Ether called *rEth* which has no real-world cost. While this can also be produced from mining, it can also be received from faucets for testing transactions without imposing a legitimate cost. In comparison with other Ethereum testnets, such as Kovan [268] or Rinkeby [269], which use a Proof-of-Authority (PoA) consensus protocol and have lower block confirmation times, Ropsten best reproduces the current state of Ethereum and is best for realistic testing of the new protection mechanism.

Meeting the Protection Requirements

Before subsequent evaluation, it is important to discuss how the protection requirements pertinent to secure calibration record keeping are met by the solution. Simply, a new solution should allow for the collaboration between interacting parties in the calibration ecosystem, in a manner which protects the integrity of calibration in a tamper-evident and fully-decentralised manner. The solution should be highly available, and allow for efficient and transparent verification of device calibration status at any level in the hierarchy. First, all parties can interact with the proposed blockchain solution via the smart contract, which provides the necessary functions required to verify device calibration (R_1 and R_3). Second, the solution provides a basis for auditing and calibration forensics via redundancy and replication across Ethereum nodes (R_1), with the chaining of transactions via crypto-

[1] At the time of writing

graphic links also preventing tampering of the data. Furthermore, by storing the technician and relative organisations involved in a device's calibration, there is more coverage available to aid with forensics and auditing. Third, in the smart contract, ECDSA signatures are used to prevent the forgery of calibration reports, which could lead to the compromise of entire traceability chains (R_3). The smart contract verifies that the report is signed by a legitimate technician, whose signing key is signed by a legitimate calibration provider for whom they calibrate, ultimately verifying the chain of trust along the traceability chain. Finally, given that the data is replicated across the blockchain network, even if one or more nodes are to be taken down, the data will still remain available to be read and written to the blockchain (R_2).

7.2.2 Scalability Testing

The first step in the evaluation, after testing the functionality of the implementation, is to determine how well the solution scales with the ubiquitous nature and size of the calibration hierarchy (R_2) and ultimately, the Internet. The primary measure for scalability in this setting is with regard to execution time of the smart contract, for example how long it takes to verify traceability for a device. As well as this, the additional security measures such as digital signatures were also tested in terms of scalability. The following experiments for evaluating scalable traceability verification have been run on a local blockchain using Ganache as the provider and Remix IDE for running the contract calls. Ganache is used to get the contract execution time, as the contract is executed almost immediately, whereas on the main Ethereum network other contracts may be executed in the same block and measuring execution time would be difficult.

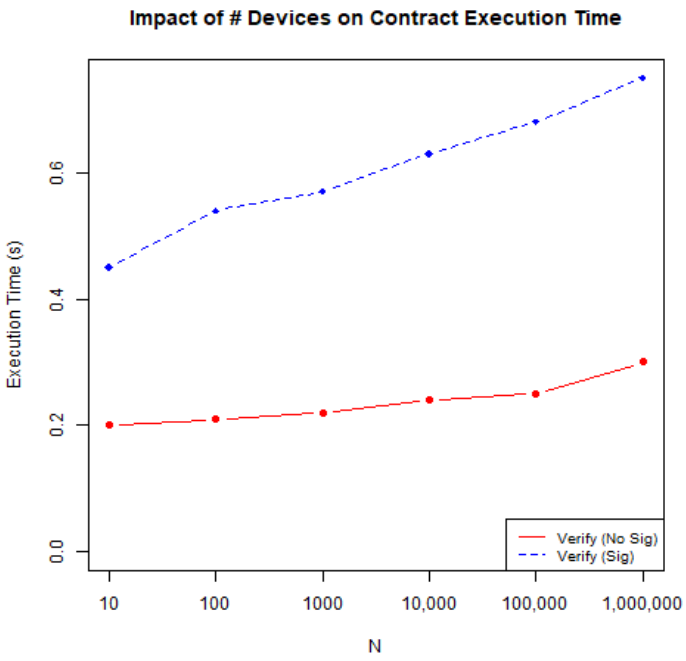


Figure 7.5: Impact of # Devices on Execution Time
Verification times remain under 1s in the worst case of 1 million devices with the use of digital signatures

Impact of # Devices on Execution Time

The first set of experiments investigated the impact of the number of devices in the calibration hierarchy on the execution time of the smart contract for traceability verification. To match more realistic calibration hierarchies, varying numbers of devices at the field level, N, was used as a baseline. From this, the number of levels in a hierarchy was represented as $log(N)$. If 100 field devices were present, for example, this will result in two parent levels and the root NMI level. Furthermore, the number of interacting organisations in the hierarchy is represented by $log_2(N-1)$. Using the same example, 100 field devices maps to 4 organisations.

In this experiment, N falls in the range of $10 \leq N \leq 10^6$. The results for execution time of traceability verification can be seen in Figure 7.5 (red). It is clear that as the number of field devices and

levels increases, the time for verifying traceability also increases. Notably, the execution time, even in the worst case of $1,000,000$ devices, never exceeds 300ms. As well as the execution time without additional overhead, the time was also measured with the addition of signatures during traceability verification (blue). The addition of signatures shows that execution time more than doubles. While this may seem like a significant jump, even in the worst case of $1,000,000$ devices, the execution time is still under 800ms, which is still a significant improvement over the current state-of-the-art [31]. Further, if calibration is performed during system prep (i.e. secure boot), this can be considered a reasonable wait with regard to safety.

Impact of # Levels on Execution Time

While N in the previous experiment was the primary variable, in realistic settings there may potentially be more than $log(N)$ levels. Therefore, a natural next step is to investigate the impact of the number of levels on contract execution time. As shown in Figure 7.6, the execution time in all cases increases as the number of levels increases. As the verification function requires reaching the root level, the time spent will of course increase as the number of antecedent levels in a device's traceability chain also increases. Similar to the previous experiment, the addition of signatures also increases the execution time. At lower numbers of levels, there is only a little impact, but in the worst case of 50 levels the impact increases over 6 fold (around 3.5s).

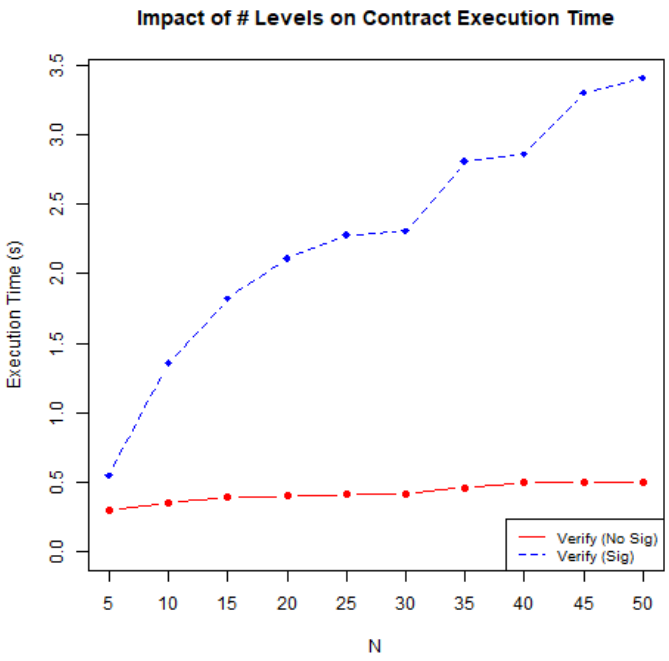


Figure 7.6: Impact of # Levels on Execution Time
Verification times significantly outperform current state-of-the-art, with the use of digital signatures adding only a few seconds to verification in the worst case

7.3 Discussion

In robotic workflows, it is critical to ensure that operations a robot can carry out are done so safely and robust to adversarial pressure. It is clear that calibration ultimately underpins safe operation. The need for digitisation brought rise to a number of problems when looking at the current state-of-the-art, to which lead to the design of this new blockchain-based solution to secure record-keeping in calibration, a foundational step.

7.3.1 System Characteristics

Latency

In the evaluation, it is seen in Figure 7.6 that in the worst case of 50 levels, verification can be carried out in under 4 seconds. In a realistic context, take the example of a surgical robot. If a typical surgical robot is assumed to have tens of sensors and other components that require calibration, such as the Raven-II [45] which uses 9 sensors alone for force-feedback and orientation, the verification can be assumed to take at most a few minutes (or less if it is done in parallel), which is also roughly the time it takes to prime a patient (i.e. IV insertion, anaesthesia preparation, etc.). As described by Hackel et al. [31], NMIs at the root level will perform around 10,000 calibrations per year, intermediaries perform around 100,000 per year and internal facilities situated in safety-critical environments to perform potentially millions of calibrations per year. Given the time to complete a traceability check, a key component of calibration itself, and the increasing numbers of IoT systems being implemented, the issues facing the current state-of-the-art will be evermore present. The current state-of-the-art, being a collection of systems and processes dealing with manual paper records [31], can take in the order of several hours to days for completing a single calibration verification check, in contrast to the proposed blockchain solution which significantly outperforms this.

Scalability

The scaling factors in the evaluation include the number of field devices, such as robot compnents, and the number of levels (interme-

diaries) in the calibration hierarchy for a device. It is clear that verification can be carried out within a reasonable wait time, such as during the device prep cycle or secure boot, using the proposed blockchain system. While it does meet the protection requirements and does well in terms of scale, there are, however, some concerns regarding scalability. Specifically, these scalability concerns are with regard to the Ethereum blockchain. Ethereum uses a Proof-of-Work (PoW) consensus and by definition is hard to solve and ultimately, not scalable by design. This leaves its use rather suboptimal, aside from certain cases such as cryptocurrency design. In Ethereum, the amount of gas [259] each transaction spends determines how many can fit into a block. If a transaction executes more operations, the more gas it costs to execute it. Therefore, since it is possible that an entire block's gas limit can be spent in one transaction, the block mining time for written calibration data to propagate through the network can also increase. Ultimately, given that PoW is not truly scalable, it puts out the question as to whether other consensus mechanisms may be more valuable in a real-world environment. Because of the limitations present with PoW consensus, Ethereum (2.0) [270] makes use of a Proof-of-Stake (PoS) consensus protocol and sharding of chains to overcome drawbacks of higher latency, low throughput and lack of scalability. Instead of relying on physical miners and electricity, PoS makes use of validators and deposits of Ethher, ultimately reducing the environmental impact. The validators lock up stakes of Ether as collateral into a deposit contract. A validator from a select pool proposes a block depending on the stakes that are deposited, and one with more stake will have a higher chance of proposing a new block and earning a reward. In the event of a cheating validator, deposited

stakes can be destroyed. With regard to sharded chains, transactions can be handled in parallel, speeding up the network and providing better scaling. As a point of future work, it would be interesting to evaluate improvements of scalability for traceability verification using Ethereum 2.0, as well as other consensus mechanisms.

7.3.2 Security and Privacy

In this chapter, it is clear that a blockchain-based solution acts as a good collaboration mechanism between different players in the calibration ecosystem. In particular, the use of a permissionless blockchain [271, 272] helps address concerns over privacy related to user profiling by internal adversaries who may apply traffic analysis techniques (e.g. Chapter 3) over calibration traffic. Combining blockchains with appropriate certificate management schemes [67, 69] also prevent malice by internal and external adversaries to resist fake calibration reports, technicians, etc. Furthermore, the ability to ensure that measurements, such as those from sensor devices, can be verified for correctness prior to operation lays a foundation for securing the operation at a fundamental level. Additionally, the solution also enables transparency in the calibration supply chain pertaining to robotic workflows that may also have positive secondary impacts. This include aspects such as extending collaboration across calibration workflows which in turn can support device provenance among robotics applications. With respect to the challenges associated with the context of this chapter, a discussion on how the blockchain solution meets various security properties follows.

System Integrity

Tampering with calibration records in the new solution is prevented by use of a consensus mechanism and cryptographic links between blocks of transactions. Calibration integrity is met through traceability verification, for example that it has not beeen revoked or invalidated. Further, the forgery of calibration reports is prevented by having the certified calibration technician sign the reports for devices thy calibrate, and verified during traceability verification.

Anonymity

For a device's traceability chain, it is important to prevent information from leaking to parties that are not involved (directly) in the device's calibration, for example others in the chain. To do this, all read operations in the permissionless blockchain are anonymous, as the data is read from the verifier's local machine and maintains unlinkability between the verifier and device being traced. Unfortunately, however, Ethereum does not protect operator anonymity. Specifically, the readers and verifiers are not kept truly anonymous and write operations are publicly recorded. While the pseudonymous electronic address used to link transactions can preserve privacy, continous verification could leak this relationship. Given that the security property of anonymity is not entirely satisfied by the Ethereum solution, it is natural to pursue an investigation into other potential solutions. First, other types of blockchain could be used in place of this, however private and consortium blockchains [271] require a known set of participants and provide no anonymity, and so other forms or uses of a blockchain might require a degree of change. It is possible that mixing services on-chain [273, 274] (also called tumblers) could allow probabilistic

anonymity, where transaction tokens are mixed with others and the same values are sent from independent addresses. However, one problem with this approach is keeping transaction costs low enough to advocate for frequent use. In robotic systems, calibration checks will be done quite regularly and potentially be required *on-the-fly*. An example of this is Hawk [275] which proposes a mix construction via ring signatures. Finally, approaches which don't make use of blockchains, while they may require more components, could also be a competitor to this solution and help solve the anonymity problem. It is possible to consider the use of another tamper-resistant data store (i.e. tamper-resistant, forward-secure audit logs [276, 277]) in combination with an appropriate access control mechanism and a group signature scheme for conditional linkability, as a solution to meet the protection requirements away from a blockchain. This point about access control is the focus of the next chapter, and group signatures for conditional linkability is a point for future work.

Availability

Currently, the use of centralised data storage for calibration records leaves the calibration ecosystem vulnerable to targeted attacks on and compromise of the storage servers. Thus, the consideration over the secure design of the storage infrastructure is important, with delays or denials of verification potentially heavily impacting operational efficiency and accuracy. Simply, a decentralised storage infrastructure can meet the requirement for high availability and scale, additional mechanisms would need to be coordinated to meet the rest of the protection requirements – which the blockchain solution fortunately provides.

System Forensics and Auditability

The system as a whole is required to be able to withstand hostile scrutiny in a curt of law. Simply, if a robot was to misbehave or malfunction, was it the operator who is to blame? Or are problems such as jerky movements the result of invalid calibration. Several lawsuits, such as those in surgical contexts [278, 279] are clear demonstrations of the potential liabilities that can arise with robotic systems. Thus, support for system forensics is required to ensure that the correctness of calibration can be verified in all aspects, and to ensure appropriate records of such are maintained which can be used at a later date. In comparison with the current state-of-the-art, where forensics is carried out over disconnected, centralised databases of calibration reports, the blockchain-based system keeps a strong, tamper-resistant trail of evidence that can be followed throughout entire traceability chains. Furthermore, additional smart contracts could be written to support generalised and feature-specific forensic and auditing applications.

7.4 Summary

As we start to rely on IoT cyberphysical systems to perform critical tasks such as surgeries or direct operational control of utilities, the security of the calibration infrastructure itself will start gaining importance and mechanisms will be required to deal with attackers in the calibration ecosystem. In the presence of adversaries, it can be argued that calibration correctness becomes a security requirement. In this chapter, a new security mechanism is proposed that enables sensed data to be subject to verification via *on-the-fly calibration checks*. This involves two requirements. First, *end-to-end* cal-

ibration traceability, i.e. tracing measurements and calibration status to their corresponding gold (national) standards by validating calibration certificates on-the-fly. Second, calibration forensics; enabling the operator, regulator(s), manufacturer(s), and calibration agencies to work together to create a tamper-resistant trail of recorded activity to aid system forensics, in a court of law when things go wrong. There have been cases of lawsuits filed by patients, accusing hospitals of negligence over safety considerations when surgical robots have inflicted accidental injuries [278, 279], and such are illustrative of the significant liabilities and stakes involved when IoT cyberphysical systems are involved in safety-critical tasks.

Ensuring system safety in the presence of adversaries is a significant security challenge. Calibration security is one part of a larger framework of security mechanisms required to secure IoT cyberphysical systems. While much attention has focused on the security vulnerabilities of IoT devices, focusing on the misapplication (or non-application) of secure channels, weak (or non-existent) authentication, and missing/invalid configurations [280, 281], among others, the security of the sensed data itself is a significant challenge. In this chapter, a concrete mechanism is proposed for securing the integrity of calibration based on smart-contracts leveraging the Ethereum blockchain. The mechanism was motivated by the need for a mechanism that is: highly available, verifiable and tamper-resistant, and works in a zero trust environment. The proposed mechanism can successfully and scalably verify traceability chains, to ensure one can maintain valid calibration and rapidly attend to adversarial faults, leveraging blockchains as a highly-available tamper-resistant chain of evidence.

7.4.1 Future Directions

Although the proposed blockchain-based solution shows to meet the requirements, there are some future directions to this work. First, in terms of scalability, an avenue for future work here would be to explore other consensus mechanisms such as Proof-of-Stake, to determine whether times will improve as we scale to larger calibration hierarchies and the Internet. Second, the focus will turn to confidentiality and anonymity. In terms of confidentiality, while NMIs are organically trusted for correctness they are not necessarily trusted with leaking information about what system operations are being carried out. This leaves a solution open to the possibility of permissioned or consortium blockchains (such as Hyperledger Fabric [282]) as the base for the protection mechanism as opposed to a public permissionless blockchain (Ethereum). This work assumes there is no need for a trusted party, however one could argue that the trust maintained by a set of international NMIs could pave the way for set of *trusted points*. One question which may arise is why a system making use of traditional public-key infrastructure (PKI) is not considered. PKI unfortunately demonstrates some key issues, for example: the compromise of private key(s); lack of public verifiability; impersonation of certificate authorities (CAs); among others. Given that in this approach the focus is on blockchains with an open, transparent and secure foundation, the nature of public verifiability (reads) eliminates relying on third-party CAs (trust). Further, the distributed nature of data eliminates availability risks found with older PKI systems. In the case of auditing, blockchains are time-stamped and records and linked, protecting the integrity of data in a secure, distributed manner which is better suited to IoT contexts. Next, with regard to

anonymity, the use of zero-knowledge proofs for verification would aid in preventing information leakage surrounding information contained within calibration reports and metadata which arises from verification checks along traceability chains. Potential avenues in this case could surface from zero-knowledge smart contracts (i.e. zk-SNARKs in ZCash smart contracts [283]) or even group signatures where each device in some child level (i.e. field level) would be grouped together with its calibrator (i.e. some intermediary facility), who can then provide revoking responsibilities etc.

8 — Access Control for Robot Calibration Traceability

During the traceability verification process, it is important to verify that an individual making the request has the appropriate rights to do so. For example, it is sensible to allow a verified system/device owner to access the calibration reports for the components a robot comprises of, but not necessarily access any information about the parent reference devices that are used to calibrate the system. As well as this, it is also important to give consideration to the ability to write calibration reports for a device. Specifically, it should be an authorised technician from a reputable organisations that has access to write reports for specific devices, as opposed to anyone that could claim they are a certified technician and forge their own calibration reports to gather sensitive information about a system's calibration. Ultimately, the key question here is how can access control be managed in a digitised calibration ecosystem to ensure the security of robotic deployments and information flows within the calibration ecosystem.

8.1 Calibration Traceability and Access Control

While there exists other challenges associated with digitising calibration processes, a key concern pertains to the accessing of calibration

reports and the authorisation of people, systems and processes trying to access them. In the calibration ecosystem, many of the interacting parties cooperate in harmony. For example, this includes NMI organisations at the top of the hierarchy who act as the root of trust. Unfortunately, a subset of these organisations within a robot's chain of traceable calibrations may share an adversarial relationship – that is, to be in direct competition, or *conflict*, with one another. Consider the case of a robotic system where each of its components, such as sensing devices, require frequent calibration to ensure that measured data is accurate and reliable. While the calibration of many robotic systems may ideally be done in-house, realistically calibration is usually performed by a third-party calibration service provider. This may be some intermediary calibration facility, who may wish to keep the relationship between themselves and the system-level organisation strictly *confidential*. This may be for a variety of reasons, but can include the likes of system-level organisations showing responsibility and capability to carry out their own calibrations, or to help maintain business relationships with stakeholders. If this information is to be leaked to competitors, this could compromise the confidentiality among participants (*who-calibrates-for-whom*). For example, these competitors could be industrial robotics companies or hospitals which employ surgical robots. In another example, allowing a technician to calibrate a robotic system for an organisation in competition with the one they already calibrate for could lead to the leakage of sensitive business secrets. As well as protecting the confidentiality among participants, there are two other considerations regarding operational confidentiality of robotic deployments. The information regarding the frequency of calibration and the components of a robotic system be-

ing calibrated could lead to its compromise. For instance, monitoring calibration processes and collecting other meta-data could reveal how system components are being used. Therefore, it is also important to protect *what-is-being-calibrated* and *how-often-it-is-calibrated*. The last consideration which pertains to access control for robot calibration is *efficiency*. In the previous chapter, access to the reports were assumed to be granted, however in a real-world environment this is not as simple. In the current state-of-the-art, to carry out a verification of the calibration status of a robot's component, a subject (i.e. device owner) must first request the report for the device from the responsible organisation. Once access is granted, for example if some imposed economic cost is satisfied, the parent devices can be identified and the process is repeated up to national standards [29]. While it is a relatively trivial process, even a conservative number of devices could easily overwhelm efficiency standards for verifying the calibration of a robotic system and all its components before it is due to begin operating. In the current state-of-the-art, this could take anywhere from a few hours to even days or weeks depending on the organisation requirements [31].

Ultimately, five key requirements arise which a solution should satisfy:

(R_1) How can the integrity of calibration, from a traceability verification standpoint, be maintained to mitigate damage to other levels in a traceability chain?;

(R_2) How can the confidentiality of business relationships (*who-calibrates-for-whom*) be protected, whilst providing transparency for calibration traceability?;

(R_3) How can operational confidentiality for robotic workflows be

maintained, by protecting *what-is-being-calibrated* and *how-often-it-is-calibrated*?;

(R_4) How can conflicts between subsets of interacting parties be managed, such that unintended disclosure of information can be avoided?;

(R_5) Can access control in calibration traceability be carried carried out efficiently (*on-the-fly*) whilst satisfying R_1–R_4?

8.2 A Unified Access Control Model for Calibration Traceability

The potential compromise of calibration integrity and confidentiality, coupled with adversarial relationships among subsets of interacting parties, presents a unique access control challenge, where meta-information flows (e.g. *what-is-being-calibrated*, etc.) need to be managed. Specifically, the key failures are exacerbated by varying access control requirements within a multi-level hierarchy.

8.2.1 Information Flow Constraints

The first step in designing an appropriate solution is to first define the information flow constraints that should be enforced. For subsequent discussion, references will be made to the information flow model depicted in Figure 8.2.

Multi-Level Integrity

The first requirement ($R1$) is to maintain the integrity of calibration. By maintaining the integrity of calibration reports at the NMI level,

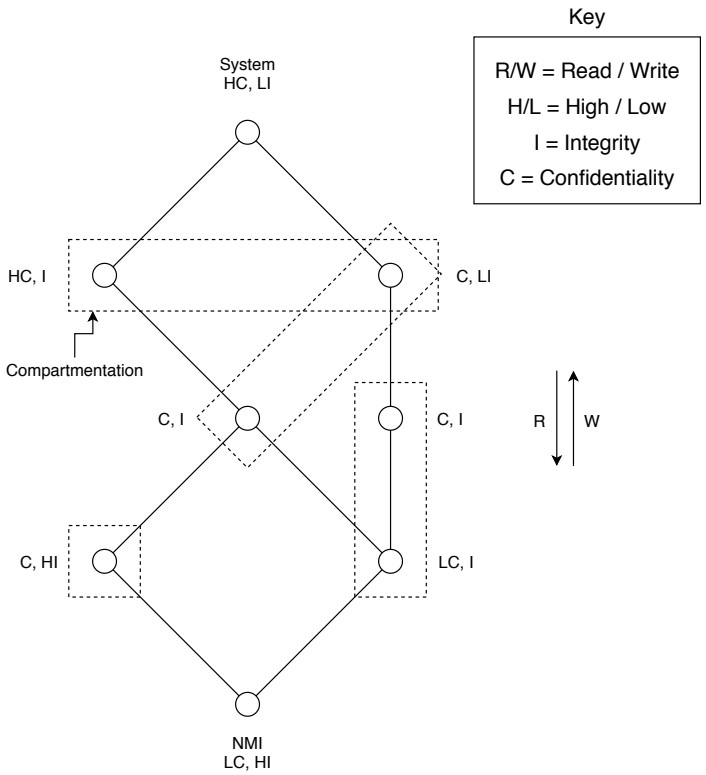


Figure 8.1: Information Flow Model

Figure 8.2: Information Flow Model

Information flows from a high integrity and low confidentiality source (root level) to a low integrity and high confidentiality destination (field level), with conflicts of interest compartmentalised to prevent unauthorised information flows between them

the damage that is inflicted to intermediary levels below the NMI can be reduced. If compromise occurs at the root level, the validity of calibration at all subsequent levels is questionable. By maintaining the integrity of reports at each level, the damage can be limited to its immediate locality, as opposed to inflicting widespread damage. Ultimately, information flows from a high integrity source, the root calibration units at NMIs, to a low integrity destination, the components which make up robotic systems.

Multi-Level Confidentiality

The next requirements, $R_{2\&3}$, relate to maintaining confidentiality among participants (business relationships between providers, stakeholders, OEMs, etc.), as well as the operational confidentiality of field-level robot deployments. An interesting observation while looking at calibration information flows is that it is relatively trivial to map components in traceability chains to real actors. For example, a robot component's calibration report contains information that can reveal the provider of its calibration, and information that describes that provider's internal calibration processes. In the context of NMIs at the root level, the mapping of components does not succumb to confidentiality concerns given calibration at this level acts as the root-of-trust and is globally available. However, intermediary providers may not wish to reveal this information to other (competing) providers, and thus this mapping is of real concern. As well as this, the timing of verification checks could potentially compromise the operational confidentiality of robotic deployments. For example, the timing of surgical procedures could be leaked as a result of continuous monitoring of calibration verification checks for a surgical robot. Furthermore, it is

also key to ensure that verification does not reveal information about robotic deployments to intermediaries in a calibration chain between the robot at the field level and NMIs at the root level. The reason for this is simple: calibration traffic can leak information to parties further down a chain. For example, a robot manufacturer may employ the services of a third-party calibration provider to calibrate the sensors of the robot, but may not wish to reveal this third-party relationship (e.g. to protect stakeholder relationships). Overall, the field level where robotic systems are deployed must retain the highest level of confidentiality, whilst calibration at the root level have the lowest confidentiality requirements.

Conflicts of Interest

The last constraint pertaining to calibration information flows is to prevent the unintended disclosure of information to competing parties – in particular that relating to a natural conflict present between a subset of calibration providers that provide services to robotic systems. For example, a hospital that employs several surgical robots may wish to hide the identity of calibration providers who are used to calibrate the robots. A (potentially malicious) technician from this company should not be allowed to calibrate robots for other hospitals, that may be in competition with the first, to prevent the leakage of sensitive business secrets. Ultimately, it is key to protect this information and those in competition should be *compartmentalised* to ensure that information cannot flow between them.

8.2.2 Existing Access Control Models and Calibration Traceability

Given the set of information flow constraints that cpme with calibration traceability, a natural course of action to pursue is to examine existing access control models and design an appropriate mechanism to enforce the constraints.

The Bell-LaPadula (BLP) model [34] meets the requirement for managing information flows that are constrained by a multi-level hierarchy of confidentiality requirements. While it is most prominent in military or government applications, this existing model matches the multi-level confidentiality constraint in which calibration information should flow from a high confidentiality source (the robot) to a low confidentiality destination (calibration providers or NMIs at the root level). Unfortunately, BLP alone does not satisfy the other requirements of integrity and conflicts of interest. Similar to the BLP model, the existing BIBA model [33] prevents unauthorised modification wherein information flows from a low integrity source to a high integrity destination are prevented. Again, however, while BIBA satisfies the integrity requirement, it does not satisfy either of the confidentiality or conflict of interest requirements. With regard to managing conflicts of interest between calibration providers, the Chinese Wall model mandates that access to data is constraint by what data the subject already holds access to and not just by the attributes of the data in question [35]. In contrast to this, BLP and BIBA place no constraints on the interrelationships between objects and structure is defined by the security attributes of the data.

Ultimately, while each of these models do satisfy individual requirements, neither of them alone will manage the undesirable information

flows previously described. Thus, the unification of the BLP, BIBA and Chinese Wall models is a natural next step.

8.2.3 Defining the Unified Model for Calibration Traceability

For subsequent discussion of the novel unification of these models, the related terminology is defined using notation based on the work of Sandhu [284].

(Definition) Conflict of Interest Set

A Conflict of Interest (COI) set is defined as the set of conflict of interest classes, that contains all calibration reports (objects) whose providers are in direct competition with each other. Following standard notation, the set of n COI sets is denoted by $\{COI_1, COI_2, \ldots, COI_n\}$, where each set $COI_i = \{P_1, P_2, \ldots, P_k\}$ and P_k is the group of calibration reports which concern the same provider k.

(Definition) Set of Integrity Labels

The set of integrity labels is denoted as $\Omega = \{\omega_1, \omega_2, \ldots, \omega_q\}$, where each label corresponds to a unique integrity level. In accordance with the information flow constraints, each integrity label also constitutes a unique confidentiality level.

(Definition) Security Label

A security label is defined as a set of two n-sized vectors $\{[i_1, i_2, \ldots, i_n], [p_1, p_2, \ldots, p_n]$ where $i_j \in \{COI_j \cup \perp \cup T\}$, $p_j \in \Omega$ and $1 \leq j \leq n$.

- Where $i_j = \perp$, the calibration traceability chain does not contain information from any provider in COI_j.
- Where $i_j = T$, the calibration traffic contains information from *at least* two facilities who are in a conflict of interest set COI_j.
- Where $i_j \in COI_j$, the calibration traffic contains information from the corresponding calibration facility in COI_j.

(Definition) Dominance Relations

The (transitive) dominance relations between security labels is defined as follows, where the notation $l_j[i_k]$ denotes the i_k^{th} element of the label l_j. A security label l_1 dominates a label l_2, denoted by $l_1 \geq l_2$, where $l_1 \geq l_2 \iff \forall i_k, p_k = (1, 2, \ldots, n)[((l_1[i_k] = l_2[i_k]) \vee (l_2[i_k] = \perp) \vee (l_1[i_k] = T)) \wedge (l_1[p_k] \leq l_2[p_k])]$.

- A label l_1 dominates a label l_2, provided that l_1 and l_2 agree whenever l_2 is not public or in conflict, and the integrity level of l_2 is higher than that of l_1.
- The security label corresponding to an NMI at the root level, $\{[\perp, \perp, \ldots, \perp], [\omega_q]\}$, is **dominated by** all other levels.
- The system high, denoted by $\{[T, T, \ldots, T], [\omega_1]\}$, **dominates** all other levels.
- The dominance relation defines a lattice structure, where the NMI label appears at the bottom and the level trusted appears at the top. Incomparable levels are not connected in this lattice structure.

In accordance with the proposed access control model for calibration traceability, the rules for information flow as they apply are as follows:

1. Simple Property: A calibration technician (S), may read a calibration report (O), only if the label, $L(S) \geq L(O)$.

2. * (Star) Confinement Property: A calibration technician (S) can only calibrate (write) a system component or unit (O), if the label of the component dominates that of the technician, i.e. if $L(O) \geq L(S)$. Specifically, a write operation corresponds to the creation of a calibration report.

8.3 Evaluation

While the proposed unified model is theoretically sound, an important factor is whether it is practical to enforce the constraints in a real-world application. This is important, as if it is not efficient or scalable enough in the real-world then it cannot suitably verify traceability in appropriate time for safety-critical contexts, such as for surgical robots. The evaluation follows a case example for which the model can be applied and investigates the performance of the model for authorising access to conduct traceability verification.

8.3.1 Case Example: *Calibration Traceability for a Robot's Sensor*

For subsequent discussion and keeping within the scope of this work, the case example will follow the lifecycle for an infrared thermometer sensor which is described in Section 6.2, and how the unified model can be applied to it. Furthermore, the uncertainty calculations which make up part of the traceability verification process are discounted. This is due to the primary concern pertaining to the authorisation

time for enforcing the model contraints, as opposed to the overall time to complete a traceability verification check.

In order to calibrate an infrared thermometer sensor, four things are required: (1) a thermal radiation source; (2) a transfer standard (an intermediate device used to compare measurements against those from the device under test); (3) an ambient temperature thermometer; and (4) a distance measuring device. As shown in Figure 8.3, an example hierarchy of traceable calibration to national standards for the sensor can be seen. In this example, the following is assumed:

- The calibration facilities $O1$–$O3$ are classed as intermediary calibration facilities, and $O4$ is a National Measurement Institute (NMI)

- The infrared thermometer sensor is calibrated by technician $T1$ at calibration facility $O1$

- The transfer standard used in calibrating the sensor is itself, calibrated by a technician $T2$ at organisation $O2$

- The facilities $O2$ and $O3$ are in direct competition with one another

- The traceability chain information flows from the sensing device to $O1$, to the transfer standard calibrated by a technician at $O2$, towards the NMI ($O4$)

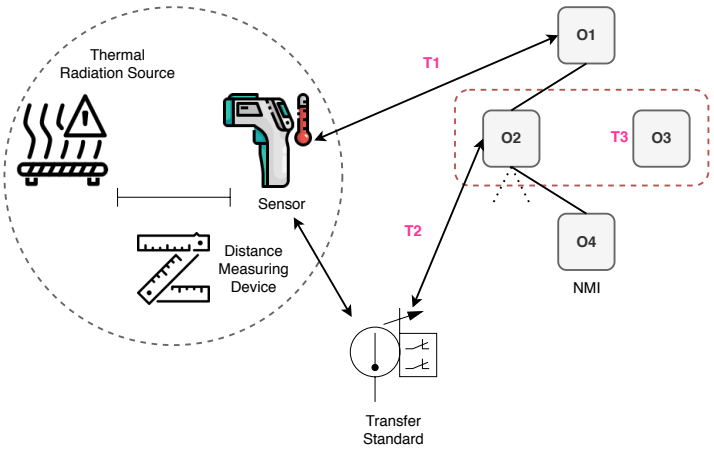


Figure 8.3: Traceability Chain for Infrared Thermometer

Given the above assumptions and a description of the calibration traceability lifecycle for an infrared thermometer sensor provided in Section 6.2, a discussion on how the unified access control model can be applied can now be described.

Initial Calibration

In accordance with the proposed unified access control model, and taking the case example of the infrared thermometer sensor, its initial calibration would be conducted by the technician $T1$ at the facility $O1$. This process will output a calibration report for the sensor and will be given the security label assigned to the technician: $\{[\bot, \bot, \bot], [\omega_1]\}$. Similarly, the transfer standard is calibrated by the technician $T2$ at the facility $O2$ and will have the security label: $\{[\bot, COI_1, \bot], [\omega_2]\}$, where $COI_1 = \{O2, O3\}$ contains the group of calibration reports which concern both the providers $O2$ and $O3$ that are in conflict. In most cases, there will be no conflicts that arise as part of a device's initial calibration. However, some devices, such as those created for military or other government organisations, may be classified in nature. Thus, if the organisation previously used a calibration facility

for calibrating a set of other devices, the new facility to be contracted could be in competition/conflict with the other and thus the labels would indicate a conflict in the chain.

Verifying Traceability

In accordance with the proposed access control model, the security label of the device being traced to national standards must dominate the label of the party carrying out the verification check. For example, in the case of recalibration which first involves a traceability check, the label for a device must dominate that of the verifier. As it is required to read all the reports of all parents at each step in the chain, up to national standards, the label of each parent should also dominate that of the verifying party, such that for a traceability chain $C = \{L(O_1), L(O_2), \ldots, L(O_n)\}$, $\forall c \in C,\ L(S) \geq c_i$, where $i, n \geq 1$ and $L(S)$ is the label of the party carrying out the verification.

Recalibration

The technician performing calibration must first be allowed to carry out the traceability check, such that the label of the component or unit being calibrated dominates that of the technician, $L(O) \geq L(S)$. Specifically, this dominance relation is satisfied when the technician is not in conflict; i.e. if the technician is the same as the one who performed the initial calibration or previous recalibration, then they will be allowed to do so. Similarly, if not then the model would verify that the new technician performing recalibration is not in conflict with the previous, or others in the chain. That is, the traceability chain of the equipment being recalibrated does not contain information from both technicians in a conflict of interest set COI_j.

8.3.2 Performance Evaluation

Now that a discussion on how the unified model can be applied to a robot component in a realistic setting is clarified, the next step is to determine its practical performance in the real world to support efficient calibration verification *on-the-fly* and scale with large, complex calibration hierarchies.

Experimental Setup

In this evaluation, an attribute-based authorisation framework following the XACML standard [285] is used. An overview of the structure of the framework used in this evaluation can be seen in Figure 8.5. The framework provides a standard for access requests and policy specification, where a client program, or Policy Enforcement Point (PEP), brokers access requests between the subject and a server running a Policy Decision Point (PDP). The experiments on the authorisation framework were carried out on a virtual machine running Ubuntu 14.04 LTS with 64GB of RAM allocated to it.

Baseline Model

To provide a more in-depth comparison on how the proposed unified model performs, the first step in the evaluation was looking at a *simple conjunction* of the three aforementioned models (BLP, BIBA and Chinese Walls) as a baseline to compare the unified model against. The term RBIBA (Reverse BIBA) is used given that the information flow constraints of BIBA are reversed to match those of BLP (Figure 8.2). To create the baseline model, an XACML *PolicySet* was established in which policies for each model are combined and enforced together using a *PolicyCombiningAlgorithm*. As the policy set contains mul-

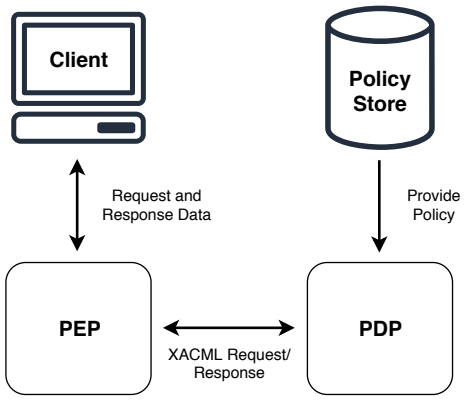


Figure 8.4: XACML Authorisation Architecture

Figure 8.5: XACML Authorisation Architecture
A client (verifier) will issue a request for calibration (e.g. reading a report) which is send to a Policy Enforcement Point. This will interact with a Policy Decision Point which verifies whether the client has the appropriate security label for successful action verification

tiple policies, with each returning different decisions based on their individual constraints, the key question is what the combination algorithm should return. In this case, the *permit-unless-deny* algoritm is used which only allows a *Permit* or *Deny* response and will deny access to calibration if any one of the combined policies returns a *Deny* response. In comparison, the unified model is a single policy which uses the *deny overrides* algorithm such that if any rule results in a *Deny* response then this decision wins.

Authorising Traceability Verification

For the first part of the evaluation, the observations pertain authorisation time taken for access requests for calibration traceability verification using the unified model. For comparison, this is also measured against the baseline model.

Naturally, in the calibration hierarchy, there are several levels in a traceability chain for a single robot component. A simple temperature sensor, for example, could be calibrated with a platinum resis-

tance thermometer, which is in turn calibrated by a more accurate thermometer, and finally by a helium gas thermometer (primary reference standard) [286]. In robotic systems, the number of levels could be much greater, with some consideration given to more generic robots which may use off-the-shelf components. For completeness, this experiment measures the authorisation time for components that may have up to 50 levels, with only a single parent (reference) device at each level. As shown in Figure 8.6, the unified model is significantly faster in authorising traceability requests compared to the baseline simple conjunction in all cases, with authorisation times not exceeding 11ms on average in the worst case, compared to roughly 30ms for the baseline model.

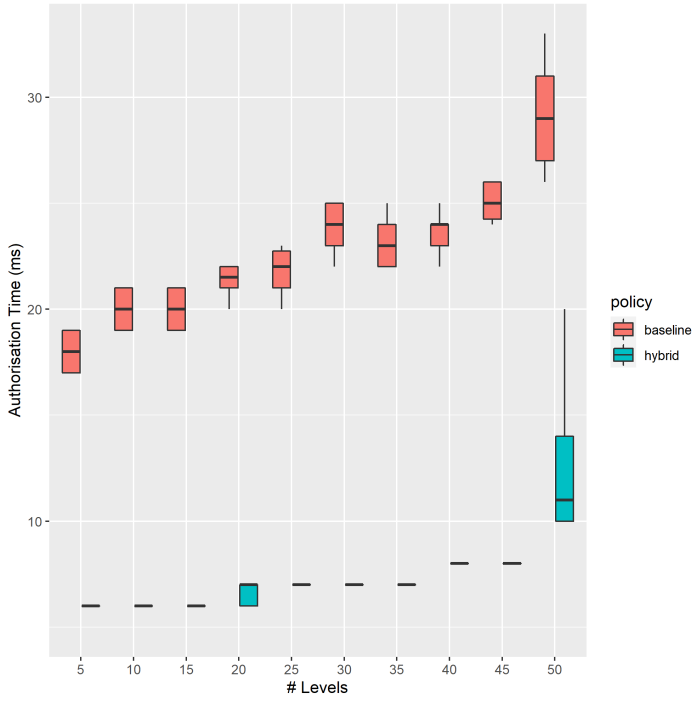


Figure 8.6: Authorisation Time for Single Traceability Chain
The unified model outperforms a simple policy conjunction by at least a factor of 2

In more realistic traceability chains, some devices are calibrated with more than one reference (parent) device, such as the case example

described in Section 8.3.1. Each of these parent devices may also, in turn, have more than one parent and so forth. This means that instead of assuming a best case of a single parent hierarchy, there are chains that may have several branches. Therefore, it is important to measure the authorisation time for multiple branches. The second experiment observes this impact for 2 and 4 branches per level. As shown in Figure 8.7 and Figure 8.8, there is a similar pattern to a single branch for a chain, with the unified model outperforming the baseline model by at least 15ms in the worst case. Furthermore, there is also an increase in the authorisation time as the number of branches per level also increases. This is because the authorisations for each parent device in respective branches also needs to take place up to the root level.

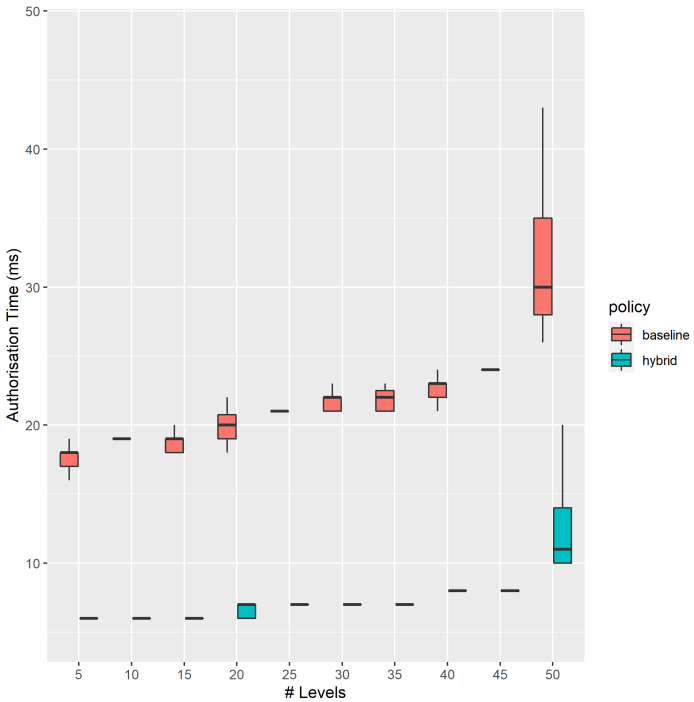


Figure 8.7: Authorisation Time for 2 Branches Per Level
A similar result is seen at 2 branches per level compared to a single chain, with authorisation times using the unified model improved by at least a factor of 2 compared to a simple policy conjunction

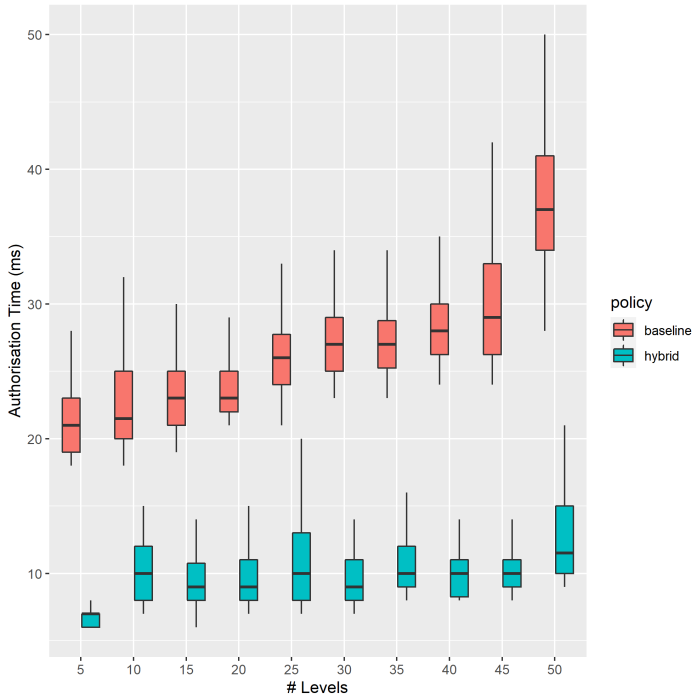


Figure 8.8: Authorisation Time for 4 Branches Per Level
More variation is seen among runs at each level, but authorisation times still improved by a factor of 2 compared to the baseline simple policy conjunction

In all cases, while the unified model does significantly outperform the baseline simple conjunction, the authorisation time does increase with respect to chain complexity. However, in the case of the unified model, the increase of just a few milliseconds as the complexity of the chain increases can be considered a reasonable wait on the device prep-cycle, or if a critical measurement was to be taken [287].

Conflict Management

Pertaining to the conflict of interest set component of the security label, it is the size of conflict sets, rather than the number of them, which requires evaluation. This is because some sets may only contain a small number of members, whilst others may contain more. Thus, aside from measuring the authorisation time for chain complexity, the

effect of the size of conflict sets on the authorisation time also needs to be explored.

For the calibration traceability dataset, a set of synthetic calibration reports were generated containing real calibration data. Each of these reports were assigned a security label, where the integrity component of the label corresponded with the level at which calibration was carried out. The conflict component of the label was generated using a $G(n, p)$ variant of the Erdös-Rényi random graph model, where n is the number of potential competitors and p is the probability of conflict. Further, the cliques of the random graph represented conflict sets where each node n_i was assigned a set of conflict of interest sets. To evaluate the impact of the size of conflict sets, the number of potential competitors and probability of conflict, n and p respectively, were increased resulting in a range of conflict set sizes from 1 to 50 members. As shown in Figure 8.9, the time taken to authorise traceability verification requests increases as the size of the conflict set increases. This experiment shows the result for a single conflict set, but as it is clear that there is an increase in verification time for a single set, it is trivial to assume that the number of sets will also increase authorisation time. In either case, this increase results in a fairly minimal impact and as with the previous experiments, this can be considered a reasonable wait on the device prep-cycle.

8.4 Discussion

Within the calibration hierarchy, a unique set of information flows present themselves with respect to accessing calibration traceability verification, to which classical access control models fail to meet the

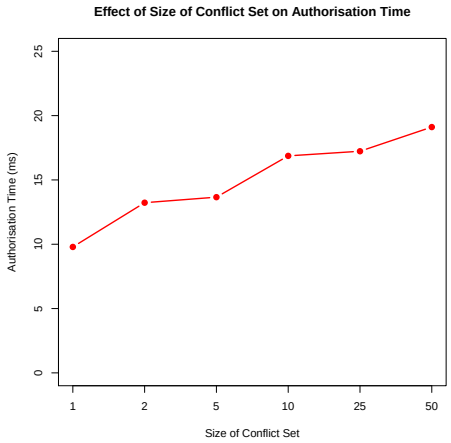


Figure 8.9: Effect of Conflict Set Size on Authorisation Time
The time taken to authorise traceability verification increases slightly as the size of the conflict set increases

desired requirements of multi-level integrity and confidentiality, and compartmentalising conflicts of interest. To this, a novel unification of three existing access control models: BLP, BIBA and Chinese Walls, can effectively constraint the information flows with authorisation times of at least 10ms lower than the baseline in all cases. With respect to calibration in digital environments, this is a critical first step to ensure the safety of devices and organisations who employ these devices before they are exposed to their own respective threat landscapes.

8.4.1 Limitations

While the proposed unified model is successful, there is a limitation that should be considered. Specifically, the enforcement mechanism used assumes that policies and calibration reports are stored in a centralised infrastructure at each calibration provider (i.e. a calibration report is stored at the location at which it was calibrated). The first consideration relating to this pertains to transparency. Specifically,

third parties will have no information about how and when data is requested. If a calibration report is invalidated at a higher level, then a traceability check at a lower level will fail as the report for the device at a higher level is invalid. The second consideration relates to the impact of additional security mechanisms, such as TLS and PKI, on the authorisation time. Trivially, additional security measures naturally presents itself with overheads that would need to be weighted. For example in certain cases, such as surgical settings where an increase in authorisation time could delay the start of a critical procedure. In both cases, while this chapter focuses on the access control aspect, in particular the authorisation times, these issues are looked at in further detail in Chapter 7.

8.5 Related Work

The security surrounding time- and safety-critical IoT systems is an active research area [288], with the main focus pertaining to attacks in the cyber domain (i.e. control system security [64, 243]) and the physical domain (i.e. physical compromise of sensors [289, 290] and physical safety of devices and surrounding environment [290, 291]). However, with the calibration of such systems contributing highly to system accuracy and precision, the compromise of its calibration can ultimately impact the ability to operate safely. Quarta et al. describe calibration parameters of robotics systems to be an essential construct used to compensate for known measurement errors [51]. They demonstrated that by manipulating calibration parameters, an adversary could cause the robot to operate unsafely, such as affecting the servo motor causing the robot to move erratically. Consider a tempera-

ture sensor mounted on a needle driver in a surgical robot, which is calibrated in a manner such that it provides accurate and reliable readings for temperatures between 0–50°C. If these calibration parameters are modified to state that it is accurate up to 100°C, then the system would accept this at face value, and any sensed data cannot be trusted.

Existing access control literature for safety-critical IoT systems focus on various aspects of the system itself, ranging from securing control systems (i.e. access to actuators) to the validation/enforcement of security policies. Hasan and Mohan [243] propose a framework based on the Simplex architecture, commonly used for time-critical cyberphysical systems for fault-tolerance, which makes use of a rule-based invariant and access control mechanism to ensure the timing and safety requirements of IoT cyberphysical systems (i.e. ensure some task can only access a given actuator if the task has the required permission given a set of invariant conditions). Frank et al. [292] describe a combination of both logical and physical access control – explain each, respectively. He describes that the most widely used multi-level security models are inadequate when logical resources obtain a physical form, that makes use of both mandatory and discretionary access control. While not directly applicable to calibration traceability, one must also consider the access constraints to the physical process of calibration which traceability verification is a key part of. Compared to this work, this approach uses attribute-based access control (ABAC) due to its flexibility, limited only by the computational language when implementing policies to enforce access control models. Specifically, it allows greater breadth for access relationships (subjects to access objects) without the need to specify the individual relationships between

them. Compared to other traditional approaches such as rule-based access control, this makes its use ideal for dynamic environments such as SC-IoT [293].

In this work, the focus is on the calibration angle which has been paid little attention to. Specifically, this focus pertains to the unification of three classical access control models (namely BLP, BIBA and Chinese Walls) which is required to solve the novel set of information flows which arise from calibration traceability. Yang et al. [294] state that while BLP is widely used to enforce multi-level confidentiality, it lacks flexibility due to strict confidentiality rules. Furthermore, they describe that BLP poorly controls integrity and that BLP is commonly combined with BIBA for increased integrity control [295–297]. In their work, they propose an improved BLP model to manage multi-level security, where the security of each level is distinguished by the security level of the accessed content itself (subjects are defined as a multi-level entity and objects are defined as a single-level entity). With regard to BIBA, Liu et al. [298] note that BIBA can possibly deny non-malicious access requests made by subjects, ultimately reducing the availability to a system. To this, they propose the integration of notions from Break The Glass (BTG) strategies – a set of (efficient) strategies used to extend subject access rights in exceptional cases (i.e. irregular system states) – with the existing BIBA model (BTG-BIBA). They show that with the proposed BTG-BIBA model, it can now provide more fine-grained access control that is context-aware for dynamic situations. In this work, traditional BIBA and BLP models are taken into account for the case of unification, however one can question the applicability of improvements made to these models over recent years. While attribute-based access con-

trol provides key advantages to earlier forms of access control, such as ACLs and role-based access control, and having a well-maintained policy declaration language and authorisation framework (XACML) for practical solutions, other forms of access control have been proposed which may also be suitable for enforcing the unified model. For example, capability-based access control has been shown to perform well in highly scalable and distributed environments, such as IoT. Similarly, like attribute-based access control, capability-based mechanisms can also be enforced in a fine-grained manner [299, 300], where tokens can be given to subjects on-the-fly containing the appropriate security label, and also be verifiable and unlinkable to preserve privacy [301]. Ultimately, it would be interesting to observe the difference in enforcement when using other approaches, such as capability-based access control, and the impact on authorisation times and efficiency.

8.6 Summary

It is clear that the shift towards a digital calibration paradigm presents itself with a novel access control challenge when there is consideration given to the calibration of rapidly adopted safety-critical IoT systems. Upon discussion of the current state-of-the-art in calibration traceability, the information flow through a systems calibration hierarchy is unique and cannot be solved using any of the traditional access control models alone. This results in the proposal of a novel access control model which unifies the BLP, BIBA and Chinese Wall models. Furthermore, by developing an authorisation framework to evaluate the performance of our model for safety-critical IoT systems, it is shown that authorisation times can suitably enforce restrictions

that enable efficient, safe calibration traceability which can scale for robotic systems and IoT.

9 — Conclusion

In this thesis, it is clear that the area of robotics security is a constantly evolving landscape due to the increase in installations across a wide array of disciplines. As advances continue to be made in the fields of computer science, AI and engineering, so will the complexity of robotic systems and the associated security landscape.

Prior to this research, little focus had been given to the capabilities of a passive adversary, particularly to the impact passive attacks can have on people and organisations and not just the robots themselves. In seeking to address this knowledge gap, three novel passive side channel attacks, in both the cyber and physical domains, were proposed and carried out. In all cases, these attacks primarily target operational confidentiality. In the context of surgical robotics for example, capturing this information could lead to the identification of surgical procedures and in combination with other meta-data (e.g. patient admission and exit times), this could lead to the compromise of patient privacy. First, the traffic analysis side channel in the cyber domain was explored, exploiting traffic characteristics captured by an attacker eavesdropping on the communication between a robot and its controller, with the aim of fingerprinting (classifying) a robot's fundamental movements and workflows. Upon evaluation, the attack was able to achieve this with around 60% accuracy. While Tor was proposed as a defence to this attack, demonstrating a significant drop in accuracy, in realistic settings the use of Tor may be less than desirable due to potential timing-related overheads. Thus, as a point of future work in this space, other defences that were proposed aside from Tor,

such as traffic mixing or padding, should be evaluated and compared. The second side channel explored was the acoustic side channel in the physical domain, where sounds produced as a robot moves are processed to extract meaningful features that are used to achieve the same attack. Specifically, such sounds are captured by an insider attacker using a smartphone, with the attack achieving at least 75% accuracy. Furthermore, during the Covid-19 pandemic, discussions surrounding telemedecine and remote medical education (i.e. remote viewing of surgeries) led to a novel use case of VoIP applications being used to enable this. Because of the use of VoIP in recent times, the same attack was used on VoIP audio of robot movements, which showed a much higher success rate (around 15% higher) and opens up new research avenues pertaining to anonymous communication networks in robotic systems. Finally, the third side channel attack explored in this thesis is radio frequency. Using a small antenna situated near a robot in operation, an attacker can successfully fingerprint robot movements with at least 90% accuracy. It is clear that in this thesis, the physical domain attacks perform better than traffic analysis in the cyber domain, which radio frequency performing the best. An interesting point of future work in this space could be combining multiple side channel attacks together (i.e. both the acoustic and radio frequency side channels) to investigate whether this greatly improves fingerprinting accuracy. Ultimately, the results as a whole reported in this first part of the thesis suggest that passive attacks can be just as devastating as active attacks (if not more so) and require urgent attention as well as further investigation.

In addition to the issue of passive attacks in both of the cyber and physical domains, a novel perspective on robotics security is brought

to light in this thesis; dealing with an issue which ultimately underpins the accuracy and safety of these systems – calibration. Upon reviewing the limitations of current state-of-the-art calibration processes, it became clear that cybersecurity is in need of important review as progress is made towards modern digital environments that are required to scale to the Internet and to the ubiquitous nature of IoT and robotics. In this thesis, two solutions are proposed which aim to tackle some immediate fundamental challenges when looking at the progression to digital environments – record keeping and access control. First, record keeping in calibration is highly inadequate and manual, preventing successful scaling to the Internet. Using a public blockchain tested on the Ethereum network, it is clear that using a blockchain can more than adequately meet not only the security requirements, but also match the requirements of scale and efficiency. The second solution involves access control, which addresses an important aspect of allowing only those authorised to interact with relatively appropriate calibration processes, such as writing and reading calibration records. Existing access control models could not satisfy the novel information flows related to calibration traceability, and requires a new model which entails a novel unification of three existing models. Upon evaluation, the unified model significantly outperforms a simple conjunction of existing policies and demonstrably scales well with large, complex calibration chains. The results presented in this thesis around better management of the calibration of robotic systems may be considered to be initial stepping stones towards a new field, which will ultimately encompass IoT and a wide range of Internet-connected robotic systems.

9.1 Summary of Contributions

The summary of the contributions brought by the work presented in this thesis will be discussed in terms of the primary research questions stated earlier in Chapter 1.1. For research question 1, the thesis provides three key examples of passive insider attack which show that an attacker can mount an information leakage attack that can compromise the operational confidentiality of industrial (warehousing) robotic workflows. Further, all three attacks demonstrate that a higher level of granularity can be achieved (speed and distance of robotic movements) for workflow inference, with the radio frequency side channel in the physical domain showing the most success. For research question 2, the thesis examines and showcases the fundamental security requirements that calibration traceability mechanisms should mandate when progressing to digitised environments, specific to safety-critical IoT systems such as robotics systems. The latter two chapters provide two mechanisms for secure record keeping and access control, respectively, which show to meet these requirements and scale to robotic systems and IoT in general. While there may exist other aspects of the calibration ecosystem that require subsequent review in the move to a digitised ecosystem and review of the newly proposed threat landscape in this thesis, the work presented in this domain is the first stepping stone in a new avenue of calibration security.

Bibliography

[1] Karel Capek. *RUR (Rossum's universal robots)*. Penguin, 2004.

[2] Fritz Lang, Thea Von Harbou, Alfred Abel, Gustav Fröhlich, Brigitte Helm, Gottfried Huppertz, and Universität der Künste Berlin. *Metropolis*. Lorrimer Pub., 1973.

[3] Isaac Asimov. Three laws of robotics. *Asimov, I. Runaround*, 1941.

[4] Marvin Minsky and Seymour A Papert. 1968-1969 progress report. 1970.

[5] David Silver. The little robot system. 1973.

[6] Michael E Moran. Evolution of robotic arms. *Journal of robotic surgery*, 1(2):103–111, 2007.

[7] Nils J Nilsson et al. Shakey the robot. 1984.

[8] Hans P Moravec. The stanford cart and the cmu rover. *Proceedings of the IEEE*, 71(7):872–884, 1983.

[9] Reuters. U.s. companies put record number of robots to work in 2018. `https://www.reuters.com/article/us-usa-economy-robots/u-s-companies-put-record-number-of-robots-to-work-in-2018-idUSK` February 2019. Last Accessed: 02/07/2021.

[10] Homa Alemzadeh, Jaishankar Raman, Nancy Leveson, Zbigniew Kalbarczyk, and Ravishankar K Iyer. Adverse events in robotic

surgery: a retrospective study of 14 years of fda data. *PloS one*, 11(4):e0151470, 2016.

[11] Wenyuan Xu, Ke Ma, Wade Trappe, and Yanyong Zhang. Jamming sensor networks: attack and defense strategies. *IEEE network*, 20(3):41–47, 2006.

[12] Aristides Mpitziopoulos, Damianos Gavalas, Charalampos Konstantopoulos, and Grammati Pantziou. A survey on jamming attacks and countermeasures in wsns. *IEEE communications surveys & tutorials*, 11(4):42–56, 2009.

[13] Matthew Jagielski, Alina Oprea, Battista Biggio, Chang Liu, Cristina Nita-Rotaru, and Bo Li. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 19–35. IEEE, 2018.

[14] Anand Handa, Ashu Sharma, and Sandeep K Shukla. Machine learning in cybersecurity: A review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 9(4):e1306, 2019.

[15] Tamara Bonaci, Jeffrey Herron, Tariq Yusuf, Junjie Yan, Tadayoshi Kohno, and Howard Jay Chizeck. To make a robot secure: An experimental analysis of cyber security threats against teleoperated surgical robots. *arXiv preprint arXiv:1504.04339*, 2015.

[16] Nicholas DeMarinis, Stefanie Tellex, Vasileios Kemerlis, George Konidaris, and Rodrigo Fonseca. Scanning the internet for ros: A view of security in robotics research. *arXiv preprint arXiv:1808.03322*, 2018.

[17] Jarrod McClean, Christopher Stull, Charles Farrar, and David Mascarenas. A preliminary cyber-physical security assessment of the robot operating system (ros). In *Unmanned Systems Technology XV*, volume 8741, page 874110. International Society for Optics and Photonics, 2013.

[18] Lav Gupta, Raj Jain, and Gabor Vaszkun. Survey of important issues in uav communication networks. *IEEE Communications Surveys & Tutorials*, 18(2):1123–1152, 2015.

[19] Yulong Zou, Jia Zhu, Xianbin Wang, and Lajos Hanzo. A survey on wireless security: Technical challenges, recent advances, and future trends. *Proceedings of the IEEE*, 104(9):1727–1765, 2016.

[20] Rafael R Teixeira, Igor P Maurell, and Paulo LJ Drews. Security on ros: analyzing and exploiting vulnerabilities of ros-based systems. In *2020 Latin American robotics symposium (LARS), 2020 Brazilian symposium on robotics (SBR) and 2020 workshop on robotics in education (WRE)*, pages 1–6. IEEE, 2020.

[21] Federico Maggi, Davide Quarta, Marcello Pogliani, Mario Polino, Andrea M Zanchettin, and Stefano Zanero. Rogue robots: Testing the limits of an industrial robot's security. *Trend Micro, Politecnico di Milano, Tech. Rep*, pages 1–21, 2017.

[22] Marcello Pogliani, Davide Quarta, Mario Polino, Martino Vittone, Federico Maggi, and Stefano Zanero. Security of controlled manufacturing systems in the connected factory: the case of industrial robots. *Journal of Computer Virology and Hacking Techniques*, 15(3):161–175, 2019.

[23] Philip McCord Morse, Acoustical Society of America, and American Institute of Physics. *Vibration and sound*, volume 2. McGraw-Hill New York, 1948.

[24] Thomas D Rossing and Neville H Fletcher. Principles of vibration and sound, 2004.

[25] Oliver J Muensterer, Martin Lacher, Christoph Zoeller, Matthew Bronstein, and Joachim Kübler. Google glass in pediatric surgery: an exploratory study. *International journal of surgery*, 12(4):281–289, 2014.

[26] Sushmita Kulkarni, Dattaprasad A Torse, and Deepak Kulkarni. A cloud based medical transcription using speech recognition technologies. *International Research Journal of Engineering and Technology (IRJET)*, 7(5):6160–6163, 2020.

[27] Azamossadat Hosseini, Hamid Moghaddasi, Samad Sajadi, and Mozhgan Karimi. Telesurgery information management systems in university hospitals of tehran. *Archives of Advances in Biosciences*, 4(4), 2013.

[28] William E Cobb, Eric W Garcia, Michael A Temple, Rusty O Baldwin, and Yong C Kim. Physical layer identification of embedded devices using rf-dna fingerprinting. In *2010-Milcom 2010 Military Communications Conference*, pages 2168–2173. IEEE, 2010.

[29] Marc L Salit and Gregory C Turk. Traceability of single-element calibration solutions, 2005.

[30] Harald Müllejans, Willem Zaaiman, and Roberto Galleano. Analysis and mitigation of measurement uncertainties in the

traceability chain for the calibration of photovoltaic devices. *Measurement Science and Technology*, 20(7):075101, 2009.

[31] Siegfried Hackel, Frank Härtig, Julia Hornig, and Thomas Wiedenhöfer. The digital calibration certificate. *Metrologie für die Digitalisierung von Wirtschaft und Gesellschaft*, pages 75–82, 2017.

[32] Tuukka Mustapää, Pekka Nikander, Daniel Hutzschenreuter, and Raine Viitala. Metrological challenges in collaborative sensing: Applicability of digital calibration certificates. *Sensors*, 20(17):4730, 2020.

[33] Kenneth J Biba. Integrity considerations for secure computer systems. Technical report, MITRE CORP BEDFORD MA, 1977.

[34] David E Bell. Secure computer systems: Mathematical foundations and model. *Technical Report ESD-TR-73-278-1*, 1973.

[35] David FC Brewer and Micheal J Nash. The chinese wall security policy. In *null*, page 206. IEEE, 1989.

[36] Boston Consulting Group. Bcg: Autonomous car market to hit 42 billion by 2025. https://www.consultancy.uk/news/2065/bcg-autonomous-car-market-to-hit-42-billion-by-2025, 2015.

[37] Sagar Behere and Martin Torngren. A functional architecture for autonomous driving. In *2015 First International Workshop on Automotive Software Architecture (WASA)*, pages 3–10. IEEE, 2015.

[38] Mohammad Farsi, Karl Ratcliff, and Manuel Barbosa. An overview of controller area network. *Computing & Control Engineering Journal*, 10(3):113–120, 1999.

[39] Bogdan Groza and Pal-Stefan Murvay. Efficient intrusion detection with bloom filtering in controller area networks. *IEEE Transactions on Information Forensics and Security*, 14(4):1037–1051, 2019.

[40] Wenhao Zong, Changzhu Zhang, Zhuping Wang, Jin Zhu, and Qijun Chen. Architecture design and implementation of an autonomous vehicle. *IEEE Access*, 6:21956–21970, 2018.

[41] Russell H Taylor, Leo Joskowicz, Bill Williamson, André Guéziec, Alan Kalvin, Peter Kazanzides, Robert Van Vorhis, Jianhua Yao, Rajesh Kumar, Andrew Bzostek, et al. Computer-integrated revision total hip replacement surgery: concept and preliminary results. *Medical image analysis*, 3(3):301–319, 1999.

[42] William L Bargar, André Bauer, and Martin Börner. Primary and revision total hip replacement using the robodoc (r) system. *Clinical Orthopaedics and Related Research (1976-2007)*, 354:82–91, 1998.

[43] Gyung Tak Sung and Inderbir S Gill. Robotic laparoscopic surgery: a comparison of the da vinci and zeus systems. *Urology*, 58(6):893–898, 2001.

[44] Ashutosh Tewari, James Peabody, Richard Sarle, Guruswami Balakrishnan, Ashok Hemal, Alok Shrivastava, and Mani Menon. Technique of da vinci robot-assisted anatomic radical prostatectomy. *Urology*, 60(4):569–572, 2002.

[45] Blake Hannaford, Jacob Rosen, Diana W Friedman, Hawkeye King, Phillip Roan, Lei Cheng, Daniel Glozman, Ji Ma, Sina Nia Kosari, and Lee White. Raven-ii: an open platform for surgical robotics research. *IEEE Transactions on Biomedical Engineering*, 60(4):954–959, 2012.

[46] N Vimal Kumar and C Selva Kumar. Development of collision free path planning algorithm for warehouse mobile robot. *Procedia computer science*, 133:456–463, 2018.

[47] Ali Bolu and Ömer Korçak. Adaptive task planning for multi-robot smart warehouse. *IEEE Access*, 9:27346–27358, 2021.

[48] Andrzej Grabowski, Jarosław Jankowski, and Mieszko Wodzyński. Teleoperated mobile robot with two arms: the influence of a human-machine interface, vr training and operator age. *International Journal of Human-Computer Studies*, 156:102707, 2021.

[49] Claudia González, J Ernesto Solanes, Adolfo Munoz, Luis Gracia, Vicent Girbés-Juan, and Josep Tornero. Advanced teleoperation and control system for industrial robots based on augmented virtuality and haptic feedback. *Journal of Manufacturing Systems*, 59:283–298, 2021.

[50] Michal Bartoš, Vladimír Bulej, Martin Bohušík, Ján Stanček, Vitalii Ivanov, and Peter Macek. An overview of robot applications in automotive industry. *Transportation Research Procedia*, 55:837–844, 2021.

[51] Davide Quarta, Marcello Pogliani, Mario Polino, Federico Maggi, Andrea Maria Zanchettin, and Stefano Zanero. An ex-

perimental security analysis of an industrial robot controller. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 268–286. IEEE, 2017.

[52] Chadrick R Evans, Melissa G Medina, and Anthony Michael Dwyer. Telemedicine and telerobotics: from science fiction to reality. *Updates in surgery*, 70(3):357–362, 2018.

[53] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, Andrew Y Ng, et al. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan, 2009.

[54] U.S. Department of Health and Human Services (HHS). Guidance to render unsecured protected health information unusable, unreadable, or indecipherable to unauthorized individuals. `https://www.hhs.gov/hipaa/for-professionals/breach-notification/guidance/index.html`, 2013.

[55] Mike Cable. Calibration: A technician's guide. 2005.

[56] Feiyan Guo, Yongfeng Hou, Qingdong Xiao, Xuerui Zhang, Shihong Xiao, and Zhongqi Wang. Reliability improvement on assembly accuracy with maximum out-of-tolerance probability analysis and prior precise repair optimization. *Advanced Engineering Informatics*, 55:101866, 2023.

[57] Sharon E Navard. Evaluating the effect of accuracy ratios on the percent of calibrations which are out of tolerance. *Houston Univ., NASA (ASEE Summer Faculty Fellowship Program, 1990, Volume 2*, 1990.

[58] Girish Vaidya, TV Prabhakar, NITHISH Gnani, Ryan Shah, and Shishir Nagaraja. Sensor identification via acoustic physically unclonable function (puf). *Digital Threats: Research and Practice*, 2021.

[59] Wenyuan Xu, Wade Trappe, Yanyong Zhang, and Timothy Wood. The feasibility of launching and detecting jamming attacks in wireless networks. In *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*, pages 46–57, 2005.

[60] Gang Zhou, Tian He, John A Stankovic, and Tarek Abdelzaher. Rid: Radio interference detection in wireless sensor networks. In *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, volume 2, pages 891–901. IEEE, 2005.

[61] Bastian Schwarz, Gunnar Ritt, Michael Koerber, and Bernd Eberle. Laser-induced damage threshold of camera sensors and micro-optoelectromechanical systems. *Optical Engineering*, 56(3):034108–034108, 2017.

[62] Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Z Morley Mao. Adversarial sensor attack on lidar-based perception in autonomous driving. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, pages 2267–2281, 2019.

[63] Daniel Kruse, John T Wen, and Richard J Radke. A sensor-based dual-arm tele-robotic system. *IEEE Transactions on Au-*

tomation Science and Engineering, 12(1):4–18, 2014.

[64] Homa Alemzadeh, Daniel Chen, Xiao Li, Thenkurussi Kesavadas, Zbigniew T Kalbarczyk, and Ravishankar K Iyer. Targeted attacks on teleoperated surgical robots: Dynamic model-based detection and mitigation. In *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pages 395–406. IEEE, 2016.

[65] Ted Krovetz. Umac: Message authentication code using universal hashing. Technical report, 2006.

[66] Eric Wagner, Martin Serror, Klaus Wehrle, and Martin Henze. Bp-mac: Fast authentication for short messages. In *Proceedings of the 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 201–206, 2022.

[67] Keita Emura and Takuya Hayashi. A revocable group signature scheme with scalability from simple assumptions and its implementation. In *International Conference on Information Security*, pages 442–460. Springer, 2018.

[68] Léo Ducas, Eike Kiltz, Tancrede Lepoint, Vadim Lyubashevsky, Peter Schwabe, Gregor Seiler, and Damien Stehlé. Crystals-dilithium: A lattice-based digital signature scheme. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 238–268, 2018.

[69] Erdem Alkim, Paulo SLM Barreto, Nina Bindel, Juliane Krämer, Patrick Longa, and Jefferson E Ricardini. The lattice-based digital signature scheme qtesla. In *International Con-*

ference on Applied Cryptography and Network Security, pages 441–460. Springer, 2020.

[70] Yansong Gao, Said F Al-Sarawi, and Derek Abbott. Physical unclonable functions. *Nature Electronics*, 3(2):81–91, 2020.

[71] THE EUROPEAN PARLIAMENT and THE COUNCIL OF THE EUROPEAN UNION. "medical device regulations – official journal of the european union. `https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32017R0745`, April 2017.

[72] Kaspar Rosager Ludvigsen and Shishir Nagaraja. Dissecting liabilities in adversarial surgical robot failures: A national (danish) and european law perspective. *arXiv preprint arXiv:2008.07381*, 2020.

[73] David Cooper (NIST) Kerry McKay (NIST). Guidelines for the selection, configuration, and use of transport layer security (tls) implementations (sp 800-52 rev. 2). `https://csrc.nist.gov/publications/detail/sp/800-52/rev-2/final`, August 2019.

[74] Zigang Cao, Gang Xiong, Yong Zhao, Zhenzhen Li, and Li Guo. A survey on encrypted traffic classification. In *International Conference on Applications and Techniques in Information Security*, pages 73–81. Springer, 2014.

[75] Petr Velan, Milan Čermák, Pavel Čeleda, and Martin Drašar. A survey of methods for encrypted traffic classification and analysis. *International Journal of Network Management*, 25(5):355–374, 2015.

[76] Shahbaz Rezaei and Xin Liu. Deep learning for encrypted traffic classification: An overview. *IEEE communications magazine*, 57(5):76–81, 2019.

[77] Nizar Msadek, Ridha Soua, and Thomas Engel. Iot device fingerprinting: Machine learning based encrypted traffic analysis. In *2019 IEEE wireless communications and networking conference (WCNC)*, pages 1–8. IEEE, 2019.

[78] Chenggang Wang, Sean Kennedy, Haipeng Li, King Hudson, Gowtham Atluri, Xuetao Wei, Wenhai Sun, and Boyang Wang. Fingerprinting encrypted voice traffic on smart speakers with deep learning. In *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 254–265, 2020.

[79] Batyr Charyyev and Mehmet Hadi Gunes. Iot event classification based on network traffic. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 854–859. IEEE, 2020.

[80] Bruhadeshwar Bezawada, Maalvika Bachani, Jordan Peterson, Hossein Shirazi, Indrakshi Ray, and Indrajit Ray. Behavioral fingerprinting of iot devices. In *Proceedings of the 2018 Workshop on Attacks and Solutions in Hardware Security*, pages 41–50, 2018.

[81] Savio Sciancalepore, Omar Adel Ibrahim, Gabriele Oligeri, and Roberto Di Pietro. Detecting drones status via encrypted traffic analysis. In *Proceedings of the ACM Workshop on Wireless Security and Machine Learning*, pages 67–72, 2019.

[82] Marc Juárez, Mohsen Imani, Mike Perry, Claudia Dıaz, and Matthew Wright. Wtf-pad: toward an efficient website fingerprinting defense for tor. *CoRR, abs/1512.00524*, 2015.

[83] Jiajun Gong and Tao Wang. Zero-delay lightweight defenses against website fingerprinting. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pages 717–734, 2020.

[84] uFactory. Ufactory – uarm. `https://www.ufactory.cc/pages/uarm`. Last Accessed: 27/06/2022.

[85] uARM Developer. uarm python sdk. `https://github.com/uArm-Developer/uArm-Python-SDK`. Last Accessed: 27/06/2022.

[86] Quan Zeng, Shou-Jun Zhou, Hao Shen, and Cheng Wang. A network communication protocols for robotic-assisted vascular intervention systems. In *2017 2nd International Conference on Biological Sciences and Technology (BST 2017)*, pages 87–96. Atlantis Press, 2018.

[87] Chris Sanders. *Practical Packet Analysis, 3E: Using Wireshark to Solve Real-World Network Problems*. No Starch Press, 2017.

[88] Brent Chun, David Culler, Timothy Roscoe, Andy Bavier, Larry Peterson, Mike Wawrzoniak, and Mic Bowman. Planetlab: an overlay testbed for broad-coverage services. *ACM SIGCOMM Computer Communication Review*, 33(3):3–12, 2003.

[89] Terry Benzel, Robert Braden, Dongho Kim, Anthony D Joseph, B Clifford Neuman, Ron Ostrenga, Stephen Schwab, and Keith Sklower. Design, deployment, and use of the deter testbed. In *DETER*, 2007.

[90] Terry Benzel, Bob Braden, Ted Faber, Jelena Mirkovic, Steve Schwab, Karen Sollins, and John Wroclawski. Current developments in deter cybersecurity testbed technology. In *2009 Cybersecurity Applications & Technology Conference for Homeland Security*, pages 57–70. IEEE, 2009.

[91] David Johnson, Tim Stack, Russ Fish, Daniel Montrallo Flickinger, Leigh Stoller, Robert Ricci, and Jay Lepreau. Mobile emulab: A robotic wireless and sensor network testbed. In *Proceedings IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications*, pages 1–12. IEEE, 2006.

[92] George F Riley and Thomas R Henderson. The ns-3 network simulator. In *Modeling and tools for network simulation*, pages 15–34. Springer, 2010.

[93] Karamjeet Kaur, Japinder Singh, and Navtej Singh Ghumman. Mininet as software defined networking testing platform. In *International Conference on Communication, Computing & Systems (ICCCS)*, pages 139–42, 2014.

[94] Ramon R Fontes, Samira Afzal, Samuel HB Brito, Mateus AS Santos, and Christian Esteve Rothenberg. Mininet-wifi: Emulating software-defined wireless networks. In *2015 11th International Conference on Network and Service Management (CNSM)*, pages 384–389. IEEE, 2015.

[95] Vern Paxson and Sally Floyd. Wide-area traffic: the failure of poisson modeling. *ACM SIGCOMM Computer Communication Review*, 24(4):257–268, 1994.

[96] Neil Spring, Larry Peterson, Andy Bavier, and Vivek Pai. Using planetlab for network research: myths, realities, and best practices. *ACM SIGOPS Operating Systems Review*, 40(1):17–24, 2006.

[97] Andrzej Chydzinski. Queues with the dropping function and non-poisson arrivals. *IEEE Access*, 8:39819–39829, 2020.

[98] Ioannis D Moscholios and Michael Logothetis. *Efficient multi-rate teletraffic loss models beyond Erlang*. Wiley Online Library, 2019.

[99] Mohammad Lotfollahi, Mahdi Jafari Siavoshani, Ramin Shirali Hossein Zade, and Mohammdsadegh Saberian. Deep packet: A novel approach for encrypted traffic classification using deep learning. *Soft Computing*, 24(3):1999–2012, 2020.

[100] Wei Wang, Ming Zhu, Jinlin Wang, Xuewen Zeng, and Zhongzhen Yang. End-to-end encrypted traffic classification with one-dimensional convolution neural networks. In *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pages 43–48. IEEE, 2017.

[101] Shishir Nagaraja and Ryan Shah. Voiploc: Passive voip call provenance via acoustic side-channels. In *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 323–334, 2021.

[102] Giuseppe Aceto, Domenico Ciuonzo, Antonio Montieri, and Antonio Pescapé. Toward effective mobile encrypted traffic classification through deep learning. *Neurocomputing*, 409:306–315, 2020.

[103] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[104] François Chollet et al. Keras. `https://keras.io`, 2015.

[105] Jason Brownlee. How to grid search hyperparameters for deep learning models in python with keras. *línea]. Disponible en: https://machinelearningmastery. com/grid-search-hyperparameters-deep-learning-models-python-keras*, 2016.

[106] Muhammad Adnan, Alaa Abdul Salam Alarood, M Irfan Uddin, and Izaz ur Rehman. Utilizing grid search cross-validation with adaptive boosting for augmenting performance of machine learning models. *PeerJ Computer Science*, 8:e803, 2022.

[107] Konstantin Eckle and Johannes Schmidt-Hieber. A comparison of deep networks with relu activation function and linear spline-type methods. *Neural Networks*, 110:232–242, 2019.

[108] Howard B Demuth, Mark H Beale, Orlando De Jess, and Martin T Hagan. *Neural network design*. Martin Hagan, 2014.

[109] Rob A Dunne and Norm A Campbell. On the pairing of the softmax activation and cross-entropy penalty functions and the derivation of the softmax activation function. In *Proc. 8th Aust. Conf. on the Neural Networks, Melbourne*, volume 181, page 185. Citeseer, 1997.

[110] Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in neural information processing systems*, pages 8778–8788, 2018.

[111] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[112] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[113] Yuanzhi Li and Yang Yuan. Convergence analysis of two-layer neural networks with relu activation. In *Advances in neural information processing systems*, pages 597–607, 2017.

[114] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

[115] Deep Medhi and Karthik Ramasamy. *Network routing: algorithms, protocols, and architectures*. Morgan Kaufmann, 2017.

[116] Samuel Muhizi, Gregory Shamshin, Ammar Muthanna, Ruslan Kirichek, Andrei Vladyko, and Andrey Koucheryavy. Analysis and performance evaluation of sdn queue model. In *International Conference on Wired/Wireless Internet Communication*, pages 26–37. Springer, 2017.

[117] Hamza Dahmouni, André Girard, and Brunilde Sansó. Analytical jitter model for ip network planning and design. In *2009 First International Conference on Communications and Networking*, pages 1–7. IEEE, 2009.

[118] Changle Li, Jiandong Li, and Xuelian Cai. Performance evaluation of ieee 802.11 wlan-high speed packet wireless data network for supporting voice service. In *2004 IEEE Wireless Communications and Networking Conference (IEEE Cat. No. 04TH8733)*, volume 3, pages 1463–1468. IEEE, 2004.

[119] Adnan Huremovic, Mesud Hadzialic, and Fatima Skaka. Analytical model for jitter in networks with ipp traffic. In *2014 International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 334–339. IEEE, 2014.

[120] Michel Mandjes, Kees van der Wal, Rob Kooij, and Harrie Bastiaansen. End-to-end delay models for interactive services on a large-scale ip network. In *Proceedings of the 7th workshop on performance modelling and evaluation of ATM & IP networks (IFIP99)*, pages 28–30. Citeseer, 1999.

[121] Jochen Kögel. *One-way delay measurement based on flow data in large enterprise networks.* Inst. für Kommunikationsnetze und Rechnersysteme, 2013.

[122] Lisa Yan and Nick McKeown. Learning networking by reproducing research results. *ACM SIGCOMM Computer Communication Review*, 47(2):19–26, 2017.

[123] Kenneth C Mansfield Jr and James L Antonakos. *Computer networking for LANS to WANS: hardware, software and security.* Cengage Learning, 2009.

[124] Pal Varga. Analyzing packet interarrival times distribution to detect network bottlenecks. In *EUNICE 2005: Networks and*

Applications Towards a Ubiquitously Connected World, pages 17–29. Springer, 2006.

[125] Gelu-Ovidiu Tirian. Automation of a warehouse by means of a robotic arm. *Annals of Faculty Engineering Hundeoara–International Journal of Engineering*, 11, 2013.

[126] Nobutaka Kimura, Kiyoto Ito, Taiki Fuji, Keisuke Fujimoto, Kanako Esaki, Fumiko Beniyama, and Toshio Moriya. Mobile dual-arm robot for automated order picking system in warehouse containing various kinds of products. In *2015 IEEE/SICE International Symposium on System Integration (SII)*, pages 332–338. IEEE, 2015.

[127] Robert Bogue. Growth in e-commerce boosts innovation in the warehouse robot market. *Industrial Robot: An International Journal*, 2016.

[128] Yinghan Wang, Yi Liang, Diansheng Chen, Yongzhan Liu, and Min Wang. A goods sorting robot system for e-commerce logistics warehouse based on robotic arm technology. In *2020 IEEE International Conference on Real-time Computing and Robotics (RCAR)*, pages 310–314. IEEE, 2020.

[129] Athanasios S Polydoros and Lazaros Nalpantidis. A reservoir computing approach for learning forward dynamics of industrial manipulators. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 612–618. IEEE, 2016.

[130] Elmar Rueckert, Moritz Nakatenus, Samuele Tosatto, and Jan Peters. Learning inverse dynamics models in o (n) time with

lstm networks. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 811–816. IEEE, 2017.

[131] Darrell Etherington @ Tech Crunch. Mit and boston dynamics team up on "dr. spot," a robot for remote covid-19 vital sign measurement. https://techcrunch.com/2020/08/19/mit-and-boston-dynamics-team-up-on-dr-spot-a-robot-for-remote-covid-19-vital-si August 19 2020.

[132] Hen-Wei Huang, Claas Ehmke, Gene Merewether, Fara Dadabhoy, Annie Feng, Akhil John Thomas, Canchen Li, Marco da Silva, Marc H Raibert, Edward W Boyer, et al. Agile mobile robotic platform for contactless vital signs monitoring. 2020.

[133] Simon DiMaio, Mike Hanuschik, and Usha Kreaden. The da vinci surgical system. In *Surgical robotics*, pages 199–217. Springer, 2011.

[134] Max B Schäfer, Kent W Stewart, and Peter P Pott. Industrial robots for teleoperated surgery–a systematic review of existing approaches. *Current Directions in Biomedical Engineering*, 5(1):153–156, 2019.

[135] International Electrotechnical Commission. Iec 62443-4-2:2019 – security for industrial automation and control systems – part 4-2: Technical security requirements for iacs components. Technical report, 2019.

[136] Centre for the Protection of National Infrastructure (CPNI). Cyber assurance of physical security systems (capss) 2022 - security characteristic. Technical report, 2022.

[137] European Commission GROW.R.2.DIR. Mdcg 2019-16 – guidance on cybersecurity for medical devices. https://ec.europa.eu/docsroom/documents/41863, June 2020.

[138] Elisabetta Biasin and Erik Kamenjasevic. Cybersecurity of medical devices: Regulatory challenges in the eu. 2020.

[139] Department of Health and Social Care. Information security management: Nhs code of practice. https://www.gov.uk/government/publications/information-security-management-nhs-code-of-practice, April 2007.

[140] Data Protection Act. Data protection act. *London Station Off*, 5, 1998.

[141] NHS England Specialised Commissioning Team. Clinical commissioning policy: Robotic-assisted surgical procedures for prostate cancer (b14/p/a), July 2015.

[142] Hyungjun Kim, Arseniy Vitkovskiy, Paul V Gratz, and Vassos Soteriou. Use it or lose it: Wear-out and lifetime in future chip multiprocessors. In *Proceedings of the 46th Annual IEEE/ACM International Symposium on Microarchitecture*, pages 136–147, 2013.

[143] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[144] Paulo Calvo, Jose Guevara-Coto, and Adrian Lara. Classifying and understanding tor traffic using tree-based models. In *2020*

IEEE Latin-American Conference on Communications (LATINCOM), pages 1–6. IEEE, 2020.

[145] Florian Tschorsch and Björn Scheuermann. Mind the gap: Towards a {Backpressure-Based} transport protocol for the tor network. In *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*, pages 597–610, 2016.

[146] Simone Fischer-Hübner and Nicholas Hopper. *Privacy Enhancing Technologies: 11th International Symposium, PETS 2011, Waterloo, ON, Canada, July 27-29, 2011, Proceedings*, volume 6794. Springer Science & Business Media, 2011.

[147] Claude E Shannon. Communication theory of secrecy systems. *The Bell system technical journal*, 28(4):656–715, 1949.

[148] Xinwen Fu, Bryan Graham, Riccardo Bettati, and Wei Zhao. On countermeasures to traffic analysis attacks. In *IEEE Systems, Man and Cybernetics SocietyInformation Assurance Workshop, 2003.*, pages 188–195. IEEE, 2003.

[149] Kevin P Dyer, Scott E Coull, Thomas Ristenpart, and Thomas Shrimpton. Peek-a-boo, i still see you: Why efficient traffic analysis countermeasures fail. In *2012 IEEE symposium on security and privacy*, pages 332–346. IEEE, 2012.

[150] Jonas Bushart and Christian Rossow. Padding ain't enough: Assessing the privacy guarantees of encrypted {DNS}. In *10th {USENIX} Workshop on Free and Open Communications on the Internet ({FOCI} 20)*, 2020.

[151] Riyad Alshammari and A Nur Zincir-Heywood. Can encrypted traffic be identified without port numbers, ip addresses and payload inspection? *Computer networks*, 55(6):1326–1350, 2011.

[152] Subhabrata Sen, Oliver Spatscheck, and Dongmei Wang. Accurate, scalable in-network identification of p2p traffic using application signatures. In *Proceedings of the 13th international conference on World Wide Web*, pages 512–521, 2004.

[153] Jawad Khalife, Amjad Hajjar, and Jesus Diaz-Verdejo. A multilevel taxonomy and requirements for an optimal traffic-classification model. *International Journal of Network Management*, 24(2):101–120, 2014.

[154] Andrew W Moore and Denis Zuev. Internet traffic classification using bayesian analysis techniques. In *Proceedings of the 2005 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pages 50–60, 2005.

[155] Se Eun Oh, Saikrishna Sunkam, and Nicholas Hopper. p1-fp: Extraction, classification, and prediction of website fingerprints with deep learning. *Proceedings on Privacy Enhancing Technologies*, 2019(3):191–209, 2019.

[156] Qiang Xu, Rong Zheng, Walid Saad, and Zhu Han. Device fingerprinting in wireless networks: Challenges and opportunities. *IEEE Communications Surveys & Tutorials*, 18(1):94–104, 2015.

[157] Kota Abe and Shigeki Goto. Fingerprinting attack on tor anonymity using deep learning. *Proceedings of the Asia-Pacific Advanced Network*, 42:15–20, 2016.

[158] Giuseppe Aceto, Domenico Ciuonzo, Antonio Montieri, and Antonio Pescapé. Mobile encrypted traffic classification using deep learning. In *2018 Network Traffic Measurement and Analysis Conference (TMA)*, pages 1–8. IEEE, 2018.

[159] Ying Yang, Cuicui Kang, Gaopeng Gou, Zhen Li, and Gang Xiong. Tls/ssl encrypted traffic classification with autoencoder and convolutional neural network. In *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pages 362–369. IEEE, 2018.

[160] Chen Song, Feng Lin, Zhongjie Ba, Kui Ren, Chi Zhou, and Wenyao Xu. My smartphone knows what you print: Exploring smartphone-based side-channel attacks against 3d printers. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 895–907, 2016.

[161] Sujit Rokka Chhetri, Arquimedes Canedo, and Mohammad Abdullah Al Faruque. Confidentiality breach through acoustic side-channel in cyber-physical additive manufacturing systems. *ACM Transactions on Cyber-Physical Systems*, 2(1):1–25, 2017.

[162] Sriram Sami, Yimin Dai, Sean Rui Xiang Tan, Nirupam Roy, and Jun Han. Spying with your robot vacuum cleaner: eavesdropping via lidar sensors. In *Proceedings of the 18th Conference on Embedded Networked Sensor Systems*, pages 354–367, 2020.

[163] Jacob Gatlin, Sofia Belikovetsky, Yuval Elovici, Anthony Skjellum, Joshua Lubell, Paul Witherell, and Mark Yampolskiy.

Encryption is futile: Reconstructing 3d-printed models using the power side-channel. In *24th International Symposium on Research in Attacks, Intrusions and Defenses*, pages 135–147, 2021.

[164] Frank Fahy and David Thompson. *Fundamentals of sound and vibration*. CRC press, 2015.

[165] Gabriele Trovato, Renato Paredes, Javier Balvin, Francisco Cuellar, Nicolai Bæk Thomsen, Soren Bech, and Zheng-Hua Tan. The sound or silence: investigating the influence of robot noise on proxemics. In *2018 27th IEEE international symposium on robot and human interactive communication (RO-MAN)*, pages 713–718. IEEE, 2018.

[166] Yinni Peng and Susanne YP Choi. Mobile phone use among migrant factory workers in south china: Technologies of power and resistance. *The China Quarterly*, 215:553–571, 2013.

[167] Tomas J Saun, Kevin J Zuo, and Teodor P Grantcharov. Video technologies for recording open surgery: a systematic review. *Surgical innovation*, 26(5):599–612, 2019.

[168] Vodafone. 5gdig: The winners – from skype for surgeons to ar. `https://www.vodafone.com/news/technology/5gdig-winners-2019-supponor-ar-surgeonmate`, July 2019.

[169] A. Al-Jabir, A. Kerwan, M. Nicola, Z. Alsafi, M. Khan, C. Sohrabi, N. O'Neill, C. Iosifidis, M. Griffin, G. Mathew, and R. Agha. Impact of the Coronavirus (COVID-19) pandemic on surgical practice - Part 1. *Int J Surg*, 79:168–179, Jul 2020.

[170] ASHM Dalen, J Legemaate, WS Schlack, DA Legemate, and MP Schijven. Legal perspectives on black box recording devices in the operating environment. *Journal of British Surgery*, 106(11):1433–1441, 2019.

[171] Pangratios Papacosta and Nathan Linscheid. The confirmation of the inverse square law using diffraction gratings. *The Physics Teacher*, 52(4):243–245, 2014.

[172] Ashish Mhaske, Jameer Manur, and Prakash Arumugasamy. Verifying the inverse square law using wifi signals. 2021.

[173] Jean-Marc Valin, Koen Vos, and Timothy Terriberry. Definition of the opus audio codec. Technical report, 2012.

[174] Jean-Marc Valin, Gregory Maxwell, Timothy B Terriberry, and Koen Vos. High-quality, low-delay music coding in the opus codec. *arXiv preprint arXiv:1602.04845*, 2016.

[175] RG Bachu, S Kopparthi, B Adapa, and BD Barkana. Separation of voiced and unvoiced using zero crossing rate and energy of the speech signal. In *American Society for Engineering Education (ASEE) zone conference proceedings*, pages 1–7. American Society for Engineering Education, 2008.

[176] Costas Panagiotakis and Georgios Tziritas. A speech/music discriminator based on rms and zero-crossings. *IEEE Transactions on multimedia*, 7(1):155–166, 2005.

[177] Phu Ngoc Le, Eliathamby Ambikairajah, Julien Epps, Vidhyasaharan Sethu, and Eric HC Choi. Investigation of spectral centroid features for cognitive load classification. *Speech Communication*, 53(4):540–551, 2011.

[178] Anssi Klapuri and Manuel Davy. Signal processing methods for music transcription. 2007.

[179] Yunus Atahan, Ahmet Elbir, Abdullah Enes Keskin, Osman Kiraz, Bulent Kirval, and Nizamettin Aydin. Music genre classification using acoustic features and autoencoders. In *2021 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pages 1–5. IEEE, 2021.

[180] Marko Kos, Zdravko Kačič, and Damjan Vlaj. Acoustic classification and segmentation using modified spectral roll-off and variance-based features. *Digital Signal Processing*, 23(2):659–674, 2013.

[181] Dan-Ning Jiang, Lie Lu, Hong-Jiang Zhang, Jian-Hua Tao, and Lian-Hong Cai. Music type classification by spectral contrast feature. In *Proceedings. IEEE International Conference on Multimedia and Expo*, volume 1, pages 113–116. IEEE, 2002.

[182] Meinard Müller. Fundamentals of music processing. 2015.

[183] Taemin Cho and Juan P Bello. On the relative importance of individual components of chord recognition systems. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(2):477–492, 2013.

[184] Jouni Paulus, Meinard Müller, and Anssi Klapuri. State of the art report: Audio-based music structure analysis. In *Ismir*, pages 625–636. Utrecht, 2010.

[185] Donald D Greenwood. The mel scale's disqualifying bias and a consistency of pitch-difference equisections in 1956 with equal

cochlear distances and equal frequency ratios. *Hearing research*, 103(1-2):199–224, 1997.

[186] Jorge Martinez, Hector Perez, Enrique Escamilla, and Masahisa Mabo Suzuki. Speaker recognition using mel frequency cepstral coefficients (mfcc) and vector quantization (vq) techniques. In *CONIELECOMP 2012, 22nd International Conference on Electrical Communications and Computers*, pages 248–251. IEEE, 2012.

[187] Martín Ortega Ortega, Gustavo Chafla Altamirano, and Mara Falconí Abad. Evaluation of the voice quality and qos in real calls using different voice over ip codecs. In *2018 IEEE Colombian Conference on Communications and Computing (COLCOM)*, pages 1–6. IEEE, 2018.

[188] Asif Ali Laghari, Rashid Ali Laghari, Asif Ali Wagan, and Aamir Iqbal Umrani. Effect of packet loss and reorder on quality of audio streaming. *EAI Endorsed Transactions on Scalable Information Systems*, 7(25), 2020.

[189] Brian McFee, Colin Raffel, Dawen Liang, Daniel P Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8, pages 18–25. Citeseer, 2015.

[190] K Sreenivasa Rao and Anil Kumar Vuppala. *Speech processing in mobile environments*. Springer, 2014.

[191] Saptarshi Sinha, Hiroki Ohashi, and Katsuyuki Nakamura. Class-wise difficulty-balanced loss for solving class-imbalance. In *Proceedings of the Asian conference on computer vision*, 2020.

[192] Bruno Rolf. Graphs to prof. sommerfeld's attenuation formula for radio waves. *Proceedings of the Institute of Radio Engineers*, 18(3):391–402, 1930.

[193] Jim H James, Bing Chen, and Laurie Garrison. Implementing voip: a voice transmission performance progress report. *IEEE Communications Magazine*, 42(7):36–41, 2004.

[194] Parvaneh Amirzade Dana, Zahra Esmaeilbeig, and Mohammad-Reza Sadeghi. Reliability enhancement and packet loss recovery of any steganographic method in voice over ip. *Wireless Networks*, 26(8):5817–5823, 2020.

[195] Anssi Rämö and Henri Toukomaa. Voice quality characterization of ietf opus codec. In *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[196] Ingrid Neumann and Gerd Schuller. Spectral and temporal gating mechanisms enhance the clutter rejection in the echolocating bat, rhinolophus rouxi. *Journal of comparative physiology A*, 169(1):109–116, 1991.

[197] Joshua M Inouye, Silvia S Blemker, and David I Inouye. Towards undistorted and noise-free speech in an mri scanner: correlation subtraction followed by spectral noise gating. *The Journal of the Acoustical Society of America*, 135(3):1019–1022, 2014.

[198] Jean-Marc Valin. Speex: A free codec for free speech. *arXiv preprint arXiv:1602.08668*, 2016.

[199] Koen Vos, Soeren Jensen, and Karsten Soerensen. Silk speech codec. *IETF draft*, 30, 2010.

[200] Krishan Rajaratnam, Kunal Shah, and Jugal Kalita. Isolated and ensemble audio preprocessing methods for detecting adversarial examples against automatic speech recognition. *arXiv preprint arXiv:1809.04397*, 2018.

[201] Rodolfo Castro. Is your company's network ready for microsoft teams, 2020.

[202] Tanvi D Desai and SR Patil. Experimental and numerical analysis of vibration isolation materials on vibration reduction within plazma torch.

[203] Ian A Gravagne, Christopher D Rahn, and Ian D Walker. Good vibrations: a vibration damping setpoint controller for continuum robots. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, volume 4, pages 3877–3884. IEEE, 2001.

[204] Ameer Hamza Khan and Shuai Li. Sliding mode control with pid sliding surface for active vibration damping of pneumatically actuated soft robots. *IEEE Access*, 8:88793–88800, 2020.

[205] Xinbo Ma, Pak Kin Wong, and Jing Zhao. Practical multi-objective control for automotive semi-active suspension system with nonlinear hydraulic adjustable damper. *Mechanical Systems and Signal Processing*, 117:667–688, 2019.

[206] Michael Backes, Markus Dürmuth, Sebastian Gerling, Manfred Pinkal, Caroline Sporleder, et al. Acoustic {Side-Channel} attacks on printers. In *19th USENIX Security Symposium (USENIX Security 10)*, 2010.

[207] S Abhishek Anand and Nitesh Saxena. A sound for a sound: Mitigating acoustic side channel attacks on password keystrokes with active sounds. In *International Conference on Financial Cryptography and Data Security*, pages 346–364. Springer, 2016.

[208] Younghyun Kim, Woo Suk Lee, Vijay Raghunathan, Niraj K Jha, and Anand Raghunathan. Vibration-based secure side channel for medical devices. In *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, pages 1–6. IEEE, 2015.

[209] Muhammad Bilal. A review of internet of things architecture, technologies and analysis smartphone-based attacks against 3d printers. *arXiv preprint arXiv:1708.04560*, 2017.

[210] Lonce Wyse. Audio spectrogram representations for processing with convolutional neural networks. *arXiv preprint arXiv:1706.09559*, 2017.

[211] Yuni Zeng, Hua Mao, Dezhong Peng, and Zhang Yi. Spectrogram based multi-task audio classification. *Multimedia Tools and Applications*, 78(3):3705–3722, 2019.

[212] Jaime Zabalza, Carmine Clemente, Gaetano Di Caterina, Jinchang Ren, John J Soraghan, and Stephen Marshall. Robust pca micro-doppler classification using svm on embedded systems. *IEEE Transactions on Aerospace and Electronic Systems*, 50(3):2304–2310, 2014.

[213] Ji Ma and Yuyu Yuan. Dimension reduction of image deep feature using pca. *Journal of Visual Communication and Image Representation*, 63:102578, 2019.

[214] J Novakovic and A Veljovic. C-support vector classification: Selection of kernel and parameters in medical diagnosis. In *2011 IEEE 9th international symposium on intelligent systems and informatics*, pages 465–470. IEEE, 2011.

[215] Chuanxia Jian, Jian Gao, and Yinhui Ao. A new sampling method for classifying imbalanced data based on support vector machine ensemble. *Neurocomputing*, 193:115–122, 2016.

[216] Aditya Krishna Menon. Large-scale support vector machines: algorithms and theory. *Research Exam, University of California, San Diego*, 117, 2009.

[217] Emiel Por, Maaike van Kooten, and Vanja Sarkovic. Nyquist–shannon sampling theorem. *Leiden University*, 1:1, 2019.

[218] Prajoy Podder, Tanvir Zaman Khan, Mamdudul Haque Khan, and M Muktadir Rahman. Comparative performance analysis of hamming, hanning and blackman window. *International Journal of Computer Applications*, 96(18), 2014.

[219] A Vigil, M Belkerdid, and D Malocha. Application of classical cosine series window functions to full response signaling offset quadrature binary modulation systems. *IEEE transactions on communications*, 41(1):11–15, 1993.

[220] RL Herndon. Electromagnetic pulse (emp) and tempest protection for facilities. *DC: Army Corps of Engineers Publication Department*, 1990.

[221] Iryna Shopina, Dmytro Khomiakov, Nadiia Khrystynchenko, Serhii Zhukov, and Dmytro Shpenov. Cybersecurity: Legal and organizational support in leading countries, nato and eu standards. *Journal of Security & Sustainability Issues*, 9(3), 2020.

[222] Ofcom. Guidance on emf compliance and enforcement. `https://www.ofcom.org.uk/spectrum/emf/compliance-and-enforcement-guidance`. Last Accessed: 12/09/2022.

[223] International Electrotechnical Commission et al. Industrial, scientific and medical (ism) radio-frequency equipment-electromagnetic disturbance characteristics-limits and methods of measurement. *CISPR 11*, 2003.

[224] James T Graham, Ronald Riley, Rusty Baldwin, and Ashwin Fisher. Block-level algorithm classification based on rf side-channel. In *Cyber Sensing 2018*, volume 10630, pages 44–50. SPIE, 2018.

[225] Asanka Sayakkara, Nhien-An Le-Khac, and Mark Scanlon. Leveraging electromagnetic side-channel analysis for the investigation of iot devices. *Digital Investigation*, 29:S94–S103, 2019.

[226] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[227] Zhicheng Cui, Wenlin Chen, and Yixin Chen. Multi-scale convolutional neural networks for time series classification. *arXiv preprint arXiv:1603.06995*, 2016.

[228] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

[229] Miller L Wilt, Megan M Baker, and Stergios J Papadakis. Toward an rf side-channel reverse engineering tool. In *2020 IEEE Physical Assurance and Inspection of Electronics (PAINE)*, pages 1–7. IEEE, 2020.

[230] Mohammad Abdullah Al Faruque, Sujit Rokka Chhetri, Arquimedes Canedo, et al. Acoustic side-channel attacks on additive manufacturing systems. pages 1–10, 2016.

[231] J-M Caudron, N Ford, M Henkens, C Mace, R Kiddle-Monroe, and J Pinel. Substandard medicines in resource-poor settings: a problem that can no longer be ignored. *Tropical Medicine & International Health*, 13(8):1062–1072, 2008.

[232] Atholl Johnston and David W Holt. Substandard drugs: a potential crisis for public health. *British journal of clinical pharmacology*, 78(2):218–243, 2014.

[233] Jude Nwokike, Aubrey Clark, and Phillip P Nguyen. Medicines quality assurance to fight antimicrobial resistance. *Bulletin of the World Health Organization*, 96(2):135, 2018.

[234] International Organization for Standardization (ISO). Iso 9001. `https://www.iso.org/iso-9001-quality-management.html`, 2017.

[235] International Organization for Standardization (ISO). Iso/iec 17025. `https://www.iso.org/`

ISO-IEC-17025-testing-and-calibration-laboratories.html, 2017.

[236] International Bureau of Weights, Measures, Barry N Taylor, and Ambler Thompson. *The international system of units (SI)*. US Department of Commerce, Technology Administration, National Institute of . . . , 2001.

[237] Thefollowing seven orgonlzatiarls Supportedthe. Guide to the expression of uncertainty in measurement. 1995.

[238] BIPM. Euramet 1187 annex 3 uncertainty budgets. https://www.bipm.org/documents/20126/56401676/EURAMET.EM-S37-Annex+3_Budgets_final.pdf/c1d2757f-c417-ac06-9b3b-0b145e01942d, June 2012. Last Accessed: 17/09/2022.

[239] J Kristiansen and JM Christensen. Traceability and uncertainty in analytical measurements. *Annals of clinical biochemistry*, 35(3):371–379, 1998.

[240] Frank Liebmann. Infrared thermometer calibration. *Cal Lab Int. J. Metrol*, pages 20–22, 2011.

[241] Hui Lin, Homa Alemzadeh, Zbigniew Kalbarczyk, and Ravishankar Iyer. Challenges and opportunities in the detection of safety-critical cyberphysical attacks. *Computer*, 53(3):26–37, 2020.

[242] Vyacheslav Kkarchenko. Big data and internet of things for safety critical applications: Challenges, methodology and industry cases. *International Journal on Information Technologies & Security*, 10(4), 2018.

[243] Monowar Hasan and Sibin Mohan. Protecting actuators in safety-critical iot systems from control spoofing attacks. In *Proceedings of the 2nd International ACM Workshop on Security and Privacy for the Internet-of-Things*, pages 8–14, 2019.

[244] Katrin Neubauer, Sebastian Fischer, and Rudolf Hackenberg. Work in progress: Security analysis for safety-critical systems: Smart grid and iot. In *ARCS Workshop 2019; 32nd International Conference on Architecture of Computing Systems*, pages 1–6. VDE, 2019.

[245] Reuters. U.s. safety agency reviewing 23 tesla crashes, three from recent weeks. `https://www.reuters.com/article/us-tesla-crash-idUSKBN2BA2ML`, 3 2021. Last Accessed: 04/07/2021.

[246] Al Jazeera. Tesla in deadly california crash was on autopilot: Authorities. `https://www.aljazeera.com/economy/2021/5/14/tesla-in-deadly-california-crash-was-on-autopilot-authorities`, 5 2021. Last Accessed: 04/07/2021.

[247] Nikolaos Pitropakis, Emmanouil Panaousis, Thanassis Giannetsos, Eleftherios Anastasiadis, and George Loukas. A taxonomy and survey of attacks against machine learning. *Computer Science Review*, 34:100199, 2019.

[248] Aung Maw, Sridhar Adepu, and Aditya Mathur. Ics-blockops: Blockchain for operational data security in industrial control system. *Pervasive and Mobile Computing*, 59:101048, 2019.

[249] Thomas Sødring, Petter Reinholdtsen, and David Massey. A record-keeping approach to managing iot-data for government agencies. *Records Management Journal*, 2020.

[250] Baddepaka Prasad and S Ramachandram. Decentralized privacy-preserving framework for health care record-keeping over hyperledger fabric. In *Inventive Communication and Computational Technologies*, pages 463–475. Springer, 2020.

[251] Edoardo Gaetani, Leonardo Aniello, Roberto Baldoni, Federico Lombardi, Andrea Margheri, and Vladimiro Sassone. Blockchain-based database to ensure data integrity in cloud computing environments. 2017.

[252] Thomas Page. *The application of hash chains and hash structures to cryptography.* PhD thesis, University of London, 2009.

[253] Chad Teat and Svetlana Peltsverger. The security of cryptographic hashes. In *Proceedings of the 49th Annual Southeast Regional Conference*, pages 103–108, 2011.

[254] Alfred C Chin. Blockchain biology. *Frontiers in Blockchain*, 3:606413, 2020.

[255] Arati Baliga. The blockchain landscape. *Persistent Systems*, 3(5), 2016.

[256] Vanessa Bracamonte and Hitoshi Okada. The issue of user trust in decentralized applications running on blockchain platforms. In *2017 IEEE International Symposium on Technology and Society (ISTAS)*, pages 1–4. IEEE, 2017.

[257] Leo Maxim Bach, Branko Mihaljevic, and Mario Zagar. Comparative analysis of blockchain consensus algorithms. In *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 1545–1550. Ieee, 2018.

[258] P Rajitha Nair and D Ramya Dorai. Evaluation of performance and security of proof of work and proof of stake using blockchain. In *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, pages 279–283. IEEE, 2021.

[259] Gavin Wood et al. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum project yellow paper*, 151(2014):1–32, 2014.

[260] Bhabendu Kumar Mohanta, Soumyashree S Panda, and Debasish Jena. An overview of smart contract and use cases in blockchain technology. In *2018 9th international conference on computing, communication and networking technologies (ICCCNT)*, pages 1–4. IEEE, 2018.

[261] Anna Vacca, Andrea Di Sorbo, Corrado A Visaggio, and Gerardo Canfora. A systematic literature review of blockchain and smart contract development: Techniques, tools, and open challenges. *Journal of Systems and Software*, 174:110891, 2021.

[262] Mishall Al-Zubaidie, Zhongwei Zhang, and Ji Zhang. Efficient and secure ecdsa algorithm and its applications: a survey. *arXiv preprint arXiv:1902.10313*, 2019.

[263] Gabriel Nyame, Zhiguang Qin, Kwame Opuni-Boachie Obour Agyekum, and Emmanuel Boateng Sifah. An ecdsa approach to access control in knowledge management systems using blockchain. *Information*, 11(2):111, 2020.

[264] Davi Pedro Bauer. Solidity. In *Getting Started with Ethereum*, pages 13–16. Springer, 2022.

[265] Wei-Meng Lee. Testing smart contracts using ganache. In *Beginning Ethereum Smart Contracts Programming*, pages 147–167. Springer, 2019.

[266] Ethereum Foundation. Ropsten testnet pow chain. `https://github.com/ethereum/ropsten`.

[267] Truffle Blockchain Group. Sweet tools for smart contracts — truffle suite. `https://www.trufflesuite.com/`.

[268] Kovan. Kovan - stable ethereum public testnet. `https://github.com/kovan-testnet/proposal`.

[269] Rinkeby. Rinkeby: Ethereum testnet. `https://www.rinkeby.io/`.

[270] Sara Tucci-Piergiovanni. Keynote: Blockchain consensus protocols, from bitcoin to ethereum 2.0. In *2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, pages 1–1. IEEE, 2022.

[271] Karl Wüst and Arthur Gervais. Do you need a blockchain? In *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*, pages 45–54. IEEE, 2018.

[272] Andrew Miller. Permissioned and permissionless blockchains. *Blockchain for distributed systems security*, pages 193–204, 2019.

[273] Usman W Chohan. The cryptocurrency tumblers: Risks, legality and oversight. 2017.

[274] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, and Huaimin Wang. An overview of blockchain technology: Architecture, consensus, and future trends. In *2017 IEEE international congress on big data (BigData congress)*, pages 557–564. IEEE, 2017.

[275] Ahmed Kosba, Andrew Miller, Elaine Shi, Zikai Wen, and Charalampos Papamanthou. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In *2016 IEEE symposium on security and privacy (SP)*, pages 839–858. IEEE, 2016.

[276] Jason E Holt and Kent E Seamons. Logcrypt: forward security and public verification for secure audit logs. *Cryptology ePrint Archive*, 2005.

[277] Enrique Soriano-Salvador and Gorka Guardiola-Múzquiz. Sealfs: Storage-based tamper-evident logging. *Computers & Security*, 108:102325, 2021.

[278] Emily R. Siegel, Cynthia McFadden, Kevin Monahan, Andrew W. Lehren, and Pauliina Siniauer. The da vinci surgical robot: A medical breakthrough with risks for patients. `https://www.nbcnews.com/health/health-news/da-vinci-surgical-robot-medical-breakthrough-risks-patients-n949341`, 12 2018.

[279] Surgical Watch. da vinci robot lawsuit. http://surgicalwatch.com/davinci-robot/lawsuit/, 2015.

[280] Md Mahmud Hossain, Maziar Fotouhi, and Ragib Hasan. Towards an analysis of security issues, challenges, and open problems in the internet of things. In *2015 IEEE World Congress on Services*, pages 21–28. IEEE, 2015.

[281] Omar Alrawi, Chaz Lever, Manos Antonakakis, and Fabian Monrose. Sok: Security evaluation of home-based iot deployments. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 1362–1380. IEEE, 2019.

[282] Qassim Nasir, Ilham A Qasse, Manar Abu Talib, and Ali Bou Nassif. Performance analysis of hyperledger fabric platforms. *Security and Communication Networks*, 2018, 2018.

[283] Aritra Banerjee, Michael Clear, and Hitesh Tewari. Demystifying the role of zk-snarks in zcash. In *2020 IEEE conference on application, information and network security (AINS)*, pages 12–19. IEEE, 2020.

[284] Ravi S Sandhu. Lattice-based access control models. *Computer*, (11):9–19, 1993.

[285] OASIS. extensible access control markup language (xacml) version 3.0.

[286] Alan S Morris and Reza Langari. *Measurement and instrumentation: theory and application*. Academic Press, 2012.

[287] Michael Chu, Thao Nguyen, Vaibhav Pandey, Yongxiao Zhou, Hoang N Pham, Ronen Bar-Yoseph, Shlomit Radom-Aizik,

Ramesh Jain, Dan M Cooper, and Michelle Khine. Respiration rate and volume measurements using wearable strain sensors. *NPJ digital medicine*, 2(1):1–9, 2019.

[288] Chien-Ying Chen, Monowar Hasan, and Sibin Mohan. Securing real-time internet-of-things. *Sensors*, 18(12):4356, 2018.

[289] Bako Ali and Ali Ismail Awad. Cyber and physical security vulnerability assessment for iot-based smart homes. *sensors*, 18(3):817, 2018.

[290] Shivani Chowdhary, Subhranil Som, Vipul Tuli, and Sunil Kumar Khatri. Security solutions for physical layer of iot. In *2017 International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions)(ICTUS)*, pages 579–583. IEEE, 2017.

[291] Timo Salmi, Jari M Ahola, Tapio Heikkilä, Pekka Kilpeläinen, and Timo Malm. Human-robot collaboration and sensor-based robots in industrial applications and construction. In *Robotic Building*, pages 25–52. Springer, 2018.

[292] Kristine Frank and Ida C Willemoes-Wissing. Combining logical and physical access control for smart environments. Master's thesis, Technical University of Denmark, DTU, DK-2800 Kgs. Lyngby, Denmark, 2004.

[293] Vincent C Hu, D Richard Kuhn, David F Ferraiolo, and Jeffrey Voas. Attribute-based access control. *Computer*, 48(2):85–88, 2015.

[294] Pengfei Yang, Quan Wang, Xin Mi, and Jingwei Li. An improved blp model with more flexibility. In *2016 13th Interna-*

tional Conference on Embedded Software and Systems (ICESS), pages 192–197. IEEE, 2016.

[295] WC Shi. Research on and enforcement of methods ofmethods of secure operating systems development [ph. d. thesis]. *Institute of Software, The Chinese Academy of Sciences, Beijing*, 2001.

[296] Jian-Bo He, Si-Han Qing, and Chao Wang. Analysis of two improved blp models. *Ruan Jian Xue Bao(Journal of Software)*, 18(6):1501–1509, 2007.

[297] Jun Zhang, Li-Jun Yun, and Zheng Zhou. Research of blp and biba dynamic union model based on check domain. In *2008 International Conference on Machine Learning and Cybernetics*, volume 7, pages 3679–3683. IEEE, 2008.

[298] Gang Liu, Can Wang, Runnan Zhang, Quan Wang, Huimin Song, and Shaomin Ji. Btg-biba: A flexibility-enhanced biba model using btg strategies for operating system. *International Journal of Computer and Information Engineering*, 11(6):765–771, 2017.

[299] Bayu Anggorojati, Parikshit Narendra Mahalle, Neeli Rashmi Prasad, and Ramjee Prasad. Capability-based access control delegation model on the federated iot network. In *Wireless Personal Multimedia Communications (WPMC), 2012 15th International Symposium on*, pages 604–608. IEEE, 2012.

[300] Sergio Gusmeroli, Salvatore Piccione, and Domenico Rotondi. A capability-based security approach to manage access control in the internet of things. *Mathematical and Computer Modelling*, 58(5-6):1189–1205, 2013.

[301] Rui Zhang, Yanchao Zhang, and Kui Ren. Dp^{2}ac: Distributed privacy-preserving access control in sensor networks. In *IEEE INFOCOM 2009*, pages 1251–1259. IEEE, 2009.

A — Confusion Matrices for Traffic Analysis Attack

A.1 Baseline

This matrix depicts the baseline classification results (TP/FP rates) for the traffic analysis attack.

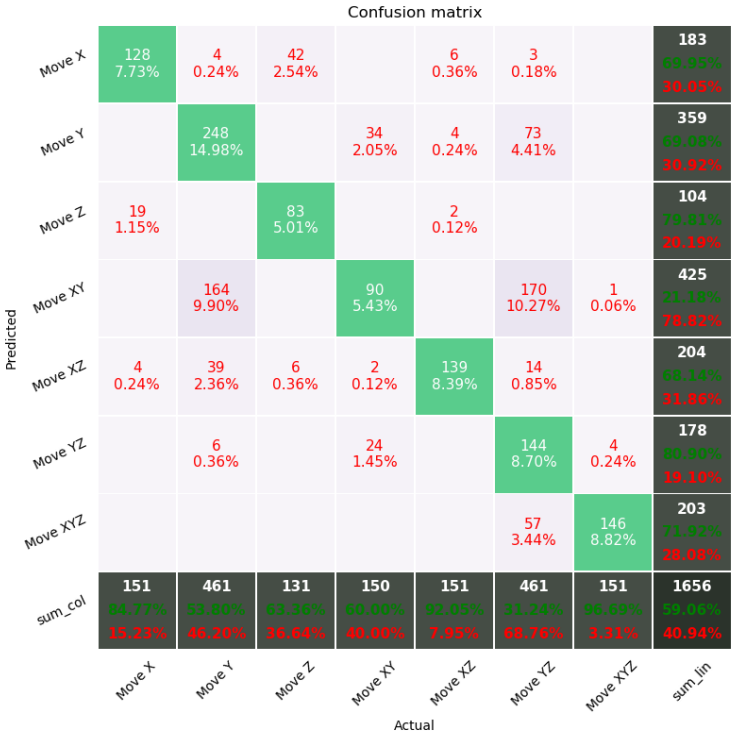


Figure A.1: Baseline Confusion Matrix

A.2 Distance

These matrices depict the classification results for the distance parameter in the traffic analysis attack, where each iteration links with an increase in robot movement distance.

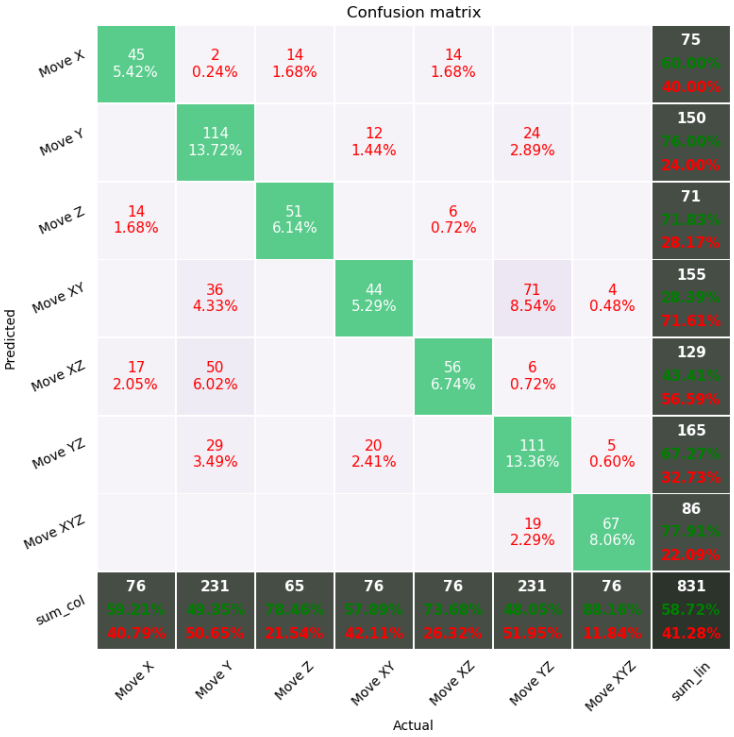


Figure A.2: Distance (2mm) Confusion Matrix

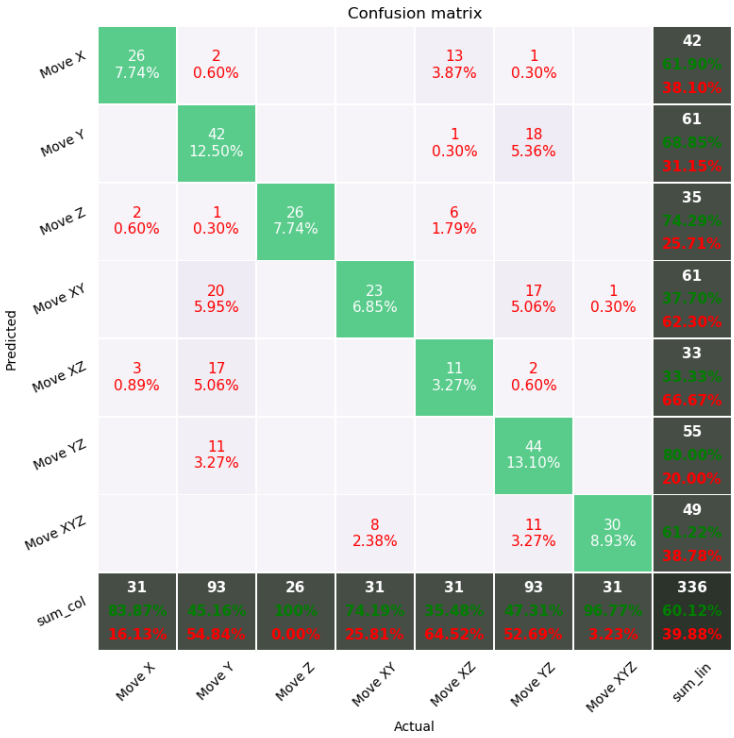


Figure A.3: Distance (5mm) Confusion Matrix

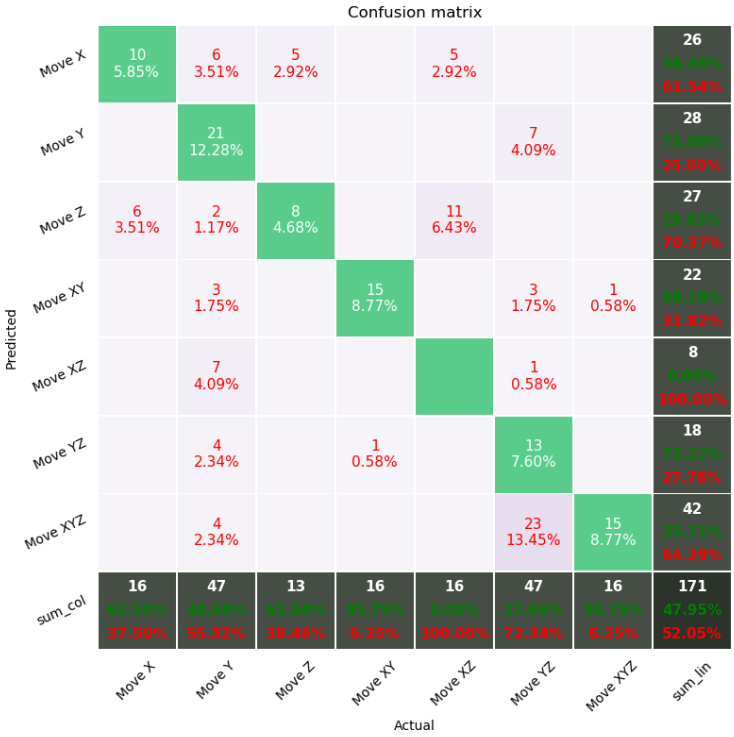


Figure A.4: Distance (10mm) Confusion Matrix

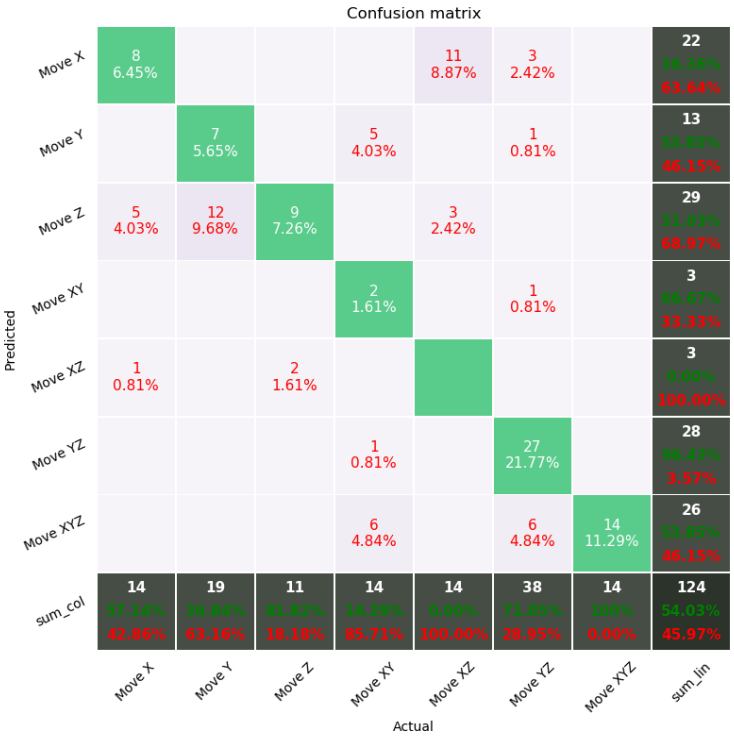


Figure A.5: Distance (25mm) Confusion Matrix

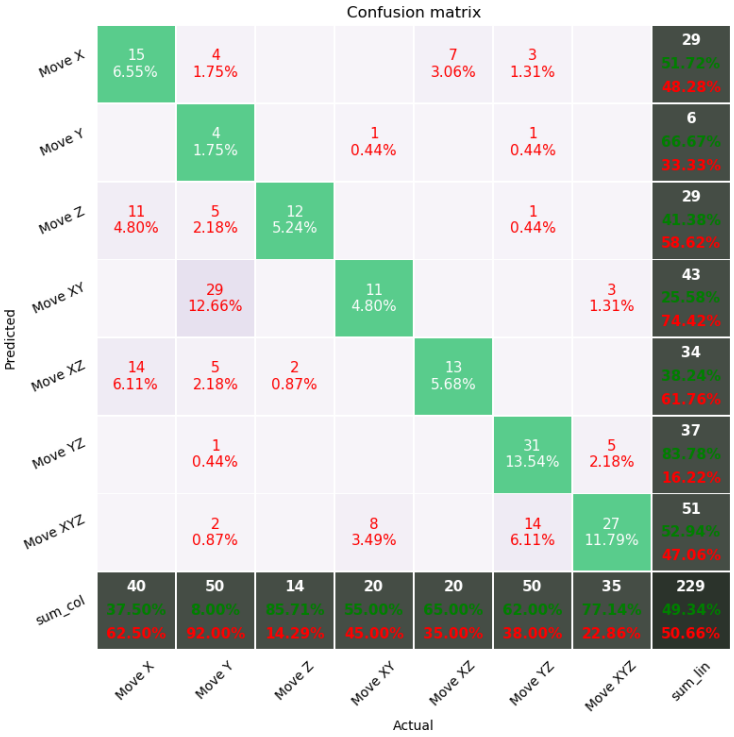


Figure A.6: Distance (50mm) Confusion Matrix

A.3 Speed

These matrices depict the classification results for the speed parameter in the traffic analysis attack, where each iteration links with an increase in robot movement speed.

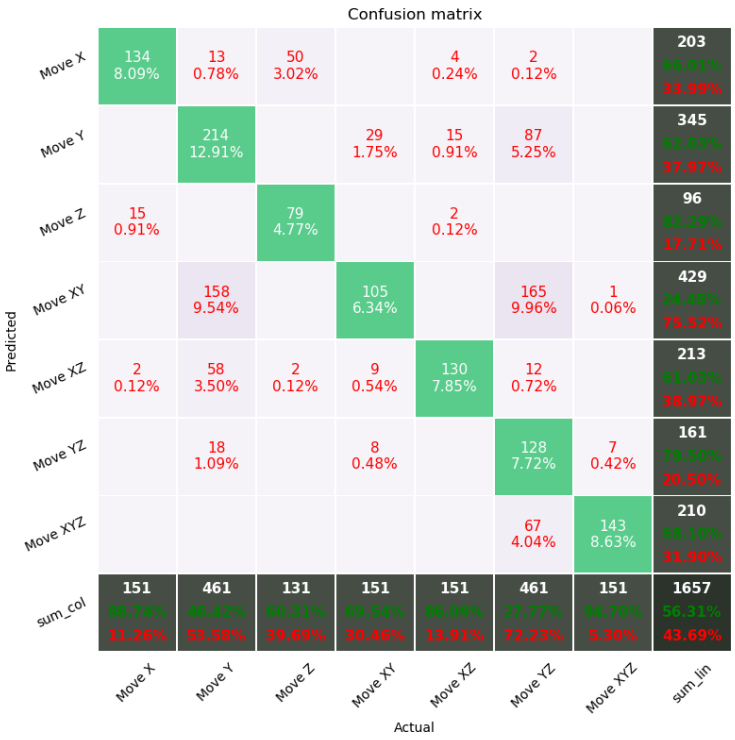


Figure A.7: Speed (25mm/s) Confusion Matrix

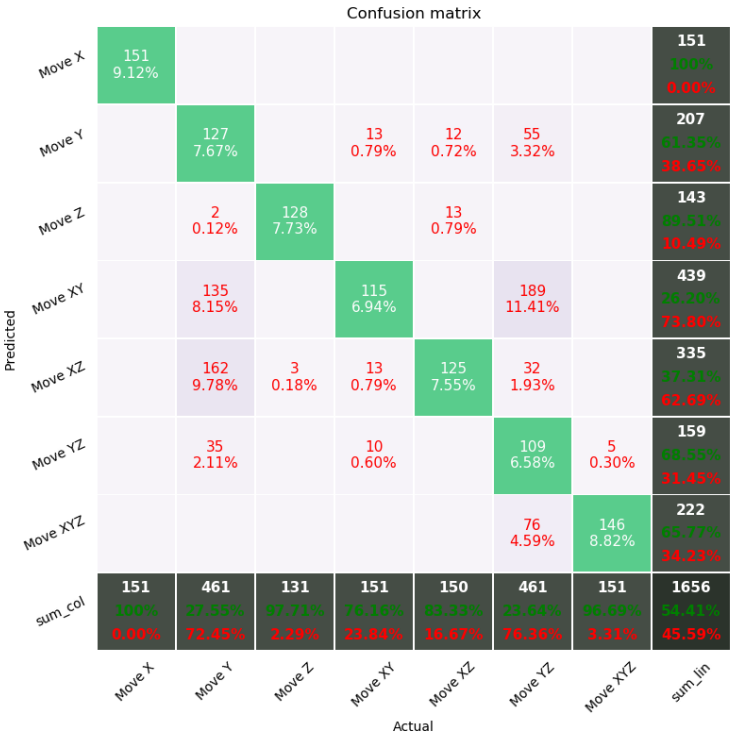


Figure A.8: Speed (50mm/s) Confusion Matrix

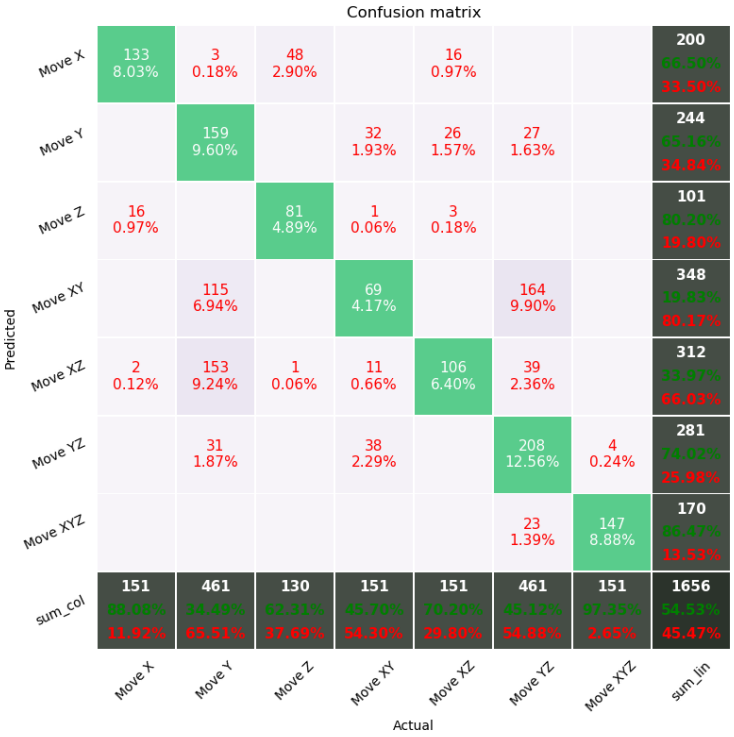


Figure A.9: Speed (75mm/s) Confusion Matrix

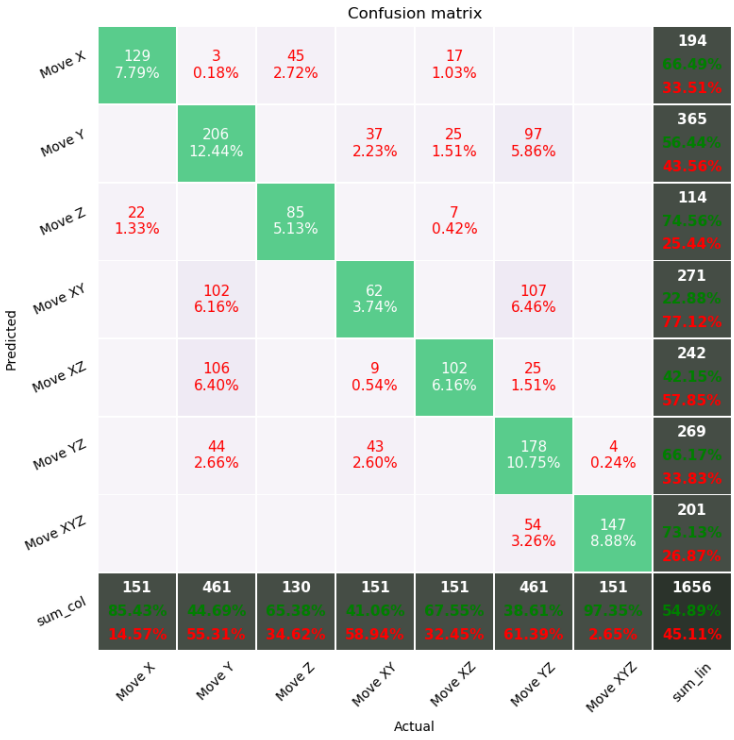


Figure A.10: Speed (100mm/s) Confusion Matrix

A.4 Open-Set Evaluation

These matrices depict the classification results for an open-set evaluation of the traffic analysis attack, where movements are subsequently removed such that they become unknown to the classifier.

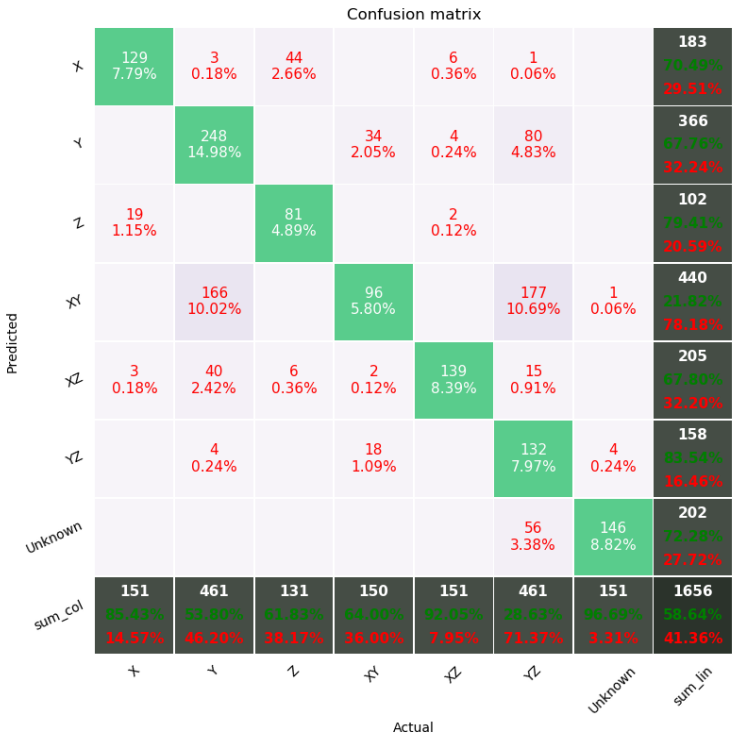


Figure A.11: Open-Set (1 Unknown) Confusion Matrix

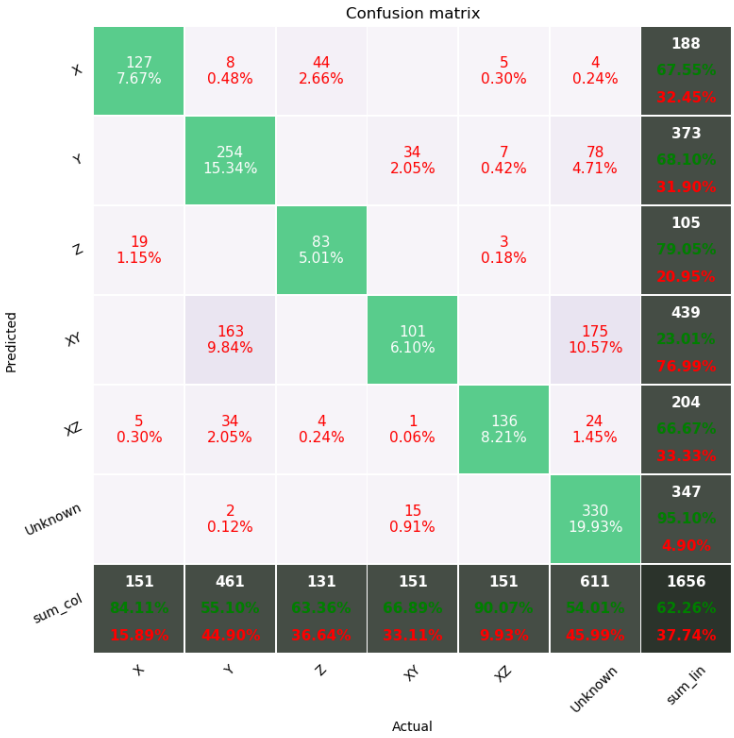


Figure A.12: Open-Set (2 Unknowns) Confusion Matrix

Figure A.13: Open-Set (3 Unknowns) Confusion Matrix

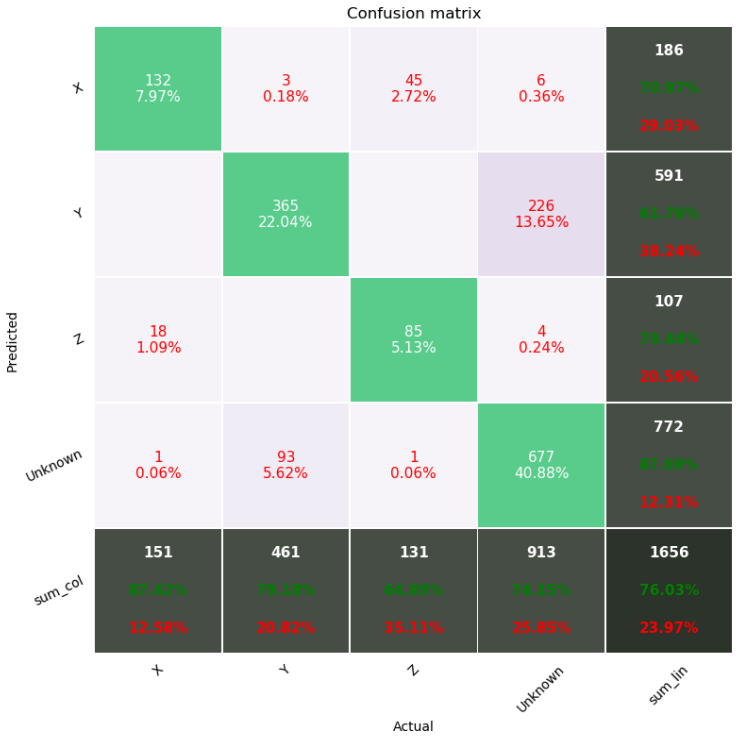


Figure A.14: Open-Set (4 Unknowns) Confusion Matrix

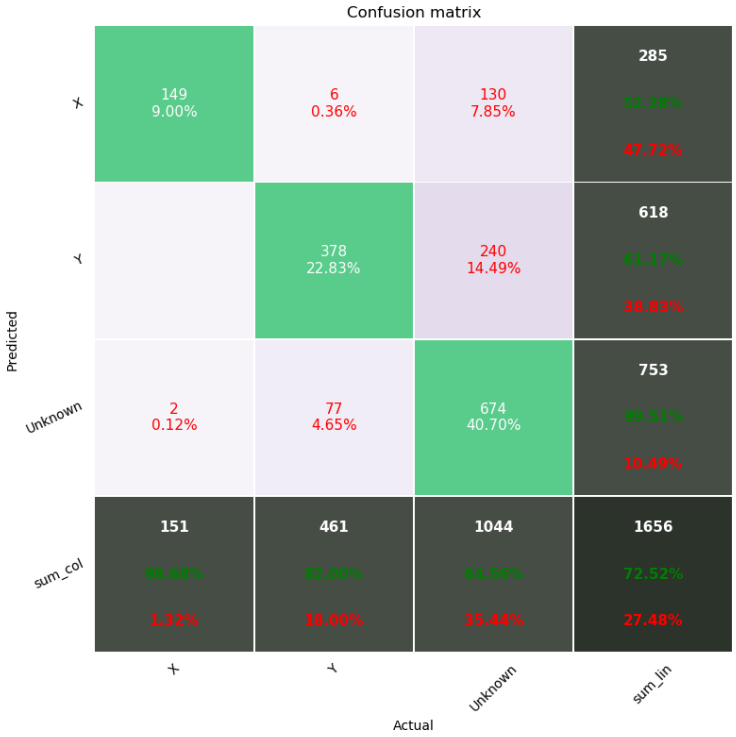


Figure A.15: Open-Set (5 Unknowns) Confusion Matrix

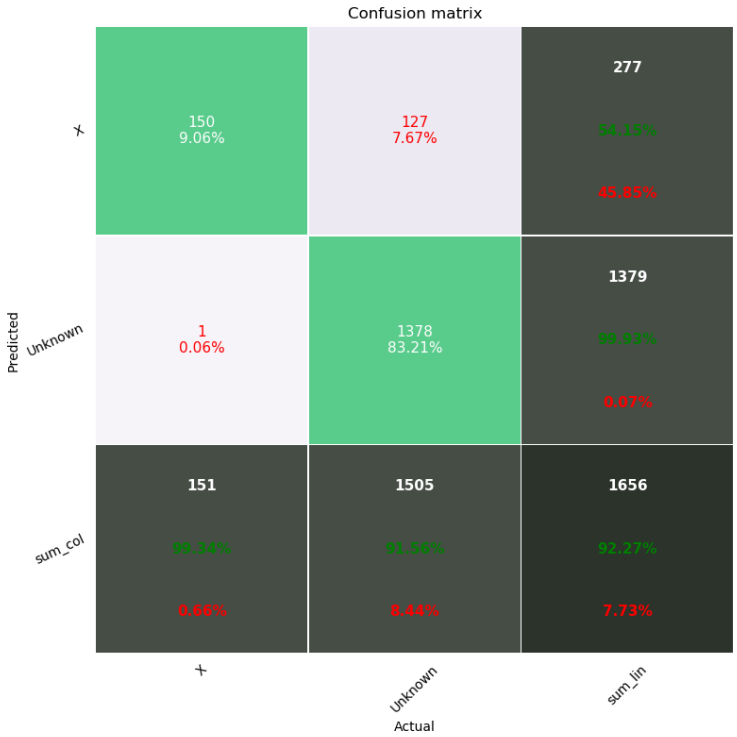


Figure A.16: Open-Set (6 Unknowns) Confusion Matrix

A.5 Tor Defence

This matrix depicts the classification results for the baseline robot movements with the Tor defence employed, used for comparing the defence against the attack baseline.

Figure A.17: Tor Defence Confusion Matrix

B — Confusion Matrices for Radio Frequency Side Channel

B.1 Baseline

This matrix depicts the classification result for baseline robot movement fingerprinting using the radio frequency (RF) side channel.

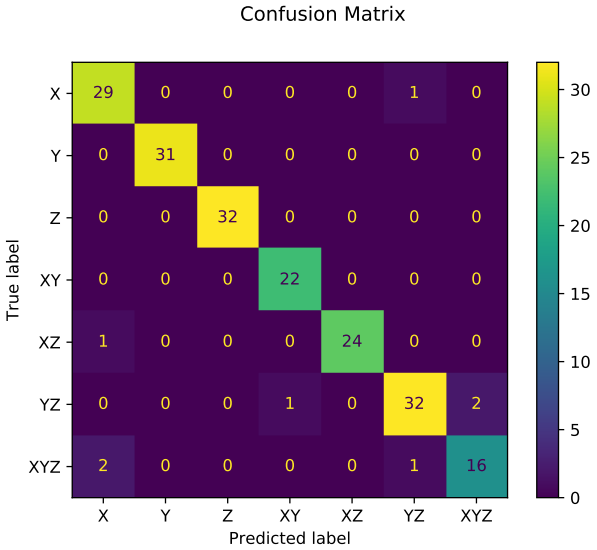


Figure B.1: Baseline Confusion Matrix

B.2 Distance

These matrices depict the classification results for the distance parameter in the RF side channel attack, where each iteration links with an increase in robot movement distance.

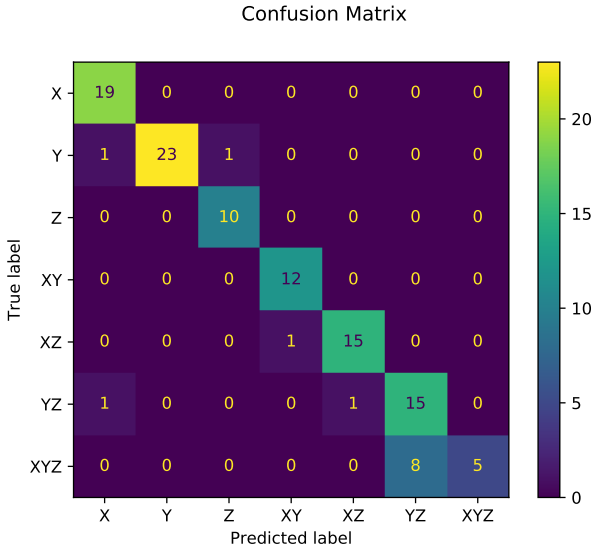


Figure B.2: Distance (2mm) Confusion Matrix

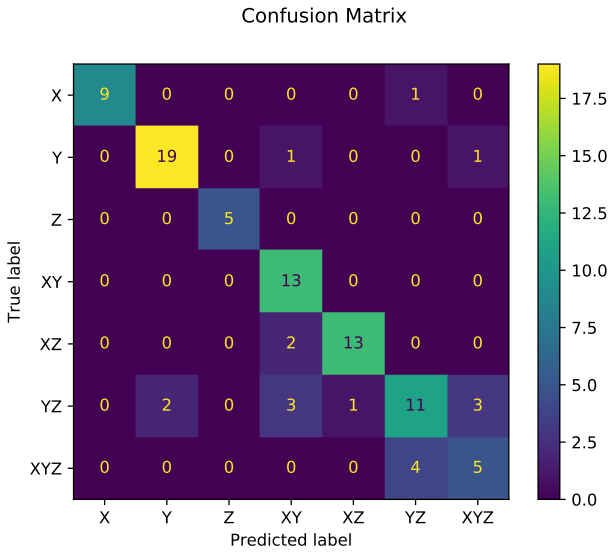


Figure B.3: Distance (5mm) Confusion Matrix

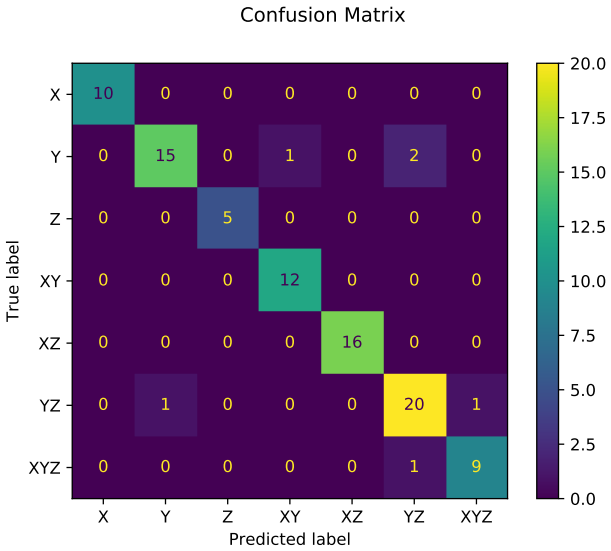


Figure B.4: Distance (10mm) Confusion Matrix

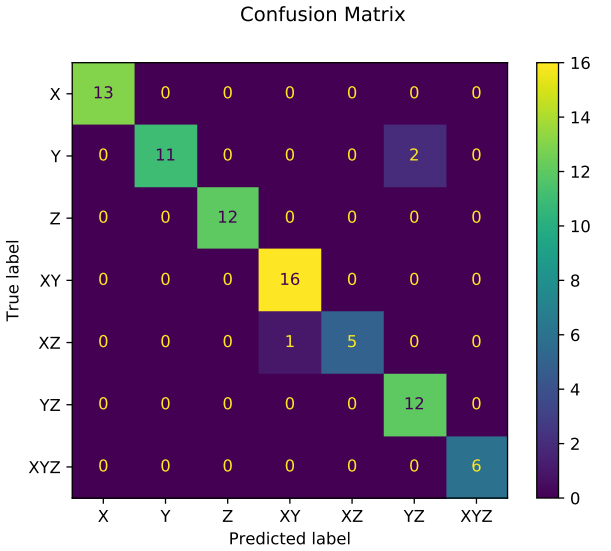


Figure B.5: Distance (25mm) Confusion Matrix

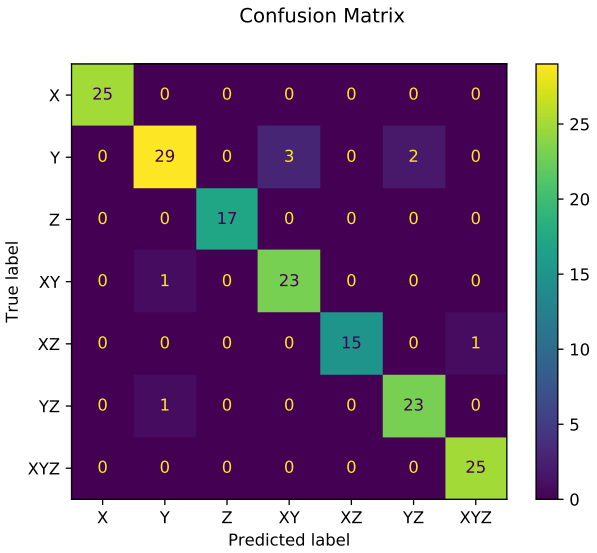


Figure B.6: Distance (50mm) Confusion Matrix

B.3 Speed

These matrices depict the classification results for the speed parameter in the RF side channel attack, where each iteration links with an increase in robot movement distance.

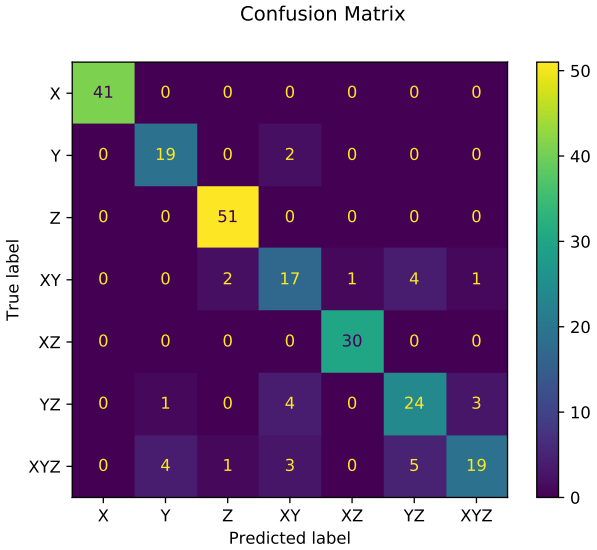


Figure B.7: Speed (25mm/s) Confusion Matrix

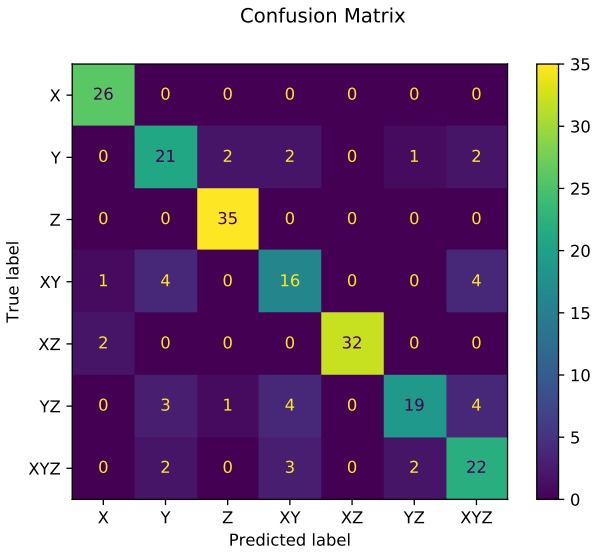


Figure B.8: Speed (50mm/s) Confusion Matrix

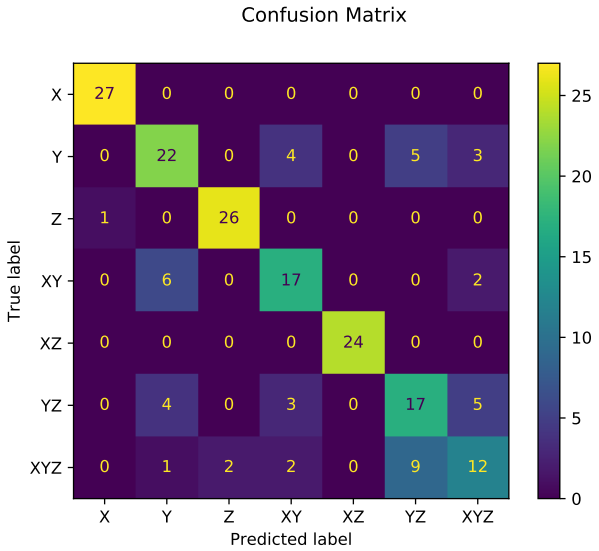


Figure B.9: Speed (75mm/s) Confusion Matrix

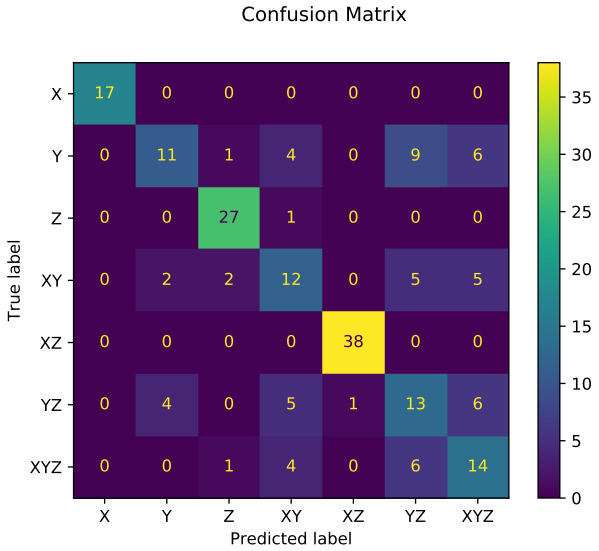


Figure B.10: Speed (100mm/s) Confusion Matrix

B.4 Antenna Distance

These matrices depict the classification results for the antenna distance parameter in the RF side channel attack, where each iteration links with an increase in the distance the RF receiver antenna is away from the robot under attack.

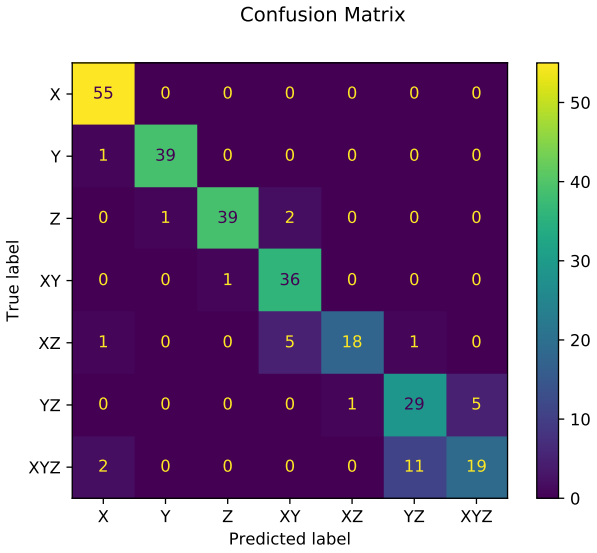


Figure B.11: Antenna Distance (25cm) Confusion Matrix

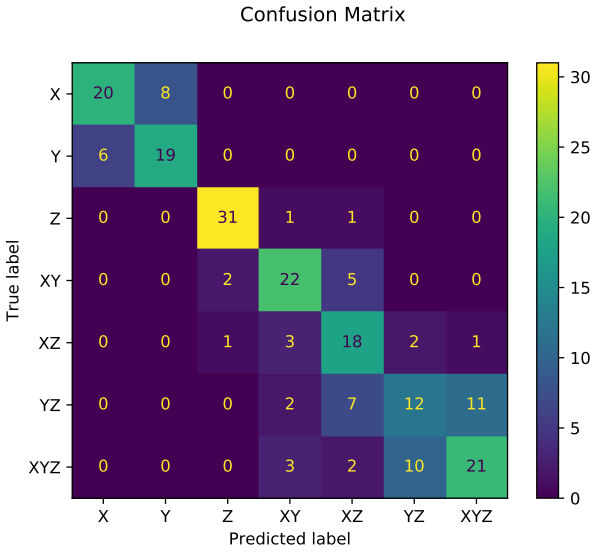


Figure B.12: Antenna Distance (50cm) Confusion Matrix

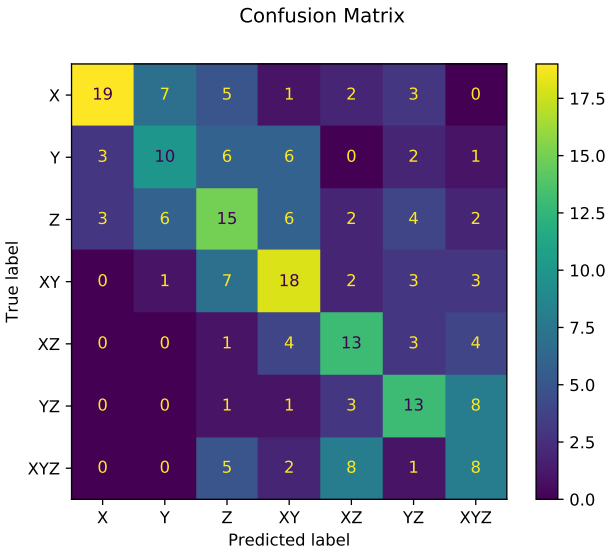


Figure B.13: Antenna Distance (100cm) Confusion Matrix

B.5 Workflow Reconstruction

These matrices depict the classification results for workflow reconstruction using the RF side channel attack. Three different sets of each of the workflows were examined to account for variations in workflows as may be seen in realistic settings.

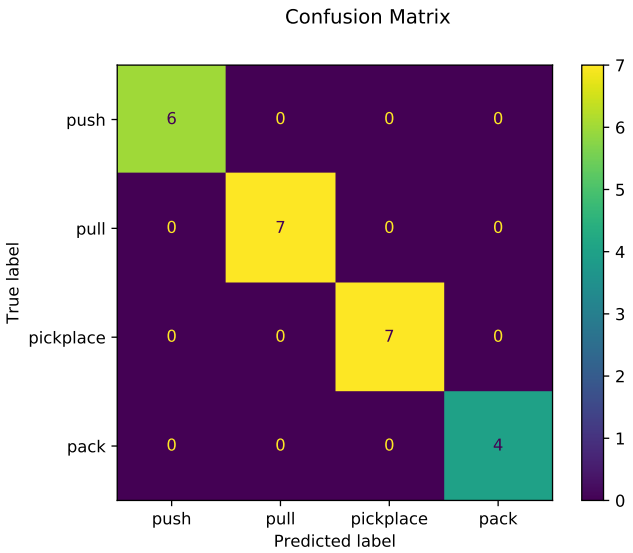


Figure B.14: Workflow Reconstruction (1) Confusion Matrix

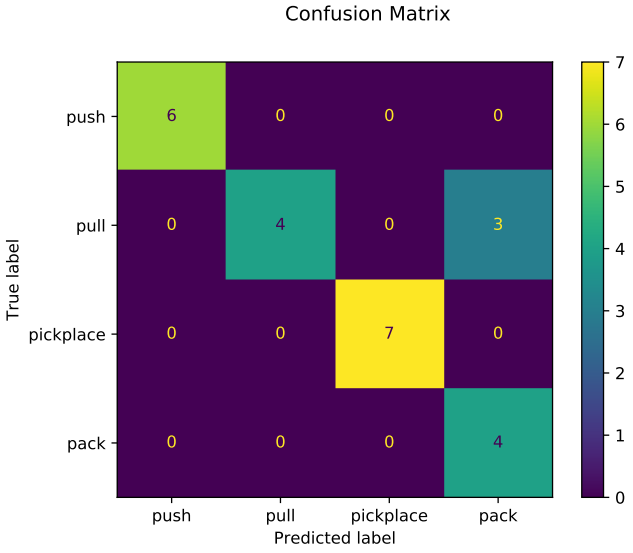


Figure B.15: Workflow Reconstruction (2) Confusion Matrix

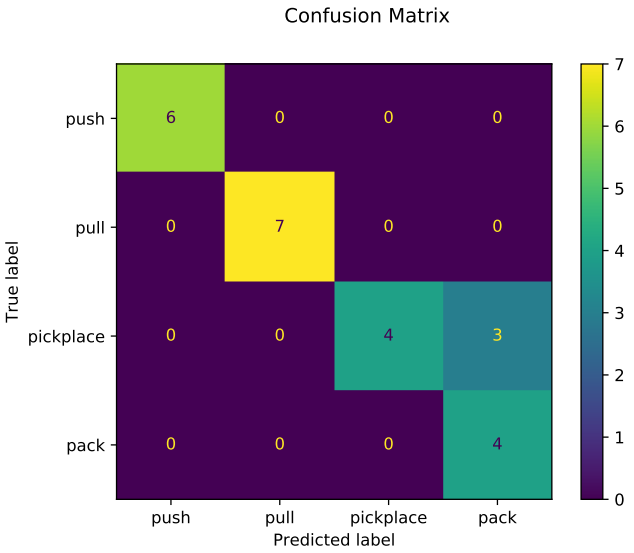


Figure B.16: Workflow Reconstruction (3) Confusion Matrix

C — Confusion Matrices for Acoustic Side Channel

C.1 Baseline

This matrix depicts the classification results (TP/FP rates) for the baseline robot movements in the acoustic side channel.

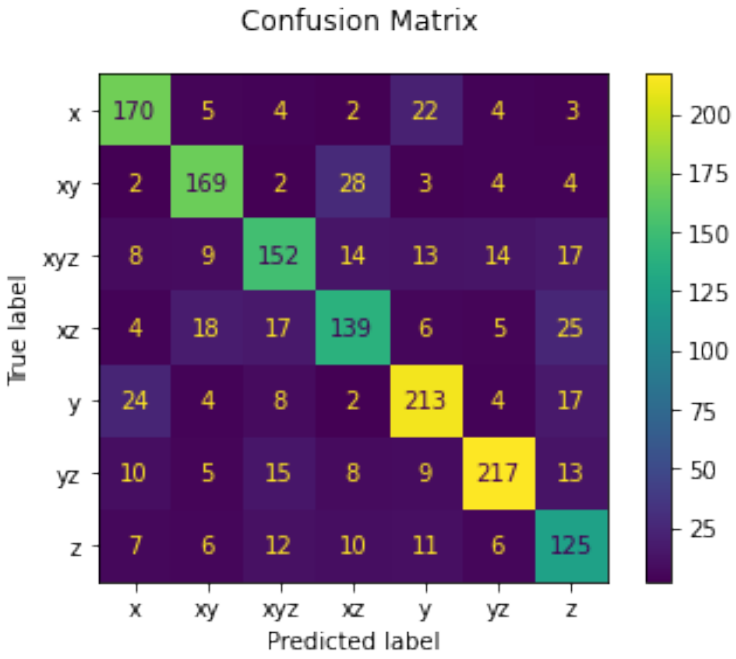


Figure C.1: Baseline Confusion Matrix

C.2 Distance

These matrices depict the classification results for the distance parameter in the acoustic side channel attack, where each iteration links with an increase in robot movement distance.

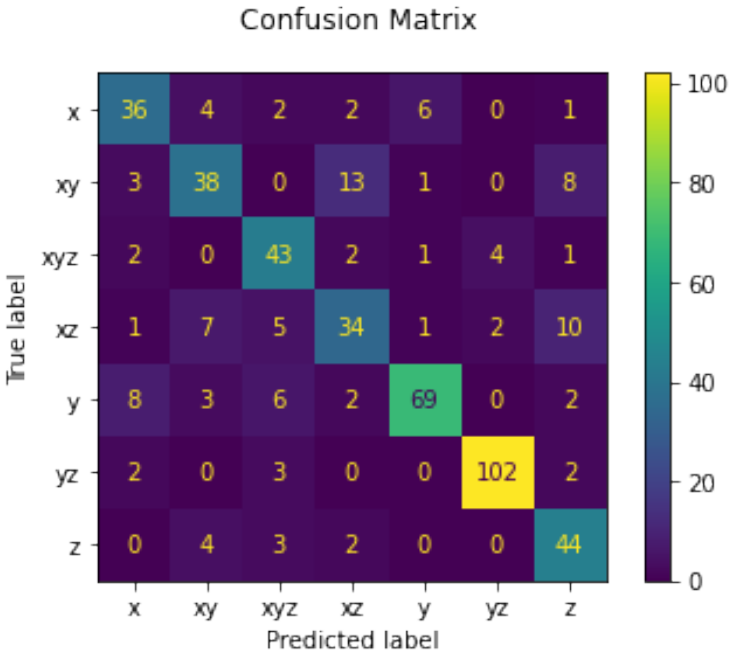


Figure C.2: Distance (2mm) Confusion Matrix

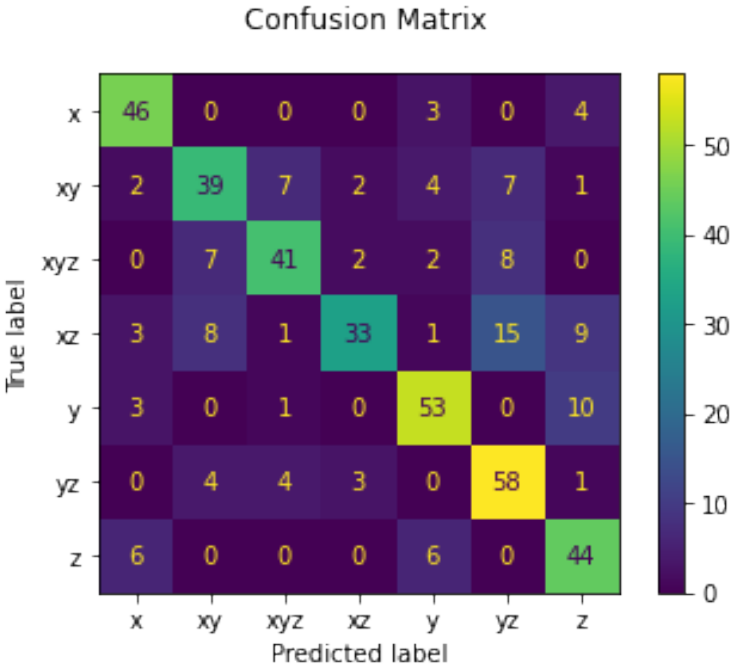


Figure C.3: Distance (5mm) Confusion Matrix

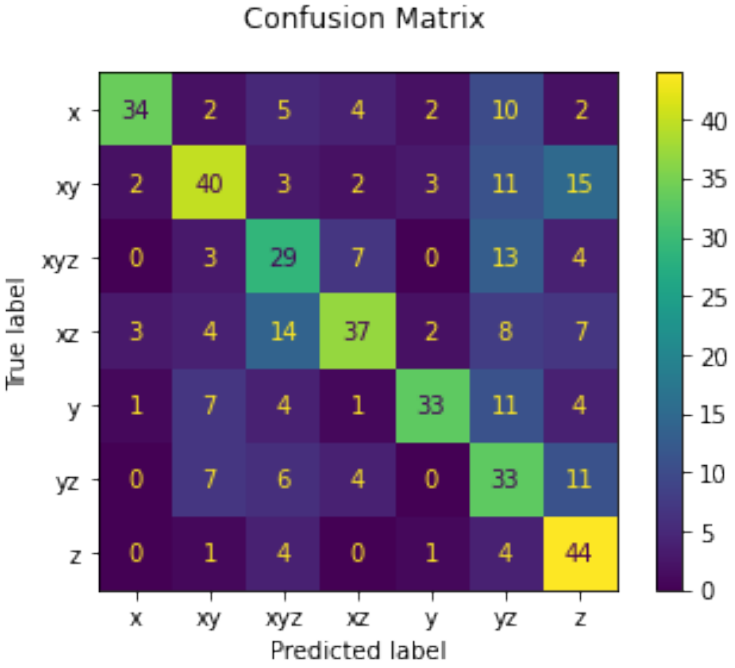


Figure C.4: Distance (10mm) Confusion Matrix

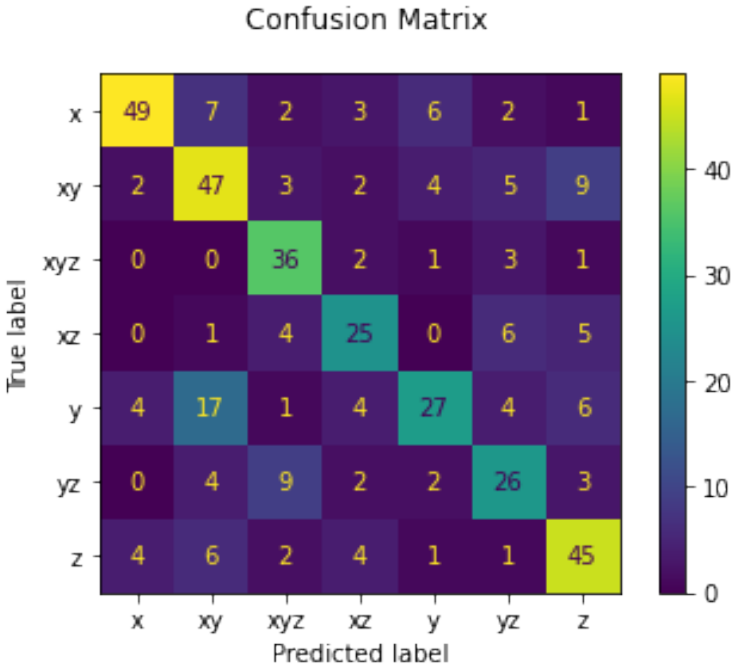


Figure C.5: Distance (25mm) Confusion Matrix

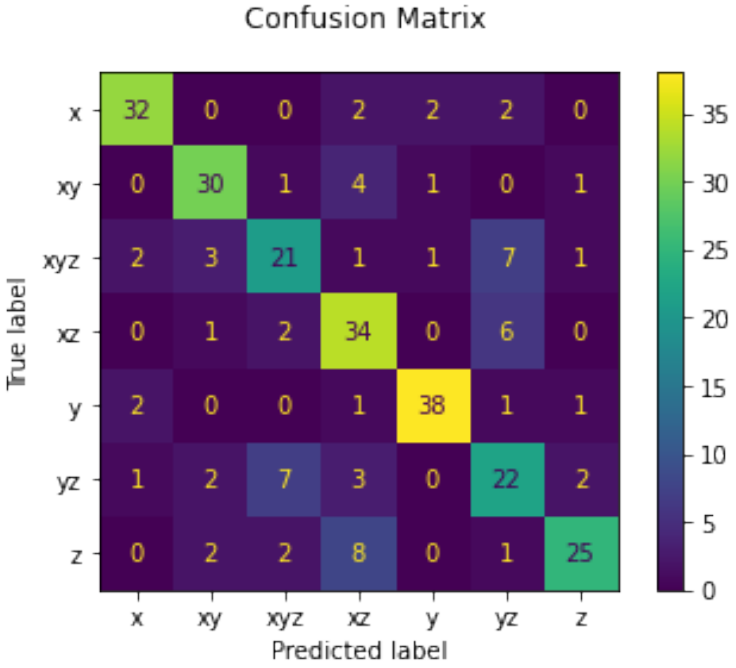


Figure C.6: Distance (50mm) Confusion Matrix

C.3 Speed

These matrices depict the classification results for the speed parameter in the acoustic side channel attack, where each iteration links with an increase in robot movement distance.

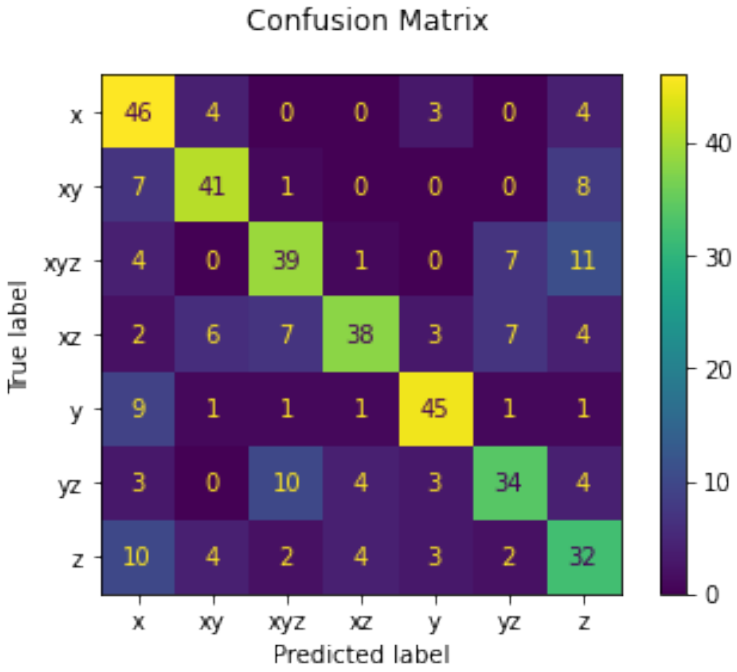


Figure C.7: Speed (25mm/s) Confusion Matrix

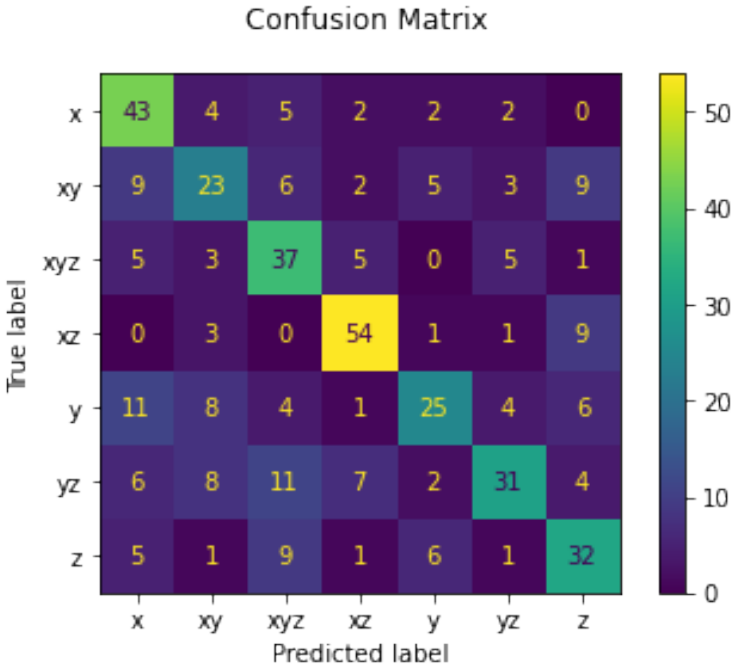


Figure C.8: Speed (50mm/s) Confusion Matrix

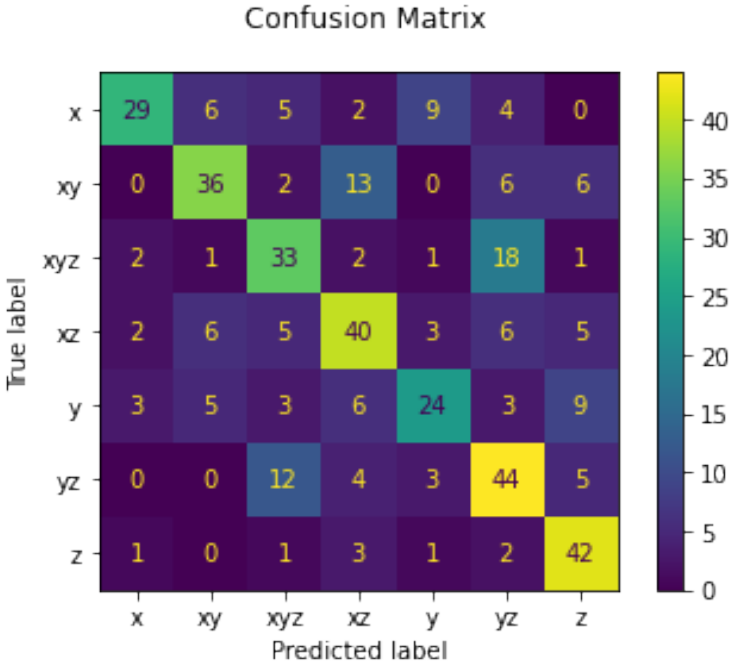


Figure C.9: Speed (75mm/s) Confusion Matrix

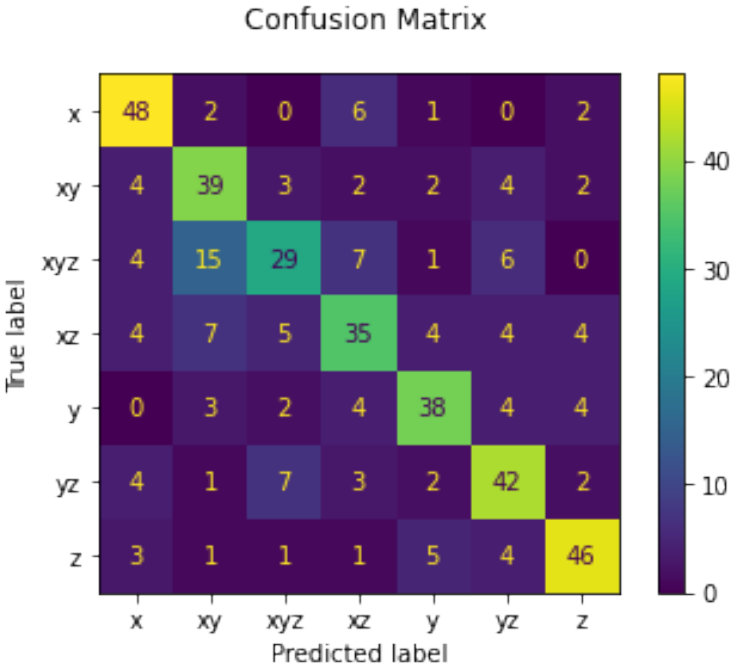


Figure C.10: Speed (100mm/s) Confusion Matrix

C.4 Microphone Distance

These matrices depict the classification results for the microphone distance parameter in the acoustic side channel attack, with each iteration relating to an increase in the distance the recording device is away from the robot under attack.

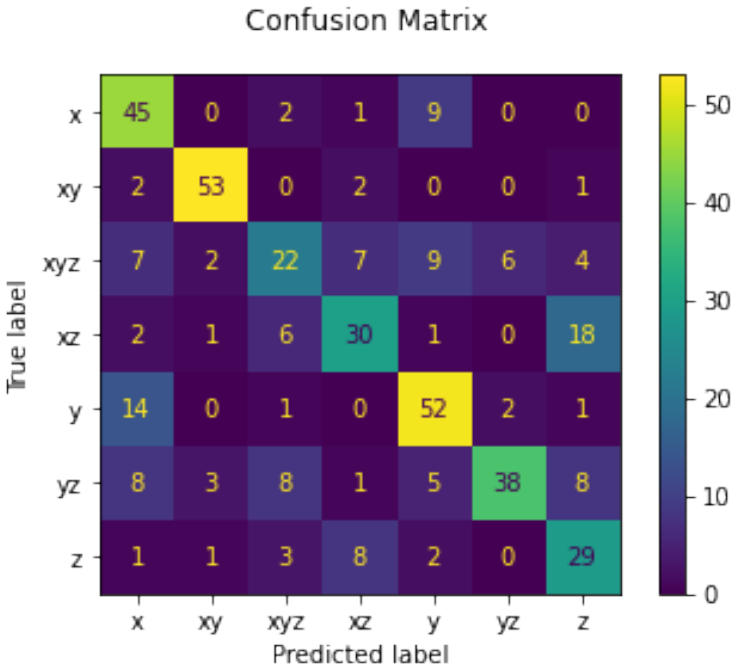


Figure C.11: Microphone Distance (50cm) Confusion Matrix

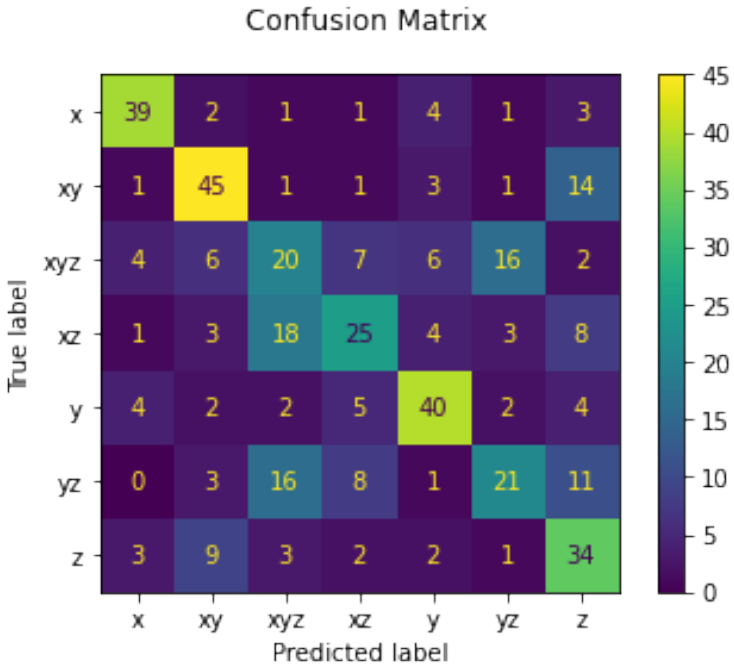


Figure C.12: Microphone Distance (100cm) Confusion Matrix

C.5 Opus (VoIP) Codec and Packet Loss

These matrices depict the classification results for the baseline robot movements with VoIP and Opus codec employed. These are used to evaluate the impact of VoIP on the accuracy of the acoustic side channel in VoIP settings.

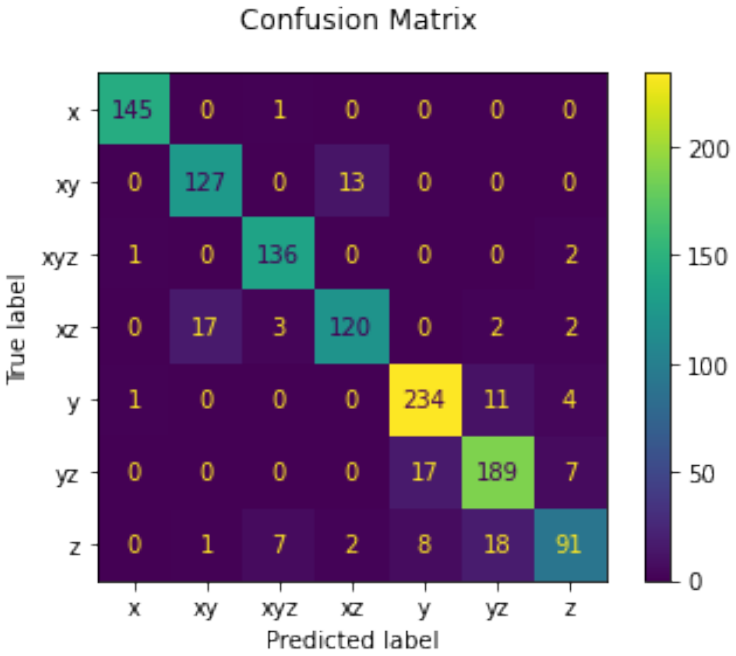


Figure C.13: Opus Packet Loss 1% Confusion Matrix

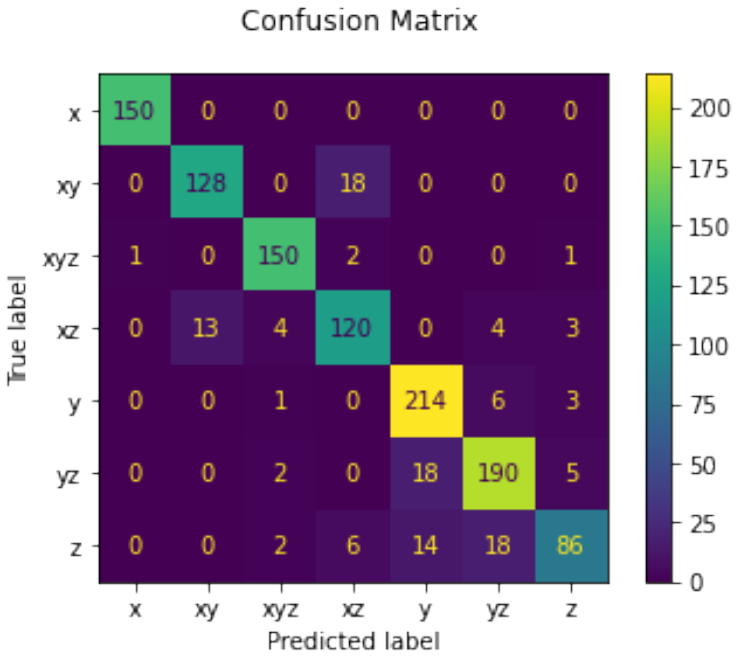


Figure C.14: Opus Packet Loss 5% Confusion Matrix

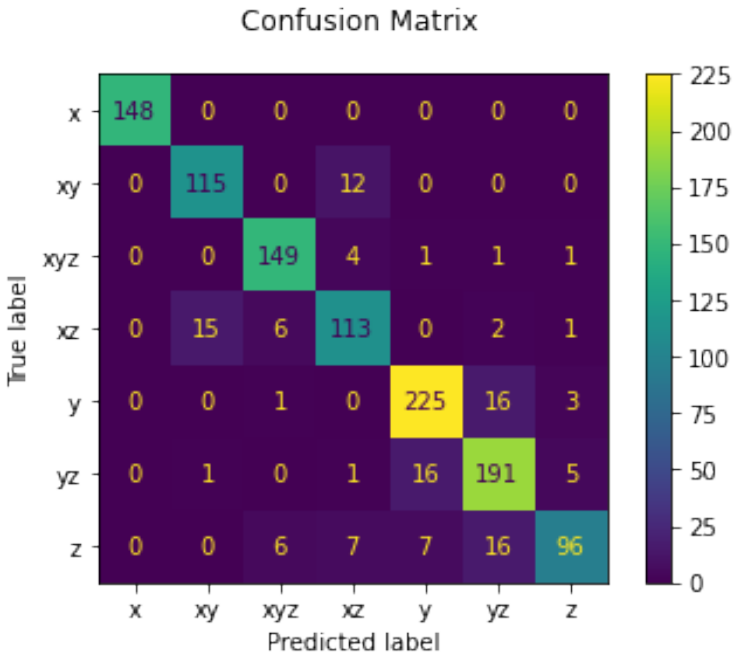


Figure C.15: Opus Packet Loss 10% Confusion Matrix

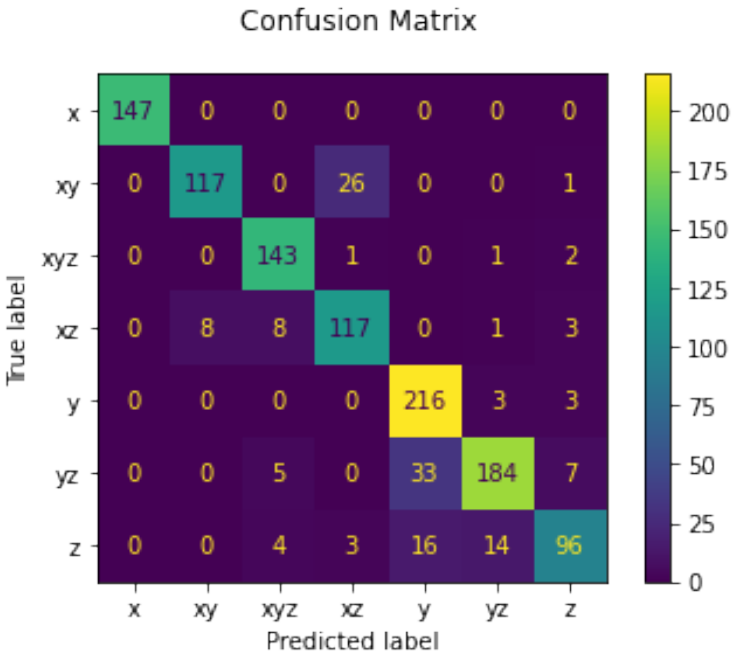


Figure C.16: Opus Packet Loss 25% Confusion Matrix

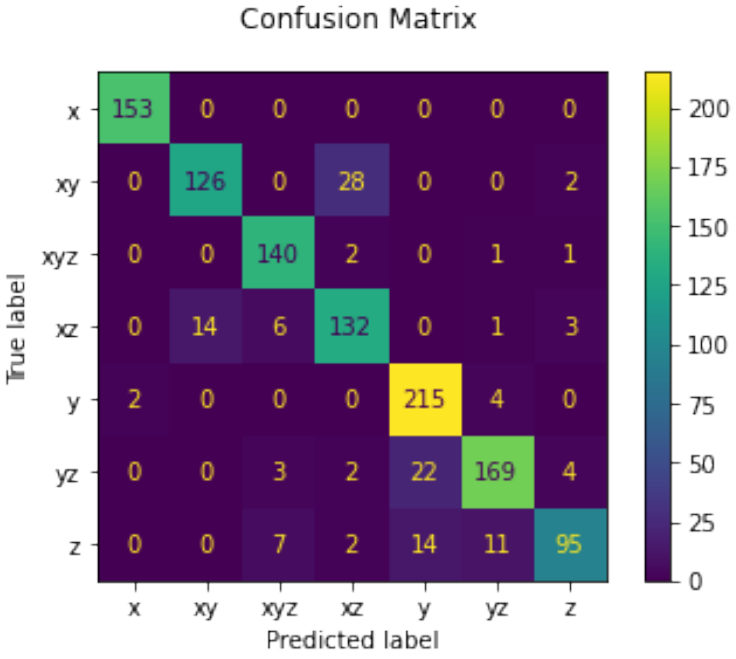


Figure C.17: Opus Packet Loss 50% Confusion Matrix

C.6 Workflow Reconstruction

This matrix depicts the classification result for workflow reconstruction using the acoustic side channel attack.

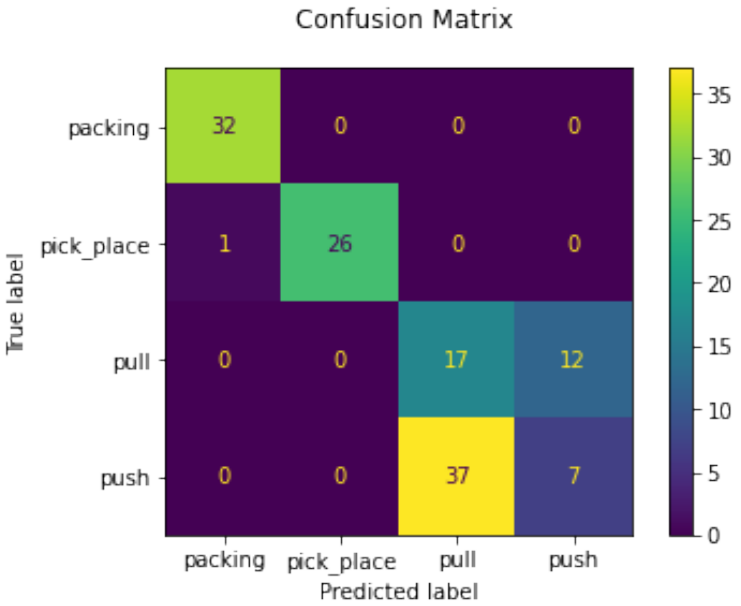


Figure C.18: Workflow Reconstruction Confusion Matrix

C.7 Noise Reduction

This matrix depicts the classification result for employing amplitude filtering as a noise reduction technique, used to compare against the baseline robot movements.

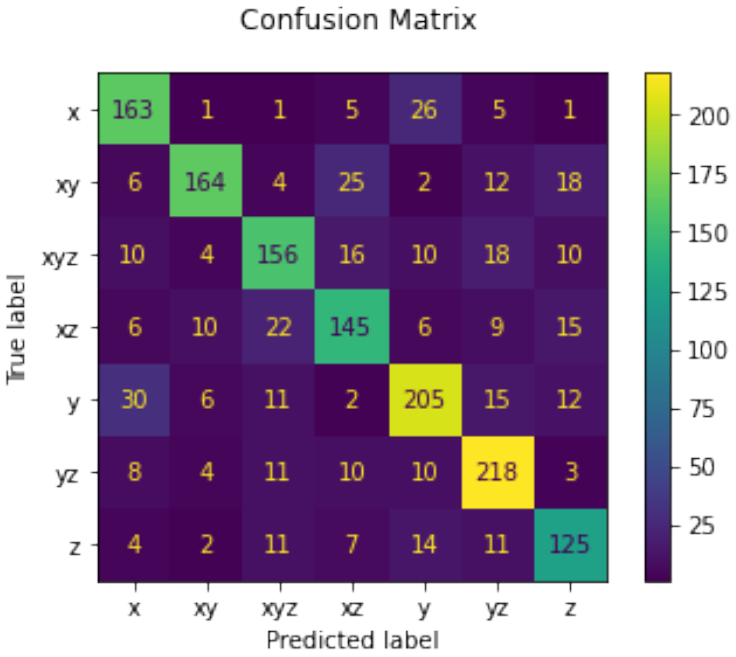


Figure C.19: Baseline (Amplitude Filtering) Confusion Matrix

D — Calibration

This algorithm describes that used for carrying out traceability verification of a device's calibration (e.g. robot sensor) within the smart contract on the Ethereum blockchain.

D.1 Traceability Verification Algorithms

Algorithm 1 Trace Verification

```
1:  procedure TraceCal_READ(device_id)
2:      device_report = reports[device_id]
3:      parent_cert = certificates[device_report.parent_id]
4:      technician_cert = certificates[device_report.technician_id]
5:      if !(key_verify(device_report, parent_cert)) then
6:          return null
7:      if !(key_verify(device_report, technician_cert) then
8:          return null
9:      org_cert = certificates[technician_cert.org_id]
10:     if verify_signature(technician_cert, org_cert) == false then
11:         return null
12:     if check_chain_of_trust(org_cert, ROOT_CERT) == false then
13:         return null
14:     root_report_id = device_id
15:     parent = reports[root_report_id].parent_device
16:     while bytes(parent).length > 0 do
17:         if key_verify(parent,
18:             certificates[parent].parent_device) then
19:             if key_verify(parent, technician_cert) then
20:                 root_report_id = parent
21:                 parent = reports[root_report_id].parent_device
22:             else
23:                 return null
24:         else
25:             return null
26:     return reports[root_report_id]
```