Context-Awareness for Adversarial and Defensive Machine Learning Methods in Cybersecurity

by

Kyle Quintal

A thesis presented to the University of Ottawa in
fulfillment of the requirement for the degree
of Master of Computer Science

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

I hereby declare that I am the sole author of this thesis. This is the only true copy of my thesis, and it includes all final revisions as requested by my examiners.

I understand that my thesis may be made available to the public.

Abstract

Machine Learning has shown great promise when combined with large volumes of historical data and produces great results when combined with contextual properties. In the world of the Internet of Things, the extraction of information regarding context, or contextual information, is increasingly prominent with scientific advances. Combining such advancements with artificial intelligence is one of the themes in this thesis. Particularly, there are two major areas of interest: context-aware attacker modelling and context-aware defensive methods. Both areas use authentication methods to either infiltrate or protect digital systems. After a brief introduction in chapter 1, chapter 2 discusses the current extracted contextual information within cybersecurity studies, and how machine learning accomplishes a variety of cybersecurity goals. Chapter 3 introduces an attacker injection model, championing the adversarial methods. Then, chapter 4 extracts contextual data and provides an intelligent machine learning technique to mitigate anomalous behaviours. Chapter 5 explores the feasibility of adopting a similar defensive methodology in the cyber-physical domain, and future directions are presented in chapter 6. Particularly, we begin this thesis by explaining the need for further improvements in cybersecurity using contextual information and discuss its feasibility, now that ubiquitous sensors exist in our everyday lives. These sensors often show a high correlation with user identity in surprising combinations. Our first contribution lay within the domain of Mobile CrowdSensing (MCS). Despite its benefits, MCS requires proper security solutions to prevent various attacks, notably injection attacks. Our smart-injection model, SINAM, monitors data traffic in an online-learning manner, simulating an injection model with undetection rates of 99%. SINAM leverages contextual similarities within a given sensing campaign to mimic anomalous injections. On the flip-side, we investigate how contextual features can be utilized to improve authentication methods in an enterprise context. Also motivated by the emergence of omnipresent mobile devices, we expand the Spatio-temporal features of unfolding contexts by introducing three contextual metrics: document shareability, document valuation, and user cooperation. These metrics are vetted against modern machine learning techniques and achieved an average of 87% successful authentication attempts. Our third contribution aims to further improve such results but introducing a Smart Enterprise Access Control (SEAC) technique. Combining the new contextual metrics with SEAC achieved an authenticity precision of 99% and a recall of 97%. Finally, the last contribution is an introductory study on risk analysis and mitigation using context. Here, cyber-physical coupling metrics are created to extract a precise representation of unfolding contexts in the medical field. The presented consensus algorithm achieves initial system conveniences and security ratings of 88% and 97% with these news metrics. Even as a feasibility study, physical context extraction shows good promise in improving cybersecurity decisions. In short, machine learning is a powerful tool when coupled with contextual data and is applicable across many industries. Our contributions show how the engineering of contextual features, adversarial and defensive methods can produce applicable solutions in cybersecurity, despite minor shortcomings.

Acknowledgements

I'm eternally grateful to both my supervisors, Dr. Kantarci and Dr. Erol-Kantarci, for not only their guidance but also their amazing flexibility and understanding, especially in difficult times. This thesis would not have been possible without them. I aspire to someday provide to others as they provide to their students.

I remember meeting and collaborating with Dr. Kantarci during my undergraduate degree. I knew right away that if graduate school was in my future, it was under his guidance. Fast-forward two years, and I now have the pleasure of adding his name to my thesis. Thanks to both him and Dr. Erol-Kantarci, I have become the scientist and engineer that I am today.

I would also like to express my gratitude to our industry partners at Blackberry for their involvement in our publications. Their industry knowledge was invaluable to the completion of my work, as were NSERC ENGAGE and OCE VIP1 programs, and other organizations that supported me along my journey.

Naturally, all the NEXTCON and NETCORE lab members have provided amazing support (and many cups of coffee) during this phase of my academic career. I hope to maintain those newfound friendships.

Finally, I would like to thank Dr. Peyton and all the other faculty professors for their guidance and teaching opportunities. A special thanks to M. Tapuc for being my biggest fan and always believing in me, and to my family members for their ceaseless support and guidance throughout my Master's study.

Table of Contents

List of Tables

List of Figures

List of Equations

Glossary

CC **C**ontinuous **C**oupling: Metric given to a pair of entities which correspond to how long such entities have been coupled within a given timeframe. See chapter 5 for more details. 5, 56–61

Co User **Co**operation: An extracted contextual property that quantifies the cooperation between peers working on shared documents in a cyber system. 4, 5, 68

DBSCAN **D**ensity **B**ased **S**patial **C**lustering of **A**pplications with **N**oise : Unsupervised clustering algorithm which creates clusters by associating nearby points together and marking any instances not within a cluster as an outlier. DBSCAN is a common algorithm in the state-of-the-art of unsupervised machine learning. 17, 21, 22, 29, 31, 34, 35, 72, 74, 89

IC **I**nstantaneous **C**oupling: A derived metric which is given to a pair of entities that correspond to how frequently these entities have been coupled together. See chapter 5 for more details. 5, 56, 59–61

IoT **I**nternet **o**f **T**hings: The principal of having a wide variety of uniquely identifiable devices which can communicate in a device-to-device manner through the internet without the need for human interaction 1, 8, 10, 13, 14, 16, 18, 36, 69

MCS **M**obile **C**rowd**S**ensing: A data collection technique that uses the abundances of available sensors within a crowd of mobile devices in a given environment to produce intelligent data-driven solutions. 4, 5, 8, 13, 18–20, 23, 27, 35, 36, 67, 71, 72

RBAC **R**ole-**B**ased **A**ccess **C**ontrol: An access control methodology that assigns roles to individuals within a system as a means of providing access. This can include additional layers, such as allowing for only subsets of actions to be performed by certain roles on certain asset types. 15, 38–41, 44

SEAC **S**mart **E**nterprise **A**ccess **C**ontrol: An adaptive weighted-voting machine learning technique which provides access control based on previous machine learning predictions called "members". See chapter 4 for more details. 5, 6, 39, 46–49, 68, 72, 73

Sh Document **Va**luation: An extracted contextual property in cyber environments which produces an estimate on the values of a document over time. 4, 5, 68

SINAM **S**mart **IN**jection **A**dversarial **M**odel: An intelligent data injection model which leverages the participation of accomplices in a crowdsensing campaign to create context-aware injections using unsupervised machine learning. 4, 5, 18–20, 22–25, 27, 29–31, 34, 35, 71, 72

Va Document **Sh**areability: This is an extracted contextual property that quantifies (relatively) how frequently a document is shared amongst peers is a cyber system over time. 4, 5, 68

Publications

- **K. Quintal**, B. Kantarci, M. Erol-Kantarci, A. Malton, and A. Walenstein, "Contextual, Behavioral and Biometric Signatures for Continuous Authentication", *IEEE Internet Computing Journal 2019*, vol. 23, no. 5, pp. 18-28, 1 Sept.-Oct. 2019, doi: 10.1109/MIC.2019.2941391.

- **K. Quintal**, B. Kantarci, M. Erol-Kantarci, A. Malton, and A. Walenstein, "Enterprise Security with Adaptive Ensemble Learning on Cooperation and Interaction Patterns," *2020 IEEE 17th Annual Consumer Communications and Networking Conference (CCNC)*, Las Vegas, NV, USA, 2020, pp. 1-7, doi: 10.1109/CCNC46108.2020.9045328.

- **K. Quintal**, E. Kara, M. Simsek, B. Kantarci, and H. Viktor. Sensory data-drivenmodeling of adversaries in mobile crowdsensing platforms. In2019 IEEE Global Com-munications Conference (GLOBECOM), Waikoloa, HI, USA, 2019, pp. 1-6, doi: 10.1109/GLOBECOM38437.2019.9014288.

Patents

- A. Walenstein, A. Malton, **K. Quintal**, M. Erol-Kantarci, B. Kantarci "Document Management System Having Context-Based Access Control and Related Methods", **U.S. Patent Filed in July 2019**

Chapter 1

Introduction

The notion of context-awareness has been notably increasing in the cybersecurity industry. The application of new innovative strategies and protocols are self-sustainable proof of a need for more secure, context-aware cybersecurity solutions [102]. Such scientific advances are constantly evolving; from advanced authentication methods seeing a conversion of passwords towards biometric dependencies, to cloud computing architectures targeting more dynamically-created architectures [49]. The domain of cybersecurity is undergoing a major refactoring phase, in part caused by the appearance of new technologies, such as artificial intelligence [116].

1.1 Motivation

Our motivations in investigating context-aware authentication methods within cybersecurity are as follows: to improve the context-related decisions which are made during authentication methodologies. In other words, introducing additional context-related metrics and including more intelligent machine learning authentication techniques to improving the sophistication of cybersecurity solutions. With such continuous improvements being made to Big Data systems, crowd-sourcing applications and IoT devices, the need for proper cybersecurity solutions are only increasing [69], and additional contributions will be presented in the upcoming chapters.

Moreover, this thesis specifically focuses on the area of authentication within cybersecurity. Authentication techniques have seen promising advancements in recent years, such as multi-factor authentication [10] and biometric authentication [31]. Despite their advantages, they still contain shortcomings in minimizing user interactivity and maximizing security. This trade-off can be further investigated by introducing context-aware strategies.

Therefore, let's clearly define our definition of context: a context contains all the encompassing information and properties within an event, declared statements, or any relative ideas. The scope of this study focuses solely on the context of occurring events, as they are

the most pertinent option in cybersecurity. To help clarify, here is an example: let's assume professor Smith teaches a class on Tuesday evenings at his local university. The occurring event in question is the teaching of his lecture; the context of this event includes the number of students attending, the start and end time of his lecture, the temperature of the room, the number of mobile devices in the room, and much more. Therefore, knowing the context of professor Smith's class may seem trivial at a glance, but the requirements of fully extracting *all* the available contextual details of every lecture is unfeasible. However, if the extraction of only the *pertinent* information was possible, then a system could (for instance) authenticate all the attending students to their courses web-portal. The quantification of *pertinent* is open to interpretation and should be one of the main focuses for future data engineers.

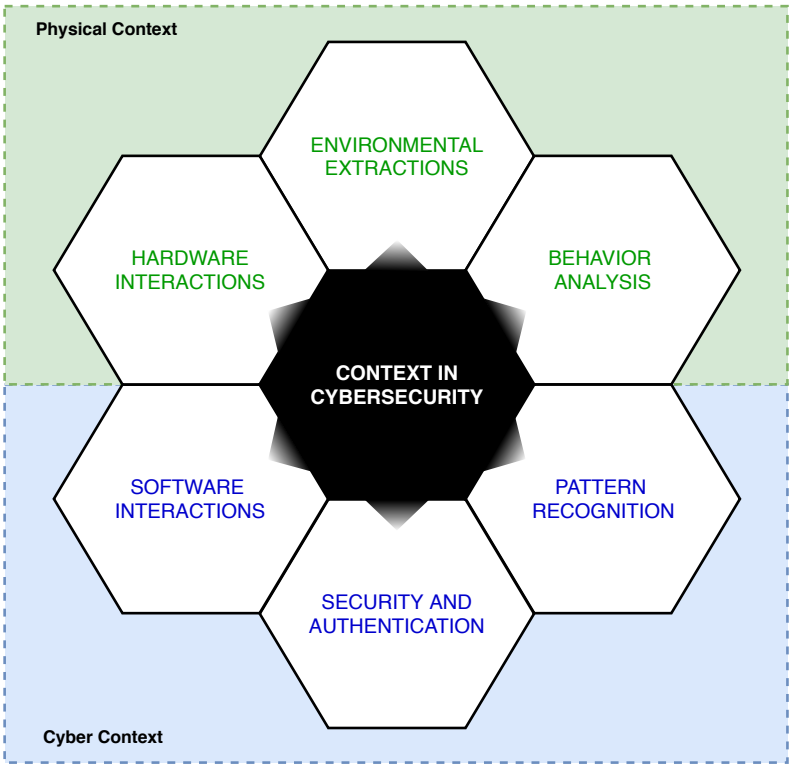


Figure 1.1: Two major contextual components in our cybersecurity studies

Such events and their corresponding properties (referred to as **contextual properties**) have already been investigated in various studies [55] and fall within two major categories. First, the concept of physical contexts, which pertains to all the characteristics of a context related to the physical environment. Second, events that occur in a cyber context, which revolve mostly around software details. Figure 1.1 represents the scope of this thesis and demonstrates how these two categories encompass the aspects of cybersecurity in this thesis. Chapters 3 and 4 are bound within the cyber domain, as are their contextual analysis's. Chapter 5 explores potential improvements as to how context-aware additions might improve cyber-physical cybersecurity decisions.

1.2 Objectives

A series of methods are proposed in this thesis to further integrate applications of contextual analysis within cybersecurity solutions. The aforementioned objective is further categorized into two distinct goals: creating intelligent applications with the intent of infiltrating cybersecurity systems and providing mechanisms to identify and prevent such malicious attempts. These overarching objectives are reached by introducing context-aware strategies that improve adversarial and defensive machine learning techniques. The analysis of the cyber-physical events that appear in the cybersecurity world is suspected to contain large amounts of concealed data goldmines [72].

The opening objective is the creation of an adversarial model to be used for quality assurance purposes. This adversarial model monitors the context of a given individual and intelligently deceives a data-collection system in accepting data not originating from an authenticated source. With the presence of this model, future solutions in context-awareness can be vetted using this model.

Another objective is identifying contextual anomalies in cybersecurity systems. With the ability to accurately raise and identify different threats, solutions can provide continuous, non-intrusive authentication to users of a participating cyber system, only requiring input from users in cases of high uncertainty. This notion of *continuous authentication* attempts to solve the requirement of devices constantly demanding identity-related information during prolonged periods of use. With the constant flow of available contextual information (and newly derived metrics), participants in different systems could, therefore, have their authentication states automatically update, and allow for longer, non-intrusive periods of use. The participants would only be prompted when critical events are triggered based on higher amounts of authentication certainty (such as using a mobile banking application).

An overview of global objectives is presented in figure 1.2 to demonstrate the process of contextual extraction being used for either objective. In this figure, the authentication in the upper-section would only be triggered if the defensive machine learning in the lower-section didn't have sufficient confidence to continuously authentication a particular user. There are vital components of authentication in the current state-of-the-art, which are mostly covered by exploring the major scientific advances in context-awareness within cybersecurity. This also includes the mention of authentication methods in both cyber and physical domains, recent advances in machine learning methods or any combinations of such. It is also worth noting that cryptography, blockchain, and different types of network analyses are not within the scope of this thesis, despite remaining critical components in the domain of cybersecurity [4].

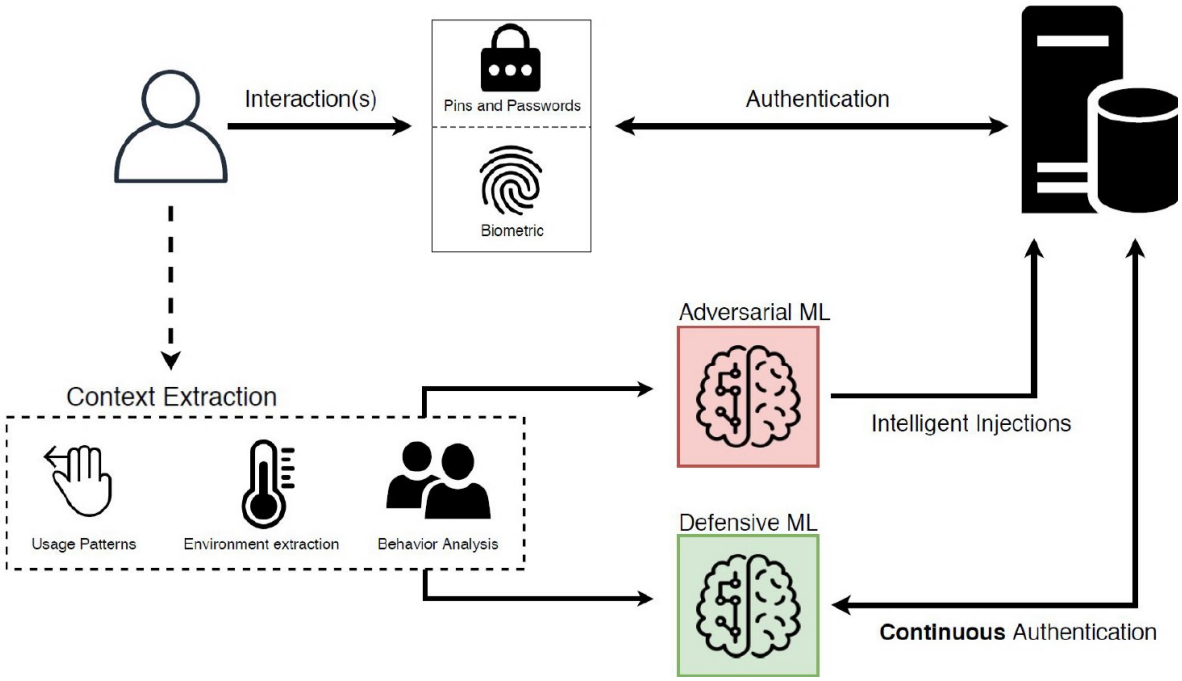


Figure 1.2: Combining Context with Machine Learning in Cybersecurity

1.3 Contributions

All presented contributions, which correspond to the objectives described in section 1.2, can be summarized as follows:

- First, we present an adversarial machine learning method to be used within the domain of MCS. This method, termed as SINAM, is an online-learning injection technique that creates malicious injections within a sensing campaign. Using intelligent behavioral analysis, SINAM successfully injects data into the campaign with low detection rates. This method also contains tunable parameters that allow it's employment to fit the context of different studies.

- Second, the focus is shifted towards defensive machine learning where we contribute to the cyber domain of cybersecurity. Specifically, we analyze the cyber context of a simulated enterprise environment to produce context-specific metrics that allow the extraction of contextual cyber-properties. We generate a total of three metrics to be used in a variety of fields, and explain how these metrics are correlated with authentic user identity in cybersecurity. The three derived metrics are document shareability(Sh), document valuation (Va) and user cooperation (Co). Full details can be found in chapter 4.

- Third, we expand on the previous contribution by providing an intelligent defensive machine learning technique. This **S**mart **E**nterprise **A**ccess **C**ontrol technique is an intelligent machine learning ensemble method that leverages a weighted-voting algorithm to associate trust to it's underlying machine learning models. SEAC is compared with competitive machine learning models in demonstrating how SEAC offers a higher security rating that its machine learning competitors.

- Fourth, the final contribution is a preliminary investigation in extracting and utilizing cyber-physical properties in cybersecurity. By intertwining characteristics in physical and cyber contexts', the extraction of two coupling metrics (CC and IC) from a hospital environment are deployed into a newly introduced context-aware algorithm. This algorithm acts as an intelligent agent that broadens the availability of security rulings in authentication-based systems. Despite being preliminary research, our outcomes show great promise when vetted against a provided security policy. See chapter 5 for the extensive particulars.

1.4 Thesis Outline

The contents of this thesis are spread across six chapters. As this chapter comes to a close, chapter 2 will present a literature review on the current studies in context awareness and machine learning within cybersecurity. Particularly, it discusses how context is integrated with current cybersecurity solutions (section 2.1). This section is further divided to distinguish between the applications of behavior analysis (section 2.1.1), biometrics (section 2.1.2) and other modern techniques (2.1.5) that use context in their cybersecurity decisions. Finally, section 2.2 discusses the contributions to the cyber domain specifically and is follow by a section on physical domain applications in cybersecurity (section 2.3.

Chapter 3 is the sole chapter on adversarial machine learning. Specifically, we demonstrate how an aggressive context-aware attacker model can infiltrate a MCS system. This is achieved by monitoring sensory data in an online manner and fooling defensive machine learning models using a technique called SINAM. This chapter naturally begins by discussing common types of attacker modeling in MCS systems (section 3.3). Then, the experimental environment is described in section 3.3.1, which allows for a dive deep into how behaviour analysis is conducted (section 3.3.2). Then, the explanation of the injection strategy in detailed in section 3.3.3. The chapter is concluded by the performance analysis section 3.4.

Shifting the focus to defensive context-aware machine learning, chapter 4 discusses to extract contextual properties from the inner workings of an enterprise network to obtain more intelligent authentication decisions. First, current access control standards are presented as a start-off point (section 4.1). Next, feature engineering is introduced (section 4.2) by presenting formulations on the three extracted contextual metrics: the quantification of how much a document is shared (Sh), how much value a document is perceived to hold (Va), and how frequently users cooperate with one another (Co) in sections 4.2.1, 4.2.2 and 4.2.3

respectively. Once defined, these metrics are combined with an intelligent classifier technique called SEAC, which stands for a Smart Enterprise Access Control - section 4.3. The results of the industrial experiment are presented in section 4.4, which have tunable parameters regarding the security-convenience trade-off, detailed in section 4.5.

In chapter 5, investigative research is performed on extracting contextual properties in the cyber-physical domain. First and foremost, the dissimilarities between both domains are elaborately explained in section 5.1. Afterward, the provided policy-based decisions — the answer sheet for this section — are explained to underline the key differences between obtained results and the expected decisions (section 5.2). Next comes another section on feature engineering (section 5.3), explaining the mathematical constructs behind both coupling metrics (subsections 5.3.1 and 5.3.2). Once the experimental details are unveiled, the consensus model is inspected in section 5.4. The initial set of results are thereupon presented and analyzed in section 5.5.

In closing, chapter 6 summarizes the contributions of each chapter, alongside their key takeaways and correlations. Furthermore, remaining challenges are discussed (section 6.1), and additional opportunities in context-awareness within cybersecurity are also brought to light (section 6.2). Future directions are discussed in section 6.3, as to guide any extension to chapters 3, 4, 5 with subsections 6.3.1, 6.3.2 and 6.3.3, respectively.

Chapter 2

Context-Awareness in Cybersecurity

The advances in data analytics and machine intelligence have led to innovative solutions in identification and authentication methods. Such advances include, but are not limited to, mining and recognizing behavioural patterns, extracting contextual properties and analyzing cyber and physical environments. Such techniques call for effective solutions to cope with their variability under different circumstances. Such solutions are often presented with the inclusion of machine intelligence in defending specific networks, or to infiltrate them for testing purposes. Such presented solutions should not be seen as replacements to conventional authentication techniques (such as multi-factor authentication) but instead, as powerful improvements [86]. With such additions, authentication mechanisms could become more convenient and less intrusive, all the while improving security metrics.

Being context-aware is seen as inferring and/or extracting additional information for the context in which a solution is derived from/in. Such a concept does unfortunately retain many technical limitations across multiple domains. In each, common strategies (such as data mining and feature engineering) extract the additional insights required for such context-aware cybersecurity improvements [97, 113, 37, 51, 8]. All of these strategies once again depend on their environment; as contexts fluctuate in variety, so would the requirements of each given solution. For instance, contexts which appear in an enterprise setting differ in magnitude and scope from those which occur on a university campus, or even a medical environment.

Furthermore, not only is the behaviour of participants a noteworthy source of data acquisition but so is the unfolding environment in which each behaviour is being conducted. Let's be clear: the behaviour of an individual encompasses what they can control, such as their movements, actions, reactions and thoughts. The unfolding context, on the other hand, not only includes every situational detail, which may or may not include one or multiple agents. For clarity, let's introduce Jane, who is waiting in line at a coffee shop. Her behaviour is passive; she replies to emails on her phones (cyber proximity), intermittently walking forwards in line until her turn arrives to place an order. Jane's movements and decisions are what we define as *behavioral properties*, as they directly correlated with the

behaviour of Jane (how fast she walks, which emails she opens, etc). On the other hand, the entire context of this example is quite dynamic. Baristas assemble an assortment of drinks behind the counter, other customers enter and exit the waiting line, and discussions occur at nearby tables, all within Jane's physical proximity. Furthermore, the environment would also see a variation in temperature every time the door opens, and new clients enter the store. All of these minute details encompass Jane's current context, even if they are not directly within her control. Responses and reactions do arise based on Jane's behaviour, causing chain-reactions of events to occur in the context. This type of contextual analysis (at a much more granular level) is what empowers machine intelligent strategies to appear in the cybersecurity literature. Simply put, further advances in cybersecurity are most likely to include a level of context-awareness, as more advanced threats are emerging in cybersecurity [39]. Such emerging threats, therefore, call for advanced frameworks that must consider the many different types of cyberattacks in different environments [39]. An analysis of extreme depth is required to extract both the behavioural and contextual information of any given situation. Alongside artificial intelligence and feature engineering, Jane could receive some context-aware authentication assistance, allowing her to continuously use her devices without the need to enter passwords or pins every few minutes. This means continuously authenticating her sessions based on her behaviour, but also sometimes requiring additional input if a higher-risk action is taken (if, for instance, she wants to open a highly private document). Moving away from this simplistic example, let's discuss more concrete strategies on how context is utilized to advance the cybersecurity of digital systems.

2.1 Context in Authentication methods

Authentication has already proven to be an efficient and effective way of adding layers of security to digital systems. However, within the past decade, a particular focus has been made on minimize the required input to maintain acceptable levels of authentication [18]. The most common and seemingly valid solution in achieving this requirement is the integration of biometrics [51]. The variation in biometric types are covered in additional detail in section 2.1.2. The current focus of this section is on how the literature uses contextual properties in producing more intelligent authentication strategies as a whole.

When environments are in constant flux, the resulting changes in context generate valuable pieces of information [77]. These different patterns emerge with great corresponding identity associations, which can be leverage in varying areas (IoT, MCS, Cybersecurity) [77]. Often enough, however, the context itself can be seen as an authentication factor by inspecting the surroundings of a device's sensory readings [77]. More specifically, two mobile devices that obtain similar environmental sensory readings can authenticate themselves within a given space. This becomes possible when devices have sufficiently similar entropy readings from the environment. Specifically, their entropy calculations combine readings of ambient noise levels with environmental readings in different areas (e.g. the difference in readings between your office and home locations). The authors conclude that only with

enough time to analyze the entropy of each particular environment can contextual authentication become possible [77]. The quantification of *enough* does entirely depend on the conducted study [77]. The unification of all those environmental sensors leads to high computational complexity, which revokes its application within real-time scenarios. Nonetheless, adaptable authentication rules become an interesting concept when you consider the constant change of context in our lives; the authors' use of ambient noise for such adaptability demonstrates significant potential.

Another context-aware study developed software called SmartAuth [92]. Presented by Preuveneers et al. as a context-aware authentication system [92], SmartAuth suggests a zero-interaction authentication scheme that extracts digital features from smart devices on a local level (system language, colour depth, screen resolution, timezone, OS platform, plugins) as well as communication details (IP range, headers, time of access, geolocation). Both aspects are combined to authenticate users using machine learning (Hoeffding trees) [92]. Their authentication results improved as the number of successful interactions rose, improving the correlation between a particular set of features and the identity of a particular individual [92]. In other words, the more users interact with their devices (and SmartAuth collects data), the easier it becomes to authenticate them since their behaviour becomes commonality associated with each respective context. This pattern appears in other mobile studies as well [44] Often, geolocation is highlighted as the most important feature in authentication decisions [46], and although it remains important, a case-by-case analysis is still required to confirm whether geolocation is the most contributing feature in every particular context.

Another interesting question is further raised in [103] where the authors investigate how continuous authentication could be achieved on devices with no user interface. For example, smartwatches and other Body-Area Network sensors may not be equipped with keyboards or touchscreens, thus limiting authentication methods. Using the interdependence of Apple products, the authors performed a case study to analyze participants' gait, echocardiogram signals, among other health metrics. Their top participants were able to achieve authentication rates around 90% [103]. However, their results require sensors to be nearby (within a six meters radius), and could only authenticate a particular selected user, and not all. As concluded by the authors, the higher the pool of features within this context, the further the analysis can be pursued for continuous authentication, but for such interdependent devices, the scope of context is rather small [103].

Such studies are interesting enough on their own, and even more so when we discuss their relevance to our personal lives in the near future. Vulnerability analysis of cybersecurity will soon be relevant to all humans, as the concept of fully-integrated smart-homes is becoming increasingly feasible [65]. This topic has not yet seen full maturity, as has yet to see deployment in the entire economy [80]. Whenever that day does occur, despite what the past and present standards are for cybersecurity, there will arguably be even more details to iron out. The author makes a valid point: you can't know what you don't know [80].

Shifting gears towards a military environment, some applications could benefit from contextual extraction and analysis. A conceptual study presented by Castiglione et al. report

of many available biometric features being integrated into a military setting [25]. Sensors located on soldiers, weapons and vehicles can all provide additional sensory information, which allows the extraction of valuable insights in military situations[25]. Such information includes both static (heart-rate) and dynamic (gestures, facial expressions) sensory data, alongside environmental metrics (e.g, operative conditions, environmental conditions, active sensors, subject motion). Notably, these features provide sufficient information for the identification of a soldier's health status[25]. Their concept of the Internet of Military Things contains a variety of dynamic sensors that results in large volumes of valuable information, which requires larger memory space. This recurring theme of extracting high volumes of contextual details often comes at the cost of higher computational requirements [25].

In a vehicular environment, driving efficiency, safety and convenience are all desirable metrics. Such considerations are taken in [45], which proposes a verification system in vehicular platoon admission which relies on vehicular context. More specifically, road and traffic conditions are extracted to verify and identify a vehicle requesting to join a moving vehicular cluster. Such road properties are combined with environmental readings to determine the authenticity of the admitting vehicle [45]. Their proposed system also considers an attacker model, as malicious entry to the platoon could cause tremendous damage. Performance results demonstrate accurate admissions of around 91% when within the same lane, and 82% when crossing multiple lanes [45]. The overwhelming details and complexities in such a domain produce many difficult and unexpected challenges. As a worthy side-note, the driving behaviour of those behind the wheel could also be considered amongst the changing contextual details [107]. Such details could potentially also provide driving assistance in vehicular contexts, also more research is still required in this field [107]. Thus, authentication is a concept not only relevant to the human context but also worth investigation within the domain of autonomous vehicles and IoT.

2.1.1 Behavior Analysis and Authentication

Although these different environments produce fascinating variations in context, the behaviours of participants in a given context contain valuable information [112], if not more than the context itself [100]. Such a concept, especially when analyzing human behaviour to determine one's authenticity, has been gaining constant interest in modern literature, especially when mobile devices are involved [119]. The immense surge of available and heterogeneous data volumes created in these ubiquitous technology-dense environments has led to rational behaviour analysis and predictions.

Our first example discusses the different social networks which continuously fight for our attention. This constant need for continuous connectivity is only every increasing in popularity within the mobile domain [88]. As such, the notion of deriving a *sociability* metric become of interest [7]. Anjomshoa et al. introduce this metric by monitoring social networking behaviour patterns (i.e. social networking data usage, usage frequency, time spent on a particular social network application, etc.) to continuously authenticate participants on their

mobile devices. Powered by artificial intelligence[7], their continuous authentication method allows uninterrupted usage of mobile devices over 90% of the time. Not only did their results demonstrate reasonable convenience, but they were able to detect anomalous behaviour with an accuracy of up to 97% [7]. Such sociability features that can be extracted from a participant's social context show potential in improving the robustness of a given system, all the while increasing the participants' comfort level by minimizing required authentication interactions[7]. One fallback of constantly polling data from theses various social platforms is the computational requirement imposed on each device. Participants might not be willing to forgo their battery life in exchange for continuous authentication, but more research is required on that front as per the authors [7].

Continuing the trend on mobile device interactions, the study in [37] investigates various mobile behavioural aspects to try and approximate user authenticity. These aspects include what text is inputted (including idiosyncrasies), which applications are most used, the participant's browser history (tagged with timestamps) and spatial-temporal geographical data. The authors, despite their initial successful results in authenticating participants [109], introduced further improvements in [37] where they included the input frequency (time-series) in improving their artificial intelligence models. Using these behavioural patterns, they were able to authenticate participates with error rates as low as 1%. Indeed, even with only a limited set of available features, acceptable authentication rates were achieved [37]. The authors did conclude, however, that even with low error rates, location-related features contain the most influential information when making authentication decisions. Despite the benefits of the proposed technique, battery consumption was once again an issue for participants, as the constant polling of data across all components remains a resource-intensive requirement [37].

Next, this similar strategy focuses on participant's behaviours, but in even more various contexts [62]. Here, the authors analyze users' behavioural patterns within their pervasive environments as they went about their day using information related to their laptops. Specifically, they installed software that performed a behavioural analysis to improve the confidence of a participant's identity by detecting behavioural anomalies. Such anomalies fall within using the following categories (taken on their mobile computers): i) Operational context, which related to performed tasks, goals and activities, ii) Interpersonal context, which denotes the information portraying to relationships and communications within that participant's cyber and social applications, iii) Spatial context, which relates to a participant's location and environmental properties at the time of access [62]. These features were combined to create what the authors define as *Contextual Identity*. This identity was then learned using a k-means clustering algorithm and vetted on future behaviours, which when non-compliant with their *Contextual Identity*, produced anomalies [62]. Such research introduces an interesting question on behavioural signatures: even with the ability to identify users based on their context, how should these solutions react when faced with anomalies? In such cases, maybe the participant is simply entering a new context which could be learned dynamically (potentially with online learning) [62]. Precautions could be implemented to keep out these anomalies, assuming there are malicious, but that is a very bold assumption.

It becomes increasingly difficult to detect such differences under rapid context changes [30].

Ending this section on behavioural authentication methods, the authors in [97] appended their previous work on sociability analysis [7] by extending the scope of their research to include another behaviour metric: gesture patterns. This includes the various swipe directions, tap and holds, amongst others. These gestures are analyzed and create gesture-specific authentication models using the random forest supervised learning algorithm [97]. In combination with their previous work on creating the *sociability* metric in [7], the authors propose a dual authentication model. Fetching gesture patterns alongside social platform usage, Rauen et al. achieved near-perfect continuous authentication, attaining 99% accuracy during their experiment [97]. Not only are such results great for authentication purposes, but anomaly detection was also successfully achieved with error rates under 3% [97]. This study is another great example of the use of artificial intelligence can empower behavioural analysis and produce powerful authentication techniques.

2.1.2 Adapting Biometrics to Context

Biometric authentication has now become one of the most successful methods of authentication [74]. Such techniques are known to require low levels of user interaction, which in turn limits the time a user needs to authentication themselves, allowing for more frequent use, without a cost to security. Theses many solid and notable contributions have found themselves influencing the realm of continuous authentication, especially when we consider using such biometric solutions in collaboration with contextual information, as seen in the studies that follow.

The first attempts at improving authentication robustness included the combination of multiple methods in the same process. Often called multi-factor authentication, this strategy improves security, but at the cost of requiring additional user interactions [48]. However, with biometrics, the same concept could be applied, but by performing two biometrics methods (i.e. using two different sensors at once) in parallel. This is exactly what was introduced in [70], where they coined the methodology as Multimodal Biometric Authentication. This method improves the convenience-security trade-off previously imbalanced in multi-factor methods. For instance, Multimodal Biometric Authentication could require fingerprint authentication while collaterally providing facial recognition. The biometrics methods selected in [70] does depend on the context of a given participant; you cannot provide facial recognition if the user is in a dark room, for instance. Minimizing such a trade-off is also something we investigate further in both chapter 4 and 5.

Moving into the domain of cloud computing, an interesting authentication strategy was introduced using a participant's context alongside Support Vector Machines, or SVM's [113]. Here, the authors analyze the association between the different features of a cloud system, with a focus on improving the User Experience (UX), which relates to the minimal amount of required input for sustained sessions. Particularly, their proposed algorithm considers the authentication habits of users, at what time authentication was normally performed, from

which device access it was granted, and at which location [113]. By using those features, their algorithm selects the most historically appropriate biometric method to use. Of course, only adequate biometric methods are presented to the user, based on the given context (similar to a previous study). The integration of behaviometrics in this study demonstrates the need to bridge the two domains (context-awareness and biometrics) to achieve efficient continuous authentication [113]. Unlike most mobile-based systems, this study stands out as a GPS-free solution. Since standard mobile anomaly detection systems often depend on location in large areas or without geographical limitations, this work could provide valuable authentication improvements when localization is unavailable [113].

2.1.3 Simulation of malicious behavior

When providing innovative authentication solutions, a strong emphasis is always made on ensuring common infiltration patterns are efficiently addressed. This can range from detecting anomalies [7], to identifying cyberattacks [85]. Secure authentication schemes should henceforth almost always consider an attacker model and risk analysis when preventing malicious attempts. Unfortunately, it remains an issue to be able to identify such models, and increasingly difficult to simulate them. For instance, it was noted in [113] that within a GPS-central solution, estimating malicious behaviour and considering anomalies requires historical data that is labelled as such, which is not always obtainable. Similarly, tracking sociability rates and gestures within smart mobile devices for authentication contains its challenges, since spoofing patterns must also be labelled up front [97]. This concept is not only an issue in the authentication domain, but across many areas in literature [64] [28] [50].

2.1.4 Sufficient and proper quality data

In almost any study within MCS and IoT, obtaining precise, accurate, trustworthy ubiquitous data remains an unsatisfied request [111]. Of course, not only do real-world applications have their technical challenges, but they also try to meet this same requirement. Similarly, many research environments still lack large pools of heterogeneous participants [7] [97] [113] [110]. Even in simulated settings, data (such as behavioural patterns [92]) is generated based on statistical models derived from a few representative participants. Such resulting data sets can come in various degrees of volume and/or heterogeneity, but the natural entropy of realistic scenarios is not a concept that is easily modelled [29]. Furthermore, participants in most of the highlighted studies in table 2.1 have no incentives when installing such research tools on their devices [79], which causes the data frequency to decline, as the lack of motivation seems directly proportionate to participants data collection [38].

As mobile devices have significantly improved in computing specifications over the past few years, extracting contextual information becomes a trivial task for modern devices, but remains challenging for older generations. For instance, certain applications require constant environment polling to minimize errors and increase convenience [37] [97]. Furthermore,

memory limitations are also a concern when extracting data from a wide variety of non-dedicated sensors [25]. Such strenuous demands of resources become a delicate balance, as IoT device applications are only growing in numbers. A potential solution for such resource demands could be using edge computing to cut down on latency, also pointed out by Castiglione [25]. Installing edge nodes into our physical environments could allow for a more wide-spread availability of digital resources. This could also allow the addition of more dedicated sensors, decreasing the memory requirements on such non-dedicated sensors, but now the enigma becomes paradoxical (the more resources, the more sensors can be supported, so the more resources are available, and so on). This results in a high amount of data being produced if this type of edge computing architecture were to be implemented [71], and would, unfortunately, require the upgrade of existing infrastructures, alongside the application of intelligent data filtering and throttling strategies [68].

Another aspect to consider would be the accuracy of all the obtained data. Sensor readings often contain small but yet significant errors. These tiny variations can have significant snowball effects in injecting noisy data patterns, thus nullifying the validity of datasets [13]. This phenomenon, exasperated across multiple devices, compromises the much-required accuracy for acceptable results in highly sensitive and/or concurrent tasks [103]. Therefore the accuracy of data collection and extraction must be as meticulous for smaller data volumes just as large ones, as to make noisy data considerations when scaling applications. The selection of optimal features based on their accuracy then becomes another challenging issue concerning data collection [53]. When a wide variety of features are available, additional analysis is required to distinguish the potential value each feature could provide. This is especially valid within the domain of biometrics, where the most optimal features fluctuate depending on personas and contexts [51].

Within the reviewed papers in this section and table 2.1, certain techniques investigate similarities in context within their decision models [62] [39] [94]. As such, there is a direct dependency on co-present devices when extracting context which becomes challenging in lower device-dense environments [76] [94][95]. On the contrary, in highly dense environments, authentication becomes increasingly feasible, but at the expense of needed considerations of attack threats. This point is discussed in [70], where brute force intrusions were nullified by increasing the guessing complexities of identifiers. However, this is but one of the many threats that could potentially arise from a highly dense environment.

2.1.5 Ethics in Authentication

Although there has been a wide range of new authentication methods recently, most of them still require some level of interaction with individuals [10]. Such methods provide higher levels of security in a single or a few technological areas, but it's very rare that a single method can be applied to all domains of IoT [101]. Furthermore, with all the readily available data that is emerging in the areas of big data and IoT [1], innovations can erupt, providing quite fruitful technological advances. Important considerations are however required with all this

data, especially in the domain of ethics and law [82].

Furthermore, an extensive survey was conducted in [6], focusing on the analysis of different social authentication applications. In detail, these applications were compared based on the variety of data located in their local (on the device itself) and online (in the cloud) environments [6]. Their analysis further included information regarding which features of each application were most frequently used, how effective participants were able to perform tasks, and discovered any potential threats to the participant's data. Three aspects were underlined when scoring the applications amongst each other: security, deployability and usability of the application [6].

These metrics were further analyzed and considered to be of critical importance, as many emerging social authentication applications are emerging, alongside respective trade-offs that need to be further analyzed [81]. As an increasingly popular and time-consuming component of current and upcoming generations, trustworthiness and effective communication are key components required from participants in obtaining good cybersecurity. Values like participant privacy, trust, identity, and application security, deployability and usability only become every increasingly important consideration [6][81].

2.2 Authentication in Cyber Domains

The prevalence of appearing cybersecurity issues in different industry networks are prompted by technological advances [26], as the difficult task of assessing cybersecurity risks is ever increasing. A common strategy for such problems is the application of RBAC systems, as already proven in the eHealth industry [16] [11] [108]. These strategies also include a particular focus on patient security and privacy, as they are essential components in making risk-orientated decisions [15]. Despite the many Bayesian approaches proving their fruitfulness for such risk assessments under cyber circumstances, there remains a gap in achieving true cyber-physical awareness[17]. Including context-aware solutions within the risk-analysis process is also a valid consideration to take when addressing such cybersecurity issues[35].

The ability to continuously monitor the risk, vulnerabilities and threats within certain systems has become an area of increasing complexity, especially with such rapid technological improvements [56]. Our cyberspace does not contain sufficient information for emergency decisions to adequately be made; additional situational (or contextual) details become a necessity in such crucial situations [60]. These situations do contain additional contextual properties, which when properly and efficiently extracted, can also provide valuable insights towards the selection of an optimal system response. As a steady dependence on cyber-physical systems increases, so should their trustworthiness of their solutions [19]. A good approach to this problem is once again the use of Bayesian risk graphs, as they can model a vast majority of cybersecurity threats in the many different cyber-physical systems [61].

2.3 Authentication in Physical Domains

There is substantial growth in data production, as evident by big data and IoT environments. Such impending changes will consequently influence the innovations in both cyber and physical data systems, as their data sources become enriched [12].

Artificial intelligence methods have been rising in popularity in recent years, especially as methods of continuously improving the cyber-physical space of particular systems [116]. Furthermore, risk modelling in the health industry is increasingly becoming an obligation, as waves of advances are being achieved to diminish human mortality rates [75]. Having a robust risk-aware evaluation and mitigation system is paramount in maintaining efficiency and secure areas in hospital environments [114]. AI methods have already proven beneficial in predicting unplanned admissions to hospitals [96], therefore the growth of the cybersecurity domain is already starting to engulf cyber-physical properties. Another metric of importance is the intrusion detection rate, which can also be diminished using machine learning techniques [14] [9].

Table 2.1: Scope of Authentication in Cybersecurity

Source	Authentication Domain	Domain Feature(s)	Authentication Method(s)	Pros/Cons
[70]	Adaptive Biometrics	MFA-MB Biometric preferences Location	SVM's by extension	⇑ Continuously improving UX ⇑ High circumvention with biometrics ⇓ Computation time not considered ⇓ Acceptability case-dependent
[113]	Adaptive Biometrics	Location , Light Accelerometer, Calls Battery, Microphone	SVM's	⇑ Persistent results despite high entropy ⇑ Distinct profiles ⇓ Malicious attempts not considered ⇓ Feature weights are device-dependent
[51]	Adaptive Biometrics	Iris recognition Face recognition Fingerprint recognition	NA	⇑ Proven historical reliability ⇑ Great potential with upcoming IoT ⇓ Distinctiveness a challenge ⇓ Traits not persistent ⇓ Many environmental constraints
[62]	Behaviometrics	Phone calls Personal schedule GPS Application usage	k-means clustering	⇑ High confidence rate ⇑ Clear behavior profiles ⇓ Unknown behavior in dense user space ⇓ Unknown contribution per feature
[7]	Behaviometrics	Social interactions Social applications	SVM's DBSCAN	⇑ FRR $< 10\%$ ⇑ 90% Continuous Authentication ⇑ 97% Anomalous detection ⇓ Limited data-set size ⇓ Homogeneous participants ⇓ Dependent on active users
[37]	Behaviometrics	Stylometry, apps, browser and GPS	SVMs Fusion-decision model	⇑ Large and heterogeneous dataset ⇑ Decreasing EER over time ⇓ Homogeneous participants ⇓ Battery intensive ⇓ Dependent on active users
[97]	Behaviometrics	Gesture Patterns	Random Forest Classifier SVM's DBSCAN	⇑ High Accuracy with dual-model ⇑ Low FAR-FRR rates with dual-model ⇓ Limited data-set ⇓ Homogeneous participants ⇓ Dependent on dual-model
[77]	Context-based	Ambient Noise levels	Similarity of fingerprint quantization model	⇑ Low FAR-FRR rates ⇑ Minimizes authentication attempts ⇓ Requires low entropy environment ⇓ Results require long fingerprints ⇓ Limited data-set
[2]	Context-based	Health metrics	NA	⇑ Resilient against cyberattacks ⇑ Low authentication time ⇓ No Learning/Reasoning model ⇓ Dependant on 5G
[103]	Context-based	Gait ECG + CSI signals	NA	⇑ No additional sensors required ⇑ Up to 93% recognition accuracy ⇓ Strict environmental limitations ⇓ Not suitable for sensitive tasks
[76]	Context-based	Ambient light Sound intensity	Fingerprint similarity using fuzzy commitment scheme	⇑ Malicious infiltration difficult ⇑ No user interaction required ⇓ Requires use of long fingerprints ⇓ Dependant on co-present devices
[25]	Context-based	Equipment Sensors Soldier Biometrics Vehicle Metrics	NA	⇑ Wide feature-set extraction ⇑ Provide strategical military advantages ⇓ Requires Edge-computing ⇓ Limited by high data traffic
[45]	Context-based	Road and traffic conditions	Fingerprint similarity using fuzzy commitment scheme	⇑ 92% similarity in same lane ⇑ 81% similarity in multi-lane ⇓ High risk requires low uncertainty ⇓ Requires robust defense against attacks
[92]	Context-based	Fingerprints Client details Server details	Hoeffding Trees	⇑ 99% accuracy after learning ⇑ Minimal performance overhead ⇑ Improved UX from passwords ⇓ Results provided from simulations ⇓ Limited to OpenAM platform

Chapter 3

Context-Aware Adversarial Behavior Modeling

During the design and implementation processes of large-scale cyber systems, large efforts are made on security measures. Such efforts range in scope, depending on the given predefined requirements, but one thing is certain; interconnected controls systems generate very complex architectures, and so intelligent attacker profiles are needed for robust testing of security solutions [93]. These profiles must also be of contrasting nature, as to cover the most cybersecurity vulnerabilities as possible [64]. Not only is it beneficial for testing security measures, but understanding the encompassing details involved in injection attacks on a particular system (for instance) is crucial to understand how to defend against them [28]. The focus of this chapter is to introduce a smart adversarial injection model (SINAM), which can be used as a means to assess the security measures of cyber systems. The domain in which SINAM was vetted on was MCS.

Mobile Crowdsensing empowers smart devices in providing non-dedicated and ubiquitous sensing tasks in environments which particularly lack the presence of other IoT devices [63, 43]. Such sensing tasks — which are commonly referred to as being smaller components of an overarching sensing campaign [54] — continuously acquire sensory data from heterogeneous nodes, generating large amounts of data volumes. Widely referred to as the big data phenomenon, all of this "big sensed data" [83] is a key enabler for various applications of smart cities and spaces [106, 91]. There is much-untapped potential in using MCS as an integral component of IoT-based services, as proven by the ongoing research of such smart applications [22, 21, 23].

With all this potential, the ubiquity of sensing devices and their corresponding participatory and opportunistic sensing services introduce the challenge of trustworthiness assurance against malicious participants [34, 42]. Malicious and untrustworthy participants in a crowdsensing network can leverage many system vulnerabilities for either personal gain or to provide disinformation to crowdsensing servers and platforms for external gain [54, 78, 90]. Such threats — sourced by malicious participants — can vary from information tampering

and privacy breaches to service denials and injections attacks, just to name a few.

These threats differ in many regards, such as frequency, complexity, and potential system damage, which limits MCS technologies in accomplishing their sensing task(s) [69]. Therefore, the modelling of an overhauling solution that covers the widest variety of threats remains an open issue in the realm of MCS systems [41]. However, breaking down such challenges into smaller components and testing their resilience again attack measures should become an integral process of MCS systems. This provides higher resemblances to the threats which may exist outside of trusted environments [85]. Such security measures need to be vetted under realistic and valid attack models, which is the focus of the subsequent sections.

This chapter is organized as follows: section 3.1 briefly overviews related work in identification and adversarial machine learning in MCS. Section 3.3 presents the details and in-depth formulations which compose SINAM. Section 3.4 presents and discusses the numerical results regarding injections rates and discusses behavioural analysis.

3.1 Machine Learning-Based Attack Models

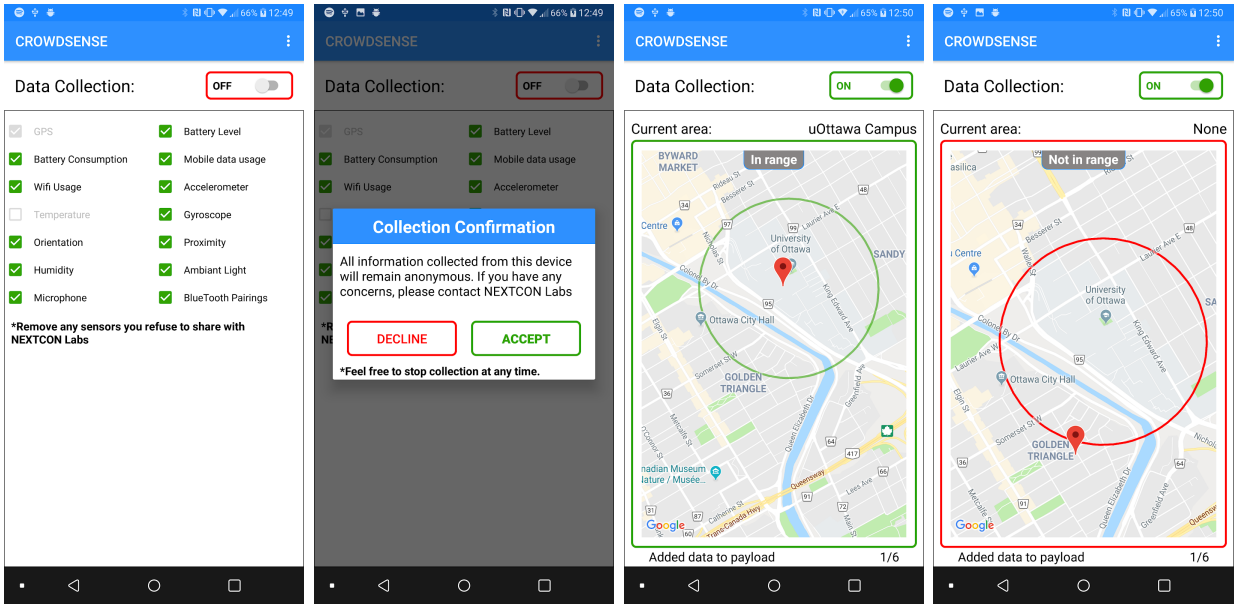
Many adversarial modelling techniques aim to use artificial intelligence as a means of misleading targeted systems astray [73]. When such systems are of the MCS type, robust security measures with the ability to prevent breaches, protect user privacy and system integrity become paramount. To verify security solutions, various machine learning techniques are being introduced to model different types of sensing attacks. One such example may be seen in [115], where Stackelberg game theory is used alongside deep reinforcement learning to detect injection attacks. In another study, the authors [121] also investigated the feasibility of AI-based detection of clogging attacks in MCS environments. The protection of a user's location and data privacy have also been achieved in a real-world experiment [67], helping reduce the gap towards industrial implementations. Based on the above observations, this study builds on similar motivation as the social network authentication studies performed on real users in [7, 97], but focuses on modelling realistic attacks for real-time "big sensed data" acquisition campaigns on a physical crowdsensing platform.

There also exist many other challenges in MCS, such as improving user participation and defining effective incentives [120]. Maximum utility or profit is often the main objective for both the MCS platforms and participants [66]. Indeed, by exploiting auction and game theories, researchers have developed successful incentive-focused mechanisms for MCS users [117]. In such studies, incentives aim at a maximum utility and must be coupled with security, trustworthiness and robustness solutions to be considered suitable defences against the intrusive behaviour of malicious users. For instance, the ability to assign multiple sensing tasks to individuals could allow parallel sensing tasks to be undertaken by the same malicious user, thus minimizing the required threshold for triggering rewards [42]. Furthermore, the efficiency of built-in sensors introduces vulnerabilities, especially if rewarding is attributed based on task completion. Continuous authentication is a strong tool to cope with such

vulnerabilities [37]. Continuously authenticating participants could protect location and data privacy during sensing tasks [67] while also respecting their anonymity [104]. Ensuring the authenticity of participants in an MCS system also helps in preventing Sybil attacks [52], and in identifying Denial-of-Service (DoS) attacks by leveraging game theory methods [98].

3.2 Sensing Campaign and data set

To evaluate the effectiveness of SINAM, a sensing campaign was conducted on the main campus of the University of Ottawa with 9 volunteering participants. Data extraction was performed on an Android application (Android 8, API 26) which was developed in-house for research purposes, called CROWDSENSE; visuals of the application can be found in figure 3.1. This mobile app extracted 28 different built-in sensors (including multidimensional values) at a set interval of 20 seconds, over three weeks during the month of March 2019. The interval is easily adaptable, but pulling sensory data at a higher frequency would have caused a higher battery strain on our participants' devices, and so the compromise was made at 20 seconds. Furthermore, the application only collected data in a predefined geographical area; within a 500m radius of the University of Ottawa campus. Since CROWDSENSE provided participants with the ability to disable any sensor collecting based on their personal preferences (the least collected sensor was the microphone sensor, which only collected sound pitch), some features became quite sparse. Such features, alongside uncommon sensors that caused similar data sparsity, were removed for fair behaviour representations (more on behaviour modelling in section 3.3.2).



The remaining features were reduced using a well-known dimension-reduction method called principal component analysis (PCA). This unsupervised statistical method is applied

Figure 3.1: Main screens of CROWDSENSE - Android application

to reduce the dimensionality of large datasets while preserving as much of their underlying variance as possible (additional details in appendix A). As our application extracted many features (see the first screenshot in fig.3.1), it becomes paramount to select a minimal amount of components without sacrifice on covered variance. Without applying PCA, our methodology would be highly prone to over-fitting and perform poorly on newly introduced data. After removing features whilst keeping as much data variance as possible (92%), 15 features were kept for behavioural analysis, as explained in appendix A. Some obvious features remained, but others were quite surprising. Two notable components were those regarding the orientation azimuth and proximity sensor. Theses contained, on average, the highest contribution to user behaviour profiling across all users. Our hypothesizes reasoning for such a result goes as follows: we tend to place mobile devices on specific flat surfaces — such as an office desk, classroom table, or cafeteria — in specific angles and orientations (always face-up or face-down, with the screen orientated in parallel with our field of view). Theses surfaces also tend to contain a specific azimuth (angle relative to the northern compass), despite their slight variations. Combining these minor — but seemingly significant — differences with proximity data (having the mobile screen close to a surface) is though to somewhat represent the phone placement habits of individuals. As interesting as it may sound, this is only a theory, and further research and validation are required to confirm such speculations. Despite containing around 70k data entries, it's important to mention that our dataset is not fully representative of the varying contexts which occur on a university campus, and so there is a potential issue of the data being biased.

The 9 participating users were split into two groups, based on their levels of participation during the study:

- **Victims = Users 1-2-3-7-8**. These users were the most active during the campaign and had the most complete behaviour profiles. They produced the highest volumes of data and the most variations in sensory data.
- **Accomplices = Users 4-5-6-9**. These users have had the most sporadic participation patterns. These could have been caused by faulty built-in sensors, older phone models, or other undiscovered reasons. These behaviour patterns match those of malicious users and are used in the creation of malicious injections.

The behaviour profile of each victim was created using an unsupervised clustering algorithm called Density-Based Spatial Clustering of Applications with Noise (DBSCAN), explained in further detail in appendix B. Applied to our study, DBSCAN clusters the different sensor values into groups which can be interpreted as behaviour profiles. In other words, the number of clusters of each participant can be seen as the varying contexts in which individuals find themselves on campus. As to avoid over-fitting and have a sufficient

validation set, we used a 75/25 data split. We favoured this ratio to emphasize our validation set without seeing any change in results. Using 75% of the data to create the behaviour clusters (and obtaining optimal parameters through hyper-parameter tuning of DBSCAN), our validation set represents the algorithm's ability in identifying if new incoming data is associated with a specific user's set of behavioural clusters. In other words, we calculate the training accuracy by using the remaining 25% to measure the correctly-clustered points over all the points in the validation set (the higher the anomalies, the lower the accuracy). The results, as seen in 3.1 is that DBSCAN can accurately associate new data to each corresponding victim. These models (one for each victim, also known as behaviour profiles) are what the SINAM model will attempt to inject upon, causing non-associative points of behaviour (created from accomplices) to be associated with a victim's behaviour cluster. Malicious data would, therefore, represent the acceptance into a behavioural cluster of data associated with another participating individual, i.e. the infiltration of accomplice data in our experiment.

Table 3.1: Behavioral clustering with DBSCAN

Users	**Total Data Points**	**Number of clusters**	**Accuracy**
User 1	8551	4	0.9979
User 2	12842	5	0.9347
User 3	22680	2	0.9985
User 7	12009	10	0.9851
User 8	6386	5	0.9432

3.3 SINAM Attack Model

Standard injection attacks create and insert data injections as continuous streams of data are collected from a particular platform. This exact concept is how SINAM is designed to operate. Using data captures during the live sensing campaign explained in the previous section, SINAM tracks behavioural similarities between its participants and uses a novel constant called the *tampering potential* to make intelligent injection decisions on targeted victims.

This threat model involves deceiving a receiving system in accepting data that is not originating from its identified source. This can lead to many unwanted scenarios, such as replacing a victim's identity, tampering with the perceived reliability of a victim's on-board sensors, or providing false information to the system regarding surrounding environmental conditions. Certain fundamental assumptions are required to maintain the effectiveness of the SINAM model:

- The campaign must contain *victims*, which are MCS users being monitored by SINAM for potential injections.
- The campaign must also contain *accomplices*, which are malicious participants within the same MCS campaign who collaborate with SINAM, allowing for the exchange and analysis of data related to sensing tasks.
- Similar contextual settings are required, as only shared contexts between victims and accomplices allow for intelligent injections to occur. The concept of injecting using alternative contexts is could be subject to future work.

If victims and accomplices were to co-exist in distinct environments, injecting malicious data based on the behaviour of accomplices may not be feasible. As data is processed in batches, SINAM uses the behaviour of accomplices within the MCS campaign to create injections, which are inserted into the victim's data streams. This is accomplished by recognizing behaviour similarities between accomplices and selected victims and identifying which features (i.e. sensors) are vulnerable for injection. Feature injections are produced using the proposed tampering potential, which defines the ratio at which victim and accomplice behaviours are used for injection (more details regarding this ratio in section 3.3.3).

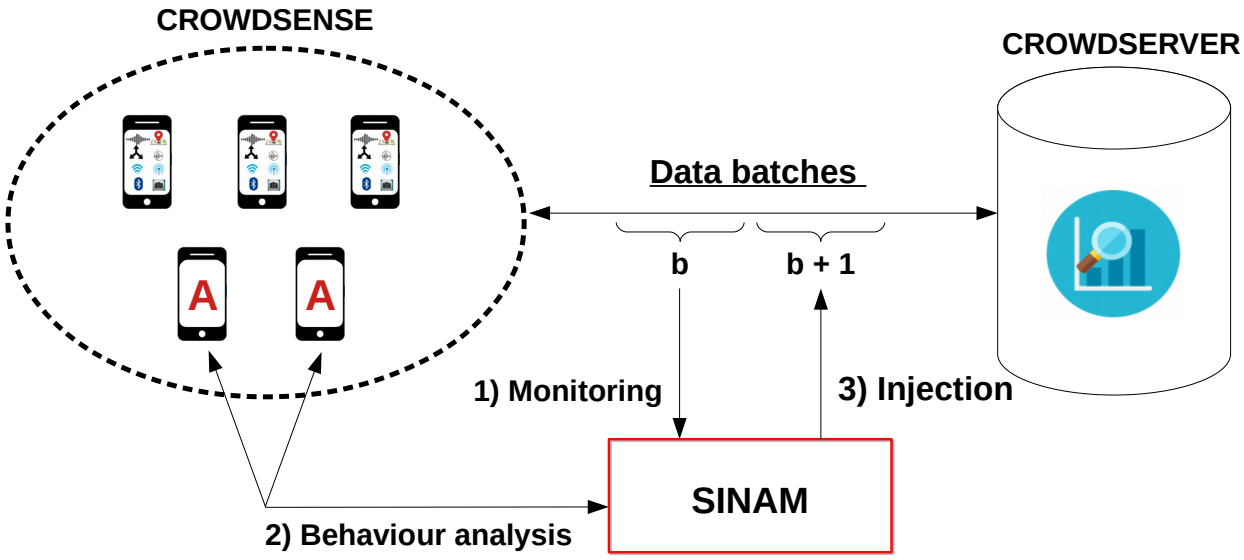


Figure 3.2: Multi-step injection model: 1) Obtaining data for attack, 2) Conducting analysis and computation of injection, 3) Injection into next batch of victim's data

During the monitoring of streams of incoming data, our injection model splits such incoming streams into batches of data, synonymous with online machine learning methods. Once data is combined into batches, injections are created for each batch in the following order:

1. **Monitoring phase**: Monitor a targeted victim's data until N data points are obtained. This creates the victim's batch of data, which is our baseline data structure, also known as b in fig. 3.2. The subsequent batch of data is defined as $b+1$, naturally.

2. **Behavior analysis phase**: Batches of data are also collected from the accomplices in accordance with the victims'. This synchronization is relative to the corresponding time frames, as explained in the next section. Specifically, each index of a victim's batch is associated with the closest index of an accomplice; this is accomplished through a distance-based selection process. Once indexes are selected, each accomplice's data (malicious behaviour) is combined with that of the victim in an influence-aware process, ensuring injections only occur in susceptible features. This ratio of injections created by SINAM is what we define as the *tampering potential*, as explained in further details in subsection 3.3.2.

3. **Injection phase**: Finally, behaviour injections of all accomplices are combined with their corresponding weights — defined using our distance function in the previous phase — to create a final injection into the succeeding batch of data of the selected victim.

These above steps are repeated each time N data points are acquired for the sensing campaign. A minimalist illustration of the overall process is presented in Figure 3.2. Therefore, if injections are to be produced on multiple victims, this would require multiple replicas of SINAM, each injecting on a single concurrent victim. An interesting notion is the collaboration of multiple instances of SINAM running in parallel (and potentially collaborating), which could potentially be the topic of future work, as discussed in chapter 6.3.1.

The subsequent sections use specific notations which should be meticulously investigated for proper comprehension. Theses formulations are presented in Table 3.2, and a corresponding visual representation of the process is also included in Figure 3.3.

Table 3.2: Notation for SINAM Model

Notation	Description
N	Amount of data points required to produce a single batch of data
n	A single index of a specific batch
F	Entire set of features within each data point
f	A single feature within F
A	The set of accomplices collaborating with SINAM
k	A single accomplice within A
δ	Binary value representing a fluctuation in data
$\mathcal{B}_v$	A batch of data belonging to the victim
$\mathcal{B}_{a_k}$	A batch of data belonging to kth accomplice
$\mathcal{B}^n_{a_k}$	The index n (or data point n) within B_{a_k}
$\mathcal{B}^n_v$	The index n (or data point n) within B_v
v^n_f	The feature f at index n within B_v
$a^n_{k,f}$	The feature f at index n within B_{a_k}
α	The tampering potential constant
i_k	The closest behavior to B^i_v that exists in $B^i_{a_k}$
W_k	Weight given to accomplice k
φ_k	Tampered injections of accomplice k
Φ_k	Injections of accomplice k filtered through susceptibility mask
I_f	Combined injections on feature f from all accomplices
$\Psi(B^{N+1}_v)$	Final injection created by combining all I_f's

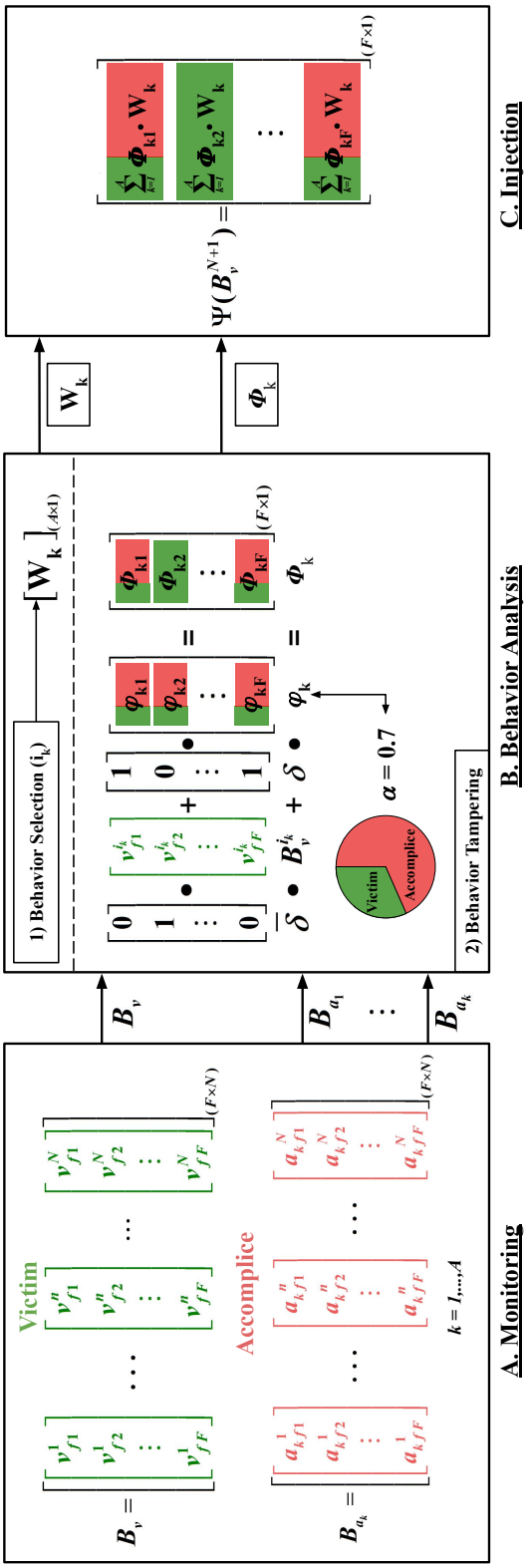


Figure 3.3: Influence of tampering potential on injected behavior

3.3.1 Victim Monitoring

As mentioned, the initial phase within SINAM is victim monitoring, which is simply a period of data collection and data formatting. First, the representation of each data point is described below. A single data point of a victim is represented as $\mathcal{B}_v^n$, containing it's corresponding indexes n and features f. For example, the first feature of the first data point would be $v_f^1 1$ and would represent a single value of a sensor (say the temperature sensor value of 21 degrees Celsius). Therefore, a single data point of n contains F features, as seen in eq. 3.1.

$$B_v^n = \begin{bmatrix} v_{f1}^n \\ v_{f2}^n \\ \dots \\ v_{fF}^n \end{bmatrix} \tag{3.1}$$

However, we do not create injections using only a single point of data but require an entire batch of data. The size of this required amount of data points is referred to as N, which only once N data points are received by the MCS platform can the victim's batch of data be complete. This is represented in eq. 3.2.

$$B_v = \begin{bmatrix} B_v^1 & \dots & B_v^n & \dots & B_v^N \end{bmatrix}_{(F \times N)} \tag{3.2}$$

Another more detailed view of eq. 3.2 can be seen in eq. 3.3, which demonstrates how a batch of data contains indeed N data points of F features each. The influence on the selected size of N is highly correlated with the success of our SINAM model; the higher the volume of data in each batch, the higher the injection rate becomes. This is also associated with a higher computational cost, as seen in SINAM's algo.3.4. As for online-learning methodologies, larger volumes of data-batches tend to require more expensive error handling, and smaller volumes result in unnecessary overhead. A comfortable medium was an N of size 10, given the size of our entire dataset (around 65k data points) and the length of our data-collection campaign (3 weeks).

$$B_v = \begin{bmatrix} \begin{bmatrix} v_{f1}^1 \\ v_{f2}^1 \\ \vdots \\ v_{fF}^1 \end{bmatrix} & \dots & \begin{bmatrix} v_{f1}^n \\ v_{f2}^n \\ \vdots \\ v_{fF}^n \end{bmatrix} & \dots & \begin{bmatrix} v_{f1}^N \\ v_{f2}^N \\ \vdots \\ v_{fF}^N \end{bmatrix} \end{bmatrix} \tag{3.3}$$

The formulation of an accomplice's data batch is almost identical; the only difference, which can be seen in eq. 3.4 is the substitution of the variable of v to a_k, which represents the *kth* accomplice's batch. Reminder: SINAM targets a single victim per created injection, but uses the participation of multiple accomplices, hence the nomenclature.

$$\mathcal{B}_{a_k} = \begin{bmatrix} \begin{bmatrix} a^1_{k,f1} \\ a^1_{k,f2} \\ \vdots \\ a^1_{k,fF} \end{bmatrix} & \dots & \begin{bmatrix} a^n_{k,f1} \\ a^n_{k,f2} \\ \vdots \\ a^n_{k,fF} \end{bmatrix} & \dots & \begin{bmatrix} a^N_{k,f1} \\ a^N_{k,f2} \\ \vdots \\ a^N_{k,fF} \end{bmatrix} \end{bmatrix} \tag{3.4}$$

3.3.2 Behavior Analysis

User behaviour is defined as a continuous stream of data points; as victims move from one context to another, their behavioural fluctuations are captured by their built-in sensors. For instance, a victim is on their laptop in their office (context 1) and then decides to move to the cafeteria to get a coffee (context 2). The transition between the two contexts is captured through environmental sensors; latitude and longitude values are updated, noise levels slowly increase as ambient noise rises, etc. Therefore, the behaviour of the victim is capture during the change of context; getting up and walking from context 1 to context 2 while potentially greeting fellow colleagues are examples of captured behaviours (noise levels, gait, proximity changes, etc.).

Behaviour analysis is broken down into two parts, which is performed after a full batch of victim data is received. First, a distance function is used to find the accomplice batch with the closest corresponding behaviour; this is done by comparing the Euclidean distance the victims batch and all the accomplices, and the closest behavioural pairs (indexed using i_k) are identified. Second, these pairs are combined to create corresponding weights for each accomplice-victim pair, rendering normalized weights at each index, and for each accomplice.

Accomplice batch selection is therefore based on behaviour resemblance to a victim's behaviour. This is calculated by comparing every batch, index-by-index (i by i), or behaviour-by-behaviour. Each index of the victims batch, therefore, has a closest corresponding accomplice behaviour, denoted as i_k, where k is associated with a corresponding accomplice. Once all i_ks are calculated, where closeness is associated with inverted Euclidean distance, as shown in eq. 3.5, only the $|A|$ closest (or smallest distances) indexes are kept. This number corresponds to the number of accomplices in the sensing campaign, although further investigation is required in finding the optimal number of i_ks that should be kept at this step. In other words, the smaller the distance between an accomplice and a victim's data batch, the higher the overall calculated weight for that accomplice.

$$i_k = \arg\min_i(||\mathcal{B}^i_v - \mathcal{B}^i_{a_k}||);\ i = 1, .., N,\ k = 1, ..., A \tag{3.5}$$

Once the $|A|$ closest i_ks have been identified, the second step is the creation of weight associated with each accomplice of each index (i_k). More specifically, a normalized weight called W_k is given to each accomplice k, as seen in eq. 3.6. Note that is it possible to have

the same accomplice associated with multiple indexes of i_k. This is often the case when an accomplice can "mimic" a victim's behaviour, thus being associated with most i_k indexes. Such occurrences did however appear quite infrequently.

$$W_k = \frac{(||\mathcal{B}_v^{i_k} - \mathcal{B}_{a_k}^{i_k}||)^{-1}}{\sum_{j=1}^{A}(||\mathcal{B}_v^{i_j} - \mathcal{B}_{a_j}^{i_j}||)^{-1}} \quad \forall k \in A \tag{3.6}$$

3.3.3 Behavior Injections

Behavioural injections produced by SINAM are defined as follows: malicious data points which are injected into a victim's behaviour profile, and subsequently incorrectly clustered into a specific behavioural cluster of that victim (see table 3.1). In other words, SINAM can produce behaviours that are acceptable by that user but maliciously derived from the set of accomplices.

When an injection is constructed, it is not created with purely malicious sensor values. There is a specific ratio of victim-to-accomplice which is used, and this ratio is called the *tampering potential* (represented by α in all notation below). This tampering potential represents the degree at which sensory data will be hindered by the algorithm, meaning the lower the tampering potential, the less each accomplice's behaviour is used in each injection. Evidently, higher injection ratios cause higher maliciousness behavioural injections but are subsequently easier to identified as anomalies by DBSCAN. On the contrary, if the ratio is too low, this would render less malicious injections, but such injections would resemble too closely the behaviour of the victim, thus causing only minor shifts in the victim's behavioural profile.

The feature tampering process is illustrated in Figure3.3, demonstrating the influence of the tampering potential(α) on the final injection.

A weighted-sum is used to create an injection on each sensor value within a particular batch, using both the victim's data ($B_v^{i_k}$) and each accomplice's data ($B_{a_k}^{i_k}$), as formulated in Eq.(3.7).

$$\varphi_k = (1-\alpha) \cdot B_{a_k}^{i_k} + \alpha \cdot B_v^{i_k}, \quad k = 1, 2, ..., A \tag{3.7}$$

There is, however, another caveat in when injections are created. Simply always creating injections on every sensor value is too easily detectable by intelligent algorithms [122], and so a more sophisticated approach was required. Our strategy is to also monitor the victim's batch and detect temporal deviations. This way, only when such deviations occur would behaviour injections be entered. For example, if a victim's location remains unchanged

within an entire batch, any behaviour injection which includes tampering on such a stable feature would be trivial to detect, and easily fall outside the victim's behavioural profile. Injections still occur on each batch of victim data, but only on features that have temporal deviations, as represented in eq. 3.8. When no deviations occur (i.e $\delta = 0$), the original sensor values are maintained for those feature(s). If no deviations occur in any sensor values for every sensor value of the batch, then no injection is created and that particular data batch ignored by SINAM.

$$\delta = \begin{bmatrix} \delta_1 \\ \delta_2 \\ \vdots \\ \delta_F \end{bmatrix} \tag{3.8}$$

Returning to our malicious injections, each accomplice creates (or ignores) individual sensor injections based on the tampering potential for each sensor value in a batch. These values are combined to make accomplice-injection-batches, which are denoted with Φ_k. Again, some values may not have injections, depending on the temporal deviations mask (δ). The specifics of a accomplice-injection-batch is represented in eq.3.9, seen below.

$$\Phi_k = \begin{bmatrix} B_{v1}^{i_k} \cdot \overline{\delta_1} + \varphi_{k1} \cdot \delta_1 \\ \vdots \\ B_{vF}^{i_k} \cdot \overline{\delta_F} + \varphi_{kF} \cdot \delta_F \end{bmatrix}_{(F \times 1)} \quad , \quad k = 1, 2 \ldots, A \tag{3.9}$$

Once all values of Φ are computed, they are combined with the previously calculated accomplice weights - back in eq. 3.6. Theses weighted sums produce every malicious sensory data value, as seen in eq. 3.10. The sensor values are then aggregated together to create the final malicious injection point. This injection is then inserted in the succeeding victim's data batch (N+1) and the process is repeated, attacking another (or potentially the same) victim. When creating an injection on a new batch of victim data containing an injection previously created, that previous injection is overlooked. This way, SINAM is only investigating true victim behaviour and does not take over a victim's behavioural cluster, although such a notion is interesting and may have the potential for subsequent work.

$$I_f = \sum_{k=1}^{A} \Phi_{k_f} \cdot W_k \quad , \forall f \in F \tag{3.10}$$

$$\Psi(B_v^{N+1}) = \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_F \end{bmatrix}_{(F \times 1)} \tag{3.11}$$

Algorithm 3.4 encompasses all the above steps into a single algorithmic process, constructed from the code-base. The SINAM runs on a single victim batch at a time, but multiple models could inject into the same victim, in theory. This multi-attack phenomenon is left for future work.

3.4 Evaluation of Injection Methodologies

Our proposed method, SINAM, conducts the three above steps (monitoring, behavioural analysis, and injection), but we also consider a randomized attacker model as a baseline model. This baseline model skips the accomplice selection phase, instead randomly selecting accomplice behaviours for injections. Thus, SINAM selects based on behaviour similarities and the Random Attack Model makes a random accomplice selection. Both models share all other properties (tampering potential, batch size) to keep comparisons relevant.

The undetected injection rates are compared between SINAM and the Random Attack Model to demonstrate the effectiveness of SINAM. Both models create injections against the same victim behaviour profiles, across varying values of batch sizes and tampering potential ratios.

DBSCAN was selected amongst many unsupervised clustering methods since it performs especially well on spatial clustering [5]. This is the equivalency of distinctive behavioural patterns in our context. DBSCAN, as explained in appendix B, led to behaviour patterns that were not identical across all victims. However, individual behaviour identification for individual users remains high, as seen in Table 3.1.

The effectiveness of both models is represented with the undetected injection rate, which denotes the rate at which injections are classified as the victim's behaviour. The impact of different *tampering potentials* and batch sizes are also presented. A batch size represents the quantity of data (behaviour patterns) that are analyzed before injection decisions are made by attackers. It is worth noting that the baseline Random Attack Model, as well, inherently leverages intelligence to some extent, as it performs the feature tampering step seen in Section 3.3.3. Different values of *tampering potential* are presented to demonstrate its impact on undetected injection rates.

As DBSCAN formed distinct behaviour clusters for each victim, an event of data injection is assumed undetected if the injected data point is clustered in one of the victim's behavioural clusters. The success metric is defined as the undetected injection rate.

Our experiment was conducted using batch sizes of 50, 100 and 200, each with variations in *tampering potentials* of 0.5 and 1. Both the SINAM and Random Attack Model are compared in Table 3.3. As depicted in our methodological overview, if a victim's feature does not change within a monitored batch, both attack models refrained from injecting accomplice behaviours into that particular feature, as explained in Figure 3.3.

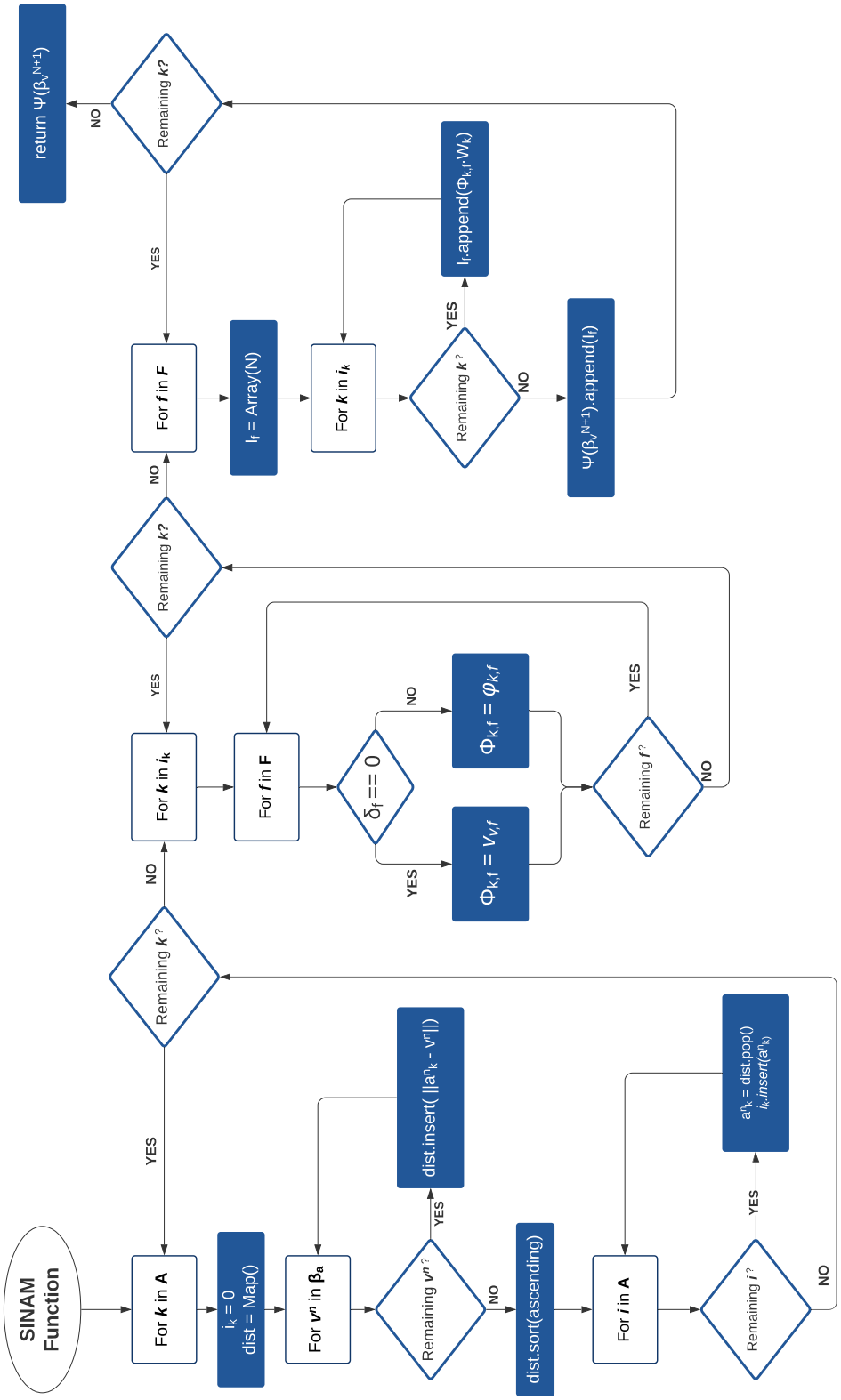


Figure 3.4: SINAM Behavior injection algorithm

Table 3.3: Undetection rates across different configurations

	Attackers	BatchSize=50		BatchSize=100		BatchSize=200	
		α:0.5	α:1.0	α:0.5	α:1.0	α:0.5	α:1.0
User 1	SINAM	1.0000	0.7143	1.0000	0.5273	1.0000	0.3500
	Random	0.7262	0.0893	0.6848	0.1273	0.7000	0.1500
User 2	SINAM	0.8537	0.7053	0.8043	0.7404	0.7619	0.7238
	Random	0.4675	0.0711	0.3617	0.0936	0.4667	0.1048
User 3	SINAM	1.0000	0.2779	1.0000	0.0000	0.9954	0.0000
	Random	0.3114	0.0011	0.3386	0.0045	0.4537	0.0000
User 7	SINAM	0.9850	0.9079	0.9780	0.8943	0.9340	0.7830
	Random	0.8158	0.4304	0.7137	0.3833	0.7547	0.4717
User 8	SINAM	0.9087	0.9627	0.8840	0.8929	0.7959	0.7347
	Random	0.7635	0.5062	0.7054	0.4732	0.7143	0.3469

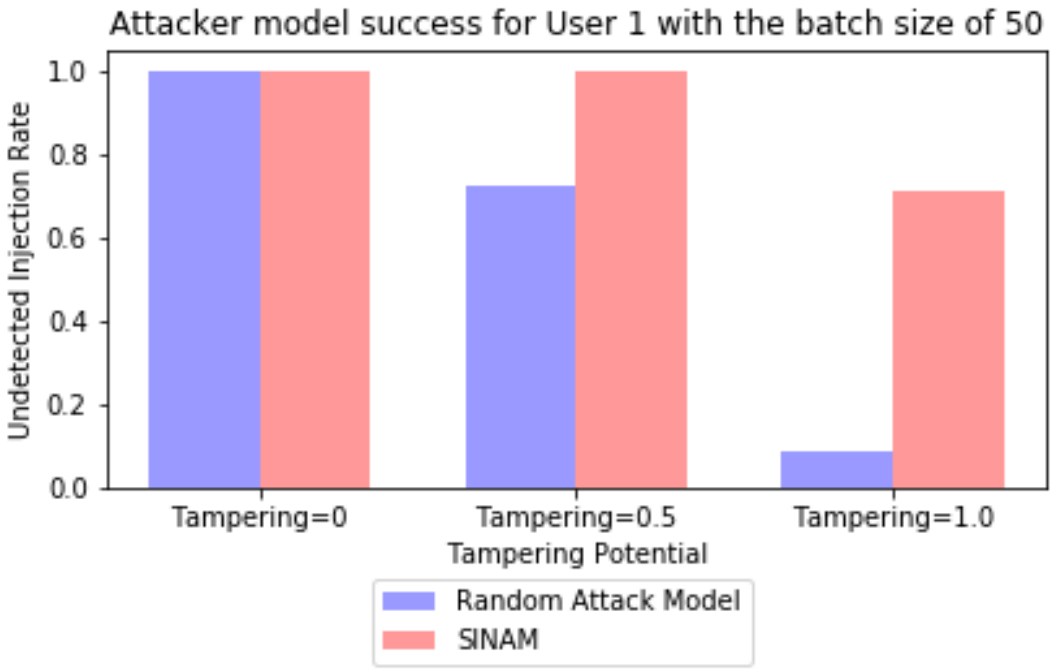


Figure 3.5: Impact of the tampering potential on Victim 1

This approach was taken as victims sometimes demonstrated little fluctuations in data. For instance, stationary users constantly send the same latitude and longitude values until

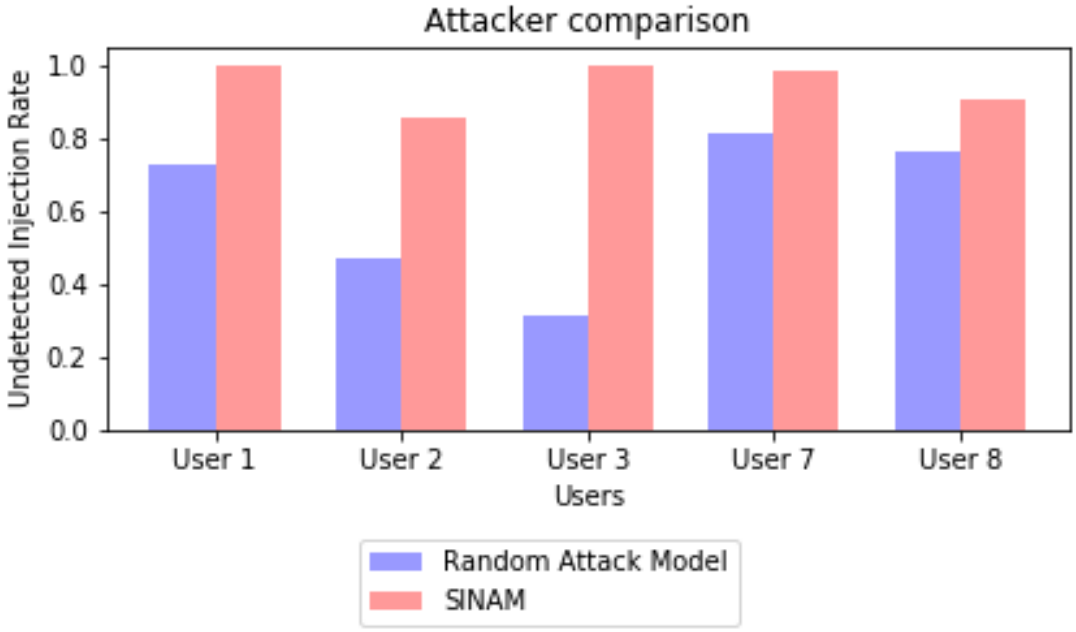


Figure 3.6: Injections rates comparison between Randomized Attacker and SINAM with a batch size of 50 and tampering potential of 0.5

they move around again. Thus producing a single injection in such stable features would result in immediate injection detection. Alternatively, when there are fluctuations within a feature of a victim's batch, the *tampering potential* kicks in. A *tampering potential* of 1 represents full accomplice injections and a *tampering potential* of 0.5 causes both attack models to use half of the victim's original features if deviations occur.

The results support the effectiveness of SINAM compared to the Random Attack Model. This gap in performance only increases as the *tampering potential* increases, causing a higher ratio of accomplice behaviour to be in the injections. The Random Attack Model is penalized as the randomness in behaviour selection from the Random Attack Model causes more frequent anomaly detection by DBSCAN, since the closeness (compatibility) between accomplice and victim behaviour is not considered systematically. SINAM's advantage in the accomplice selection phase (subsection 3.3.2) leads injections to have always had a higher undetected injection rate, regardless of batch size and *tampering potential*. The only exceptions to this statement are results of user 3, which based on the number of clusters in table 3.1, was a user with only two distinct behavioural clusters, meaning their behaviour fits only two distinct profiles, causing the DBSCAN algorithm to efficiently detect anomalies.

Another remark is made when batch sizes decrease: the undetected rates increase. These batch sizes cause a high frequency of similar behavioural data, causing many victim-

accomplice comparisons to be made. With more comparisons, there is a higher potential to find similar behaviours in the resulting set of behavioural patterns (set S). S then becomes less diluted, containing more sporadic behaviour patterns, which are in turn more vulnerable to injections. Having increasingly small batch sizes would render behaviour patterns into behavioural noise, as behaviour is defined as a series of actions to events, but requires sufficient actions for any fruitful behavioural extraction to be made.

Fig. 3.5 demonstrates different *tampering potential* values and their influence on injection rates for both the Random Attack Model(blue bars) and SINAM (red bars). As shown, the tampering potential dramatically affects the Random Attack Model, as it is the only property keeping that model from being purely random. A tampering potential of 0 signifies injections are purely based on the previously repeated victim behaviours, and tampering potential of 1 indicates full behaviour injections produced by accomplices. Hence, increasing the tampering potential decreases the undetected injection rates for the Random Attacker Model more drastically compared to SINAM, which receives a drop at 0.5. With a tampering potential of 1, the undetected injection rate drops to near zero for the Random Attack Model, but the SINAM model remains around 70%. This observation is only on a single victim (user1) with a batch size of 50.

Another interesting remark is that some victims were much easier to successfully injection upon. This could be correlated to how frequently they switched contexts. User 3, for instance, has a low amount of behavioural clusters in their behavioural profile, and so incoming data could be categorized as more homogeneous. In that sense, it becomes much easier for SINAM to create successful injections since the complexity threshold to fool the DBSCAN model is lower. On the other hand, large amounts of variation in a victim's behavioural profiles are also seen to be more successful. In a similar fashion, SINAM is able to create successful injections in very chaotic and heterogeneous data batches. The higher the number of clusters, the more active the victim was in their context switching. It seems to be most difficult somewhere in between both extremes, as seen in Users 2 and 8.

Fig.3.6 presents an overall comparison of the Random Attack Model and SINAM with a batch size of 50 and a *tampering potential* of 0.5. As seen in the Table 3.3, behavioral patterns of a certain users (user 3 for example) had more clear cutoffs. As expected, the more behavioural clusters a victim had (see Table 3.1), the better both models performed. However, for most users and the majority of the population, SINAM was able to inject data while remaining 90% undetected when using a *tampering potential* of *0.5*.

This study has provided us with the conclusion that malicious behavioural modelling can successfully remain undetected when injected in an intelligent fashion. SINAM provides an example of this, showing how random behavioural approaches are easily detectable and therefore preventable, but more intelligent and proactive methods, such as SINAM, would require MCS campaigns to be increasingly vigilant in defending and monitoring the data that is received from its participants.

We believe that our SINAM can be utilized in further advancing the state of the art of MCS defensive methodologies. Future research can use our algorithm when data is collecting

in a real-time data collection scenario, not only limited to MCS campaigns.

Through the collection of real-time data during a 3 week period, we have successfully demonstrated how a smart injection attacker model can tamper with data being sent to a central platform. As the areas of IoT, MCS and machine learning see continuous improvements and application in our everyday lives, the robustness and security of our infrastructure must continuously be stress-tested, as to maintain sophisticated and applicable digital solutions to society.

Chapter 4

Context-Aware Defensive Machine Learning

As internet-powered software applications are ever increasing in our society, it has been a continuous struggle to adequately protect all our inputted information [57]. Even with the inclusion of SSL encryption, there exist methods and strategies for infiltrating cyber systems [33]. These types of cyberattacks are of an increasing threat as our digital infrastructures (including access control protocols) are migrating to cyber plane [118], specifically in cloud systems. There are many advantages to including our software infrastructures in the cloud, as it solves scalability and accessibility issues [118]. Therefore, when attempting to design an approach to increase the security of these systems, knowing the extent of cyber attacks becomes essential. Not only is it essential, but it also exposes the vulnerabilities of each different cyber context [64]. This does require a vast amount of knowledge in cyber systems to properly adhere to modern cybersecurity standards [3].

Many context-based solutions exist within the cyber domain. For instance, the ability to detect the biological context of individuals [59, 105] is made possible using an array of sensors. Knowing the severity and context in real-time allows for first responders to make more efficient health interventions. In the same light, mobile device ubiquity has also allowed a deeper analysis of spontaneous events [88], which was more extensively covered in chapter 2.

If one combined all modern cybersecurity solutions, they might be able to cover a vast majority (if not all) of the vulnerabilities found within the cyber domain; the challenge lies in the attempt to cover as many areas as possible using a single — or a few — solution(s), without compromising security or causing unwanted intrusion to any person conducting actions on these types of systems. This chapter attempt's to take a step in that direction; by attempting to provide high levels of security, without impeding heavily on the convenience of participants. This trade-off is achieved through the means of smart authentication technique, which using the power of artificial intelligence, triggers authentication prompts depending on the cyber context of the user.

4.1 Context-aware Access Control

Managing large groups of individuals all accessing the same cyber system requires sophisticated software; when large volumes of events happen in a system's cyberspace, even the slightest algorithmic error can cause catastrophic repercussions. The deployment of intelligent access control algorithms is thus required and has demonstrated great success in dealing with large volumes of cyber events [123, 16]. Such strategies usually define a subset of roles, each having specific rights to accessing resources of varying levels of importance. This common pattern is found in large enterprises [11] and is referred to as Role-based Access Control (RBAC) systems, dating back to 1996 [99]. These systems leverage a combination of roles (Administrators, Managers, Viewers, Administrative roles, etc.), permissions (Read, Write, Delete, Update, Move, Administer, etc.) and authorization methods — PINs, passwords, and more modern techniques like authenticators and biometrics — in improving their cybersecurity. The multiplicity of these attributes has been presented in later years, allowing users to have multiple roles, with a multitude of permissions and use multiple authentication methods. There have been many advances that now allow for a near-complete coverage of potential use-cases [124], even expanding to blockchain technologies [27].

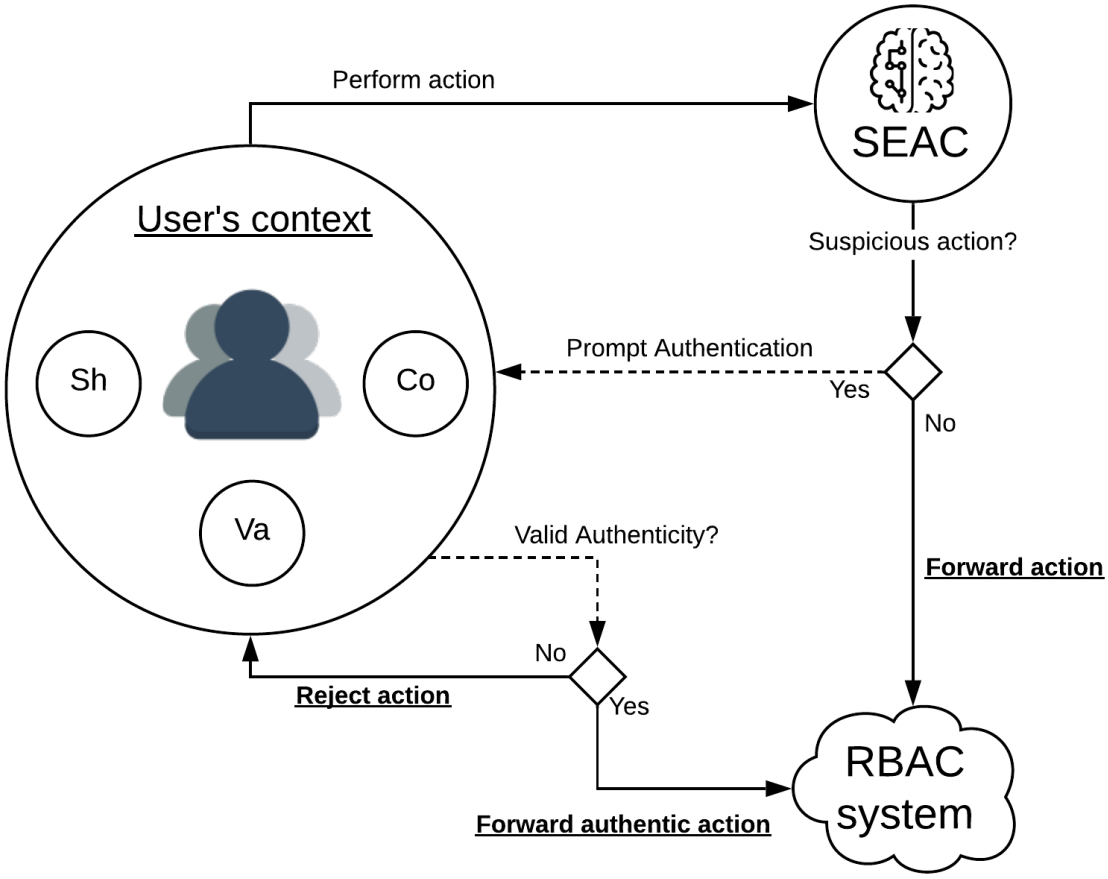


Figure 4.1: SEAC technique inserted into a typical system

Such a combination of strategies is precisely the method that we exert in this chapter. Specifically, our hypothesis in question is as follows: how can we leverage cyber-context analysis in improving RBAC security models? These improvements focus on two areas: improving cybersecurity and improving user convenience. To this end, we have derived cyber metrics that extract contextual properties of users within their cyber domain (referred to as cyber-feature engineering in the subsequent section). These metrics are combined with our own artificial intelligence ensemble method which uses weighted-voting in creating predictions on which actions in the varying cyber-contexts should trigger authentication prompts within for the RBAC system. Such an extension aims to assist RBAC systems in their authentication decisions, by providing valuable insights from our Smart Enterprise Access Control (SEAC) method, as defined in fig. 4.1. This study does not intend to replace RBAC methodologies, but rather by both filtering actions and providing additional levels of authenticity when suspicious actions take place in their system. The access-control itself is the combination of rules and regulations which provide users with access to their digital assets, whereas authentication is a measure of confidence on if a user is whom they claim to be.

Our experimental settings focused on extracting the contextual properties of individuals at the time of any performed action, to improve their measure of authenticity. Our dataset was obtained from simulations ran based on a genuine behaviour model — sourced from real-world employee data — and is composed of users (**who**) which perform actions (**what**) at a specific time (**when**) in a specific environment (**where**). These 4W's are the basis of understanding the circumstantial details of an occurring event. The ensemble of the contextual properties composes both the 7 original features + our 3 extracted metrics (found in table 4.1). The derived metrics are as follows: i) an estimate on how much **value a certain digital asset contains**, ii) an attempt to quantify how much a **document is shared**, iii) the quantification on the **cooperation** between all users in our system.

Features	**Description**
loc	Where the action was taken
user	Who performed the action
time	When the action was performed
action	What was performed
document	The affected asset
op1	Optional parameter 1
op2	Optional parameter 2
Va	Document Valuation
Sh	Document Shareability
Co	User Cooperation

Table 4.1: Feature definitions

Here are the exact names and a more in-depth overview of the derived metrics created from the cyber-contextual details. Included are examples and some assumptions, and each mathematical formulation is situated in the next section.

- **Document Shareability(Sh):** The concept of having multiple authors ties into the amount a document is shared. This notion states that documents with higher quantities of interactions and therefore revisions (such as annual reports, drafts upon drafts of the same document) are of higher importance to a RBAC model. On the contrary, those with fewer revisions (an email, for instance) would see lower sh arability metrics. Therefore, this ties considerably into the number of contributors; if many people contribute to the same document, then it could potentially be of added value (such as active pull request in access control platforms), but not necessarily. The planning of an office pot-luck would have many iterations and additions, but should not be seen as valuable to the RBAC model, and so additional considerations are thus needed.

- **Document Valuation(Va):** Producing value in a document is quite disputable, but empirical agreements were obtained based on the following rationales: an estimation on the value of a document depends on the frequency and type of interaction that was made on a digital asset. For instance, there would be a larger sum of value in a document if two individuals contribute to the same digital asset rather then one. An emphasis is made here on *contribute*, which be code snippets, paragraphs of text, or other formats (referred to as **write** actions in our study). This type of interaction differs from a user who would instead read (or consume) the information. If anything, the concept of reading such information should be seen as a decrease in value, since the information would no longer be private, and more widely accessible. Evidently, this is not self-sustaining in assumptions, it depends on the context of who is reading the digital asset, and what the content contains. As data is often encrypted, there would be no way of knowing without invading privacy, and so looking at the interactions of users and their roles could be a good indication of who should have access to which document.

- **User Cooperation(Co):** Modelling the cyber interactions between users in an entirely cyber environment contains two important aspects: whether or not they have a shared action to which they both cooperate on, and what responses are performed from one individual to another. Considering the high level of abstraction, let's explain using an example: suppose there are two users in our system: Jane and John. Both of them have a set of documents to which they interact; some they may simply read, others may take many revisions before they are submitted to their superiors. In this case, it is safe to assume that Jane and John do not interact on a cyber level if their sets of documents are mutually exclusive. However, in the converse situation, how John interacts with Jane is not necessarily symmetric; Jane may only read documents that John writes, or perhaps John only opens a document to give Jane access to it. These types of interactions are modelled through what we call a User Cooperation

metric, which leverages weighted cooperations between sets of documents of all users. A higher frequency of predicable interactions would render higher trustworthiness to a RBAC model. Again, this metric is not self-sufficient, but in combination with the other two contextual metrics, the cyber context of individuals becomes less obscured.

4.2 Cyber-Feature Engineering

As previously described in layman's terms, three contextual metrics were derived to extract the cyber contextual properties of individuals in an enterprise setting. This chapter details all the technicalities of each metric. The table 4.2 identifies all the variables in this section. First and foremost, the eq.4.1 represents an array of actions in which specific user u did on a document i within the given timeframe Δ_k.

Table 4.2: Environment Definitions

Variable	Description
D	set of Documents
i, j	instances $\in$ D
U	set of Users
u, v	instance $\in$ U
τ_k	kth instance of timeframe τ
Δ_k	length of each timeframe τ_k
α	weight given to past calculations
A:{R,W,X,S,C}	Actions: **R**ead, **W**rite, Delete(**X**), **S**et and **C**lear permission
$\gamma_{u,i}$	Action matrix of u on i
$\delta_{u,i}$	Interaction weight between u and i
$\omega_{u,i}$	Valued action weight of u performing i
$\Omega_{u,i}$	Average valued action on i
$\lambda_{u,i}$	Highest social action by u on i
Λ_u	Action matrix of u
$\mu_{u,v}$	Weighted cooperation of u with v
$Sh_i(\tau_k)$	Shareability of document within τ_k
$Va_i(\tau_k)$	Valuation of document within τ_k
$Co_u(\tau_k)$	Cooperation of user within τ_k

$$\gamma_{u,i}(\Delta_k) = [\Sigma W, \Sigma S, \Sigma C, \Sigma X, \Sigma R] \tag{4.1}$$

If a user reads a particular document three times (Read action), set his colleague as a collaborator (Set permission), and wrote 4 times into the document(Write action), their array would look as follows:

$$\gamma_{u,i}(\Delta_k) = [5, 1, 0, 0, 3]$$

In this example, the value of $|W|$, $|S|$ and $|R|$ would be 5, 1 and 3, respectively. The γ function is the starting point of each of all three metrics in the subsequent section.

4.2.1 Document Shareability

The first extracted contextual factor is how much a document is shared between users, referred to as document *Shareability*, or $Sh_i(\tau_k)$. The initial step in calculating this metric is defining the weight of specific interactions - $(\delta_{u,i}(\Delta_k))$ between users and documents, as seen in eq.4.2. A document is considered to be shared when it is either written upon or read, where writing is considered twice the "sharing" weight as reading. Only the largest number of the two is kept, as is true for all partial functions in this section.

$$\delta_{u,i}(\Delta_k) = \begin{cases} 2, & \gamma_{u,i}(\Delta_k)|W| > 0 \\ 1, & \gamma_{u,i}(\Delta_k)|R| > 0 \\ 0, & else \end{cases} \tag{4.2}$$

The second step in engineering this feature is normalizing the values of $\delta_{u,i}(\Delta_k)$. This results in what is referred to as instant Shareability(Sh_i), which quantifies how much a document was shared by each user relative to every other document. This thus provides added weight to documents that have received high amounts of interactions within this timeframe (eq.4.3).

$$Sh_i(\Delta_k) = \frac{\sum_{u=0}^{U} \delta_{u,i}(\Delta_k)}{\sum_{j=0}^{D} \sum_{u=0}^{U} \delta_{u,j}(\Delta_k)} \tag{4.3}$$

The final step (eq.4.4) includes the notion of remembrance. Specifically, previous Shareability values (τ_{k-1}) are used to calculate its current value - **Shareability factor(Sh_i)**, where α_1 defines the degree of importance given to past Sh_i values, just as α_2 and α_3 are used in subsequent metrics — Note the t variable is an exponent, not a superscript —.

$$Sh_i(\tau_k) = \sum_{t=1}^{k} \alpha_1^t (1 - \alpha_1) Sh_i(\Delta_t) \tag{4.4}$$

4.2.2 Document Valuation

The second extracted contextual factor a quantification on the value of a document itself. Each document is deemed to change in value as the number of edits(W action), reads(R action) and permission changes(C action) fluctuate over time. This four-part process begins by defining a partial function, giving different weights to different actions based on how much "value" they provide (eq.4.5). As previously mentioned, reading a document is seen as consuming its information, thus making it more widely available, which in turn would reduce its assumed value:

$$\omega_{u,i}(\Delta_k) = \begin{cases} 4, & \gamma_{u,i}(\Delta_k)|W| > 0 \\ -1, & \gamma_{u,i}(\Delta_k)|R| > 0 \\ 1, & \gamma_{u,i}(\Delta_k)|C| > 0 \\ 0, & else \end{cases} \tag{4.5}$$

Let's re-iterate; in the context of our study, writing (**W**) on a document is deemed four times as beneficial as clearing permission(**C**), and reading a document(**R**) without any other contributions within this timeframe degrades document valuation by a single factor. These values were empirically agreed upon with our industry collaborators, and obtaining results using alternative values is subject to future work. Using the valued weights brought on by each user(u), the averaged valued weight - $\Omega_i(\Delta_k)$ - is defined on each document(i). This normalization pattern (eq.4.6) is present in all of the engineered metrics.

$$\Omega_i(\Delta_k) = \frac{\sum_{u=0}^{U} \omega_{u,i}(\Delta_k)}{|U'_u(\Delta_k)|} \tag{4.6}$$

$U'_u(\Delta_k)$ specifically defines all users who have performed an action on this document i within this timeframe (Δ_k), or if the actions sum to 0 (i.e. negative contributions balance-out). The only shortcoming is the inability to distinguish between both cases, as seen in eq.4.7:

$$U'_u(\Delta_k) : \omega_{u,i}(\Delta_k) \ \neq 0 \mid u \in U_u(\Delta_k) \tag{4.7}$$

Using the $\Omega_i(\Delta_k)$ calculated in eq.4.6, intervals are defined to represent levels of document importance. Intervals are quite necessary since the true quantification on a document's value requires more than simply action analysis. A total of five different levels were needed in our study based on industrial requirements. Here (eq.4.8), what is referred to as the instant document value ($Va_i(\Delta_k)$) is represented as:

$$Va_i(\Delta_k) = \begin{cases} 1, & \Omega i(\Delta_k) > 3 \\ 0.75, & \Omega i(\Delta_k) > 2 \\ 0.5, & \Omega i(\Delta_k) > 1 \\ 0.25, & \Omega i(\Delta_k) > 0 \\ 0, & else \end{cases} \tag{4.8}$$

When a document thus only receives read actions (R) in sequencing timeframes, $Va_i(\Delta_k)$ will diverge to 0, as net negative contributions diminish overall values over time (at a rate of α_2). In other words, once a document has been published (as an example), many users simply read it and degrade it's value to almost nothing in the eyes of the system. This is represented with the Valuation factor($Va_i(\tau_k)$) in eq.4.9.

$$Va_i(\tau_k) = \sum_{t=1}^{k} \alpha_2^t (1 - \alpha_2) Va_i(\Delta_t) \tag{4.9}$$

4.2.3 User Cooperation

Finally, user cooperation represents the interaction between users which have access to the system which is secured by the RBAC model. This metric is interpreted using actions performed on common documents between different users. The starting point for this engineered feature is defining the highest collaborative action - $\lambda_{u,i}(\Delta_k)$ - similarly to our previous $\gamma_{u,i}$ definition (eq.4.1). However, $\lambda_{u,i}(\Delta_k)$ only contains a single value depicting the highest evaluated action on a specific document(i) by that user(u) (eq.4.10). x represents each index of the row or each possible action as defined in A:

$$\lambda_{u,i}(\Delta_k)[x] = \begin{cases} 1, & \gamma_{u,i}(\Delta_k)[x] > 0 \\ & \wedge (\exists! y | \gamma_{u,i}(\Delta_k)[y] > 0 \wedge (y < x)) \\ 0, & else \end{cases} \tag{4.10}$$

If we take the same example previously used — three Reads, one Set, and four Writes — their $\gamma_{u,i}(\Delta_k)$ and $\lambda_{u,i}(\Delta_k)$ arrays would look as follows:

$$\gamma_{u,i}(\Delta_k) = [5, 1, 0, 0, 3]$$
$$\lambda_{u,i}(\Delta_k) = [1, 0, 0, 0, 0]$$

As seen in the later eq.4.12, there is a hierarchy of importance that is given to actions. There is only a single Boolean value which is true, as again seen here in another example:

$$\gamma_{u,i}(\Delta_k) = [0, 0, 1, 7, 3]$$
$$\lambda_{u,i}(\Delta_k) = [0, 0, 1, 0, 0]$$

Once defined, all the highest social actions(λs) are combined in a matrix which represent all actions performed by each specific user(u) on each document(i), eq.4.11. Each row represents the highest social action on a specific i by u, and columns represent the set of documents(D). An empty row thus signifies no actions were performed on this i by that u within the timeframe (Δ_k).

$$\Lambda_u(\Delta_k) = [\lambda_{u,i}(\Delta_k)], \forall i \in D \tag{4.11}$$

As cooperation only exists between two different users, the weighted cooperation - ($\mu_{u,v}$) - compares the matrices ($\Lambda_u(\Delta_k)$) of every pair of users and assigns a cooperative weight to it, specifically selected for this our study once again, seen in eq.4.12.

$$\mu_{u,v}(\Delta_k) = \begin{cases} 5, & \Lambda_u(\Delta_k)[i, |W|] = 1 \wedge C1 \\ 4, & \Lambda_u(\Delta_k)[i, |S|] = 1 \wedge C1 \\ 3, & \Lambda_u(\Delta_k)[i, |C|] = 1 \wedge C1 \\ 2, & \Lambda_u(\Delta_k)[i, |X|] = 1 \wedge C1 \\ 1, & \Lambda_u(\Delta_k)[i, |R|] = 1 \wedge C1 \\ 0, & else \end{cases} ; C2 \tag{4.12}$$

$C1 = \exists a | \Lambda_v(\Delta_k)[i, a] = 1$
$C2 = \forall u, \forall v \in U, u \neq v, a \in A$

It should be noted, $\mu_{u,v}$ may not be equal to $\mu_{v,u}$, as the highest cooperative weight may not be symmetric between users. After all the weighted cooperations are defined ($\mu_{u,v}(\Delta_k)$), these values are then normalized in eq.4.13, as is the customary pattern in these formulations.

$$Co_u(\Delta_k) = \frac{\sum_{v=0}^{U} \mu_{u,v}(\Delta_k)}{\sum_{u=0}^{U} \sum_{v \neq u}^{U} \mu_{u,v}(\Delta_k)} \tag{4.13}$$

Once again, the previously calculated values are considered using the same time-degrading weight function (using α_3) to determine the current Cooperation factor($Co_u(\tau_k)$) value in eq.4.14:

$$Co_u(\tau_k) = \sum_{t=1}^{k} \alpha_3^t (1 - \alpha_2) Co_i(\Delta_t) \tag{4.14}$$

4.3 Smart Enterprise Access Control

Extracting contextual metrics from enterprise interactions is but the first step in providing more intelligent system decisions. An advanced access control scheme is also presented namely the Smart Enterprise Access Control (SEAC) technique. SEAC uses an ensemble learning approach, and is influenced by previous work in the domain of adaptive supervised machine learning methods [84]. These techniques use adaptive weighted-voting to attribute varying weights to participating classifiers based on their True Positive Rates (TPR). This TPR translates into good security ratings in our study. Individual classifiers (referenced as SEAC members) provide confidence scores instead of simple binary predictions as inputs to this technique. The combination of predictions and assigned Trust Scores (TS) aim to improve the trade-off found when considering user convenience and system security. Security rates are equivalent to precision scores, and efficiency rates are equivalent to recall scores. The notation for this section can be found in table 4.3.

Notation	Description
N	Set of i classifiers
T	Set of t time-frames
TP	True Positive
FN	False Negative
$\text{TPR}_{i,t}$	True Positive rate of i during t
α_4	Weight of previous TPR
$\text{Pf}_{i,t}$	Performance Factor of i during t
$\text{TS}_{i,t}$	Trust Score of i during t
$\text{P}_{Auth}(i)$	Predicted Rejection Confidences of i
$\text{P}_{Allow}(i)$	Predicted Acceptance Confidences of i

Table 4.3: Notation for SEAC ensemble model

Let us begin by defining the TPRs, which define the weight given to each SEAC members, being the standard TPR formula:

$$TPR_{i,t} = \frac{TP}{TP + FN} \tag{4.15}$$

Here, true positives (TP) are occurrences of suspicious actions being predicted as rejected, thus requiring authentication. False negatives (FN) are occurrences of suspicious actions being predicted as non-suspicious, prompting no authentication. Thus, a true negative would represent an action not receiving authentication, but this focus (user efficiency. i.e. True Negative Rate) is not the driving focus for SEAC. Instead, SEAC rewards its members based on their security ratings (TPR). The Performance factor (eq.4.16) defines the importance of each member's prediction over time, using α_4 as the threshold of time-lapsed evolution (in an identical fashion as the previous section).

$$Pf_{i,t} = \alpha_4 TPR_{i,t-1} + (1 - \alpha_4) TPR_{i,t} \tag{4.16}$$

Next, Trust Scores (TS) are computed and relate to the weight given to their predictions. TS is essentially the normalized $Pf_{i,t}$ across all members within the ensemble, as defined by eq.4.17. In some cases, the TF of underperforming members may converge to 0, if constantly outperformed by other members. Such occurrences would only improve results, as they identify members which provide no valuable predictions, and are therefore ignored.

$$TS_{i,t} = \frac{Pf_{i,t}}{\sum_{j=0}^{N} Pf_{j,t}} \tag{4.17}$$

Each trust score is combined with the members corresponding predictions. The cumulative weighted predictions of each member on each performed action is either P_{Auth} and P_{Allow}. More specifically, they represent the two final predictions for the entire SEAC technique, to either authenticate this action or to allow it without involvement. The final prediction of SEAC is then simply the highest confidence between eq.4.18 and eq.4.19, as seen by eq.4.20.

$$P_{Auth} = \sum_{i=0}^{N} (TS_{i,t} \cdot P_{Auth}(i)), \quad \forall t \in T \tag{4.18}$$

$$P_{Allow} = \sum_{i=0}^{N} (TS_{i,t} \cdot P_{Allow}(i)), \quad \forall t \in T \tag{4.19}$$

$$Prediction = \begin{cases} Authenticate, & P_{Auth} > P_{Allow} \\ Allow, & else \end{cases} \tag{4.20}$$

As the SEAC ensemble technique requires base classifiers at its core, three classifiers were empirically selected for comparison purposes. Random Forest (RF) represents logical classifiers, and already contains proven success in multiple areas of authentication [97] [32] by using the power of ensemble binary trees. Next is Naive Bayes (NB), a suitable candidate for statistical learning, as the most contributing features are almost independent of each other. There exist many geometric classifiers, but only a select few have proven successful in authentication studies, one being Support Vector Machines (SVM), as shown in [70], [7] and [37].

4.4 Defensive Machine Learning Results

Two different datasets are subject to analysis in this study. Namely, both a shareability-influenced dataset (SD) and a valuation-influence dataset (VD) were generated using historical user behaviour within an enterprise environment. These datasets are the basis on which we make comparisons in this section. It's important to note that document shareability and valuation are intuitively correlated, as they both pertain to conducted actions on documents by individuals. The comparison between our VD and SD datasets are a good benchmark of comparison between these two metrics, despite their potential correlation.

The experimental environment on which VD and SD where generated can be seen as a small city, where occurring events are tagged with GPS coordinates — the **where**, also known as **loc** in table 4.2 — and are scattered across the map. Most actions are located within a central radius, which should be interpreted as the head office. The remaining actions are scattered into multiple smaller clusters (e.g. off-site workplaces or secondary offices). All the clustered actions have an inherent system reaction: authentication the user, or allow them to perform their action. This binary decision is the label that is subject to our classification predictions (SEAC). It becomes evident that higher percentages of actions within the large cluster should be accepted comparatively to their smaller cluster counterparts, as authenticity is less in question when performing actions in a well-defined location (common workplace). 70% of the data was used for training and the remaining portion was divided into 30 equally-sized timeframes, which was the input to the SEAC technique for predictions.

	Security (%)		Efficiency (%)	
Models	SD	VD	SD	VD
RF	82.17	86.13	64.31	90.94
NB	98.35	95.70	47.57	58.7
SVM	81.56	85.80	63.44	92.52

Table 4.4: Optimal results with original features

First, security and efficiency metrics (i.e. precision and recall) are presented using the original features only (upper section of Table 4.2); no engineered features included. This

way, we receive an understanding of how the underlying members of SEAC perform without any of our additions. The predictions of each member (RF, NB, and SVM) are presented in Table 4.4, where the security rate of NB is seen as the highest. However, NB does also has the lowest efficiency rates. As for the other two members, both SVM and RF performed similarly, but SVM does require additional computational resources. Overall, the average improvement of security ratings was a disappointing 3.6%, but the efficiency ratings greatly improved by 17.43%. This opened discussions on the optimal selection of machine learning models, and how such selections improve our security-efficiency trade-offs.

Using the SEAC adaptive ensemble voting technique, we aim to optimize security at a much more efficient trade-off compared to all other base models (Table 4.5). This resolution was made possible since NB contains a quite precise security awareness, which almost always had the highest TS during validation. But when it comes to efficiency (which translates to user convenience), the improvement is minor. This is caused by the primary attribute of weight distribution being TPR(True-Positive-Rate), as seen in eq.4.15 and eq.4.6. Using a different focus would evidently produce different results. Overall, however, SEAC outperforms all models and improves the trade-off found between having a secure system and provide convenient access to users. The overall improvement after the inclusion of our engineered features and SEAC are 11.65% better security and 23.28% better efficiency compared to original features and models.

Table 4.5: Performance considering *Sh*, *Va* and *Co*

	Security (%)		Efficiency (%)	
Models	SD	VD	SD	VD
RF	89.90	95.37	72.46	95.79
NB	99.93	97.50	64.74	90.13
SVM	71.6	94.68	71.33	94.79
SEAC	99.60	97.50	74.01	96.80

SEAC and NB also produce quite similarly when investigating the Area Under the ROC curve (AUC). This comparison (figure 4.2) shows how security ratings are maintained successfully, while improving efficiency rates, resulting in a more improved trade-off. The dataset selection is a large influence in the results, as Shareability values were the most influential component based on our PCA analysis (explained in appendix A). An interesting consideration is the lack of influence users have on their contextual features. As our metrics use the normalization of events in most equations, contextual decisions are often based on the average events occurring in the system. This assumption may not always be valid and may be subject to future investigations.

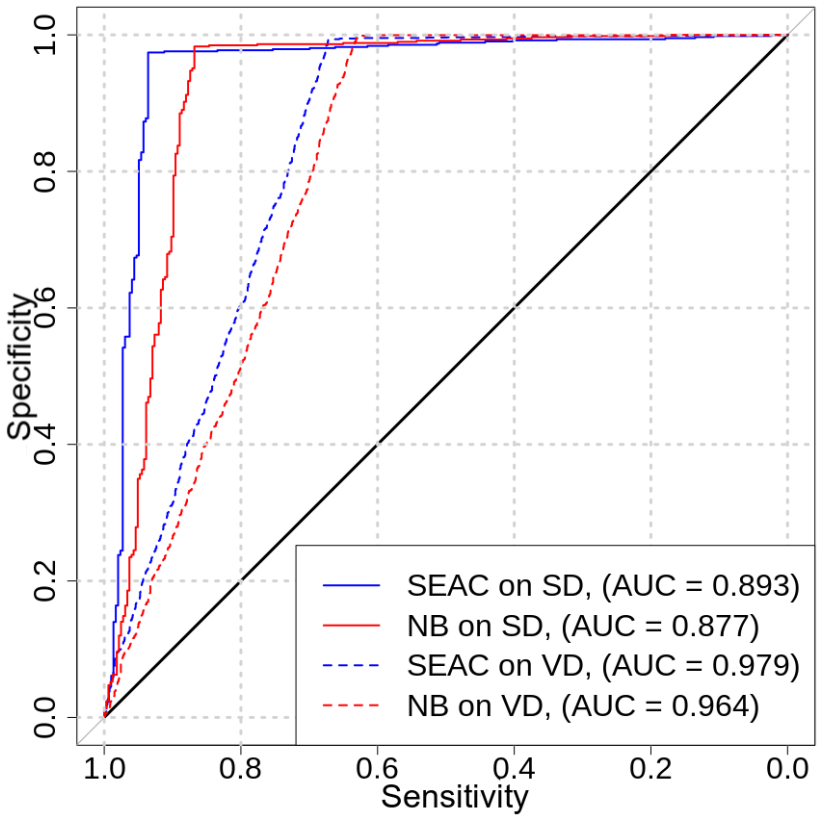


Figure 4.2: Improved trade-off of SEAC

4.5 Tunable Parameters

We begin by exploring the sampling rates seen in figure 4.3. Both undersampling and oversampling techniques were investigated, as there was a high ratio of class imbalance (95% positive with 5% negative). Using sampling offset this distribution, but too high a rate dilutes the dataset, causing misleading results. For instance, a sampling rate of 0.5 would result in high amounts of information loss, as 90% of the positive label would be lost (as to obtain a 50/50 split). A good medium was 30%, but additional data and experimental runs would be needed to further optimize this selection.

Regarding time degradation, only α_4 is presented in results in Figure4.4, although the concepts are similar for all time thresholds present in Shareability, Valuation and Cooperation metrics. When we look at figure 4.4, the optimal value is 0.2, which is hypothesized to be caused by the quantity of data in each timeframe. Changing the rate at which data is generated in each time-frame would influence the optimal selection of this weight, along with other factors. This is understandable since the influence past weights have on newly calculated values is directly correlated with how far away a slice in the past is from the current value.

Finally, variations in timeframes were investigated; a single day, a single week, every two weeks and every month. Again, timeframes correlate with the defined weights used in the degrading calculations $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$, but still performed best with timeframes set at weekly intervals (fig.4.5). Shorter timeframes thin-out the data and make events more sporadic, and longer timeframes lack in performance since metrics are not able to update frequently enough.

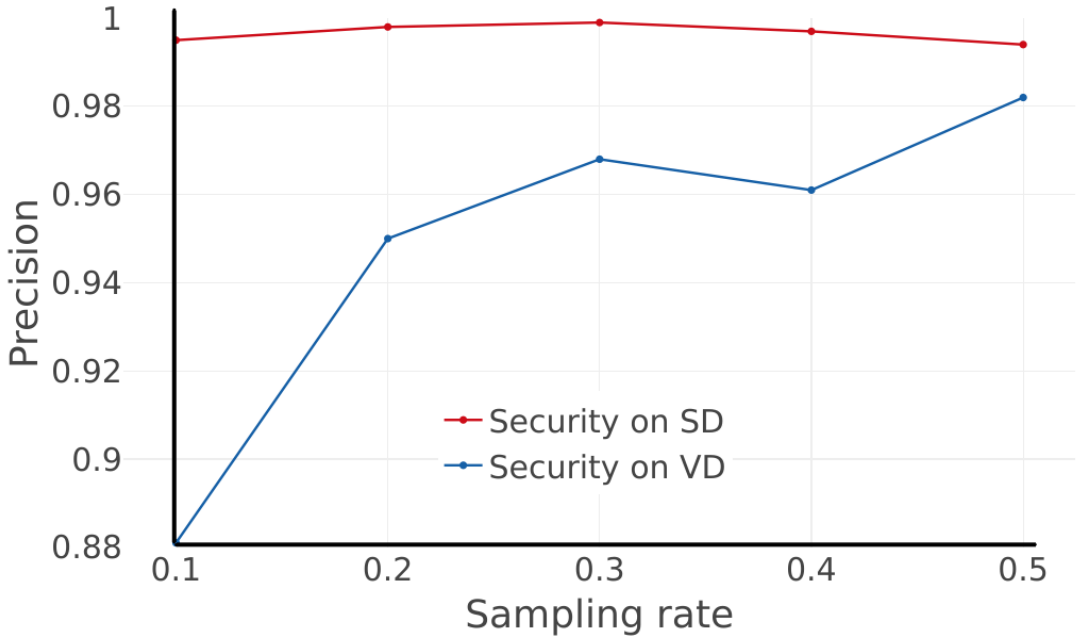


Figure 4.3: SEAC undersampling rates

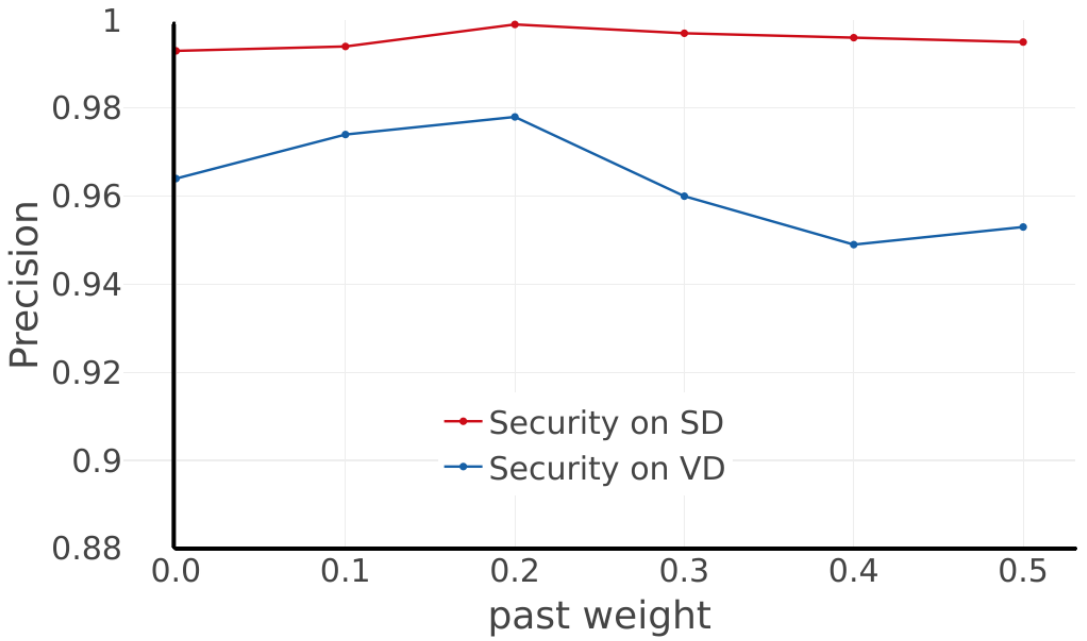


Figure 4.4: SEAC consideration of past values

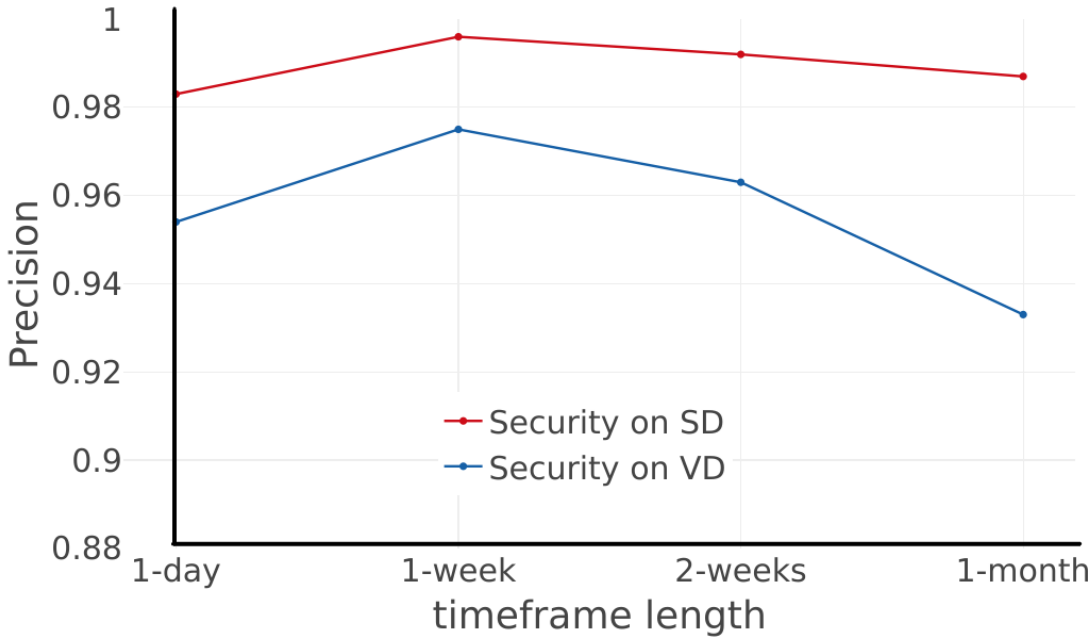


Figure 4.5: Variations in SEAC timeframes

Chapter 5

Towards Physical Context-Awareness in Cybersecurity: A Feasibility Study

With the rapid advancements in cybersecurity, it is increasingly important to analyze the many properties correlated with risk assessment decision strategies [24]. As such, nearly unreachable security standards are becoming prominent in this industry, creating misconceptions regarding the inner workings of cybersecurity systems, ranging from the lowest hardware to the most abstract software solutions [89]. As seen in [40, 87, 47], not addressing these vulnerabilities comes at a dreadful cost. However, it then becomes infeasible to cover each component of every type of system, be they cyber properties or physical attributes[56] (not to mention human behaviours and interactions [36]). All the while, the search for the most accurate risk management strategy is underway, with advancements being made due to the power of artificial intelligence and IoT devices [20]. This leaves many organizations and researchers to face a particular enigma: finding a reliable middle-ground between covering all properties within each arising context all the while maintaining a high enough confidence in risk assessment.

Our contributions in this chapter are to explore contextual physical properties in search of the existence of such a happy medium. Using the interactions which occur in a standard hospital setting, we initially focus on providing a clearer understanding of the different contextual characteristics extracted from our environment: the physical and cyber properties. With newly engineered features, we hope to expand the scope of authentication decisions under the consideration of physical properties. Then, we combine such features with artificial intelligence to attempt in improving the overall risk assessment strategy. Our improvements are tested against provided policy-based answers as preliminary results of feasibility. Again, the expansion of the contextual scope provided here refers to a hospital setting, to which providing a more extensive estimation on the security risk of appearing contexts could be beneficial to system administrators. As a precise quantification of risk profiles remains a challenge, we used specific thresholds (section 5.3) to categorize our contexts' into three different risk buckets. The takeaway from this chapter is whether or not physical domain properties can provide additional insights into context-aware solutions in cybersecurity.

5.1 Physical Context vs Cyber Context

The main focus in the provided hospital scenarios is the analysis of the contextual risk of spontaneous use-cases. In other words, we attempt to quantify how unorthodox occurring events in a cyber-physical tend to spring into existence. Particularly, we analyze captured data regarding different user roles (physicians, nurses, patients and visitors) which move around in different locations (waiting rooms, recovery rooms and hallways) all the while interacting with a variety of electronic devices (tablets, mobile phones and electronic charts). As a clearer distinction, physical components are related to the physical worldly attributes and influences between these entities. Some physical contextual patterns may emerge from (but are not limited to) these example scenarios: Does a particular nurse always carry the same tablet? Is a particular device located within the same subset of locations? Does a physician routinely frequent certain recovery rooms in any particular order? In short, these physical properties (also called physical features within the dataset) are the initial components in estimating the risk of each specific use-case, without consideration of digital characteristics.

On the opposite spectrum, the purely cyber properties of our hospital contexts only associate with the interactions between digital components; the sensitive data being accessed by specific roles, specific accounts being accessed in different typologies, and so on. Of course, simple access control protocols are in place within our experimental settings, but these are more simplistic in nature, as later described with the policy-based decisions in section 5.2. These cyber properties refer to three types of events: i) the action of reading sensitive documents, ii) identifying specific user groups who have access to a shared document in question (similar to user cooperation in chapter 4.2.3), iii) analyzing when actions where previous performed on each document.

The combination of these properties and their fluctuations are what we called the cyber-physical context of each event. Theses cyber-physical contexts are strongly correlated to the spontaneity of each use-case, which is precisely the point: as arbitrary as these types of events may seem, they shouldn't be scrutinized simply because of their lack in volume. This is where the deeper analysis of cyber-physical properties will provide insights as to how such events should be handled in dynamic environments. All experimental data is generated from realistic user behaviours, although from a different industry. All names, documents, and all other identifiable properties have been removed and replaced to preserve the privacy of individuals, as needed to maintain the integrity of this type of generative study.

5.2 Policy-based Decisions

The provided dataset contains a label system that represents a system's reaction to every occurring event; these labels are determined using a predicate-based algorithm, which shall henceforth be defined as the policy. As the policy labels are a combination of rule-based,

role-based and trigger-based decisions that are not included in our work, they are being used as the benchmark to validate our suggestions. In other words, the policy does not include any type of artificial intelligence, but should instead is our baseline for improvement (which we hope to improve using cyber-physical context-awareness). Also, the policy only takes into consideration the cyber properties of the given medical settings. On the contrary, our cyber-physical contextual risk assessment predictions are based on the clustering of engineered cyber-physical contextual properties. Additional details on the engineering features are in the subsequent section, and the clustering technique is elaborated in section 5.4).

Furthermore, the intention is to not only match the outputs provided by the policy but to also expand on those scope of those decisions. Policy decisions create outputs as a binary results, i.e. **accepting** (lower risk) or **rejecting** (higher risk) performed actions in the experimental environment. This differs from our predictions which results in either a context of **low**, **medium** or **high** cybersecurity risk, which can be assigned appropriate system responses (such as assigning **medium** to an *authentication* prompt) by its implementing owner. Our system outputs are oblivious to the policy's decisions, and solely consider the cyber-physical attributes and features derived during the decision process, as explained in section 5.3.

Thus, the improvements which are to be used alongside the current policy-based decisions are two-fold: first, to expand the underlying decisions into a ternary output, as to include more room for improved authentication decisions; and second, to include the physical context in the analysis of contextual risk estimations. If results show good potential, then we can successfully claim that using cyber-physical contextual awareness can improve risk assessment strategies (Discussed more in section 5.2).

5.3 Feature engineering in a cyber-physical context

The extrapolation of features in the cyber-physical context of our experimental scenarios are accomplished through a well-known mathematical phenomenon: coupling. More specifically, coupling metrics are derived to create correlations between certain entities in their Spatio-temporal environments. In this sense, the habitual and systematic interactions between entities are captured and clustered in the machine learning phases of the study. These entities which are the subjects of all engineered features are as follows:

- **Users**: Participants in a hospital context with a variety of possible roles: Physicians, Nurses, Administration Officers and Visitors are all included in our experimental settings. Users move around in the environment, and whenever an action is performed, their associated identifiers and GPS coordinates are added as metadata to the action.
- **Devices**: Electronic devices on which actions are performed by a user. These devices are simply represented as mobile devices in our study, but in theory could represent

any electronic-wireless device found within a hospital (cellphones, tablets, E-charts and laptops). Adapting our predictions based on the device type is subject to future work.

- **Locations**: Specific areas that compose the environment where users and devices navigate. These areas were modelled after (but not limited to) hallways, waiting for areas, emergency rooms, consultations rooms and nurse pits.

Therefore, the correlation between all three entities is circular in nature; as such, we make associations in a unidirectional coupling system, as to not derive duplicate features from the same contextual properties. In layman's terms, the coupling between a single user and multiple devices contains the same information as the coupling between a single device and multiple users but represented as an alternate metric. That is not to say it is not worth investigating in future work, but for this study, we focus on each entity being coupling with multiples of others, in turn, as seen in fig. 5.1.

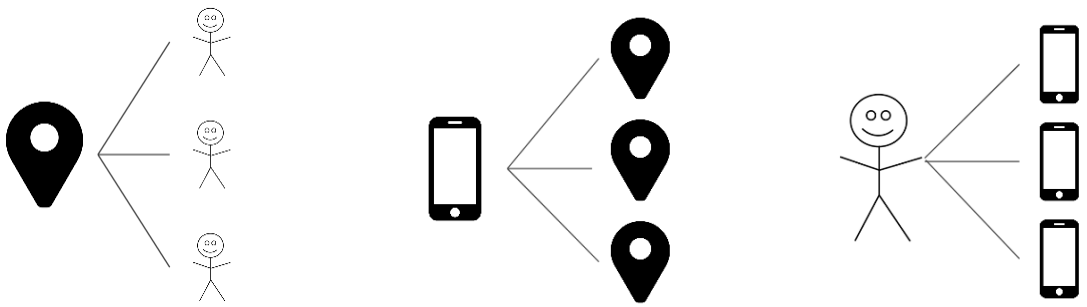


Figure 5.1: Coupling explained - Devices, Users and Locations

Knowing the ideology of coupling between entities, we create two types of coupling metrics: continuous coupling (CC) and instantaneous coupling (IC). The former is a representation of coupling over a continuous time frame, representing a kind of coupling average. Continuous coupling is accordingly a metric that captures common and frequent behaviours (such as hourly checkups on patients) and uncommon and infrequent events (such as a visitor entering a surgical room). On the flip-side, instantaneous coupling represents the sporadic events that occur between entities. These include uncommon and frequent events (ad-hoc meetings in the hallway) and common and infrequent events (weekly sync meetings occurring in different rooms every time). Table 5.1 demonstrates the notation for the composition of both CC and IC in the subsequent sections.

5.3.1 Continuous Coupling

The notion of adaptive access control over time is the motivation behind both coupling metrics. CC achieves this requirement by forcing events and behaviours to slowly transition

Notation	Description
$d \in D$	Set of devices
$u \in U$	Set of users
$l \in L$	Set of locations
$TC_{x,y}$	Time Consumed by entity x with/in y
CC	Continuous Coupling
IC	Instantaneous Coupling
Δ_t	Length of timeframe, unit of time in days
α	period of day: Morning, Afternoon, Evening or Overnight
β	day of week : Monday through Sunday
γ	running count of weeks (week 1, week 2, ...)
a_1	First relative threshold (20% in our study)
a_2	Second relative threshold (60% in our study)

Table 5.1: Definitions of physical context nomenclature

from higher risk to lower risk, and vice-versa when they leave the system. As a concrete example, a newly employed nurse who begins her training appears in our experimental environment. This new appearance will cause her involved contexts to be seen at a higher risk compared to nurses who have already established trust in the system. Of course, this is only temporary, and once enough trust has been built (i.e. enough timeframes, Δ_t, have passed), the system will then consider our new nurse's involvements to be more trustworthy.

As previously mentioned, CC is effective at capturing the risk levels of common and frequent behaviours and events, and also uncommon and infrequent behaviours and events. This is accomplished by calculating a running average of coupling between entities, and flagging anomalies that fall below our defined thresholds. These thresholds are defined as $a1$ and $a2$, which were agreed upon at 20% and 60% variance from the norm (with same industry partners), although other values could be investigated in future studies. Each CC metric is the normalized calculated value between a single entity and all entities of another type; a one-to-many relationship in other words, but calculated for every single entity.

The first CC metric represents the time each user spends in a particular location. This is calculated by observing the time each user spent in this location, and normalizing it across all users, as seen in eq. 5.1.

$$CC_{u,l}(\Delta_t) = \frac{TC_{u,l}(\Delta_t)}{\sum_{v=0}^{U} TC_{v,l}(\Delta t)} \tag{5.1}$$

In an identical pattern, the CC between each device and all locations is found in eq. 5.2, as to represent which locations a device spends its time. Again, these metrics are recalculated at every timeframe (Δt) to allow for interactions of entities entering and leaving the system.

$$CC_{l,d}(\Delta_t) = \frac{TC_{l,d}(\Delta_t)}{\sum_{i=0}^{L} TC_{i,d}(\Delta_t)} \tag{5.2}$$

Finally, the last CC relationship with the same pattern is the relationship between each user and all devices. Following the same normalization pattern once again, this relationship is represented in eq. 5.3.

$$CC_{d,u}(\Delta_t) = \frac{TC_{d,u}(\Delta_t)}{\sum_{j=0}^{D} TC_{j,u}(\Delta_t)} \tag{5.3}$$

Each of the above CC metrics is filtered through our defined thresholds to produce a CC risk metric, which is a ternary categorical value. The values of x and y are replaced by our three entities (d, u, l) to produce three categorical values, one for each CC metric. We compare the difference in CC between the current and previous timeframe in producing the risk level. The eq. 5.4 is updated continuously as contexts in the system evolves.

$$CC_{x,y} = \begin{cases} low, & |CC_{x,y}(\Delta_t) - CC_{x,y}(\Delta_{t-1})| < a_1 \\ medium, & a_1 <= |CC_{x,y}(\Delta_t) - CC_{x,y}(\Delta_{t-1})| < a_2 \\ high, & |CC_{x,y}(\Delta_t) - CC_{x,y}(\Delta_{t-1})| >= a_2 \end{cases} \tag{5.4}$$

This risk-level is referred to as the categorical labelling and is solely used for majority voting in the cluster identification stage of the consensus algorithm (section 5.4). In an environment with three users, two locations and three devices, there would be twenty-one CC values:

- Three users coupled with every device = 3 * 3 = 9
- Two locations coupled with every user = 2 * 3 = 6
- Three devices coupled with every location = 3 * 2 = 6

Many of these values would be 0 if those entities are never in contact with each other. As such, when new relationships are made (such as a nurse being assigned to a new section), this event would be seen as a higher risk, and influence many contexts in that new location. This is intended behaviour, as the appearing contexts in this section need to take into account the newcomer which still needs to prove their trustworthiness.

Despite CC being capable of adapting for changes in our environments, there are still events and behaviour which are not properly captured. Theses uncaptured scenarios are the rationale behind the creation of the instantaneous coupling metrics.

5.3.2 Instantaneous Coupling

The major flaw of the CC metrics is its inability to correctly adjust to events and behaviours which occur in predictable patterns, but with alternating contextual features. Such ad-hoc events are common throughout different systems, and thus need a distinct coupling metric for proper representation. This is where instantaneous coupling (IC) comes in. By looking back at specific moments in time, IC allows for ad-hoc events which happen at similar times of the day, days of the week, and/or weeks of the month (noted by variables α, β and γ, respectively) to not be seen as anomalies in the system, even if instantiated in varying contextual properties. Coupling between metrics in IC are much more rigid; they are calculated on a one-to-one basis and do not normalize across entities, but instead consider the average time consumed (TC) in the past K weeks. This way, uncommon but frequent events can be seen as normal and trusted events, granted they involve the same two entities. The same is also true for common and infrequent behaviours and events. IC takes into consideration the following time metrics:

- Period of the day (α): Represents the time of day in which events occur. Each day is segregated into either the Morning (5am - 11:59am), the Afternoon (12pm to 4:59pm), the Evening (5pm to 9:59pm) or Overnight (10pm - 4:59am). These categories rotate through values 0-3, respectively.
- Day of the week (β): A simple representation on which day of the week an event occurs. Mondays are represented with 0 through to Sundays with are represented as 6.
- Count of weeks (γ): An ever-increasing integer, which represents the number of weeks that have elapsed since the beginning of the experiment.
- Weeks in timeframe (K): The number of weeks that are within a timeframe, not to be confused with γ. This value is calculated at the start of the experiment and remains constant throughout all calculations.

The initial step in calculating IC metrics is the definition of the value of K, which is simply the number of weeks within a timeframe. As timeframes are defined in days, the value of K is simply as follows:

$$K = \left\lfloor \frac{\Delta_t}{7} \right\rfloor \tag{5.5}$$

The value of K gives us an indication of how many weeks back to look upon. Each IC metric is calculated by taking the average time consumed by two entities in the current instant and compare it with the average of the previous K weeks. For instance, the IC between each user and device is calculated as follows:

$$IC_{d,u}^{\alpha,\beta,\gamma} = \frac{K \cdot TC_{d,u}^{\alpha,\beta,\gamma}}{\sum_{k=1}^{K} TC_{d,u}^{\alpha,\beta,\gamma-k}} \tag{5.6}$$

An identical pattern applies to the other two coupling relationships, as seen in eq. 5.7 and eq. 5.8. All three IC relationships are recalculated as time progresses, incrementally increasing the value of γ (as both α and β are cyclic variables).

$$IC_{l,d}^{\alpha,\beta,\gamma} = \frac{K \cdot TC_{l,d}^{\alpha,\beta,\gamma}}{\sum_{k=1}^{K} TC_{l,d}^{\alpha,\beta,\gamma-k}} \tag{5.7}$$

$$IC_{u,l}^{\alpha,\beta,\gamma} = \frac{K \cdot TC_{u,l}^{\alpha,\beta,\gamma}}{\sum_{k=1}^{K} TC_{u,l}^{\alpha,\beta,\gamma-k}} \tag{5.8}$$

Once quantified using the past K values, IC values are again parsed through our predefined thresholds ($a1$ and $a2$) as to create categorical values for use in our consensus algorithm. The thresholds are identical to those used in CC formulations.

$$IC_{x,y} = \begin{cases} low, & IC_{x,y}^{\alpha,\beta,\gamma} < a_1 \\ medium, & a_1 <= IC_{x,y}^{\alpha,\beta,\gamma} < a_2 \\ high, & IC_{x,y}^{\alpha,\beta,\gamma} >= a_2 \end{cases} \tag{5.9}$$

These thresholds can also be seen as allowances, to which values can fluctuate before flags are being raised. A visual representation of theses thresholds can be found in figure 5.2.

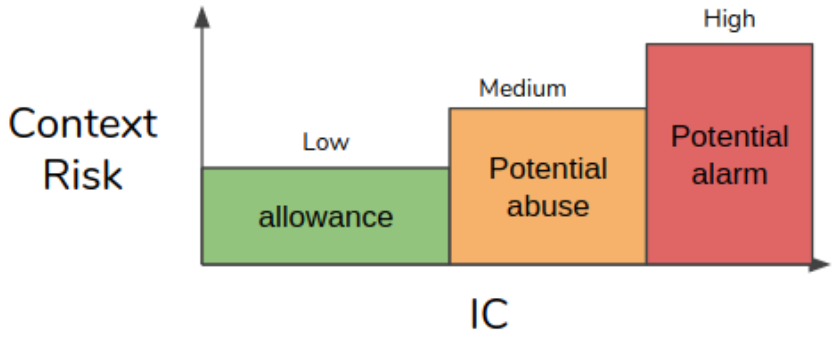


Figure 5.2: How risk thresholds are labelled

As the engineered features are calculated, they are the inputs to our consensus algorithm, which is defined in the next section. This algorithm uses artificial intelligence to cluster and label events and behaviours with a risk level in real-time, as they appear in the system.

5.4 Risk Prediction using Machine Learning

Once all engineered features are extracted, the risk-prediction phase is evoked, also known as the consensus model. This model analyzes all the provided metrics and attempts to make sense as to how the system should handle each given context, represented by single data points. It is worth noting that policy decisions (section 5.2) remain unknown to this model. The consensus model first clusters each incoming data point then uses majority voting to identify each cluster, to then input each data point into the consensus algorithm (algorithm 1), which finally outputs the final decisions. Here is an overview of each distinct step:

- Clustering: Initially, all quantitative values (not to be confused with categorical values) of both CC and IC are clustered into three clusters by three different unsupervised learning methods: K-means, DBSCAN and Gaussian Mixture Model. In the case were less or more clusters are created (especially common with DBSCAN), the largest (in volume) three clusters are used in subsequent steps of this phase.

- Majority voting: After each unsupervised algorithm has made their corresponding clusters, they require the identification of each other's clusters. This is accomplished using majority voting, specifically by using the categorical values calculated in eq. 5.4 and 5.9. By looking at the counts of categorical values of a cluster, each cluster can be identified as being *low*, *medium* or *high* risk. For instance, if 300 data points are clustered together, and 250 of them are categorized as being *low*, then the entire cluster would then be identified as containing only *low* data points, as long as no other cluster contains more than $250/300 = 83\%$ *low* data points.

- Consensus Algorithm: An algorithm which takes the clustered data points of each model in the majority voting stage and outputs a risk prediction on each data point, or each context. Using the predicted risk level of each given context, the proper authentication decisions can be applied by system administrators.

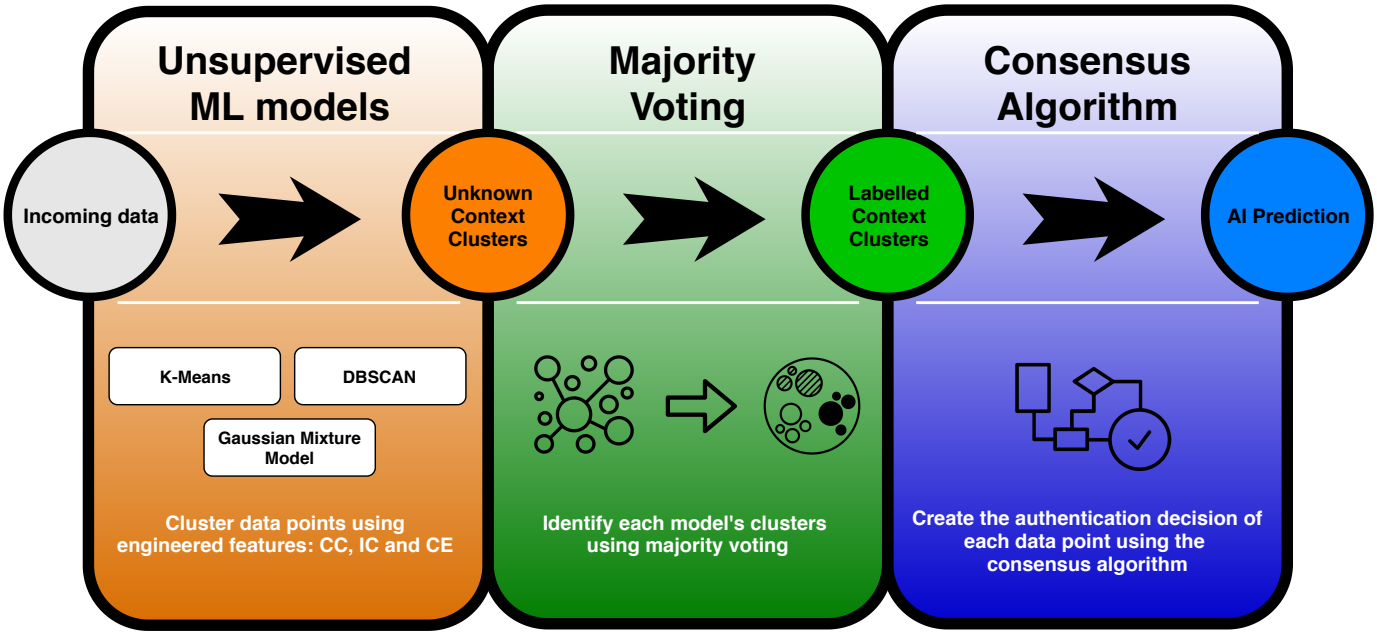


Figure 5.3: Consensus Model Overview

An overview of all three steps can be found in figure 5.3. One detail of note is the output of the consensus model. As previously mentioned, the final output is a ternary decision: the risk level is either categorized as being *low*, *medium* or *high* contextual risk level. This differs from the Policy, which outputs a binary output by design. The correlation between *low* consensus model outputs and the *permit* outputs of the policy, just as the *high* consensus model output and the *deny* output of the policy, are investigated and used as performance metrics in the ensuing section.

As seen in algorithm 1, the consensus algorithm outputs risk levels based on the number of outputs of each type; this algorithm comes however with a caveat. It must contain an odd number of predictors, as to break any ties in predictors outputs. Three clustering techniques were used in our study, but incrementally increasing the value to a higher odd number of predictors could be the subject of interesting future work.

Algorithm 1 Consensus algorithm pseudocode

Require: Clustered points(N) of each model (M)

```
1: function CONSENSUS(clusterPoints[N][M])
2:     for n in N do
3:         decisionArray[n] ← Unknown
4:         if clusterPoints[n].unanimous() then
5:             decisionArray[n] ← clusterPoints[n].any()
6:         else
7:             clusterPoints[n] ← clusterPoints[n].removeMediums()
8:             if clusterPoints[n].unanimous() then
9:                 decisionArray[n] ← clusterPoints[n].any()
10:            else
11:                countLow ← clusterPoints[n].CountLow()
12:                countHigh ← clusterPoints[n].CountHigh()
13:                if countLow == countHigh then
14:                    decisionArray[n] ← Medium
15:                else
16:                    decisionArray[n] ← highest(countLow, countHigh)
17:                end if
18:            end if
19:        end if
20:    end for
21: return decisionArray
22: end function
```

5.5 Initial Results regarding Feasibility

The performance of our consensus model alongside our engineered features requires a comparison of outputs between both the policy-based decisions and the consensus-based outputs. However, the integration of the *medium* contextual risk from the consensus model becomes troublesome. This challenge is initially addressed during the quantification of our performance metrics. In detail, three metrics are created to produce a means of comparison between the policy outputs and the consensus model outputs. Theses three metrics were created empirically, and future work is required in the precision and derivation of their performance. As a quick preface to these definitions; Positives are seen as *denying* actions which occur in a *high* risk environment, and negatives are seen as *permitting* actions to occur in a *low* risk environment. Thus, the three performance metrics are:

- **Security:** Percentage-based metric representing the system's security robustness. The security metric is defined based on precision of the consensus's model *high* and *medium* clusters added together (predicted positives) compared with the policy's *deny* decisions

(actual positives). This calculation is similar to standard precision calculations, with the addition of the added *medium* dimension. The true-positives (TP) are then defined as predicted positives which are indeed positive, and the false-positives(FP) are the predicted positives that are negative, meaning the policy would *permit* a predicted positive. Therefore, the security rating can be seen as follows:

$$Security = \frac{TP\ (highs) + (mediums)}{TP\ (highs) + (mediums) + FP\ (lows)} \tag{5.10}$$

- **Convenience:** Similarly, the convenience factor is how adequately the system responds and grants continuous authentication to users in low-risk contexts, as to decrease the need for user interaction. Using a similar pattern to the security rating, convenience is calculated by adding the *medium* and *low* predictions (TN) of the consensus model (predicated negatives) which would be *permitted* by the policy (actual negatives). In combination with any predicted negatives which would have otherwise been identified as *denied* by the policy (FN). This gives us the following equation:

$$Convenience = \frac{TN\ (lows) + (mediums)}{TN\ (lows) + (mediums) + FN\ (highs)} \tag{5.11}$$

- **Matching Ratio:** Since there is some discrepancy between using the *medium* contexts as both a negative and positive prediction (caused by comparing a ternary with a binary output), the matching ratio is used to measure the direct association between the policy's *deny* and *permit* outputs with the predicted *low* and *high* predictions, removing any *medium* predictions. Therefore, the quantification of the matching ratio becomes the standard definition of accuracy, but was renamed as to alleviate any confusion:

$$MatchingRatio = \frac{TP\ (highs) + TN\ (lows)}{TP\ (highs) + TN\ (lows) + FP\ (highs) + FN\ (lows)} \tag{5.12}$$

It is worth noting that around 45% of outputs are labelled as a medium contextual risk by the consensus model. Even with such a high ratio, the matching ratio demonstrates that all the low and medium points are successfully matching with the policy decisions, and this is true for most models, not only our ensemble model. The results of all individuals models and their combination into the consensus model are shown in table 5.2.

Considering the many tunable variables which also exist within this study, the most influential input in our model was the size of each timeframe. This factor alone directly

Models	Security	Convenience	Matching Ratio
DBSCAN	82.35	84.44	97.87
K-Means	86.69	80.74	91.89
GMM	67.94	97.81	96.94
Consensus	88.51	77.52	99.90

Table 5.2: Consensus results using $\Delta_t = 35$ days

pushed the security-convenience trade-off to either side. The longer the length of each timeframe, the better the security rating (at the cost of lower convenience). The shorter, the better the convenience was, at the cost of security. This phenomenon becomes simple when analyzed; longer timeframes cause clusters to require more actions to be performed by users with varying devices and locations before trust can be built. In other words, the longer the timeframe, the harder it is to gain the system's trust, but the easier it becomes to detect anomalies. Depending on the severity of the system and the privacy of its data, the exploration of the crossover point (i.e. around 30 days in fig.5.4) becomes interesting, as this would be the pinnacle of security-efficiency trade-off. However, just as in most cybersecurity studies, security is more often seen as the primary focus, and convenience falls more as a "nice to have" metric.

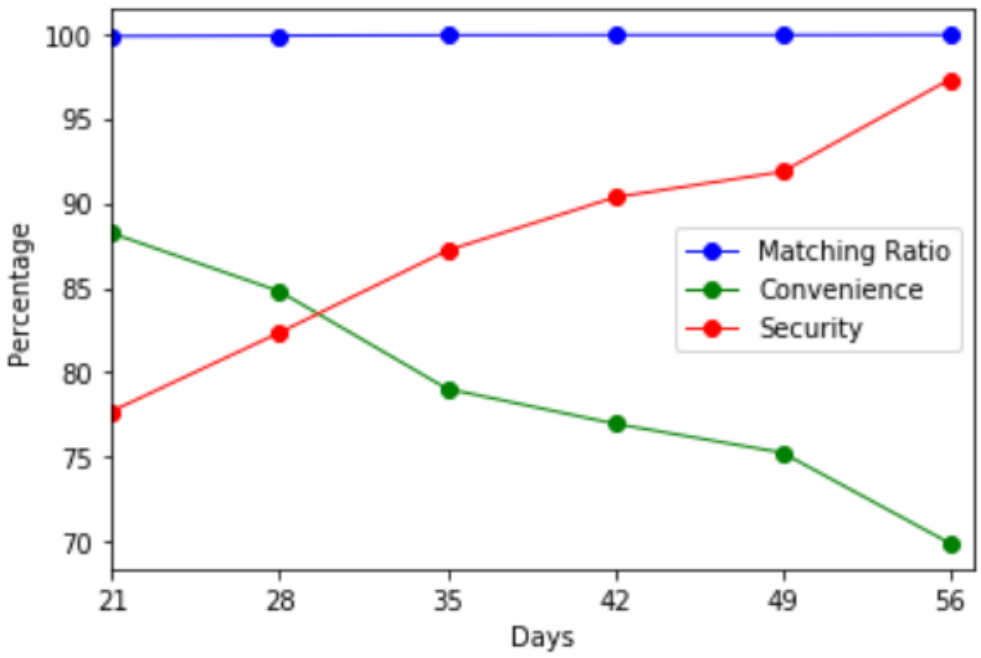


Figure 5.4: Fluctuation of Consensus model results over time

In conclusion, we've successfully replicated the reactions of the policy-based decisions using context-aware artificial intelligence. Despite not being perfect, we were able to utilize

cyber-physical contextual properties in our model to extract sufficiently valuable metrics for our consensus algorithm to produce acceptable performance metrics. This confirms the feasibility of applying context-awareness and machine learning to improve the cybersecurity of a hospital setting, despite the minor requirements.

Chapter 6

Conclusion

We've discussed how emerging contexts in our everyday lives contain information that is yet to be captured, and we've taken the challenge of extracting, analyzing and utilize this information. As we focus on cybersecurity, only certain domains of information are integral to improve decisions relating to authentication; our scope pertains to the cyber-physical domains. If only the most relevant information regarding user identity is extracted, then the authenticity of each individual can be estimated, without knowing every exact detail of every unfolding scenario. This is possible thanks to artificial intelligence, which allows the high complexity of contextual analysis and all of its encompassing features to be interpreted using machine learning algorithms. However, such techniques have two-fold applications; to attempt to inject and interfere with systems wanted to authenticate their participants, or to mitigate and identify participants despite all of the anomalies and malicious attempts. The technological advances have allowed such methods to not only pull information from the digital world but also extract information from our physical environments, thanks to the wide array of sensors available on modern devices. Initially, we extracted and analyzed the context and behaviour-based features that appear in the ever-expanding scope of authentication techniques. The recognition of contextual details is the initial step in closing the gap to achieve true continuous authentication. All this untapped data potential can be leveraged to increase user security, privacy, convenience, and progressively reduce uncertainty on the verified identity of the device owner. As multiple studies contain proven methodologies that can improve existing authentication methods, they also come with their own newly introduced challenges.

We first investigated the application of these many contextual features in an adversarial attacker model called SINAM, in the hopes of providing a tool for future developments in mobile crowdsensing security solutions. This study was possible thanks to an internally developed mobile application that collected sensory data on willingly participating graduate students for 3 weeks on the university's campus. Our data collection campaign was utilized in creating our main SINAM model, which we compared against modern clustering techniques and a less novel random injection model. Our algorithms relied on the collaborations of accomplices within our MCS campaign in successfully injecting behaviour patterns into our

victim's data batches. Our experimental results validate that the behaviour injections of SINAM were adequately effective, being undetected around 85-100% of instances (when using a batch size of 50 and tampering potential of 0.5). The introduced model may be applied to test authentication or identification methods — or other applications requiring user authenticity — that could benefit from an intelligent injection model within any real-time data collection system.

Next, we investigate how contextual features can enrich authentication decisions within an enterprise environment. With the assistance of collaborators from the private sector, we engineer three new features in a typical enterprise context to improve the authenticity accuracy of a cyber system. We discuss how our three new features — document shareability (Sh) document valuation(Va) and user cooperation (Co) were successful in improving both the security and efficiency of system usage by an average of 4% and 17 % respectively. Such results can also obtain security ratings of up to 99% with the right model selections. Alongside these findings, we discovered that an interesting trade-off exists between system security and efficiency depending on the selection of machine learning models. This led to the next contribution: constructing a method that focuses on improving security or efficiency rates, depending on priority.

Our subsequent contribution does just as aforementioned: enhance the trade-off between an enterprise's system security and the efficiency of its accessing user. By utilizing a weighted voting technique with the underlying members, SEAC was able to detect security anomalies at a rate of up to 99.6%, all the while improving the affected efficiency of active users. Compared to the original data and machine learning results, SEAC improved the system's security by 11.65% and system efficiency by 23.28%. Our contribution is not solely tied to our experimental settings and domain, as we provide different parameters that can be tuned to meet the needs of other specific studies in different fields. Our main takeaway from this work is how much unsuspecting potential the varying contextual features contain, and how these features have varying levels of improvements on the different machine learning models.

The final contribution aims to provide insight on future research regarding cybersecurity within the health industry, specifically regarding cyber-physical aspects of evolving contexts. To explore the evolving circumstances and estimate risk levels of appearing contexts, we suggest six-cyber-physical coupling metrics. Then, we demonstrate how these features could be analyzed using a formulated consensus model to produce our three different tiers of risk: low, medium and high-risk contexts. Our proposed consensus algorithm obtains initial results of 90% system security under specific circumstances. Such a result also came with an interesting crossover-point when investigating security and convenience trade-offs. The overarching takeaway from our results is clear; cyber-physical systems require time in creating a trust for its participating to diminish the risk involved in providing continuous authentication to authentic users, but the utilization of cyber-physical contextual-awareness strategies can improve the cybersecurity of hospital environments.

6.1 Current Challenges

Our final remarks are re-iterated; formulating and extracting the many characteristics of underlying contexts is not an easy task. There remain many unknowns, but the potential that these undiscovered properties and correlations contain have shown to make positive advances in research. A common difficulty seen across all studies is the quantification of these contextual metrics without a common baseline. We attempted to address this issue with a variety of tunable parameters, as to allow enough flexibility for applications in different experiments.

Furthermore, our generated datasets and applied methodologies were mostly created for experimental purposes, and may not be suitable for live software applications. Providing multiple levels of authentication decisions — or fine-grained authentication decisions — could render our solutions more applicable to the many industries which rely on user interactions for authentication purposes.

Also, the scope of each of our defensive studies did not include elaborate attacker models; evidently, we are currently working to apply our attacker model (SINAM) to them, but such a task is still not yet complete. Only after we explore such experiments and potentially expand even further our formulations can we begin the exploration of cross-discipline solutions, which a challenging future endeavour indeed.

As diligent as developments in context-based authentication may be, there is an unfortunate truth regarding data availability [55]. Contextual analysis is only valid with the presence of measurable and specific contextual properties. Without them, or even if properties exist in which we cannot associate with any underlying context, then such data is rendered useless and a missed opportunity. For instance, the presented social authentication solutions [7] [97] extract contextual features from a limited set of applications, but it fairly possible that other features may be overlooked (i.e. regarding a hidden context). Therefore, mining individual social contexts are of paramount importance in a continuously evolving social trending community [6].

As vasts amounts of data are created in the era of IoT [1], the privacy of individuals is a reoccurring and increasing concern amongst data scientist. Collaterally, identical privacy concerns need to be addressed during the extraction of contextual information [94]. Not only privacy but also other metrics such as user trust and identity are crucial components requiring transparent solutions. As such, proven methods, guarantees and incentives need to be implemented for users to willingly participate [79] in providing context-aware data for theses context-aware strategies and solutions to see commercial applications.

Many third-parties would be interested in obtaining personal information for different applications, e.g. advertisement. Considering the wide array of sensors available on mobile phones, such information will grow in value in the upcoming years. However, different strategies exist which can improve users' privacy, protect their identity and gain their trust. As pointed out in [79], the following five strategies can be considered: i) Privacy-triggered networking: delay network communications until an appropriate privacy level is achieved (using

predefined metrics) ii) Privacy-preserving schedule: use distributed computing and cryptography while keeping a private schedule iii) Pseudonyms: constantly change pseudonyms which represent location and identity to confuse potential eavesdroppers iv) Usage Control: Service providers must increase transparency on information use v) Context-Aware Policy Management: adjustment of access control can enhance privacy by increasing authentication mechanisms in unsafe contexts or decreasing them in familiar contexts. These are the exact requirements for realistic deployment of context-aware authentication techniques; not many users willingly reveal their information, and there are few incentives for participation.

6.2 Remaining Opportunities

The ability to select the most applicable authentication methodology (or combination of methodologies) requires methodologies to have clear quantifications of what successful and failed methods would look like [58]. In the domain of biometrics, the concept of using multiple sensors in a single authentication process is gaining popularity, despite having its pros and cons [74].

One such example of an optimal biometric method has been investigated in [70], with the sole objective of improving User Experience (UX). The metric of optimality is maximized in this study, even if other aspects falter as a result; providing effective resource allocation, having direct biometric availability or even consistent biometric success rates all fall short as a result of optimizing UX [70]. However, since larger volumes of extracted features translate into higher data variety [106], then vast pools of identification methods can be born as a result.

Moreover, further advancements are also possible by analyzing offline social factors (e.g. cultural backgrounds, beliefs, and prior experiences) [100]. Such hidden features are yet another example of hidden contexts containing potentially valuable features regarding identification and authentication techniques.

As an example of creating such large pools of data, we could extract and learn the behaviour of individuals (known as behaviometrics) [7], while simultaneously exploring a device's noise levels [77], and use other additional sensors [113] to make a combined or iterative decision. Even once authenticated, the confidence of authenticity may have its limits. By potentially solving the complexity issue found in [70], continuous monitoring of environmental sensors could provide higher data quality, translating into even higher authentication confidence. As increasing volumes of data becomes available [69], the iteration and selection of different techniques could lead to further promising results.

Certain situations with higher risk and strict requirements necessitate higher confidence levels for real-time authentication applications. Such circumstances, therefore, have potential in data enrichment with additional contextual extraction from underlying contexts. One such example can be seen in [45], where vehicular verification confidence contains high repercussions when errors occur. Admission to vehicular clusters requires robust defences

and near-perfect confidence scores, as malicious entry could be detrimental to the safety of the passengers and the corresponding environment [45]. For such critical applications, convenience and minimal user intrusion should not have higher priorities than security, and applications should be surgical and transparent in their uses of contextual properties.

6.3 Future Directions

As our digital footprint ever so increases, so too should our utilization of it's newly created contextual information. Each contribution of this thesis is centralized around the notion of utilizing contextual properties in different environments to increase authentication efficiency, but this latter section discusses not only its potential expansions in such a field but also how contexts can be analyzed in other industries for similar technological improvements.

6.3.1 Expanding on Attacker models

Beginning with our first contribution, many adjustments could improve SINAM's efficiency, expand its scope, or even increase its scalability. The first and most interesting potential improvement is the adaptation of the tampering potential. Instead of keeping it as a constant (eq. 3.11), explorations could be made in dynamically generating the tampering potential. More specifically, a formulation could be made to increase the weight of the tampering potential based on the temporal deviations found within the features (eq.3.10). Currently, these steps are distinct, but creating a dependence between them could dynamically create more aggressive injections in the event of high feature fluctuations. The aim would be to attempt to maximize the ratio of malicious data injected while maintaining an acceptable threshold of undetected injection rates.

Also, SINAM monitors and injects on a single victim at a time, but multiple instances of SINAM might create interesting results. Specifically, multiple injections could be created, and so instead of injecting a single behaviour in every N behaviours of a batch, a certain percentage of injections could be created; say to fill 20, 30 or 40% of an incoming batch with malicious data. This multi-attack phenomenon would be interesting also in its ability to then takeover an entire MCS campaign, by synchronizing the attacks on all the victims for macro-manipulation of user behaviour. On a similar note, a similar idea is the collaboration of multiple instances of SINAM communication in parallel while injecting on their victims. In other words, each 1-to-1 victim-SINAM relationship would remain, but the many instances of SINAM could report each other's successful injections. This way, they potentially categorize and adapt to details relating to feature susceptibility in varying contexts. They can further be expanded in creating an overarching successful-injection cluster, to be used by future SINAM models. This new cluster would be categorized data points as successful or failed injections, and new injections could be "pre-clustered" to foreshadow the success of the new injections. This would render more intelligent and adaptive (as context-based success would

evolve, and so would the overarching cluster) behaviour injections over time, expanding the current creation process (eq. 3.1).

Another idea was started, but not included in this thesis: the injection on entire features instead of injecting on entire behaviours. The idea behind using SINAM to instead create malicious injections for the manipulation of behaviour can be slightly adjusted, instead of creating injections to manipulate the value of a specific set of sensors. Instead of creating injections on columns (i.e. behaviours), SINAM would use similarities in sensor values to "falsify" the data passed by a victim. As an example in a typical MCS experiment, the GPS sensor could be targeted on all victims, and if such injections go undetected, SINAM could cause victims to not be selected for geo-spatial demanding tasks, despite being in the correct location. This way, the accomplices could obtain the monopoly of the system's utility from those particular sensing tasks, and essentially take over a MCS environment.

Also, the mobile application which collected data (CROWDSENSE) could be adjusted in obtaining additional data. Specifically, sensory values could be read at different rates, rather that all being capture at a 20 seconds interval. Furthermore, the concept of average-sensory data pulling could be investigated, instead of simply pulling at each interval. In other words, when data is sent to our collection server (CROWDSERVER), sensory values could be constantly polled and averaged out, instead of only pulled at a provided set interval. This concept could further be enhanced to have different rates for different sensors, depending on their fluctuation rates.

Finally, as certain accomplices might be able to "mimic" a victim's behaviour, this would cause them to dominate the behaviour closeness distance function (eq.3.5). Therefore, if an accomplice is associated with most (if not all) i_k indexes, then they could start to take over their behavioural identity. Essentially, by aggressively and successfully creating injections on a particular victim by using only a single accomplice's behavioural data, SINAM could cause DBSCAN to only accept behaviours to the victim's behaviour profile if they came from that accomplice since the injections would have "invaded and overwritten" the victim's behaviour profile. Using SINAM to replace all victims with accomplices in an online-learning setting may be a worthwhile investigation.

6.3.2 The Future of Enterprise Access Control

Next, we investigate the potential improvements regarding our study in chapter 4. Since this chapter contains two contributions, we shall begin by describing opportunities relating to our three engineered features before moving on to our SEAC improvements.

Beginning with the feature engineering phase, certain equations had pre-determined values as agreed with our industry partners. On example of this are the weights associated to action types in each of the three contextual metrics (i.e. eq.4.2, eq.4.5 and eq.4.10). These weights were quantified based on the value produced by users and documents in an enterprise context, but additional investigations might appropriate varying quantifications depending on the domain of study.

Furthermore, the concept of time-value degradation using the values of $\alpha_1, \alpha_2, \alpha_3 and \alpha_4$ were briefly investigated in section 4.5, but a more dynamic approach to these quantifications could be investigated. Maybe the concept of hyper-parameter tuning could include not only the parameter of the different SEAC members, but could also include these weights. This would come at the cost of additional computation complexity, which might be a problem for real-time applications of our model, but still worth investigations.

Moving on to our SEAC technique, the concept of obtaining performance factors of zero was briefly introduced. Specifically, when a member of the majority voting stage does not provide any additional value (i.e. their predictions are never the uniquely correct ones), such members no longer provide valuable information to SEAC. If we allow such members to fall to a performance factor of zero, then high volumes of uniquely-correct timeframes would be required to regain trust. Therefore, an interesting concept exists in replacing members dynamically whenever their performance factor hits zero: in such a case, another model of a different type — geometrical, statistical or logical supervised learning — could replace it, and the corresponding weights would be recalculated. This idea is but a starting point, but interesting nonetheless.

Finally, a change in focus would be an interesting notion to explore. Our system's overall security was the primary focus of our study; hence the selection of the TPR to define the Pf in eq.4.16. However, in a different field of study, the main metric of focus could deviate from user convenience or even another metric. Such an expansion would almost definitely influence the calculation of equation 4.16 in an interesting way.

6.3.3 Concept Adoption in Physical Domains

The final contribution of this thesis is the introduction of utilizing physical-contextual properties in authentication decisions. Let's first discuss how our initial results compared to the provided policy decisions; the difference in scope between the policy (which makes decision uniquely on cyber properties) and our proposed model (which analyzes the cyber-physical domains) creates avenues yet to be explored. For starters, the idea of having additional layers of risk (instead of simply three: *low*, *medium* and *high*) could provide underlying systems with even more flexibility in the access control mechanisms. Furthermore, both the policy and datasets used in this study could be replaced and/or migrated into another context besides the medical industry. Risk is not uniformly quantified across industries, therefore its application in different fields would promptly accelerate the validity of this work.

Regarding the provided data specific to the healthcare industry, data augmentations could enrich this study. This would include (but is not limited to) additional details regarding device types (replacing the concept of "mobile" devices with more precise definitions like cellphones, tablets, E-charts, laptops ...) or additional details regarding the types of locations (augmenting the GPS coordinates with a tag on location; hallways, receptions, emergency rooms and so on, even expanding into indoor-localization would be of interest).

In more technical details, our current proposed consensus model uses three unsupervised methods as a means to create data clusters: DBSCAN, K-mean and GMM. As these models were selected empirically, there is room for improvement in both the selection of type and quantity of models. Assuming the rule of having an odd number of models is respected (as required by the consensus algorithm), interesting results could emerge from varying combinations. These variations could even be expanded to include deep learning technologies. Also, our process of inter-cluster-identification (algorithm 1) currently uses the majority-voting technique, but more efficient methods may exist.

When we consider our formulations, there are also many areas for improvement. The initial cyclic nature of our coupling metrics could be expanded, as to confirm the inter-dependency between their values. Exploring all permutations would render many variables, which could then be run through correlation analysis. Even better, user behaviours could be added to the data. Such additions could include concepts like sedentary and nomadic movement behaviours, as to further simulate the entropy of realistic dynamic environments. Additionally, the concept of user co-existence could be explored; tracking how frequently and quantitatively two individuals spend time together could bring additional insights to the machine learning algorithms. This metric could be expanded to create user-connection-graphs, which based on their level of completeness (not connected, connected or complete) would be associated with their own corresponding risk level. These graphs could be temporal in nature, but are not limited as such, and could also rely on pre-defined thresholds, just as with our engineered metrics. On the topic of thresholds, our risk thresholds which are defined in section 5.3 are broken down into three risk buckets, but this number (three) was simply selected as to adhere with our consensus-model outputs. These thresholds could be increased in further numbers and adapted into a type of weighted-sum. This sum would could then formulate a quantitative risk measurement, instead of attributing risk into categorical values. Such a concept could also apply to the $a1$ and $a2$ thresholds. Representing the variance from the norm, $a1$ and $a2$ (and potentially creating $a3$ and $a4$) all fit the weighted-sum quantification potential just previously mentioned.

Finally, with all these potential improvements the domain and mathematical formulations, there is also room for expansion in the derivation of our risk metrics (section 5.2). As our methods of comparison were specific to the industry and clients we had at the time, the investigation of more precise quantitative metrics would need to be created to transition our consensus model into a different field of study.

References

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys Tutorials*, 17(4):2347–2376, Fourthquarter 2015.

[2] F. Al-Turjman and S. Alturjman. Context-sensitive access in industrial internet of things (iiot) healthcare applications. *IEEE Transactions on Industrial Informatics*, 14(6):2736–2744, June 2018.

[3] C. Alcaraz and J. Lopez. Wide-area situational awareness for critical infrastructure protection. *Computer*, 46(4):30–37, April 2013.

[4] M. S. Ali, M. Vecchio, M. Pincheira, K. Dolui, F. Antonelli, and M. H. Rehmani. Applications of blockchains in the internet of things: A comprehensive survey. *IEEE Communications Surveys Tutorials*, 21(2):1676–1717, Secondquarter 2019.

[5] T. Ali, S. Asghar, and Naseer Ahmed Sajid. Critical analysis of dbscan variations. In *2010 International Conference on Information and Emerging Technologies*, June 2010.

[6] N. Alomar, M. Alsaleh, and A. Alarifi. Social authentication applications, attacks, defense strategies and future research directions: A systematic review. *IEEE Communications Surveys Tutorials*, 19(2):1080–1111, Secondquarter 2017.

[7] F. Anjomshoa, M. Aloqaily, B. Kantarci, M. Erol-Kantarci, and S. Schuckers. Social behaviometrics for personalized devices in the internet of things era. *IEEE Access*, 5:12199–12213, 2017.

[8] F. Anjomshoa, M. Catalfamo, D. Hecker, N. Helgeland, A. Rasch, B. Kantarci, M. Erol-Kantarci, and S. Schuckers. Mobile behaviometric framework for sociability assessment and identification of smartphone users. In *2016 IEEE Symposium on Computers and Communication (ISCC)*, pages 1084–1089, June 2016.

[9] G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti. On the effectiveness of machine and deep learning for cyber security. In *2018 10th International Conference on Cyber Conflict (CyCon)*, pages 371–390, May 2018.

[10] B. S. Archana, A. Chandrashekar, A. G. Bangi, B. M. Sanjana, and S. Akram. Survey on usable and secure two-factor authentication. In *2017 2nd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, pages 842–846, May 2017.

[11] Z. Asaf, M. Asad, S. Ahmed, W. Rasheed, and T. Bashir. Role based access control architectural design issues in large organizations. In *2014 International Conference on Open Source Systems Technologies*, pages 197–205, Dec 2014.

[12] R. Atat, L. Liu, J. Wu, G. Li, C. Ye, and Y. Yang. Big data meet cyber-physical systems: A panoramic survey. *IEEE Access*, 6:73603–73636, 2018.

[13] F. Ayatolahi and J. Karlsson. Sources of variation in error sensitivity measurements, significant or not? In *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, pages 71–72, June 2018.

[14] H. Azwar, M. Murtaz, M. Siddique, and S. Rehman. Intrusion detection in secure network for cybersecurity systems using machine learning and data mining. In *2018 IEEE 5th International Conference on Engineering Technologies and Applied Sciences (ICETAS)*, pages 1–9, Nov 2018.

[15] X. Bellekens, A. Hamilton, P. Seeam, K. Nieradzinska, Q. Franssen, and A. Seeam. Pervasive ehealth services a security and privacy risk awareness survey. In *2016 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (CyberSA)*, pages 1–4, June 2016.

[16] K. Z. Bijon, R. Krishnan, and R. Sandhu. A framework for risk-aware role based access control. In *2013 IEEE Conference on Communications and Network Security (CNS)*, pages 462–469, Oct 2013.

[17] M. A. Bode, S. A. Oluwadare, B. K. Alese, and A. F. Thompson. Risk analysis in cyber situation awareness using bayesian approach. In *2015 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)*, pages 1–12, June 2015.

[18] J. Bonneau, C. Herley, P. C. v. Oorschot, and F. Stajano. The quest to replace passwords: A framework for comparative evaluation of web authentication schemes. In *2012 IEEE Symposium on Security and Privacy*, pages 553–567, May 2012.

[19] H. A. Boyes. Trustworthy cyber-physical systems — a review. In *8th IET International System Safety Conference incorporating the Cyber Security Conference 2013*, pages 1–8, Oct 2013.

[20] A. L. Buczak and E. Guven. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys Tutorials*, 18(2):1153–1176, Secondquarter 2016.

[21] A. Capponi, C. Fiandrino, B. Kantarci, L. Foschini, D. Kliazovich, and P. Bouvry. A survey on mobile crowdsensing systems: Challenges, solutions, and opportunities. *IEEE Communications Surveys Tutorials*, 21(3):2419–2465, thirdquarter 2019.

[22] Giuseppe Cardone, Antonio Corradi, Luca Foschini, and Raffaele Ianniello. Participact: A large-scale crowdsensing platform. *IEEE Transactions on Emerging Topics in Computing*, 4(1):21–32, Jan 2015.

[23] Giuseppe Cardone, Luca Foschini, Paolo Bellavista, Antonio Corradi, Cristian Borcea, Manoop Talasila, and Reza Curtmola. Fostering participaction in smart cities: a geo-social crowdsensing platform. *IEEE Communications Magazine*, 51(6):112–119, 2013.

[24] L. Carin, G. Cybenko, and J. Hughes. Cybersecurity strategies: The queries methodology. *Computer*, 41(8):20–26, Aug 2008.

[25] A. Castiglione, K. Raymond Choo, M. Nappi, and S. Ricciardi. Context aware ubiquitous biometrics in edge of military things. *IEEE Cloud Computing*, 4(6):16–20, November 2017.

[26] M. Cheminod, L. Durante, and A. Valenzano. Review of security issues in industrial networks. *IEEE Transactions on Industrial Informatics*, 9(1):277–293, Feb 2013.

[27] J. P. Cruz, Y. Kaji, and N. Yanai. Rbac-sc: Role-based access control using smart contract. *IEEE Access*, 6:12240–12251, 2018.

[28] R. Deng, G. Xiao, R. Lu, H. Liang, and A. V. Vasilakos. False data injection on state estimation in power systems—attacks, impacts, and defense: A survey. *IEEE Transactions on Industrial Informatics*, 13(2):411–423, April 2017.

[29] N. Desai and M. Gogolla. A catalogue of scenario patterns for validating and verifying model behavior. In *2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, pages 519–523, Sep. 2019.

[30] T. B. Dinh, N. Vo, and G. Medioni. Context tracker: Exploring supporters and distracters in unconstrained environments. In *CVPR 2011*, pages 1177–1184, June 2011.

[31] S. C. Eastwood, V. P. Shmerko, S. N. Yanushkevich, M. Drahansky, and D. O. Gorodnichy. Biometric-enabled authentication machines: A survey of open-set real-world applications. *IEEE Transactions on Human-Machine Systems*, 46(2):231–242, April 2016.

[32] A. El Masri, H. Wechsler, P. Likarish, C. Grayson, C. Pu, D. Al-Arayed, and B. B. Kang. Active authentication using scrolling behaviors. In *2015 6th International Conference on Information and Communication Systems (ICICS)*, pages 257–262, April 2015.

[33] A. E. W. Eldewahi, T. M. H. Sharfi, A. A. Mansor, N. A. F. Mohamed, and S. M. H. Alwahbani. Ssl/tls attacks: Analysis and evaluation. In *2015 International Conference on Computing, Control, Networking, Electronics and Embedded Systems Engineering (ICCNEEE)*, pages 203–208, Sep. 2015.

[34] Wei Feng, Zheng Yan, Hengrun Zhang, Kai Zeng, Yu Xiao, and Y Thomas Hou. A survey on security, privacy, and trust in mobile crowdsourcing. *IEEE Internet of Things Journal*, 5(4):2971–2992, 2017.

[35] P. Feth, R. Adler, and D. Schneider. A context-aware, confidence-disclosing and fail-operational dynamic risk assessment architecture. In *2018 14th European Dependable Computing Conference (EDCC)*, pages 190–194, Sep. 2018.

[36] F. Foroughi and P. Luksch. Observation measures to profile user security behaviour. In *2018 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*, pages 1–6, June 2018.

[37] L. Fridman, S. Weber, R. Greenstadt, and M. Kam. Active authentication on mobile devices via stylometry, application usage, web browsing, and gps location. *IEEE Systems Journal*, 11(2):513–521, June 2017.

[38] H. Gao, C. H. Liu, W. Wang, J. Zhao, Z. Song, X. Su, J. Crowcroft, and K. K. Leung. A survey of incentive mechanisms for participatory sensing. *IEEE Communications Surveys Tutorials*, 17(2):918–943, Secondquarter 2015.

[39] P. Giura and W. Wang. A context-based detection framework for advanced persistent threats. In *2012 International Conference on Cyber Security*, pages 69–74, Dec 2012.

[40] B. Grobauer, T. Walloschek, and E. Stocker. Understanding cloud computing vulnerabilities. *IEEE Security Privacy*, 9(2):50–57, March 2011.

[41] B. Guo, Q. Han, H. Chen, L. Shangguan, Z. Zhou, and Z. Yu. The emergence of visual crowdsensing: Challenges and opportunities. *IEEE Communications Surveys Tutorials*, 19(4):2526–2543, Fourthquarter 2017.

[42] B. Guo, Y. Liu, W. Wu, Z. Yu, and Q. Han. Activecrowd: A framework for optimized multitask allocation in mobile crowdsensing systems. *IEEE Trans on Human-Machine Systems*, PP, 08 2017.

[43] Hadi Habibzadeh, Zhou Qin, Tolga Soyata, and Burak Kantarci. Large-scale distributed dedicated-and non-dedicated smart city sensing systems. *IEEE Sensors Journal*, 17(23):7649–7658, Dec 2017.

[44] G. Han, J. Jiang, C. Zhang, T. Q. Duong, M. Guizani, and G. K. Karagiannidis. A survey on mobile anchor node assisted localization in wireless sensor networks. *IEEE Communications Surveys Tutorials*, 18(3):2220–2243, thirdquarter 2016.

[45] Jun Han, Madhumitha Harishankar, Xiao Wang, Albert Jin Chung, and Patrick Tague. Convoy: Physical context verification for vehicle platoon admission. In *Proceedings of the 18th International Workshop on Mobile Computing Systems and Applications*, HotMobile '17, pages 73–78, New York, NY, USA, 2017. ACM.

[46] L. Heng, D. B. Work, and G. X. Gao. Gps signal authentication from cooperative peers. *IEEE Transactions on Intelligent Transportation Systems*, 16(4):1794–1805, Aug 2015.

[47] G. Hug and J. A. Giampapa. Vulnerability assessment of ac state estimation with respect to false data injection cyber-attacks. *IEEE Transactions on Smart Grid*, 3(3):1362–1370, Sep. 2012.

[48] Emin Huseynov and Jean-Marc Seigneur. Chapter 50 - context-aware multifactor authentication survey. In John R. Vacca, editor, *Computer and Information Security Handbook - Third Edition*, pages 715 – 726. Morgan Kaufmann, Boston, 2017.

[49] Y. Jadeja and K. Modi. Cloud computing - concepts, architecture and challenges. In *2012 International Conference on Computing, Electronics and Electrical Technologies (ICCEET)*, pages 877–880, March 2012.

[50] G. Jagadamba and B. Sathish Babu. Adaptive security schemes based on context and trust for ubiquitous computing environment: A comprehensive survey. *Indian Journal of Science and Technology*, 9(48), 2017.

[51] Anil K. Jain, Karthik Nandakumar, and Arun Ross. 50 years of biometric research: Accomplishments, challenges, and opportunities. *Pattern Recognition Letters*, 79:80 – 105, 2016.

[52] L. Jiang, X. Niu, J. Xu, Y. Wang, Y. Wu, and L. Xu. Time-sensitive and sybil-proof incentive mechanisms for mobile crowdsensing via social network. *IEEE Access*, 6:48156–48168, 2018.

[53] X. Jing, Z. Yan, and W. Pedrycz. Security data collection and data analytics in the internet: A survey. *IEEE Communications Surveys Tutorials*, 21(1):586–618, Firstquarter 2019.

[54] Burak Kantarci and Hussein T Mouftah. Trustworthy sensing for public safety in cloud-centric internet of things. *IEEE Internet of Things Journal*, 1(4):360–368, 2014.

[55] K. M. Khan and Q. Malluhi. Identifying contextual properties of software architecture in cloud computing. In *2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing*, pages 561–568, Dec 2011.

[56] A. Kott and C. Arnold. The promises and challenges of continuous monitoring and risk scoring. *IEEE Security Privacy*, 11(1):90–93, Jan 2013.

[57] D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig. Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1):14–76, Jan 2015.

[58] E. P. Kukula, M. J. Sutton, and S. J. Elliott. The human–biometric-sensor interaction evaluation method: Biometric performance and usability measurements. *IEEE Transactions on Instrumentation and Measurement*, 59(4):784–791, April 2010.

[59] O. D. Lara and M. A. Labrador. A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys Tutorials*, 15(3):1192–1209, Third 2013.

[60] V. Lenders, A. Tanner, and A. Blarer. Gaining an edge in cyberspace with advanced situational awareness. *IEEE Security Privacy*, 13(2):65–74, Mar 2015.

[61] S. Li, S. Zhao, Y. Yuan, Q. Sun, and K. Zhang. Dynamic security risk evaluation via hybrid bayesian risk graph in cyber-physical social systems. *IEEE Transactions on Computational Social Systems*, 5(4):1133–1141, Dec 2018.

[62] J. C. D. Lima, C. C. Rocha, I. Augustin, and M. A. R. Dantas. A context-aware recommendation system to behavioral based authentication in mobile and pervasive environments. In *2011 IFIP 9th International Conference on Embedded and Ubiquitous Computing*, pages 312–319, Oct 2011.

[63] J. Liu, H. Shen, and X. Zhang. A survey of mobile crowdsensing techniques: A critical component for the internet of things. In *ICCCN*, pages 1–6, Aug 2016.

[64] S. Liu and B. Cheng. Cyberattacks: Why, what, who, and how. *IT Professional*, 11(3):14–21, May 2009.

[65] Y. Liu, S. Hu, and T. Ho. Vulnerability assessment and defense technology for smart home cybersecurity considering pricing cyberattacks. In *2014 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 183–190, Nov 2014.

[66] T. Luo, H. Tan, and L. Xia. Profit-maximizing incentive for participatory sensing. In *IEEE INFOCOM 2014*, pages 127–135, April 2014.

[67] Q. Ma, S. Zhang, T. Zhu, K. Liu, L. Zhang, W. He, and Y. Liu. Plp: Protecting location privacy against correlation analyze attack in crowdsensing. *IEEE Trans. on Mobile Computing*, 2017.

[68] P. Mach and Z. Becvar. Mobile edge computing: A survey on architecture and computation offloading. *IEEE Communications Surveys Tutorials*, 19(3):1628–1656, thirdquarter 2017.

[69] T. Mahmood and U. Afzal. Security analytics: Big data analytics for cybersecurity: A review of trends, techniques and tools. In *2013 2nd National Conference on Information Assurance (NCIA)*, pages 129–134, Dec 2013.

[70] A. Mansour, M. Sadik, E. Sabir, and M. Azmi. A context-aware multimodal biometric authentication for cloud-empowered systems. In *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, pages 278–285, Oct 2016.

[71] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief. A survey on mobile edge computing: The communication perspective. *IEEE Communications Surveys Tutorials*, 19(4):2322–2358, Fourthquarter 2017.

[72] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2929–2936, June 2009.

[73] P. McDaniel, N. Papernot, and Z. B. Celik. Machine learning in adversarial settings. *IEEE Security Privacy*, 14(3):68–72, May 2016.

[74] W. Meng, D. S. Wong, S. Furnell, and J. Zhou. Surveying the development of biometric user authentication on mobile phones. *IEEE Communications Surveys Tutorials*, 17(3):1268–1293, thirdquarter 2015.

[75] M. Mesgarpour, T. Chaussalet, and S. Chahed. Risk modelling framework for emergency hospital readmission, using hospital episode statistics inpatient data. In *2016 IEEE 29th International Symposium on Computer-Based Medical Systems (CBMS)*, pages 219–224, June 2016.

[76] Markus Miettinen, N. Asokan, Thien Duc Nguyen, Ahmad-Reza Sadeghi, and Majid Sobhani. Context-based zero-interaction pairing and key evolution for advanced personal devices. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, pages 880–891, New York, NY, USA, 2014. ACM.

[77] Markus Miettinen, Thien Duc Nguyen, Ahmad-Reza Sadeghi, and N. Asokan. Revisiting context-based authentication in iot. In *Proceedings of the 55th Annual Design Automation Conference*, DAC '18, pages 32:1–32:6, New York, NY, USA, 2018. ACM.

[78] Jianbing Ni, Aiqing Zhang, Xiaodong Lin, and Xuemin Sherman Shen. Security, privacy, and fairness in fog-based vehicular crowdsensing. *IEEE Communications Magazine*, 55(6):146–152, 2017.

[79] V. Niemi. Privacy, identity and trust in context-aware mobile services. In *2011IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications*, pages 9–10, Nov 2011.

[80] J. R. C. Nurse, S. Creese, M. Goldsmith, and K. Lamberts. Guidelines for usable cybersecurity: Past and present. In *2011 Third International Workshop on Cyberspace Safety and Security (CSS)*, pages 21–26, Sep. 2011.

[81] J. R. C. Nurse, S. Creese, M. Goldsmith, and K. Lamberts. Trustworthy and effective communication of cybersecurity risks: A review. In *2011 1st Workshop on Socio-Technical Aspects in Security and Trust (STAST)*, pages 60–68, Sep. 2011.

[82] D. E. O'Leary. Ethics for big data and analytics. *IEEE Intelligent Systems*, 31(4):81–84, July 2016.

[83] Sharief MA Oteafy and Hossam S Hassanein. Big sensed data: Evolution, challenges, and a progressive framework. *IEEE Communications Magazine*, 56(7):108–114, July 2018.

[84] S. Otoum, B. Kantarci, and H. Mouftah. Adaptively supervised and intrusion-aware data aggregation for wireless sensor clusters in critical infrastructures. In *2018 IEEE International Conference on Communications (ICC)*, pages 1–6, May 2018.

[85] M. Pajic, J. Weimer, N. Bezzo, O. Sokolsky, G. J. Pappas, and I. Lee. Design and implementation of attack-resilient cyberphysical systems: With a focus on attack-resilient state estimators. *IEEE Control Systems Magazine*, 37(2):66–81, April 2017.

[86] V. M. Patel, R. Chellappa, D. Chandra, and B. Barbello. Continuous user authentication on mobile devices: Recent progress and remaining challenges. *IEEE Signal Processing Magazine*, 33(4):49–61, July 2016.

[87] K. Pelechrinis, M. Iliofotou, and S. V. Krishnamurthy. Denial of service attacks in wireless networks: The case of jammers. *IEEE Communications Surveys Tutorials*, 13(2):245–257, Second 2011.

[88] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos. Context aware computing for the internet of things: A survey. *IEEE Communications Surveys Tutorials*, 16(1):414–454, First 2014.

[89] L. Pietre-Cambacedes, M. Tritschler, and G. N. Ericsson. Cybersecurity myths on power control systems: 21 misconceptions and false beliefs. *IEEE Transactions on Power Delivery*, 26(1):161–172, Jan 2011.

[90] M. Pouryazdan, C. Fiandrino, and B. Kantarci. Intelligent gaming for mobile crowdsensing participants to acquire trustworthy big data in the internet of things. *IEEE Access*, 5:22209–22223, 2017.

[91] Maryam Pouryazdan and Burak Kantarci. The smart citizen factor in trustworthy smart city crowdsensing. *IT Professional*, 18(4):26–33, July 2016.

[92] Davy Preuveneers and Wouter Joosen. Smartauth: Dynamic context fingerprinting for continuous user authentication. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, SAC '15, pages 2185–2191, New York, NY, USA, 2015. ACM.

[93] E. Pricop and S. F. Mihalache. Fuzzy approach on modelling cyber attacks patterns on data transfer in industrial control systems. In *2015 7th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pages SSS–23–SSS–28, June 2015.

[94] K. Quintal, B. Kantarci, M. Erol-Kantarci, A. Malton, and A. Walenstein. Contextual, behavioral and biometric signatures for continuous authentication. *IEEE Internet Computing*, pages 1–1, 2019.

[95] K. Quintal, E. Kara, M. Simsek, B. Kantarci, and H. Viktor. Sensory data-driven modeling of adversaries in mobile crowdsensing platforms. In *2019 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, Dec 2019.

[96] M. Rathi and T. Chaussalet. Risk prediction model using fuzzy regression method for predicting unplanned hospital admissions. In *2013 IEEE International Systems Conference (SysCon)*, pages 595–598, April 2013.

[97] Zachary I. Rauen, Fazel Anjomshoa, and Burak Kantarci. Gesture and sociability-based continuous authentication on smart mobile devices. In *ACM Intl. Symp. on Mobility Management and Wireless Access*, pages 51–58. ACM, 2018.

[98] N. Ruan, L. Gao, H. Zhu, W. Jia, X. Li, and Q. Hu. Toward optimal dos-resistant authentication in crowdsensing networks via evolutionary game. In *IEEE ICDCS 2016*, pages 364–373, June 2016.

[99] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. Role-based access control models. *Computer*, 29(2):38–47, Feb 1996.

[100] I. H. Sarker, M. A. Kabir, A. Colman, and J. Han. An approach to modeling call response behavior on mobile phones based on multi-dimensional contexts. In *2017 IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft)*, pages 91–95, May 2017.

[101] M. Savvides, B. V. K. Vijaya Kumar, and P. K. Khosla. Cancelable biometric filters for face recognition. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 3, pages 922–925 Vol.3, Aug 2004.

[102] K. N. Sevis and E. Seker. Cyber warfare: terms, issues, laws and controversies. In *2016 International Conference On Cyber Security And Protection Of Digital Services (Cyber Security)*, pages 1–9, June 2016.

[103] M. Shahzad and M. P. Singh. Continuous authentication and authorization for the internet of things. *IEEE Internet Computing*, 21(2):86–90, Mar 2017.

[104] Y. Shi, Q. Zhang, J. Liang, Z. He, and H. Fan. Obfuscatable anonymous authentication scheme for mobile crowd sensing. *IEEE Systems Journal*, 13(3):2918–2929, Sep. 2019.

[105] A. Solanas, C. Patsakis, M. Conti, I. S. Vlachos, V. Ramos, F. Falcone, O. Postolache, P. A. Perez-martinez, R. D. Pietro, D. N. Perrea, and A. Martinez-Balleste. Smart health: A context-aware health paradigm within smart cities. *IEEE Communications Magazine*, 52(8):74–81, Aug 2014.

[106] Yunchuan Sun, Houbing Song, Antonio J Jara, and Rongfang Bie. Internet of things and big data analytics for smart and connected communities. *IEEE access*, 4:766–773, 2016.

[107] T. Taniguchi, S. Nagasaka, K. Hitomi, K. Takenaka, and T. Bando. Unsupervised hierarchical modeling of driving behavior and prediction of contextual changing points. *IEEE Transactions on Intelligent Transportation Systems*, 16(4):1746–1760, Aug 2015.

[108] P. Tasatanattakool and C. Techapanupreeda. User authentication algorithm with role-based access control for electronic health systems to prevent abuse of patient privacy. In *2017 3rd IEEE International Conference on Computer and Communications (ICCC)*, pages 1019–1024, Dec 2017.

[109] R. R. Tenney and N. R. Sandell. Detection with distributed sensors. *IEEE Transactions on Aerospace and Electronic Systems*, AES-17(4):501–510, July 1981.

[110] Y. Tong, J. Sun, S. S. M. Chow, and P. Li. Cloud-assisted mobile-access of health data with privacy and auditability. *IEEE Journal of Biomedical and Health Informatics*, 18(2):419–429, March 2014.

[111] K. Wang, X. Qi, L. Shu, D. Deng, and J. J. P. C. Rodrigues. Toward trustworthy crowdsourcing in the social internet of things. *IEEE Wireless Communications*, 23(5):30–36, October 2016.

[112] Wei Liu, Xue Li, and Daoli Huang. A survey on context awareness. In *2011 International Conference on Computer Science and Service System (CSSS)*, pages 144–147, June 2011.

[113] H. Witte, C. Rathgeb, and C. Busch. Context-aware mobile biometric authentication based on support vector machines. In *2013 Fourth International Conference on Emerging Security Technologies*, pages 29–32, Sept 2013.

[114] D. Wu and D. D. Wu. Risk-based robust evaluation of hospital efficiency. *IEEE Systems Journal*, 13(2):1906–1914, June 2019.

[115] L. Xiao, Y. Li, G. Han, H. Dai, and H. V. Poor. A secure mobile crowdsensing game with deep reinforcement learning. *IEEE Transactions on Information Forensics and Security*, 13(1):35–47, Jan 2018.

[116] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang. Machine learning and deep learning methods for cybersecurity. *IEEE Access*, 6:35365–35381, 2018.

[117] D. Yang, G. Xue, X. Fang, and J. Tang. Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones. *IEEE/ACM Transactions on Networking*, 24(3):1732–1744, June 2016.

[118] S. Yu, C. Wang, K. Ren, and W. Lou. Achieving secure, scalable, and fine-grained data access control in cloud computing. In *2010 Proceedings IEEE INFOCOM*, pages 1–9, March 2010.

[119] Ö. Yürür, C. H. Liu, Z. Sheng, V. C. M. Leung, W. Moreno, and K. K. Leung. Context-awareness for mobile sensing: A survey and future directions. *IEEE Communications Surveys Tutorials*, 18(1):68–93, Firstquarter 2016.

[120] X. Zhang, Z. Yang, W. Sun, Y. Liu, S. Tang, K. Xing, and X. Mao. Incentives for mobile crowd sensing: A survey. *IEEE Communications Surveys Tutorials*, 18(1):54–67, Firstquarter 2016.

[121] Y. Zhang and B. Kantarci. Invited paper: Ai-based security design of mobile crowdsensing systems: Review, challenges and case studies. In *2019 IEEE International Conference on Service-Oriented System Engineering (SOSE)*, pages 17–1709, April 2019.

[122] P. Zhao, J. Li, F. Zeng, F. Xiao, C. Wang, and H. Jiang. Illia: Enabling k -anonymity-based privacy preserving against location injection attacks in continuous lbs queries. *IEEE Internet of Things Journal*, 5(2):1033–1042, April 2018.

[123] L. Zhou, V. Varadharajan, and M. Hitchens. Achieving secure role-based access control on encrypted data in cloud storage. *IEEE Transactions on Information Forensics and Security*, 8(12):1947–1960, Dec 2013.

[124] Y. Zhu, D. Huang, C. Hu, and X. Wang. From rbac to abac: Constructing flexible data access control for cloud storage services. *IEEE Transactions on Services Computing*, 8(4):601–616, July 2015.

Appendices

Appendix A

Principal Component Aanalysis (PCA) of mobile sensors

Data collection which occurred in chapter 3 produced a dataset of approximately 70k entries originating from 28 built-in mobile devices sensors of students on a university campus. However, not all the collected information provides additional value to our analysis. Hence why we conduct a statistical analysis on the overall dataset called Principal Component Analysis (PCA).

PCA aims to reduce the dimensionality of datasets with large sets of features. This is conducted by standardizing each feature, computing a covariance matrix to obtain Eigenvalues and Eigenvectors (which are linear algebra concepts) to determine the principal components of the dataset. These principal components are essentially new variables created from the combination of our standardized features, Eigenvalues and Eigenvectors.

Each component covers a certain percentage of the data variance, and aren't as insightful on their own. However, knowing the ratio of the original features within each principal component does help us determine which features capture the most information of our data, i.e. cover the most data variance. As each principal component is uncorrelated with one another, we can look at the eigenvalues of the features that compose the principal components and discard those which lesser significance. As we were aiming for 90% data variance, we selected our 15 built-in sensor values as seen in table A.1.

Table A.1: Features contributing to data variance - All Users

Ranking	**Built-in Sensor**	**PCA variance**	**Variance - Cumulative**
1	Orientation azimuth	8.2%	8.2%
2	Accelerometer x	8.0%	16.2%
3	Latitude	7.6%	23.8%
4	Longitude	7.6%	31.4%
5	Proximity	6.8%	38.2%
6	Wifi usage	6.3%	44.5%
7	Accelerometer Z	6.1%	50.6%
8	Accelerometer Y	6.0%	56.6%
9	Battery Consumption	5.8%	62.4%
10	Client ID	5.5%	67.9%
11	Orientation roll	4.9%	73.8%
12	Day of week	4.8%	78.6%
13	Microphone intensity	4.7%	83.3%
14	Orientation pitch	4.5%	87.8%
15	Battery level	4.2%	**92.0**%
16	Time of day	3.8%	95.8%
17	Ambient light	2.0%	97.8%
18	Data usage	1.7%	99.5%
19	Minute	0.2%	99.7%
20	Battery consumption	$< 0.1\%$	99.8%
21	Ambient temperature	$< 0.1\%$	99.8%
22	Gyroscope X	$< 0.1\%$	99.8%
23	Gyroscope Y	$< 0.1\%$	99.9%
24	Gyroscope Z	$< 0.1\%$	99.9%
25	Humidity	$< 0.1\%$	99.9%
26	Ambient light	$< 0.1\%$	99.9%
27	Microphone pitch	$< 0.1\%$	99.9%
28	Bluetooth pairings	$< 0.1\%$	100%

Appendix B

DBSCAN implementation and details

The density-based spatial clustering of applications with noise (DBSCAN) is a data clustering technique that can create arbitrarily-shaped clusters of data, based on the geometrical proximity of points from one another. When points (data) are clustered, they are either inserted into an existing cluster or inserted as an outlier (noise). This process depends on two parameters:

- epsilon: the minimal distance to another point to consider these two points neighbours.
- minPts: the number of neighbours near one another to create a cluster

The selection of these values was conducted through a process called *hyperparameter tuning*. To oversimplify, this process attempts to iterate through the subset of the possible parameter values to find the most adequate values based on the provided data. Combined with manual testing, this process is how the parameters of DBSCAN were selected for each user's behavioural profile (described in section 3.3.2). The parameters of $\epsilon = 0.4$ and the $mintPts = 50$ lead to reasonable accuracy results.

Therefore, in the context of our study, data points represent the behavioural profiles of participants across a university campus. Clusters represent common behavioural patterns (such as always frequenting the same coffee stand, or attending the same lecture each week). Data anomalies occur when participants venture outside of their habitual behaviour unless these ventures occur frequently enough to create a cluster of its own.