Anomaly Detection and its Adaptation: Studies on Cyber-Physical Systems

by

Massimiliano Raciti
Anomaly Detection and its Adaptation: Studies on Cyber-Physical Systems

by

Massimiliano Raciti

March 2013
Linköping Studies in Science and Technology
Licentiate Thesis No. 1586
ISSN 0280–7971
LiU–Tek–Lic–2013:20

ABSTRACT

Cyber-Physical Systems (CPS) are complex systems where physical operations are supported and coordinated by Information and Communication Technology (ICT). From the point of view of security, ICT technology offers new opportunities to increase vigilance and real-time responsiveness to physical security faults. On the other hand, the cyber domain carries all the security vulnerabilities typical to information systems, making security a new big challenge in critical systems.

This thesis addresses anomaly detection as security measure in CPS. Anomaly detection consists of modelling the good behaviour of a system using machine learning and data mining algorithms, detecting anomalies when deviations from the normality model occur at runtime. Its main feature is the ability to discover the kinds of attack not seen before, making it suitable as a second line of defence.

The first contribution of this thesis addresses the application of anomaly detection as early warning system in water management systems. We describe the evaluation of an anomaly detection software when integrated in a Supervisory Control and Data Acquisition (SCADA) system where water quality sensors provide data for real-time analysis and detection of contaminants. Then, we focus our attention to smart metering infrastructures. We study a smart metering device that uses a trusted platform for storage and communication of electricity metering data, and show that despite the hard core security, there is still room for deployment of a second level of defence as an embedded real-time anomaly detector that can cover both the cyber and physical domains. In both scenarios, we show that anomaly detection algorithms can efficiently discover attacks in the form of contamination events in the first case and cyber attacks for electricity theft in the second.

The second contribution focuses on online adaptation of the parameters of anomaly detection applied to a Mobile Ad hoc Network (MANET) for disaster response. Since survivability of the communication to network attacks is as crucial as the lifetime of the network itself, we devised a component that is in charge of adjusting the parameters based on the current energy level, using the trade-off between the node’s response to attacks and the energy consumption induced by the intrusion detection system. Adaption increases the network lifetime without significantly deteriorating the detection performance.

This work has been supported by the Swedish National Graduate School of Computer Science (CUGS) and the EU FP7 SecFutur Project.
Acknowledgments

When I came to Sweden for the first time I would have never imagined what was going to happen to me. The time I spent here writing my Master’s thesis in 2009 was so exciting that when I was done with it I had the feeling that six months at LiU were definitely not enough. Once I got the opportunity, I headed back to Sweden for a new, longer adventure.

My first gratitude goes to my supervisor Simin Nadjm-Tehrani. Thank you for accepting me as PhD student and guiding me through the years. Your never-ending energy and optimism have an impact on our inspiration and motivation. Thank you for your advice in teaching and research.

A special thank goes to my current and former colleagues at RTSLab for your valuable feedback and for making the environment more than just a workplace. We have slowly polluted the swedish habits with mediterranean standards, it was interesting to see even the most swedish-style fellows knocking on our doors at 6pm for a fika. Thank you for the nice time we have spent together inside and outside the lab.

I also want to thank Anne, Eva, Åsa and Inger for the administrative support, and all the other colleagues at IDA for making the environment so enjoyable.

While pursuing PhD studies one inevitably has to encounter hills and valleys, sometimes your mood reaches the peak and sometimes it slopes down to the ground. I would like to express my gratitude to my family who has always been there to support me. I dedicate this thesis to all of you.

Last but not least, I would like to thank my dear Simona for lighting up my heart and giving me your full support in the last stages of this work. I love you with all my heart. Finally, an acknowledgement goes to all my friends around the world for the pleasant time spent together in Linköping.

Massimiliano Raciti
Linköping
March 2013
# Contents

1 **Introduction**  
1.1 Motivation .......................................................... 2  
1.2 Problem formulation ................................................ 4  
1.3 Contributions ...................................................... 4  
1.4 Thesis outline ...................................................... 5  
1.5 List of publications ................................................. 6  

2 **Background**  
2.1 Anomaly Detection with ADWICE ................................. 7  
2.2 Water Quality in Distribution Systems ......................... 9  
2.2.1 Security considerations ........................................ 11  
2.2.2 Related Work on Contamination Detection ................ 11  
2.3 Advanced Metering Infrastructures .............................. 15  
2.3.1 Security considerations ........................................ 16  
2.3.2 Related Work on AMI Security ............................... 17  
2.4 Disaster Area Networks ............................................ 18  
2.4.1 Random-Walk Gossip protocol ............................... 19  
2.4.2 Security considerations ........................................ 20  

3 **Anomaly Detection in Water Management Systems** ............ 21  
3.1 Scenario: the event detection systems challenge ............. 21  
3.2 Modelling Normality ............................................... 24  
3.2.1 Training ......................................................... 24  
3.2.2 Feature selection ............................................... 24  
3.2.3 Challenges ..................................................... 25  
3.3 Detection Results .................................................. 26  
3.3.1 Station A .................................................... 26  
3.3.2 Station D .................................................... 28  
3.4 Detection Latency .................................................. 30  
3.5 Missing and inaccurate data problem ............................ 32  
3.6 Closing remarks ................................................... 33
List of Figures

2.1 NIST smart grid interoperability model [1] ........................................ 15
3.1 Effect of contaminants on the WQ parameters ................................. 22
3.2 Example of Event Profiles .......................................................... 23
3.3 Station A contaminant A ROC curve .............................................. 28
3.4 Concentration Sensitivity of the Station A ........................................ 28
3.5 ROC curve station D contaminant A ................................................. 29
3.6 Concentration sensitivity station D ................................................ 30
3.7 Detection latency in station A ..................................................... 31
3.8 Detection latency in station D ..................................................... 31
4.1 Trusted Smart Metering Infrastructure [2] ......................................... 35
4.2 Trusted Meter ............................................................................ 36
4.3 Manipulation of consumption values .............................................. 36
4.4 Manipulation or injection of control commands ................................. 37
4.5 Proposed Cyber-Physical Anomaly Detection Architecture ................. 38
5.1 The General Survivability Framework control loop .......................... 46
5.2 Energy-aware adaptation of IDS parameters .................................. 47
5.3 State machine with associated power consumption .......................... 50
5.4 CPU power consumption of the RWG and IDS framework over some time interval ................................................................. 51
5.5 Energy consumption and number of active nodes in the drain attack .......................... 53
5.6 Survivability performance in the draining attack .............................. 54
5.7 Energy consumption and number of active nodes in the grey hole attack ................................................................. 55
5.8 Survivability performance in the grey hole attack ........................... 55
5.9 Energy consumption and number of active nodes in the drain attack with random initial energy levels .................................................. 56
5.10 Survivability performance in the drain attack with random initial energy levels ................................................................. 56
List of Tables

3.1 Station A detection results of 1mg/L of concentration ........ 27
5.1 Aggregation interval based on energy levels ................. 48
5.2 Current draw of the RWG application ...................... 52
5.3 Current draw of the IDS application ...................... 52
Chapter 1

Introduction

Since computer-based systems are now pervading all aspects of our everyday life, security is one of the main concerns in our digital era. New systems are often poorly understood at the beginning from the security point of view. Unprotected assets can be exploited by attackers, who can target the vulnerabilities of the systems to get some kind of benefit out of it. In addition to that, security is often the less developed attribute at design time [3]. Designers tend to focus and optimise other functional and non-function requirements rather than security, which is typically added afterwards. This is the case, for instance, for smartphone security, where the proliferation of malware that exploits weaknesses of architectures and protocols has raised the need of efficient detection techniques and this is a hot topic at the time of this thesis.

Old and well understood systems, on the other hand, still expose vulnerabilities that can be discovered later after their deployment and security is typically a never-ending challenge.

Security mechanisms can be classified as active and passive. Active security mechanisms are normally proactive, meaning that the asset they protect are secured in advance. Data encryption, for instance, is one of the main active security mechanisms. Passive security mechanisms, on the other hand, do not perform any action until the attack occurs.

Intrusion detection, the main mechanism of this category, consists of passively monitoring the system in order to detect attacks and, eventually, apply the opportune countermeasure. There are two subcategories of intrusion detection that differ on the way they discover attacks. Misuse detection, the most common in commercial tools, consists on creating models (often called signatures) of the attacks. When the current condition matches any known signatures, an alarm is raised. Misuse detection requires exact characterisation of known attacks based on historical data set and gives accurate matches for those cases that are modelled. While misuse detection provides immediate diagnosis when successful, it is unable to detect cases for which no previous information exists (earlier similar cases in history, a known signature, etc.).
1. INTRODUCTION

Anomaly detection, the complementary technique of misuse detection, is a paradigm in which what is modelled is an approximate normality of a system, using machine learning or data mining techniques. This requires a learning phase during which the normality model is built using historical observations. At runtime, anomalous conditions are detected when observations of the current state are sufficiently different from the normality model. As opposed to misuse detection, anomaly detection is able to uncover new attacks not seen earlier, since it does not include any knowledge about them. This feature makes it suitable both in environments where vulnerabilities are not known in advance and as a second line of defence in an operational system when new vulnerabilities are discovered. Its main limitation is however related to the construction of a good model of normality. A typical problem that occurs when applying anomaly detection algorithms is the high rate of false alarms if attacks and normality data are similar in subtle ways, and when normality is subject to natural changes over time, referred in the literature as concept drift [4]. The availability of a sufficiently large number of labelled instances that can constitute the basis for a training dataset is also a challenge in some domains.

A model of normality often includes a number of parameters that need to be tuned to fit the training data. These typically determine a tradeoff between a number of factors such as the desired detection accuracy, tolerable false alarms rate, resource utilisation etc. The parameters are often set statically by the system manager. Recently the focus has also been on adaptation approaches devised to make anomaly detection completely autonomic [5].

Anomaly detection has been applied to a range of applications including network or host-based intrusion detection, fraud detection, medical and public health anomaly detection [6]. It is now being considered as prominent technique to discover attacks in new domains, such as smartphone malware detection [7, 8]. This thesis addresses the more general application of anomaly detection techniques to protect cyber-physical systems, as motivated in the next section.

1.1 Motivation

Critical infrastructure systems, such as electricity, gas and water distribution systems have been subject to changes in the last decades. The need for distributed monitoring and control to support their operations has fuelled the practice of integrating information and communication technology to physical systems. Supervisory Control and Data Acquisition (SCADA) systems have been the first approach to distributed monitoring and control by means of information technology. From a larger perspective, this integration has led to the term "Cyber-Physical System" (CPS), where physical processes, computation and information exchange are coupled together to provide improved efficiency, functionality and reliability. The inherently distributed nature of production and distribution and the incorporation of mass scale sensors and faster management dynamics, and fine-grained adaptability to local failures and overloads are the means to achieve this. The notion of cyber-physical systems, aiming to cover the "virtually global and locally physical" [9] is
1.1. Motivation

often used to encompass smart grids as an illustrating example of such complex distributed sensor and actuator networks aimed to control a physical domain. Pervasive computing, on the other hand, has generated a subcategory of cyber-physical systems, the Mobile Cyber-Physical Systems (MCPS) [10]. Using the advances of development on Wireless Sensor Networks (WSN) and Mobile Ad hoc Networks (MANETs) mobile devices, such as smartphones, are prominent tools who allow applications that are based on the interaction between the physical environment and the cyber world. Having adequate resources to perform local task processing, storage, networking and sensing (camera, GPS, microphone, light sensors etc.), a number of application of MCPS have been implemented including healthcare, intelligent transportation, navigation and disaster response [9].

Security in critical systems has historically been an important matter of concern, even when the cyber domain was not present. Attacks to the physical domain can have severe impacts on society and can have disastrous consequences. In the past, most of the security mechanisms were implemented using physical protection. Critical assets were typically located in controlled environments, and this often prevented the occurrence of undesired manipulations.

In some cases, however, physical protection is not always fully applicable. Water management systems, for example, deserve a special attention in critical infrastructure protection. In contrast to some other infrastructures where the physical access to the critical assets may be possible to restrict, in water management systems there is a large number of remote access points difficult to control and protect from accidental or intentional contamination events. Since quality of distributed water affects every single citizen with obvious health hazards, in the event of contamination, there are few defence mechanisms available. Water treatment facilities are typically the sole barrier to potential large scale contaminations and distributed containment of the event leads to widespread water shortages. Today’s SCADA systems provide a natural opportunity to increase vigilance against water contaminations. Specialised event detection mechanisms for water management could be included such that 1) a contamination event is detected as early as possible and with high accuracy with few false positives, and 2) predictive capabilities ease preparedness actions in advance of full scale contamination in a utility. The available data from water management system sensors are based on biological, chemical and physical features of the environment and the water sources. Since these change over seasons, the normality model is rather complex. Also, it is hard to create a static set of rules or constraints that clearly capture all significant attacks since these can affect various features in non-trivial ways and we get a combinatorial problem. This suggests that learning based anomaly detection techniques should be explored as a basis for contamination event detection in water management systems.

Power distribution networks are less vulnerable to physical attacks, since their assets can be protected by access restriction more easily. In this context, however, physical attacks typically target the end point devices, the electricity meters. Electricity fraud by meter tampering is a known problem, especially in developing countries [11]. In this context, anomaly detection algorithms for electricity fraud detection are employed to detect considerable deviations of user profiles from av-


1. INTRODUCTION

erage profiles of customers belonging to the same class. To summarise, ICT technology for real-time monitoring and control has a potential to the physical layer in various domains of CPS.

Cyber security, on the other hand, is a new issue in cyber-physical critical systems. While security is indeed part of the grand challenges facing large scale development of cyber-physical systems, the focus has initially been to threats to control systems [12]. The ICT itself suffers from vulnerability to attacks, and can bring with it traditional security threats to ICT network. Since sensitive information is exchanged through the network, integrity, authenticity, accountability and confidentiality are new fundamental security requirements in cyber-physical systems. Smart grids, once again, are the typical example of CPS where cyber security has become an important matter of concern [13]. Although active security mechanisms have been extensively designed, studies [14] show that there is still room for anomaly detection as second line of defence.

1.2 Problem formulation

This thesis addresses anomaly detection in the context of cyber-physical systems. The hypothesis is that anomaly detection can be adopted for cyber-physical systems security in both the cyber and physical domain. In the physical domain, anomaly detection should be explored in order to detect events that can be attributed to malicious activity. The challenge is to provide reliable alerts with few false positives and low latency. In the cyber domain, anomaly detection should be explored to discover attacks on the communication networks.

We then proceed to study the problem of automatic parameter adaptation in intrusion detection. We focus our attention to wireless communications where the more complicated dynamics (mobility and network topology), the constrained resources (battery, bandwidth) and the frequent miscommunications make automatic adaptation a desired property in anomaly detection.

1.3 Contributions

The contributions of this thesis are twofold; the study of the anomaly detection for critical infrastructure protection and the study of adaptive strategies to adjust the parameters of an intrusion detection architecture during runtime. Our studies have been performed in three different domains related to cyber-physical systems: two critical infrastructures, such as water management systems and smart grids and a disaster response scenario. More specifically:

1. Anomaly detection as event detection system in water management systems

In the first contribution, we apply a method for Anomaly Detection With fast Incremental ClustErng (ADWISE) [15] in a water management system for water contamination detection. We analyse the performance of the approach
1.4 Thesis outline

The thesis is organised as follows. Chapter 2 presents background information on the anomaly detection technique adopted in this thesis and the considered domains. The chapter gives first an overview of ADWICE, the anomaly detection algorithm used as reference algorithm in our studies, then presents and discusses the security issues on the two domains in the forthcoming chapters: Water Management Systems and Smart Metering Infrastructures. The chapter is concluded with an introduction to the mobile ad-hoc communication security framework in which an approach to adaptation has been proposed. Chapter 3 presents the evaluation of ADWICE when implemented as an event detector system for water quality anomalies. Chapter 4 analyses the design of a smart metering infrastructure, proposing an architecture for embedded anomaly detection in smart meters and evaluates the detection performance on cyber attacks performed on a smart meter prototype. In Chapter 5 the energy-based security adaptation approach in mobile ad hoc networks is described and evaluated. Finally, Chapter 6 concludes the work and presents our future work.

on real data using metrics such as detection rate, false positives, detection latency, and sensitivity to the contamination level of the attacks, discussing of reliability of the analysis when data sets are not perfect (as seen in real life scenarios), where data values may be missing or less accurate as indicated by sensor alerts.

2. Anomaly detection in smart meters

The second contribution focuses on a smart metering infrastructure which uses trusted computing technology to enforce strong security requirements, and we show the existence of weaknesses in the forthcoming end-nodes that justify embedded real-time anomaly detection. We propose an architecture for embedded anomaly detection for both the cyber and physical domains in smart meters and create an instance of a clustering-based anomaly detection algorithm in a prototype under industrial development, illustrating the detection of cyber attacks in pseudo-real settings.

3. Adaptive Intrusion Detection in disaster area networks

In this contribution we present the impact of energy-aware parameter adaptation of an intrusion detection framework earlier devised for disaster area scenarios, built on top of an energy-efficient message dissemination protocol. We show that adaptation provides extended life time of the network despite attack-induced energy drain and protocol/intrusion detection system overhead. We demonstrate that evaluation of energy-aware adaptation can be based on fairly simple models of CPU utilisation applied to networking protocols in simulation platforms, thus enabling evaluations of communication scenarios that are hard to evaluate by large scale deployments.

1.4 Thesis outline
1. INTRODUCTION

1.5 List of publications

The work presented in this thesis is the result of the following publications:


The following publications are peripheral to the work presented in this thesis and are not part of its content:


- J. Sigholm and M. Raciti, *Best-Effort Data Leakage Prevention in Inter-Organizational Tactical MANETs*, in Military Communications Conference 2012 - MILCOM 2012, IEEE, October 2012
Chapter 2

Background

This chapter provides background information to introduce the reader to the algorithms and domains where anomaly detection and its parameter adaptation have been applied. In the first section, we describe ADWICE, an instance of a clustering-based anomaly detection algorithm earlier developed for securing IP networks. We present the main mechanisms of the algorithm and describe the main features that have been considered while selecting the approach to anomaly detection suitable to our studies on security in critical systems. Next, we introduce the water management system and the smart metering infrastructure, the two domains where the algorithm has been applied, first as event- and then as intrusion-detection system. We give an introduction to these domains and present their security issues and related work on security. Finally, we present the context of disaster area networks, describing a protocol for which a comprehensive security framework has been devised into which our adaptation module is integrated.

2.1 Anomaly Detection with ADWICE

ADWICE (Anomaly Detection With fast Incremental Clustering) \cite{16} is a clustering based anomaly detector that has been developed in an earlier project targeting infrastructure protection. Originally designed to detect anomalies on network traffic sessions using features derived from TCP or UDP packets, ADWICE represents the collection of features as multidimensional numeric vectors, in which each dimension represents a feature. Thus, vectors are therefore data points in the multidimensional features space. Similar observations (i.e. data points that, using a certain distance metric, are close to each other) can be grouped together to form clusters. The basic idea of the algorithm is then to model normality as a set of clusters that summarise all the observed normal behaviour during the learning process. ADWICE assumes semi-supervised learning, where only the data instances provided to represent the normality model are labelled and assumed not to be affected by malicious activity. In the detection phase, if the new data point is close enough (using a threshold) to any normality clusters, it can be classified as an observation
2. BACKGROUND

of normal behaviour, otherwise it is classified as an outlier.

In ADWICE, each cluster is represented through a summary denoted Cluster Feature (CF). CF is a data structure that has three fields \( CF_i = (n, S, SS) \), where
\( n \) is the number of points in the cluster, \( (S) \) is the sum of the points and \( (SS) \) is the square sum of the points in the cluster. The first two elements can be used to compute the average for the points in the cluster used to represent its centroid
\[ v_0 = \frac{\sum_{i=1}^{n} v_i}{n}. \]

The third element, the sum of points, can be used to calculate how large is a circle that would cover all the points in the cluster, through the radius
\[ R(CF) = \sqrt{\frac{\sum_{i=1}^{n} (v_i - v_0)^2}{n}}. \]

With all this information one can measure how far is a new data point from the centre of the cluster (as euclidean distance between the cluster centroid and the new point) and whether the new point falls within or nearby the radius of the cluster. This is used for both building up the normality model (is the new point close enough to any existing clusters so it can become part of it or should it form a new cluster?), and during detection (is the new point close enough to any normality clusters or is it an outlier?).

Using the above structure, during the training phase, a new point can be easily included into a cluster and two clusters \( CF_i = (n_i, S_i, (SS)_i) \) and \( CF_j = (n_j, S_j, (SS)_j) \) can be merged to form a new cluster just by computing the sums of the individual components of the cluster features \( (n_i + n_j, S_i + S_j, (SS)_i + (SS)_j) \).

When a new data point is processed, both during training and detection, the search of the closest cluster needs to be efficient (and fast enough for the application). We need therefore an efficient indexing structure that helps to find the closest cluster to a given point. The cluster summaries, that constitute the normality observations, are organised in a tree structure. Each level in the tree summarises the CFs at the level below by creating a new CF which is the sum of them. The search then proceeds from the root of the tree down to the leaves, in a logarithmic computational time.

ADWICE is based on the original BIRCH data mining algorithm which has been shown to be fast for incremental updates to the model during learning, and efficient when searching through clusters during detection. The difference is the indexing mechanism used in one of its adaptations (namely ADWICE-Grid), which has been demonstrated to give better performance due to fewer indexing errors [15].

The ease of merging or splitting clusters provided by the way of describing them (using CFs) and the efficiency in the reconfiguration of their indexing enable incremental updates even during the deployment phase in order to cope with changes to normality of the system. The possibility of forgetting unused clusters or incorporating new ones makes ADWICE a good choice for exploring adaptation strategies in changing environments, where normality is subject to concept drift and the detector needs to be efficiently updated online without the need of recomputing the whole normality model from scratch.

The implementation of ADWICE consists of a Java library that can be embedded in a new setting by feeding the preprocessing unit (e.g. when input are
2.2. Water Quality in Distribution Systems

A water distribution system is an infrastructure designed to transport and deliver water from several sources, like reservoirs or tanks, to consumers. This infrastructure is characterised by the interconnection of pipes using connection elements such as valves, pumps and junctions. Water flows through pipes with a certain pressure, and valves and pumps are elements used to adjust this to desired values. Junctions are connection elements through which water can be served to customers. Before entering the distribution system, water is treated first in the treatment plants, in order to ensure its potability. Once processed by the treatment plant, water enters the distribution system so it can be directly pumped to the final user, or stored in tanks or reservoirs for further use when the demand on the system is greater than the system capacity.

Modelling hydraulic water flow in distribution systems has always been an aspect of interest when designing and evaluating water distribution systems [17]. Water distribution networks are typically modelled using graphs where nodes are connection elements and edges represent pipes between nodes. The flow of water through the distribution system is typically described by mathematical formulation of fluid dynamics [18], and computer-based simulation (EPANET [19] is an example of a popular tool) is very common to study the hydraulic dynamics thought the system. Since water must be checked for quality prior to the distribution to the user, system modelling and water quality decay analysis have especially been helpful for finding the appropriate location to place treatment facilities.
2. BACKGROUND

Water quality is determined by the analysis of its chemical composition: to be safe to drink some water parameters are allowed to vary within a certain range of values, where typically the boundary values are established by law. In general, the water quality (WQ) is measured by the analysis of some parameters, for example:

- **Chlorine (CL2) levels**: free chlorine is added for disinfection. Free chlorine levels decrease with time, so for instance levels of CL2 in water that is stagnant in tanks is different from levels in water coming from the treatment plants.

- **Conductivity**: estimates the amount of dissolved salts in the water. It is usually constant in water from the same source, but mixing waters can cause a significant change in the final conductivity.

- **Oxygen Reduction Potential (ORP)**: measures the cleanliness of the water.

- **PH**: measures the concentration of hydrogen ions.

- **Temperature**: is usually constant if measured in short periods of time, but it changes with the seasons. It differs in waters from different sources.

- **Total Organic Carbon (TOC)**: measures the concentration of organic matter in the water. It may decrease over the time due to the decomposition of organic matters in the water.

- **Turbidity**: measures how clear the water is.

Online monitoring and prediction of water quality in a distribution system is however a highly complex and sensitive process that is affected by many different factors. The different water qualities coming from multiple sources and treatment plants, the multiplicity of paths that water follows in the system and the changing demand over the week from the final users make it difficult to predict the water quality at a given point of the system life time. System operations have a consistent impact on water quality. For instance, pumping water coming from two or more different sources can radically modify the quality parameters of the water contained in a reservoir.

In normal conditions, it is possible to extract some usage patterns from the system operations relating the changes of WQ parameters with changes of some system configurations: for example the cause of a cyclic increment of conductivity and temperature of the water contained in a reservoir can be related to the fact that water of a well known characteristic coming from a treatment plant is cyclically being pumped into the reservoir. Other factors, however, can have an impact on the prediction of water quality, making it difficult. The presence of contaminants, which constitutes safety issues as discussed in the next section, can affect water quality making its prediction hard.
2.2. Water Quality in Distribution Systems

2.2.1 Security considerations

As mentioned earlier, water distribution systems have been subject to particular attention from a security point of view. Since physical protection is not easily applicable, intentional or accidental injection of contaminants in some points of the distribution system constitutes a serious threat for citizens. Supervisory control and data acquisition (SCADA) systems provide a natural opportunity to increase vigilance against water contaminations. A specialised Event Detection System (EDS) for water management can be included such that a contamination event is detected as early as possible and with high accuracy with few false positives. EDSs must distinguish changes caused by normal system operations with events caused by contaminations; since different contaminants affect the water quality parameters in different ways, this distinction is not always clear and represents one of the main challenges of EDS tools.

Chapter 3 discusses the application of ADWICE within an event detection system for water quality when integrated on a SCADA system.

2.2.2 Related Work on Contamination Detection

In this section we first describe work that is closely related to ours (water quality anomaly detection), and then we continue with an overview of other works which are related to the big picture of water quality and monitoring.

Water quality anomalies

The security issues in water distribution systems are typically categorised in two ways: hydraulic faults and quality faults [20]. Hydraulic faults (broken pipes, pump faults, etc.) are intrinsic to mechanical systems, and similar to other infrastructures, fault tolerance must be considered at design time to make the system reliable. Hydraulic faults can cause economic loss and, in certain circumstances, water quality deterioration. Online monitoring techniques are developed to detect hydraulic faults, and alarms are raised when sensors detect anomalous conditions (like a sudden change of the pressure in a pipe). Hydraulic fault detection is often performed by using specific direct sensors and it is not the area of our interest.

The second group of security threats, water quality faults, has been subject to increased attention in the past decade. Intentional or accidental injection of contaminant elements can cause severe risks to the population, and Contamination Warning Systems (CWS) are needed in order to prevent, detect, and proactively react in situations in which a contaminant injection occurs in parts of the distribution system [21]. An EDS is the part of the CWS that monitors in real-time the water quality parameters in order to detect anomalous quality changes. Detecting an event consists of gathering and analysing data from multiple sensors and detecting a change in the overall quality. Although specific sensors for certain contaminants are currently available, EDSs are more general solutions not limited to a set of contaminants.
2. BACKGROUND

Byers and Carlsson are among the pioneers in this area. They tested a simple online early warning system by performing real-world experiments [22]. Using a multi-instrument panel that measures five water quality parameters at the same time, they collected 16,000 data points by sampling one measurement of tap water every minute. The values of these data, normalised to have zero as mean and 1 as standard deviation, were used as a baseline data. They then emulated a contamination in laboratories by adding four different contaminants (in specific concentrations) to the same water in beakers or using bench scale distribution systems. The detection was based on a simple rule: an anomaly is raised if the difference between the measured values and the mean from the baseline data exceeds three times the standard deviation. They evaluated the approach comparing normality based on large data samples and small data samples. Among others, they evaluated the sensitivity of the detection, and successfully demonstrated detection of contaminants at concentrations that are not lethal for human health. To our knowledge this approach has not been applied in a large scale to a broad number of contaminants at multiple concentrations.

Klise and McKenna [23] designed an online detection mechanism called multivariate algorithm: the distance of the current measurement is compared with an expected value. The difference is then checked against a fixed threshold that determines whether the current measurement is a normal value or an anomaly. The expected value is assigned using three different approaches: last observation, closest past observation in a multivariate space within a sliding time window, or by taking the closest-cluster centroid in clusters of past observations using k-mean clustering [24]. The detection mechanism was tested on data collected by monitoring four water quality parameters at four different locations taking one measurement every hour during 125 days. Their contamination has been simulated by superimposing values according to certain profiles to the water quality parameters of the last 2000 samples of the collected data. Results of simulations have shown that the algorithm performs the required level of detection at the cost of a high number of false positives and a change of background quality can severely deteriorate the overall performance.

A comprehensive work on this topic has been initiated by U.S. EPA resulting in the CANARY tool [25]. CANARY is a software for online water quality event detection that reads data from sensors and considers historical data to detect events. Event detection is performed in two online parallel phases: the first phase, called state estimation, predicts the future quality value. In the state estimation, history is combined with new data to generate the estimated sensor values that will be compared with actually measured data. In the second phase, residual computation and classification, the differences between the estimated values and the new measured values are computed and the highest difference among them is checked against a threshold. If that value exceeds the threshold, it is declared as an outlier. The number of outliers in the recent past are then combined by a binomial distribution to compute the probability of an event in the current time step.

CANARY integrates old information with new data to estimate the state of the system. Thus, their EDS is context-aware. A change in the background quality due
to normal operation would be captured by the state estimator, and that would not generate too many false alarms. Singular outliers due to signal noise or background change would not generate immediately an alarm, since the probability of raising alarms depends on the number of outliers in the past, that must be high enough to generate an alarm. Sensor faults and missing data are treated in such way that their value does not affect the residual classification: their values (or lack thereof) are ignored as long as the sensor resumes its correct operational state.

CANARY allows the integration and test of different algorithms for state estimation. Several implementations are based on the field of signal processing or time series analysis, like time series increment models or linear filtering. However, it is suggested that artificial intelligence techniques such as multivariate nearest neighbour search, neural networks, and support vector machines can also be applied. A systematic evaluation of different approaches on the same data is needed to clearly summarise the benefits of each approach. This is the target of the current EPA challenge of which our work is a part.

So far, detection has been carried out on single monitoring stations. In a water distribution network, several monitoring stations could cooperate on the detection of contaminant event by combining their alarms. This can help to reduce false alarms and facilitate localisation of the contamination source. Koch and McKenna have recently proposed a method that considers events from monitoring stations as values in a random time-space point process, and by using the Kulldorff’s scan test they identify the clusters of alarms [26].

Contamination diffusion

Modelling water quality in distribution networks allows the prediction of how a contaminant is transported and spread through the system. Using the equations of advection/reaction Kurotani et al. initiated the work on computation of the concentration of a contaminant in nodes and pipes [27]. They considered the topographical layout of the network, the changing demand from the users, and information regarding the point and time of injection. Although the model is quite accurate, this work does not take into account realistic assumptions like water leakage, pipes aging, etc. A more realistic scenario has been considered by Doglioni et al. [28]. They evaluate the contaminant diffusion on a real case study of an urban water distribution network that in addition to the previous hypothesis considers also water leakage and contamination decay.

Sensor location problem

The security problem in water distribution systems was first addressed by Kessler et al. [29]. Initially, the focus was on the accidental introduction of pollutant elements. The defence consisted of identifying how to place sensors in the network in such way that the detection of a contaminant can be done in all parts of the distribution network. Since the cost of installation and maintenance of water quality sensors is high, the problem consists of finding the optimal placement of the minimum number of sensors such that the cost is minimised while performing the best
2. BACKGROUND

detection. Research in this field has been accelerated after 2001, encompassing the threat of intentional injection of contaminants as a terrorist action. A large number of techniques to solve this optimisation problem have been proposed in recent years [30, 31, 32, 33, 20].

Latest work in this area [34] proposes a mathematical framework to describe a wider number of water security faults (both hydraulic and quality faults). Furthermore, it builds on top of this a methodology for solving the sensor placement optimisation problem subject to fault-risk constraints.

Contamination source identification

Another direction of work has been contamination source identification. This addresses the need to react when a contamination is detected, and to take appropriate countermeasures to isolate the compromised part of the system. The focus is on identifying the time and the unknown location in which the contamination started spreading.

Laird et al. propose the solution of the inverse water quality problem, i.e. backtracking from the contaminant diffusion to identify the initial point. The problem is described again as an optimisation problem, and solved using a direct nonlinear programming strategy [35, 36]. Preis and Ostfeld used coupled model trees and a linear programming algorithm to represent the system, and computed the inverse quality problem using linear programming on the tree structure [37].

Guan et al. propose a simulation-optimisation approach applied to complex water distribution systems using EPANET [38]. To detect the contaminated nodes, the system initially assumes arbitrarily selected nodes as the source. The simulated data is fed into a predictor that is based on the optimisation of a cost function taking the difference between the simulated data and the measured data at the monitoring stations. The output of the predictor is a new configuration of contaminant concentrations at (potentially new) simulated nodes, fed again to the simulator. This process is iterated in a closed-loop until the cost function reaches a chosen lower bound and the actual sources are found. Extensions of this work have appeared using evolutionary algorithms [39].

Huang et al. use data mining techniques instead of inverse water quality or simulation-optimisation approaches [40]. This approach makes possible to deal with systems and sensor data uncertainties. Data gathered from sensors is first processed with an algorithm to remove redundancies and narrow the search of possible initial contaminant sources. Then using the maximum likelihood method the nodes are associated with the probability of being the sources of injection.

Attacks on SCADA system

A further security risk that must be addressed is the security of the event detection system itself. As any other critical infrastructure, an outage or corruption of the communication network of the SCADA infrastructure can constitute a severe risk, as dangerous as the water contamination. Therefore, protection mechanisms have to be deployed in response to that threat, too. Since control systems are often
sharing components and making an extensive use of information exchange to coordinate and perform operations, several new vulnerabilities and potential threats emerge [41].

### 2.3 Advanced Metering Infrastructures

Another critical infrastructure that has been subject to widespread attention of governments, industry and academia is the electricity distribution grid. An electricity distribution grid is an infrastructure in which electricity is generated, transmitted over long distances at high voltages and finally delivered to the end users at low voltages. Today’s power demand combined with the limitations of the current infrastructure and the need for sustainable energy in a deregulated marked has led to promotion of smart grid infrastructures. The term *smart* emphasises the idea of a more intelligent process of electricity generation, transmission, distribution and consumption where automatic control provides improved efficiency, reliability, fault-tolerance, maintainability and security. All of this is enabled by the support of advanced Information and Communication Technology (ICT), adding the "cyber" network to the traditional "physical" electricity distribution network.

![Figure 2.1](image)

**Figure 2.1: NIST smart grid interoperability model [1]**

Due to the number of different solutions designed in the early stage, the U.S National Institute of Standards and Technology (NIST) has devised a reference model to be used for smart grid standardisation [1] in order to improve the interoperability of different smart grid architectures. The model, depicted in Figure 2.1, describes the system as an interconnection of six different domains, namely bulk generation (power plants where electricity is produced), transmission (electricity carriers over long distances at high voltage), distribution (electricity suppliers at low voltages), customers (residential, commercial or industrial customers), operations (management and control of electricity) and markets (actors involved in price setting and trading). The actors participate in an open market in the process of generation of electricity and distribution where generation can occur at any stage (therefore consumers can produce and sell electricity generated using photovoltaic
2. BACKGROUND

or wind power systems) and a multiplicity of operators can interact during the activities. In the model we can distinguish the flow of energy and secure information between the domains.

The main components of a smart grid infrastructure, from a technical point of view, have been summarised as follows [42]:

- Smart infrastructure system: is the (cyber-physical) infrastructure for energy distribution and information exchange. It includes smart electricity generation, delivery and distribution; advanced metering, monitoring and communication.

- Smart management system: in the subsystem that provides advanced control services.

- Smart protection system: is the subsystem that provides advanced services to improve reliability, resilience, fault-tolerance and security.

The Advanced (Smart) Metering Infrastructure (AMI), which is the focus of our study in Chapter 4, is the part of the smart infrastructure system that is in charge of automatically collecting consumption data read from the electricity meters for its storage in central databases. The Smart Meters (SM), the new type of computer-based electricity meters devised to replace the old electromechanical versions, are connected to the communication network (proprietary or public IP-based) allowing the operator to perform real-time bi-directional communication. This enables a number of innovative features such as fine-grained electricity billing for new pricing schemes, real-time demand side power monitoring and analysis and remote meter management. The last feature, in particular, allows the operator to connect, configure and disconnect the smart meter remotely. This reduces the management costs of the infrastructure and improves the control over the meters as opposed to the past when electromechanical meters were working offline and physical access was required for maintenance, control and electricity consumption reporting.

2.3.1 Security considerations

The information system which is now integrated into the electricity grid enables in one hand intelligent real-time control for increased efficiency, reliability and resilience of the power grid. On the other hand, it exposes the system to new threats which are inherent to the ICT domain. Attacks performed on the communication network can exploit software, protocol or hardware vulnerabilities for gaining access to the network nodes or control software. In analogy with the intentional contamination threat in water distribution systems, Denial of Service attacks (DoS) or attacks that affect integrity and availability of information on the state estimation of the grid can cause severe damages or even disasters when performed on a large scale.

The advanced metering infrastructure, as part of the smart infrastructure system, is especially vulnerable to security violations since the end devices, the smart meters, are not located in controlled environments. One of the driving motivations
2.3. Advanced Metering Infrastructures

that has lead the development of this practice was the reduction of the so called Non-Technical Losses (NTL, losses that are not caused by normal power loss along the distribution network). The annual revenue losses due to NTL where estimated up to 40% in developing countries [11], where meter tampering and illegal connections to the low-voltage distribution network are the main practices for electricity theft. Smart metering was devised as part of a strategy to prevent NTL, thinking that online load monitoring and tamper detection could alleviate the phenomenon. The ICT technology supporting smart metering however contributed to more vulnerabilities exploitable for electricity theft compared to the past [43]. Apart from physical tampering (which is still feasible to some extent), typical vulnerabilities inherent to networked devices are offered, allowing a potential attacker to operate also remotely. Among different types of attacks, modification or fabrication of fake consumption data can be new means of performing electricity theft. In addiction, the granularity of the measurements and the sensitivity of the information that is exchanged through the communication network have raised valid privacy concerns.

2.3.2 Related Work on AMI Security

Smart grid cyber security is a crucial issue to solve prior to the deployment of the new systems and a lot of effort has been spent on it. Academy, industry and organisations have been actively involved in the definition of security requirements and standard solutions [44, 45, 46, 47, 48].

The Advanced Metering Infrastructure is particularly vulnerable to cyber attacks, and careful attention has been given to its specific security requirements analysis [49]. Confidentiality is a crucial aspect in smart metering, since sensitive information about the user’s activity or habits is available. Although accumulated consumption has always been displayed on electromechanical meters without concern, detailed load profiles available with fine-grained measurements can be analysed to determine which home appliances are creating the load and give detailed description of the activities. Confidentiality of data and credentials should be assured at all the stages, from the metering phase till the storage and management in the operator side. Integrity is strict requirement since data or commands must be authentic, i.e. it should not be possible to modify or replace them with bogus equivalents. Accountability (or non-repudiation) is required since entities that generate data or commands should not negate their actions. Finally, availability is now becoming critical since operation of online components is an integrated part of electricity delivery. Since metering data can be used to estimate the power demand adjusting the supply generation according to it, a large number of unavailable measurements can severely affect the stability of the grid. Cleveland points out that encryption alone is not the solution that matches all the requirements, and automated diagnostics, physical and cyber intrusion detection can be means of preventing loss of availability.

Intrusion detection has been considered as a possible defence strategy in AMIs. Berthier et al. [14, 50] highlight the need for real-time monitoring in AMI systems. They propose a distributed specification-based approach to anomaly detec-
2. BACKGROUND

ition in order to discover and report suspicious behaviours during network or host operations. The advantage of this approach, which consists of detecting deviation from high-level models (specifications) of the system under study, is the effectiveness on checking whether the systems follows the specified security policies. The main disadvantages are the high development cost and the complexity of the specifications.

A recent paper of Kush et al. [51] focuses on the gap between conventional IDS systems and the specific requirements for smart grid systems. They find that an IDS must support legacy hardware and protocols, due to the variety of products available, be scalable, standard compliant, adaptive to changes, deterministic and reliable. They evaluate a number of existing IDS approaches for SCADA systems, the Berthier et al. approach and few conventional IDS systems that could be applied to the AMI, and they verify that none of them satisfies all the non-functional requirements.

Beside cyber attacks, physical attacks are also a major matter of concern. As mentioned earlier, stealing electricity is the main motivation that induces unethical customers to tamper with the meters, and the minimisation of energy theft is a major reason why smart metering practice has been initiated in the early 2000s. However, McLaughlin et al. [43, 52] show that smart meters offer even more vulnerabilities compared to the old electromechanical meters. Physical tampering, password extraction, eavesdropping and meter spoofing can be easily performed with commodity devices.

An approach for discovering theft detection with smart metering data is discussed in Kadurek et al. [53]. They devise two phases: during the first phase the energy balance at particular substations of the distribution system is monitored. If the reported consumption is different from the measured one, an investigation phase aims at localising the point where the fraud is taking place.

In Chapter 4 we present the design of a smart metering infrastructure which uses trusted computing technology to enforce strong security requirements, and we show that the existence of a weakness in the forthcoming end-nodes makes them exploitable for electricity theft and justifies presence of real-time anomaly detection.

2.4 Disaster Area Networks

The last domain where we have focused our studies is mobile ad hoc networking. This is a networking paradigm in which the network nodes communicate by creating peer-to-peer connections without the support of an existing infrastructure. Routing and message dissemination in MANETs, an extensive area of research, is performed by the nodes themselves who create chains of connections. Mobile ad hoc networks can be deployed in different application scenarios in which infrastructure based systems are hard to deploy. Among them, a disaster scenario is presented as a context in which existing infrastructures can be disrupted and spontaneous networks deployed with commodity devices. The hastily formed network can be a fast and early communication system to support rescue operations. The
main challenge in such a scenario is the presence of partitions, i.e. disruption in pockets of connectivity that change over time due to mobility, lack of power etc. In the following section, we describe a dissemination protocol for disaster area networks designed to overcome partitions.

2.4.1 Random-Walk Gossip protocol

The Random-Walk Gossip (RWG) [54] is a manycast partition-tolerant protocol designed to efficiently disseminate messages in disaster area networks. The protocol does not assume any knowledge about the network topology and the identity of the participants, since this information could not be available before deployment time, and in such environments they are expected to be highly dynamic. To overcome this limitation, the protocol is intended to disseminate a message to \( k \) nodes in the network, irrespective of their identity. The number of recipients \( k \) is a parameter configurable by the user.

To cope with network partitions, the protocol uses a store-carry-and-forward approach, meaning that a node stores the messages into a buffer in order to forward them to other nodes when links are available. The name of the protocol derives from the way the messages are spread in the network. When a message is injected by a node, it will perform a random walk on that partition until all the nodes have been reached, or the message is \( k \)-delivered. In the first case, when one of message holders moves to another partition, the spreading of the message will be resumed, and this loop is repeated until the message is \( k \)-delivered or expires.

The random walk of the message is performed by a three-way handshake using specific control packets. When a node is actively trying to disseminate a message, it is said to be the custodian of that message. In order to start the dissemination, the custodian sends a Request to Forward (REQF) packet, which contains the message payload. The nodes in the vicinity who hear that packet store the message in their buffer and reply with an acknowledgement (ACK) packet. The custodian, after updating a data structure, called bit vector, that tracks how many nodes and who has received the message, randomly selects one of the nodes from whom it received ACK packets and sends an OK to Forward (OKTF). The recipient of this packet, now the new custodian, will start again this process to continue the dissemination. The first node that realises that a packet has been \( k \)-delivered sends a Be Silent (BS) packet to inform its neighbours about the completed dissemination of the message.

The gossip component of the name derives from the fact that each time a node receives a packet, it checks whether the sender has not been informed about one of the messages it has in its buffer. In such a case, the node holding the message will start disseminating it with the same procedure described above. This is very useful when a node moves to another partition to quickly resume the dissemination process.
2. BACKGROUND

2.4.2 Security considerations

Mobile ad-hoc networking has been a subject of intense research during the last decade. Apart from the development of protocols and architectures to improve network robustness, delay tolerance, throughput rates, routing performance etc., the application areas of such networks have also raised the need of techniques for protecting them from various security threats. Malicious nodes can exploit protocol vulnerabilities to disrupt the communication, cause node failures or simply behave selfishly exploiting the network resources without participating in the collaborative routing efforts [55]. These issues are present in RWG, where nodes rely on each other for message dissemination but trust relationships between them cannot be assumed. Malicious nodes can freely join the network to perform attacks by exploiting vulnerabilities of the three-way handshake mechanism, as shown in Cururull et al. [56]. Several approaches to intrusion detection have been proposed for MANETs, ranging from standalone fully-distributed architectures, where every node in the network works independently to discover anomalies, to hierarchical solutions with some centralised elements, where nodes collaborate to increase detection performance. For a broad overview of MANET security architectures, the reader is referred to the comprehensive surveys [57] and [58].

Chapter 5 presents our work on an adaptation component of a fully distributed standalone framework for surviving attacks in disaster area networks where every node works independently to capture the state of the network, detect anomalies and take countermeasures to mitigate the impact of the attacks.
Chapter 3

Anomaly Detection in Water Management Systems

This chapter addresses the first application of ADWICE in the physical domain of a cyber-physical system. The hypothesis is that ADWICE, which has been earlier successfully applied to detect attacks in IP networks, can also be deployed for real-time anomaly detection in water management systems. Analysis of physical domain's values and indicators should raise accurate alarms with low latency and few false positives when changes in quality parameters indicate anomalies.

The chapter therefore describes the evaluation of the anomaly detection software when integrated in a SCADA system of a water distribution infrastructure. The analysis is carried within the Water Security initiative of the U.S. Environmental Protection Agency (EPA), described in Section 3.1. Performance of the algorithm in terms of detection rate, false positive rate, detection latency and contaminant sensitivity on data from two monitoring stations is illustrated in Sections 3.2 to 3.4. Finally, improvements to the collected data to deal with data unreliability that arise when dealing with physical sensors are discussed in Section 3.5.

3.1 Scenario: the event detection systems challenge

The United States Environmental Protection Agency, in response to Homeland Security Presidential Directive 9, has launched an Event Detection System challenge to "develop robust, comprehensive, and fully coordinated surveillance and monitoring systems, including international information, for water quality that provides early detection and awareness of disease, pest, or poisonous agents." [59].

In particular, EPA is interested in the development of Contaminant Warning Systems (CWS) that in real-time detect the presence of contaminants in the water distribution system. The goal is to take the appropriate countermeasures upon unfolding events to limit or cut the supply of contaminated water to users.

The challenge is conducted by providing water quality data from sensors of
3. ANOMALY DETECTION IN WATER MANAGEMENT SYSTEMS

six monitoring stations from four US water utilities. Data comes directly from the water utilities without any alteration from the evaluators, in order to keep the data in the same condition as if it would come from real-time sensing of the parameters. Data contains WQ parameter values as well as other additional information like operational indicators (levels of water in tanks, active pumps, valves, etc.) and equipment alarms (which indicate whether sensors are working or not). Each station differs from the others in the number and type of those parameters. A baseline data is then provided for each of the six stations. It consists of 3 to 5 months of observations coming from the real water utilities. Each station data has a different time interval between two observations, ranging in the order of few minutes. The contaminated testing dataset is obtained from the baseline data by simulating the superimposition of the contaminant effects on the WQ parameters. Figure 3.1 [60] is an example of effects (increase or decrease) of different types of contaminants on Total Organic Carbon, Chlorine level, Oxygen Reduction Potential, Conductivity, pH and Turbidity.

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
<th>TOC</th>
<th>Cl2</th>
<th>ORP</th>
<th>COND</th>
<th>pH</th>
<th>TURB</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Petroleum products</td>
<td>↑</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Pesticides (reactive)</td>
<td>↑</td>
<td>↓</td>
<td>↓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Inorganic compounds</td>
<td></td>
<td></td>
<td></td>
<td>↑</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Metals</td>
<td></td>
<td></td>
<td></td>
<td>↑</td>
<td>↓</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Pesticides (non-reactive)</td>
<td>↑</td>
<td></td>
<td></td>
<td>↑</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Chemical warfare agents</td>
<td>↑</td>
<td></td>
<td></td>
<td>↑</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Radionuclides (metal-salt)</td>
<td></td>
<td></td>
<td></td>
<td>↑</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Bacterial toxins (with dechlor agent)</td>
<td></td>
<td></td>
<td></td>
<td>↑</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Plant toxins</td>
<td>↑</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Pathogen (clean with dechlor agent)</td>
<td></td>
<td></td>
<td></td>
<td>↑</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Pathogen (dirty with growth media)</td>
<td>↑</td>
<td></td>
<td></td>
<td></td>
<td>↑</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Persistent chlorinated organics</td>
<td>↑</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 3.1:** Effect of contaminants on the WQ parameters

EPA has provided a set of 14 simulated contaminants, denoting them contaminant A to contaminant N. Contaminants are not injected along the whole testing sequence, but the attack can be placed in a certain interval inside the testing data, with a duration limited to a few timesteps. Contaminant concentrations are added following a certain profile, which define the rise, the fall, the length of the peak concentration and the total duration of the attack. Figure 3.2 shows some examples of profiles.

To facilitate the deployment and the evaluation of the EDS tools, a software called EDDIES has been developed and distributed by EPA to the participants. EDDIES has four main functionalities:
3.1. Scenario: the event detection systems challenge

- Real-time execution of EDS tools in interaction with SCADA systems (collecting data from sensors, analysing them by the EDS and sending the response back to the SCADA tool to be viewed by the utility staff).
- Offline evaluation of EDS tool by using stored data.
- Management of the datasets and simulations.
- Creation of new testing datasets by injection of contaminants.

Having the baseline data and the possibility to create simulated contaminations,
3. ANOMALY DETECTION IN WATER MANAGEMENT SYSTEMS

EDS tools can be tuned and tested in order to see if they suite this kind of application. In the next sections we will explain how we adapted an existing anomaly detection tool and we will present the results obtained by applying ADWICE to data from two monitoring stations.

3.2 Modelling Normality

3.2.1 Training

The training phase is the first step of the anomaly detection. It is necessary to build a model of normality of the system to be able to detect deviations from normality. As mentioned in Section 2.1, ADWICE uses the approach of semi-supervised anomaly detection, meaning that training data is supposed to be unaffected by attacks. Training data should also be long enough to capture as much as possible the normality of the system. In our scenario, the data that EPA has provided is clean from contaminants. The baseline data contains the measurements of water quality parameters and other operational indicators of uncontaminated water over a period of some months. Semi-supervised anomaly detection is thereby applicable.

For our purpose, we divide the baseline data into two parts: the first is used to train the anomaly detector, while the second one is first processed to add the contaminations and then used as testing data. To see how the anomaly detector reacts separately to the effect of each contaminant, 14 different testing datasets, each one with a different contaminant in the same timesteps and with the same profile, are created.

3.2.2 Feature selection

A feature selection is made to decide which parameters to consider for the anomaly detection. In the water domain, one possibility is to consider the water quality parameters as they are. Some parameters are usually common to all the stations (general WQ parameters), but some other station-specific parameters can also be helpful to train the anomaly detector on the system normality. The available parameters are:

- **Common WQ Parameters**: Chlorine, PH, Temperature, ORP, TOC, Conductivity, Turbidity
- **Station-Specific Features**: active pumps or pumps flows, alarms, CL2 and PH measured at different time points, valve status, pressure.

Sensor alarms are boolean values which indicate whenever sensors are working properly or not. The normal value is 1, while 0 means that the sensor is not working or, for some reason, the value is not accurate and should not be taken into account. The information regarding the pump status could be useful to correlate the changes of some WQ parameter with the particular kind of water being pumped to the station. There are other parameters that give information about the status of the
3.2. Modelling Normality

system at different points, for example the measurement of PH and CL2 of water coming from other pumps.

Sensor values and indicators are collected at a certain frequency, as mentioned in Section 3.1. The resulting vectors of numerical values are what we consider as observations of the state of the physical system at these points in time.

Additional features could be considered in order to improve the detection or reduce the false positive rates. Those features can be derived from some parameters of the same observation, or they can consider some characteristic of the parameters along different observations. For instance, to emphasise the intensity and the direction of the parameter changes over the time, one possible feature to be added would be the difference of the value for a WQ parameter with the value in previous observations. This models the derivative function of the targeted parameter. Another feature, called sliding average, is obtained by adding for each observation a feature whose value is the average of the last $n$ values of a WQ parameter. Feature selection and customisation must be made separately for each individual station, since they have some common parameters but they differ in many other aspects.

ADWICE assumes the data to be in numerical format to create an n-dimensional space state vector. Since the timestep series of numerical data from water utilities suit the input requirements of ADWICE, our testing data does not require any particular preprocessing phase before feeding it to the anomaly detector, other than the creation of the derived features and normalisation.

3.2.3 Challenges

The earlier application of ADWICE has been in IP packet header networks. In its native domain, the main problem is finding a pure dataset, not affected by attacks, but the quantity and quality of data is always high. Network traffic generates a lot of data, which is good for having a reasonable knowledge of normality as long as resources for labelling the data are available. Feature selection from IP headers, for example, is easy and does not lead to many problems, while the difficult issues would arise if payload data would need to be analysed, where we would face privacy concerns and anonimisation. In a SCADA system, sensors could give inaccurate values and faults can cause missing observations. This makes the environment more complicated, thus feature selection and handling is complex. Dealing with inaccurate or missing data requires more efforts to distinguish whenever an event is caused due to those conditions or due to contamination. Furthermore, the result of the anomaly detection is variable depending on where the attack is placed. It is not easy, for example, to detect a contamination when at the same time some sensor evaluations for some WQ parameters are inaccurate and some others are missing. Training the system with a limited dataset can result in a sub-optimal normality model, and this causes raising of a lot of false alarms when testing with data that resembles uncovered normality conditions of the system. In the next section we show some results that we obtained testing our anomaly detector with data from two different monitoring stations, proposing some possible solutions for the kinds of problems described.
3. ANOMALY DETECTION IN WATER MANAGEMENT SYSTEMS

3.3 Detection Results

Over six available stations, we have chosen to test our anomaly detection with two representative cases: a station that is slightly affected by sensor inaccuracies and another where faulty values are more frequent.

As mentioned before, we have generated testing datasets by using the second half of the baseline data and adding one contamination per dataset. The contamination has been introduced in the middle of the dataset according to the profile A, depicted in Figure 3.2, which is a normal distribution of the concentration during 64 timesteps. Details about the single stations will be presented separately.

Each testing dataset then contains just one attack along several timesteps. The detection classifies each timestep as normal or anomalous. The outcome of the detection is expressed in terms of DR and FPR, as described in Section 2.1.

3.3.1 Station A

Station A is located at the entry point of a distribution system. It is the best station in terms of reliability of values. It only has the common features and three pump status indicators. Values are not affected by inaccuracies and there are no missing values both in the training and testing datasets. The baseline data consists of one observation every 5 minutes during the period of five months. The first attempt in generating a dataset is done by injecting contaminants according to the normal distribution during 64 timesteps, in which the peak contaminant concentration is 1 mg/L.

Table 3.1 shows the results that we obtained doing a common training phase and then running a test for each of the contaminants. Training and testing have been carried out using a threshold value E in ADWICE set to 2 and the maximum number of clusters M is set to 150. Considering the fact that the amount of contaminant is the lowest possible the results from Table 3.1 are not discouraging. Some contaminants affect more parameters at the same time and their effect is more evident; some others only affect few parameters with slight changes. Contaminant F for instance only affects the ORP, which is not measured in this station, therefore it is undetectable in this case. As can be observed from the table, the FPR is the same irrespective of the contaminant injected. This is due to legitimate normal data that is classified erroneously as anomalous, a misclassification that is not dependent on the contaminant injected.

The anomaly detector must be tuned in order to fit the clusters over the normality points and let the furthest points to be recognised as attacks. To determine the best threshold values of E in ADWICE the ROC curves can be calculated by plotting the detection rate (Y axis) as a function of the false positive rate (X axis) while changing the threshold value (the label beside the point).

Figure 3.3 shows an example of ROC curve obtained with the peak concentration of 1 mg/L of contaminant A, injected according to profile A (Figure 3.2). It is possible to observe that by setting E to 2.0 we obtain the DR and FPR presented in Table 3.1. By increasing E the detector expands the region around the clusters.
### 3.3. Detection Results

<table>
<thead>
<tr>
<th>Contaminant ID</th>
<th>False Positive Rate</th>
<th>Detection Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.057</td>
<td>0.609</td>
</tr>
<tr>
<td>B</td>
<td>0.057</td>
<td>0.484</td>
</tr>
<tr>
<td>C</td>
<td>0.057</td>
<td>0</td>
</tr>
<tr>
<td>D</td>
<td>0.057</td>
<td>0</td>
</tr>
<tr>
<td>E</td>
<td>0.057</td>
<td>0</td>
</tr>
<tr>
<td>F</td>
<td>0.057</td>
<td>0</td>
</tr>
<tr>
<td>G</td>
<td>0.057</td>
<td>0</td>
</tr>
<tr>
<td>H</td>
<td>0.057</td>
<td>0.422</td>
</tr>
<tr>
<td>I</td>
<td>0.057</td>
<td>0</td>
</tr>
<tr>
<td>J</td>
<td>0.057</td>
<td>0.547</td>
</tr>
<tr>
<td>K</td>
<td>0.057</td>
<td>0</td>
</tr>
<tr>
<td>L</td>
<td>0.057</td>
<td>0.406</td>
</tr>
<tr>
<td>M</td>
<td>0.057</td>
<td>0.156</td>
</tr>
<tr>
<td>N</td>
<td>0.057</td>
<td>0.516</td>
</tr>
</tbody>
</table>

**Table 3.1:** Station A detection results of 1mg/L of concentration

where data is considered normal. In this circumstance, the FPR tends to decreases, since legitimate normal data that is further away from the cluster centroids and was previously misclassified as anomalous is now falling within the boundaries of the accepted region. On the other hand, attack data that falls in that region is misclassified as normal, affecting the DR negatively. The contrary holds by decreasing E, leading to a higher DR at the cost of a higher FPR.

Evaluation of the ROC curves of all the contaminants can give hints to determine the best trade-off that gives good detection rates and false positives, but all of those curves refer to a contaminant concentration peak of 1 mg/L. As non-experts it was not clear to us whether this could be a significant level of contamination. For this reason we have tested the sensitivity of the anomaly detection by incrementally increasing the contaminant concentration. In our tests, we increased the concentration in steps of 4 mg/L a time, up to 24 mg/L, under the same configuration of E set to 2 and M to 150. Figure 3.4 shows the variation of the detection rates of three significant contaminants with respect to the increase of the concentration. Contaminants A and L are efficiently detected already at low concentrations, and their DR increases steadily above 80% (up to 98%) with higher concentrations. Contaminant E is not detected at all below and equal to 4 mg/L. This is justifiable since this contaminant slightly affects only the TOC. Increasing the concentration to higher levels, we obtain a DR up to 67%. In this figure the false positive rate is not represented since with the higher concentration of contaminants it is easier to detect the deviation from normality without any increase in the false alarms. These results confirm that even if ADWICE was not originally designed for this kind of application, by finding the optimal tradeoff between detection and false positive rates for 1mg/L, this anomaly detector would give good results for any other greater concentration. We conclude therefore that ADWICE is a good candidate.
3. ANOMALY DETECTION IN WATER MANAGEMENT SYSTEMS

![Station A ROC curve](image1)

**Figure 3.3:** Station A contaminant A ROC curve

![Concentration Sensitivity of the Station A](image2)

**Figure 3.4:** Concentration Sensitivity of the Station A

The situation becomes more complicated when a source of uncertainty is added to the system. Station D is located in a reservoir that holds 81 million gallons of water. The water quality in this station is affected by many operational parameters of co-located pump stations and reservoirs. Station D contains more parameters than station A and some sensors are affected by inaccuracy. In detail, Station D has the following parameters:

- **Common features:** PH, Conductivity, Temperature, Chlorine levels, TOC, Turbidity.

- **Alerts:** CL2, TOC and Conductivity; 1 means normal functioning, while 0 means inaccuracy.
3.3. Detection Results

- **System indicators**: three pump flows, two of them supply the station while the third is the pipe which the station is connected to.
- **Valves**: indicates the position of the key valve; 0 if open, 1 if closed.
- **Supplemental parameters**: Chlorine levels and pH measured in water coming from pump1 and pump2.

By checking the data that EPA has provided, we noted that the only sensor inaccuracy alert that is sometimes raised is the TOC alert, but in general we will assume that the other alerts could be raised as well. There are some missing values in different points scattered within the baseline file. The baseline data consists of one observation every 2 minutes during the period of three months. The same procedure for station A has then been applied to this station.

Figure 3.5 shows the ROC curve obtained with the peak concentration of 1 mg/L and the same profile (profile A, Figure 3.2). In this case, we can observe that the relation between DR and FPR is not as smooth compared the case with station A. Furthermore, the results are generally worse both in terms of DR and FPR.

![Station D ROC curve](image)

**Figure 3.5: ROC curve station D contaminant A**

An accurate feature selection has been carried out to get reasonable results, since trying with all the station WQ parameters the false positive rate is very high. This makes it not worthwhile to explore concentration variations with such bad results.

To mitigate the effects of the missing data and the accuracy, the derivatives and sliding averages of the common parameters have been added as new features. The outcome was that the derivatives emphasise the intensity of the changes, thus improving the detection of the effects of the contaminations, while the sliding window averages mitigated the effect of the abrupt changes in data caused by the inaccuracies or missing data. Some parameters have been ignored, like the pumps flows and the key valve, since they caused lots of false positives if included as features.

After applying the changes described above, ADWICE was run with the parameter E set to 2. Since the dataset is more complex and there are more possible
3. ANOMALY DETECTION IN WATER MANAGEMENT SYSTEMS

combinations of data to represent, the maximum number of clusters M was set to 200.

Figure 3.6 shows the sensitivity of the detection with the increase of the concentrations. The reported FPR, not represented here, was 4%, an acceptable level according to the specifications provided by EPA. Again, Contaminant A is detected already at low concentrations, even though Figure 3.6 shows a lower DR compared to the previous case. Contaminant L this time is not detected at a concentration of 1mg/L, but the detection performance increase suddenly at higher concentrations. Contaminant E follow the same trend as in the previous case, but the range between 4 mg/L and 12 mg/L is shows improvements in this case. This is due to the addition of the derived features based on the TOC, the only WQ parameter affected by this contaminant.

Data from from the above two stations have resulted in clustered models consisting of 115 clusters for station A and 186 clusters for station D.

3.4 Detection Latency

This section focuses on adequacy of the contaminants detection latency. As mentioned in section 3.1, the final goal of the EPA challenge is to apply the best EDS tools in real water utilities to proactively detect anomalous variations of the WQ parameters. Real-time monitoring allows to take opportune countermeasures upon unfolding contaminations. This makes the response time to be as crucial as the correctness of the detection in general, since even having a good detection rate (on average) a late response may allow contaminated water to leave the system and be delivered to users causing severe risks for their health.

The first issue that comes when measuring the detection latency is from when to start counting the elapsed amount of time before the first alarm is raised. This problem is caused by the fact that different event profiles make it necessary to consider the latency in different ways. In case of the normal distribution depicted as profile A (Figure 3.2), a possible approach could be counting the latency of the
3.4. Detection Latency

detection event from the initiation of the contamination event, since the concentration rapidly reaches its peak. If the peak concentration was reached very slowly, the evaluation of latency based on the first raised alarm from the beginning would result in an unnecessary pessimism (e.g. see profile D in Figure 3.2). In this case it would be more appropriate to start counting the reaction time from the time when the peak of the event has taken place. In that case, an earlier detection would give rise to a negative delay and this would signal a predictive warning. For the purpose of our experiments, the normal distribution of profile A suits the computation of the latency based on counting the number of samples from the beginning of the event.

Since in the baseline data time is represented as discrete timesteps, we measure the latency by counting the number of timesteps elapsed before the first alarm is raised. Figure 3.7 and 3.8 show the measured latencies for Station A and Station D respectively, with respect to the detection of the three contaminants presented in the previous section. The curves indicate that in the case of the lowest concentration the latencies are high. For instance the detection of contaminant A and L in station A has a latency of 13 and 20 timesteps respectively. They are around one
3. ANOMALY DETECTION IN WATER MANAGEMENT SYSTEMS

fourth and one third of the total duration of the contamination event (64 timesteps). Contaminant E is not detected at all, so its latency is not represented in the graph. The situation changes positively when the concentrations are gradually increased. The latencies of Contaminant A and L drop sharply until a concentration of 8 mg/L is reached, decreasing 60% and 75% respectively. At the same time, there are some detections of Contaminant E, characterised by a high latency. From this point the latencies of Contaminant A and L steadily drop, while the latency of Contaminant E decreases more rapidly. Latencies for the station D follow the same pattern, although the values are slightly higher.

Checking the results against reality, a latency of 13 discrete timesteps for Contaminant A in Station A would correspond to a latency of 65 minutes, which is quite a long time. One should note that time interval between two observations has a high impact on the real latency, since for example 20 timesteps of detection latency for Contaminant A in Station D with a concentration of 1 mg/L corresponds to 40 minutes of latency, 25 minutes less than the latency in Station A. Even in this case the results are definitely improved by the increases in the contaminant concentrations, but domain knowledge is required to evaluate whether the selected increments to a certain concentration are meaningful.

3.5 Missing and inaccurate data problem

In Section 3.3 we have seen that data inaccuracies and missing data were a major problem in station D. Our approach for the tests carried out so far has been to use workarounds but not provide a solution to the original problem.

More specifically, our workaround for missing data was as follows. We have replaced the missing data values with a zero in the preprocessing stage. When learning takes place the use of a derivative as a derived feature helps to detect the missing data and classify the data points in its own cluster. Now, if training period is long enough and includes the missing data (e.g. inactivity of some sensors or other operational faults) as normality, then these clusters will be used to recognise similar cases in the detection phase as long as no other sensor values are significantly affected. During our tests we avoided injecting contaminants during the periods of missing data.

Sensor inaccuracies are indicated with a special alert in the provided data set (a 0 when the data is considered as inaccurate, i.e. the internal monitoring system warns for the quality of the data). According to our experiments it is not good to train the system during periods with data inaccuracies, even when workarounds are applied. First, learning inaccurate values as normality may result in excessive false positives when accurate values are dominant later. Second, the detection rate can be affected if the impact of the contaminant is similar to some of the inaccurate values. Thus a more principled way for treating this problem is needed.

Our suggestion for reducing the impact of both problems is the classical approach in dependability, i.e. introducing redundancy. Consider two sensor values (identical or diversified technologies) that are supposed to provide measurements for the same data. Then the likelihood of both sensors being inaccurate or both
having missing values during the same time interval would be lower than the likelihood of each sensor "failing" individually. Thus, for important contaminants that essentially need a given sensor value’s reliability we could learn the normal value based on both data sets. When a missing data is observed (0 in the alert cell) the preprocessing would replace both sensor values with the "healthy" one. When one sensor value is inaccurate the presence of the other sensor has an impact on the normality cluster, and vice versa. So, on the whole we expect to have a better detection rate and lower false positive rate with sensor replicas (of course at the cost of more hardware).

3.6 Closing remarks

This chapter focused on application of anomaly detection in the physical domain of water management systems. Thus, our implicit assumption was that the sensor values received by our anomaly detection system had not been subject to manipulation in the cyber layer. This assumption is relaxed in the next domain of application in chapter 4.
Chapter 4

Anomaly Detection in Smart Metering Infrastructures

This chapter focuses on anomaly detection applied to another critical system, the Advanced Metering Infrastructure. The chapter starts (Section 4.1) by describing the design of a smart metering infrastructure that uses trusted computing technology to meet the smart grid security requirements \[49\]. In that section we show that despite the hard core security, there will still be weaknesses in the forthcoming smart meters that justify embedded real-time anomaly detection as second line of defence. In Section 4.2 we propose an architecture for embedded anomaly detection for both the cyber and physical domains in smart meters. Section 4.3 presents the evaluation of an embedded instance of ADWICE for the cyber domain in a prototype under industrial development and illustrates the detection performance of some examples of cyber attacks.

4.1 Scenario: the trusted sensor network

The Trusted Sensor Network (TSN) \[2\] is a smart metering infrastructure defined as a use case within the EU FP7 SecFutur Project \[3\]. The main goal of this solution is to ensure authenticity, integrity, confidentiality and accountability of the metering process in an environment where multiple organisations can operate and where legal calibration requirements must be fulfilled \[2\], in accordance to the model provided by NIST \[1\]. This goal is achieved by a careful definition of the security requirements supported by trusted computing techniques. As depicted in Figure 4.1, a Trusted Sensor Module (TSM) is located in each household. The energy measurements produced by one or more sensors are encrypted and certified by the TSM, which sends them to a Trusted Sensor Module Collector (TSMC). This component gathers the data coming from several TSMs and relays it to the operator server infrastructure for its storage. Through the general purpose network several organisations can get remote access to the functionality of the metering sys-
tem for installation, configuration and maintenance, but strict access policies and accountability of the actions are enforced.

A smart meter in this architecture is called Trusted Meter (TM), and it can be composed of one of more physical sensors, one or more TSMs and one TSMC. A detailed description of the architecture is presented elsewhere [2].

MixedMode™, a partner company within the SecFutur project, has developed a prototype of a Trusted Meter, described in the next section.

4.1.1 Trusted Meter Prototype

The prototype of a TM (depicted in Figure 4.2) is composed of one physical sensor and includes the functionalities of a TSM and a TSMC. The sensor is an ADE7758 integrated circuit, which is able to measure the accumulated active, reactive and apparent power. The functionality of the sensor is accessible via several registers that can be read or written through its interface to the Serial Peripheral Interface (SPI) bus. There is a variety of registers that can be accessed for reading out energy measurements, configuring the calibration parameters, operational states etc. The sensor is then interfaced with an OMAP 35x system where the functionalities of the TSM and TSMC are implemented in software, with the addition of specialised hardware, namely Trusted Platform Module (TPM), that provides the trusted computing functionalities. The system, running Ångström Linux, uses secure boot to ensure that the hardware and software modules are not corrupted and encryption is used to send out the readings to the operator servers.
4. ANOMALY DETECTION IN SMART METERING INFRASTRUCTURES

4.1.2 Threats

After a careful study of the security requirements of the system, the applied security mechanisms and the design of the meter prototype, we came up with the following observations:

- The software, certificates, and credential recorded in the processor module (OMAP 35x) will not be subject to change by malware or external applications due to the use of TPM technology, which also prevents typical smart meter vulnerabilities reported in an earlier work [43].

- The metering data that is sent out to the operator servers is encrypted by the (secure) application using the TPM, hence secure while in transmission.

- The weakest point in the system is represented by the unprotected connection between the sensor and the OMAP 35x system where the TSM and TSMC functionalities are implemented.

The main threat is hence represented by potential man-in-the-middle attacks on the SPI bus that affect the values of data or commands transmitted, as depicted in Figures 4.3 and 4.4.
4.2 Embedded Anomaly Detection

A possible solution would be again based on encryption of the messages prior to the transmission on the SPI bus. This could be applicable in the cases when the sensor and the TSM are two physically separate modules, but when it comes to an embedded system, encryption would dramatically increase the complexity of the sensor circuitry, that must be kept cheap due to the large scale deployment.

The above analysis shows the need for real-time monitoring for intrusion detection is still present although trusted platforms offer higher level of protection than earlier solutions. This motivates proposing an embedded anomaly detection as potential technology to explore.

4.2 Embedded Anomaly Detection

A proposed embedded anomaly detection architecture is devised to be included in the functionality of the Trusted Meter. Figure 4.5 illustrates the main components of the architecture. It consists of five modules: a data logger, a data preprocessor, two anomaly detection modules and an alert aggregator. The data logger is in charge of listening for communication events and data exchange through the sensor-TSM channel. It will record both the cyber domain information, i.e. packet headers or connections, as well as the physical domain information, i.e. energy measurements.

The data preprocessor is in charge of transforming the raw signals detected on the channel into feature vectors that can be fed to the anomaly detection modules for evaluation of current state.

Anomaly detection consists of two modules: one is used for the cyber layer, i.e. the communication protocol on the SPI bus in the prototype meter, while the second is used to detect anomalies on the physical layer, the actual energy consumption that is reported by the sensor. The motivation for this distinction is the need for having two different time scales on the state estimation in the two domains: while the cyber communication can be monitored and suspicious events detected within
4. ANOMALY DETECTION IN SMART METERING INFRASTRUCTURES

Figure 4.5: Proposed Cyber-Physical Anomaly Detection Architecture

seconds, anomalies on the physical domain need to be discovered in the order of days or weeks. This is due to the fact that load profiles change detection is only meaningful when based on a sufficiently long time window. The two modules complement each other: while command injection on the bus can be detected by the cyber layer anomaly detector, consumption data manipulation leading to changing consumption statistics can be detected by the physical layer anomaly detector.

The last component of the architecture, the alert aggregator, takes as input the alarms generated by the two anomaly detector modules and decides whether anomalous behaviour should be reported to the central system.

In the following sections we will describe the modules in depth and present their implementation in our current test system.

4.2.1 Data Logger

In the prototype meter, the unprotected communication channel between the sensor and the TSM+TSMC module is the SPI bus. The SPI communication is always initiated by the processor (in the OMAP 35X system) who writes, in the communication register of the sensor, a bit that specifies whether the operation is a read or a write command, followed by the address of the register that needs to be accessed. The second part of the communication is the actual data transfer from or to the addressed register of the sensor. The application that implements the TSM and TSMC functionalities performs an energy reading cycle every second, sending calibrations or configuration commands when required.

Our bus logger records the following information: the timestamp of the operation, the command type (read or write), the register involved in the operation of the value that is read or written. In our experimental setup, as described later, the bus logger has been implemented at the driver API level of the SPI interface of the TSM+TSMC, intercepting the communication prior to the transfer to the higher
levels.

4.2.2 Data Preprocessing and Feature Extraction

The data preprocessor receives records in the format presented in the previous section, and produces vectors of features that will be processed by the anomaly detectors. A common data preprocessor for both domains avoids processing the received data twice. The features selected are numerical variables that all together represent the normal operation of the system. For the cyber domain, these are based on information regarding the frequency and types of operations carried out on the SPI bus during a period of observation time \( I \). There are three categories of features:

- **Operation type**: percentage of number of read or write operations performed in the period of observation \( I \). An additional feature counts the number of times the read-only registers are accessed, which is useful to capture the fact that most of the time (every second in our case) the communication is performed to read out energy measurements.

- **Category type**: percentage of the number of times the registers of the following categories are accessed in the period of observation \( I \): *reading, configuration, interrupt, calibration, event, info*. The first category includes registers used for accumulation of active, reactive and apparent energy for the three different phases. Register categorised as *configuration* are those used for configuring different operational parameters of the energy measurement. Registers in the category *interrupt* are interrupt status flags. Registers included in the *event* category are used to store information on events such as voltage or current peak detection etc. The category *calibration*, groups the important registers used to calibrate the different parameters of the sensor. Finally, the *info* category groups registers where checksums and the version of the sensor are stored.

- **Register frequencies**: usage frequency of each individual register addressed in the period of observation \( I \).

These features are designed to characterise the typical communication patterns, therefore anomalous communication sequences or register access rates should be discovered by the anomaly detector. Note that these features are specific to the smart metering device that is being monitored, in contrast with the physical domain indicators, that as we see below are of a general nature.

In the physical domain, commonly used indices for customers characterisation, based on load profiles, can be utilised as features. These include daily indices, as the widely used indices proposed in Ernoult et al. [61], such as the non-uniformity coefficient \( \alpha = \frac{P_{\text{min}}}{P_{\text{max}}} \), the fill-up coefficient \( \beta = \frac{P_{\text{avg}}}{P_{\text{max}}} \), the modulation coefficient at peak hours \( MC_{\text{ph}} = \frac{P_{\text{min}};\text{ph}}{P_{\text{avg}};\text{ph}} \) and the modulation coefficient at non-peak hours \( MC_{\text{oph}} = \frac{P_{\text{min}};\text{oph}}{P_{\text{avg}};\text{oph}} \), where \( P_{\text{min}} \) is the minimum power demand reported during the day, \( P_{\text{max}} \) is the maximum power demand, \( P_{\text{avg}} \) is the average power.
4. ANOMALY DETECTION IN SMART METERING INFRASTRUCTURES

Demand, $P_{avg,ph}$ is the average power demand during the peak hours and $P_{avg,oph}$ is the power demand during the off-peak hours. More refined indices that take into account weekly patterns (working days and weekends) can be added, as those described in Chicco et al. [62].

4.2.3 Cyber-layer Anomaly Detection Algorithm

The sensor-processor communication is based on a series of messages exchanged through the bus. In this context, the set of possible combinations is not very large, due to the fact the set of registers accessible is bounded. The behaviour in terms of the commands sequences, captured by the features selected, can be considered as data points that fall into certain regions of the multidimensional features space. In order to identify the good behaviour, an algorithm that is able to identify these regions and consider them as the normality space would be needed.

Therefore, we have adopted and embedded an instance of ADWICE, since its computationally efficient smart indexing strategy makes it suitable for use in a resource constraint system such as an embedded system for smart metering. Section 4.3 presents the evaluation of this algorithm.

4.2.4 Physical-layer Anomaly Detection Algorithm

The features available for modelling the physical domain suggest that when the load profile changes due to an eventual attack, the statistics would be affected over a long period. A lightweight change detection algorithm can therefore be embedded into the smart meter. A statistical anomaly detector using the indicators described in Section 4.2.2 has been developed. However, due to absence of long term data and ability to train and test the anomaly detector on consumed electricity profiles, we have focused development and tests on the cyber level attacks.

4.2.5 Alert Aggregator

The last component of the architecture is in charge of collecting the alerts generated by the anomaly detector modules, and performing aggregation in order to reduce the number of alarms sent to the central operator. The alert aggregation module can gather additional information in order to provide statistics that show the evidence of an attack or the anomalous conditions. This creates a smart meter health although individual analysis would still require a lot of effort and privacy concerns would hinder its "careless" deployment. However, this can be useful when investigating areas in which non-technical losses (i.e. losses that are not caused by transmission and distribution operations) are detected, supporting for example localisation strategies as in [53]. Future works include further investigation and privacy-aware development of this module.
4.3 Evaluation

In this section we present the evaluation of the anomaly detection on a number of attacks performed in the cyber domain. We start presenting the methodology to collect the data for evaluation. Then we introduce the cyber attacks we performed and finally we show the outcomes of the clustering-based anomaly detection algorithm.

4.3.1 Data collection

In order to obtain data for training and testing the anomaly detection algorithm, the trusted meter prototype has been installed in a household and real energy consumption measurements on one electrical equipment have been collected during a period of two weeks in January 2012. Although this frame of time is not long enough to capture normality for the physical domain, it is representative enough for the cyber domain, where a 187MB data log file has been collected. The log, produced by the data logger module as explained in section 4.2.1, is composed of bus communication transactions that involve several registers for energy reading, sensor configuration, calibration commands and sensor events. The energy reading operations are performed with a period of one second, and they are predominant in the dataset. The data preprocessor, as presented in section 4.2.2, gathers the transactions during a period of observation $I$ which has been set to 10 seconds, and produces a feature vector that is processed by the anomaly detection algorithm.

4.3.2 Cyber attacks and data partitioning

Four types of attack have been implemented:

1. **Data manipulation attack**: In this scenario, the attacker performs a man-in-the-middle attack in which the values of the registers involved in the energy measurement are lowered. This can be easily done by overwriting the signal on the bus on every reading cycle.

2. **Recalibration attack**: This command is injected on the bus in order to change the value of some registers that hold calibration parameters, causing the sensor to perform erroneous measurement adjustments during its operation.

3. **Reset attack**: This command causes the content of the energy accumulation registers to be wiped out. It has to be executed within every reading cycle in order to reduce the reported energy consumption. In our scenario, we executed it with a period of one second, interleaving it with the period of the measurement process, which has the same length.

4. **Sleep mode attack**: This command puts the sensor into sleep mode, e.g. no measurements are taken. While in sleep mode, the sensor SPI interface still replies to the commands executed by the processor module, but the energy consumption is not accumulated by the sensor.
In our evaluation, we tested the attacks 2 to 4, since attack 1 does not produce new messages on the bus and it would only be detectable by the physical layer anomaly detector. The attacks were first implemented as malware at application level, acting through the SPI interface drivers of the processor module, in order to see their impact on the messaging through the bus. Since the data was collected in an attack-free scenario, a script has been implemented to weave and reproduce the attack information into the clean data. Our traces consist of two weeks of logs in which 2/3 of the data has been left clean to represent normal conditions, and the remaining 1/3 is affected by one attack at a time, thus 3 different testing traces were generated. A physical hardware that can implement such attacks on the bus (SecFat) is under development in the SecFutur project.

The anomaly detector algorithm is therefore trained with the feature vectors obtained by the first third of the data, while we tested with the 3 traces in which half of the trace contains the remaining one third of normal data and the rest the normal data interleaved with the attack. When the data preprocessor computed the feature vectors, we manually set an oracle bit to indicate whether the features are affected by an attack or not. This will be helpful for comparison when evaluating the outcomes of the anomaly detection algorithm.

4.3.3 Results

During our evaluation, we have tuned the two classical parameters of the clustering-based anomaly detection algorithm which need to be configured manually in order to create a good normality model and optimise the search efficiency. As mentioned in Section 2.1, these are the maximum number of clusters (M), and a cluster centroid distance threshold (E), that is used when determining whether a new data point falls within its closest cluster or not. The optimal number of clusters typically depends on the distribution of the input data into the multidimensional space. The threshold is also important, since during real-time monitoring it determines whether a new feature vector belongs to any pre-existing cluster or not. Therefore, in order to select a suitable combination of the two parameters, the outcomes of the detection were explored with M ranging from 10 to 100, and E ranging from 1 to 2.5. The metrics used for evaluating the detection algorithm were the detection rate (DR) and the false positive rate (FPR).

Our results show that the algorithm does not build a correct partitioning of the normality data when M is set to 10 and 20 with all the possible combinations of E. In the detection phase, all the observations (with or without attacks) are classified as anomalous, leading to 100% DR but with a 100% FPR rate. The explanation for that result is that with such few clusters the multidimensional space is exactly partitioned in order to index the clusters in fine-grained detail. Overfitting the points during learning normally leads to a large classification error at detection time, in fact all the observation are marked as attacks, leading to 100% FPR and DR. Better partitioning of the normality data is found when M is set to at least 30. In this case, the algorithm uses 18 clusters to model the data, and for every configuration of E in the range between 1 and 2 we get 100% DR with no false positives for all three
types of attacks. In the cases when E is over 2 we allow a very large threshold and
the detection rate is reduced to zero for the recalibration attack, since it is the at-
tack type that is more similar to normal conditions where recalibration takes place
in the training period. The results are similar when increasing M up to 100. This
means that the algorithm finds 18 clusters to be the best number for modelling the
normality data.

4.4 Closing remarks

This chapter proposed the application of anomaly detection in the cyber and phys-
ical domain of a smart grid infrastructure, at the smart metering end point. Evalua-
tion of the proposed architecture with examples of cyber attacks has indicated that
anomaly detection can be efficient as second line of defence even in systems where
security is a well developed attribute.

In the evaluations presented in this and the previous chapter we have shown that
the selection of the parameters configuration of the anomaly detector (the threshold
E and the number of clusters M in the case of ADWICE) has an impact on the trade-
off between detection rate and false positive rate. A suitable configuration should
be selected in a pre-study phase prior to the deployment of the system. However,
there has been increasing attention to adaptation strategies in order to obtain an
autonomic configuration of the parameters [5].

Chapter 5 focuses on the problem of online adaptation of the parameters of an
intrusion detection framework in order to adjust this trade-off at runtime, applied
to a domain where adaptation is a challenging but desired functionality.
Chapter 5

Online Parameter Adaptation of Intrusion Detection

The first challenge that occurs when applying anomaly detection is the definition of the normal operating region of the observed system. Modelling normality is often complex and relevant features must be selected to capture the normal behaviour of the system. Another challenge is the definition of the boundaries between normal and anomalous behaviour. In the previous chapters we have focused our studies on anomaly detection on two domains of cyber-physical systems, showing that with careful feature selection and parameter tuning for a given anomaly detector we could achieve acceptable tradeoffs between detection performance and false alarm rates.

A third challenge of anomaly detection that typically hinders its applicability in real settings is the autonomic adaptation when applied to domains where normality might naturally change over time (this is often referred as concept drift). In machine learning literature there are efficient techniques able to detect and classify changes due to concept drift [4], allowing online adaptation of the normality model [63] or detection parameters [64]. Unfortunately, their general applicability to real-time anomaly detection is limited. The unconstrained distinction of whether a change is due to benign drifts of the normality or due to an attack is still an open issue when learning in adversarial environments [65]. Most of the existing solutions that cope with adaptation applied to anomaly detection either assume gradual drifts [66] [63] or rely on human intervention, which is the case for ADWICE [15].

In this chapter we focus our attention to an intrusion detection and response architecture devised to protect a mobile cyber-physical system: a disaster area communication network, as presented earlier in Chapter 2. In this scenario, ad hoc mode of communication is employed whereby each mobile device opportunistically connects to neighbours in its vicinity in order to establish an ad hoc chain of dissemination and collaboration for rescue operations management. As shown in previous work [56], even in disaster scenarios there is a need for intrusion detection to provide network survivability in case of attacks. In this domain, how-
ever, changes can occur at any time. Different load conditions, changing number of nodes and mobility patterns could radically modify the "state" of the network compared to what could have been considered as normality in the past. Adaptation is therefore a required property but extremely challenging in this context.

When implementing a distributed intrusion detection scheme for ad hoc communication, another important factor must also be considered: handheld devices are power-hungry. The adoption of security solutions in this domain is in fact hampered due to the power drain on the handsets. There is an obvious trade-off one should consider with an impact on survivability: we get more protection if we have endless energy.

In this chapter we show that we can adapt the sensitivity of security mechanisms, tuning them based on energy estimates. We demonstrate this idea of energy-based adaptation using RWG (described in Section 2), an ad hoc protocol that was created for persistent dissemination in a disaster scenario, even in hostile environments, e.g. under partitioned networks and energy deficiency. This dissemination protocol and the associated general survivability framework, in which local anomaly detection, diagnosis, and mitigation are part of the application needs, have been developed in a larger project on Hastily Formed Networks [67] and published elsewhere [68, 69, 70].

Since large scale ad hoc networking experiments are difficult to realise, this chapter focuses also on how the environment for large scale studies, such as ns-3 simulation platform can be extended in order to model energy-based adaptations of protocol and service layer modules. This will enable studies of network life time in presence of various threats and different mobility patterns.

The chapter is organised as follows. Section 5.1 describes the General Survivability Framework, the fully-distributed intrusion detection and response architecture designed to protect the disaster area communication network; Section 5.2 presents the proposed adaptation module; Section 5.3 describes the proposed model for accounting CPU energy consumption in network simulation, and finally simulations results are presented in Section 5.4.

### 5.1 Scenario: challenged networks

The General Survivability Framework (GSF) [69] is a modular architecture designed to provide a coherent security approach for mobile ad hoc communication in challenging environments, such as disaster area networks. In such environments, with the hypothesis that spontaneous ad hoc networks need to be created on the fly, pre-existing trust relationships among nodes cannot be assumed, thus limiting the use of encryption-based or collaborative protection mechanisms. The GSF, instead, is a fully-distributed architecture which is installed on each node. The framework is composed of four functional modules that cooperate in order to detect, diagnose, and react to network attacks (see Figure 5.1).

Here we describe an instance of the framework in which we have embedded our adaptation module. The first module, the anomaly detector, is responsible for monitoring the network traffic from the vicinity in order to detect anomalous
conditions. The features gathered by the module are based on variables that are believed to characterise the statistical behaviour of the routing protocol, and are represented as a numerical vector. The anomaly detection algorithm is based on a distance function that compares the current observation vector with an average vector that represents what is learnt during a training phase prior to deployment in an attack state. If the distance is greater than a threshold (which is also obtained during the training phase) the algorithm raises an alarm.

When an alarm is raised, the diagnosis component is triggered to classify the attack type among the known cases, comparing the distance of the observation vector with exemplar vectors that are known to represent attack situations.

Once the attack is classified, a relevant mitigation component is engaged to apply the appropriate countermeasures to reduce or eliminate its impact. If the attack cannot be matched to any of the known cases, a generic mitigation strategy that may help to protect the communication in hostile environments is employed.

The adaption module, which is the focus of this chapter, is responsible for changing the parameters of the other components in order to cope with the changes in the state of the network and the node. Figure 5.1 shows the interconnections of the four modules, which is similar to a closed loop control system, with the difference that the state of a given node is highly dependent on the global state of the network that is outside our control, i.e. the collective behaviour of other nodes. The framework has been tested on top of Random-Walk Gossip (RWG) [68], a manycast partition-tolerant protocol designed to energy-efficiently disseminate messages in disaster area networks, introduced earlier in Chapter 2 of this thesis.

5.2 Adaptive detection

The adaptation process normally involves monitoring the system under control, detecting changes, deciding and reacting to adjust the system parameters in order to bring it to the desired state, which often optimises towards some target performance. The survivability framework presented earlier differs from this concept
due to the fact that the state of the network is not fully observable and controllable from the point of view of an individual node, which has a partial view restricted to its vicinity. The emerging global response of the network determines a node’s reaction to the attack. In this context the adaptation process that we aim for is a self-adaptation [71] approach, meaning that the control system itself (i.e. the GSF) should be adjusted with the changing conditions of both the node and the network.

Our approach to adaptation consists of adapting the tradeoff between the sensitivity of the intrusion detector and its energy consumption. In its essence, it falls in the class of adaptation algorithms that tradeoff between security provisioning and resource consumption [72] [73]. The solution proposed in this chapter takes into account the perceived state of the network, focusing mainly on the most valuable feature of the internal state of the node, the energy. The assumption is that while supporting resistance to the attacks, the nodes should also be aware of their energy budget, adapting their behaviour in an efficient way in order to extend their lifetime as much as possible. With the support of energy modelling and simulation, we show that aggressive energy-agnostic attack survivability strategies, with excellent detection performances, could become useless once their impact on the energy consumption reduces the lifetime of the network.

A similar approach based on a mathematical model of the system has been proposed by Mitchell et al. [74]. In their case, however, the intrusion detection scheme is collaborative, where the final decision is based on a voting strategy, and their results have only been proven theoretically without the support of simulation.

![Figure 5.2: Energy-aware adaptation of IDS parameters](image)

5.2.1 Adaptation component

The adaptation component proposed (see Figure 5.2) consists of a function that takes as input the current energy level of the node, the perceived attack situation and the current parameter set configuration. The output is a new set of parameters that is both relevant for the detection accuracy and has a strong impact on the energy consumption.

The function is based on a decision table. Each action is a parameter set configuration according to the perceived attack situation, modelled as conditions, and different energy states, modelled as rules. Rules based on energy levels could make the system reactive to energy changes, tilting the trade-off towards the energy aspect when this resource is limited. Prediction of the energy depletion based on the actual traffic load and CPU consumption could be the basis of a more proactive
adaptation of the IDS parameters. All the combinations can be selected during a pre-design study on the trade-off between resource usage and security provisioning.

5.2.2 Energy-based adaptation in the ad hoc scenario

As a case study, we consider the GSF presented earlier. In this framework, there is a relevant parameter that governs the detection, diagnosis and mitigation cycle and has a strong impact on the CPU utilisation. It is the aggregation interval $I_a$ in which the network state observations are aggregated and the alarm state is evaluated. The shorter the interval the faster the response to the attacks, but at the same time the detection accuracy is lower and the power consumption is higher. The longer the interval, the better the detection accuracy, but the detection latency would increase as well. In this case the power consumption of the detector would be lower, but a high latency could allow the attack to spread throughout the network causing subsequent negative consequences.

In the work presented in [69], a trade-off between detection accuracy and minimum latency has been obtained with $I_a$ at 50 seconds. A longer interval, providing a better detection accuracy at the cost of a higher latency could still be acceptable if the overall impact on the energy consumption outperforms the case in which the latency is shorter.

The decision table implemented in this scenario takes into account two conditions: attack or non-attack. A reactive solution has been implemented considering the following four battery level ranges: 100% to 60%, 60% to 40%, 40% to 20%, and 20% to 0% (this makes sense with the linear battery discharge model, otherwise it should be replaced with a lifetime prediction based on the current load and the specific discharge model). Since, as mentioned earlier, the global response to the attack is given by the collective response of the nodes in the network, the choice of the aggregation interval is based on the idea that nodes with more energy should compensate the higher latency of nodes with a limited battery level that try to extend their lifetime slowing their detection engine. Nodes that have a battery level bounded between 100% to 60%, in normal conditions, set the shortest interval $I_a$ (namely IDS_FAST), to react more quickly in case of attacks. During attacks, instead, the interval is set to longer value (IDS_MEDIUM) in order to improve the detection accuracy and save energy at the same time. With a lower battery level, between 60% and 40%, the interval during attack is further increased (IDS_SLOW).

<table>
<thead>
<tr>
<th>Energy fraction (%)</th>
<th>Non-Attack</th>
<th>Attack</th>
</tr>
</thead>
<tbody>
<tr>
<td>100-60</td>
<td>IDS_FAST</td>
<td>IDS_MEDIUM</td>
</tr>
<tr>
<td>60-40</td>
<td>IDS_FAST</td>
<td>IDS_SLOW</td>
</tr>
<tr>
<td>40-20</td>
<td>IDS_MEDIUM</td>
<td>IDS_SLOW</td>
</tr>
<tr>
<td>20-0</td>
<td>IDS_SLOW</td>
<td>IDS_SLOW</td>
</tr>
</tbody>
</table>

Table 5.1: Aggregation interval based on energy levels
5.3 Energy modelling

Energy aware applications require access to the energy level of the system in which they are running. Network simulators typically focus on the energy consumption of the network interface, ignoring the contribution of other power-hungry components such as the CPU. When evaluating the energy efficiency of network protocols in simulators, it is common practice to estimate their energy consumption from the amount of network traffic they create. The wireless card is assumed as the most power-consuming component of the system used by the network protocols, hence other elements are not considered. More realistic simulations should also include, for example, the contribution of the processor.

In ns-3, a widely used network simulator, an energy modelling framework is available [75]. In this framework, the device energy consumption and energy source are modelled as two separate elements. Energy sources are modelled by mathematical equations that describe their discharge curves. Device energy models specify the current draw at the states in which the device can operate. Each device drains a certain amount of current from the source, whose total energy consumption is computed from the sum of all the individual current draw contribution. Although there are many energy source models, characterised by different discharge curves, there is currently only one device energy model available, the WiFi energy consumption model.

A similar energy model targeting wireless sensor networks (thus not really suitable to our scenario) has earlier been proposed by Chen et al. [76] for OMNeT++ [77]. In addition, a simple CPU model, that accounts the energy consumption in the active or inactive state, is included. For the same simulation platform, a generic energy model for wireless networks has been proposed by Feeney et al. [78]. It improves the battery depletion handling, compared to the existing model, but CPU energy consumption is not modelled. The common limitation of network simulators is that they normally assume unlimited computational power and ignore process execution time. This hinders CPU modelling and more detailed energy accounting.

Simulation of power consumption of programs running on real systems can be done with instruction sets simulators, such as ARMulator [79] or SimpleScalar [80] among the others. Unfortunately, they are not suitable for simulating complex programs, neither can they simulate networking scenarios.

Combined network simulation and CPU instruction set simulation has been proposed in SunFlower [81], Real-Time Network Simulator (RTNS) [82] or Slice-Time [83]. However, the first lacks nodes mobility support, while the two latter works do not simulate energy consumption. In all the cases the CPU instruction set simulator and the network simulator need to be synchronised, and the simula-
5. ONLINE PARAMETER ADAPTATION OF INTRUSION DETECTION

The computation time is rather slow. The CPU energy model for network simulation proposed in the next section could represent a way to include CPU energy consumption, thus enabling adaptation studies based on energy.

5.3.1 A CPU model for network simulation

Ideally, a CPU utilisation indicator can be directly translated into current draw values. Unfortunately, this information in network simulators is not available due to the limitation discussed earlier. In order to account for the CPU energy consumption based on the impact of network protocols and/or additional applications, a simple CPU energy consumption model is introduced. Our model is based on the assignment of an energy footprint to each application simulated. The model consists of a state machine for each application that runs in the simulation environment (Figure 5.3). Each state represents a possible operational mode of the application, that differs from the others in terms of behaviour and consequent CPU usage. By the application running at each of its states in a real device, the isolated impact on the power consumption can be profiled. Functions that produce power consumption values depending on the inputs of the application which have an impact on the energy consumption can then be associated to the corresponding states. In the case in which the power consumption cannot be directly measured, one could for example analyse the CPU utilisation increase caused by the application, which may be translated into power consumption values for simulation purposes. In a work published later, for instance, we profile the energy consumption of the GSF and RWG in terms of CPU increase on modern Android-based handsets [84]. In the simulation, the global CPU energy consumption is then given by the combination of all the individual power contributions of the modelled applications at the current inputs.

![Figure 5.3: State machine with associated power consumption](image)

To illustrate the approach, we show how this model can be applied to account for the CPU energy consumption of our case study.

5.3.2 Application of the model

In our environment, the applications that need to be accounted for their energy consumption are the RWG protocol and the GSF. In order to assign the current draw at each state, we isolate the power consumption of real devices when running in
similar conditions. For the RWG protocol, we can characterise the following states: \textit{RWG, RWG\_mit\_gray} and \textit{RWG\_mit\_drain}. The first state represents the normal working conditions of the RWG protocol. The individual energy footprint of the protocol running at different transmission rates can be measured as in [70]. The other states represent the protocol behaviour when attack mitigation strategies (to the grayhole and drain attacks) change the way the protocol operates. In the current implementation, RWG can be operated in grey hole or drain attack mitigation. A power consumption function can be assigned to those states with the same logic as before.

A separate finite state machine is created for the GSF application, which consists of the intrusion detection, diagnosis and mitigation selection components. Three states, that capture the frequency at which the analysis cycle is performed, are defined: \textit{IDS\_FAST}, \textit{IDS\_MEDIUM} and \textit{IDS\_SLOW} for short, medium and long interval $I_a$ used to tune the GSF operation frequency. Again, power consumption functions for each of these states can be extracted from real devices emulating this application under similar workloads. Finally, the total CPU energy consumption is then given by the sum of the power consumption contribution of the RWG application and the GSF application, as depicted in Figure 5.4

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure5.4.png}
\caption{CPU power consumption of the RWG and IDS framework over some time interval}
\end{figure}

Given this CPU energy consumption model, the impact of RWG and the GSF operations can be made more visible, while energy-awareness and self-adaptivity can be implemented and simulated.

### 5.4 Evaluation

The goal of this section is to show that local adaptation based on per-node estimates of the available energy leads to better overall performance in the network and extends its lifetime.

#### 5.4.1 Implementation and simulation setup

As baseline, the same scenario presented in [69] has been considered. The network consists of 25 nodes moving in a disaster area network [85]. The load of the network is 15 messages per second sent from randomly chosen nodes. The messages are disseminated in manycast to at least $k=10$ nodes and have an expiration time of 400 seconds.
5. ONLINE PARAMETER ADAPTATION OF INTRUSION DETECTION

The ns-3 energy framework has been included and extended with the proposed CPU energy model from Section 5.3 to create a model for energy awareness. For simplicity, we have used the energy source model characterised by an ideal linear discharge curve. To consider the energy impact of the wireless device, the WiFi energy model has been employed in the simulation. The current draw values assigned to the different operational states of the wireless interface are the same as in the work by Wu et al. [75].

In order to assign the power consumption values to the states that characterise the RWG application model, the results from Vergara et al. [70] have been considered. To be compliant to the ns-3 energy model, the CPU energy model should specify current draw values in Ampere instead of power in terms of Watts. As the ns-3 source model assumes a constant battery voltage, the conversion between Watts and Ampere is immediate. Assuming, for example, that the transmission rate is 15 messages per second, according to the load injected in the network, the energy consumed by the RWG application in the normal operation mode at this rate is 0.025W, as depicted in Figure 7 in Vergara et al. [70]. Furthermore, we consider the additional contribution of 0.1W as the power consumption due to message deletion at the same rate. The power consumption value at a rate of 15 messages per second is then 0.125W. Considering 3.7V constant battery voltage, the current draw associated to this state is 0.034A. RWG in mitigation mode usually performs less operations, since the information contained in some signalling packets is discarded or not processed. A lower footprint has been estimated and assigned to both of the considered mitigation states. Table 5.2 summarises the current draw assigned to RWG.

<table>
<thead>
<tr>
<th>RWG application state</th>
<th>Current draw (A)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RWG</td>
<td>0.034</td>
</tr>
<tr>
<td>RWG_Drain_Mitigation</td>
<td>0.018</td>
</tr>
<tr>
<td>RWG_Greyhole_Mitigation</td>
<td>0.018</td>
</tr>
</tbody>
</table>

**Table 5.2: Current draw of the RWG application**

For GSF, as mentioned earlier, the three modelled states correspond to when the detection-diagnosis-mitigation analysis are performed within short, medium or long, intervals, as specified by the parameter $I_a$. The interval of aggregations $I_a$ chosen for IDS_FAST is 50 sec, IDS_MEDIUM is 75 seconds and IDS_SLOW is set to 100 seconds. The assigned constant energy footprint of the IDS states is shown in Table 5.3.

<table>
<thead>
<tr>
<th>IDS application state</th>
<th>Current draw (A)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDS_FAST</td>
<td>0.040</td>
</tr>
<tr>
<td>IDS_MEDIUM</td>
<td>0.025</td>
</tr>
<tr>
<td>IDS_SLOW</td>
<td>0.010</td>
</tr>
</tbody>
</table>

**Table 5.3: Current draw of the IDS application**
5.4. Evaluation

In order to illustrate the benefits of adaptation, we focus on two of the possible attack types: the drain attack and the grey hole attack. In the first attack type the malicious nodes act in order to drain the battery of the victims, injecting fake signalling packets that cause benign nodes to perform a lot of unnecessary disseminations. In the second type of attack, malicious nodes target the message dissemination, by sending fake signalling packets that cause the interruption of the process before the messages have actually been delivered to the intended \( k \) nodes. Both attacks are interesting to analyse with regard to the adaptive function, since they are complementary in terms of energy consumption. In the drain attack nodes waste energy as a consequence of the attack, thus good detection accuracy and low latency are necessary to avoid energy waste. On the contrary, the goal of the grey hole attack is disrupt the expedient dissemination of the messages in the network, thus the nodes should consume less energy due to the reduced amount of network traffic. Longer detection latency in this case could be tolerated, from the energy perspective, but adaptation should still guarantee good detection to ensure network dissemination goals to be fulfilled.

5.4.2 Simulation results

In order to test the complete set of actions performed by the adaptivity function during an entire simulation period (3000 seconds as in [69]), the initial energy level assigned to all nodes is selected to be lower than needed to conclude the simulation over 3000 seconds, which turns out to be 500J. In this way, one can determine whether the adaption extends the lifetime of the network compared with the non-adaptive case. In all of the following simulations, five randomly placed malicious nodes start the attack at second 2067 and this lasts until the end of the simulation (details same as [56]). Ten runs of the same simulation are performed, and the results are averaged.

The first attack type we simulated is the drain attack. Figure 5.5a shows the comparison of the average available energy in the network in the adaptive case.
5. ONLINE PARAMETER ADAPTATION OF INTRUSION DETECTION

compared to the non-adaptive one when the network is under the drain attack with mitigation enabled. As it can be observed from the figure, the adaptive function outperforms the non-adaptive case by extending the lifetime of the network by over 250 seconds. The difference of the energy level is larger in the second half of the time window, since there the nodes start preserving their battery adopting longer aggregation intervals $I_a$. The confidence interval of the ten rounds is up to 2% as long as all the nodes have battery capacity, but can drop to 50% when the nodes start to drop out due to depletion. Figure 5.5b shows the number of active nodes in the network. A good detection accuracy and the minor energy consumption due to the lower CPU utilisation of the adapted IDS application has an impact on the lifetime of the network, although the linear discharge behaviour in both cases is still caused by the energy consumption of the wireless interface in ad hoc mode (recall this mode has a strong impact on the consumption due to the constant idle listening [86]). This causes the nodes to have a similar discharge behaviour, which results in a sharp drop of the number of active nodes, as can be observed in Figure 5.5b.

![Figure 5.6: Survivability performance in the draining attack](image)

(a) Number of packets sent  
(b) k-delivery rate

The impact of the adaption on the survivability performance during the same attack is shown in Figure 5.6. Two metrics are used to measure the network performance: the packet transmission rate and the packet k-delivery rate. The first indicates the number of transmitted packets (including data and signalling packets) during the interval of study, which is useful to analyse the impact of the attack on the bandwidth usage. The second metric shows the performance of the network as number of packets successfully delivered to $k$ nodes over the interval of study. We expect that if adaptation of the interval is successful the results of attack detection measured in terms of network performance are not any worse than when we are not adapting. In Figure 5.6a, we can see a peak on the number of transmissions at the beginning of the drain attack (at second 2100). This is caused by some detection latency, that leaves some bogus messages being disseminated in the network. This number is slightly higher in the adaptive case, but afterwards the number of messages sent during the attack is similar to the case in which the
5.4. Evaluation

interval is not adapted, meaning that the adaptation does not decrease the detection accuracy. In Figure 5.6b, we can see how the delivery ratio is also very similar to the non-adaptive case, indicating that the overall performance is preserved.

![Figure 5.7: Energy consumption and number of active nodes in the grey hole attack](image)

The results of the adaption on the greyhole attack are presented in Figures 5.7 and 5.8. As in the case of the drain attack, the average available energy in the network is higher in the adaptive case compared to the non-adaptive one, as shown in Figure 5.7. However, a more accurate analysis of the survivability performance should be undertaken since in this attack scenario a bad detection (i.e. a persistent greyhole attack) could give energy saving by disrupting the attempted dissemination flows. As shown in Figure 5.8, there is again some latency at the beginning of the attack, in which we can see that a sharp decrease in the number of transmissions is caused by the malicious nodes causing packets to be dropped. After that, however, the number of transmissions is greater in the adaptive case compared to the non-adaptive one, showing that a longer interval of aggregation in this case is
5. ONLINE PARAMETER ADAPTATION OF INTRUSION DETECTION

a benefit and improves the detection accuracy. Figure 5.8b in fact shows that the number of deliveries is higher after second 2200.

For a more realistic scenario, we studied the case in which nodes started with random initial battery levels. This test shows how heterogeneity on adaptation and attack response impacts our major metrics: the available energy and the survivability of the network. In this case, all the nodes are assigned a random initial energy between 100J and 500J.

![Image](a) Average energy in the network ![Image](b) Number of active nodes

**Figure 5.9:** Energy consumption and number of active nodes in the drain attack with random initial energy levels

![Image](a) Number of packets sent ![Image](b) k-delivery rate

**Figure 5.10:** Survivability performance in the drain attack with random initial energy levels

Again, Figure 5.9 shows that adaptation provides energy efficiency while Figure 5.10 shows that the global impact on the network survivability is further improved.
5.5 Closing remarks

This chapter described the design and implementation of an on-line energy-based adaptation component for an attack survivability framework in the context of mobile ad hoc communication, where the cyber and physical aspects of the system are extremely intertwined. The results, based on a model for CPU energy consumption in network simulations, have shown that the adaptation component gives an extension of about 14% of lifetime without significantly degrading the survivability performance. Further improvements can address proactive energy-based approaches, as for example foreseeing the depletion based on the current workload and adapting in advance. Although our simple CPU energy consumption model allowed us to study the network lifetime and understand the impact of adaptation, a more detailed CPU model should also be included in network simulators to enable more realistic CPU usage and energy consumption simulation.
Chapter 6

Conclusion

In dealing with security challenges in cyber physical systems our hypothesis was that anomaly detection could be applicable as a second line of defence in both the cyber and physical domains. This hypothesis was explored by investigations in three application domains.

The first domain of application has been the physical layer in water distribution systems, described in Chapter 3. These systems are monitored by water quality sensors that provide chemical properties of the water which are processed and used to discover contaminations. Despite the fact that specialised sensors and detectors of specific pollutant substances are available, the water security research community is seeking to find generalised event detectors that may help to detect any kind of contamination event. Once again, there is a need of both misuse detection to match specific known cases, and anomaly detection to discover any patterns that do not conform with normal behaviour.

The introduction of this system is challenging since the chemical properties of the water can change along the time depending on its source and can be confused as a contamination event. Nevertheless, the use of a learning based anomaly detection technique, which allows the characterisation of all the variations of the system normality, has proved to be effective. Besides, additional features based on sliding windows and derivatives of the data analysed have been introduced to improve the efficiency of the solution under certain circumstances. The performance of the approach has been analysed using real data of two water stations together with synthetic contaminants superimposed with the EDDIES application provided by EPA. The results, in terms of detection rate and false positive rate, have shown some contaminants are easier to detected than others. The sensitivity of the anomaly detector has also been been studied by creating new testing data sets with different contaminant concentrations. The results have shown that with more contaminant concentration the detector obtains higher detection rates with low false positive rates. Domain knowledge is however required to draw conclusions about the performance related to the increasing level of concentrations.

The inaccuracy of the data provided in one of the stations has negatively af-
ected the performance. This is a typical issue that arises when dealing with a physical domain where sensors can be subject to faults. We have discussed the potential of classical dependability measures to improve the detection outcome, but the drawback is the cost of additional hardware. This problem could also be addressed with software by estimation of the features trend in the preprocessing phase, using expectation-maximisation algorithms for instance.

The latency of the detection, a very important factor, has also been analysed, showing reasonable results that are again qualitatively improved as the contaminant concentration is increased.

Next, anomaly detection has been explored as a security mechanism in a cyber layer of a CPS. In Chapter 4 we have focused our attention towards smart metering infrastructures, the prominent example of CPS where cyber and physical layer are strictly related to each other. In our case study, although confidentiality, authenticity, accountability, integrity and privacy are provided by the use of trusted computing technology embedded into smart meters, some vulnerabilities persist and real-time monitoring for cyber and physical tampering attacks is still a security solution that must be considered when designing new smart meters. Therefore, we have explored deployment of a lightweight embedded anomaly detection architecture that takes into account both cyber and physical domains and implemented and evaluated part of this architecture on a smart meter prototype. The evaluation performed on attacks in pseudo-real settings has shown that our anomaly detection approach is able to efficiently detect several types of cyber attacks without emitting any false positives. The cyber domain in this case has been easier to model for normality, since the set of messages that can be exchanged through the bus is bounded and primarily because the application generating the communication events has a regular behaviour. Misuse detection could of course be able to efficiently detect the same set of attacks with less effort, but the advantage of anomaly detection is that not any information about attacks is modelled.

The work discussed in thesis confirms the known research insights that challenge anomaly detection:

1. Difficulty of the definition of the normal behaviour of the monitored system,
2. Difficulty of the definition of the boundaries between normal and anomalous behaviour
3. Lack of availability of labelled training data

With regard to challenges 1 and 2, we have seen that feature selection and parameter tuning play an important role in the success of applicability. Although we have proven good detection/alarm rate trade-offs, the definition of the boundaries between normality and attack is still a challenge: we have selected them based on a pre-study phase with attack information, using a training-validation-testing approach. Will the trade-off be effective when new attack types occur? This is particularly challenging when change of normality occurs.
6. CONCLUSION

Our work on online adaptation addresses challenge 4 by tuning of the parameters of intrusion detection. Chapter 5 described the design and implementation of an on-line energy-based adaptation component for an attack survivability framework in the context of mobile ad hoc communication. This component adjusts the trade-off between attack response time and energy consumption based on the available energy. Intrusion detection parameters were initially tuned based on detection-latency tradeoffs, again using training, validation and testing on some examples of attack. Our work on adaptive trade-off selection targeted improvements towards that aspect: a set of parameters can be devised in a phase prior to deployment, but online adaptation selects the parameters to adjust the detector to the current condition. The results have shown that adaptation increases the network lifetime. However, the adaptation strategy is based on fixed energy intervals and is rather simple. Smarter strategies to forecast the energy depletion and adapt in advance could be adopted in order to further improve network availability.

6.1 Future work

One of the remaining challenges in anomaly detection is the availability of labelled data. This creates difficulties for research community. Already with trials in three application domains of cyber-physical systems it has been difficult to obtain data to build the normality model. In the first domain real data was available but had sensor-related issues and we suffered from lack of domain knowledge. In the second domain capturing real consumption profiles was hard due to privacy issues. In the third domain we could not even get rudimentary data, but had to rely on simulation.

Availability of data that is affected by concept drift, in order to address the challenge of dealing with change of normality, is even a harder problem. However, dealing with concept drift is an interesting challenge that is worth to study, especially in cyber-physical systems, that are likely to be subject to this phenomenon. Creation on synthetic datasets, in particular for the physical domain, would be a way to perform such studies. In the water scenario, we have seen how synthetic contaminations can be superimposed by a software; with the same principle, one could synthetically extend the real dataset with estimated values and emulate a possible change of normality.

Concept drift research is well established in the context of machine learning; extensions of similar techniques in adversarial environment are possible research directions. The clustering algorithm adopted in the first two domains has retraining and forgetting capabilities which can in principle be exploited to autonomously adapt the normality model to changes over time. The adaptation strategies included in the existing package are quite rudimentary, relying on human intervention for incremental learning (when a new cluster should be added) and using simple windowing for forgetting of unused clusters. Future research will focus on algorithms for online self-adaptation of normality to cope with concept drift.

The thesis has focused on ad hoc communication as a third domain of study. In such networks, the chaotic environment where nodes mobility, miscommunication
6.1. Future work

and constrained resource availability (energy, bandwidth, etc.) makes concept drift
difficult to capture from the point of view of a single node. Since the survivability
framework is deployed in an adversarial environment, an attacker could learn the
adaptive behaviour of the node and act against it. A game-theoretical approach
to adaptation therefore can be devised to improve the survivability and lifetime in
presence of two types of players in the network, attackers and defenders.
Bibliography


[10] J. White, S. Clarke, C. Groba, B. Dougherty, C. Thompson, and D. Schmidt, “R&d challenges and solutions for mobile cyber-physical applications and
BIBLIOGRAPHY


BIBLIOGRAPHY


BIBLIOGRAPHY


Department of Computer and Information Science
Linköpings universitet
Licentiate Theses
Linköpings Studies in Science and Technology
Faculty of Arts and Sciences

No 17  Vojin Plavšić: Interleaved Processing of Non-Numerical Data Stored on a Cyclic Memory. (Available at: FOA, Box 1165, S-581 11 Linköping, Sweden. FOA Report B30062E)
No 73  Ola Strömberg: A Structure Editor for Documents and Programs, 1986.
No 177 Peter Åberg: Design of a Multiple View Presentation and Interaction Manager, 1989.
No 181 Henrik Eriksson: A Study in Domain-Oriented Tool Support for Knowledge Acquisition, 1989.
Niclas Andersson:
Compilation of Mathematical Models to Parallel Code, 1996.

Johan Jenvaldi:
Simulation and Data Collection in Battle Training, 1996.

Niclas Ohlsson:

Mikael Eriksén:

Jörgen Lindström:
Chefer’s användning av kommunikationsteknik, 1996.

Esa Falkenroth:
Data Management in Control Applications - A Proposal Based on Active Database Systems, 1996.

Niclas Wahlöö:
A Default Extension to Description Logics and its Applications, 1996.

Annika Larsson:

Ling Lin:

Rego Granlund:

Peter Ingels:

Per-Arne Persson:

Jonas S Karlsson:
A Scalable Data Structure for a Parallel Data Server, 1997.

FiF-a 4
Carita Åbom:
Videomötesteknik i olika affärs situationer - möjligheter och hinder, 1997.

FiF-a 6
Tommy Wedlund:

Silvia Coradeschi:

Jan Ollinen:

David Byers:

Fredrik Öberg:
Declarative Error Diagnosis of GAPLog Programs, 1997.

Gunilla Iverfors:

Jens-Olof Lindh:
Analysing Traffic Safety from a Case-Based Reasoning Perspective, 1997

Jukka Mäki-Turja:

Juha Takkinnen:

Man Lin:

Mats Gustafson:
Bringing Role-Based Access Control to Distributed Systems, 1997.

Boris Karlsson:
Metodanalys för förståelse och utveckling av systemutvecklingsverksamhet. Analys och värdering av systemutvecklingsmodeller och dess användning, 1997.

Marcus Bjäreland:

Jan Håkégard:

Per-Ove Zetterlund:
Normering av svensk redovisning - En studie av tillkommsten av Redovisningsrådets rekommendation om koncerntredovisning (RR01:91), 1998.

Jimmy Tjäder:

Ulf Melin:

Tim Heyer:

Patrik Häglund:

Marie-Therese Christiansson:
Inter-organisatorisk verksamhetsutveckling - metoder som stöd vid utveckling av partnerskap och informationssystem, 1998.

Christina Wennestam:
Information om immateriella resurser. Investeringar i forskning och utveckling samt i personal inom skogsindustrin, 1998.

Joakim Gustafsson:

Henrik André-Jönsson:

Erik Larsson:

Carl-Johan Westin:

Åse Jansson:

Thomas Paterson:

Anders Bäckström:

Ulf Seierroth:

Fredrik Öberg:

Jonas Mellin:

Joakim Eriksson:

Bengt E W Andersson:
Samverkande informationssystem mellan aktörer i offentliga åtaganden - En teori om aktörsärenden i samverkan om utbyte av information, 1998.

Pawel Pietrzak:
Static Incorrectness Diagnosis of CLP (FD), 1999.

Tobias Ritzau:

Anders Ferntoft:
Elektronisk affärs samkopplande - kontaktkontakter och kontaktprocesser mellan kunder och leverantörer på produktsmarknader, 1999.

Jo Skåmedal:
Arbeta på distans och arbetsformens påverkan på resor och resmööster, 1999.

Johan Alvehus:


CHARLOTTE BJÖRKEGREN: Learning for the next project - Bearers and barriers in knowledge transfer within an organisation, 1999.


SVEN BERGMAN: Managerial communication in telework, 2000.


JEAN PAUL MEYNARD: Control of industrial robots through high-level task programming, 2000.


STEFAN HOLGERSSON: IT-system och filtrering av verksamhetskunskap - kvalitetsproblem vid analyser och beslutsfattande som bygger på uppgifter hämtade från polisens IT-system, 2001.


FREDRİK BÅG: Ett dynamiskt perspektiv på individuella skillnader av huristisk kompetens, intelligens, mentala modeller, mål och konfident i kontroll av mikrovärlden Moro, 2002.


ROBERT SEVENIUS: On the instruments of governance - A law & economics study of capital instruments in limited liability companies, 2002.


PETER BUNUS: Debugging and Structural Analysis of Declarative Equation-Based Languages, 2002.


No 1190 David Dinka: Role and Identity - Experience of technology in professional settings, 2005.
No 1191 Andreas Hanson: Increasing the Storage Capacity of Recursive Auto-associative Memory by Segmenting Data, 2005.
No 1197 Andreas Käll: Oversättningar av en managementsmödel - En studie av innehållet av Balanced Scorecard i ett landsting, 2005.
No 1198 He Tan: Aligning and Merging Biomedical Ontologies, 2006.
No 1230 Erik Kuiper: Mobility and Routing in a Delay-tolerant Network of Unmanned Aerial Vehicles, 2008.
No 1232 Martin Karresand: Completing the Picture - Frameworks and Back Again, 2008.
No 1237 Mirko Thorstensson: Using Observers for Model Based Data Collection in Distributed Tactical Operations, 2008.
No 1239 Maria Holmqvist: Word Alignment by Re-using Parallel Phrases, 2008.
Mohammad Saifullah: Exploring Biologically Inspired Interactive Networks for Object Recognition, 2011
Anna Vapen: Contributions to Web Authentication for Untrusted Computers, 2011.
Kristian Stavåker: Contributions to Parallel Simulation of Equation-Based Models on Graphics Processing Units, 2011.