Towards Quantitative Cybersecurity Risk Assessment in Transportation Infrastructure

Final Report
July 31st, 2022

Principal Investigator: Dr. Song Han

Computer Science and Engineering
University of Connecticut

Authors
Song Han

Sponsored By
Transportation Infrastructure Durability Center

A report from
University of Connecticut
Computer Science and Engineering
371 Fairfield Way, Unit 4155
Storrs, CT 06269-4155
Phone: (860)-486-8771
Website: https://cps.cse.uconn.edu/

About the Transportation Infrastructure Durability Center

The Transportation Infrastructure Durability Center (TIDC) is the 2018 US DOT Region 1 (New England) University Transportation Center (UTC) located at the University of Maine Advanced Structures and Composites Center. TIDC's research focuses on efforts to improve the durability and extend the life of transportation infrastructure in New England and beyond through an integrated collaboration of universities, state DOTs, and industry. The TIDC is comprised of six New England universities, the University of Maine (lead), the University of Connecticut, the University of Massachusetts Lowell, the University of Rhode Island, the University of Vermont, and Western New England University.

U.S. Department of Transportation (US DOT) Disclaimer

Acknowledgements

Funding for this research is provided by the Transportation Infrastructure Durability Center at the University of Maine under grant 69A3551847101 from the U.S. Department of Transportation's University Transportation Centers Program.

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient Catalog No.
4 Title and Subtitle Towards Quantitative Cybersecurity Risk Assessment in Transportation Infrastructure		5 Report Date: July 31st, 2022 6 Performing Organization Code
7. Author(s) Song Han ORCID:0000-0002-1491-7675		8 Performing Organization Report No.
9 Performing Organization Name and Address University of Connecticut Computer Science and Engineering 371 Fairfield Way, Unit 4155 Storrs, CT 06269-4155		10 Work Unit No. (TRAIS) 11 Contract or Grant No. 69A3551847101
12 Sponsoring Agency Name and Address Transportation Infrastructure Durability Center University of Maine		13 Type of Report and Period Covered Final Report 10/1/2018 – 6/30/2022 14 Sponsoring Agency Code

15 Supplementary Notes

16 Abstract

Along with the emergence of Intelligent Transportation Systems (ITS), cybersecurity in transportation infrastructure plays a key role in ensuring transportation safety and enhancing the infrastructure durability and longevity. Cybersecurity protection in transportation infrastructure however is challenging because ITS are in essential cyber-physical systems where cyber systems and components (computing facilities, communication networks, etc.) are used to control physical systems and components (vehicles, traffic lights, etc.), and impact human individuals (motorists, riders, etc.) as well. It is of critical importance to identify the vulnerabilities of existing cyber infrastructures deployed in ITS, the consequences of an attack, and the resulted safety-related incidents along with their impact on the infrastructure durability/longevity. Considering the broad range of cyberinfrastructures in ITS, this project focuses on the cybersecurity analysis for wireless networks that are deployed in ITS for various sensing applications. We aim to gain deep understandings on i) the challenges and design goals of the wireless networks deployed in representative ITS, and ii) the cybersecurity requirements associated with those wireless solutions. We perform a comprehensive security analysis on those wireless networks in a layer-by-layer fashion to study i) the vulnerabilities and potential attacks that may happen in each layer of the protocol stacks, ii) the advantages and disadvantages of existing countermeasures, and iii) potential solutions to further improve the cybersecurity protection. To counter the attacks on the network topology at the network layer, we develop a suite of Intrusion Detection Systems (IDS) to detect attacks against the routing protocol that could disrupt the stability and availability of the wireless networks. These IDS include ARM (Authenticated Rank and Routing Metric), ARM-Pro and FORCE (Forged Rank and Routing Metric Detector). All these proposed IDS are implemented in the simulation tools and on the real-life testbed for design validation and performance evaluation.

17 Key Words	18 Distribution Statement		
Cybersecurity, risk assessment, intelligent transportation systems, wireless networks, intrusion detection systems.	No restrictions. This document is available to the public through		
19 Security Classification (of this report) Unclassified	**20 Security Classification (of this page)** Unclassified	**21 No. of pages** TBA	**22 Price**

Form DOT F 1700.7 (8-72)

Contents

List of Figures

List of Tables

List of Key Terms

ITS	Intelligent Transportation Systems
CPS	Cyber-Physical Systems
IIoT	Industrial Internet of Things
WSN	Wireless Sensor Networks
IDS	Intrusion Detection Systems
5G NR	5G New Radio
LTE	Long Term Evolution
QoS	Quality of Service
PHY	Physical Layer of a Communication Protocol
MAC	Medium Access Control
DLL	Data Link Layer of a Communication Protocol
NWK	Network Layer of a Communication Protocol
TL	Transport Layer of a Communication Protocol
APP	Application Layer of a Communication Protocol
RSS	Received Signal Strength
PDR	Packet Delivery Rate
PER	Packet Error Rate
ECC	Error Correcting Codes
ICMP	Internet Control Message Protocol
DSSS	Direct-sequence spread spectrum
DCF	Distributed Coordination Function
RTS/CTS	Request to Send / Clear to Send
PUF	Physical Unclonable Functions
DoS	Denial-of-Service
DDoS	Distributed Denial-of-Service
AoA	Angle of Arrival
TDoA	Time Difference of Arrival
ARM	Authenticated Rank and Routing Metric
FORCE	Forged Rank and Routing Metric Detector

IETF	Internet Engineering Task Force
6TiSCH	IPv6 over the TSCH mode of IEEE 802.15.4e
RoLL	Routing over Low-Power and Lossy Links
6LoWPAN	Low-Power Wireless Personal Area Networks
CoAP	Constrained Application Protocol
DODAG	Destination Oriented Directed Acyclic Graph
DIO	DODAG Information Object
DAO	DODAG Advertisement Object
DIS	DODAG Information Solicitation

Abstract

Along with the emergence of Intelligent Transportation Systems (ITS), cybersecurity in transportation infrastructure plays a key role in ensuring transportation safety and enhancing the infrastructure durability and longevity. Cybersecurity protection in transportation infrastructure however is challenging because ITS are in essential cyber-physical systems where cyber systems and components (computing facilities, communication networks, etc.) are used to control physical systems and components (vehicles, traffic lights, etc.), and impact human individuals (motorists, riders, etc.) as well. It is of critical importance to identify the vulnerabilities of existing cyber infrastructures deployed in ITS, the consequences of an attack, and the resulted safety-related incidents along with their impact on the infrastructure durability/longevity. Considering the broad range of cyberinfrastructures in ITS, this project focuses on the cybersecurity analysis for wireless networks that are deployed in ITS for various sensing applications. We aim to gain deep understandings on i) the challenges and design goals of the wireless networks deployed in representative ITS, and ii) the cybersecurity requirements associated with those wireless solutions. We perform a comprehensive security analysis on those wireless networks in a layer-by-layer fashion to study i) the vulnerabilities and potential attacks that may happen in each layer of the protocol stacks, ii) the advantages and disadvantages of existing countermeasures, and iii) potential solutions to further improve the cybersecurity protection. To counter the attacks on the network topology at the network layer, we develop a suite of Intrusion Detection Systems (IDS) to detect attacks against the routing protocol that could disrupt the stability and availability of the wireless networks. These IDS include ARM (Authenticated Rank and Routing Metric), ARM-Pro and FORCE (Forged Rank and Routing Metric Detector). All these proposed IDS are implemented in the simulation tools and on the real-life testbed for design validation and performance evaluation.

Chapter 1: Introduction and Background

1.1 Project Motivation

Along with the emergence of Intelligent Transportation Systems (ITS) [2], cybersecurity in transportation infrastructure plays a key role in ensuring transportation safety and enhancing the infrastructure durability and longevity. Cybersecurity protection in transportation infrastructure however is challenging because ITS are in essential cyber-physical systems where cyber systems and components (computing facilities, communication networks, etc.) are used to control physical systems and components (vehicles, traffic lights, etc.), and impact human individuals (motorists, riders, etc.) as well. It is of critical importance to identify the vulnerabilities of existing cyber infrastructures deployed in ITS, the consequences of an attack, and the resulted safety-related incidents along with their impact on the infrastructure durability/longevity. In addition, organizational complexity of the transportation systems, potential cascading effects to dependent infrastructures, and the ever-growing deployment of novel computing and communication systems to existing legacy systems compound the problem.

Considering the broad range of cyberinfrastructures in ITS, it is not feasible to perform comprehensive security studies for all these cyberinfrastructures. Instead, in this project we focus on the cybersecurity analysis for wireless networks that are deployed in ITS for various sensing and control applications. As we have witnessed in recent years, many wireless technologies have been developed or under development to meet the increasing needs of various wireless communications in ITS [3]. These technology advances include but are not limited to variations of WiFi protocols (e.g., 802.11p), LTE/5G NR, wireless sensor networks, wireless Mesh/Ad hoc networks, mobile IP, smart antenna, cognitive radio, and so on. These technologies have already or will significantly impact the design and operation of ITS, which aims to effectively provide higher vehicles safety, traffic management, and communications among vehicles and transportation infrastructure. Figure 1 shows several representative applications of these wireless networks, including parking lots monitoring, bridge heath monitoring, and traffic monitoring and control in roadways and intersections.

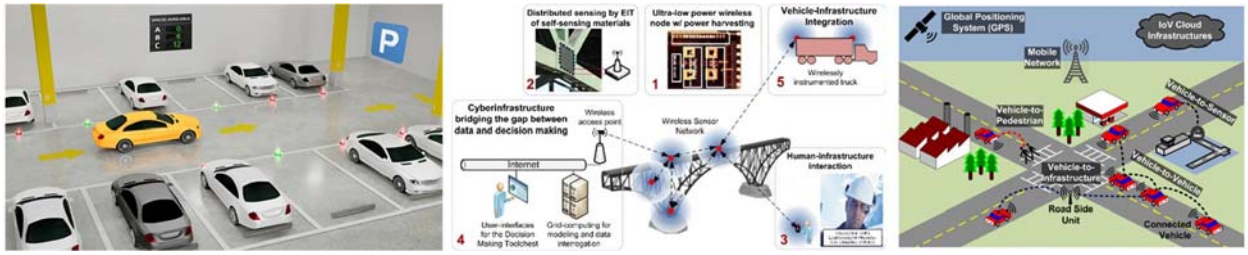


Figure 1. Examples of wireless sensing and control applications in ITS. (Photo Credit: Google)

Although the deployment of these wireless technologies provides significant benefits for ITS, most of the deployed wireless devices are resource constraints in terms of their computing capability, communication bandwidth, memory size and power supply. These wireless networks are usually deployed in remote places and left unattended, and they should be equipped with security mechanisms to defend against various attacks, such as node capture, physical tampering, eavesdropping, denial of service (DoS), etc. However, traditional security mechanisms with high

overhead are not feasible for these resource-constrained wireless nodes. Thus, security is one of the most challenging problems faced by these networks due to the following reasons: i) wireless communication is difficult to protect since it is realized over a broadcast medium, which adversary can easily eavesdrop on, intercept, inject, and alter transmitted data; ii) the sensor nodes may be deployed in a variety of physically insecure environments, then the adversary can steal nodes, recover their cryptographic material, and pose as authorized nodes in the network; iii) since sensor nodes are vulnerable to resource consumption attacks, the adversary can repeatedly send packets to drain a node battery and waste network bandwidth. Thus, secure transmission of sensitive information over the wireless network is essential. Meanwhile, the use of encryption or authentication primitives between two wireless devices requires an initial link key establishment process, which must satisfy the low power and low complexity requirements.

1.2 Research Objectives and Tasks

The research objective of this project is to gain deep understandings on i) the challenges and design goals of the wireless network systems deployed in representative ITS, and ii) the cybersecurity requirements associated with those wireless network solutions. Based on this knowledge, a comprehensive security analysis will be performed on those wireless networks in a layer-by-layer fashion to study i) the vulnerabilities and potential attacks that may happen in each layer of the protocol stacks, ii) the advantages and disadvantages of existing countermeasures, and iii) potential solutions to further improve the cybersecurity protection. Some of the proposed solutions will be implemented in simulation tools and/or real-life testbed for design validation and performance evaluation. This project is cross-disciplinary research which offers opportunities for the students and faculty member to bring expertise from different domains and study complex scientific and engineering problems in a collaborative manner.

Specifically, we performed the following three tasks in this three-year project.

Task 1: Context establishment. In this task, we study different use cases in representative ITS. In this project, instead of exploring the security issues for the cyberinfrastructure of ITS in general which is too broad in the scope, we focus on the cybersecurity analysis for wireless networks that are deployed in ITS to perform various sensing and control tasks. This task summarizes the challenges and deign goals of these wireless network systems and their associated cybersecurity requirements from a high level.

Task 2: Vulnerability and attack identification. This task performs a comprehensive security analysis on the wireless networks deployed in ITS in a layer-by-layer fashion to identify the vulnerability and potential attacks that may happen in the wireless protocol stack and discusses the root cause of each attack. In the physical layer (PHY), we study four types of attacks including jamming, tampering, eavesdropping and spoofing. In the data link layer (MAC), we study three types of attacks including collision, exhaustion and replay. In the network layer (NWK), we study six types of attacks including black-hole attack, gray-hole attack, sink-hole attack, wormholes, sybil, and flooding. We also study security-related reliability issues in the transport layer (TL) and the security issues in the application layer (APP) designed for resource-constrained devices and applications.

Task 3: Consequence identification and countermeasure study. Based on the identified attacks in each layer of the wireless protocol stack, this task studies the consequences of each attack and the pros and cons of their existing countermeasures. For some attacks, we propose solutions to further improve the effectiveness of the state of the art. More specifically, to counter the attacks on the network topology at the network layer of resource-constrained wireless networks, we develop a suite of intrusion detection systems (IDS) to detect attacks against the routing protocol that could disrupt the stability and availability of the wireless networks. These IDS include ARM (Authenticated Rank and Routing Metric), ARM-Pro and FORCE (Forged Rank and Routing Metric Detector). All these proposed IDS are implemented in the simulation tools and on the real-life testbed for design validation and performance evaluation.

1.3 Report Overview

The remainder of this report is organized as follows. Chapter 2 presents the design goals of the wireless network solutions to be deployed in representative ITS and summarizes the associated security requirements of these networks from high level. Chapter 3, Chapter 4 and Chapter 5 analyze the vulnerabilities, potential attacks and their countermeasures in the physical layer, data link layer and network layer of the wireless network stack, respectively. Readers are referred to our 78-page full technical report for all the details. Chapter 6 describes the proposed intrusion detection systems for protecting the stability of the network topology in resource-constrained wireless networks. Chapter 7 concludes the report and provides our recommendations.

Chapter 2: Design Goals and Security Requirements of Wireless Networks in ITS

In order to perform security analysis on the wireless networks deployed in ITS, it is essential to consider not only the security requirements, but also the design requirements of such networks. This is because some of those design requirements have direct influence on the security requirements of the network, and vice versa. For instance, if one uses an end-to-end secure channel to establish communication between a wireless sensor node and a central system, it will increase the overhead of that node, not only increasing the response time, but also consume more computing resources (e.g., MCU cycles and memory) on the node. In the following, we summarize some common design goals of the wireless networks deployed in ITS.

2.1 Design Goals of the Wireless Networks in ITS

Low-cost and small form factor: Wireless sensor nodes deployed in ITS are typically compact and low-cost which is an essential feature to accomplish large-scale deployment. Thus, the system designer of such wireless networks should consider the cost issue, such as ownership cost (packaging requirements, modifications, maintainability, etc.), implementation costs, replacement and logistics costs, as well as the per unit costs all together.

Scalable architectures and efficient protocols: Typically, the wireless networks deployed in ITS need to support heterogeneous applications with different requirements so that they can collaborate automatically. It is necessary to develop the flexible and scalable architectures that can

accommodate the requirements of all these applications in the same infrastructure. To enhance the system flexibility, robustness, and reliability, it is recommended to apply modular and hierarchical systems design principle. Also, to integrate into other systems, the interoperability with existing legacy solutions is required.

Resource-efficient design: Due to the constrained energy of wireless nodes, energy efficiency is an important design principle to maximize the network lifetime while providing the QoS required by the application. Energy saving solutions can be, and should be, achieved in every component of the network by integrating network functionalities with energy-efficient protocols, such as energy-efficient mode on MAC layer, energy-efficient routing on the network layer, etc.

Self-configuration and self-organization: Many wireless networks in ITS are subject to the topologies change, for example, caused by node failure/mobility and temporary power down. Large-scale wireless network deployments necessitate self-organizing architectures and protocols. With the use of self-configurable network architecture, new wireless nodes can be added to replace failed nodes in the field, and existing nodes can also be moved from one system to another without affecting the performance of the application.

Data fusion and local processing: In wireless networks deployed in ITS for the monitoring purposes, redundant sensor nodes are typically deployed to collect environmental data. In general, the raw data collect by those sensor nodes are very large in size, which consumes a lot of bandwidth. Instead of sending the raw data to the sink node directly, the sensor nodes can locally filter the sensed data based on the application requirements and transmit only the processed data, i.e., through in-network processing. Thus, only necessary information is transported to the end user and the communication overhead can be significantly reduced.

Adaptive network operation: Due to the dynamic nature of the wireless networks in ITS, their adaptability enables end users to cope with dynamic/varying wireless channel conditions in harsh environments. The network also needs to deal with different requirements. For instance, to balance the tradeoffs among resources, accuracy, latency, and time-synchronization requirements, adaptive signal-processing algorithms and communication protocols are required.

Time synchronization: In the wireless networks deployed in ITS, the collected data are typically time-sensitive. Thus, time synchronization is one of the key design goals for such networks to meet the deadlines of the application, especially for performing control tasks.

Fault tolerance and reliability: The wireless networks in ITS require that the sensed data should be reliably transferred to the sink node. However, there exists many security vulnerabilities. The programming data for sensor operation, command, and queries should be reliably delivered to target sensor nodes to assure the proper functioning of the network. The wireless transmission medium is easily subject to errors, which causes unreliable performance. Thus, data verification and correction and self-recovery procedures are critical to provide accurate results to the end user.

Availability and performance: As the wireless networks deployed in ITS are usually performing critical tasks, the data produced by the sensors must always be available in order to react to problematic situations and ensure the integrity of the whole system. There are in fact two dimensions of availability: one related to reliability, i.e., using the redundancy of the system to avoid single points of failure, and one related to security, i.e., existence of DoS attacks and use of self-healing mechanisms to provide the services even in the case of attacks/system failure.

In general, it is very challenging to meet all the above design goals simultaneously. Fortunately, most wireless network designs have different requirements and priorities on design objectives. This will require that the network designers to consider and balance the tradeoffs among the different parameters when design wireless network protocols and architectures.

2.2 Security Requirements of the Wireless Networks in ITS

In this section, we first present some general security requirements in computer networks. We then focus on those resource-constrained wireless networks deployed in ITS. The goal of security services in those networks is to protect the information and resources from attacks and misbehavior. Table 1 summarized the general security requirements in wireless networks [4].

Security Requirements	Definition
Confidentiality	Confidentiality ensures that a given message cannot be understood by anyone other than the desired recipients.
Integrity	Integrity ensures that a message sent from one node to another is not modified by malicious intermediate nodes.
Availability	Availability ensures that the desired network services are available even in the presence of denial-of-service attacks.
Authorization	Authorization ensures that only authorized sensor nodes can be involved in providing information to network services.
Authentication	Authentication ensures that the communication from one node to another node is genuine, that is, a malicious node cannot masquerade as a trusted network node.
Non-repudiation	Non-repudiation denotes that a node cannot deny sending a message that it has previously sent.
Forward Secrecy	A sensor node should not be able to read any further messages after it leaves the network.
Backward Secrecy	A joining sensor should not be able to read any previously transmitted message.

Table 1. General security requirements in wireless networks and their definitions.

In general, the security services in wireless networks are centered with cryptography. However, due to the resource constraints in wireless networks deployed in ITS, many existing secure algorithms are not practical for their applications. For these wireless networks, the properties associated with security can be more specific.

Attack Impact: Adversaries typically target subsystems of these networks that provide the biggest payoff. Therefore, it is necessary not only to identify the potential weak points, but also to understand the extent to which an attacker can manipulate the infrastructure once these weak points are subverted. As a property, it refers to the actual impact caused by an adversary that take control of a section of the network.

Secure Channel: In the general case that two devices that belong to the same system communicate, it is important to set up a secure channel that supports end-to-end integrity and confidentiality services. If the integrity of the data stream is not protected, then attacker can falsify any reading or alarm. Also, once the confidentiality of the information flow is assured, adversaries

will be unable to read any sensitive information. As a property, it refers to the type of machines and mechanisms, e.g., end-to-end secure channels, that are involved in the creation of a secure communication channel to support the systems confidentiality and integrity.

Authentication: It has two types authentication requirements for user and property. As for user authentication, the devices should be confident about the identity of the user that is requesting a certain operation. As a property, authentication is concerned with the location and the nature of the mechanisms and elements that can be used to prove the identity of a human user, i.e., whether the mechanisms are distributed or centralized.

Authorization: After the authentication processes to prove the identify of any user, either human user or a machine, the network may be necessary to check whether that user has the rights to access the information. Both the access to the data and the granularity of the data should be controlled. Beyond data, it is also necessary to monitor control operations. As a property, authorization deal with the types of mechanisms, credentials and tools that can be used to check whether a certain entity is authorized to perform an operation.

Accountability and detection: It is important to record the interactions with heterogeneous users who access the services provided by the wireless networks. By sorting all interactions, it can recreate security incidents and abnormal situations. In this way, the control/detection systems can detect specific attacks in real time. As a property, accountability and detection refers to the structure of the accountability subsystems (e.g., detection rules) and the mechanisms that can be used to analyze them.

Trust Management: Within one wireless network, there exists the situation that several nodes provide the same services for the purpose of redundancy. Moreover, various nodes can also collaborate with each other to provide better service. However, with the more redundant data, it needs to deal with the issues of uncertainty (i.e., which source data is the best and which node should be trusted and collaborated?). This issue typically would be fulfilled by a trust management system. As a property, trust management is related to the nature of the mechanisms that are used to (i) measure and share the reputation of the different elements of a network, and (ii) use those values as input when determining specific trust values.

Chapter 3: Physical Layer Security Analysis for Wireless Networks in ITS

The physical layer of the resource-constrained wireless networks in ITS is responsible for frequency selection, signal detection, carrier frequency generation, modulation, as well as data encryption [5]. It can provide an interface to transmit a stream of bits over physical medium. The two fundamental characteristics of the wireless communication medium namely broadcast and superposition, present different challenges to ensure reliable and/or secure communication in the presence of adversarial users [6]. Typically, the broadcast characteristic of WSN makes it much more difficult to shield transmitted signals from unintended recipients compared to wired communication, while the superposition nature can lead to the overlapping of multiple signals at the receiver. Generally, the adversarial users of a wireless network are modeled either as (a) a malicious transmitter, such as jammer, who tries to degrade the signal at the intended receiver, or

(b) an unauthorized receiver that tries to extract information from an ongoing transmission without being detected, such as eavesdropper.

The fundamental principle behind physical layer security is to exploit the communication channels and inherent randomness of noise to limit the amount of information that can be extracted in 'bit' level by an unauthorized receiver. Typically, the physical layer security can be classified into two groups [7]. One group focuses on passive attacks, such as eavesdropping. It refers to the processes of receiving/listening to the legitimate wireless channel illegally without authorization. The eavesdropper in the eavesdropping is usually passive, e.g. not propagating the signal, thus the legitimate transmitter or receiver in the communication cannot detect the potential eavesdropper. The other group is about the active attacks, such as jamming, tampering and spoofing. Each active attack has its own mechanisms. In general, the active attacks participate into the legitimate transmission, whose main purpose is to harm the wireless communication. In the following of this chapter, we summarize our findings on these passive and active attacks and their existing countermeasures. For the details, please refer to our full technical report [1].

3.1 Jamming Attacks and Solutions

Jamming is a type of attack which interferes with the radio frequencies that a network's node is using. It is defined as the disruption of the existing wireless mediums by decreasing the signal-to-noise ratio at receiver sides through the transmission of interfering wireless signals. Typically, jamming makes use of intentional radio interference to affect the wireless communications by keeping shared-wireless communication channel busy, causing the transmitter to back-off whenever it senses busy wireless channel, or corrupted signal received at the receivers [8]. Jamming attack mostly happens at the physical layer, however, it sometimes occurs during cross-layer attacks. Depending on the attack capability and strategy, such as radio transmitter power and location, a jammer can either have the same or different capabilities from legitimate nodes which it tries to attack. There are many different types of wireless jammers, which may be classified into the following five categories [9].

3.1.1 Classifications of Jamming Attacks

Constant Jammer: The constant jammer continuously transmits a jamming signal over the shared wireless medium. The jamming signal can have random waveform associated with the constrained power and limited bandwidth without following the specific protocol. The effect of a constant jammer is twofold: (a) it increases the interference and noise level for the sake of degrading the signal reception quality at the legitimate receiver; (b) it also causes a legitimate transmitter always find the wireless channel busy, which prevents the legitimate nodes from communicating with each other by causing the wireless channel to be constantly busy. This type of attack is energy inefficient and easy to detect, however, it is very easy to launch and can damage the network communications entirely.

Intermittent Jammer: The intermittent jammer intermittently transmits regular packets [10], instead of emitting random bits as in constant jammer, which transmits a jamming signal from time to time for the sake of interfering with the legitimate communication [11]. Contrary to the constant jammer, it aims at saving energy. It continuously switches between two transmission states: sleep phase and jamming phase. The intermittent jammer transmits for a certain time and

then sleeps for the remaining time. Compared to the constant jammer, the intermittent jammer can be energy efficient which is attractive for energy-constrained jammers.

Reactive Jammer: The reactive jammer starts to transmit its jamming signal only when it observes that a network activity by the legitimate node occurs on a certain channel [12]. It needs first to sense the wireless channel and upon detecting that the channel is busy, and then transmits a jamming signal for the sake of corrupting the data reception at the legitimate receiver. As a result, a reactive jammer targets on compromising the reception of a message. The success of a reactive jammer depends on its sensing accuracy regarding to the legitimate node's status. Compared to constant and intermittent jammers, reactive jammer is more energy efficient and more difficult to be detected because the packet delivery ratio (PDR) cannot be determined accurately in practice.

Adaptive Jammer: The adaptive jammer is implemented by an attack who can adjust its jamming power to any specific level required for disrupting the legitimate receiver [13]. Typically, the Received Signal Strength (RSS), in wireless communication systems, mainly depends on the time-varying fading. The attacker implementing the adaptive jammer should have the corresponding RSS knowledge of the targeted legitimate receiver for adapting its jamming power, so that it can adjust its jamming power. However, it is challenging for a jammer to obtain the RSS in practice, because the RSS of main channel typically varies in time and it is unknown to the potential jammer.

Smart Jammer: The smart jammer is assumed to have a good understanding of the upper-layer protocols and attempts to jam the vitally critical network control packets, instead of data packets, by exploiting the associated protocol vulnerabilities [14]. Compared to above four jammers, the smart jammer considers the upper-layer protocol, while above four jammers belong to the family of physical-layer jammers operating without considering any upper-layer protocol specifications. Typically, the smart jammer can achieve much higher energy efficiency and jamming effectiveness, which however, requires some prior knowledge, such as protocol parameters.

3.1.2 Solutions to Tackle Jamming Attacks

Jamming attacks are very harmful DoS attacks, it is important to have effective detection and countermeasure against them. In the following, we summarize some of these existing techniques, in terms of attack model and detection metric, for detection and countermeasures.

Solutions for Constant/Intermittent Jammer: To detect the potential presence of a constant or intermittent jammer, the basic idea is to identify the abnormal signals received at the legitimate receiver [9] [15]. In literature, there exist certain already documented statistical tests that can be used to exploit for the detection of the constant jammer, such as received signal strength (RSS), carrier sensing time (CST), and packet error rate (PER), etc. For the details of these tests, please refer to our technical report [1]. One way to defend against jamming attack is frequency hopping, which is a well-known classic anti-jamming technique [16] [17] [18]. Frequency hopping rapidly changes the carrier frequency with the aid of a pseudo-random sequence generator known to both the transmitter and receiver. The frequency hopping can be either proactive, proactively performing channel switching, and reactive, only performing channel switching in the presence of a detected jamming. In general, frequency hopping is highly resistant to jamming attacks, if the jammer has no knowledge of the pseudo-random hopping pattern. Typically, cryptographic primitives are used for generating the pseudo-random hopping pattern based on the secret key.

Solutions for Reactive jammer: To detect reactive jammer, it is not efficient to observe the CST since the reactive jammer only affects the reception's activity without influencing the transmitter to access the wireless channel. However, it still can use the statistical measurements of RSS and PER to detect the reactive jammer, in which the abnormal values of RSS and/or PER indicate the presence of a reactive jammer. The frequency hopping technique can be effective way against a reactive jamming attack, if the hopping rate is sufficiently high. Another effective way to prevent reactive jammer is to assist the legitimate node in becoming undetectable, in this case the jammer keeps in its sleep mode. Direct-sequence spread spectrum (DSSS) uses the techniques that spread the radio signal over a wide frequency bandwidth. In DSSS, the signal has a very low power spectral density, even below the background noise level, which is difficult to detect for reactive jammer. In this way, it is unable to track the legitimate traffic activity for reactive jammer, thus it cannot disrupt the wireless transmission.

Solutions for Adaptive jammer: It is insufficient to detect the adaptive jamming using only one of statistic measurements of RSS, CST and PER since it can adjust its jamming power to accommodate its existence. A method is proposed in [15], called consistency check, to detect the adaptive jamming, relying on the joint use of both RSS and PER measurement to detect adaptive jamming. Another simple and effective defense strategy to adaptive jammer is to use the channel suffering and spatial retreating [15]. Channel suffering countermeasure provides a migration to another channel when a jammer is detected, and then blocks communication on a channel. Spatial retreating moves the jammed nodes from the location where they experience jamming to another safe location. For spatially retreating to a new position, it is crucial to accurately determine the jammer's position, so that the victims can move away from the jammed area.

Solutions for Smart jammer: The smart jammer needs to know the specific upper-layer parameters to launch the jamming attack, indicating that smart jammer is a upper-layer specific attack. A game-theoretic framework was formulated in [19] for modeling the interactions between a smart jammer and the protocol function. In the game theoretic model, it uses a clustering algorithm to identify whether a node belongs to a normal cluster (non-jammed) or anomalous cluster (jammed) based on four features: the retransmit RTS, retransmit DATA, carrier sensing failure count, and network allocator value. This method generalizes the physical-layer frequency hopping solution, which allows legitimate nodes to hop across various protocol parameters what the jammer may exploit using the specific upper-layer parameters.

3.2 Tampering Attacks and Solutions

Tampering involves the deliberate altering or adulteration of a node, package, or communication system. In wireless networks, tampering is one of methods to produce the compromised nodes in physical layer. Typically, an adversary can tamper with nodes physically, interrogate and compromise these nodes. The adversary, after tampered the node, can gain full control of these nodes and try to extract sensitive information, such as secret key shared between nodes when using symmetrical encryption. Specially, on one hand, an adversary may modify the functionality of tampered node, which may further bring severe damage to the network in various ways, such as message corruption, injection of bogus data, misrouting information packets. On the other hand, the adversary may modify the secure communication algorithm or cryptographic secret keys and inject the compromised node in the network. The tampering attacks can be classified into two categories: invasive attacks and non-invasive attacks [20]. For the invasive attacks, they require

access to the hardware components of sensor nodes, such as chips, and need high-tech expensive equipment used in semiconductor manufacturing; while, for the non-invasive attacks, they are easier than invasive and require less times to launch attacks.

Tamper protection falls into two categories: passive and active [21], which can be applied to the physical or electrical design. Active tamper protection involves the special hardware circuits within the sensor node to prevent sensitive data from leakage in wireless environment. Passive tamper protection includes those that do not require energy and includes technologies that protect a circuit or provide detection, such as tamper seals, protective coating. In general, active protection may require extra circuitry which add the cost to a sensor node and consume more energy, thus, due to the resource constraints, active protection may not be typically found in resource-constrained wireless networks.

To mitigate tampering attacks, the physical locations of the wireless nodes need to be hidden from unauthorized adversaries. Lack of physical protection of these nodes may lead to node tampering attacks, which further result in a severe risk of the network. We must first to detect the tampered sensor nodes, and then design the corresponding defense mechanisms.

Tampering Detection: The authors in [22] proposed one method to detect the tampering attacks, called Node Attack Detection (NAD) block. The NAD block can help identify the state or severity of the attack on a node. According to NAD block, the severity of the attack can be divided into three states and each state has its corresponding behaviors. Identification of these states eliminates the possibility of destroying a legitimate node mistakenly. However, the severity of the tampering attacks may vary from application to application. The authors in [23] proposed a tamper-aware authentication framework for resource-constrained wireless nodes. In this framework, it proposes the use of an interrupt-based procedure that makes use of low-cost components to generate interrupt signals upon detection of a tamper. This interrupt triggering device can be very cheap and light, such as a light dependent resistor (LDR), or a spring switch in detecting tampering. Compared to the use of specialized crypto-processors, the use of low-cost triggers is more feasible, due to its cost, to ensure the security. In the tamper detection phase, it uses the state of interrupt pin and a flag of hard interrupt to indicate the existence of tampering attacks.

Besides the tampering detection in the physical layer, several methods use the upper-layer program or software to detect the potential tampering. For example, the author in paper [24] use the run-time result checking to detect the tampering attacks; and the authors in paper [25] apply the program integrity verification process to achieve the soft tamper-proofing.

Tampering Defense: It is not practical to employ direct surveillance on hundreds and thousands of sensor nodes to avoid tampering attacks, while designing tamper-resistant sensor nodes may not be economically viable. The authors in [22] proposed one method to defense the tampering attacks, called Defence Advocating Measure (DAM) block. The DAM is an intelligent self-destructive protocol that initially identifies the severity of the attack and then takes defense measurements against it by erasing vital information from memory of the wireless nodes. In each node, it contains one table, Vital Memory Address Table (VMAT), when detecting the tampering attacks, it may initialize the VMAT into zero, or called zero overwritten, through a special procedure. The zero

overwritten operation simply replaces the vital information bits by zero bits, which makes the adversary could not extract the actual information.

One of the important tasks of tampering defense is to provide tamper-proof security to physically unprotected devices with limited resources. While existing security primitives may be enough for Internet, but they are not suitable for resource-constrained wireless networks. Modern security protocols need to be immune to physical and tampering attacks. Physical Unclonable Functions (PUF) provide a unique way to identify integrated circuits, especially sensor nodes [26]. PUF is a physical entity that is embodied in a physical structure and is easy to evaluate but hard to predict. Typically, an individual PUF device must be easy to make but practically impossible to duplicate, even provided the exact manufacturing processes [27]. Sensor PUF is a PUF that comingles sensing with the challenge response processing of a PUF. In general, the sensor PUF extends the functionality of conventional PUF to provide the physical layer's authentication, unclonability, and verification of the sensor nodes [28], which can potentially prevent the tampering attacks.

3.3 Eavesdropping Attacks and Solutions

Due to the openness of wireless signal propagation, any network node within the radio range can capture the signal. Typically, there exist two kinds of nodes in wireless systems: legitimate nodes and unauthorized/illegitimate nodes. As the legitimate nodes follow the designed communication standards, the neighboring nodes can obtain meaningful information from the captured signal. However, an unauthorized node may eavesdrop the data transmission and access the credential information from the wireless channels. By doing this, the eavesdropping attack violates the confidentiality requirement of secure communication. For eavesdropping attacks, it usually has two properties: 1) It can be quickly launched. With the technological advancement, an off-the-shelf radio module can be made as an eavesdropping device with a short line of script; 2) the attack cannot be easily detected by communication systems because the adversary does not expose its activity. In short, eavesdropping attacks are difficult to detect. However, they can be mitigated by advanced cryptography. Encrypting packets hinders unauthorized nodes from reading data easily. That means, we allow the existence of eavesdropping, but use other techniques to prevent the useful information from leakage.

It is typically assumed that, for the eavesdropper, there is no limitations on its computational resources or network knowledge in wireless networks. That means, the eavesdropper is assumed to have unbounded computational power, knowledge of the transmit coding scheme, and access to an identical copy of the signal at the intended receiver. With unbounded computational power, the eavesdropper can crack the encrypted wireless communications with the aid of exhaustive key search, known as the brute-force attack. It is thus necessary to design effective mechanisms to protect the confidentiality of wireless networks against the eavesdropping attack.

Physical-Layer Secret Key Generation: Ideally, based on the secret key generation, the wireless channel is envisioned as an effective means for confidential communications by exploring the random physical characteristics of the radio propagation process [29]. However, a real-life wireless network consists of a transmitter node communicating with the sink node in the presence of the eavesdropper which tries to wiretap the credential information. We assume that the wireless communication channel between the transmitter node and the sink node is reciprocal, and they can directly estimate their interconnected reciprocal channels. However, due to the presence of

estimation errors, the estimated channel gains may not be identical at the transmitter node and the sink node. To achieve the reciprocal, an additional agreement protocol, such as Slepian-Wolf coding [30], is needed at the transmitter node and sink node to eliminate the difference between the estimated channel gains, which is then used for the secret key generation process. By contrast, the eavesdropper typically lies at another location, and experiences independent channel fading process. Since both the transmitter node and sink node estimate their interconnected channel without exchanging the estimated fading gains over the air, it is impossible for the eavesdropper to duplicate the transmitter-sink channel.

Information-Theoretic Security: To achieve the confidential wireless communications in physical layer, it is widely accepted that physical-layer key generation approach still relies on the secret key for data encryption, which prevents the eavesdropper's wiretapping. The information-theoretical security, operating without secret keys, can be considered as an alternative, which is a promising paradigm of protecting the wireless confidentiality against eavesdropping. It is proved from the theoretical aspect in [31] that if the legitimate channel spanning from the source to destination has a better condition than the wiretap channel from source to eavesdropper, the legitimate communication between the source and destination can reliably and securely can be conducted at a non-zero rate. Furthermore, with the introduction of the notion of secrecy capacity, it can be shown as the difference between the capacity of the legitimate channel and that of the wiretap channel. However, due to the imperfectness of communication environment, such as the multipath fading effect, the secrecy capacity of wireless communications is degraded. To mitigate this problem, significant research efforts have been devoted to improving the wireless secrecy capacity against multipath fading by employing the MIMO and cooperative relaying techniques.

Artificial-Noise Aided Security: Another effective way to enhance the wireless secrecy capacity is to use the artificial noise generation, which enables a wireless node to generate the artificial noise (a specifically-designed signal) to interfere with an eavesdropping without affecting the desired sink sensor node. By using the artificial noise, the wiretap channel from the source node to the eavesdropper is severely deteriorated by the specifically designed artificial noise, while it can keep the desired legitimate transmitter-receiver channel unchanged. This potentially distinguish the capacity of legitimate transmitter-receiver channel from that of unauthorized source-eavesdropper channel, explicitly showing an increase of the secrecy capacity by exploiting the artificial noise. However, when using the artificial noise generation to generate the noise, a certain amount of transmit power should be allocated to generate the artificial noise for the sake of confusing the eavesdropper.

Security-Oriented Beamforming: In the family of security-oriented beamforming techniques, it allows transmitter node to transmit its information signal in a predefined direction to the legitimate receiver. In this way, the signal received at the eavesdropper, typically lying in a direction different from the transmitter node, experiences destructive interference and resulting much weaker signal by eavesdropper. Hence, with the help of security-oriented beamforming techniques, the received signal strength (RSS) of receiver node would become much higher than that of the eavesdropper, leading to an improved secrecy capacity [32] [33]. Naturally, the security-oriented beamforming technique may also be combined with the technique of artificial noise to further enhance the physical layer security of wireless transmission against eavesdropping attack.

Security Diversity Approaches: Although both artificial noise aided security and the security-oriented beamforming approaches are effective in terms of enhancing the wireless secrecy capacity, they either waste power resources for generating the artificial noise or show a high

computational complexity for the beamforming design. Diversity approaches are also capable of improving the physical-layer security of wireless transmission. Traditionally, the diversity approaches have been used for improving the attainable transmission reliability, however, they can also be used to enhance the wireless security against eavesdropping attacks. One advantage of diversity aided approach, compared to artificial noise approaches, is that the diversity aided security can enhance the wireless security without consuming any additional power and without increasing the computational complexity. Due to the resource-constraints on the wireless nodes, implying the security diversity approaches are more suitable for wireless networks deployed in ITS to enhance the security. The mainstream diversity approaches include multiuser diversity [34] [35], multiple-antenna diversity [36], and cooperative diversity [37] [38] [39].

3.4 Spoofing Attacks and Solutions

Due to the broadcast nature of wireless medium, attackers can use specific techniques to gather useful identity and/or credential information during passive monitoring, such as eavesdropping, and further utilize this information to launch malicious attacks, such as spoofing attacks. Fig. 2 shows two types of spoofing which can be performed in the physical layer of wireless networks. After the spoofer successfully spoofs the receiver nodes, it can perform: 1) when the legitimate transmitter stops transmitting the signal, the spoofer can start to transmit a deceiving signal to the receiver node; 2) when the transmitter is still in the transmission phase, the spoofer can still transmit the deceiving signal with a much higher power to the receiver. Since the spoofer in this case has much powerful signal, the receiver node would potential accept the spoofing signal as legitimate signal while it rejects the legitimate signal coming from the transmitter node.

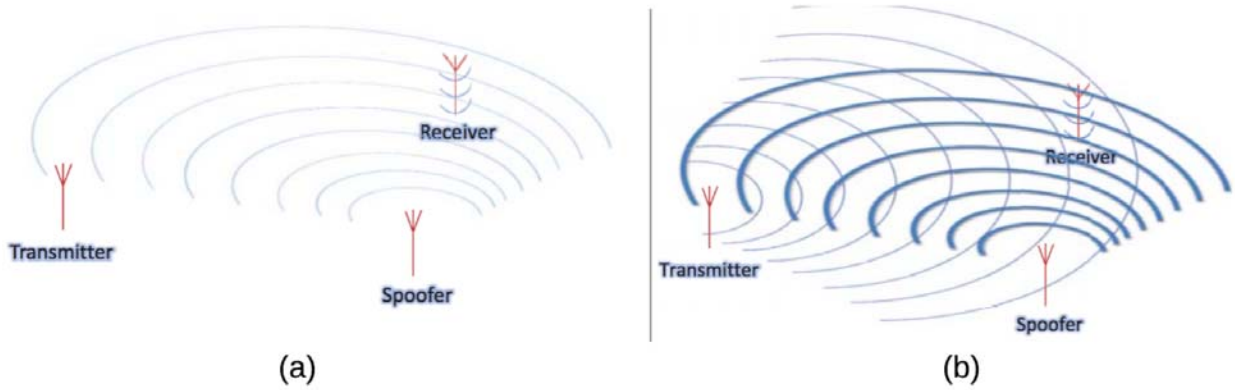


Figure 14: Two types of spoofing attacks [85].

Traditional approaches in wireless networks to deal with the spoofing attack is to apply cryptographic authentication, which only authenticate the legitimate nodes to communicate with each other. However, authentication-based approach requires additional infrastructure and computational power associated with distributing and maintaining cryptographic keys, which subsequently increase the wireless network overhead. Due to the limited resources and power available to the wireless devices and sensor nodes, it is not always possible to deploy authentication schemes in resource-constrained wireless nodes deployed in ITS.

Detecting the presence of spoofing attacks in the physical layer provides the first-order information toward defending against the potential attackers. In the following, we first provide several well-known methods to detect the spoofing attacks, and then present several ways to prevent the spoofing attack in physical layer.

Localization of Spoofing Attacks: To prevent spoofing attacks, it is necessary to estimate the location of the potential attacker. Several localization techniques have been proposed in the literature based on the physical layer characteristics, such as Received Signal Strength (RSS), Angle of Arrival (AoA), and Time Difference of Arrival (TDoA) [40].

Channel Based Prevention Methods Against Spoofing Attacks: The wireless channel has many unique properties, which can be used to provide secure communication and prevent the spoofing attacks [41] [42]. In channel-based prevention, link signatures or fingerprints can provide useful information against spoofing attacks, such as amplitude, phase, multipath delay of signals. There exist close relations among link signature, channel impulse response, the transmitter's place, which one's change would cause another parameters' change [43]. For example, if the legitimate transmitter sends a message, the receiver performs the ink estimation and assigns a link signature to the transmitter as a reference. Due to different positions, the link signature of the spoofer would differ from the legitimate transmitter's one. When the spoofer's link signature is compared with reference, the receiver would decide that the signal is not transmitted from the legitimate transmitter, thus the receiver would reject it.

After the detection of spoofing attacks in the physical layer, it is necessary to provide the potential countermeasure. Literally, the countermeasures for spoofing attacks are mainly proposed based on encryption methods [44], such as authentication. In [45], the authors use one specific example, the implantable medical device (IMD), to demonstrate how to prevent IMD from the spoofing attacks. The method also applies to devices deployed in ITS. The authors designed the new device called shield, which acts as a relay between the IMD and programmer. One of the most important duties of the shield device is to protect the tranceived data against potential attackers. The shield provides the security from two aspects: On one hand, when an IMD sends a report to the programmer about the patient, the shield jams this data to prevent it from being captured by eavesdroppers. Since the shield knows the jamming signal produced by itself, it can decode the transmitted data and forward it to the programmer. This would potentially prevent the eavesdropping attack. On the other hand, the shield device can prevent the spoofing attacks. When a spoofer sends a deceiving data to the IMD, the shield again jams this signal to preclude the IMD to be able to decode this deceiving data. By this way, the shield prevents the spoofing attacks.

Chapter 4: Data Link Layer Security Analysis for Wireless Networks in ITS

The data link layer (link layer or MAC layer for short) is the protocol layer that transfer data between adjacent network nodes, providing the functional and procedural means to transfer data between network entities and might providing the means to detect and possibly correct errors that may occur in the physical layer. In resource-constrained wireless networks, many nodes contend for a single shared medium, and thus one of the main functions of the data link layer is to regulate the access sequences and scheduling the accesses to the shared medium in such a way that all the

nodes can meet the functional and timing requirement. There exist different ways to categorize the potential attacks in the data link layers, which eventually cause the security issues [46]. They are summarized as follows.

Attacks based on Damage/Access Level: This classification of link layer attacks is based on their damage level or attacker's access level.

Passive Attacker: The passive attackers usually do not interrupt the regular communication channel on MAC layer. This type of attacks includes monitoring and eavesdropping from communication channel by unauthorized attacker [47], naturally against privacy, mimicing the normal node and gathering information from the network, etc. By conducting these attacks, the attackers intend to obtain the following purposes: eavesdropping, gathering and stealing information; compromising privacy and confidentiality requirements; degrading the network functionality; network partition by non-cooperate in operations, etc.

Active Attacker: The active attackers typically take some malicious operations on the MAC layer, such as: injecting faulty data into the network; impersonating [48]; packet modification [49]; overloading the network; creating hole in security protections; unauthorized access and modify resources and data stream, etc. By conducting these attacks, the attackers aim to achieve the following purposes, such as network functionality disruption or performance degradation, data alteration, inability in use the network's services; node destruction, etc.

Attacks based on attacker location: This classification of link layer attacks is based on the malicious node's location which can be deployed inside or outside the network. If the malicious node is deployed within the network's range, it can be called insider or internal attacker; otherwise, it can be called outsider or external attacker.

Internal Attacker: The insider typically has stronger ability to perform malicious operations which is a main challenge in resource-constrained wireless networks. These operations include access to all other nodes in the network within its range [50], authorizing the malicious/compromised nodes, executing malicious data or use of cryptography contents of the legitimate node [49], legitimating authenticated entity compromising several nodes, etc. Through launching these attacks, the attackers may want to achieve the following purposes, such as access to cryptography keys or other nodes, revealing secret keys, partial/total degradation or disruption, etc.

External Attacker: The outsider generally has the most common features on the data link layer, such as external to the network, committed by illegally parties [48], initiating attacks without even being authenticated, etc. By having these special features, the attackers want to have some effects on the link layer, such as jamming the entire communication of the network, triggering DoS attacks, and consuming resources in the network, etc.

Attacks based on function/operation: This classification of link layer attacks is based on the main functionality or operations.

Secrecy: For the function of secrecy, it usually operates stealthy on the communication channels including eavesdropping, packet replay, spoofing or modification, injecting false data into the network [50], etc. By doing so, the attackers want to achieve certain goals, such as passive eavesdrop, packet replication, spoofing or modification, etc.

Availability: For the function of availability, it is known as DoS attacks, which can lead to the network's unavailability and degrade the whole network performance, etc. By performing these DoS attacks, the attackers want to achieve certain goals, such as performance degradation, the network useless/unavailable; the network's service destruction/disruption, etc.

Based on the different types of attacks in resource-constrained wireless networks, in this section we focus on analyzing four types of attacks in the data link layer, including collision, resource exhaustion, and replay attacks. For each type of the attacks, we discuss the security challenges and the current solutions. More details of the discussion can be found in our full technical report [1].

4.1 Collision Attacks and Solutions

Due to their ease of deployment and simplicity, distributed MAC protocols, such as IEEE 802.11 Distributed Coordination Function (DCF) are widely used in computer networks to allow user to statistically share a common channel for their data transmission. However, one critical drawback of distributed MAC protocols is the inability of nodes to detect collisions while they are transmitting. Hence, bandwidth is wasted in transmitting the corrupted packets, and the throughput also degrades. As the number of nodes in the network increases, this situation exacerbated, since the rate of collisions increases as well. For collisions happened in the MAC layer, there exist two types of collisions: hidden terminal problem and malicious collision. The collision caused from hidden terminal problem is mainly due to the node deployment issues; while malicious collision is typically caused by the adversary. This report focuses on malicious collision attacks.

In a malicious collision attack, the adversary sends its own signal when it hears that a legitimate node will transmit a message in order to make interferences. Theoretically, causing collision in only one byte is enough to create a CRC error and to cripple the message [51]. To the potential adversary, the advantage to launch a collision attack compared to a jamming attack is the much smaller energy to be consumed and the difficulty to be detected, since the only evidence of collisions attacks is the incorrect message.

The malicious collision attack can be easily launched by a compromised or hostile node. The adversary does not follow the MAC protocol and causes collisions with its neighbor nodes' transmissions by sending a short noise packet. In general, this attack can cause a lot of disruptions to the network operation without consuming much energy of the attacker [52]. Meanwhile, due to the wireless broadcast nature, it is not trivial to identify and detect the attacker. For example, the adversary may only need to induce a collision in one byte of a transmission to disrupt the packet with little effort. However, due to the errors and lost packets, it incurs a high retransmission rate of the legitimate nodes and can severely slow down the victims' normal operation and eventually render it unavailable for the intended users [53] [54].

In general, for malicious collision attacks, all countermeasures that can be used against jamming attacks can be applied to the collision attacks with little modification. Another solution is to use Error Correcting Codes (ECC) which are efficient in some situations. For example, errors occur on a limited number of bytes. However, the countermeasures using ECC presents an expensive communication overhead and additional processing. In general, the ECC scheme has already been incorporated into many current Medium Access protocols.

4.2 Exhaustion Attacks and Solutions

We introduce two types of exhaustion attacks in this section: resource exhaustion attacks and node exhaustion attacks. Resource exhaustion attack is a collision attack taken a bit further. Even if the designs, implementations, and configurations are all correction, resource exhaustion is still possible. A malicious node may conduct a collision attack repeatedly in order to exhaust resource of the communicating nodes. Thus, repeated collisions can be used by an attacker to cause resource exhaustion. Also, the malicious node that can generate large amounts of traffic can flood a victim's network link [55].

4.2.1 Types of Exhaustion Attacks

On exhaustion attack is exergy exhaustion. In such attacks, one original link layer implementation might constantly attempt to re-transmit the corrupted packets or same packet. The energy resources of the transmitting node as well as those surrounding it will be quickly depleted. Exhaustion attacks consist of introducing collisions in the link layer frames and force the node to re-transmit the packets continuously until its death. Meanwhile, poorly authenticated memory allocation or code execution may also enable an attacker to consume these resources and causes DoS [55].

Another exhaustion attack is to explore the vulnerability of control packets, such as RTS/CTS in 802.11. Typically, RTS/CTS-based MAC protocols are sender invitation MAC protocols which the source node sends the invitation to the receiving node. In this type of protocols, when a sender sends out RTS control packets to initialize a transmission, the receiver has to acknowledge the sender's invitation with CTS control packet if it is available. Since the adversaries are also the normal nodes in the network, the receiver cannot distinguish whether the RTS packet was sent by the intended normal node or by an adversary. Under this condition, the adversaries can attempt to re-transmit RTS control packets to normal nodes repeatedly, enforcing the receiver to acknowledge these requests. This kind of abnormal re-transmissions could culminate in the exhaustion of battery resources of receivers, which is a kind of exhaustion attack. The exhaustion attack is particularly disastrous if the attacker intends to exhaust critical nodes. Finding the corresponding countermeasures for such an attack can be difficult since the detecting node should know, at the link layer whether the packet it received could be classified as an attack [56].

Resource exhaustion may also happen when an adversary transmits a consistent, high volume of packets from one or more attack nodes. All nodes that are within the communication range of those attack nodes are possible targets and their power supplies are subject to intentional exhaustion. Moreover, the degradation of the batteries will be accelerated if the packets emanating from the adversaries elicit a transmitted response from the target nodes. Resource exhaustion attacks executed in this manner are more severe than other DoS attacks. In this situation, more nodes become unavailable at the same time and the network may be isolated into sub-networks that cannot communicate with one another [57].

There exists another type of malicious attack that may be launched by the adversary-class from either within or outside the network, called distributed node exhaustion attacks [58]. The adversary-class is defined as a set of malicious entities, intending to inflict loss either directly, or via other entities, on the network. Its responsibility is to define and introduce the malicious nodes

into the network, with the intentional purpose of launching a distributed node exhaustion attack. Distributed node exhaustion attacks are launched from multiple end nodes of a network towards a set of victim nodes, with the intent of exhausting their limited resources; exploiting the disparity which exists between the network bandwidth and the target's limited resource availability. These attacks can be analogous to Distributed Denial of Service attacks [59] in high performance computer networks. The target nodes, typically sensors, in the network are overwhelmed with higher than normal intensities of traffic inflow. This may lead to the rapid exhaustion of their limited energy resources. This further incapacitates the victim nodes from participation in the network operations [60] [61]. In distributed node exhaustion attack, the malicious nodes can be classified into three groups: compromised nodes, malicious (injected) nodes, and laptop-class adversarial nodes. The adversary-class launches the attacks and instigates the malicious nodes into the network to generate massive attack packets from multiple ending sensors of the network, towards the victim nodes. The success of this attack is achieved by the collusion feature of such an attack, where participation of multiple malicious nodes takes place.

4.2.2 Prevention Methods for Resource Exhaustion

In general, effects of exhaustion attacks are more severe than that of DoS attack [62]. Therefore, it is very crucial and necessary to prevent these attacks. In this section, we describe several existing prevention methods against exhaustion attacks in resource-constrained wireless networks. There exist four defensive mechanisms to prevent the resource exhaustion attack [63]. One possible solution is to apply rate limits to the MAC admission control. A second method is to use the only shared token to transmit the packets. A third technique is to use time-division multiplexing where each node is allotted a time slot for transmission. This eliminates the need of arbitration for each frame and can solve the indefinite postponement problem in a back-off algorithm. However, it is still susceptible to collisions. The fourth mechanism is to use the cryptography to prevent the resource exhaustion attack in wireless networks.

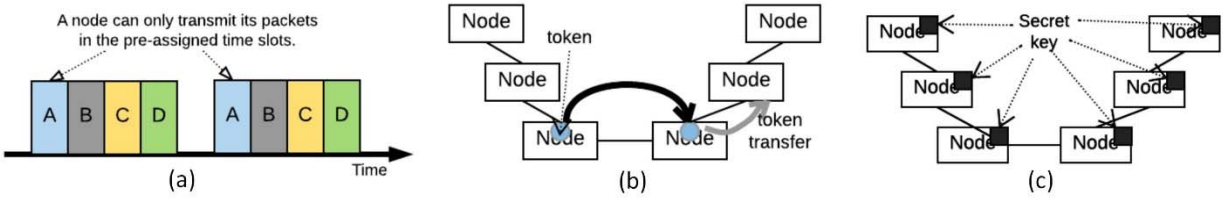


Figure 15: Resource exhaustion prevention: (a) time slot method, (b) token method and (c) secret key method.

The first prevention method uses the rate limits. The rate limits make sure that the network can ignore excessive requests, thus preventing the energy drain caused by repeated transmissions. Excessive requests will be queued or ignored without sending expensive radio transmissions. The rate must be high enough to provide enough bandwidth and timeliness for authorized users.

The second prevention method uses time slots [64]. In this method, each node is pre-assigned with a time slot and can only transmit packets during that slot. All nodes know all time slots for all other nodes. In this scheme, since the adversarial node does not belong to this network, it does not have an assigned time slot to transmit its packets. Thus, legitimate nodes can detect and discard illegitimate packets from the adversarial node when they receive the illegitimate packets. The

drawbacks of this method is that it requires all the nodes to have a synchronized clock and temporal knowledge of the pre-assigned time slots for all the other nodes that they directly communicate with. While this method precludes expenditure of power in forwarding attack frames, the immediately attacked nodes still waste energy to receive, validate, and discard illegitimate packets.

The third prevention method uses token [65]. In this method, each node may only transmit its packets when it has received a token from the network. An adversarial node cannot receive any token and therefore transmit its packets without one. Legitimate node can easily detect and discard illegitimate packets from the adversarial node when they receive the packets. The legitimate node can determine packet validity by checking the packet header. Any packet found that does not contain the token in the header information is discarded. This potentially prevent it from being forwarded to other legitimate nodes. The drawbacks of this method are that each node may only transmit its packets when it obtains the token. Besides, the node must transfer the token to the next node when it is done with transmitting its packets or after a timeout. However, handling a token in the wireless environment is difficult. Moreover, nodes which receive adversarial packets from the attack nodes consume their batteries faster than other nodes which do not receive these packets.

The fourth prevention method uses a secret key. In this method, each node in the network transmits its packets with a common assigned secret key, either from public-key encryption or from symmetric-key encryption, which is given when it joins the network. Because an adversarial node does not belong to this network, it cannot obtain the secret key, and thus transmits the packets without one. Legitimate sensor nodes that receive the attack packets directly from the adversarial node can detect and discard the illegitimate packets. Using this method, although illegitimate packets are not transmitted to other nodes from the received nodes, node which receive packets are transmitted from the legitimate nodes by checking its packets header for secret key information. Any packet found that does not contain the correct header information is discarded directly. This consumes some of their battery power by checking the headers. However, the required resources to check the packet headers are less than that of transmitting them to other nodes. The drawback of this method is that each node must obtain the secret key when it joins the network and that is must include the secret key in the header of each packet transmitted, introducing more overhead in the transmission process. Also, involving the secret key into the transmission is not secure, and is subject to the eavesdropping attacks.

4.3 Replay Attacks and Solutions

Replay attacks are a unique class of network infiltration that have harmful effects both online and offline. In general, the replay attack, often known as the man-in-the-middle attack [67], can be launched by external as well as internal nodes [68]. An external malicious node can eavesdrop on the broadcast communication between the sender node and the receiver node. It can then transmit legitimate messages at a later stage of time to gain access to the network resources. Generally, the authentication information is replayed where the attacker deceives a node to believe that the attacker is a legitimate node. The packets used for the attack came from a previous transaction between the sender node and the receiver node, in which the old packets were replayed in a separate transaction to carry out the attack. Similarly, an internal malicious node can keep a copy of all relayed data. Then, it can re-transmit this same data at a later point in time to gain the unauthorized access to the network resources. There are variations of the replay attacks, either without

modifying the data packets or after manipulating their contents (typically the header), which can be referred as a copycat attack [66]. One objective of the attacker is to make the packet look like a legitimate unit to avoid the detection at the receiver. Typically, the replay attacks try to convince two kinds of receivers: i) the MAC level recipients of a packet to accept and forward it; and ii) the final destination to believe that the received packets was a legitimately re-transmitted packet and that no attack is being launched.

There exist the following two categories of replay attacks from intelligence of the adversary:

Unintelligent Replay attack: In this kind of replay attack [69], the adversary does not have MAC protocol knowledge and no ability to penetrate the wireless network. The adversary simply records the overtapped packets, and then replays back into the network which prevent nodes from entering the sleep mode and lead to waste in energy in receiving and processing the extra packets. It is like the resource exhaustion attacks and this action of replaying events has adverse effect on the network lifetime and the overall performance of the network.

Intelligent Replay attack: Another replay attack is intelligent replay attack [69], which has full protocol knowledge but no network penetration. The adversary can use traffic analysis to determine which MAC protocol is being used in the network. With this knowledge, an attacker could expand the attack type, such as intelligent/smart jamming attack. It then injects unauthenticated unicast or broadcast traffic into the network or be more selective about replaying previous traffic.

As discussed above, the attacker in replay attack can manipulate the headers of overheard packets, by using spoofed addresses. In general, the use of digital signature [70] can somewhat help prevent the propagation of such manipulated packets. The use of Bloom filters [71] in routers can determine with probability whether a newly arrived packet is original or replayed. A detailed overview on signature procedures and algorithms can be find in [70] and a detailed overview of Bloom filters and its applications in solving various networking problems can be found in [71].

Chapter 5: Network Layer Security Analysis for Wireless Networks in ITS

One of the major tasks of the network layer is routing, delivering data from one node to another. Most of the data in the wireless networks deployed in ITS can be directed towards the sink node. However, some multi-hop routing protocols are needed between the sink and sensor nodes for these types of networks. In general, most of the data generated from a node will pass through one or multiple nodes before reaching the sink. Direct communications from individual nodes to the sink are typically energy consuming, so multi-hop communications are preferred in many of the applications in ITS.

When attacking the network layer, many attacks focus on the routing mechanisms. Due to the complex networking mechanisms, they expose many vulnerabilities. Typically, these routing mechanisms are responsible for routing information exchanged among the nodes. The attackers may spoof or alter routing data, such as repeat sending data captured earlier (known as replay attacks in the network layer) in order to disrupt the normal network data flow. Furthermore, by

performing operations on the routing data, the attacker can intentionally create routing loops, repel or attract network traffic, change original routes, intentionally increase delay or generate false alerts. Cryptography primitives are good options to prevent the potential attacks on the network layer. For instance, the message authentication code (MAC) may be used as a potential countermeasure against spoofing and unauthorized modification of routing data, including control and message data. By applying MAC code, the receiver node may verify the data integrity and sender authenticity. Meanwhile, the use of counters and timestamps may also prevent unauthorized repetition of older messages by the attackers.

In the following, we summarize several classic vulnerabilities in the network layer and present their existing countermeasures. For the details, please refer to our technical report [1].

5.1 Selective Forwarding Attacks and Solutions

Many resource-constrained wireless networks deployed in ITS employ the multi-hop communication paradigm, which assumes that all intermediate nodes will fairly and faithfully forward received messages toward their destination. This assumption is often misused by the adversary where in most cases the attacker selectively forwards received packets [73] [72]. Particularly, the malicious nodes may refuse to forward certain messages and simply drop them, ensuring that they are not propagated any further. The simplest case of a selective forwarding is the situation where a malicious node refuses to forward any packet, which all packets are dropped during the data transmission. The malicious node in this attack behaves like a black hole and refuses to forward every packet it sees. This variant of a selective forwarding attack is usually called black hole attack. In the network layer, there exist different types of holes, such as black-hole, gray-hole and sink-hole. And the holes cannot be predicted, they may even extend if sensor nodes at their boundaries are solicited with increasing data communication requests [74]. In general, there are two categories of selective forwarding attacks: passive and active. In passive selective forwarding attacks, such as black-hole and gray-hole attacks, the malicious nodes only take action on the incoming packets, instead of attracting the network packets; while in the active selective forwarding attacks, such as sink-hole attack, the malicious node tries to fake the network and attract all the traffic from the network to towards itself.

Black-hole Attack: Black-hole attack is a kind of denial of service (DoS) in nature. It can affect the performance of network, especially the end-to-end delay and throughput, if without effective control mechanisms in the network. Thus, it becomes very important to detect and prevent black-hole attack. Black-hole attacks occur when an attacker captures and reprograms a set of nodes in the network to block the packets that they receive instead of forwarding them towards the base station. As a result, any incoming data flow that enters in the black hole region or black-hole nodes is captured and terminated in the black hole region. Typically, black-hole attacks are easy to constitute, and they are capable of undermining network effectiveness by partitioning the network, such that important data/control messages do not arrive the base station successfully.

Gray-hole Attack: The gray hole attack is a refined form of black hole attack, in which malicious node drops only selected packets and forwards the others, depending on the source or the destination of the packets. Another kind of gray hole may behave maliciously for a given period by dropping all packets, and then switch to normal behavior later, which can skip the detection mechanisms designed for black hole attacks [75]. Due to this behavior, it is very difficult for the

network to figure out such kind of attack. Both types of gray hole attacks seek to disrupt the network without being detected by the security measures in place.

Sink-hole Attack: The sink-hole attack is a type of active selective forwarding attacks where a malicious node attempts to attract all traffic from the network (or at least one part of the network) [72]. The adversary's goal is to lure nearly all the traffic from an area through a compromised node, creating a metaphorical sinkhole with the adversary at the center [73]. Typically, this attack is achieved by falsifying routing data. For example, the attacker advertises quality routes to the base station, and encourages neighboring nodes to send their packets to the malicious node. Thus, the malicious node acts like a sinkhole for all traffic from its neighbor surrounding. By doing that, the nodes on, or near, the path that packets follow have many opportunities to tamper with application data, sinkhole attacks can enable many other attacks, such as passive selective forwarding attacks.

Existing solutions for selective forwarding attacks: the black-hole attack is the simplest case of selective forwarding attack. It does not need complex countermeasures to detect or prevent this kind of attacks. The gray hole attack is a kind of DoS attack in wireless networks. It is a specialized type of black hole attack which potentially changes its state from honest to malicious and vice versa. Meanwhile, due to this changeable role in the network, detection of gray hole attack is harder because the sensor nodes can drop the packets partially not only due to its malicious nature but also due to congestion [76]. In general, the gray hole attack is an event to degrade the overall network's performance by intentional malicious activity. Thus, it needs efficient schemes to detect and prevent this kind of attack. Several effective countermeasures have been proposed to against the gray-hole attack and we refer the readers to our full-version technical report for the details.

To detect sinkhole attacks, many intrusion detection systems (IDS) have been proposed to recognize abnormal route updates. In addition, sinkholes in the network typically show some characteristics. One characteristic is that the nodes around the sinkhole deplete their energy faster than other nodes since the routes passing through the sinkholes are more attractive thus are used more frequently. Thus, an energy hole forms around each sinkhole. In [77], the authors proposed two approaches to detecting and mitigating sinkhole attack based on the characteristic of energy hole. The first approach allows the base station to utilize a geostatistical method to sample the residual energy of every sensing region and estimates the possibility of existence of the sinkhole in each region using an extracted statistical estimator. The second approach uses a distributed monitoring method to detect regions with lower average residual energy level. The authors provide an analytical model to capture the interactions between various contributing parameters in the proposed detection methods. They also provided mitigation strategies to eliminate suspicious regions from routing in order to eventually quarantine the sinkholes.

5.2 Wormholes Attacks and Solutions

Wormhole attacks usually occur by connecting at least two malicious nodes via an out-of-band connection that is called tunnel. The attacker utilizes one or more devices with significantly larger resources than the average sensor node. These devices enable the attacker to perform more complex attacks. The attacker most frequently uses two devices to establish a low-latency link (tunnel) between two distant parts of the network (usually called a wormhole). The first malicious device node eavesdrops or receives packets in one area and then tunnels the packets to the next

malicious device node that is placed at another point of the network. Wormhole attacks can be achieved from different techniques, such as packet encapsulation, high-transmission power, high-quality communication links, packet relaying and protocol distortion [79] [80]. Wormhole attacks can be classified into the following categories:

Encapsulation based Wormhole Attack: In encapsulation-based wormhole attacks, it usually exists several nodes between two malicious nodes. However, the data transmissions between the malicious nodes are encapsulated. The encapsulated data packets sent between the malicious nodes does not increase the actual hop count during the reversal [80]. Thus, routing protocols that use hop count for path selection, such as ad-hoc on-demand distance vector (AODV), are particularly susceptible to encapsulation-based wormhole attacks. This type wormhole attack can prevent sensor nodes from discovering legitimate paths that are more than two hops ways. In general, the encapsulation-based wormhole attack is not difficult to launch since the two ending nodes in wormhole do not need to have any cryptographic information, or any special capabilities, such as a high-power source or a high-speed link.

High-quality/Out-of-band Channel based Wormhole Attack: Wormhole attacks can also be launched by using a high-quality, single-hop, out-of-band link, which also called the tunnel, between the malicious nodes. Typically, there are two ways to establish this tunnel, for example, by using a direct wired link or a long-range directional wireless link. However, this type of attack is more difficult to launch compared to the encapsulation-based attacks, since it needs specialized hardware capability [80].

High-power Transmission Capability based Wormhole Attack: Compared to the above two types of wormhole attacks which need at least two nodes to cooperate, only one malicious node with high-power transmission capability also can achieve wormhole attack, and this single node can act as a normal node to communicate with other normal nodes from a long distance.

Packet Relay based Wormhole Attack: Replay-based attack [81] can also be used in wormhole attacks, which can be launched by one or more malicious nodes. In this type of attacks, a malicious node replays data packet of two distant nodes to convince them that they are neighbors [80].

Protocol Distortion based Wormhole Attack: Protocol distortion-based wormhole attack, also called "rushing attack" [82], tries to attract network traffic by distorting the routing protocol. Typically, "shortest delay" based routing protocols, compared to traditional "smallest hop count" based protocol, is at the risk of wormhole attacks by using protocol distortion.

Wormhole attacks are difficult to detect as the malicious nodes replay valid date packets into the network. In general, most routing protocols employ lightweight cryptographic solutions to prevent unauthorized nodes from injecting false data packets into the network, known as intrusion detection systems. However, the replayed data packets pass all cryptographic checks in wormhole attacks, which cannot easily be detected by the intrusion detection systems. Due to the nature of easy implementation and hard detection, wormhole attacks prevention and detection schemes have been an attractive research problem. Most proposed protocols to defend against wormhole attacks use positioning devices, synchronized clocks or directional antennas. Also, there are plenty of detection mechanisms exits in literature [78].

Distance-bounding/Consistency-based Approaches: Distance-bounding techniques allow two communication nodes to estimate the actual distance between them to prevent wormhole attacks. Typically, this approach can be based on some transmission characteristics, such as message traveling time information, directional antennas, and geographical information.

Synchronized Clock-based Approaches: In synchronized clock-based approaches, they assume that all nodes are tightly synchronized, and each data packet contains the sent-out time information. The main idea is the receiver node can compare the receiving time of the received data packets with the time at which the packet is sent out. Typically, the receiver node has the knowledge of transmission distance and consumed time, it can detect if the received data packets have traveled too far. If so, such as beyond the maximum allowed travel distance, the network probably undergoes a wormhole attack.

Multi-dimensional Scaling-Visualization-based Approaches: Typically, the network with malicious nodes has different visualization from that with only normal nodes. Based on this observation, multi-dimensional scaling-visualization of worm-hole (MDS-VOW) [83] can be used to prevent wormhole attacks. In this approach, the network topology is first constructed and visualized. Wormhole attack then can be detected by visualizing the anomalies introduced by the adversary. To visualize the anomalies, each sensor node estimates the distance to its neighbors using the received signal strength.

Trust-based Approaches: Trust information among the sensor nodes can be used to detect the wormhole attacks. Sensor nodes can monitor the behavior of their neighboring nodes and rate them to build the trust information. In trust-based systems, each source node uses its trust information to compute the most trustworthy path to a specific destination by circumventing intermediary malicious nodes [84] [85].

Secure Neighbor Discovery Approaches: Securely discovering one's neighbors is an effective technique for countering wormhole attacks. In [86], the authors presented a method that can be applied for detecting each mode of the wormhole attack except the protocol deviation. The fundamental mechanism under this approach is local monitoring of sensor node, which monitors the traffic in and out of its neighboring nodes and use a data structure for the first and second hop neighbors. This protocol can isolate and remove the malicious node in case of future damage. The use of consistency tests can also be used to detect the wormholes [87].

5.3 Sybil Attacks and Solutions

In a Sybil attack, a single node illegally presents multiple identities to other nodes in the network by either forging new (false) identities or stealing legal identities. In such a situation, a node can pretend to be more than one node using the identities of other legitimate nodes. A Sybil node is a misbehaving node's additional identity. Therefore, a single entity may be selected multiple times, based on multiple identities, to participate in an operation that relies on redundancy, thereby partially controlling the output of the operation, and defeating the redundancy mechanisms. In general, Sybil attack tries to degrade the integrity of data, security and resource utilization that the distributed algorithm attempts to achieve. Sybil attack can be performed to attack the distributed storage, routing mechanism, data aggregation, voting, fair resource allocation and misbehavior detection [88] [89]. Sybil attack operations can be grouped into three methods, namely: direct and indirect communication with the legitimate nodes, fabricated and stolen identity, and simultaneous/non-simultaneous attacks [88] [90].

Direct vs. Indirect Communication: In direct communication, a Sybil communicates directly to the legitimate nodes while one of the Sybil nodes listen. Likewise, messages sent from Sybil nodes are sent from one of the malicious devices. In indirect communication, no legitimate nodes can communicate directly with the Sybil nodes. Instead, one or more of the malicious devices claim to be able to reach the Sybil nodes. Messages sent to a Sybil node are routed through one of these malicious nodes, which pretends to pass on the messages to a Sybil node.

Fabricated vs. Stolen Identities: A Sybil node can get an identity in two ways. In fabricated identities, the attacker can simply create arbitrary new Sybil identities. Given a mechanism to identify legitimate node identities, an attacker cannot fabricate new identities. In this case, the attacker needs to assign other legitimate identities to Sybil nodes. This identity theft may go undetected if the attacker destroys or temporarily disable the impersonated nodes. Meanwhile, identity replication is another related issue, in which the same identity is used many times and exists in multiple places in the network. The identity replication attack can be performed and defended against independently of the Sybil attack.

Simultaneous vs. Non-simultaneous: The attacker may try to have its Sybil identities all participate in the network at once. It usually uses the cycle method to go through its identities to make it appear that they are all present simultaneously. In non-simultaneous method, the attacker might present many identities over a period of times, while only acting as a smaller number of identities at any given time. Also, the attacker could have several physical devices in the network and have these devices swap identities. Based on the methods of Sybil attacks, there exist different types of Sybil attacks, which can be used to attack several types of protocols [88] [90].

The countermeasures to block Sybil attacks are growing as fast as the attacks are growing [91]. Security paradigm in protecting wireless networks include preparing to intercept these attacks even before they are invented [92]. For example, in [93], the author presented a method against Sybil attacks and promise an appreciable likelihood of tracking down malicious attempts with up to 99.8% reliability; others indicate 100% reliability [94]. Most of methods have proposed the use of symmetric cryptographic keys; others provide the use of asymmetric cryptographic keys [95]. Well-established detection algorithm are reviewed in [88] [90] and the readers are referred to our full technical report for the details.

5.4 Flooding Attacks and Solutions

Flooding attack is a type of Distributed DoS (DDoS) attack, which attempts to stop legitimate users from accessing a specific network resource. Typically, the DDoS attacks involve an attacker trying to do one or both of the following [97]: i) disrupt a legitimate user's connectivity by exhausting bandwidth, router processing capacity or network resources; these are essential network/transport-level flooding attacks [98]; or ii) disrupt a legitimate user's services by exhausting the server resource (e.g. sockets, CPU, memory, disk/database bandwidth, and I/O bandwidth); these essentials include application-level flooding attacks [99].

In this section, we focus more on the first category in network-level flooding attacks. The network-level flooding attacks can be further categorized into three types of attacks [97] [96]:

Flooding Attacks: This type of attacks focuses on disrupting legitimate user's connectivity by exhausting victim network's bandwidth, such as ICMP flood. For example, ICMP Ping Flood

Attack is based on sending huge number of ping packets, usually using "ping" commend from unix-like host. In this way, attacked system cannot respond to legitimate traffic.

Reflection flooding attacks: In this type of attacks, attackers usually send forged requests, such as ICMP echo request, instead of direct requests to the reflectors; hence, those reflectors send their replies to the victim and exhaust victim's resources, such as Smurf and Fraggle attacks [100] [101].

Amplification-based flooding attacks: In this type of attacks, attackers exploit services to generate large message or multiple messages for each message they receive to amplify the traffic towards the victim. Reflection and amplification techniques are usually employed in tandem as in the case of Smurf attack where the attackers send requests with spoofed source IP addresses (Reflection) to many reflectors by exploiting IP broadcast feature of the packets (Amplification) [100] [101].

Usually, by the time a flooding attack is detected, there is nothing can be done except to disconnect the victim node from the network and manually fix the problem. Since flooding attacks waste a lot of resources on the paths that lead to the target node, the goal of any countermeasures is to detect them as soon as possible and stop them as near as possible to their sources.

	Types	Techniques
Centralized	Source-based	Ingress/Egress filtering at the sources' edge routers
		D-WARD
		MUlti-Level Tree for Online Packet Statistics (MULTOPS)
		Tabulated Online Packet Statistics (TOPS)
		MANAnet's Reverse Firewall
	Destination-based	IP Trackback mechanism
		Management Information Base (MIB)
		Packet marking and filtering mechanisms
		Packet dropping based on the level of congestion
	Network-based	Route-based packet filtering
		Detecting and filtering malicious routers
Distributed	Hybrid/Distributed	Hybrid packet marking and throttling/filtering mechanisms
		DEFensive Cooperative Overlay Mesh (DEFCOM)
		COSSACK
		Capability-based mechanisms
		Active Internet Traffic Filtering (AITF)
		StopIt

Table 2: Summary of Countermeasures of flooding attacks based on their deployment locations

Several mechanisms to combat flooding attacks have been proposed to date in the literature. Please see our technical report for the details. According to the location of defense mechanism implemented, the countermeasures that defense against network/transport-level flooding attacks can be categorized into four groups: source-based, destination-based, network-based, and hybrid (or distributed). The source-based, destination-based, and network-based schemes are centralized scheme in which there is no strong cooperation among the deployment points. Detection and response are mostly done centrally either by each of the deployment points, such as source-based mechanism, or by some responsible points within the group of deployment points, such as network-based mechanisms. Hence these categories of defense mechanisms are called centralized. As opposed to centralized defense mechanisms, the hybrid scheme is distributed-based scheme, in

which defense mechanism are deployed at multiple locations and there is usually cooperation among the deployment points. Some of the categorized defense mechanisms for flooding attack are shown in Table 2.

Chapter 6: Intrusion Detection System Design for Wireless Networks

In this chapter, we are using the emerging 6TiSCH network protocol as an example to present our designs of intrusion detection systems (IDS) for wireless networks that are being deployed in various ITS. Our methodology, however, can be generalized to many different network protocols with necessary customization.

6.1 Introduction

It is used to be believed that the Internet architecture along with its protocols are impractical and neither suitable for the IEEE802.15.4e Time-Slotted Channel Hopping (TSCH) mode networks which led to the creation of new Working Groups (WGs) by the Internet Engineering Task Force (IETF), such as the IPv6 over the TSCH mode of IEEE 802.15.4e (6TiSCH) and the Routing over Low-Power and Lossy Links (RoLL). The goal behind those groups is to design and develop new standardize protocols specifically crafted for the time-sensitive industrial networks. These efforts help in shaping the newly born technology known as the Industrial Internet of Things (IIoT). The 6TiSCH WG was created to seamlessly integrate the industrial performance of the IEEE 802.15.4e TSCH link-layer standard and the ease-of-use of the IP-enabled protocols. 6TiSCH architecture reuses most of the protocols in the Internet stack to meet the high reliability, determinism, and scalability requirements of the industrial networks [102], to name few: the IPv6 in Low-Power Wireless Personal Area Networks (6LoWPAN) as an adaptation layer, the Routing Protocol for Low-Power and Lossy Networks (RPL) as a routing protocol, and the Constrained Application Protocol (CoAP) as a transfer protocol for its application layer. A logical sublayer called the 6top was introduced in the 6TiSCH to monitor and reschedule cells on the TSCH schedule and to collect statistical connectivity information and provide them to the upper layers. All the resource-constrained devices in the 6TiSCH networks are trusted to stay tightly synchronized during their life-time in the network to a time source that has been chosen according to the routing protocol. The architecture heavily depends on the RPL protocol to construct a synchronized and loop-free topology. The RPL is a proactive and source routing protocol that was proposed by the IETF RoLL Work Group to fulfill the routing requirements of the Low power and Lossy Networks (LLNs). The routing protocol constructs a logical Destination Oriented Directed Acyclic Graph (DODAG) over the physical topology according to an Objective Function (OF) that computes the best routes as a function of routing metrics and constraints.

Security is an important aspect of the industrial networks, and it was part of the design plan of the 6TiSCH architecture. As discussed in [103], 6TiSCH provides a secure mechanism to authenticate and authorize new devices before they can join the topology to limit the chances of suspicious entities to be part of the network. Also, 6TiSCH stack adapts most of the security protocols in the IPv6 stack -the Extensible Authentication Protocol (EAP)-Transport Layer Security (TLS) and the Datagram Transport Layer Security (DTLS) to name a few [28]. Moreover, the RPL protocol

provides different security modes for 6TiSCH devices to choose among. Such precautions, however, are in a great help in defending the networks from any external attacks whom have not joined the topology. Unfortunately, 6TiSCH devises are defenseless against any logical or physical tampering meaning that the aforementioned security techniques will not provide any protection against the internal attackers which already have access to all the keying materials and aim to disrupt the routing topology that could severely impact the whole 6TiSCH network. Certain features in any routing protocol that left without any proper protection are likely to be targeted by the internal attackers and RPL is quite flourish ground with some; for example, mote's Rank and loops' avoidance and detection rules [104]. Any mote in the RPL topology should maintain a Rank value which is a numerical representation of the mote's position in the RPL graph relative to the root of the DODAG and it has to be advertised to the neighbors. The Rank value monotonically decreases towards the root and increases in the opposite direction and accordingly each mote selects a preferred parent with a lower Rank than its own. In 6TiSCH terminology, a preferred parent is a time source and the Rank is a synonym term for the Join Priority (JP) that helps motes in selecting the best time source. One of the attacks using the Rank is when the attackers fake their own Ranks and send a lower value to other motes to lure them that they have a better route toward the root. Similarly, the compromised mote might lure its neighbors by faking the routing metric in the advertised control messages. Therefore, any attack against the Rank or the routing metric is an attack against the JP as well which eventually could disrupt the time source selection process in 6TiSCH. Moreover, the RPL protocol specifies certain rules regarding the Rank that each mote has to follow to ensure but not to guarantee that the routing topology will be loop-free; for instance, no mote should get greedy and increase its Rank to get more parents. An internal attacker could misuse this rule to create routing loops between a child and its parent which eventually leads to a synchronization loop in the 6TiSCH networks.

Most of the existing mitigation techniques detect some but not all the internal attacks. Some of them [105, 106, 107] rely on the cryptographic mechanisms that consume a lot of motes' processing power and storage space such as the one-way hash function. Others [108, 109, 110] propose Intrusion Detection Systems (IDSs) that are capable of detecting most of the attacks but might either require devices' to monitor their neighborhood by switching into the promiscuous mode which cannot be possible in TSCH-based networks, or they might intend to have high false positive rates. Set of efforts have proposed an enhancement to the RPL routing loop's rules or mitigation techniques to limit the impact of such attacks in the network. Protecting the RPL protocol against such vicious attacks remains a challenge.

The aim of our work to be summarized in this chapter is to develop more sophisticated intrusion detection methods to overcome the drawbacks in the existing literature and able to detect most of the internal attacks against the RPL protocol that could disrupt the stability and availability of the 6TiSCH networks. First, we introduce a centralized IDS named ARM (Authenticated Rank and routing Metric) [111], that could only detect some internal attacks, but not all of them. Therefore, we propose ARM-Pro, where we enhance and extend some of ARM's modules to cover the missing attacks. ARM-Pro consists of two parts centralized modules located in the DODAG root and distributed components installed in all RPL motes where the whole detection process is performed by the root. The downside of ARM-Pro is that it is a centralized IDS which limits its application to only small-scale networks. Thus, we present a fully distributed specification-based IDS referred to as FORCE (FOrged Rank and routing metriC dEtector). In FORCE, each mote

monitors every received ICMPv6 control message and locally decide if a neighbor's behavior is suspicious or not. Both ARM-Pro and FORCE are two complementary specification IDSs that can detect attacks designated to disrupt the 6TiSCH time source selection process and create routing loops with high accuracy rates and incurring only moderate computation and communication overheads.

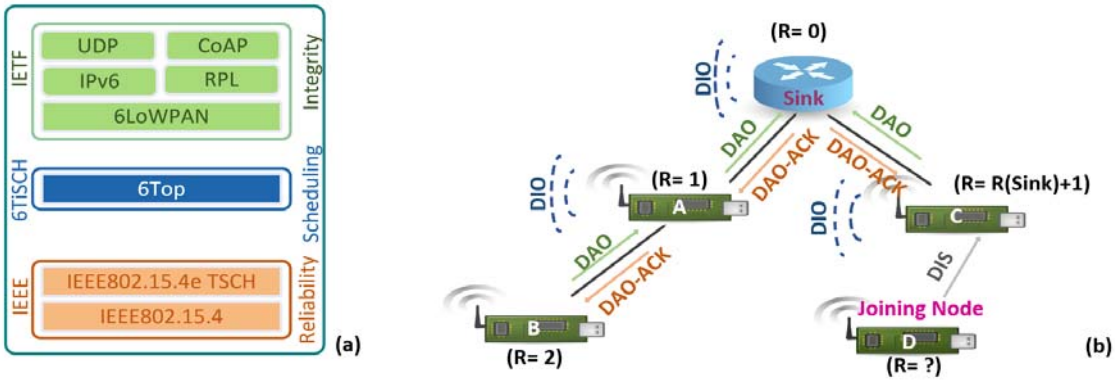


Figure 16: (a) Overview of the 6TiSCH architecture; (b) an example of an RPL DODAG graph. The black arrows indicate a parent-child relationship and the numbers in parenthesis are node's Rank value. DIO, DIS, DAO, and DAO-ACK are the four different ICMPv6 control messages used in the RPL protocol.

6.2 Background

This section includes a brief description of the 6TiSCH network stack and its routing protocol RPL. It also describes the internal attacks and reviews the existing countermeasures.

6.2.1 6TiSCH Architecture and RPL Routing Protocol

6TiSCH architecture fills the gap that is needed to connect the low-power and time-critical industrial networks to the Internet. It enables the connectivity between the IPv6 protocols over the TSCH mode of the IEEE 802.15.4e protocol. The 6TiSCH stack includes most of the existing IPv6 upper layer protocols as 6LoWPAN, RPL, and COAP, while improving others as the 6LoWPAN Neighbor Discovery (ND). As illustrated in Fig. 4(a), 6TiSCH introduces a logical operational sublayer called 6top that resides between the link and the network layers to monitor and collect statistical information and manage the TSCH schedule which is a common schedule between all the sensor motes that specifies the time slot and the channel frequency a sensor can use to communicate with its neighbors. In 6TiSCH, each mote has to select one of its neighbors to be a time source in order to be tightly synchronized to the network. 6TiSCH standard specifies that motes should select the time source that is closer to the root of the topology according to a value called the Join Priority (JP) that is being advertised in the TSCH Synchronization Information Element (IE) in the Enhanced Beacon (EB); the smaller the number the better is the time source. 6TiSCH leaves the selection process of the time source to the RPL protocol since it provides a loop-free topology and translates the selection into the JP.

RPL is a distance-vector routing protocol that generates a mesh tree-like topology named DODAG according to some routing metrics and constraints defined by the Objective Function, as shown in

Fig. 4(b). The construction of the topology starts with the root of the DODAG, that multicasts a RPL specific Internet Control Message Protocol for IPv6 (ICMPv6) named DODAG Information Object (DIO) which carries the required routing information to finish constructing the DODAG such as the root's IPv6 address, the mote's Rank, and the RPL Instance ID, etc. When a nearby mote received the DIO, it must calculate its Rank which represents the mote's location in the topology regarding the root of the DODAG where the calculation is subjective to the OF. Then, the mote selects the root as its preferred parent and multicasts its Rank in a new DIO and used the Rank value to substitute the JP in the 6TiSCH EBs. Also, it has to unicast another ICMPv6 control message called the DODAG Advertisement Object (DAO) to propagate its destination information to its preferred parent in order to support downward routing traffic and has to receive a DAO-ACK as an acknowledgment that the DAO has been received by its parent. The process of receiving DIOs, selecting a preferred parent or a neighbor with the lowest Rank among the others, then multicasting a new DIO, and sending DAO to the parent is going to be propagated through all RPL motes until the last one joins the network. As an example shown in Fig. 4(b), the Sink mote which is the DODAG root multicasts a DIO and the direct children motes A, and C calculate their own Ranks and for simplicity and clarity reasons, the Rank is calculated by increasing the parent's Rank by 1. Then, both motes multicast new DIOs, and send DAOs to the Sink. Each RPL mote has to maintain the followings: 1) neighbors set which includes all the neighbors in its transmission range; 2) a subset from the previous set called the parents set to identify the motes with better upward routes to the root and the best among them will be chosen as its preferred parent or a time source in 6TiSCH terminology. The network is stabilized when the number of DIO control messages are degraded over time which is determined according to the Trickle algorithm [112]. In case a new mote is added to the network and it does not receive any DIOs from its neighbors, then it can send a DODAG Information Solicitation (DIS) message to solicit the up-to-date DIO message and its neighbor has to reply with a unicast DIO; for instance, the new joining mote D, in the provided example in Fig.4(b), sends DIS to the nearest neighbor mote C and the latter sends back the current routing information in a unicast DIO. RPL depends on the Rank to avoid routing loops; for example, a mote is not allowed to accept any DIO messages from a neighbor with higher Rank than its own, or it should not increase its Rank and move deeper in the DODAG to have more parents. When a loop is detected, the RPL first performs a local repair by selecting another alternative route through another parent that may not be the optimal, but it will resolve the inconsistency. If it cannot be resolved through local repairs, then the root reconstructs the DODAG by initiating a global repair process.

6.2.2 RPL-Based Internal Attacks

Figure 5(a) describes a healthy RPL topology with 9 motes, where the cost of the path (P) between any mote and the DODAG root (S) is calculated by adding the link cost between the mote and its preferred parent (L) to the parent's path cost. Usually, it is referred to the link cost as the Expected Transmission Count (ETX) which is calculated according to the OF. We categories RPL-Based internal attacks according to the vulnerable component they attack as follows:

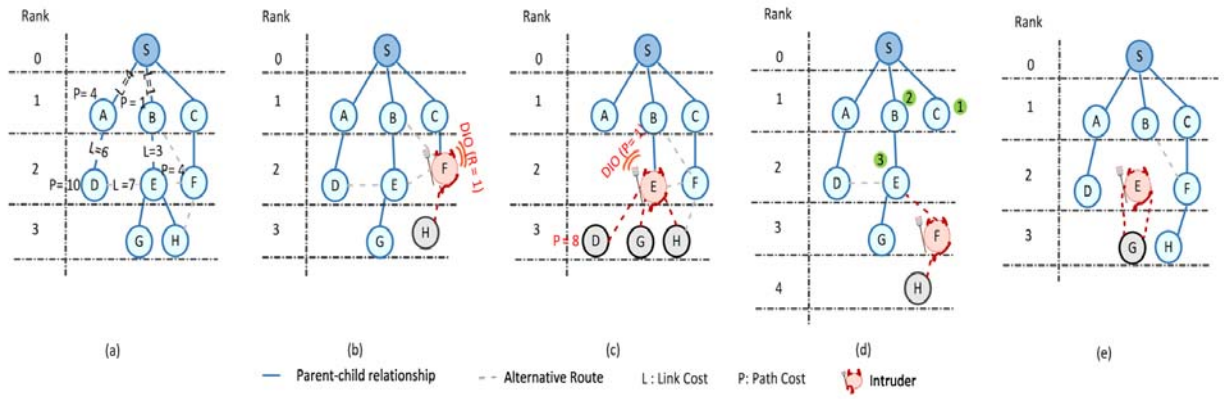


Figure 17: Different scenarios of Rank-related attacks: (a) Healthy RPL topology; (b) Malicious node F establishes DR attack by multicasting DIO with fake Rank; (c) Malicious node E initiates RAOF attack by advertising better path cost; (d) Malicious node F performs WPS attack by selecting the worst parent in its parent set that is node E; (e) Malicious node E selects one of its direct children node G to perform IR attack.

Rank value: where the mote's Rank is being victimized such as in the Decreased Rank (DR) and Increased Rank (IR) attacks. In the former attack, the compromised mote multicasts DIO message with a Rank value that is far better than the rest of the motes in its neighborhood; for instance mote F in Fig. 5(b) performs the attack by advertising falsified Rank value that is not its own in order to create un-optimized paths and disrupt the device-to-device synchronization in the time-critical networks. Where in the latter, the compromised mote does not follow the RPL loop avoidance and detection rules, by selecting one of its children as its preferred parent to create a routing loop. Hence, both motes the attacker and its child may get into the count-to-infinity situation, where they keep incrementing their Rank values until they reach the maximum or the Infinity Rank value and then they detach from the DODAG, as illustrated in Fig. 5(e), the attacker (mote E) and its child (mote G) create a routing loop and they detach from the rest of DODAG when they reach the Infinity Rank value. This attack creates more loops and congest the RPL topology with more control messages which further exhaust the devices' resources and shorten their life. Due to this attack, part of the DODAG is enforced to be isolated and desynchronized from the topology.

Routing metric: where the compromised mote forges its own routing metric and multicasts a better routing path cost toward the DODAG's root that is the minimum among all its neighbors to increase its chance of being selected as a preferred parent, such attack is known as the Rank Attack using Objective Function (RAOF). For example in Fig. 5(c), mote E multicasts a DIO with a fake path cost that is not the real one (P=1), and it has been selected by mote D as its preferred parent since its current path cost toward the root through the parent mote A (P=10) is worse than the new path cost through mote E (P=8). This attack might incur up to 50% additional delay in data forwarding depending on the number of RAOF attackers and their locations in the DODAG.

Parent selection: where the attacker selects the least preferable parent, that is the last one in its parent set, as its preferred parent to create suboptimal routes and to prevent the routing OF from being fully achieved, in the literature this attack is known as the Worst Parent Selection (WPS) attack. As described in Fig. 5(d), the compromised mote F selects the worst parent (mote E) out of the three motes in its parent set which are numerically ordered to reflect their preferability. This attack creates instability in the RPL topology motes change their parents more frequently than usual, which creates instability in the topology; thus, leading to an increase in the control message

overhead and a decrease in the packet delivery ratio up to 60% depending on the number of attackers [113].

6.3 Proposed Intrusion Detection Systems

Our main motivation is to overcome the flaws in the existing detection methodologies and to detect the RPL-based internal attacks. We first propose ARM that is a centralized specification-based IDS which composes of two major modules: centralized ones located in the DODAG's root, and distributed one resided in every RPL mote. Unfortunately, ARM was not fully capable of detecting all the previous discussed RPL-based attacks. Hence, we decide to enhance ARM's modules and hereafter referred to as ARM-Pro. Also, we present FORCE which is another complementary specification-based IDS. FORCE is a fully distributed IDS where it enables each mote to locally analyze any received ICMPv6 control messages.

In the following subsections, we will briefly describe ARM where we cover most of its details in ARM-Pro since it is the enhanced version of it, and later we present the distributed IDS, FORCE.

6.3.1 ARM Intrusion Detection System

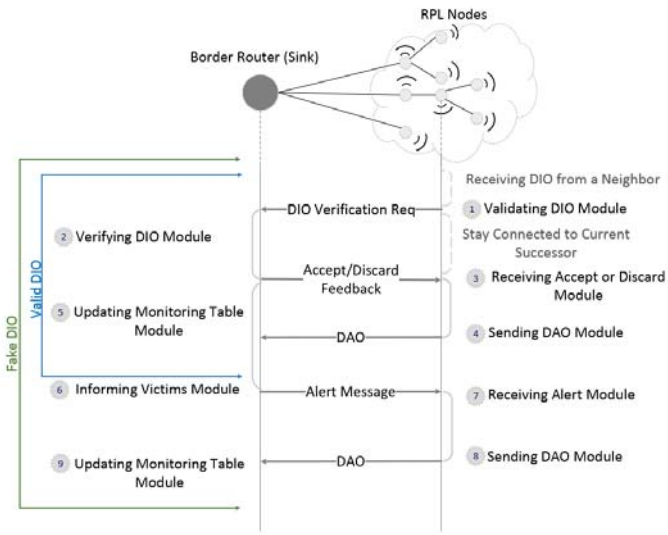


Figure 18: The interaction between the centralized and distributed modules of ARM when the received DIO message is valid and fake.

In ARM, each mote sends a DAO message to the root to construct a monitoring table that contains information about the motes in the topology such as their IDs, parents' IDs, and the link cost between them and the preferred parent. It helps the root in validating the DIOs' verification requests which have been sent by some motes in the topology and making decisions accordingly. A black list could be found in the root as well in all RPL motes to identify suspicious motes. A DIO table is maintained by the motes where its purpose is to keep a copy of the DIOs that need to be verified by the root for a short period of time until the mote receives a feedback from the root and then clear the entry from the table.

ARM has seven modules in total where three out of them are located on the root and known as the centralized modules, while the rest are installed on each RPL mote and called the distributed ones. Briefly, the root analyzes the received DIO's verification request from a mote and makes decisions, and the motes must inform the root with any changes occur to their sub-DAG, verifies new coming DIOs, and acts according to the received feedback from the root. Fig. 6 describes the interaction between the centralized and distributed modules when a mote receives a legitimate or fake DIO message. The finite state machines (FSMs) of both modules are illustrated in Fig. 7.

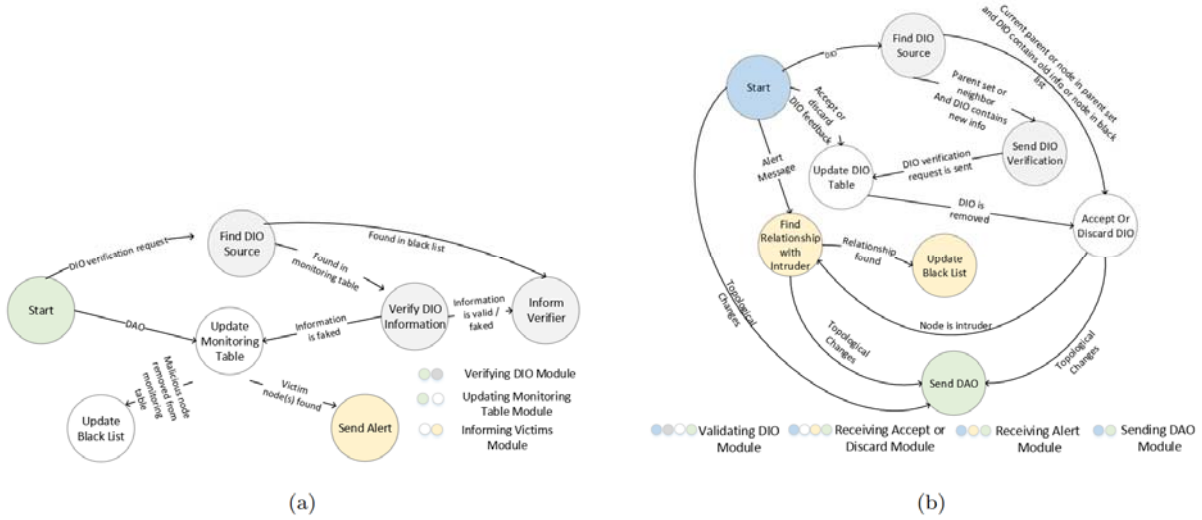


Figure 19: (a) The finite state machine of the centralized modules and (b) the distributed ones in ARM.

6.3.2 Performance Evaluation on ARM

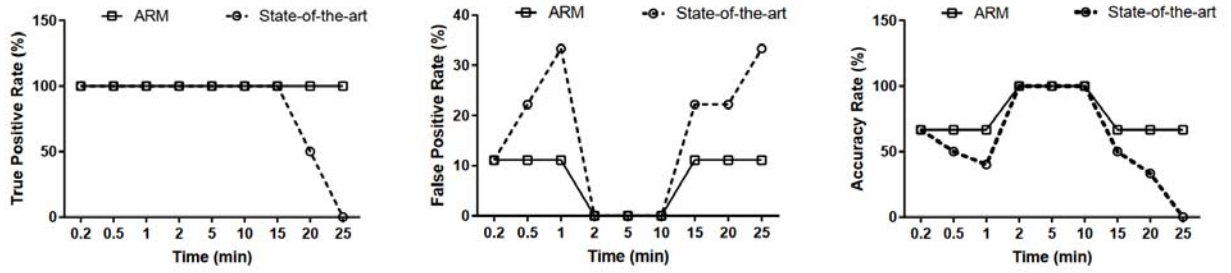


Figure 20: Comparison of detection efficiency between ARM and the state-of-the-art approach.

The Contiki network emulator Cooja [114] has been used to simulate ARM and to compare it with the stand-alone RPL protocol where the topology is not accompanying by any IDS and the state-of-the-art IDS [34] on the basis of detection efficiency of RA, RAOF, and the join occurrence of both attacks and resources overhead. The two IDSs have been tested on three different network topologies: 11, 16, and 32 motes, however the results of the average runs of the 32-mote topology are being represented in the following subsections.

Detection Efficiency: when the topology is stabilized, both IDSs successfully detected the intruder with 100% True Positive Rate (TPR) and 100% Accuracy Rate (AR). The TPR shows how good the system is in detecting the attacks and the AR is the total number of successfully raised alerts

divided by the actual total number of triggered alarms by the IDS. ARM was able to maintain 100% TPR during the time period when some of the motes have been moving up and down in the topology and has a lower False Positive Rate (FPR) equals to 10% comparing to the 40% of the state-of-the-art as illustrated in Fig. 8. The FPR measures how bad the IDS is in reporting normal healthy motes as suspicious ones. However, ARM has approximately 60% AR when the network was not stable, while the AR of the state-of-the-art IDS degraded from 40% to 0%.

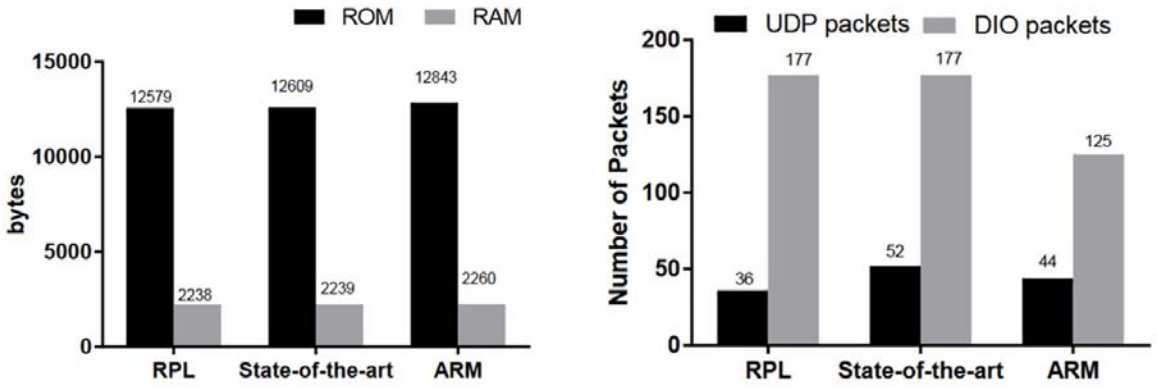
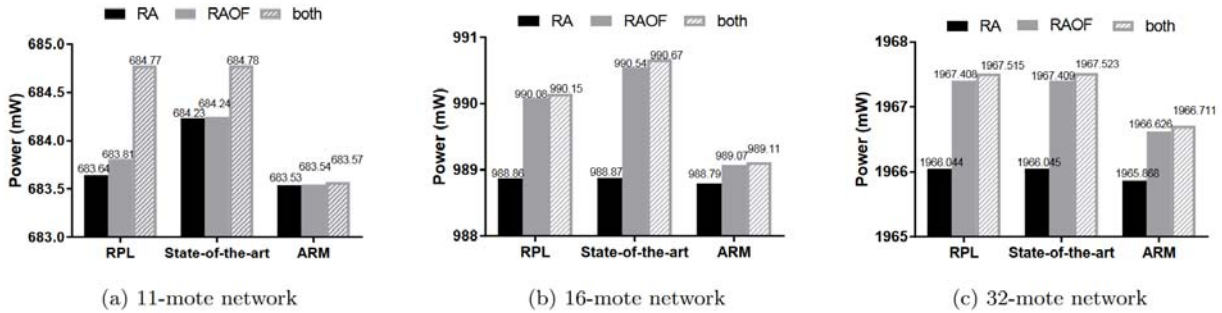


Figure 21: (a)Comparison of ROM and RAM usage. (b) Total number of DIO messages and UDP packets transmitted by RPL, the state-of-the-art IDS, and ARM.



(a) 11-mote network (b) 16-mote network (c) 32-mote network

Figure 22: The average network power consumption under RPL, the state-of-the-art IDS, and ARM under the three attacks, namely: the DR, RAOF, and the occurrence of both attacks together.

Memory Overhead: as illustrated in Fig. 9(a), ARM is a lightweight IDS where it required almost 12.843 KB ROM and 2.26 KB of RAM per mote and that is well below the total available ROM (48 KB) and RAM (10 KB) in the Tmote Sky resource-constrained devices.

Communication Overhead: Compared to the stand-alone RPL and the state-of-the-art IDS [34], ARM has the lowest number of exchanged DIOs (see Fig. 9(b)), because in the stand-alone RPL and the state-of-the-art IDS [115], motes accept DIOs and multicast their own ones, while in ARM they might multicast their own DIOs only after the received ones have been verified by the root. Although, ARM has increased the number of UDP messages since the motes have to send their routing information and DIOs verification requests to the root and the latter has to reply to these requests, the state-of-the-art IDS [115] incurred more communication overhead comparing to ARM because the monitoring motes had to transmit more alert messages to inform all neighbors.

Energy Overhead: ARM maintained the lowest power consumption comparing to the stand-alone RPL and the state-of-the-art IDS, as shown in Fig. 10. The overhead of ARM slightly increased with the number of motes in the topology, but it considered to be negligible when the network size is small (up to 11 nodes).

6.3.3 ARM-Pro Intrusion Detection Systems

ARM was designed to detect only two forms of RPL-based internal attacks, namely DR and RAOF attacks, and will not be able to detect the rest of the attacks for two main reasons: 1) the motes in ARM follow the RPL's specification in discarding DIOs with higher Ranks or with less efficient routing metric; thus ARM cannot detect the IR attack; 2) the IDS lacks the ability in knowing if a mote selected the least preferable parent meaning it cannot detect the existence of the WPS attack in the topology. Hence, ARM-Pro is introduced and supported by an enhanced distributed module to help in detecting the IR attack as shown in Fig. 11. Each mote closely monitors its neighborhood and once a neighbor increases its Rank more than a predefined threshold, the motes sends a suspicious message toward the root with the neighbor's ID. Also, the new centralized modules have been enhanced to prevent any mote from selecting the least preferable parent and to perform cross-checking over the reported suspicious IDs to find the suspicious mote that is the common one among all the reported ones.

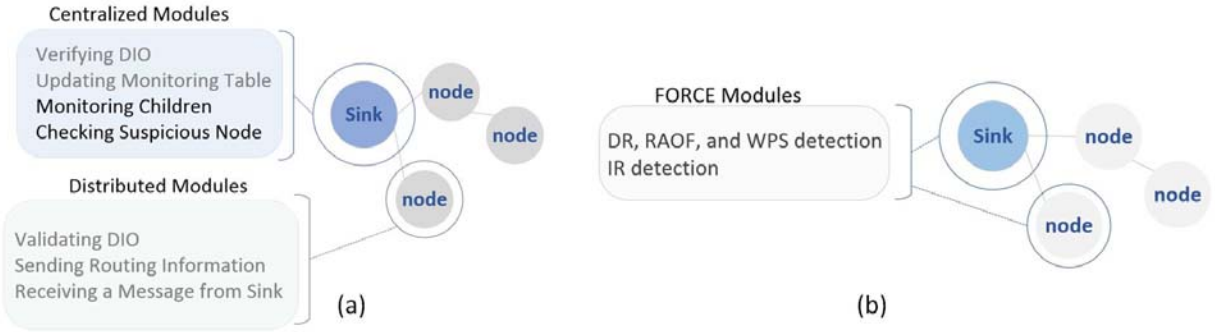


Figure 23: (a) Overview of ARM and ARM-Pro architectures. The old modules in ARM are in gray and the new added modules in ARM-Pro are in black. (b) The main modules in FORCE IDS.

As in ARM, the DODAG root in ARM-Pro has a monitoring table which maintains all the routing information of each mote in the DODAG such as motes' IDs, motes' Ranks, their parent's ID, parent's Rank and path cost, the link cost between them and their preferred parent, and the Rank value of each neighbor and its path cost. The monitoring table includes a "suspicious" numerical counter that is incremented by one every time a mote increases its Rank. The root also has a list of reported motes whom their neighbors have already sent them to the root due to their malicious behavior. A black list is installed in the root to identify already detected suspicious motes by the IDS. All RPL motes have a black list and a DIO table to keep tracking of all the new DIOs that have been received but needed to be verified by the root and once a feedback is received regarding those DIOs, their entries will be removed accordingly.

6.3.3.1 Centralized Modules in ARM-Pro Data Structures

Updating Monitoring Table: the DODAG root updates the table under two circumstances: 1) recipient of a UDP message sent from a mote to inform the root of its current routing information; 2) discovering a suspicious mote that must be moved to the black list. Before updating any mote's entry in the monitoring table, the root has to prioritize the mote's parent set according to the OF and compares the mote's preferred parent in the UDP message to the least one in the set, if it is the case then it considers this mote as a suspicious mote whom is trying to initiate the WPS attack. The root updates both the monitoring table and black list, then sends an alert message.

Monitoring Children: normally, the root will discard any DIOs sent to it according to the RPL specification, but its highly applicable its direct children might be compromised to perform any of the attacks. In ARM-Pro, hence the root will check every received DIO. If the child advertises a higher or lower Rank value than the current one, or if the DIO has better path cost, then the suspicious counter entry of that child mote in the monitoring table will be incremented by one. The child is suspicious whenever the counter exceeds the predefined threshold.

Verifying DIO: the root uses both the monitoring table and its black list to find the source of the DIO that is needed to be verified according to the mote who sent the verification request. The root might reply with one of the two messages: 1) DISCARD DIO message and that if the mote is in the black list, or when it is initiating DR, RAOF or both attacks; 2) an ACCEPT DIO message when the mote's behavior is totally normal.

Checking Suspicious Node/Mote: when most motes reporting that one of their neighbors is behaving maliciously, then the root considers that mote as a suspicious one, updates the monitoring table and the black list, and sends an alert message. If there is no common mote between the reported ones, then there is a high chance that one of the reported motes is a parent to the other ones and by increasing its Rank, it enforces its direct children to do the same. With the help of the monitoring table, the root finds the descendant mote among the reported ones and considers it as a suspicious mote.

6.3.3.2 Distributed Modules in ARM-Pro Data Structures

Sending Routing Information: upon any destination routing information changes occur to the mote; for instance, a new preferred parent has been selected by the mote, and in order to reflect that in the root's monitoring table, the mote has to send a UDP message with the new routing information.

Validating DIO: upon the reception of a DIO message and if the sender is a new neighbor, or an existing one with a better Rank or routing metric than it is used to have, then the mote sends a DIO verification request to the root, inserts the DIO into the DIO table and resumes its normal operation until it receives a feedback from the root and reacts accordingly. If the sender however has higher Rank, then the mote counts the number of times that sender increases its Rank and discards the DIO. Once the counter reaches the predefined threshold, the mote sends a UDP message to the root with the suspicious mote ID. Moreover, if it happens that the preferred parent is the sender, then the child mote detaches and places the parent into its local black list and thus ARM-Pro eliminates the chances of having the RPL motes to get into the count-to-infinity situation.

Receiving a Message from Sink: a mote might receive one of the following messages from the root: ACCEPT, DISCARD, or alert. If its ACCEPT or DISCARD, then the mote retrieves the corresponding DIO's entry from the DIO table and process it according to the RPL protocol rules.

Upon the reception of alert, the mote finds the relationship between itself and the suspicious mote. If it's a parent-child relationship, then the mote performs a local repair and selects a new parent and sends a UDP message to the root and places it into the black list. If the suspicious mote could be one of the parents in the mote's parent set or a direct child, then it has to be removed and placed into the black list.

Although ARM-Pro is a lightweight and an efficient IDS, it might add extra communication overhead since it is a centralized IDS, it is expected that the traffic load between each mote in the topology and the root might increase. Therefore, we propose a fully distributed technique, where each mote in the DODAG locally performs the detection.

6.3.4 FORCE Intrusion Detection System

FORCE [118] is a fully distributed specification-based IDS where it takes the constructed parent-child relationship by the RPL protocol into consideration; meaning a parent mote can easily detect DR, RAOF, and WPS attacks, while the IR attack can be detected by a direct child of the suspicious parent. In FORCE, each mote is a monitoring mote that should monitor its neighborhood and analyze each received DIO and DAO messages. Basically, FORCE specifies a set of rules that a mote has to follow in order to detect the RPL-based attacks, and if one of its neighbors violates at least one rule, then the monitoring node considers that as a suspicious behavior and sends an alert message to the nearby motes. Eventually, the alert message will be propagated into the whole topology and the suspicious mote will be recognizable by all motes. We describe the essential elements of FORCE below.

FORCE introduces the parents list that accommodates the Rank value of the least preferable parent of each direct child and the total number of candidate parents the child might have in order to help in detecting the WPS attack. It is expected that list will be updated whenever changes occur to the direct child's routing topology where it must send UDP messages to its parent. As in ARM-Pro, each mote keeps a local black list to be updated upon the discovery of attacks or receiving an alert message from a neighbor.

DR, RAOF, and WPS Detection Module: The monitoring mote inspects every received DIO and DAO and considers the sender as a suspicious mote whom is initiating either the DR or RAOF attack if, and only if, two certain conditions are met: 1) the monitoring mote is the sender's preferred parent, and 2) the sender has a better Rank or routing metric. If the sender is being announced as a suspecios mote, then it will be placed into the monitoring mote's black list. The latter will also send an alert message to notify its neighbors to do the same. To detect the WPS attack, each mote in FORCE shares with each parent belongs into its parent set the list of candidate parents along with their Ranks and path costs in a UDP message. Upon the reception of the message, the parent or the monitoring mote finds the least preferable parent and updates the child's entry in the data structure (parents list). Whenever a child mote sends a DIO, the parent calculates the Rank value of that child through its worst parent's Rank and compares it with the one in the DIO. If no match could be found, then the child is behaving normally; otherwise the parent announces the direct child as a suspicious mote.

IR Detection Module: Whenever a monitoring mote receives a DIO or DAO from its current preferred parent that has a higher Rank and has not sent a poison message earlier as specified by

the RPL (to notify the direct children it is no longer capable of serving them as a parent), the monitoring mote declares the parent as a suspicious mote.

6.3.5 ARM-Pro and FORCE Performance Evaluation

We evaluate the effectiveness of ARM-Pro and FORCE in detecting all the RPL-based attacks over 50-motes simulated topology using Contiki's simulation Cooja and compare the results to the ones of the state-of-the-art IDS SVELTE [116] accompany with the extended modules in [117]. The evaluation is based on the detection rate, speed and overheads incurred on the motes' resources. In the following subsections we discuss the average of multiple simulation's runs.

IDS/Attack	DR	RAOF	IR	WPS	ON/OFF
ARM-Pro	0%	0%	0%	0%	0%
FORCE	0%	0%	0%	2%	4%
SVELTE	0%	2%	6%	2%	6%

Figure 24:Average False Positive Rates

Detection Rate: The three IDSs successfully identify the four attacks namely: DR, RAOF, IR, and WPS with 100% AR. Unfortunately, FORCE has declared some healthy motes whom with a direct relationship with the attackers especially when it detects the WPS attack as shown in Figure 9. This is because a child of a suspicious parent might decrease or increase its Rank due to the changes in the ETX between the two motes and that is totally normal in the RPL topology. Therefore, a monitoring neighbor may falsely consider this behavior as a suspicious one, and thus the FPR is 2%. The percentage increases in the scenario where a normal mote leaves the topology for about couple of minutes and then re-joins as illustrated in the ON/OFF field in Fig.12, since the neighboring motes of this mote have not yet updated their local parent lists. However, the percentage is negligible in comparing to SVELTE where removing or excluding the attackers from the RPL topology is not part of the IDS design and thus the damage left by those attacks remain persistent in the DODAG.

Detection Speed: The detection speed is defined as the time instant when the attack took place in the topology until the IDS sends an alert. Figure 13(a) describes the average detection speed of the three IDSs in seconds. FORCE is the fastest where it took between 0 to 6 seconds to identify the attackers since it relies on the attackers' neighbors to perform the detection. ARM-Pro is considerable to be fast comparing to SVELTE where it took up to 59 seconds because the detection performed by the root only while the rest of the motes must verify the new DIOs with the former.

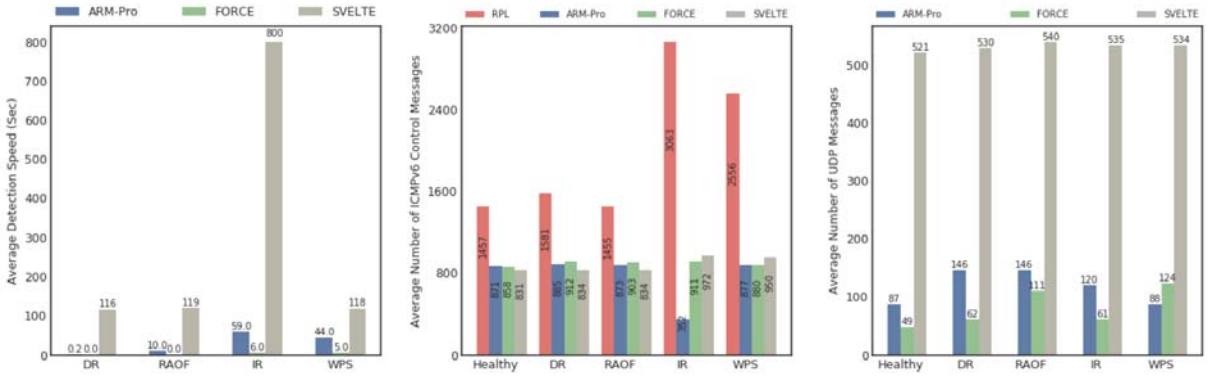


Figure 25: (a) The average detection speed of ARM-Pro and FORCE comparing to SVELTE. (b) The average number of ICMPv6 control messages by RPL, ARM-Pro, FORCE, and SVELT. (c) The average number of UDP messages by ARM-Pro, FORCE, and SVELTE.

Memory Usage: ARM-Pro, FORCE, and SVELTE have consumed around 32, 30, 27 Kbyte of mote's ROM respectively that is below the actual size of ROM (48 Kbyte) in a resource-constrained device such as the simulated Sky mote. ARM-Pro, and SVELTE consumed approximately 4 Kbyte of RAM, while FORCE required around 2 Kbyte; in any how they all used less than the 10 Kbyte of the available RAM space. The storage consumption by the proposed IDSs are needed in order to maintain their data structures and to enable them to work effectively without adding any extra overhead to the motes' memory nor to their functionality.

Communication Overhead: When the 50-motes topology is healthy, the three IDSs have almost the same amount of advertised control messages, but far less comparing to the numbers generated in the RPL topology without any IDS as illustrated in Fig. 13(b). It is noticeable that the numbers are increasing, but they are considerable to be less than the ones in the stand-alone RPL. Both ARM-Pro and FORCE have reduced to some certain degree the number of DIOs in the DODAG. In the two IDSs, the motes will not advertise a DIO if the validation process failed. On the other hand, the root in SVELTE constructs the topology and maintains its stability. SVELTE requires the transmission of additional UDP messages comparing to the proposed IDSs since the root requests all motes to update their local routing information more frequently. While in the other two IDSs, each mote shares such information either with the root or with its neighbors only when there is a routing change in its sub-DAG. It is expected for these number to increase when an attack is presented in the topology since in FORCE and ARM-Pro, each mote has to perform a local repair and remove the suspicious mote from its sub-DAG which requires the mote to send its parent list to each possible parent or its routing information to the root respectively. It should be noted that the number of alert messages is included as well, however such messages have not been part of SVELTE.

Energy Consumption: Although the IDSs do not require motes to perform heavily computational procedures, they still drain some energy from the motes. None of the three IDSs drain any extra power from the mote when the topology is healthy comparing to the same topology accompany with no IDS. However, this is not the case when the network is under one of the attacks as described in Fig. 14. The most majority of the power drainage is drawn from the CPU powers, transmission, and reception in FORCE and ARM-Pro because its expected from the motes to send alert messages and to perform a local repair to remove the suspicious mote from its sub-DAG, which increases the number of transmitted and received control messages. These numbers are negligible comparing

to the ones in the stand-alone RPL network when it is under the attack and when it is supported with SVELTE.

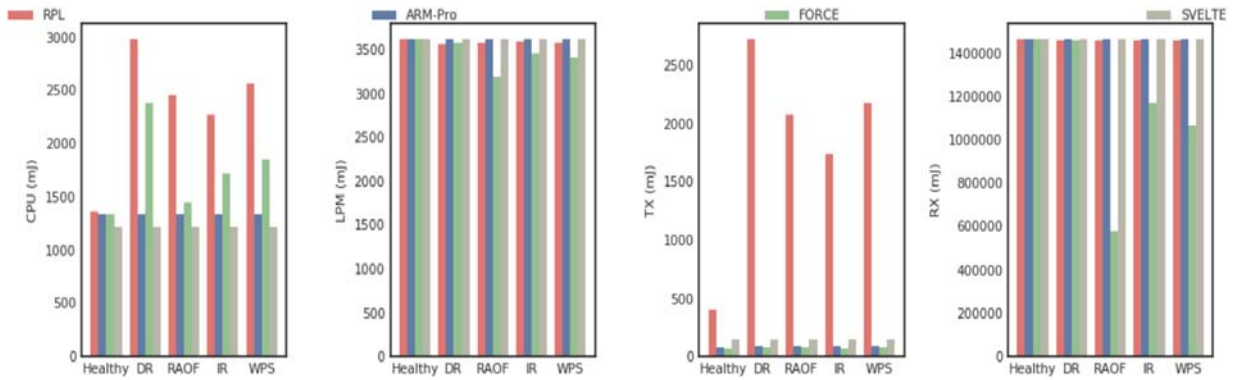


Figure 26: The average power consumption of CPU, LPM, TX, and RX per node when the topology is healthy and under different attacks.

Chapter 7: Conclusions and Recommendations

This project focuses on the cybersecurity analysis for wireless networks that are deployed in representative ITS for various sensing and control applications. We aim to gain deep understandings on the challenges and design goals of the wireless networks deployed in ITS and the cybersecurity requirements associated with those wireless solutions. We perform a comprehensive security analysis on those wireless networks in a layer-by-layer fashion to study i) the vulnerabilities and potential attacks that may happen in each layer of the protocol stacks, ii) the advantages and disadvantages of existing countermeasures, and iii) potential solutions to further improve the cybersecurity protection. To counter the attacks on the network topology at the network layer, we also develop a suite of Intrusion Detection Systems (IDS) to detect attacks against the routing protocol that could disrupt the stability and availability of the wireless networks. These IDS include ARM (Authenticated Rank and Routing Metric), ARM-Pro and FORCE (Forged Rank and Routing Metric Detector). These proposed IDS are implemented using simulation tools and on the real-life testbed for design validation and performance evaluation.

References

[1] Song Han, Gang Wang, Zelin Yun, Jiachen Wang, Areej Althubaity, Peng Wu, “Security Issues in Industrial Wireless Networks: A Comprehensive Review”, Technical Report, Computer Science and Engineering Department, University of Connecticut, 2022.

[2] Dimitrakopoulos, George, and Panagiotis Demestichas. "Intelligent transportation systems." IEEE Vehicular Technology Magazine 5, no. 1 (2010): 77-84.

[3] Tubaishat, Malik, Peng Zhuang, Qi Qi, and Yi Shang. "Wireless sensor networks in intelligent transportation systems." Wireless communications and mobile computing 9, no. 3 (2009): 287-302.

[4] Wang, Yong, Garhan Attebury, and Byrav Ramamurthy. "A survey of security issues in wireless sensor networks." (2006).

[5] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” IEEE Communications magazine, vol. 40, no. 8, pp. 102–114, 2002.

[6] A. Mukherjee, S. A. A. Fakoorian, J. Huang, and A. L. Swindlehurst, “Principles of physical layer security in multiuser wireless networks: A survey,” IEEE Communications Surveys & Tutorials, vol. 16, no. 3, pp. 1550–1573, 2014.

[7] M. H. Yılmaz and H. Arslan, “A survey: Spoofing attacks in physical layer security,” in Local Computer Networks Conference Workshops (LCN Workshops), 2015 IEEE 40th. IEEE, 2015, pp. 812–817.

[8] K. Grover, A. Lim, and Q. Yang, “Jamming and anti–jamming techniques in wireless networks: a survey,” International Journal of Ad Hoc and Ubiquitous Computing, vol. 17, no. 4, pp. 197–215, 2014.

[9] K. Pelechrinis, M. Iliofotou, and S. V. Krishnamurthy, “Denial of service attacks in wireless networks: The case of jammers,” IEEE Communications Surveys & Tutorials, vol. 13, no. 2, pp. 245–257, 2011.

[10] W. Xu, W. Trappe, Y. Zhang, and T. Wood, “The feasibility of launching and detecting jamming attacks in wireless networks,” in Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing. ACM, 2005, pp. 46–57.

[11] O. Besson, P. Stoica, and Y. Kamiya, “Direction finding in the presence of an intermittent interference,” IEEE transactions on signal processing, vol. 50, no. 7, pp. 1554–1564, 2002.

[12] Y. Liu and P. Ning, “Bittrickle: Defending against broadband and high-power reactive jamming attacks,” in IEEE INFOCOM, 2012, pp. 909–917.

[13] J. Jeung, S. Jeong, and J. Lim, “Adaptive rapid channel-hopping scheme mitigating smart jammer attacks in secure wlan,” in IEEE MILCOM, 2011, pp. 1231–1236.

[14] X. Liu, G. Noubir, R. Sundaram, and S. Tan, “spread: Foiling smart jammers using multi-layer agility,” in 26th IEEE International Conference on Computer Communications. 2007, pp. 2536–2540.

[15] W. Xu, W. Trappe, Y. Zhang, and T. Wood, “The feasibility of launching and detecting jamming attacks in wireless networks,” in Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing. ACM, 2005, pp. 46–57.

[16] D. Torrieri, “Frequency hopping with multiple frequency-shift keying and hard decisions,” IEEE transactions on communications, vol. 32, no. 5, pp. 574–582, 1984.

[17] V. Navda, A. Bohra, S. Ganguly, and D. Rubenstein, “Using channel hopping to increase 802.11 resilience to jamming attacks,” in INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE. IEEE, 2007, pp. 2526–2530.

[18] R. Gummadi, D. Wetherall, B. Greenstein, and S. Seshan, “Understanding and mitigating the impact of rf interference on 802.11 networks,” ACM SIGCOMM Computer Communication Review, vol. 37, no. 4, pp. 385–396, 2007.

[19] X. Liu, G. Noubir, R. Sundaram, and S. Tan, “spread: Foiling smart jammers using multi-layer agility,” in INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE. IEEE, 2007, pp. 2536–2540.

[20] W. Znaidi, M. Minier, and J.-P. Babau, “An ontology for attacks in wireless sensor networks,” Ph.D. dissertation, INRIA, 2008.

[21] D. W. Carman, P. S. Kruus, and B. J. Matt, “Constraints and approaches for distributed sensor network security (final),” DARPA Project report,(Cryptographic Technologies Group, Trusted Information System, NAI Labs), vol. 1, no. 1, 2000.

[22] S. Jokhio, I. A. Jokhio, and A. H. Kemp, “Node capture attack detection and defence in wireless sensor networks,” IET wireless sensor systems, vol. 2, no. 3, pp. 161–169, 2012.

[23] H. Nunoo-Mensah, K. O. Boateng, and J. D. Gadze, “Tamper-aware authentication framework for wireless sensor networks,” IET Wireless Sensor Systems, vol. 7, no. 3, pp. 73–81, 2017.

[24] H. Jin, “Higher dependability and security for mobile applications,” Lecture notes in computer science, vol. 3934, p. 89, 2006.

[25] T. Park and K. G. Shin, “Soft tamper-proofing via program integrity verification in wireless sensor networks,” IEEE Transactions on mobile computing, vol. 4, no. 3, pp. 297–309, 2005.

[26] Muhammad Naveed Aman, “Physical unclonable functions for iot security,” https://www.ece.nus.edu.sg/stfpage/bsikdar/papers/asiaccs 16.pdf.

[27] Wikipedia, “Physical unclonable functions for iot security,” https://en.wikipedia.org/wiki/Physical unclonable function.

[28] A. Kanuparthi, R. Karri, and S. Addepalli, “Hardware and embedded security in the context of internet of things,” in Proceedings of the 2013 ACM workshop on Security, privacy & dependability for cyber vehicles. ACM, 2013, pp. 61–64.

[29] Q. Wang, K. Xu, and K. Ren, “Cooperative secret key generation from phase estimation in narrowband fading channels,” IEEE Journal on selected areas in communications, vol. 30, no. 9, pp. 1666–1674, 2012.

[30] E. Abbe, “Randomness and dependencies extraction via polarization, with applications to slepian–wolf coding and secrecy,” IEEE Transactions on Information Theory, vol. 61, no. 5, pp. 2388–2398, 2015.

[31] A. D. Wyner, “The wire-tap channel,” Bell Labs Technical Journal, vol. 54, no. 8, pp. 1355–1387, 1975.

[32] J. Zhang and M. C. Gursoy, “Collaborative relay beamforming for secrecy,” in Communications (ICC), 2010 IEEE International Conference on. IEEE, 2010, pp. 1–5.

[33] A. Mukherjee and A. L. Swindlehurst, “Robust beamforming for security in mimo wiretap channels with imperfect csi,” IEEE Transactions on Signal Processing, vol. 59, no. 1, pp. 351–361, 2011.

[34] S. Srikanth, P. M. Pandian, and X. Fernando, “Orthogonal frequency division multiple access in wimax and lte: a comparison,” IEEE Communications Magazine, vol. 50, no. 9, 2012.

[35] Y. Zou, X. Wang, and W. Shen, “Physical-layer security with multi-user scheduling in cognitive radio networks,” IEEE Transactions on Communications, vol. 61, no. 12, pp. 5103–5113, 2013.

[36] L. Zheng and D. N. C. Tse, “Diversity and multiplexing: A fundamental tradeoff in multiple-antenna channels,” IEEE Transactions on information theory, vol. 49, no. 5, pp. 1073–1096, 2003.

[37] A. Sendonaris, E. Erkip, and B. Aazhang, “User cooperation diversity. part i. system description,” IEEE Transactions on communications, vol. 51, no. 11, pp. 1927–1938, 2003.

[38] Y. Zou, Y.-D. Yao, and B. Zheng, “Opportunistic distributed space-time coding for decode-and-forward cooperation systems,” IEEE Transactions on Signal Processing, vol. 60, no. 4, pp. 1766–1781, 2012.

[39] Y. Zou, X. Li, and Y.-C. Liang, “Secrecy outage and diversity analysis of cognitive radio systems,” IEEE Journal on Selected Areas in Communications, vol. 32, no. 11, pp. 2222–2236, 2014.

[40] M. H. Yılmaz and H. Arslan, “A survey: Spoofing attacks in physical layer security,” in Local Computer Networks Conference Workshops (LCN Workshops), 2015 IEEE 40th. IEEE, 2015, pp. 812–817.

[41] M. H. Yilmaz and H. Arslan, “Impersonation attack identification for secure communication,” in Globecom Workshops (GC Wkshps), IEEE, 2013, pp. 1275–1279.

[42] F. J. Liu, X. Wang, and S. L. Primak, “A two dimensional quantization algorithm for cir-based physical layer authentication,” in Communications (ICC), 2013 IEEE International Conference on. IEEE, 2013, pp. 4724–4728.

[43] F. J. Liu, X. Wang, and S. L. Primak,, “A two dimensional quantization algorithm for cir-based physical layer authentication,” in Communications (ICC), 2013 IEEE International Conference on. IEEE, 2013, pp. 4724–4728.

[44] K. Wang, M. Wu, P. Xia, S. Xie, W. Lu, and S. Shen, “A secure authentication scheme for integration of cellular networks and manets,” in Neural Networks and Signal Processing, 2008 International Conference on. IEEE, 2008, pp. 315–319.

[45] S. Gollakota, H. Hassanieh, B. Ransford, D. Katabi, and K. Fu, “They can hear your heartbeats: non-invasive security for implantable medical devices,” ACM SIGCOMM Computer Communication Review, vol. 41, no. 4, pp. 2–13, 2011.

[46] S. Mohammadi and H. Jadidoleslamy, “A comparison of link layer attacks on wireless sensor networks,” arXiv preprint arXiv:1103.5589, 2011.

[47] Y. Wang, G. Attebury, and B. Ramamurthy, “A survey of security issues in wireless sensor networks,” 2006.

[48] K. Sharma and M. Ghose, “Wireless sensor networks: An overview on its security threats,” IJCA, Special Issue on “Mobile Ad-hoc Networks” MANETs, pp. 42–45, 2010.

[49] Y. Zhou, Y. Fang, and Y. Zhang, “Securing wireless sensor networks: a survey,” IEEE Communications Surveys & Tutorials, vol. 10, no. 3, 2008.

[50] M. Saxena, “Security in wireless sensor networks-a layer based classification,” Department of Computer Science, Purdue University, 2007.

[51] W. Znaidi, M. Minier, and J.-P. Babau, “An ontology for attacks in wireless sensor networks,” Ph.D. dissertation, INRIA, 2008.

[52] P. Reindl, K. Nygard, and X. Du, "Defending malicious collision attacks in wireless sensor networks," in Embedded and Ubiquitous Computing (EUC), 2010 IEEE/IFIP 8th International Conference on. IEEE, 2010, pp. 771–776.

[53] M. Malekzadeh and M. Ashrostaghi, "Col-mod: A new module to quantify the weight of damage incurred by collision attacks." IJ Network Security, vol. 19, no. 4, pp. 583–592, 2017.

[54] M. N. Sudha, M. L. Valarmathi, G. Rajsekar, M. K. Mathew, N. Dineshraj, and S. Rajbarath, "Minimization of collision in energy constrained wireless sensor network," Wireless Sensor Network, vol. 1, no. 04, p. 350, 2009.

[55] A. D. Wood and J. A. Stankovic, "A taxonomy for denial-of-service attacks in wireless sensor networks," Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems, pp. 739–763, 2004.

[56] A. Ouadjaout, M. Bagaa, A. Bachir, Y. Challal, N. Lasla, and L. Khelladi, "Information security in wireless sensor networks," Encyclopedia on Ad Hoc and Ubiquitous Computing, pp. 427–472, 2009.

[57] J. H. Abawajy, Internet and Distributed Computing Advancements: Theoretical Frameworks and Practical Applications: Theoretical Frameworks and Practical Applications. IGI Global, 2012.

[58] Z. A. Baig, "Pattern recognition for detecting distributed node exhaustion attacks in wireless sensor networks," Computer Communications, vol. 34, no. 3, pp. 468–484, 2011.

[59] R. K. Chang, "Defending against flooding-based distributed denial-of-service attacks: a tutorial," IEEE communications magazine, vol. 40, no. 10, pp. 42–51, 2002.

[60] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in Security and Privacy, 2003. Proceedings. 2003 Symposium on. IEEE, 2003, pp. 197–213.

[61] A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," computer, vol. 35, no. 10, pp. 54–62, 2002.

[62] M. Tanabe and M. Aida, "Preventing resource exhaustion attacks in ad hoc networks," in Autonomous Decentralized Systems, 2007. ISADS'07. Eighth International Symposium on. IEEE, 2007, pp. 543–548.

[63] M. Tanabe and M. Aida, "Secure communication method in mobile wireless networks," in Proceedings of the 1st international conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, p. 17.

[64] H. Lee, J. Yeo, S. Kim, and S. Lee, “Time slot assignment to minimize delay in ad-hoc networks,” in IST Mobile Communications Summit, 2001.

[65] D. E. H. Damian, M. L. Shaw, and B. R. Gaines, “Token-passing bus access method and physical layer specifications,” in IEEE Standard 802.4-1985. Citeseer, 2000.

[66] Z. Feng, J. Ning, I. Broustis, K. Pelechrinis, S. V. Krishnamurthy, and M. Faloutsos, “Coping with packet replay attacks in wireless networks,” in Sensor, mesh and ad hoc communications and networks (SECON), 2011 8th Annual IEEE Communications Society Conference on. IEEE, 2011, pp. 368–376.

[67] A. Mishra and W. A. Arbaugh, “An initial security analysis of the ieee 802.1 x standard,” Tech. Rep., 2002.

[68] C. Li, Z. Wang, and C. Yang, “Secure routing for wireless mesh networks.” IJ Network Security, vol. 13, no. 2, pp. 109–120, 2011.

[69] D. R. Raymond, R. C. Marchany, M. I. Brownfield, and S. F. Midkiff, “Effects of denial-of-sleep attacks on wireless sensor network mac protocols,” IEEE transactions on vehicular technology, vol. 58, no. 1, pp. 367–380, 2009.

[70] FIPS-186-2, “In digital signature standard (dss), fips pub 186-2,” Digital Signature Standard (DSS), 2000.

[71] A. Broder and M. Mitzenmacher, “Network applications of bloom filters: A survey,” Internet mathematics, vol. 1, no. 4, pp. 485–509, 2004.

[72] K. Grgi´c, V. Kri˘zanovi´c ˘Cik, and V. Mandri´c Radivojevi´c, “Security aspects of ipv6-based wireless sensor networks,” International journal of electrical and computer engineering systems, vol. 7, no. 1., pp. 29–37, 2016.

[73] C. Karlof and D. Wagner, “Secure routing in wireless sensor networks: Attacks and countermeasures,” Ad hoc networks, vol. 1, no. 2, pp. 293–315, 2003.

[74] N. Jabeur, N. Sahli, and I. M. Khan, “Survey on sensor holes: A cause-effect-solution perspective,” Procedia Computer Science, vol. 19, pp. 1074–1080, 2013.

[75] P. N. Raj and P. B. Swadas, “Dpraodv: A dyanamic learning system against blackhole attack in aodv based manet,” arXiv preprint arXiv:0909.2371, 2009.

[76] J. Sen, M. G. Chandra, S. Harihara, H. Reddy, and P. Balamuralidhar, “A mechanism for detection of gray hole attack in mobile ad hoc networks,” in Information, Communications & Signal Processing, 2007 6th International Conference on. IEEE, 2007, pp. 1–5.

[77] H. Shafiei, A. Khonsari, H. Derakhshi, and P. Mousavi, “Detection and mitigation of sinkhole attacks in wireless sensor networks,” Journal of Computer and System Sciences, vol. 80, no. 3, pp. 644–653, 2014.

[78] M. Meghdadi, S. Ozdemir, and I. G¨uler, “A survey of wormhole-based attacks and their countermeasures in wireless sensor networks,” IETE Technical Review, vol. 28, no. 2, pp. 89–102, 2011.

[79] S. Han, E. Chang, L. Gao, and T. Dillon, “Taxonomy of attacks on wireless sensor networks,” EC2ND 2005, pp. 97–105, 2006.

[80] I. Khalil, S. Bagchi, and N. B. Shroff, “Liteworp: a lightweight countermeasure for the wormhole attack in multihop wireless networks,” in Dependable Systems and Networks, 2005. DSN 2005. Proceedings. International Conference on. IEEE, 2005, pp. 612–621.

[81] T. Korkmaz, “Verifying physical presence of neighbors against replay-based attacks in wireless ad hoc networks,” in Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on, vol. 2. IEEE, 2005, pp. 704–709.

[82] Y.-C. Hu, A. Perrig, and D. B. Johnson, “Rushing attacks and defense in wireless ad hoc network routing protocols,” in Proceedings of the 2nd ACM workshop on Wireless security. ACM, 2003, pp. 30–40.

[83] W. Wang and B. Bhargava, “Visualization of wormholes in sensor networks,” in Proceedings of the 3rd ACM workshop on Wireless security. ACM, 2004, pp. 51–60.

[84] S. ¨Ozdemir, M. Meghdadi, and´Y. G¨uler, “A time and trust based wormhole detection algorithm for wireless sensor networks,” in manuscript in Turkish), in 3rd Information Security and Cryptology Conference (ISC 08), 2008, pp. 139–4.

[85] A. A. Pirzada and C. McDonald, “Circumventing sinkholes and wormholes in wireless sensor networks,” in IWWAN’05: Proceedings of International Workshop on Wireless Ad-hoc Networks, vol. 71, 2005.

[86] T. R. Andel and A. Yasinsac, “The invisible node attack revisited,” in SoutheastCon, 2007. Proceedings. IEEE. IEEE, 2007, pp. 686–691.

[87] R. Shokri, M. Poturalski, G. Ravot, P. Papadimitratos, and J.-P. Hubaux, “A practical secure neighbor verification protocol for wireless sensor networks,” in Proceedings of the second ACM conference on Wireless network security. ACM, 2009, pp. 193–200.

[88] J. Newsome, E. Shi, D. Song, and A. Perrig, “The sybil attack in sensor networks: analysis & defenses,” in Proceedings of the 3rd international symposium on Information processing in sensor networks. ACM, 2004, pp. 259–268.

[89] A.-S. K. Pathan, H.-W. Lee, and C. S. Hong, “Security in wirelesssensor networks: issues and challenges,” in Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference vol. 2. IEEE, 2006, pp. 6–pp.

[90] “A review of a sybil attack in wireless sensor network,” http://www.ivoryresearch.com/writers/15429-2/

[91] B. N. Levine, C. Shields, and N. B. Margolin, “A survey of solutions to the sybil attack,” University of Massachusetts Amherst, Amherst, MA, vol. 7, p. 224, 2006.

[92] R. Panko, Corporate computer and network security, 2/e. Pearson Education India, 2004.

[93] K.-F. Ssu, W.-T. Wang, and W.-C. Chang, “Detecting sybil attacks in wireless sensor networks using neighboring information,” Computer Networks, vol. 53, no. 18, pp. 3042–3056, 2009.

[94] M. Demirbas and Y. Song, “An rssi-based scheme for sybil attack detection in wireless sensor networks,” in Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks. IEEE Computer Society, 2006, pp. 564–570.

[95] M. Rahbari and M. A. J. Jamali, “Efficient detection of sybil attack based on cryptography in vanet,” arXiv preprint arXiv:1112.2257, 2011.

[96] M. Bogdanoski and A. Risteski, “Wireless network behavior under icmp ping flood dos attack and mitigation techniques,” International Journal of Communication Networks and Information Security, vol. 3, no. 1, p. 17, 2011.

[97] S. T. Zargar, J. Joshi, and D. Tipper, “A survey of defense mechanisms against distributed denial of service (ddos) flooding attacks,” IEEE communications surveys & tutorials, vol. 15, no. 4, pp. 2046–2069, 2013.

[98] J. Mirkovic and P. Reiher, “A taxonomy of ddos attack and ddos defense mechanisms,” ACM SIGCOMM Computer Communication Review, vol. 34, no. 2, pp. 39–53, 2004.

[99] S. Ranjan, R. Swaminathan, M. Uysal, and E. W. Knightly, “Ddos-resilient scheduling to counter application layer attacks under imperfect detection.” in INFOCOM, 2006.

[100] T. Peng, C. Leckie, and K. Ramamohanarao, “Survey of network-based defense mechanisms countering the dos and ddos problems,” ACM Computing Surveys (CSUR), vol. 39, no. 1, p. 3, 2007.

[101] C. Douligeris and A. Mitrokotsa, “Ddos attacks and defense mechanisms: classification and state-of-the-art,” Computer Networks, vol. 44, no. 5, pp. 643–666, 2004.

[102] Diego Dujovne, Thomas Watteyne, Xavier Vilajosana, and Pascal Thubert. 6tisch: deterministic ip-enabled industrial internet (of things). IEEE Communications Magazine, 52(12):36–41, December 2014.

[103] Malisa Vucinic, Jonathan Simon, Kris Pister, and Michael Richardson. Minimal Security Framework for 6TiSCH. Internet-Draft draft-ietf-6tisch-minimal-security-13, Internet Engineering Task Force, oct 2019. Work in Progress.

[104] Patrick Olivier Kamgueu, Emmanuel Nataf, and Thomas Djotio Ndie. Survey on RPL enhancements: A focus on topology, security and mobility. Computer Communications, 120:10 – 21, 2018.

[105] Amit Dvir, Tam´as Holczer, and Levente Butty´an. Vera - version number and rank authentication in rpl. 2011 IEEE Eighth International Conference on Mobile Ad-Hoc and Sensor Systems, pages 709–714, Oct 2011.

[106] Perrey Heiner, Landsmann Martin, Ugus Osman, W¨ahlisch Matthias, and Schmidt Thomas C. Trail: Topology authentication in rpl. In Proceedings of the 2016 International Conference on Embedded Wireless Systems and Networks, EWSN ’16, pages 59–64, USA, 2016. Junction Publishing.

[107] Ye Lin, Fodor Viktoria, Giannetsos Thanassis, and Papadimitratos Panos. Path metric authentication for low-power and lossy networks. In Proceedings of the 1st ACM International Workshop on Cyber-Physical Systems for Smart Water Networks, CySWater’15, pages 1–6, New York, NY, USA, 2015. ACM.

[108] Anthea Mayzaud , and Anuj Sehgal , and R´emi Badonnel , and Isabelle Chrisment , and J¨urgen Sch¨onw¨alder. Using the RPL protocol for supporting passive monitoring in the Internet of Things. In NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium, pages 366–374, April 2016.

[109] Shreenivas Dharmini, Raza Shahid, and Voigt Thiemo. Intrusion detection in the rpl-connected 6lowpan networks. In Proceedings of the 3rd ACM International Workshop on IoT Privacy, Trust, and Security, IoTPTS ’17, pages 31–38, New York, NY, USA, 2017. ACM

[110] Anhtuan Le, Jonathan Loo, Yuchen Luo, and Aboubaker Lasebae. Specification-based ids for securing rpl from topology attacks. 2011 IFIP Wireless Days (WD), pages 1–3, Oct 2011.

[111] Areej Althubaity, Huayi Ji, Tao Gong, Mark Nixon, Reda Ammar, and Song Han. Arm: A hybrid specification-based intrusion detection system for rank attacks in 6tisch networks. In 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), pages 1–8, Sep. 2017.

[112] Levis Philip, Patel Neil, Culler David, and Shenker Scott. Trickle: A self-regulating algorithm for code propagation and maintenance in wireless sensor networks. In Proceedings of

the 1st Conference on Symposium on Networked Systems Design and Implementation - Volume 1, NSDI'04, Berkeley, CA, USA, 2004. USENIX Association.

[113] Anhtuan Le, Jonathan Loo, Yuan Luo, and Aboubaker Lasebae. The impacts of internal threats towards routing protocol for low power and lossy network performance. In 2013 IEEE Symposium on Computers and Communications (ISCC), pages 000789–000794, July 2013.

[114] Fredrik Österlind, Adam Dunkels, Joakim Eriksson, Niclas Finne, and Thiemo Voigt. Cross-level sensor network simulation with cooja. In Proceedings. 2006 31st IEEE Conference on Local Computer Networks, pages 641–648, Nov 2006.

[115] Lan Zhang, Gang Feng, and Shuang Qin. Intrusion detection system for rpl from routing choice intrusion. 2015 IEEE International Conference on Communication Workshop (ICCW), pages 2652–2658, June 2015.

[116] Shahid Raza, Linus Wallgren, and Thiemo Voigt. Svelte: Real-time intrusion detection in the internet of things. Ad Hoc Networks, 11(8):2661 – 2674, 2013.

[117] Shreenivas Dharmini, Raza Shahid, and Voigt Thiemo. Intrusion detection in the rpl-connected 6lowpan networks. In Proceedings of the 3rd ACM International Workshop on IoT Privacy, Trust, and Security, IoTPTS '17, pages 31–38, New York, NY, USA, 2017. ACM.

[118] Althubaity, Areej, Tao Gong, Kim-Kwang Raymond, Mark Nixon, Reda Ammar, and Song Han. "Specification-based distributed detection of rank-related attacks in rpl-based resource-constrained real-time wireless networks." In *2020 IEEE Conference on Industrial Cyberphysical Systems (ICPS)*, vol. 1, pp. 168-175. IEEE, 2020.

Transportation Infrastructure Durability Center
AT THE UNIVERSITY OF MAINE

35 Flagstaff Road
Orono, Maine 04469
tidc@maine.edu
207.581.4376

www.tidc-utc.org