Fall 12-14-2018

Black Networks in Smart Cities

Shaibal Chakrabarty
Southern Methodist University, shaibalc@smu.edu

BLACK NETWORKS IN SMART CITIES

Approved by:

Dr. Sukumaran VS Nair
(Computer Science and Engineering)
Dissertation Committee Chairperson

Dr. Daniel W. Engels
(Office of the Provost, SMU)

Dr. Jennifer Dworak
(Computer Science and Engineering)

Dr. Eric Larson
(Computer Science and Engineering)

Dr. Jeff Tian
(Computer Science and Engineering)

Dr. Glenn Ricart
(Chief Technology Officer, US-Ignite)

BLACK NETWORKS IN SMART CITIES

A Dissertation Presented to the Graduate Faculty of the

Lyle School of Engineering

Southern Methodist University

in

Partial Fulfillment of the Requirements

for the degree of

Doctor of Philosophy

with a

Major in Computer Science

by

Shaibal Chakrabarty

M.S., Computer Engineering, Southern Methodist University
B.S., Computer Engineering, University of Houston-Clear Lake

December 15th, 2018

ACKNOWLEDGMENTS

Many people contributed to this work, by encouraging me to return to academics full-time for a PhD. After joining the PhD program, full-time, many have nurtured me through this process. Sumita Chakrabarty, a completely non-technical person, and Dr. Glenn Ricart, an acclaimed technologist and mentor, were the two people who pushed me to start the PhD. To help me jump in, it was Dr. Suku Nair, whose signature also adorns my masters thesis, and Mr. George Brody who asked me to consider SMU, after I started auditing classes at another university. Since 2014 it has been the patience and tenacity of Dr. Daniel Engels, in guiding my research and publications. Dr. Sukumaran VS Nair, Dr. Glenn Ricart, Dr. Jennifer Dworak, Dr. Eric Larson and Dr. Jeff Tian have taken my dissertation over the finish line, despite my distractions and my excuses.

A big Thank You! to all.

Chakrabarty, Shaibal M.S., Computer Engineering, SMU, 1996
B.S., Computer Engineering, University of Houston-Clear Lake, 1990

BLACK NETWORKS IN SMART CITIES

Advisor: Dr. Daniel W. Engels

Committee Chair: Dr. Sukumaran V.S. Nair

Doctor of Philosophy conferred December 15th, 2018

Dissertation completed December 15th, 2018

In this dissertation, we present the Black Networks solution to protect both the data and the metadata for mobile ad-hoc Internet of Things (IoT) networks in Smart Cities. IoT networks are gaining popularity with billions of deployed nodes, and increasingly carrying mission-critical data, whose compromise can lead to catastrophic consequences. IoT nodes are resource-constrained and often exist within insecure environments, making them vulnerable to a broad range of active and passive attacks. Black IoT networks are designed to mitigate multiple communication-based attacks by encrypting the data and the metadata, within a communication frame or packet, while remaining compatible with the existing IoT protocol.

A network of IoT nodes communicating using Black packets is called a Black Network. We transform IoT communications protocol packets into Black packets. This mechanism secures (encrypts using an authenticating cipher like Grain-128a or AES in the EAX mode) the metadata for an IoT communications protocol, in fixed-length packets (maximum allowed packet size by the protocol in use), while remaining compatible with the existing IoT protocol in use. We demonstrate Black packet design for IEEE 802.15.4, ZigBee, 6LoWPAN, Bluetooth Low Energy (BLE) and IPv6 (broadband, non-IoT communications).

We extend Black IoT packets to simple nodal communications (point to point). Simple Black network communications are inefficient, and either don't reach their destination (IoT nodes sleep a majority of the time to save power) or have high communications overhead,

rendering them impractical for deployment. We present a practical, gateway-based, star network topology, towards a Black network solution to overcome the inefficient broadcast and flooding IoT communications. We simulate simple Black communications for Flooding, Broadcast and Black Gateway and demonstrate the practicality and efficiency of Black gateway communications, compared to Shortest Path routing.

We evolve the Black IoT communication mechanisms to mesh networks which require routing. Securing the metadata (encrypted headers) creates significant challenges in routing Black packets, using traditional routing mechanisms. We present Black SDN, a Software Defined Networking (SDN) architecture for a secure Internet of Things (IoT) networking and communications. SDN architectures were developed to provide improved routing and networking performance for broadband networks by separating the control plane from the data plane. This basic SDN concept is applicable to broadband networks. However, the common SDN implementations designed for wired networks are not directly amenable to the distributed, ad hoc, low-power, mesh networks commonly found in IoT systems. SDN promises to improve the overall lifespan and performance of IoT networks. However, the SDN architecture changes the IoT network's communication patterns, allowing new types of attacks, and necessitating a new approach to securing the IoT network. Black SDN is a novel SDN-based secure networking architecture that secures both the metadata and the payload within each layer of an IoT communication packet while utilizing the SDN centralized controller as a trusted third party for secure routing, key management and optimized system performance management. We demonstrate the feasibility of Black SDN in IoT networks where nodes are asleep most of their lives, and specifically examine a Black SDN IoT network based upon the IEEE 802.15.4 LR WPAN (Low Rate - Wireless Personal Area Network) protocol, through simulations.

We extend the Black SDNs to route Black packets in a mesh network, called Black routing. This novel approach uses an SDN-based architecture for routing fixed-length, metadata-secured, Black packets from source to destination, using a ciphertext-based forwarding algo-

rithm. Both data and control Black packets hide all information on communicating parties and communication type. Fixed length packets prevent the packet-length based attacks (and prevent inference of the type of communications). Black routing configurations are extensively simulated to prove feasibility and measure the efficiency compared to traditional Shortest Path routing.

Completely secured metadata is insufficient to hide the communicating parties from sustained traffic analysis, when nodal transmission and receptions are observed. We present Node Obscuring, using tokens and a subway-model, where empty tokens traverse the network, on fixed routes, and pick up and drop off data between source and destination. Since the tokens originate at a node different from the source, and continue to traverse the network after passing thru the destination node, an external observer is unable to determine the source and the destination. This is akin to a subway picking up and dropping off passengers (data) between two stations (source and destination), while the subway (token) originates and terminates at fixed locations. We present Black routing and node obscuring algorithms, for various configurations as a part of our research. Our simulations reveal that Black routing and Node Obscuring are feasible, and can provide for much higher levels of confidentiality and privacy, resistance to a range of attacks, with a cost trade-off in overhead traffic, travel and wait times with an increase in the number of nodes.

We conclude this dissertation by applying Black networks to the Smart Cities domain, enabling secure smart cities. Smart Cities have IoT-enabled critical infrastructure (such as energy, transportation and environmental monitoring), that have already been subject to cyber attacks. Our dissertation proposes a secure IoT framework for Smart Cities that includes Black Networks, SDN Control, Key Management and Unified Registry. We further improve availability and privacy of Secure Smart City services by offering key management and mobile node authentication using distributed ledger technologies.

TABLE OF CONTENTS

LIST OF FIGURES xiv

LIST OF TABLES xvii

CHAPTER

1. **INTRODUCTION** 1

1.1. Black Packets 5

1.2. Black Networks 6

1.3. Black SDN 7

1.4. Black Routing 9

1.4.1. Some Black Routing Requirements 10

1.5. Node Obscuring (Sender and Receiver) 10

1.6. Secure Smart Cities 11

1.7. Dissertation Organization 11

1.7.1. Chapter 2: An Overview of The Internet of Things (IoT) 12

1.7.2. Chapter 3: Black IoT Communications Protocols 13

1.7.3. Chapter 4: Black IoT Communications 14

1.7.4. Chapter 5: Network Architectures 14

1.7.5. Chapter 6: Black Routing 15

1.7.6. Chapter 7: Node Obscuring 16

1.7.7. Chapter 8: Introduction to Smart Cities 16

1.7.8. Chapter 9: Secure Smart Cities 17

1.7.9. Chapter 10: Secure Smart City Services with Distributed Ledgers . 18

I Black IoT Networks 20

2. AN OVERVIEW OF THE INTERNET OF THINGS (IoT) 21

2.1. Introduction 21

2.2. IoT Devices 21

2.3. IoT Networks 23

2.3.1. Sigfox 24

2.3.2. Ingenu 25

2.3.3. LoRa 25

2.3.4. Narrowband for the Internet of Things (NB-IoT) and LTE-M (Long Term Evolution-Machine) 26

2.4. IoT Communication Protocols 28

2.4.1. Security Overview 29

2.4.2. IEEE 802.15.4 Security 31

2.4.3. Protocol Security 32

2.4.3.1. 6LoWPAN 32

2.4.3.2. ZigBee 34

2.4.3.3. WirelessHART 37

2.4.3.4. Bluetooth Low Energy (BLE) 41

2.4.4. Security Analysis 43

2.5. Software Defined Networks and IoT 44

2.5.1. SDN applied to IoT 46

2.5.1.1. Payload Size 47

2.5.1.2. Routing 51

2.5.1.3. Failure Recovery 54

3. BLACK IoT COMMUNICATIONS PROTOCOLS 58

3.1. Introduction . . . 58
3.2. Black Bluetooth Low Energy (BLE) . . . 59
3.2.1. BLE Black Advertising PDU . . . 60
3.2.2. Black BLE Data PDU . . . 62
3.2.3. Security Analysis . . . 63
3.2.4. Conclusions and Future Work . . . 64
3.3. Black IEEE 802.15.4 . . . 65
3.3.1. Black 802.15.4 Link Layer Frame . . . 66
3.4. Black ZigBee . . . 66
3.5. Black 6LoWPAN and Black IPv6 . . . 68
4. **BLACK COMMUNICATIONS** . . . 71
4.1. Introduction . . . 71
4.2. IoT Security Review . . . 74
4.3. Simple Black Network Communications . . . 76
4.3.1. Flooding . . . 77
4.3.2. Broadcast Communications . . . 78
4.3.3. Black Gateway in a Star Network . . . 78
4.4. Black Network Simulations . . . 78
4.4.1. Wait time . . . 81
4.4.2. Travel time . . . 81
4.4.3. Traffic overhead . . . 81
4.4.4. Node Reachability in Black networks . . . 81
4.4.5. Simulations for Packet Delay and Congestion . . . 82
4.5. Evaluation and Analysis . . . 85
4.5.1. How is Black Communications Different? . . . 85

4.5.2. Performance Analysis . . . 86
4.5.3. Security . . . 87
4.5.4. Black Network Compute Efficiency . . . 89
4.5.5. Black Packet Payload Efficiency . . . 89
4.6. Conclusions and Future Work . . . 89

II Black Routing **90**

5. **NETWORK ARCHITECTURES** . . . 91
5.1. Introduction . . . 91
5.2. SDN Controller: The Need for a Trusted Third Party . . . 92
5.3. Black SDN for IoT Networks . . . 93
5.3.1. Scenario 1 - Broadcast on Star Network . . . 96
5.3.2. Scenario 2 - Synchronized Mesh Network . . . 96
5.3.3. Scenario 3 - Unsynchronized Mesh Networks . . . 97
5.4. Evaluation and Analysis . . . 98
5.4.1. Network Performance . . . 98
5.4.2. Security . . . 100
5.5. Conclusions and Future Research . . . 101
6. **BLACK ROUTING** . . . 102
6.1. Introduction . . . 102
6.2. Black Packets . . . 105
6.3. Black Routing . . . 105
6.3.1. Star Control . . . 107
6.3.2. Mesh Control . . . 109
6.4. Black Routing & Node Obscuring Simulations . . . 110

6.4.1. Black Routing Simulations . . . 111
6.5. Black Routing Analysis . . . 113
6.5.1. Security Analysis . . . 114
6.5.2. Performance Analysis . . . 115
6.6. Conclusions and Future Research . . . 117
7. **NODE OBSCURING** . . . 118
7.1. Introduction . . . 118
7.1.1. Node Obscuring Simulations . . . 121
7.1.2. Node Obscuring Analysis . . . 122
7.2. Conclusions and Future Research . . . 123

III Black Networks in Secure Smart Cities **124**

8. **INTRODUCTION TO SMART CITIES** . . . 125
8.1. Introduction . . . 125
8.2. Smart City Basics . . . 126
9. **SECURE SMART CITIES** . . . 129
9.1. Introduction . . . 129
9.2. Security Overview . . . 130
9.3. A Secure IoT Architecture for Smart Cities . . . 132
9.3.1. Black Networks . . . 132
9.3.2. SDN (Software Defined Networking) Controller . . . 133
9.3.3. Unified Registry . . . 133
9.3.4. Key Management . . . 134
9.4. Evaluation of the Secure IoT Architecture . . . 135
9.5. Conclusions and Future Research . . . 136

10. **SECURE SMART CITY IoT SERVICES WITH DISTRIBUTED LEDGERS** 137

10.1. Introduction 137

10.2. Related Work in IoT and Smart Cities 139

10.2.1. External Key Management 141

10.2.2. Mobile Node Authentication 142

10.3. Distributed Ledger Technologies (DLTs) 144

10.4. Distributed Key Management with DLTs 145

10.5. Mobile Node Authentication with DLTs 146

10.6. Threat Models and Security Analysis 147

10.7. Conclusions and Future Research 148

11. **CONCLUSIONS** 149

BIBLIOGRAPHY 152

LIST OF FIGURES

Figure Page

1.1. Encryption to Black packet 5

1.2. Black Gateway Star Network Configuration 7

1.3. Black SDN in IoT Networks 8

2.1. The Internet of Things Networks [1] 23

2.2. Example of an IoT node. Source: Texas Instruments 24

2.3. Comparison of the TCP/IP and the 6LoWPAN Protocol Stacks [2]. 33

2.4. ZigBee Protocol Stack Architecture [3](www.zigbee.org) 35

2.5. WirelessHART, HART Protocol Stack [4] 38

2.6. Bluetooth Low Energy Security Stack 41

2.7. SDN Architecture and OpenFlow V1.0 Flow Table [5] [6] 45

2.8. IoT deployment with 6LoWPAN [2] 47

2.9. Sensor OpenFlow architecture [7] 48

2.10. Sensor OpenFlow flow tables, illustrating OXM and CAV extensions [7] . . . 49

2.11. Packet sizes for IPV6 networks, 6LoWPAN networks, and SDN-based networks [2]. Note the significant loss of payload size for the traditional IPv6 protocol. 50

2.12. DODAG Routing Method in 6LoWPAN [8] 51

2.13. SDN-based routing methodology 53

2.14. WirelessHART architecture, showing double-connectivity [9]. 56

2.15. SDN-based failure recovery. Broken nodes/links shown in red, with new flows shown in blue. 57

3.1. BLE Advertising PDU encryption to Black Advertising PDU 61

3.2. BLE Data PDU encryption to Black Advertising PDU 62

3.3. Black ZigBee packet and Black 802.15.4 frame 67

3.4. Black 6LoWPAN packet and Black 802.15.4 frame 69

3.5. Black IPv6 packet and Black 802.11 frame 70

4.1. Black ZigBee packet and Black 802.15.4 frame 76

4.2. BLE Data PDU encryption to Black BLE Data PDU 77

4.3. Black packet delivery probability to IoT sleeping nodes 79

4.4. Shortest Path Simulations for Black Networks 80

4.5. Flooding Simulations in Black Networks 82

4.6. Broadcast Simulations in Black Networks 83

4.7. Gateway Simulations in Black Networks with Star Topology 84

4.8. Traffic Overhead in Black Networks 86

5.1. IoT Network with an SDN Controller 94

5.2. Black SDN for IoT Control Packet example 96

5.3. Scenario 2: Black SDN Packet Latency 98

5.4. Scenario 2: Black SDN Packet Latency 99

6.1. Black 6LoWPAN packet and Black 802.15.4 frame 104

6.2. Black IPv6 packet and Black 802.11 frame 106

6.3. BSDNC configurations: Star and Mesh Control 107

6.4. BSDNC messaging to nodes using Black control packets 108

6.5. Black Routing performance in per-Black packet forwarding 111

6.6. Black Routing performance with updates and pre-defined routes 112

6.7. Black Routing performance with timed Flow refresh 113

6.8. Black Routing performance with timed Flow refresh for Pre-determined routes 114

7.1. Node Obscuring in a Black Network 119

7.2. Node Obscuring-Linear (NO-l) token travel times 122

7.3. Node Obscuring-Subway (NO-g) token travel times 123

8.1. An IoT-enabled smart city. Source: Libelium [1] 126

8.2. IoT Networks for various Smart City Functions 127

9.1. Components of a Secure IoT Architecture for Smart Cities 134

10.1. Smart City Centralized Applications 139

10.2. Key Management with Distributed Ledger in Smart Cities 140

10.3. IoT mobile node roaming architecture: Intra-cluster and Inter-cluster 143

LIST OF TABLES

Table Page

1.1. IoT Communication Protocol Security Mechanisms and Threats 2

1.2. Comparison of Anonymity Protocols 3

1.3. Main contributions of the dissertation by chapter 19

2.1. A Comparison of Widely Deployed Public IoT Networks 27

3.1. Black PDU Payload Efficiency 64

4.1. IoT Communication Protocol Security Mechanisms do not address Metadata threats 73

4.2. Black Networks Simulation Numbers for 100 nodes 88

5.1. Payload efficiency of Black Frame. 100

6.1. Performance of Black Routing, Shortest Path Routing, and Node Obscuring (T_{MW}, T_{MT} (TUs), OT (packets)) 116

9.1. Secure IoT Smart City Architecture Services 136

Dedicated to Boo and Bubli, my current wisest teachers.... ...and to Rakhal Das and Geeta Chakrabarty, my prior wisest teachers

Chapter 1

INTRODUCTION

The Internet of Things (IoT) includes networks of smart energy-efficient devices, for monitoring and control, communicating via ad-hoc, wireless networks. IoT networks are growing rapidly in healthcare, buildings, industrial control systems, energy, transportation and environmental monitoring. IoT communications protocols often run on resource-constrained devices and are vulnerable to multiple attacks, including metadata attacks. Many popular IoT protocols are based on the IEEE 802.15.4 LR-WPAN, and have well-known security vulnerabilities [10]. As billions of IoT devices are deployed, the resulting IoT networks are increasingly carrying vulnerable mission-critical data, whose compromise can result in data theft and catastrophic consequences [11]. Given resource restrictions in IoT nodes and networks, in many cases, security is either not a priority or implemented with an IT approach which makes IoT solutions economically infeasible and insecure for a range of attacks. For example, 6LoWPAN suggests an IPsec approach to security, which is not mandated. IPsec is unviable for resource-restricted IoT nodes that generally use symmetric key cryptography as opposed to public key cryptography. Key management in WirelessHART is not well-defined in the standard and can lead to improper and insecure implementations [12]. In many cases, security is just not a priority for low-power, low duty cycle resource-restricted IoT solutions. In *every* case, metadata is not secured in IoT protocols. Metadata is sent in the clear and can be traffic analyzed, manipulated and/or might be used to decipher the payload. Table 1.1 shows the security vulnerabilities of IoT communication protocols and the lack of metadata security.

IoT communication protocols, and networks, must provide security and privacy per-hop and end-to-end, and protect the metadata. Black Networks achieve this by encrypting the metadata AND the data of a frame/packet, at each layer of the communications protocol, using a stream-based cipher such as Grain-128a or AES in the EAX mode [13]. The re-

Table 1.1: IoT Communication Protocol Security Mechanisms and Threats

IoT Communication Security Mechanisms and Threats			
Protocol	*Security Services*	*Security Mechanisms*	*Security Vulnerabilities*
IEEE 802.15.4	Confidentiality	Encryption (AES-CCM* mode)	Acknowledgements are unencrypted and can be exploited. Default security is NULL. NO timed frame counters.
	Integrity	MIC	Weak integrity at 16 bits
	Replay	Frame Counter	Frame counters in the clear and can be exploited
	Privacy	None	Subject to metadata attacks
ZigBee	Confidentiality	Encryption	Trust Center is vulnerable. Network Keys can be extracted
	Traffic Analysis	None	Subject to metadata attacks
6LoWPAN	Confidentiality	Encryption	IPsec/IKE unsuitable for IoT
	Integrity	MIC	IPsec unsuitable for IoT networks
	Authentication	Node Authentication	Subject to device-based attacks
	Privacy	None	Subject to metadata attacks
	IP services	None	IP attacks (HELLO flood, sinkhole and selective-forwarding)
WirelessHART	Confidentiality	Encryption	Default security always ON.
	Integrity	MIC	
	Availability	Channel Hopping, Channel Blacklisting	Jamming
	Exhaustion	10ms time slot execution	Limits resource exhaustion, but does not eliminate
	Privacy	None	Possible metadata attacks
BLE	Confidentiality	Encryption	Key stolen during key exchange
	Integrity	CRC	CRC seed can be recovered
	Availability	Channel Hopping	Channels tracked via Access Address
	Privacy	None	Subject to metadata attacks

sulting frame/packet is then routed using a trusted third party (TTP) SDN controller [14]. Obscuring the source and destination nodes further anonymizes the network and mitigates sustained traffic analysis attacks.

The IEEE 802.15.4 which defines the PHY and MAC sublayer, forms the basis for multiple higher layer protocols - ZigBee, 6LoWPAN and WirelessHART - being the most widely-

used. IoT communication protocol vulnerabilities of IEEE 802.15.4, ZigBee, 6LoWPAN, WirelessHART and Bluetooth Low Energy (BLE) have been exhaustively researched [14] [15]. A survey of the literature related to IoT networks present a range of security issues that have not been completely evaluated or solved with the mechanisms presented in the literature (such as node obscuring). Black networks mitigate internal and external threats, a range of active and passive attacks, and secure the communications per-hop and end-to-end. To prevent inference attacks, and packet-length attacks, Black networks communicate using fixed length packets (the maximum size allowed by the IEEE 802.15.4 protocol). Securing the metadata leads to routing challenges for the network. An SDN Controller, functioning as a TTP (Trusted Third Party), for IoT networks, provides guaranteed node reachability [14], by forwarding the encrypted packets to their destination, without intermediate packets having any knowledge of the source or destination. However, sustained external traffic analysis can determine the source and destination of packets, thereby identifying the communicating parties. The objective of Black Networks is to hide all information between source and destination, including the source and destination. Obscuring the source and destination during communications mitigates this threat.

Table 1.2: Comparison of Anonymity Protocols

Routing Protocol	**ARMR [16]**	**MASK [17]**	**DASR [18]**	**AnonDSR [19]**
Identity Anonymity	Yes	No	Yes	Yes
Location Anonymity	Yes	Yes	Yes	No
Route Anonymity	Yes	Yes	Yes	No

Several anonymous routing protocols such as AASR, ODAR, OLAR and USOR have been developed for mobile ad hoc networks but they are not suitable for sensor networks due to resource limitations and also the incapability to totally obscure the identity of the communication nodes (see Table 1.2) [20] [21] [22] [23] [24] [18] [16]. Onion Routing (Tor) is

the de-facto standard for anonymity and privacy in web-based applications [25]. A fixed route is selected by the Tor client. Intermediate nodes (called onion routers) have no knowledge of other nodes in the network, except for the node before, and the node after it. Tor uses public key cryptography for transmission. The source negotiates a session key with the every successive hop. Once this hop is completed, the key is destroyed, thereby mitigating replay attacks. Tor cannot be directly applied to IoT networks that use symmetric keys and are resource-constrained. We propose mechanisms for sender node obscuring, receiver node obscuring, and path obscuring, for IoT networks, using existing protocol compliance and symmetric keys.

In the following sections, we will introduce the core research components of the dissertation: Black packets, Black networks, Black SDN, Black routing, node obscuring and secure smart cities. Section 1.1 outlines a high-level Black packet design with a specific example of the ZigBee PDU (Packet Data Unit). In Section 1.2, we introduce Black Networks, along with existing work of privacy-preserving and anonymous protocols. We demonstrate how Black networks provide metadata security, anonymity and compatibility with the existing IoT protocol in use, for simple communications with a Black Gateway. We present Black SDN network architecture in Section 1.3 to facilitate routing for Black packets in mesh networks, beyond the simple point-to-point communication. The Black SDN functions as a TTP and performs the key management function. In Section 1.4 we present Black Routing between IoT nodes, using a Black SDN Controller. The two configurations analyzed are: a) the Black SDN controller has a direct communication path to ALL IoT nodes; b) the Black SDN controller has a direct signalling path to only SOME of the IoT nodes. We introduce further mechanisms to obscure communicating nodes and the communicating route for Black Routing in Section 1.5. Lastly, we apply Black networks to a use case of IoT-enabled Smart Cities in Section 1.6. With authentication, key management, SDN networking and Black networks, we propose Secure Smart Cities - safe from cyberterror and cyberattacks. Section 1.7 introduces each chapter of the dissertation and its main contributions, concluding with a final contribution summary.

1.1 Black Packets

The PHY and MAC sublayer of ZigBee is defined by the IEEE 802.15.4, the Network and Application layers are defined by the ZigBee specification. Security for the NWK (network) and APL (application) layers are provided by the Security Services Provider. A 14-byte Auxiliary Header is included to provide security specific information, such as frame counters for replay attacks, security levels and nonce fields. A secured ZigBee NWK packet is not always encrypted, as IEEE 802.15.4 contains security options of NO security and integrity protection only [26]. Figure 1.1 shows the transformation of a secure ZigBee Data PDU into a Black ZigBee packet.

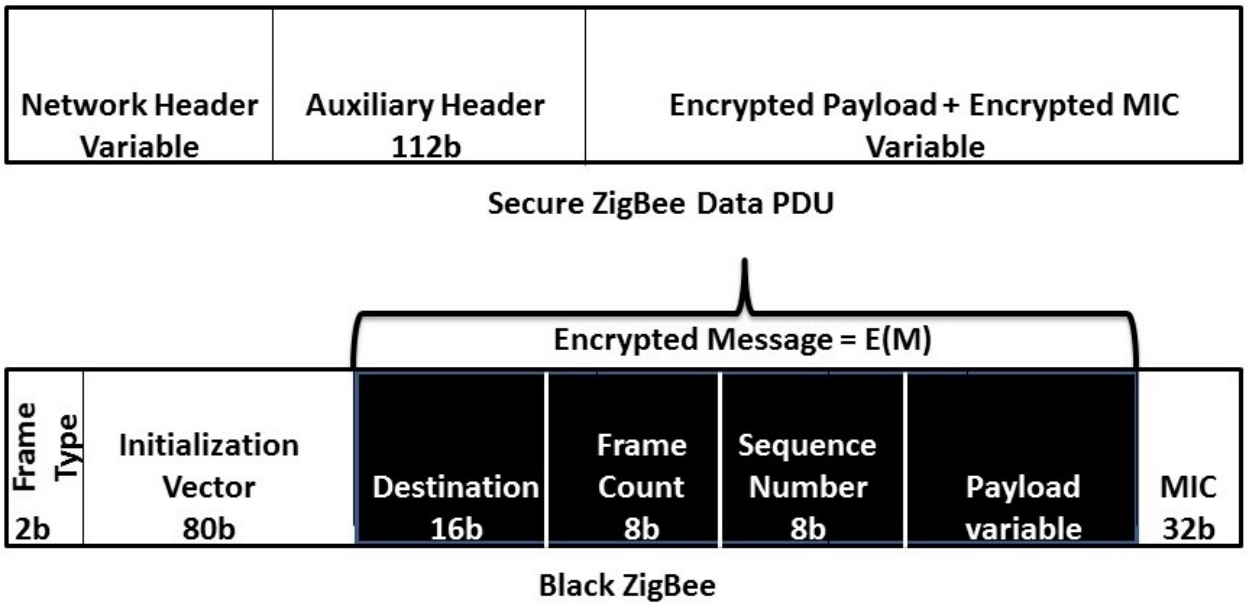


Figure 1.1: Encryption to Black packet

The Link layer transformation, from standard 802.15.4 to Black 802.15.4, is outlined in [14] [27]. The ZigBee network layer packet header contains 16 bits of frame control information. The first subfield in the frame control field is - Frame Type (data, control or command) is maintained for the Black ZigBee packet. The Frame Type reserved bits 11 are used to indicate a Black ZigBee packet. Except the first 2 bits, the rest of the packet is

encrypted using the AES-EAX cipher, as in Figure 1.1. Since packet forwarding is done via an IoT SDN Controller (a TTP), many of the addressing and routing sub-fields in the ZigBee network packet are no longer necessary. The recommended 80-bit Initialization Vector (IV) is included with each packet. AES-EAX is an authenticating cipher mode where the ciphertext is equal to the message length, with a flexible IV size, allowing for better payload efficiency in small frame sizes like the IEEE 802.15.4. This is a Black packet design that will form the basis of a Black IoT network of ZigBee nodes.

1.2 Black Networks

A Black Network is a network that secures each layer of the communication stack by encrypting all of the metadata contained within the communication (including the source and the destination addresses), in addition to the payload. We introduce Black Networks to mitigate traffic analysis and data gathering attacks. With encrypted source and destination addresses, only simple communications are possible in Black networks - Flooding and Broadcast routing. Our extensive simulations examine the impact of the routing performance of the Black network, in comparison to Shortest Path routing. The inefficiency and high overheads in Black networks with simple communications identify the need of a Trusted Third Party (TTP) in order to maintain efficient communications. A star network with a Black Gateway as the TTP is proposed. A majority of deployed IoT networks have star configurations (including public IoT networks discussed in Chapter 2) [28] [29]. Our proposed architecture using a Black Gateway is simple, practical and easily deployed to provide end-to-end confidentiality, integrity and privacy with Black networks.

Figure 1.2 shows a Black Gateway configuration with nodes communicating with each other using Black packets. The Black Gateway acts as a TTP and performs the key management function. If Node A wants to send a packet with Node D, then Node A encrypts the packet with K_A (unique key of Node A) and sends it to the Black Gateway. The Black Gateway decrypts and re-encrypts the packet with key K_D and forwards the packet to Node D. The Black Gateway is not an IoT node and our assumption is that it does not have compute, memory and capacity constraints. Our simulations show the Black Gateway performance to be equivalent or better than Shortest Path routing, while providing much higher levels of

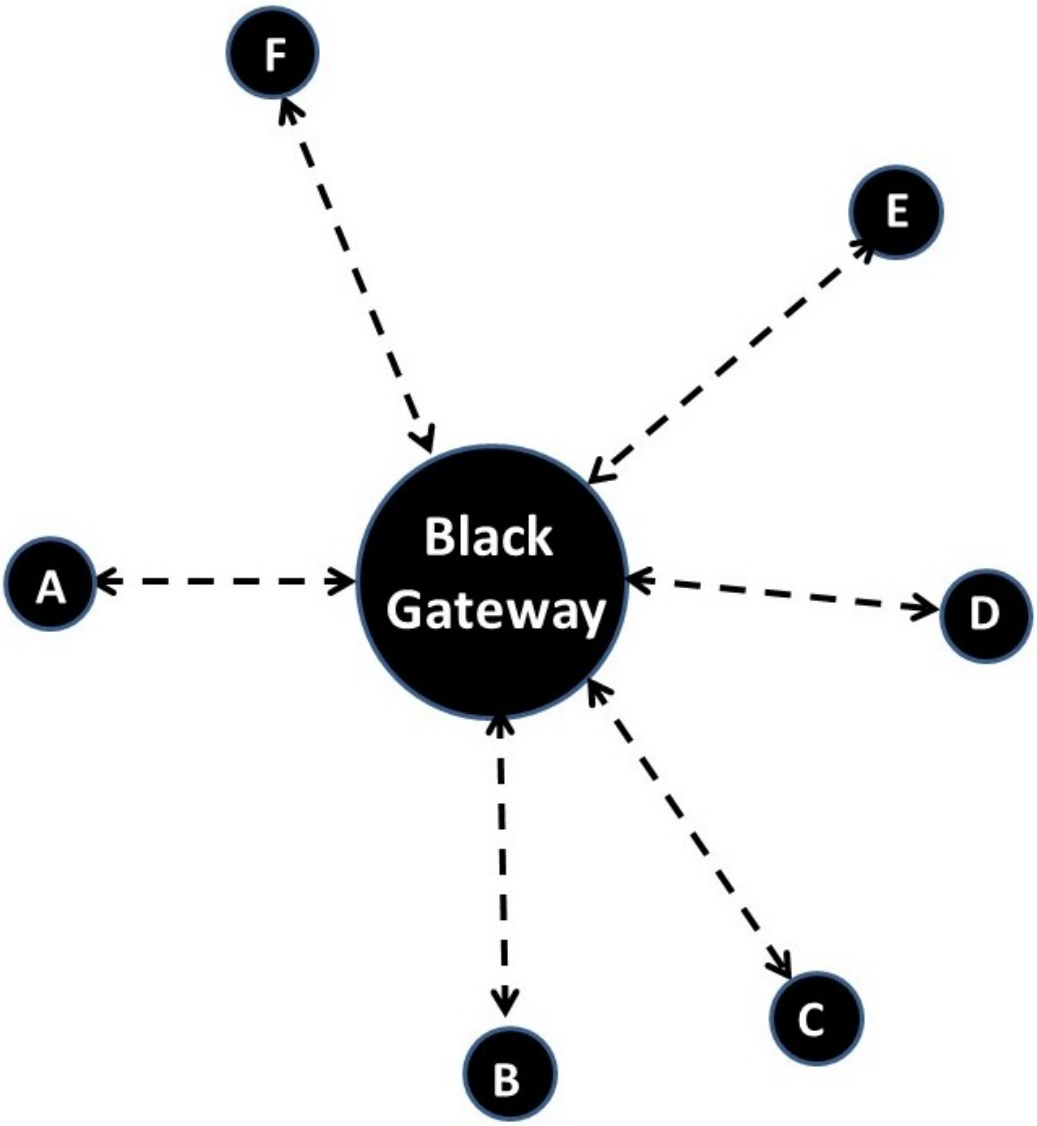


Figure 1.2: Black Gateway Star Network Configuration

metadata, traffic analysis and insider threat security.

1.3 Black SDN

The Black Gateway configuration in Section 1.2 allows for simple, efficient and practical point-to-point deployments for Black Networks. To go further, a routing mechanism for Black networks is needed, for mesh networks. How does a system route a packet with

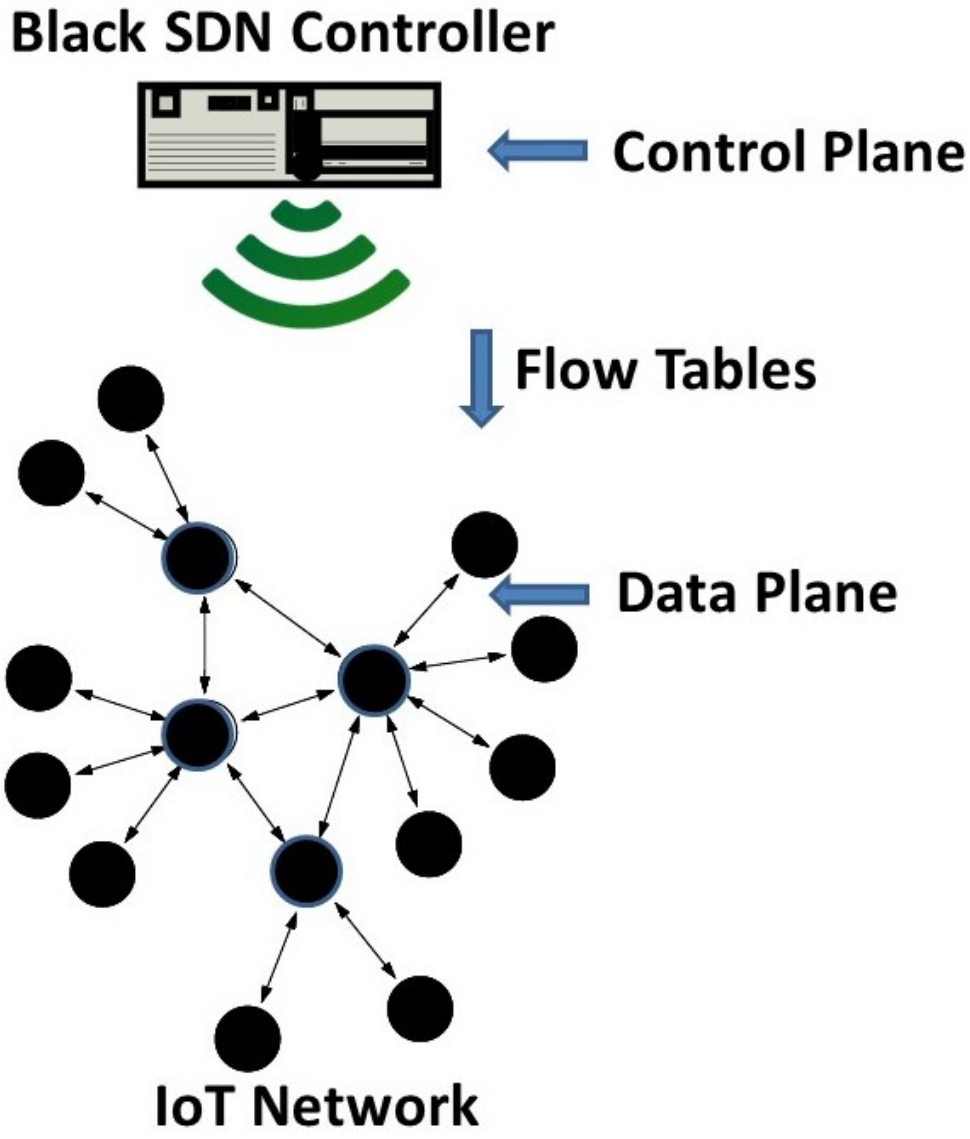


Figure 1.3: Black SDN in IoT Networks

encrypted headers? We propose a Software Defined Networks (SDN) approach to routing Black packets. All control messaging to and from the Black SDN Controller (BSDNC, used interchangeably with TTP) with the nodes is done using Black packets. The data packets being forwarded by the Black SDN are also Black packets. We employ a ciphertext-based forwarding mechanism to efficiently route Black packets from source to destination. Chapter 5 discusses the network architecture, requirements, messaging and security of Black

SDN. Two Black SDN configurations are evaluated - Star Control and Mesh Control. The Star Control Black SDN configuration is one in which the BSDNC/TTP can talk to all network nodes directly. The Mesh Control Black SDN configuration can talk to *some* of the nodes directly, and to other nodes indirectly. The Mesh Control is a more complex scenario, albeit a practical one. While the dissertation focuses on IoT networks overall, starting in this chapter we begin to introduce concepts beyond IoT networking and indicate that the security mechanisms described in this dissertation are applicable to ANY communications protocol. The concept of a Black SDN for an IoT network is shown in Figure 1.3

1.4 Black Routing

To present Black Routing, we apply the concepts of Software Defined Networks (SDN) [30], a new routing paradigm, to an IoT mesh network. Existing IoT protocols like ZigBee, 6LowPAN and WirelessHART, utilize nodal routing mechanisms (where nodes designated as FFDs (Full Function Devices) are capable of routing IoT packets. A PAN co-ordinator, which is necessarily an FFD, co-ordinates the join, remove, and neighbor lists of nodes). An SDN Controller performs the routing functionality, by downloading forwarding tables to the IoT nodes. This simplified routing architecture (by separating the control plane and the data plane) allows for multi-protocol support, and less routing functionality (i.e. more compute and memory) within the forwarding nodes. Specifically SDN Controllers use protocols (OpenStack, OpenFlow and OpenDaylight) that are currently applicable to broadband networks and are changing the way traditional routing is done within data centers, enterprise networks and the Internet. An SDN architecture applied to IoT networks requires a simplified controller protocol and interfaces [7] [31]. The security of IoT networks, using an SDN-based architecture is a recent field of study [14] [32].

The primary objective of Black Networks is to secure all data, including the metadata, associated with each frame or packet. This security is achieved by encrypting all information contained in a frame that may be used by an adversary as shown in 1.1. Routing becomes challenging when packet headers are encrypted. Black packets use an SDN architecture to achieve routing from Point A to Point B in a mesh network. The SDN controller, routing Black packets, in an IoT network is called a Trusted Third Party (TTP). The shared secret

(symmetric key) used between the TTP and Node n is called a node key. Each node has a different node key, and is designated by K_N for Node n. Black packets, Black SDNs and their routing challenges are introduced in [14]. We consider two network topologies for Black Routing:

- The TTP is 1 hop away from all nodes
- The TTP is more than 1 hop away from nodes

1.4.1 Some Black Routing Requirements

Routing the Black ZigBee packet from source to destination requires:

- *Destination Address*: The final destination address must be included in the Black Packet (16 bits For ZigBee networks).
- *Initialization Vector (IV)*: The 128-bit unique IV must be included in each Black packet, to synchronize the cryptographic engines on both sides.
- *Frame and Sequence Counters*: Frame counters are needed for mitigating replay attacks. Sequence counters are needed for re-assembly of multiple data packets.

In this dissertation, we focus on a IoT network packet transformation to a Black packet, and the subsequent routing of the Black packet from source to destination within the network. We assume NO ACKs, NO obfuscation (source, destination or path), NO node authentication and with zero sleep time for ALL nodes.

1.5 Node Obscuring (Sender and Receiver)

Despite link layer encryption being applied to Black networks, the identities of the communicating parties is put to risk. An intruder performing sustained traffic analysis on a network can find the origin and destination of a Black packet. Black networks are vulnerable to revealing the identity of the communicating nodes.

We present security mechanisms to obscure the sender, the receiver, and the path of the communication in IoT networks, thereby securing their identities. We obscure the sender by generating tokens from a random node (which is a random number of hops before the sender)

to the sender, and along the path between the sender and the receiver. Black network tokens are empty, fixed length (127-byte) packets. The token is sent along the precalculated path until the sending node receives the token. It adds the payload to the packet and resends it along the same path, thereby obscuring the identity of the sender. To obscure the receiver, we resend the Black packet beyond the receiving node, for a random number of hops. When the receiver gets the packet, it processes the content intended for it and re-sends the packet for a random number of hops to a random node, obscuring the receiver.

1.6 Secure Smart Cities

Smart Cities are increasingly IoT-dependent, with critical-infrastructure data and control messaging being exchanged over vulnerable IoT networks. Black networks enable Secure Smart Cities. Secure Smart Cities have Black networks to secure the critical-infrastructure communications; a Unified Registry to authenticate and certify the IoT devices in its network; an SDN architecture to enable Black networks and streamline its broadband communications infrastructure; and an external key management system to mitigate the vulnerable key management systems of IoT protocols. The centralized security and services architecture of smart cities today are vulnerable to targeted cyberattacks or natural disasters. A distributed Secure Smart City services model is suggested for key management and IoT mobile node authentication.

1.7 Dissertation Organization

The dissertation is organized in three parts, based on the phases of our research. Part I consists of three chapters: Chapter 2 is an overview of IoT systems - from devices, to a review of popular public IoT networks, a comprehensive survey and security analysis of IoT communications protocols and advanced networking technologies (Software Defined Networks - SDN) for IoT, enabling global sensor networks for Smart Cities; Chapter 3 presents the design of Black packets for each of the popular IoT communication protocols discussed in this dissertation - Bluetooth Low Energy (BLE), IEEE 802.15.4, ZigBee, 6LoWPAN and IPv6 (non-IoT protocol) and their compatibility with existing IoT communications protocols; Chapter 4 presents Black communications. We evaluate the security and performance of simple Black communications in comparison to Broadcast and Flooding in IoT networks.

Our results are based on extensive simulations of Black networks of increasing scale (up to 1000 nodes) in different configurations. The outcome allows us to propose a practical Black gateway solution for an IoT network that allows for much better performance, simple implementation, yet maintains a high resistance to a broad range of active and passive attacks, and some level of node obscurity. The three chapters of Part II are: Chapter 5 describes the network architectures enabling Black communications, introducing an SDN-based architecture for routing Black packets in mesh networks. We evaluate and analyze the security and performance of a Black SDN architecture for different configurations of Black networks; Chapter 6 introduces Black Routing, using an SDN architecture and comprehensively simulates the performance of Black routing to Shortest Path routing, for networks of increasing scale, and for various SDN configurations. We provide algorithms for Black routing and methods for improving the performance. Chapter 7 evolves the security of Black routing using Node Obscuring of source and destination communicating nodes, and the communications path. The final three chapters of Part III are: Chapter 8 provides an introduction to Smart Cities and their security framework. Chapter 9 presents Secure Smart Cities using a Black networks architecture. Chapter 10 proposes a distributed architecture for key management and mobile IoT nodes for secure smart cities. Chapter 11 draws relevant conclusions and suggests new areas of research.

Part I - Black IoT Networks: Chapter 1, Chapter 2 and Chapter 3

1.7.1 Chapter 2: An Overview of The Internet of Things (IoT)

This chapter provides an overview of the IoT ecosystem and its role in forming complex, aggregate systems that enable smart cities. The IoT ecosystem is huge - IoT standards, IoT chipsets, IoT nodes, public IoT networks, low power and energy harvesting IoT modules, IoT gateways and IoT security, to name a few. We overview IoT devices (nodes) and popular public IoT networks (Sigfox, Ingenu, LoRa/LoRaWAN), including the recently deployed cellular public IoT networks (LTE-M and NB-IoT). We evaluate popular IoT communications protocols based on IEEE 802.15.4 (ZigBee, 6LoWPAN, WirelessHART) and BLE to provide an exhaustive security analysis for wireless IoT communications protocols. We introduce

Software Defined Networks (SDNs), and propose a simplified SDN architecture for use in wireless IoT networks, as a trusted third party (TTP) or gateway, to route packets between nodes in an IoT network. The main contributions of this chapter are: an evaluation and comparison of public IoT networks - Sigfox, Ingenu, LoRa, NB-IoT and LTE-M; a comparison and security analysis of popular IoT communications protocols - IEEE 802.15.4, ZigBee, 6LoWPAN, WirelessHART and BLE; and the introduction of SDN architectures for IoT and the evaluation of payload effciency, failure recovery and routing in such an architecture.

1.7.2 Chapter 3: Black IoT Communications Protocols

In this chapter, we evaluate and compare the security capabilities and vulnerabilities of three popular Internet of Things (IoT) protocols: 6LowPAN, ZigBee and WirelessHART. The IoT is exploding in its deployments across a broad range of objects, devices, environments and applications [33]. Wireless communications are the primary means by which many of the objects and devices are connected [34], and the IEEE 802.15.4 standard for low-rate wireless personal area networks is a commonly used foundation for IoT communication protocols. The IEEE 802.15.4 standard defines the PHY and Link layers in the communication stack while 6LowPAN, ZigBee and WirelessHART define the Network, Transport and portions of the Application layers. Security is a primary concern for many IoT applications, particularly for 6LowPAN, ZigBee and WirelessHART, due to the personal, financial and automated operational control nature of the applications that use these protocols. In our analysis, we find that WirelessHART has the greatest array of security mechanisms available to secure its communications and basic operations from traditional attacks such as eavesdropping, message modification and message injection. However, all three protocols are vulnerable to an array of more sophisticated attacks including node capture and resource exhaustion.

The main contributions of this paper are: the presentation of a novel security framework that may be used to assess the security of all IEEE 802.15.4-based protocols; using our framework to perform an analysis and comparison from the security standpoint of the 6LowPAN, ZigBee and WirelessHART protocols; and a brief analysis of the security provided within IEEE 802.15.4-2011.

1.7.3 Chapter 4: Black IoT Communications

In this chapter, we present Black Networks for secure communications in the Internet of Things (IoT). IoT networks have billions of deployed nodes that are communicating mission-critical data and control messages. The compromise of a node, the data, or the control messages can lead to catastrophic consequences. And, in many cases, the compromise of the metadata associated with these communications can reveal compromising information to an attacker. Black Networks are designed to mitigate multiple communications-based attacks by securing both the data and the metadata within a communication frame or packet, while remaining compatible with the existing communication protocol in use. We focus on the Black Network approach for Black ZigBee and Black BLE (Bluetooth Low Energy). The metadata is protected, in part, by encryption and by using maximum sized packets for all communications. Our simulation results for Flooding, Broadcast and Gateway communications, demonstrates the viability of Black Networks in wireless IoT networks and complex topology wired networks.

The main contributions in this chapter are the presentation of Black Zigbee and Black BLE packet designs and the simulation and characterization of two simple communication mechanisms (Flooding and Broadcast) for a mesh network and a Black Gateway star network using Black packets.

Part II - Black Routing: Chapter 5, Chapter 6 and Chapter 7

1.7.4 Chapter 5: Network Architectures

In this chapter, we present Black SDN, a Software Defined Networking (SDN) architecture for secure Internet of Things (IoT) networking and communications. SDN architectures were developed to provide improved routing and networking performance for broadband networks by separating the control plain from the data plain. This basic SDN concept is amenable to IoT networks; however, the common SDN implementations designed for wired networks are not directly amenable to the distributed, ad hoc, low-power, mesh networks commonly found in IoT systems. SDN promises to improve the overall lifespan and performance of IoT networks. However, the SDN architecture changes the IoT network's communication

patterns, allowing new types of attacks, and necessitating a new approach to securing the IoT network. Black SDN is a novel SDN-based secure networking architecture that secures both the metadata and the payload within each layer of an IoT communication packet while utilizing the SDN centralized controller as a trusted third party for secure routing and optimized system performance management.

We demonstrate through simulation the feasibility of Black SDN in networks where nodes are asleep most of their lives, and specifically examine a Black SDN IoT network based upon the IEEE 802.15.4 LR WPAN (Low Rate - Wireless Personal Area Network) protocol.

The major contributions of this chapter are: the concept of Black Networks and Black Routing with an SDN-based architecture.

1.7.5 Chapter 6: Black Routing

In this chapter, we present *Black Routing* solutions, the routing of fully encrypted, fixed length packets, called Black packets, through a network. Black packets secure both the data payload and the packet metadata from an eavesdropper. Networks that utilize black packets (Black networks) mitigate a wide variety of passive, active, insider, and metadata-based attacks. Standard network routing protocols do not work with encrypted metadata, and they reveal both the source and destination nodes to attackers. Thus, Black packets traditionally require the use of expensive broadcast routing or flooding to communicate a single packet from source to destination. We present source and destination obscuring Black Routing algorithms that provide efficient routing of Black packets utilizing a Software-Defined Network (SDN) architecture to achieve both high performance and secure communications. We simulated our algorithms on a range of network topologies and found that Black Routing with node obscuring can achieve performance within 50% of traditional shortest path routing while providing node anonymity and resistance to attacks including traffic analysis.

The main results of this chapter are: Black routing algorithms with an SDN architecture in the star control and mesh control configurations and Black routing performance simulations, based on 6LoWPAN and IPv6 Black packets.

1.7.6 Chapter 7: Node Obscuring

In this chapter we present Node obscuring mechansims. Despite link and network layer encryption in Black networks, the identities of communicating parties can be determined via sustained traffic analysis on an IoT network. We present security mechanisms to obscure the source and the destination in IoT networks. We obscure the source by generating tokens from a random node along the path connecting the source and the destination, before the source. Black network tokens are empty, fixed-length packets. The token is sent along the pre-calculated path until the source node receives the token. It adds the payload to the packet and resends it along the same path, thereby obscuring the source identity. When the destination receives the token, it processes the content intended for it and re-sends the packet for a random number of hops, beyond the destination, obscuring the destination. During the communication session, we establish a pre-defined path to obscure the source and the destination from an adversary. The term subway communications is used to describe this algorithm, where a token travels along a pre-determined path, picks up and drops off data, obscuring the source and destination points.

Analysis of Node obscuring shows messaging and compute proportional to the number of nodes in the path, with better payload efficiency significantly higher security than existing IoT protocols.

The chapter contributions are: the node obscuring algorithms (node obscuring linear (NO-l) and grid (NO-g)); security and performance analysis of node obscuring mechanisms.

Part III - Black Networks in Secure Smart Cities: Chapter 8, Chapter 9 and Chapter 10

1.7.7 Chapter 8: Introduction to Smart Cities

Half of the world's population resides in urban areas [35]. This drive towards urbanization is caused by many factors including a search for better opportunities, healthcare and citizen services that are not widely available in rural areas. As cities grow due to this trend, there is increasing stress on them to continue providing the necessary citizen services like emergency response and improve quality of life (air, water and food quality), and make the delivery of these services seamless and efficient to deal with a growing urban population [36]. City

managers have turned to ICT (internet and communications technologies) to deliver these services to citizens for Smart Cities. Smart cities are increasingly IoT-enabled and providing ever more sophisticated services of emergency response and management of critical infrastructure. They are also a growing target for cyberterror and cyberwarfare, by exploiting IoT vulnerabilities [37]. An adversary could cripple a city, by shutting off critical infrastructure, being managed by IoT networks - without being physically present. This dissertation focuses on this aspect, and presents an introduction to Smart Cities, before presenting Secure Smart Cities in Chapter 9.

1.7.8 Chapter 9: Secure Smart Cities

In this chapter, we introduce a secure Internet of Things (IoT) architecture for Smart Cities. Smart cities are increasingly deploying IoT networks for improved city management such as critical infrastructure monitoring, energy management and environment monitoring. Mission-critical Smart City data carried over IoT networks must be secured to prevent cyber attacks that might cripple city functions, steal personal data and inflict catastrophic harm. The security of Smart Cities is based on the security provided by IoT networks. We propose four architectural blocks needed to secure an IoT-enabled Smart City against advanced attacks – Black Networks, Trusted Third Party (TTP), Unified Registry and Key Management System. The resulting design provides identity, authentication, authorization, confidentiality, integrity, availability and privacy. We propose that this security architecture can be replicated across multiple city functions, for a holistic approach to securing Smart Cities.

The main contribution of this chapter is a secure IoT architecture for Smart Cities. The framework consists of Black Networks, SDN Controller as TTP, Unified Registry and Key Management. The security services extend beyond the basic security provided by IoT protocols to confidentiality, integrity, availability, privacy, identity management, authentication, authorization, and accounting - across heterogeneous IoT networks, across multiple device types, and for multiple Smart City functions. The security services provided mitigate the vulnerabilities of basic IoT networks, for mission-critical data, at the Link and Network layers.

1.7.9 Chapter 10: Secure Smart City Services with Distributed Ledgers

In this chapter, we present a distributed model for delivering Secure Smart City services using distributed ledger technologies (DLTs). A centralized management model for Smart Cities is the norm - Network Operations Centers (NOCs) monitor smart cities, respond to emergencies and deliver citizen services. The data collected for smart city services, such as parking, lighting, environment and/or citizen records are stored in a centralized data center, or cloud. Such a centralized architecture is vulnerable to sustained malicious attacks, breaches, and natural catastrophe. With smart cities becoming increasingly IoT enabled, we propose a distributed architecture to deliver two IoT services: key management and node mobility, using DLTs, to mitigate attacks on a centralized architecture. The main contributions of this paper are: a pooled, distributed model for Key Management in smart cities, and decentralized mobile node authentication for IoT networks in smart cities.

Table 1.3: Main contributions of the dissertation by chapter

Table of Contributions		
Major Contribution	*Minor Contributions*	*Notes*
IoT Security Survey	Security Analysis of IEEE 802.15.4 Security Analysis of ZigBee Security Analysis of 6LoWPAN Security Analysis of WirelessHART Security Analysis of Bluetooth Low Energy	Chapter 2 Thesis Motivation: Metadata Security Comparitive Security Analysis of all protocols
Black Packets "Best Paper" Award: "Black Networks for BLE" Published in IEEE ICCE	Black IEEE 802.15.4 Black ZigBee Black 6LoWPAN Black BLE Black IPv6	Chapter 3 Black packet design; Security analysis and payload efficiency of Black packet vs. corresponding IoT protocol. non-IoT Black packet
Black Communications "Black Networks" Submitted: IEEE IoT Journal	Black networks Flooding simulation Black networks Broadcast simulations Black Gateway architecture	Chapter 4 Performance Analysis Security and Compute Analysis
Black SDN "Black SDN for IoT" Published in IEEE MASS	Applying SDN architecture to IoT Applying SDN architecture to wireless SDN control messaging using Black packets SDN used for communications security	Chapter 5 Node reachability for Star and Mesh Black SDN networks
Black Routing "Black Routing" Submitted: IEEE IoT Journal	Black routing algorithm with Star SDN config Black routing algorithm with Mesh SDN config Black routing simulations in 4 configurations	Chapter 6 Security and Performance analysis vs Shortest Path
Node Obscuring Included in:"Black Routing"	Linear Node Obscuring Algorithm Grid Node Obscuring Algorithm	Chapter 7 Security and Performance analysis
Secure Smart Cities "Secure IoT Architecture for Smart Cities Published in IEEE CCNC	Framework for a Secure Smart City Black Networks, SDN Networking Key Management, Unified Registry	Chapter 9 IoT-enabled Secure Smart City
Secure Smart City Services with Distributed Ledgers	Key Management Mobile IoT node Authentication	Under submission IEEE IoT Journal or IEEE Smart Cities Journal

Part I

Black IoT Networks

Chapter 2

AN OVERVIEW OF THE INTERNET OF THINGS (IoT)

In this chapter we review the technologies related to, and the current work on the Internet of Things (IoT). The IoT ecosystem encompasses semiconductors and devices, networks and communication protocols. As billions of IoT devices are deployed, the business of IoT grows exponentially [33]. And just like normal businesses, they can be disrupted due to vulnerabilities.

2.1 Introduction

IoT deployments present a huge and easy attack surface to adversaries, given their resource constraints, small form factor and low cost. All of these factors have caused the IoT ecosystem to be vulnerable to accidental or intentional attacks [15]. We present IoT devices in Section 2.2. We introduce public IoT networks - SigFox, LoRa and Ingenu, as well as cellular IoT networks based on NB-IoT and LTE-M in Secton 2.3. In Section 2.4, we introduce the popular IoT communications protocols of IEEE 802.15.4, ZigBee, 6LoWPAN, WirelessHART and Bluetooth Low Energy (BLE), and provide an exhaustive security analysis of these protocols. Lastly, In Section 2.5 we introduce Software Defined Networks (SDNs), and present an SDN architecture for managing an IoT network, evaluating routing, payload size and failure recovery.

2.2 IoT Devices

At the most basic level, IoT devices are primarily embedded systems, that collect and transmit data. Many are implemented as SoCs (System on a Chip) - an IC with a collection of sensors (such as temperature, pressure, gyroscope, etc) [38]. We encounter the Internet of Things (IoT) regularly in our daily lives - in homes, manufacturing automation, smart meters, supply chain management, automated tolling and healthcare. IoT systems are becoming pervasive across a broad range of applications and in a diverse set of environments. This

expansive adoption is fueled by reduced costs and improved performance of IoT devices. It is estimated that 50 billion devices will be connected by 2020 [33], leading to the formation of networks that will continuously monitor (sense), and transmit data as shown in Figure 2.1. IoT integration, both in our surroundings and on our persons, combined with the mission-critical nature of many of these systems in automated control, personal monitoring and financial transactions, mandates strong security mechanisms within IoT devices to protect us from intentional and accidental harm. IoT devices are often small, resource-constrained devices that may be either mobile or mounted to a fixed location. Many IoT devices are powered by batteries that must power the device for several years. They may form low-power, ad-hoc, low data rate, wireless networks. The devices conserve power by utilizing intermittent 'sleep' functionality where the device is neither transmitting nor receiving in order to maximize the lifetime of the limited battery source.

Given the size of these nodes, they have computational, memory, range of operation and energy constraints and must run efficient software protocols. A collection of these IoT nodes form ad-hoc IoT networks, and communicate by means of an energy-efficient, wireless, personal area network, protocols. A widely used base protocol for IoT is IEEE 802.15.4 LR-WPAN (Low Rate Wireless Personal Area Networks) [39]. 802.15.4 defines the Physical layer and the MAC-sublayer of the Link layer of the communications protocol. The network, transport and application layers are defined by protocols that are built on top of 802.15.4 such as 6LoWPAN [40], ZigBee [3] and WirelessHART [9]. Another commonly used high rate IoT protocol is Bluetooth Low Energy based on IEEE 802.15.1 WPAN [41]. Figure 2.2, shows the Texas Instruments CC2250 IoT microcontroller.

These chipsets, along with sensors, are used in IoT nodes connected by gateways to form wide-area networks (WANs) of nodes using the above wireless protocols. The limited resources and power constraints impact the security capabilities of IoT devices. IoT protocols have well-known vulnerabilities [15]. In addition, practical attacks, such as node capture (where the node is physically accessed) and resource exhaustion, and metadata attacks (where the meta-data is modified, or used for inference and traffic analysis attacks) are all specific to IoT nodes and networks.

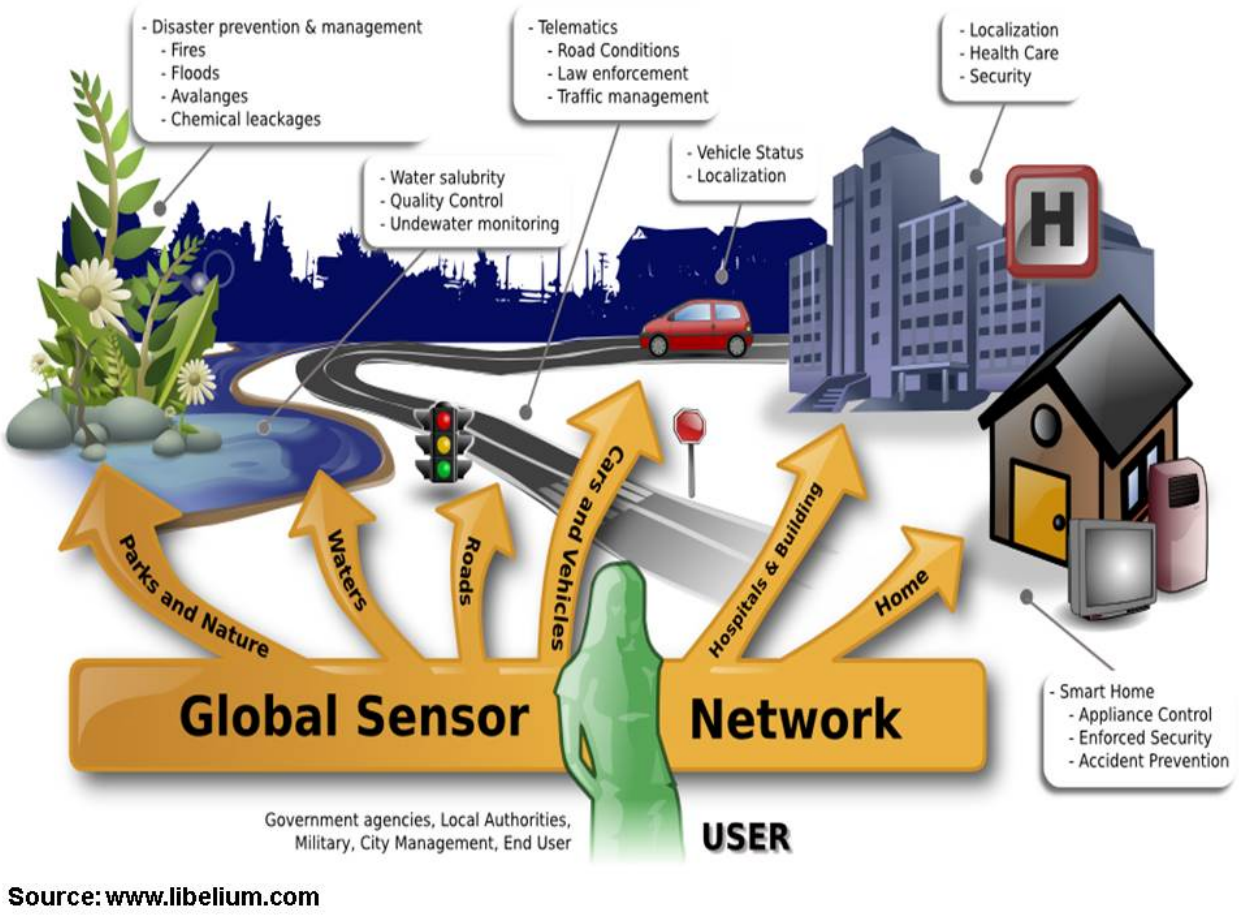


Figure 2.1: The Internet of Things Networks [1]

2.3 IoT Networks

A majority of wireless IoT devices are manufactured to operate in the ISM (Industrial Scientific and Medical) radio frequency band of 2.4GHz [34]. While IEEE 802.15.4, ZigBee, 6LoWPAN and WirelessHART form private LPWAN self-organizing networks, the explosion of the IoT business has spawned IoT service providers. These may include public IoT-specific networks, or cellular providers llocating a portion of their spectrum for IoT devices to communicate directly using the same 3G, 4G and 5G spectrum used by cellphones. In this section we will overview popular public IoT service providers, and the mobile operators IoT networks. In addition, new business models and protocols have evolved to support public

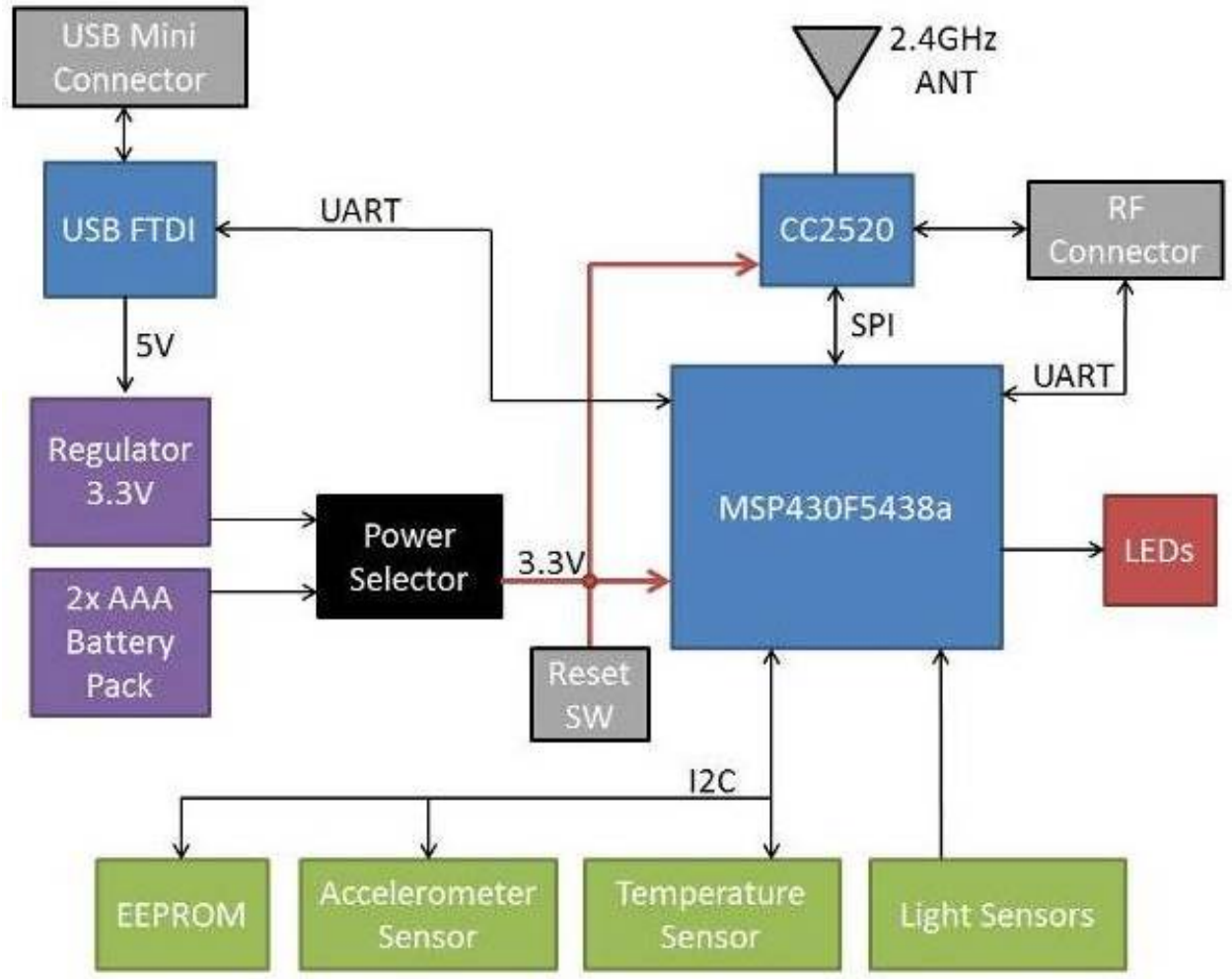


Figure 2.2: Example of an IoT node. Source: Texas Instruments

WANs over different frequencies, with different technologies, with companies managing these networks - networks such as SigFox, LoRA and Ingenu [42].

2.3.1 Sigfox

Sigfox is the largest and first company to offer public IoT services. Sigfox partners with mobile operators around the world (present in 45 countries) to co-locate equipment and share towers. The network operates in the ultra-narrowband frequency (UNB), in the 868 MHz (European) or 915 MHz (US) bands, using D-BPSK modulation. The uplink bitrates

are 100bps and 600bps respectively. The protocol uses a non-standard PDU with different uplink and sizes. The uplink (29-byte total) frame format contains a 12-byte payload, a 10-byte header and a 7-byte footer. The downlink contains an 8-byte payload, a 17-byte header and a 3-byte footer [43]. 4-byte global device ID are included in the headers and keys are pre-shared. The service is well adapted for very low and infrequent data rates (300bps average, 12-byte max and limited to 140 messages a day). Specialized low-cost IoT nodes and modules are required to support the service and are available through a hardware partner network comprising of TI, Silicon Labs, Avnet Silica and many others.

2.3.2 Ingenu

Ingenu offers an IoT public network for monitoring oil, gas, and critical infrastructure, and other applications. The service operates on a 2.4GHz-based Random Phase Multiple Access (RPMA) proprietary protocol on the uplink, and CDMA protocol (Code Division Multiple Access) on the downlink. The network is well suited for operating in harsh environments with a higher power, a smaller coverage and higher throughput, than its peers [44] [29]. More recently, Ingenu is the founding member of the IEEE 802.15.4k-2013 [45] body to standardize the protocol and outsource the development and manufacture of its custom hardware. The frame format is unknown given the proprietary nature of the protocol, but the standardized version of the protocol follows the frame structure of IEEE 802.15.4-2011, with support for alternate PHYs - LECIM DSSS (Low Energy Critical Infrastructure Monitoring Direct Sequence Spread Spectrum) for the uplink and LECIM FSK (Frequency Shift Keying) for the downlink. The IEEE 802.15.4-2011 MAC sublayer is described in Section 2.4.2. The protocol uses higher power for increasing range, throughput, signal strength and supporting great numbers of endpoints. Other features include 2 downlink channels for a BCH (broadcast channel) and dedicated (DCH)

2.3.3 LoRa

LoRa (for Long Range) is a proprietary PHY layer technology using Chirp Spread Spectrum (CSS - in the 868MHz, 915MHz and 433MHz (Asia) bands), for transmission in IoT networks [46]. The end-to-end architecture of the wide area network comprising of base stations, devices, gateways and applications is termed LoRaWAN. The LoRa PHY layer is

developed by Semtech (patented technology and makes the chipset). LoRaWAN refers to the MAC layer, an open standard developed by Actily, Microchip and IBM. The network components are provided, and used, by a large global ecosystem called the LoRaWAN Alliance^{TM}. LoRa provides a long range operation for IoT devices, but trades off with fixed, lower bandwidth, IoT communications. A study performed by [47] provides an overview of LoraWAN performance characteristics. The stated throughput is between 0.3kbps to 27kbps, depending on many factors (spreading factor - SF, modulation technique - Chirp vs. FSK). The MAC frame payload is variable depending on the modulation rate and the operational frequency (eg. EU: 52-223 bytes, US: 12-243). Typical operation ranges are between 3-5km in urban areas and >10km in rural areas. Bidirectional, encrypted communications are supported across three devices classes. Devices are connected to LoRaWAN gateways which further connected to application servers or the cloud. Class A devices are the most power efficient and mostly used for monitoring (downlink is only possible after uplink transmission occurs, using simple ALOHA protocol). Class B devices are synchronized via beacons and can have additional scheduled downlink, without prior uplink. Class C devices can downlink anytime unless they are transmitting. Power requirements for each device class are successively higher [48] [34].

2.3.4 Narrowband for the Internet of Things (NB-IoT) and LTE-M (Long Term Evolution-Machine)

NB-IoT and LTE-M are LPWAN services offered by cellular operators for IoT. The specifications are defined by the 3GPP (3rd Generation Partnership Project) Release 13. Both of these technologies offer different capabilities and have seen global deployments [49], and both operate in licensed cellular operator bands, and tend to have lower interference, compared to unlicensed bands. The objective is to reuse spectrum and maximize spectrum utilization and cellular infrastructure (such as antennae, base stations, towers and data centers). NB-IoT is a new specification (meaning new 3GPP devices) operating in 180KHz bands for both uplink and downlink. The bandwidth can fit into a 200KHz GSM carrier (stand-alone), and can also be deployed within the LTE carrier as a Physical resource block, or within the guard bands of the LTE carrier. The operating frequency is in the licensed 700-900MHz. The downlink transmission is OFDMA-based (Orthogonal Frequency Division Multiple Access). The up-

link can be either be a single tone transmission or two-tone transmission using SC-FDMA (Single Carrier-Frequency Division Multiple Access). Expected coverage area is upto 15km, with data rates ranging from 20-65kbps, and support for 52,000 end-points per cell [50].

LTE-M is a higher power, faster throughput technology that works for both fixed and mobile IoT nodes, that support higher bandwidth capabilities. LTE-M is also referred to as eMTC, Category-M1 or Cat-M1. Data rates can go up to 375kbps, which is the highest bandwidth among LPWAN technologies. LTE-M operates with a 1.4MHz shared bandwidth and uses specialized algorithms for power management (Power Saving Mode - PSM and extended Discontinuous Reception - eDRx). LTE-M is compatible with existing cellular deployments and can be deployed without additional infrastructure, operating in the licensed 700-900MHz band with uplink using SC-FDMA and downlink using OFDMA with 16 QAM (Quadrature Amplitude Modulation) and coverage upto 10km [51]. LTE-M has better coverage and signal acquisition in dense urban areas and within buildings. In the best case scenarios, LTE-M has been estimated to support about 10^6 IoT nodes per cell sector [52]. Table 2.1 displays the key characteristics of the above-mentioned public IoT networks.

Table 2.1: A Comparison of Widely Deployed Public IoT Networks [34] [29] [53]

Features of Public IoT Communication Networks				
Network	*Frequency*	*Throughput*	*Range*	*Other*
Sigfox	868 (EU), 915(US) MHz	UL: upto 300bps, DL: 8bps	Urban: 3-10km, Rural: 10-50km	upto 10^6 devices/-cell; star topology
Ingenu	2.4GHz	UL:78-624 kbps, DL: 19-156 kbps	Rural: 5-15km, Urban: 1-3km	10^6 devices/cell, star and tree topology
LoRA	433/868 (EU), 915 (US), 430 (Asia) MHz	UL/DL: 0.3-50 kbps	Rural: upto 15km, Urban: 2-5km	gateway dependent, star of stars topology
NB-IoT	600, 700, 850, 1700 (US); 800, 900, 1800 (EU) MHz	UL:204.8 kbps, DL:234.7 kbps	Rural: upto 35km, Urban:	upto 55k devices/-cell, star topology

The public IoT networks mentioned in this section are the most well-known deployments.

Other alliances that have deployments are Weightless, Dash7, QOWISIO and Telensa [29]. In Section 2.4 we present and analyze the communication protocols commonly used by IoT nodes for ad-hoc, mesh, wireless sensor networks that are localized.

2.4 IoT Communication Protocols

We encounter the Internet of Things (IoT) regularly in our daily lives including in financial transactions such as automated tolling, home and manufacturing automation and supply chain management. The significant and measurable benefits provided by networked smart things, combined with the continued improvements in device performance and longevity and reductions in device cost, will ensure the continued deployment of new IoT systems across a broad range of applications and in a diverse set of environments. The pervasiveness of the Internet of Things, both in our surrounding environments and on our persons, combined with our reliance upon many of these systems for automated control, personal monitoring and financial transactions, necessitates a strong integration of security mechanisms within all IoT devices to protect us from intentional and accidental harm.

A number of standardized communication protocols that include security mechanisms have been developed for, and adopted by, IoT enabled applications. Three popular protocols are 6LowPAN, ZigBee and WirelessHart. Each of these protocols is designed to operate on top of the IEEE 802.15.4 Low Rate Wireless Personal Area Network (LR-WPAN) protocol. IEEE 802.15.4 defines the Physical (PHY) layer (how bits are sent over the air) and the Data Link (Link) layer (how data is sent directly from one node to another node) of the communication stack. Routing, network management and applications are the responsibility of the protocols built on top of IEEE 802.15.4 [41].

IEEE 802.15.4-based IoT devices are typically small, resource constrained devices that may be either mobile or mounted to a fixed location. Additionally, these devices are typically powered by a single, small non-rechargeable battery that must power the device for several years. This often results in the devices utilizing intermittent 'sleep' functionality where the device is neither transmitting nor receiving in order to maximize the lifetime of the limited battery source. The limited resources and power constraints impact the security capabilities of these devices [45].

The 6LowPAN, ZigBee, and WirelessHART protocols were designed to operate within the resource limitations of a typical IEEE 802.15.4 based IoT device. Consequently, their standard security mechanisms have limitations that may prevent their usability in a range of new applications that are integrating smart objects. In the rest of this chapter, we evaluate the capabilities of the standard security mechanisms of the 6LowPAN, ZigBee and WirelessHART protocols. Our evaluation focuses on the ability of these mechanisms to prevent or mitigate a range of attacks that are expected to arise in a ubiquitous Internet of Things world. While these protocols have an array of defined mechanisms that may be used to thwart basic attacks such as eavesdropping and message modification, we find that additional practical attacks, such as node capture and resource exhaustion, necessitate the inclusion of security mechanisms beyond what is defined in these protocols. In addition, we compare the defined security mechanisms of these three protocols to one another, and we find that WirelessHART has a set of security mechanisms that mitigates the broadest array of potential attacks on IoT smart objects.

2.4.1 Security Overview

There are several fundamental security services that should be provided by even a simple IoT communication protocol: *access control, message integrity, message confidentiality, and replay protection.* These security services provide a basic level of protection and, ideally, are provided at each layer in the communication protocol stack. Higher layer protocols should provide additional security services, such as *routing integrity* and *routing assurance* which should be provided at the Network layer.

Access control services limit the impact of communications from unauthorized devices. At the link layer, access control services should prevent an authorized device from responding to communications from or communicating with unauthorized devices. Access control at the Link layer is the first, and lowest cost, layer of defense preventing unauthorized devices from accessing the network. Access control at the Network and higher layers prevents a device from accessing or using a resource for which it is not authorized. Access control at the Network and higher layers can be costly to implement but may protect individual resources and functions contained within a smart object or within the network itself.

A *message integrity code* (MIC)[1] may be included with each message sent in order to provide both message authentication and message integrity. A MIC is a cryptographically secure digest of the message, or a portion thereof, that is typically computed using a secure hash function such as SHA-256 or SHA-3. Computing a MIC requires both the sender and the receiver to share a secret key that is used in computing the digest. Consequently, if an adversary alters a message from an authorized sender or injects an unauthorized message, the adversary will not be able to generate a correct MIC without knowledge of the secret key. A MIC may be used to protect the integrity of a complete message that is communicated using multiple packets and at each level of encapsulation within a packet. In addition to providing for message integrity, a MIC is perhaps the simplest approach to providing access control at the link layer.

Message confidentiality services assure that the transmitted message is disclosed to the intended recipient(s) only. Message confidentiality is typically achieved by encrypting the data portion of a packet. Within a communication layer, such as the Link layer or Network layer, any source and destination addresses for that layer typically are not encrypted. Symmetric key ciphers, such as AES, are used most often for message confidentiality. This requires that both the sender and the receiver share a secret key that is used in encrypting and decrypting the message. However, a shared secret requires symmetric key management services to be provided by the higher layers of the communication stack even though message confidentiality is best provided at least at the Link layer.

Replay protection (also known as sequential freshness) services ensure that duplicate messages between authenticated parties are detected and dropped. A replay attack is simply the intentional retransmission of valid packets in an attempt to either gain access to a resource or deny that resource to others. One of the difficulties in detecting replay attacks is that repeated messages may arrive at a receiving node through normal operation due to the manner in which packets are routed through a network. A simple mechanism that can be used for replay protection is an incrementing packet counter. The monotonically increasing nature of the counter ensures that messages with lower packet counter values than the next

[1]The IEEE 802.15.4 specification refers to a *message authentication code* as a MIC to differentiate it from media access control. In this chapter, we follow the IEEE 802.15.4 convention of referring to a message authentication code as a message integrity code.

expected packet counter value are rejected, thus limiting the effects of repeated messages and of replay attacks [39].

These four basic security services mitigate a range of fundamental attacks including eavesdropping, packet replay, packet injection, packet modification and simple resource exhaustion attacks. Advanced attacks, such as jamming, node capture and advanced resource exhaustion attacks, and higher layer attacks, such as selective forwarding, flooding and desynchronization attacks, are not completely mitigated by these basic security services.

The main security challenges presented by Wireless Personal Area Networks (WPANs) are low computational resources, small memory resources, limited physical protections and limited power on the WPAN connected smart objects. Node capture is a practical attack in most WPAN deployments due to direct physical access to the devices. Node capture refers to an adversary directly accessing the device, either through physical access or electronic access, allowing the adversary to extract keys, inject messages, operate as an authenticated node and remove nodes from the network. As a result, security services for WPAN protocols such as 6LowPAN, ZigBee and WirelessHART must be extremely efficient to run on nodes that can preserve battery life for years while protecting against authorized nodes that may be captured by an adversary. In addition, the consequences of node capture can be reduced by enforcing certain security requirements such as erasing secure key information when the node is disassociated from a network [54].

The limited resources of WPAN devices, particularly the limited power supply that is common in these devices, requires the communication protocols to protect the resources. Power depletion attacks, where a device is forced to utilize all of its available energy to manage malicious communications or perform activities requisted by an adversary, require explicit power management services in order to limit the consequences of the attack. Power depletion attacks have created specific security guidelines that are normally not considered in standard networks [55].

2.4.2 IEEE 802.15.4 Security

IEEE 802.15.4 security is provided by the MAC sublayer on incoming and outgoing frames via services supporting [39]:

- Data Confidentiality (transmitted information is encrypted)
- Data Authenticity (transmitted information is not modified)
- Replay protection (Sequential Freshness, transmitted information is not replayed)

The MAC PIB (PAN Information Base) maintains a device table that allows authenticated devices to communicate and set the security level between them. Security is requested by the upper layers [56] [10]. The keys and their management are provided by the high layer protocols (6LoWPAN, ZigBee and WirelessHART), and per frame MAC layer encryption, of varying authenticity levels may be requested (to minimize security overhead where required). If not requested, or keys not provided, then the default is no security at the MAC layer. The encryption scheme supported by IEEE 802.15.4 is AES-CCM* [39] [10] [57].

IEEE 802.15.4-based networks are vulnerable to node capture, power drain (resource exhaustion), replay attacks and attacks that forge the unencrypted ACK frame. Thus, the security architecture must establish and maintain trust relationships, which are defined at the higher layers.

In the IEEE 802.15.4 Link layer, replay protection is provided with all security levels, except security level 0 (no security). Frame protection uses either a link key (shared between peer-peer devices) or a group key (shared among a group of devices). When a group key is used for peer-to-peer communication, protection is only against outsider devices and not against potential malicious devices in the key-sharing group [39].

2.4.3 Protocol Security

In this section, we detail the 6LoWPAN, Zigbee and WirelessHART protocols that utilize IEEE 802.15.4. These protocols are designed for personal area networks and are well suited for IoT applications. 6LoWPAN maps the IPv6 protocol to the IEEE 802.15.4 header. This enables IPv6 addressing directly to each node. Zigbee and WirelessHART have been deployed in industrial systems with specific needs for networking and small packet size.

2.4.3.1 6LoWPAN

6LoWPAN is defined in a collection of IETF standards that define the use of IPv6 for low power WPANs (IETF RFC 4919 [40], RFC 6282 [58], RFC 6775 and RFC 6550 [59]). It uses

the MAC and PHY sublayers of IEEE 802.15.4 (Figure 2.3). The large address space of IPv6, and the widespread use of IP allow 6LoWPAN to make smart objects directly addressable to an IP network. 6loWPAN has deployments in automation, control and energy sectors [60] [2]. 6LoWPAN uses the IPSec security architecture. 6LoWPAN security requirements include:

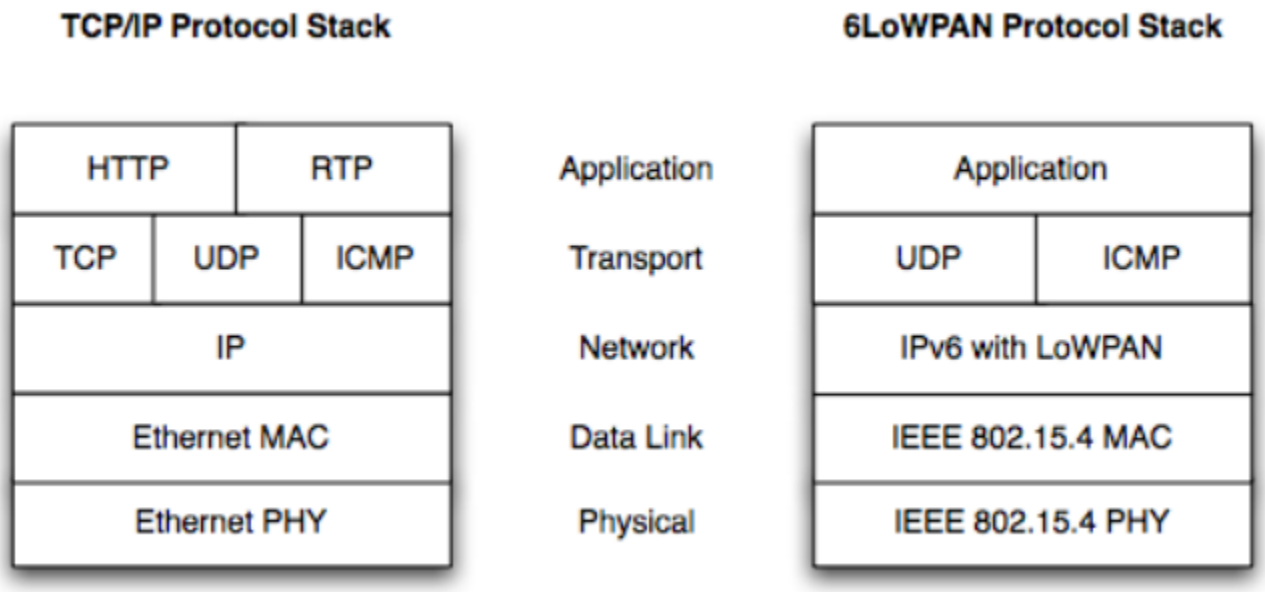


Figure 2.3: Comparison of the TCP/IP and the 6LoWPAN Protocol Stacks [2].

- Information Security: Confidentiality, Integrity, Authentication, sequential freshness
- Operational Security: Availability, Robustness, Resiliency, Resistance, Energy Efficiency, Assurance (operate at multilevel security modes).

Park et al. [54] refer to a set of security considerations for 6LoWPAN. They include efficient adaptation of network layer security for 6LoWPAN including authentication and key management. However, the threats to the IP network remain the same, or increase e.g., physical access to smaller devices is easier, and effective defence against node capture attack is minimal. IPv6 network layer security (IPSec) is resource-intensive to small devices and cannot be directly applied to 6LoWPAN. Internet key exchange (IKEv2) messaging (RFC5996) has a high signaling cost for low power, low data rate devices. Efficient key management and

distribution algorithms will have to be defined for 6LoWPAN. Standard IP network threats remain for 6LoWPAN such as DoS, intrusion, sinkhole, replay and insecure routing attacks. To mitigate IPSec vulnerabilities, a combination of Application Level Security SSL, with link layer security IEEE 802.15.4 MAC is recommended. IPSec is not mandated for the network layer.

Network Layer attacks necessitate the following security requirements for 6LoWPAN:

- End-to-end security (use AES-CCM* IEEE 802.15.4 security profile)
- Node Authentication (node joining a network)
- Key establishment, initial key transfer (using out-of-band methods) and key revocation schemes. The ZigBee key management scheme is recommended for 6LoWPAN. Nodes in active state must have their key information. Nodes in passive state (sleep mode) should not have their key information. Key must be re-instated once in active state.

The above security mechanisms are not effective measures against node capture and resource exhaustion attacks.

2.4.3.2 ZigBee

Zigbee is a set of application protocols, based on the IEEE 802.15.4 PHY and MAC sublayer. ZigBee application profiles form energy-efficient, low data-rate and self-configuring mesh networks of up to 2^{16} devices, using Zigbee devices which are RFD (Reduced Function Devices) or FFD (Full Function Devices). The Zigbee alliance hosts the multiple Zigbee specifications, standards, member companies and Zigbee device certifications (http://www.zigbee.org). ZigBee has been deployed in a wide variety of consumer electronics, industrial, control, lighting, home, telecom, healthcare and energy segments. The ZigBee-2007 standard has two feature sets: ZigBee and ZigBee PRO. ZigBee PRO has larger node support and additional security (supports high security mode and master key) [26]. A newer version of Zigbee, known as Zigbee IP is based on 6LoWPAN, and associated IETF protocols. Security is provided by TLS1.2 protocol, with support for public key infrastructure using standard X.509 v3 certificates and ECC-256 cipher suite. Zigbee Smart Energy Profile 2 is the application profile supported by Zigbee IP.

The security architecture of ZigBee has certain architectural guidelines, sublayer interfaces, key definitions and usage models. The security services for the ZigBee protocol is provided by the Security Service Provider, and specified in the Security Services Specification within the ZigBee standard. Services include key establishment, key transport, frame protection and device management.

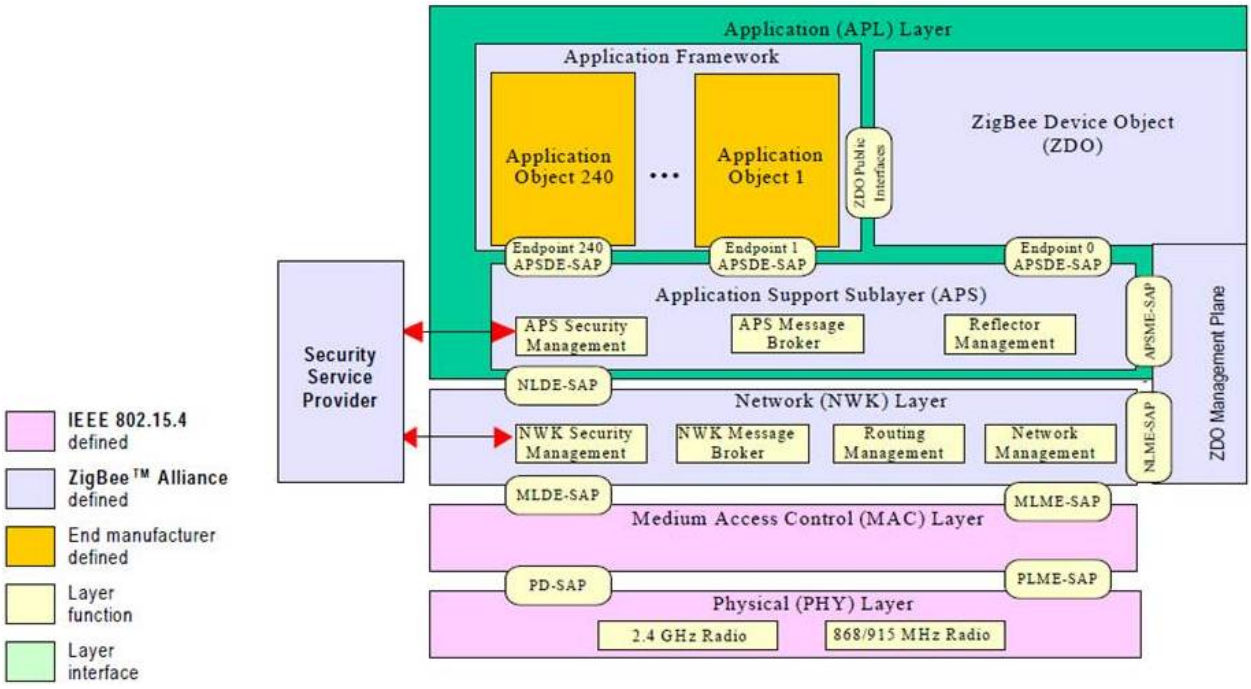


Figure 2.4: ZigBee Protocol Stack Architecture [3](www.zigbee.org)

Figure 2.4 shows the ZigBee Protocol Stack Architecture. The Security Service Provider outlines security mechanisms for the Network Layer (NWK) and the Application Support Sublayer (APS). The ZigBee Device Object (ZDO) manages the security configuration and security policies of a device. The security architecture defines security functionalities based on an Open Trust Model which assumes: symmetric keys are safe; applications on a device are trusted; APS, NWK and MAC sublayers are accessible to applications and are trusted.

ZigBee outlines architectural design choices that provide for operational security - preventing a malicious device from utilizing the network. The ZigBee standard specifies:

- *The layer that originates the frame is responsible for initially securing it.*
- *If protection from theft of service is required, then NWK layer security shall be used for all frames.*
- *Security can be based on the reuse of keys by each layer.*
- *End-to-end security is enabled, based on a shared key between only two devices.*
- *The security level used by all devices in a given network, and by all layers of a device shall be the same.*

Finally, the ZigBee standard outlines the security keys, the Trust Center (TC) design and the APS sublayer and NWK layer security.

Security keys defined by the ZigBee standard are AES-128-CCM (or AES-CCM*, derived from IEEE 802.15.4) symmetric encryption keys of three types - Master key (MK), the Link key (LK), and Network keys (NK). MK is required to join the network. MK is used to generate the LK. LK provides the highest level of end-to-end unicast encryption between two ZigBee peer devices (APL layer for key transport, authentication, in High Security (HS) Mode). The network key is shared between all devices, encrypting all network broadcast communications (authentication, frame security in Standard Security (SS)mode). Keys can be factory set or be distributed via the *trust center* (residing in the network coordinator). The NK is used by MAC, NWK and APL sublayers. The MK and the LK are used only by the APS sublayer. Key acquisition is via

- Key Transport: Key distributed to the device by the Trust Center
- Key Establishment: pairwise LK distribution between two devices. A pre-shared MK is required in both devices
- Pre-Installation: Key provided to device before joining network (usually MK, but all key types can be pre-installed, but are vulnerable to attack)

The Trust Center is an application in a secure ZigBee network (typically within the ZigBee Network Coordinator) that distributes keys for the purpose of network and end-to-end application configuration management. The Trust Center is trusted by all devices in the

network. There is only one TC per network. Each device is associated with only one TC. TC operates in a commercial High Security (HS) mode and a Residential Standard Security (SS) mode. In the HS mode, the TC maintains a centralized repository for NKs, LKs and MKs. The TC establishes and maintains the list of all devices, associated keys and freshness counters. TC also enforces NK renewal and network access control policies. In the SS mode, the TC maintains the device list and associated MK, LK. TC maintains the NK and network access control policies [61]. The functions provided by the TC are:

- Trust Manager: Identity Management, Authentication of a device sending a network join request.
- Network Manager: Maintain and distribute NKs to the network devices
- Configuration Manager: Establish peer-to-peer, end-to-end, security between network devices.

ZigBee Network Layer Security ensures the network frame is secured, and appropriately interfaces to the APL Layer. The ZigBee APS Sublayer Security provides Key Establishment, Key Transport, Device Update, Device Remove, Request Key, Switch Key, Entity Authentication and Permission Configuration Table services [26].

2.4.3.3 WirelessHART

WirelessHART is a robust, time synchronized, self-organizing, self-healing, mesh networking protocol, using the IEEE 802.15.4 PHY layer. The TDMA-based MAC sublayer is defined in the WirelessHART standard using TSMP (time synchronized mesh protocol) technology. WirelessHART is primarily used for process control and measurement environments, because of its backward compatibility to the widely-deployed, industrial HART protocol. Industrial control environments require deterministic timing and often have harsh radio interference. WirelessHART is the IEC 62591 standard, and operates on the 2.4GHz ISM band, with 16 channels.

The WirelessHART peer-peer mesh network consists of devices, a Network Manager (NM) and a Security Manager (SM). Devices are of four types and ALL are required to have routing capabilities. [62]

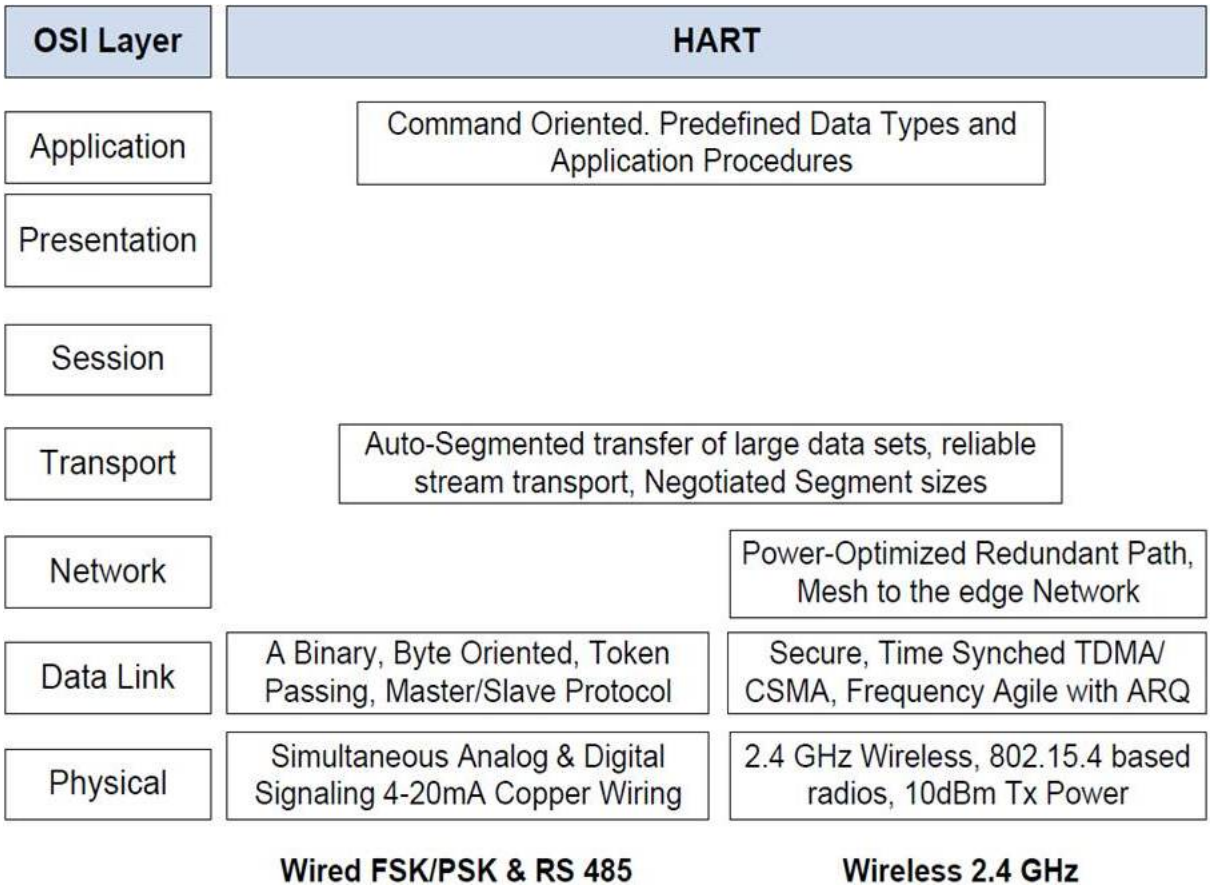


Figure 2.5: WirelessHART, HART Protocol Stack [4]

- Field Devices: connected to sensors and actuators
- Router Devices: for communications
- Handheld Devices: for operations and maintainence
- Adapter Devices: for legacy HART or non-wireless devices to be connected to the mesh network

The Gateway Device is 1+1 redundant, and connects the network of devices to the plant automation system. The Network Manager generates optimized routes and communication schedules. The Security Manager (not mandated or defined by the standard), provides security keys generation and management (storage, renewal, revocation).

Physical layer security on the radio layer consists of Frequency Hopping Spread Spectrum (FHSS) across 16 channels (logical channels 11-26). Clear Channel Assessment (CCA) is optional to determine channel efficiency and configure transmit power levels. *Channel Blacklisting* is a mechanism to disallow the use of rogue channels (channels with interference). Active Channel lists are maintained by each device to ensure the network understands which channels are in use.

WirelessHART security services are provided by the MAC sublayer and the network layer.

The WirelessHART mesh network provides Graph routing and Source routing mechanisms to bypass link failure, interference and inoperative devices - any or all of which could be a result of a malicious attack. Each device is a router and connects to two other devices for path diversity.

Data integrity, frame encryption and message authentication are provided by AES-128-CCM [57] and used by both receiver and sender. For end-to-end, per-hop and peer-to-peer communications, the network layer and the data link layer uses various key types [4] [63].

Security keys are not well-defined in the WirelessHART standard. Eight different keys can be used for payload encryption/decryption and MIC generation over the Network and Link layer PDUs [12]:

- The *Network Key* computes the MIC over the link layer payload, and is used for device authentication.
- The *Join Key* secures network frame payload when the device joins a network, and computes the MIC. It is used by the NM to renew unicast session keys.
- The *Unicast-Gateway Key* is used to encrypt network layer payload between the Gateway and devices. It is used to compute the MIC at the network layer.
- The *Unicast-NM Key* is used to secure the session between Network Manager and devices. It is also used for renewing the Join Key, post device authentication.
- The *Broadcast-Gateway Key* is used to secure broadcast messages between Gateway and field devices.

- The *Broadcast-NM Key* is used to secure Network Manager broadcasts to the Gateway and to field devices. It is also used for renewing the Network Key.

- The *Handheld Key* is used for a secure session between a field device and a handheld device. The key is provided by the Network Manager to both devices after the Handheld device has been authenticated.

- The *Well-known Key* is a pre-assigned (factory flashed) network key used to authenticate new devices joining the network (always 0x 777 772E 6861 7274 636F 6D6D 2E6F 7267).

WirelessHART is designed to be a robust, reliable protocol providing greater than 99.73% availability, but there are limitations with the security architecture. The Security Manager specifications and architecture are not defined in the WirelessHART standard. For example, the interface and messaging between the Network Manager and the Security Manager is not defined. Along with that, key management definitions are incomplete (only key distribution by the Network Manager is defined). This may lead to a compliant, but insecure, implementation. Public key cryptography remains unsupported. Therefore the standard provides Confidentiality, Integrity, Authentication and Availability, but no Authorization, Non-repudiation and Access Control. This may prevent containment of a malicious node within effective limits. Secure multicast between field devices is not supported, but secure multicast between Gateway to all field devices, and Network Manager to all field devices is defined.

There is an exhaustive threat analysis for WirelessHART by [12]. Threats such as interference, jamming, Sybil, Traffic Analysis, Denial of Service, De-synchronization, Wormhole, Tampering, Eavesdropping, Selective Forwarding Attack, Resource Exhaustion, Spooling and Collisions. The more sophisticated threats such as wormhole, de-synchronization, traffic analysis, spoofing, resource exhaustion and jamming require additional security criteria. To mitigate node capture threats, it is suggested that the field device reformat memory contents when it disassociates from the network. Resource exhaustion may occur by frequent link scheduling and routing. The 10ms active time slot in WirelessHART limits any continuous resource drain.

2.4.3.4 Bluetooth Low Energy (BLE)

In this section we discuss the security architecture and features of Bluetooth Specification Version 4.2 [64]. The security components of BLE in relation to the BLE protocol stack is shown in Fig 2.6. BLE has two primary components, the Controller (PHY and Link), and the Host (upper layers) Security services are provided at both the Host and the Controller [65]. In the Host part, security services are provided by the Generic Access Profile (GAP) and the Security Manager (SM) via a dedicated channel using the Security Management Protocol (SMP).

Host	GAP	LE Security Module 1	Level 1: No Security Level 2: UNAUTH pairing + ENC Level 3: AUTH pairing + ENC Level 4: AUTH LE SC pairing + ENC
		LE Security Module 2	Level 1: UNAUTH pairing + DS Level 2: AUTH pairing + DS
	SM	Security Keys	IRK (privacy), CSRK(DS), LTK (encryption): 128 bit EDIV, RAND: LTK identifier, 16/64 bit TK, STK: Pairing, Temp Keys, 128 bit
		Pairing	Just Works, Numeric Comparison Passkey Entry, Out-of-Band
Controller	Link PHY	AES-CCM encryption with 13 byte nonce	

Figure 2.6: Bluetooth Low Energy Security Stack

The BLE protocol security is based on five standard security services: device authentication, encryption, message integrity, pairing and bonding.

Authentication is a method to prove that an entity is who they say they are. Authentication is performed between devices, for service requests and for messages. Authentication between two devices in BLE is performed after a connection is established (LE Security Mode 1).

A *message integrity code* (MIC) may be included with each frame to provide both message authentication and message integrity. Using a MIC requires both the sender and the receiver

to share a secret key that is used in computing the MIC.

Message confidentiality services assure that the transmitted message is disclosed only to the intended recipient(s). Message confidentiality is typically achieved by encrypting the payload portion of a frame. The header information is not encrypted. At the Controller, Link layer security in BLE provides confidentiality and integrity via AES-CCM. Link layer connections are of two types: encrypted and authenticated or unencrypted and unauthenticated. In the former, Data Channel PDUs are authenticated with a 4-byte MIC (the MIC is computed over the payload and first byte of the header). The encryption is done over the Data Channel PDU payload and the MIC. Advertising Channel PDUs are not encrypted or authenticated and this provides opportunities for a range of attacks like inference attacks, eavesdropping, message modification and packet injection with incorrect control sequences. Data signing is used for transferring authenticated data over an unencrypted connected used for fast data transfers and fast connections.

Pairing and Bonding are related to key management. Pairing is the generation of a shared secret between two entities through association models. Bonding is the storage and subsequent use of the keys generated during pairing. Key generation for BLE is done in the Host, per device, for ease of upgrades. Prior non-LE Bluetooth versions generated the key in the Controller, which required a Controller upgrade for a change in the key generation algorithm. BLE uses the following keys: LTK (Long Term Key) for encryption; IRK (Identity Resolving Key); CSRK (Connection Signature Resolving Key) for data signing; EDIV (Encrypted Diversifier) and RAND (Random Number) for identifying the LTK during pairing.

BLE uses four association models for pairing: Just Works, Numeric Comparison, Out of Band and Passkey Entry. Association models are deployed based on the IO capability of the device. BLE Just works and Passkey Entry do not provide passive eavesdropping protection. BLE pairing uses the following two keys: TK (Temporary Key) and STK (Short Term Key). The TK is used to generate the STK, which is used to encrypt the connection after device pairing is complete.

In addition, BLE also provides replay protection and privacy services. *Replay protection* services ensure that duplicate messages between authenticated parties are detected and

dropped. A replay attack is simply the intentional retransmission of valid packets in an attempt to either gain access to a resource or deny that resource to others. Replay protection is provided via the SignCounter field for authenticated data over an unencrypted channel. It is also provided during the pairing procedure of Numeric Comparison, Just Works and LE Secure Connections. The BLE *privacy* feature can be activated after a node has joined the network. The privacy feature changes the BLE device address (BD_ADDR) frequently to avoid being tracked. That random address is generated and resolved using the IRK, and is mapped to a lookup table of private addresses.

2.4.4 Security Analysis

We have reviewed the security analysis for IEEE 802.15.4, 6LoWPAN, ZigBee and WirelessHART. We now discuss the threats applicable to each protocol and possible mitigation with suggestions.

Starting with IEEE 802.15.4-2011, we note that the standard has a separate security section. The main threats to this protocol are NO encrypted ACK frames, NO timed frame counters and NULL security level. When the ACK frame is NOT encrypted, an intruder can intercept a MAC frame, forge an ACK frame with a sequence number, resulting in frame loss with no retransmission. Replay attacks send a large number of intercepted frames, with large counters. Valid frames with smaller counters are then rejected by the security mechanisms which do not evaluate based on time-stamps. Unless defined by the application there is no security set up by default for IEEE 802.15.4. This could result in insecure and compromised systems using IEEE 802.15.4

6LoWPAN is an IETF standard, but its adaptation to IEEE 802.15.4 WPANs currently includes IPv6 protocols that are ill-suited for WPANs. For example, IPSec provides network layer security. IKEv2, within IPSec, used for key management results in heavy computational and extra packets that reduce the efficacy of end nodes operating in a low-data-rate, low-power environment. Additionally key revocation and secure node joining authentication methods are undefined and currently insecure.

ZigBee is a well-defined protocol that addresses secure per-hop, end-to-end and peer-to-peer communications (can encrypt at three layers - MAC, NWK and APS layers). The

centralized ZigBee Trust Center (TC) that generates and updates keys for all devices within the network is a vulnerability. Additionally, when a node disassociates from the network, it still contains the network key (NK), and creates a vulnerability.

WirelessHART is the most reliable of the above protocols because of direct channel access to the node, based on a TDMA MAC. WirelessHART does not support public key cryptography. The Security Manager specifications are undefined in the standard and are susceptible to insecure implementation.

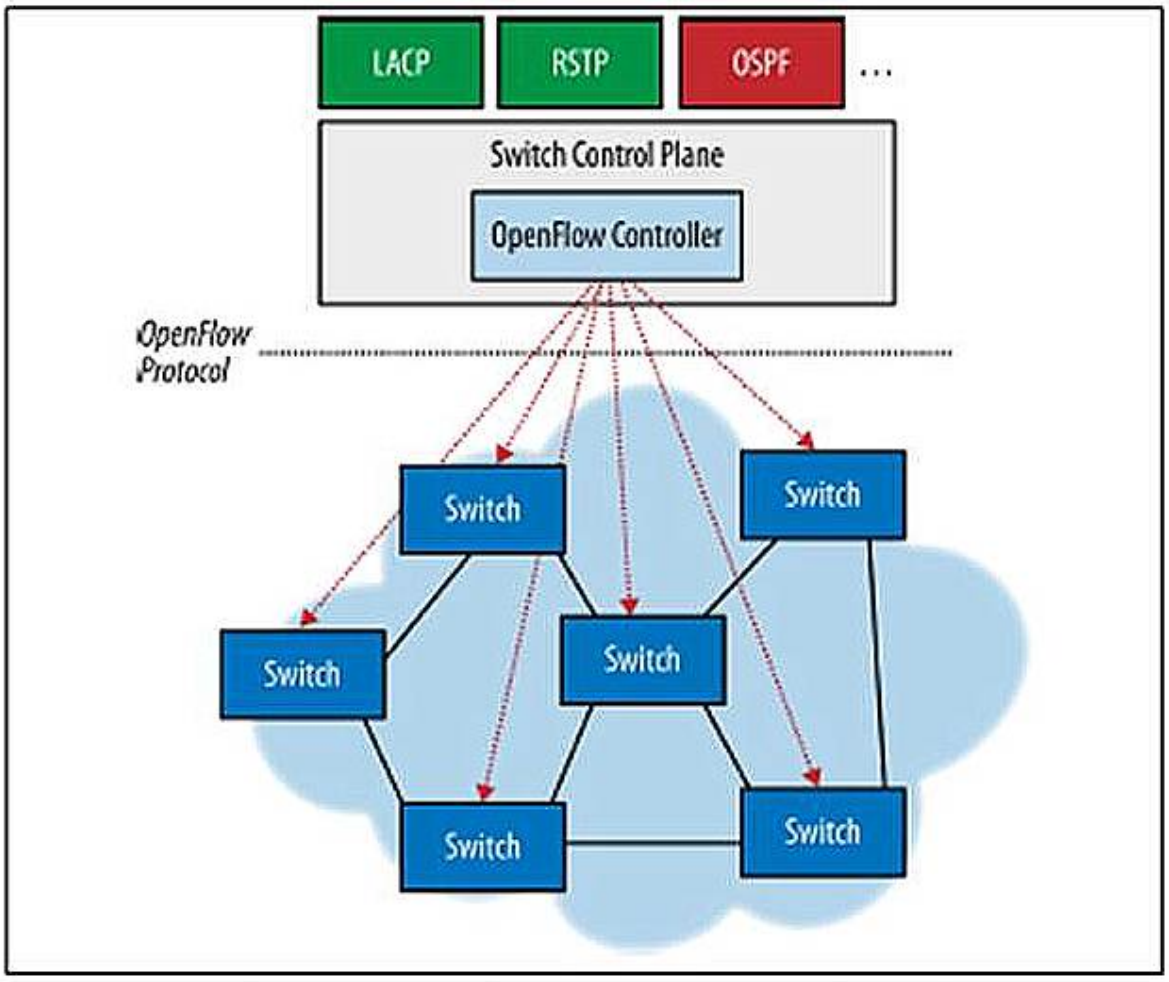
In all of the above protocols, node capture is a vulnerability. Node authentication and access control are security mechanisms to ensure malicious or "zombie" nodes (defined as good nodes captured and turned bad) do not join, or are revoked, within a network. Security mechanisms such as node memory formatting (erasing key information), after node capture, should be considered.

2.5 Software Defined Networks and IoT

Software Defined Networking (SDN) is a new paradigm in computer networks. The Internet has become the backbone of our information society. The Internet is managed all over the world and large portions of it are managed by Service Providers. As millions of users have subscribed to this universal service, the demands on Service Provider networks have grown exponentially with data, video and mobile traffic. Networks are scaling poorly. Service provider capital expenditure and operating expenses are rising. To add to this, over the years networking standards have been open, but implementations have been proprietary. Line speeds for routers and switches have increased, but signaling (control) and media (data) have been on the same physical equipment. The complexity of traffic and networks has given rise to 'middleboxes' for targeted applications such as firewalls, load-balancing, xml processing. The network complexity and architecture to carry data traffic, along with the thousands of standards and protocols being introduced is creating an Internet traffic jam [66].

A confluence of factors have come together to create a paradigm shift - cloud computing (the ability for a user to rent compute, storage and applications on-demand without physically acquiring the assets), Data Center evolution (for a more streamlined architecture), Mobile Content Access (universal mobility and data access globally) and Big Data (analyt-

ics for decision support and control) are driving Internet traffic. The search for a scalable, efficient, simpler network architecture has become essential.



Flow Entry 0		Flow Entry 1		...	Flow Entry F		...	Flow Entry M	
Header Fields	Inport 12 192.32.10.1, Port 1012	Header Fields	Inport * 209.*.*.*, Port *		Header Fields	Inport 2 192.32.20.1, Port 995		Header Fields	Inport 2 192.32.30.1, Port 995
Counters	val	Counters	val		Counters	val		Counters	val
Actions	val	Actions	val		Actions	val		Actions	val

Figure 2.7: SDN Architecture and OpenFlow V1.0 Flow Table [5] [6]

One way to scale large networks and make them cost effective via is via SDN. SDN separates the control plane (signaling) from the data plane (media). This results in an architecture that is scalable and very cost effective, as the routers (now called forwarding elements or switches) have minimal logic to forward data (and can be very simple computing elements, without complex, expensive logic). The forwarding decisions are based on simple Flow Tables that are downloaded to the switches by a centralized controller, which has a global view of the network. The controller communicates with the switches using a open, industry-defined, open-source protocol called OpenFlow. For the purposes of this paper, we will need to know the architecture of SDN and the format of communications of OpenFlow 1.0 (OF V1.0, later versions have followed). An example of this architecture, and OpenFlow 1.0 format is shown in Fig. 2.7. For the rest of this paper we need to understand the SDN architecture and the Flow Table format, which we will use on IoT networks in Section 2.5.1 for better efficiencies and reliability [6]. Further information about SDN in general, including a description of OpenFlow, can be found in [66].

2.5.1 SDN applied to IoT

Now that we have established the basics of SDN and wireless sensor networks in the Internet of Things, how exactly do we bridge the gap? While we can translate the components of IoT networks to the SDN paradigm rather simply, as in Fig. 2.8, this does not come for free. As with most issues regarding IoT, the main problem comes down to scale: how exactly do we take SDN and reduce it to a scale that can perform well on these tiny devices? While a formal protocol recommendation or proposal is officially a topic for future research, we will highlight a few of the potential solutions from Sensor OpenFlow, an SDN-for-IoT protocol based on the OpenFlow standard, described in [7].

The basic architecture of Sensor OpenFlow is shown in Fig. 2.9, and is very similar to that of OpenFlow except with the flow tables rewritten to better support addressing schemes of wireless sensor networks. In particular, it utilizes two ”classes” of addressing schemes:

- Class-1 defines an addressing scheme for 'compact network-unique addresses', e.g. tag-based unique identifiers. It is implemented via OpenFlow's OXM (OpenFlow Extensible Match) fields to introduce a number of customized match fields as depicted in Fig.

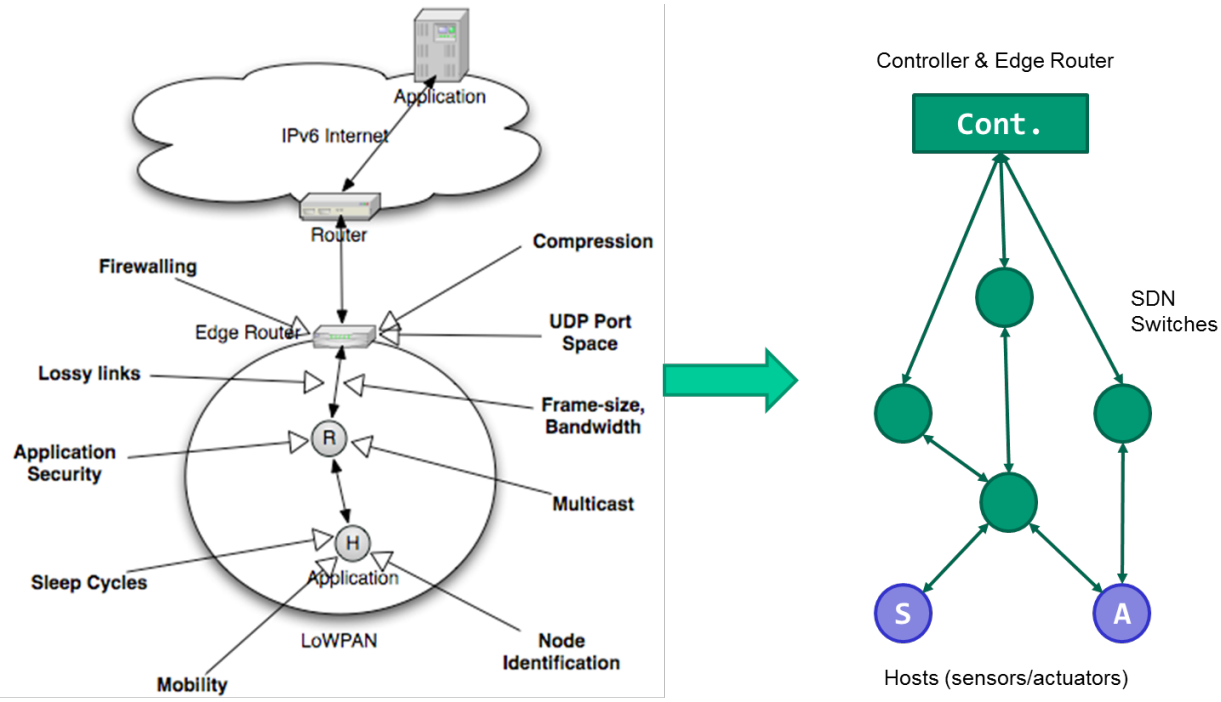


Figure 2.8: IoT deployment with 6LoWPAN [2]

2.10.

- Class-2 addresses are content-based addresses employing a new "concatenated attribute-value pairs (CAV)" match paradigm. This is a full extension to OpenFlow's 12-tuple match structure that allows packets to define a parameter, operator, and match value (e.g. "temperature <12") that allows one to route packets to destinations based on the actual content of the packet. This allows for some very basic "deep packet inspection," to a degree, supported natively as part of the SDN protocol.

2.5.1.1 Payload Size

The first notable benefit we get with SDN-based routing is payload size. Networking bandwidth is easily taken for granted these days, but such concerns become paramount when discussing the Internet of Things, as devices' memory and bandwidth limits are incredibly small. Of particular note is the fact that 802.15.4 devices have a maximum payload size

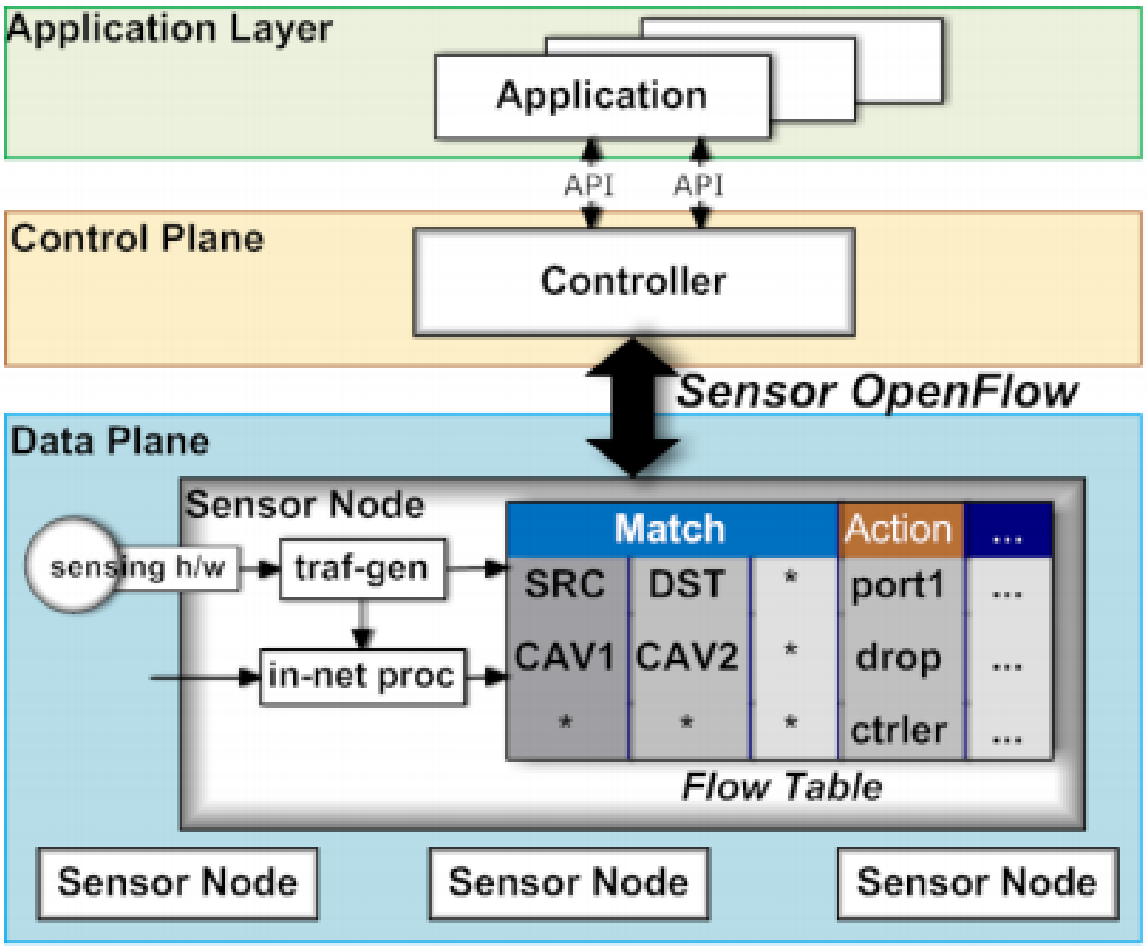
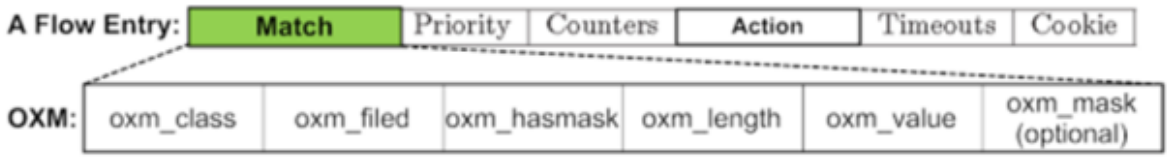


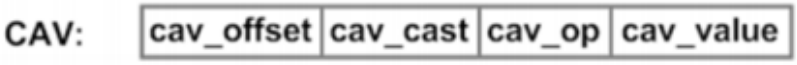
Figure 2.9: Sensor OpenFlow architecture [7]

(MTU) of 127 bytes [39], which is small enough that packet header size can become an issue. In order to reduce header size, we need to define an efficient addressing scheme for packets. The current standard, IPv6, is problematic as the headers take up more space than the actual payload, as is depicted in Fig. 2.11. In fact, the principal technology behind 6LoWPAN is that of header compression, which is what enables IPv6 to be usable on such small devices in the first place. However, this compression comes at a cost, as the devices themselves must spend significant processing time (on a very small chip) to reduce the headers to a usable state.



Explanation:

Name		Width	Usage
oxm_type	oxm_class	16	Match class: member class or reserved class
	oxm_field	7	Match field within the class
	oxm_hasmask	1	Set if OXM include a bitmask in payload
	oxm_length	8	Length of OXM payload



Name	Width (bits)	Usage
cav_offset	16	starting position of the attribute (e.g., temperature)
cav_cast	4	data type of the attribute (e.g., int32)
cav_op	4	operator (e.g., >,<,=)
cav_value	determ. by cav_cast	value

Figure 2.10: Sensor OpenFlow flow tables, illustrating OXM and CAV extensions [7]

While 6LoWPAN does a good job at compressing the headers, we can do better with SDN. Based on the fact that wireless sensor networks are typically a relatively small number of nodes communicating with some sort of central gateway or data store (in our case, an SDN controller), we make the following observations and optimizations:

- Firstly, we propose using a simple tag-based addressing scheme, limited to 4 bytes. This may be a 4-byte destination address, a 2-byte source destination combination, or an MPLS-like chain of forwarding addresses; in fact, any tag assignment scheme is possible given the fact that the networking application on the controller is completely configurable, so given some customization ability of the wireless nodes themselves, we can determine the best system for the network and reduce the size of the tag field to

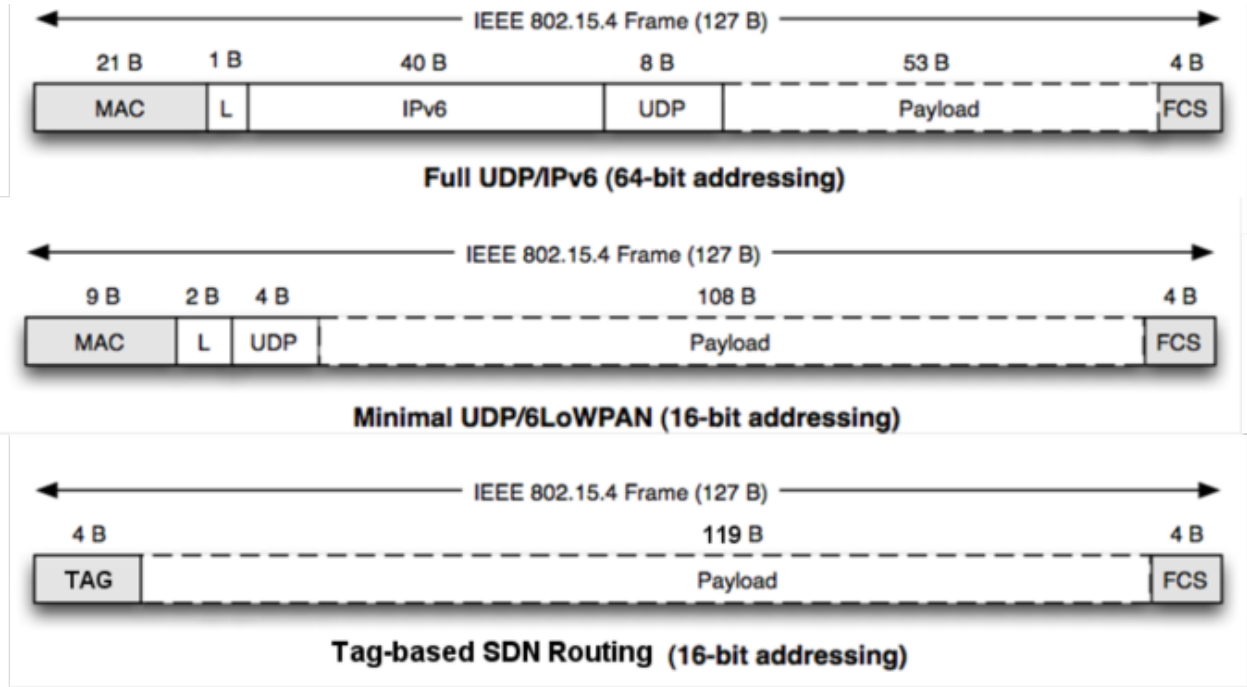


Figure 2.11: Packet sizes for IPV6 networks, 6LoWPAN networks, and SDN-based networks [2]. Note the significant loss of payload size for the traditional IPv6 protocol.

whatever seems most applicable. This addressing method maps well to Sensor OpenFlow's "Class-1" addressing schemes, making it a viable choice should we implement this protocol [7].

- With SDN, we gain complete control of the transport protocol, meaning there is no explicit need to implement the UDP protocol. Because of this, we can exclude the UDP header and save a few more bytes of space.

- As a final "trick," if we keep the size of each packet across all nodes fixed such that the controller knows the packet structure exactly, we avoid having to store a "length" field (L) in the packet. This gives us a couple more bytes, which can add up in the long run. These optimizations plus the fact that devices no longer need to spend CPU cycles shrinking headers mean that we can achieve better "goodput" (non-header throughput) using the SDN paradigm.

2.5.1.2 Routing

Traditional wireless sensor architecture tends to rely on a node-initiated network building approach, which naturally includes some limitations. As an example, we will consider 6LoWPAN's RPL, Routing Protocol for Low-Power and Lossy Networks, depicted in Fig. 2.12. This protocol uses destination-advertisement broadcasting to allow the individual nodes to form a routing network without outside input [60].

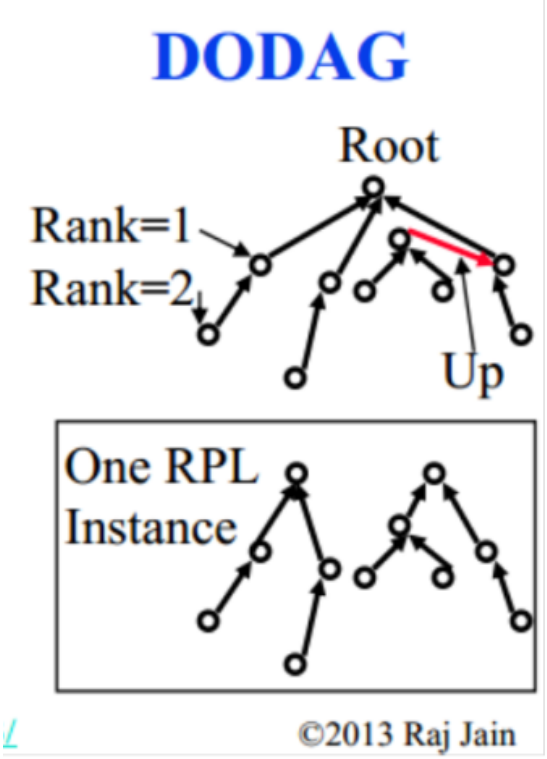


Figure 2.12: DODAG Routing Method in 6LoWPAN [8]

In 6LoWPAN RPL, the routing network is formed by building a Destination-Oriented Directed Acyclic Graph, or DODAG. In general terms, a DODAG is a tree-like directed graph in which all nodes point upward to a common "root" node, typically a data sink. Without diving into the mechanics of forming a DODAG, the essence of this scheme is that the formation of such networks can be done in a purely-distributed manner. Nodes broadcast their existence, and neighbors that discover these nodes can build the necessary routing links

to establish itself as part of the DODAG [60].

While this method works well for wireless sensor networks because of its purely-distributed nature, there exist significant limitations in this scheme which can be resolved by adopting an SDN approach instead. First of all, DODAG routing is completely data-oriented; that is, fundamentally, all routing in a DODAG leads to a singular destination node, typically a data sink of some sort. While this is generally desirable for sensor networks, this leaves no room for actuators or alternative destinations in the routing scheme. The one-way links of a DODAG prevent the controller (or other nodes) from communicating "downstream," meaning that any sort of action-takers must be set up on a separate network. Traffic must be routed through the data sink back out to the network again, meaning it is not possible to set up "priority" routes between sensors and alternative destinations (e.g. a temperature sensor to a fire suppression system) without making a trip through the central system first. This can be inefficient in such cases.

As part of our SDN paradigm, we propose an alternative routing protocol, SDN-RPL, depicted in Fig. 2.13. It uses a similar method of broadcast-discovery as 6LoWPAN RPL, with the important distinction that the SDN controller handles assignment of flows. From a high level, the following steps are performed:

- A new node wishing to join the network broadcasts its existence.
- Neighboring nodes that are within range of the join request will forward this toward the SDN controller. Nodes that have already received a join request from the node within a certain time window will ignore the request, to prevent redundant requests from being sent (e.g. two nodes within range that send requests to a common "parent" node).
- SDN controller determines the best place in the network for the new node and sends a downstream add-flow request to add the node to the network. The exact "how" is intentionally left black-boxed, as the global view of the SDN controller means that a wide range of network-building algorithms may be used depending on the application.
- SDN controller may optionally deny nodes from joining the network based on some criteria, in which it sends an add-flow request to instruct neighbors to drop packets

from the ”new” node (to prevent flooding of excess packets from unauthorized nodes). While security is still a to-be-researched topic in our proposal, an initial idea is to use a pre-configured symmetric encryption key on devices in a single network; if the key of the new nodes does not match (i.e. packets cannot be decrypted by the controller), the request is denied.

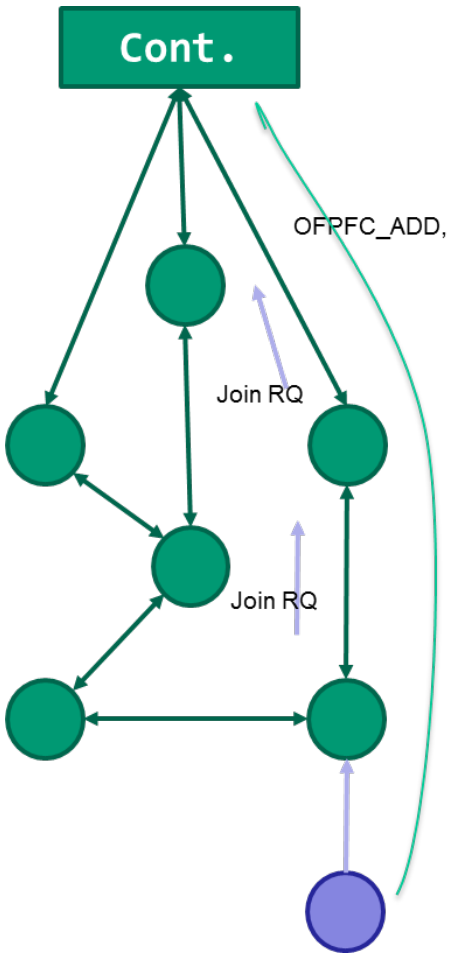


Figure 2.13: SDN-based routing methodology

The benefits of this methodology fill in some of the gaps of the DODAG-based approach. The controller is able to intelligently set up bi-directional traffic; that is, nodes can communicate ”down” and ”sideways” as well as ”up.” This may be used, for example, to create ”critical” flows directly from a sensor to an actuator. Next, the actual operation of the nodes

is simplified, as no routing algorithms must actually run on the devices themselves. Finally, the controller-originated scheme means that more complex network-building algorithms may be employed to create an efficient network, a luxury which the purely-distributed DODAG approach does not have.

A potential downside to this methodology is that building an initial network is likely to be slower, as the nodes must field requests to and from the SDN controller, which can be problematic en masse as the nodes are likely to be overwhelmed with packets. For this reason, it may be desirable to still implement DODAG-based network building to create the initial network, with the modification of creating two-way links between nodes. From here, the SDN controller may adjust the network as needed using the above methodology. While we do not dive into the specifics on implementing this approach, we recommend further research into this modified DODAG network-building algorithm as part of an SDN implementation step.

2.5.1.3 Failure Recovery

The next advantage that SDN brings is more efficient failure recovery. Systems like WirelessHART, of which the basic architecture is depicted in Fig. 2.14, employ redundant links for fault tolerance. In fact, WirelessHART mandates that each node has two links, and its functionality is such that each node must be a FFD (full-function device) [63]. The key idea is that if any link goes down, its neighboring nodes are guaranteed not to be isolated and will quickly reconfigure themselves to establish a second connection once more, resulting in quick recovery.

While there is nothing wrong with this type of redundancy per se, the fact that is absolutely enforced as part of the architecture may pose efficiency and performance problems for certain types of networks where 100% reliability is not absolutely required. We will show that our SDN-based approach not only provides robust options for failure recovery, but also provides the best of both worlds in terms of redundancy, allowing for it when it is desired without enforcing it strictly.

Fig. 2.15 shows an alternative SDN-based scheme for failure recovery, based on the idea of "heartbeats." Each node will periodically send out a "heartbeat" signal to its neighboring

nodes, and if a node fails to receive a heartbeat from a neighbor after a certain period of time, it sends an "alert" message to the controller. The controller then will attempt to contact the non-responsive node to diagnose it as non-functional; if it is indeed offline, it can then reconfigure the neighbors of the dead node by sending add-flow requests.

The recovery scheme described is intentionally generic, as the fact that the controller has a global network view means that a myriad of recovery options are possible. For example, if an entire subnet goes down but the controller knows the spatial positions of the nodes (i.e. X, Y, and Z coordinates on some system that maps to real-world space), it can determine if any neighbors are in physical range to a node on the missing subnet and instruct the node to re-establish a connection to the isolated system. The actual method by which it determines the best nodes to connect is up to the implementer to decide, and can be configured per-network.

In general, this means that our failure recovery architecture does not necessarily require that redundant physical links be actually established as in WirelessHART; in fact, we can implement the exact same scheme by simply having multiple nodes within range of each other. Since the controller is able to detect the absence of a node and send the instruction to reconnect, the nodes themselves simply have to wait for such a command and continue processing as normal with as few links as desired.

As a side-note, the reduced functionality on the individual node side may allow the devices to be RFDs (reduced function devices), depending on the complexity of other features (security, etc.), which may be another point in SDN's favor compared to WirelessHART.

As before, we do not propose a formal algorithm for determining where to add the nodes to the network, instead opting to emphasize the fact that this exposed "black box" allows developers of wireless sensor networks to implement highly-customized networking functions on the SDN controller that find the best solution for the network's need. The SDN approach opens the door for creative and engineering freedom in this regard, which we believe is a huge boon for the Internet of Things.

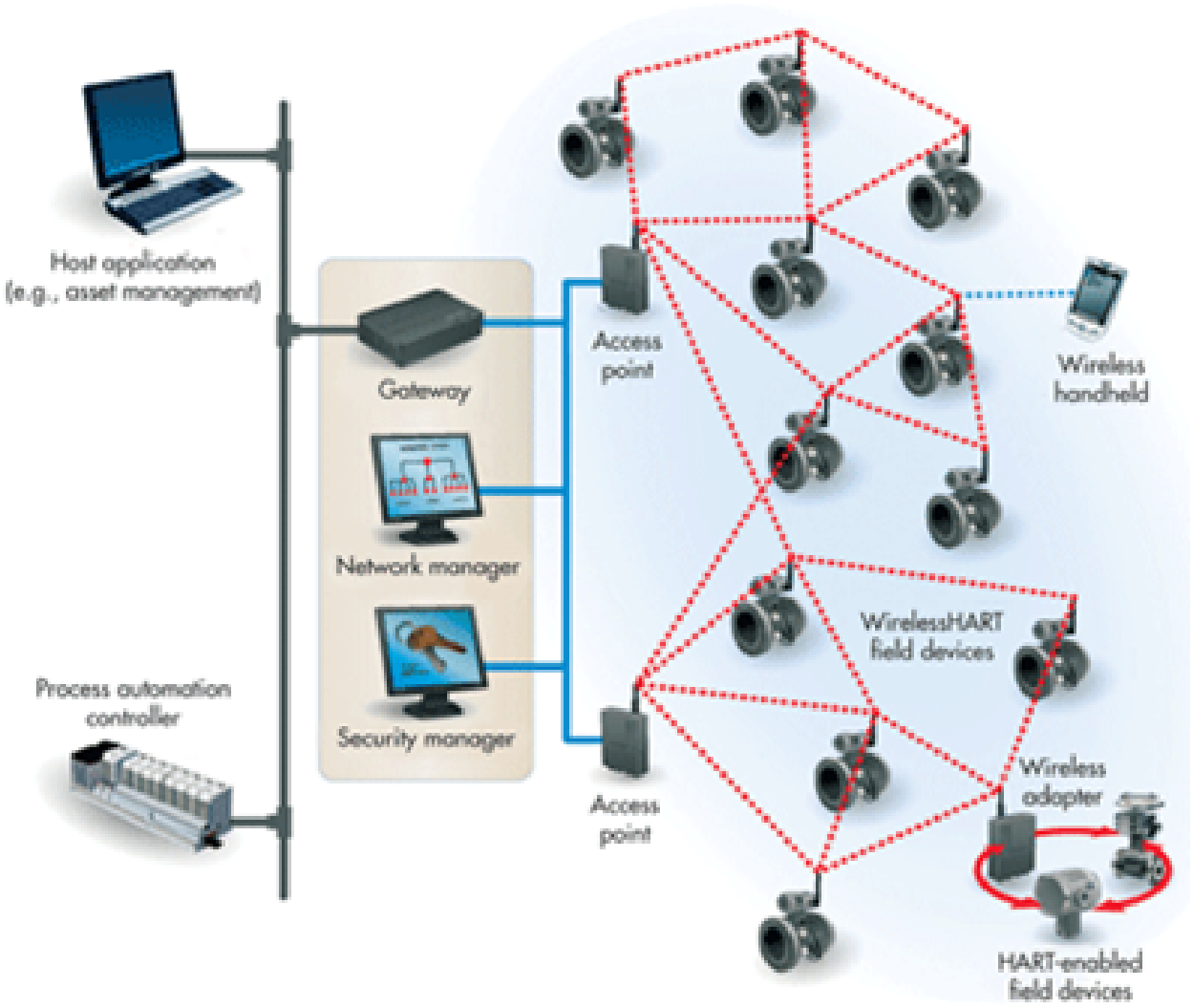


Figure 2.14: WirelessHART architecture, showing double-connectivity [9].

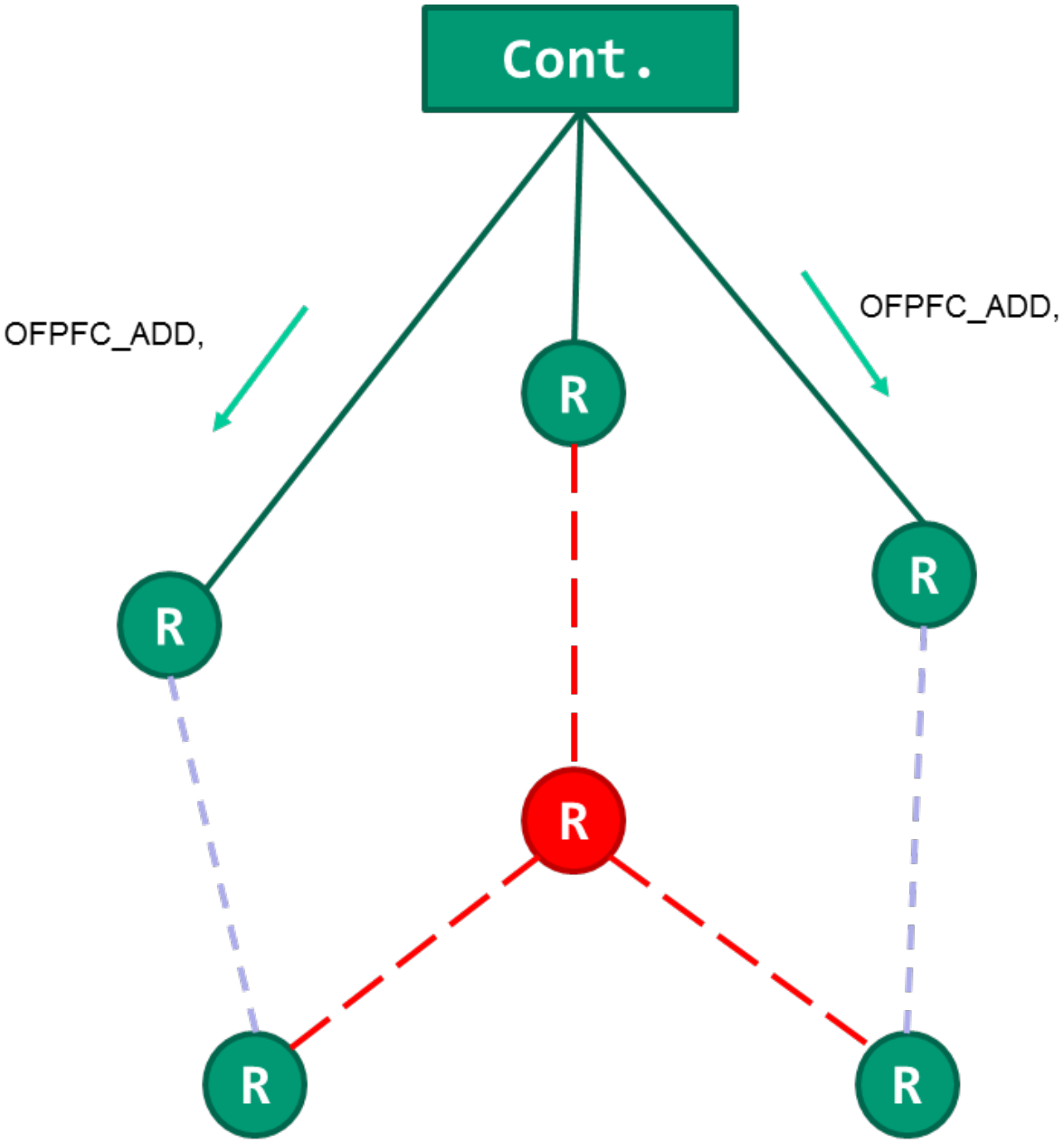


Figure 2.15: SDN-based failure recovery. Broken nodes/links shown in red, with new flows shown in blue.

Chapter 3

BLACK IoT COMMUNICATIONS PROTOCOLS

In this chapter, we introduce the concept of Black packets. Black packets encrypt both the data and the metadata, at each layer in the communications protocol. Black packets remain compatible with the existing protocol in use. Black packets are the first step to simple Black networks, which may use simple Black communications, or Black routing. Black networks may then be implemented as part of an architectural framework for Secure Smart Cities. We present Black packet designs for a variety of popular IoT protocols - Bluetooth Low Energy (BLE), ZigBee and 6LoWPAN. Black packets may be designed for any communications protocol. We extend Black packet design to IPv6 packets. We offer detailed security analysis and payload efficiency (a comparison of the maximum payload that can be carried by a Black packet vs. a regular packet of the corresponding protocol) calculations for BLE Black packets.

3.1 Introduction

The Internet of Things (IoT) is impregnating our world, impacting our lifestyles, our personal lives and the industries adopting IoT. The impact is so great that we are now dependent upon the IoT in a plethora of realms - from remote medical device monitoring to vehicle operations to automated lighting and heating, cooling and ventilation. IoT nodes are small, often powered by a tiny battery expected to last for several months or even several years. Typical IoT nodes communicate wirelessly forming ad-hoc networks where no infrastructure exists. To conserve battery power, IoT nodes *sleep* for a majority of their life-cycle (up to 90% of the time), ẅaking to sense and to communicate sporadically. Widespread deployment of IoT systems has lead to mission-critical information communicated across IoT networks that, therefore, need to be secured end-to-end.

While security is provided at all levels of the communications protocol, IoT nodes have

unique security challenges that are not mitigated by its security mechanisms. Small in size and resource-constrained, IoT nodes are susceptible to physical accessibility by malicious actors. Cryptographic keys and data may be physically extracted from an IoT node, or a node may be replaced by another node, under the influence of a malicious actor. Resource draining attacks (like spurious messaging) keep IoT nodes awake and computationally active. These and other advanced attacks are not accounted for in IoT communications protocol security mechanisms.

Additionally, the metadata associated with IoT communications is in the clear – presenting eavesdropping, track and trace, packet corruption, and packet injection vulnerabilities. Metadata includes header information, such as source address, destination address, frame counters, frame sequence, and frame type (e.g. acknowledgements). For mission-critical data, the protocol must build-in confidentiality, integrity, authentication and privacy for even the metadata.

The remainder of this chapter is organized as follows: In Section 3.2 we review BLEv4.2, present Black BLE packet designs for Advertising and Data PDUs and perform a security analysis on Black BLE. In Section 3.3, the packet design for Black IEEE 802.15.4 frame is presented. Section 3.4 presents the Black ZigBee packet design and Section 3.5 presents Black 6LoWPAN, and Black IPv6 packets. All of the above Black packets are constructed using an authenticated encryption, stream-based cipher such as the AES cipher in EAX mode (AES-EAX) or Grain-128a.

3.2 Black Bluetooth Low Energy (BLE)

Bluetooth Low Energy (BLE) is a wireless Personal Area Network (PAN) protocol for IoT commonly used in a range of consumer electronics and in medical, fitness, automotive and home monitoring applications [64]. BLE is a low-power (1mW), synchronous, medium range (100 meters), high-rate protocol (1 Mbps/sec). BLE operates within the ISM band (2.4GHz-2.438GHz).

We present a Black Networks [14] approach to BLE security that encrypts both the data (payload) and the metadata. We do this at the BLE Link Layer using AES in EAX mode (an authenticating cipher) [67]. The resulting Link Layer Advertising and Data PDUs

(Packet Data Units) are BLE compatible, and we evaluate them for payload efficiency. Our Black Network increases the frame and packet overhead while securing all of the metadata information at each layer without exceeding the maximum frame and packet size respectively.

The primary objective of Black Networks is to secure all data, including the metadata, associated with each frame or packet [14]. This security is achieved by encrypting all information contained in a frame that may be used by an adversary. Adversaries should not be able to determine the source, the destination, the frame sequence number or the replay counter [27]. We generate a Black BLE Link layer frame using AES in EAX mode. EAX mode works as an authenticating stream cipher (uses constant memory to process a data stream), is simpler, and more efficient than CCM. EAX has flexible tag, nonce and key sizes up to the block size. The Black BLE Link layer frame is created to keep the maximum frame size of 39 bytes (header + payload) for an Advertising PDUs, and 257 bytes for a Data PDU. The BLE node and the controller communicate by means of a shared secret. A connection-oriented link does not require an IV/nonce to be sent with each packet, and the sender/transmitter of information is always known due to the shared secret used to decrypt and authenticate.

3.2.1 BLE Black Advertising PDU

ALL Advertising PDUs are kept at a fixed length of 39 bytes for a BLE-compatible PDU (the payload is fixed at 37 bytes). In the header field, the first 4 bits are the PDU Type and indicate a Black BLE Advertising PDU set at 0111 (this is the only metadata field left unencrypted), and the last 4 bytes are the MIC, resulting from the AES-EAX encryption, over the header and the payload. To distinguish between the different PDUs, the 4 Reserved for Future Use (RFU) bits are used to identify the advertising payload type. The header and payload are encrypted as a single block after the above transformations and a 32-bit MIC is added.

Figure 3.1 shows an Advertising PDU and its transformation to a Black BLE Link layer PDU.

The different Advertising PDUs form Black BLE Advertising PDUs as follows:

- ADV_DIRECT_IND and SCAN_REQ PDUs have payloads that are 96 bits. We

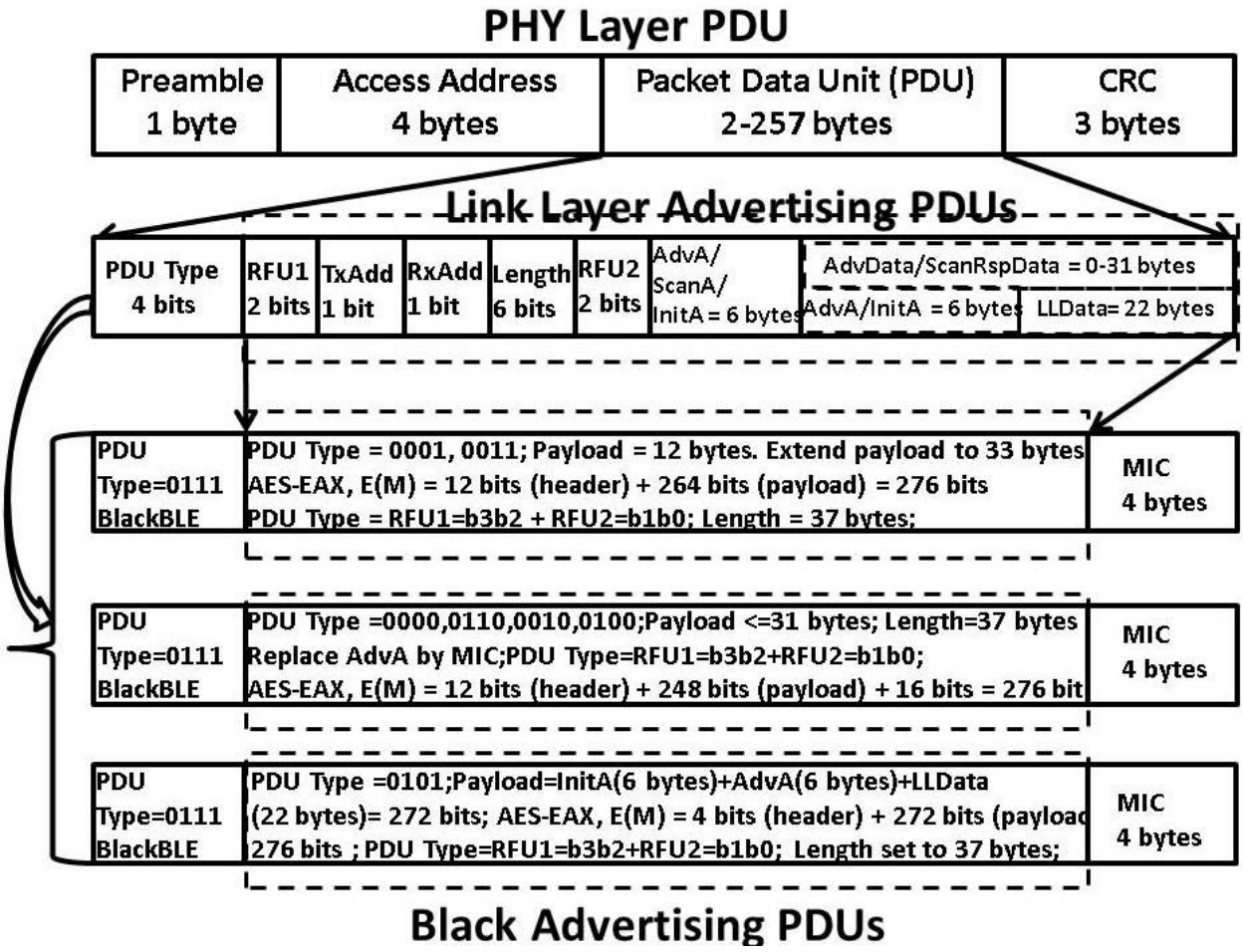


Figure 3.1: BLE Advertising PDU encryption to Black Advertising PDU

extend the payload to 264 bits. These Advertising PDUs have adequate space for a MIC, and the payload to be included in 37 bytes.

- ADV_IND,ADV_NONCONN_IND,ADV_SCAN_IND, SCAN_RSP PDUs have a payload of upto 296 bits. These Advertising PDUs may not have additional space in the payload to include the MIC, therefore some transformations have to be performed. We use the 4 bytes of the AdvA address for our MIC, and effectively do not send an AdvA. We encrypt over the 12 bit header (4 bits is kept to indicate the PDU Type) and 264 bit payload (of which 16 bits are "dont care", left over from the AdvA).

- The CONNECT_REQ PDU payload is 34 bytes. After setting our PDU Type to Black, and setting our RFU bits for the actual PDU Type, we are left with the Length, TxAdd, RxAdd fields which we use to get an extra 8 bits for the MIC. Now we have a full frame of a 4 bit header, 276 bit payload and a 32 bit MIC.

3.2.2 Black BLE Data PDU

Fig 3.2 shows a BLE Data PDU and its transformation to a Black BLE Link layer Data PDU using AES-EAX encryption. This Black BLE Data PDU transformation is relatively simpler since a 32-bit MIC field is already provisioned.

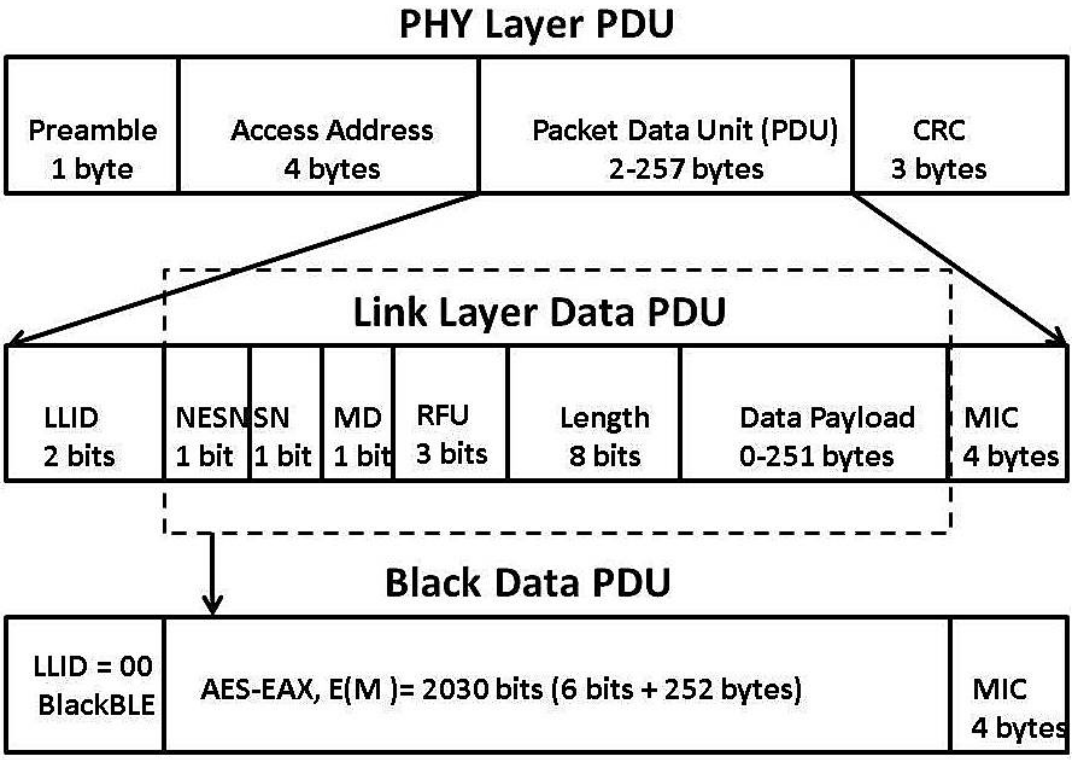


Figure 3.2: BLE Data PDU encryption to Black Advertising PDU

We use the LLID field (2 bits) in the header to indicate a Black BLE Data PDU (LLID = 00 is Black BLE). To indicate an LL Data PDU or an LL Control PDU, we use the 3 RFU bits in the header. Except the LLID bits, we encrypt the remaining header (14 bits) and

payload (upto 2008 bits) as a single block using AES-EAX. In Black BLE communications, Symmetric Link layer keys are used for secure communications and authentication. For nodes that are relatively close to each other, broadcast routing, which mitigates an adversary from determining the destination, is used as the basic approach. While the encrypted source and destination addresses protect against insider attacks, they require specialized routing, and increase the network traffic.

We have shown the transformations required for converting a BLE Advertising PDU and a BLE Data PDU to their corresponding Black PDUs. The resulting Black PDUs are BLE-compatible. Once received, a Black PDU is decrypted using a shared symmetric key. Correct decryption and authentication provided by the MIC allows the sender to be securely identified and authenticated.

3.2.3 Security Analysis

BLE Advertising PDUs are transmitted on channels 37, 38 and 39, allowing eavesdropping and traffic analysis on specific channels. Black BLE PDUs always provide Privacy, Confidentiality, Integrity and Authenticity by encrypting both the header and the payload using AES-EAX. Black BLE PDUs mitigate the NO security Level 1 of Security Mode 1 and do not require the privacy feature to be implemented in the devices. However, this Black security functionality comes at the expense of a payload increase. Table 5.1 compares the payload efficiency of Black BLE Link layer Advertising and Data PDUs with existing BLE PDU types. Advertising PDUs ADV_DIRECT_IND and SCAN_REQ increased by a constant 178.6% with their Black transformations for minimum, maximum and average PDU sizes. ADV_IND, ADV_NONCONN_IND, ADV_SCAN_IND, SCAN_RSP Black transformations ranged from no increase to a maximum of 387.5% increase with an average of 69.6% increase for corresponding Black PDU. CONNECT_REQ ranged from a minimum increase of 8.3% to a maximum increase of 178.6%, and an average of 56% for the Black version. For BLE Data PDU, the payload increase ranges from 0% to 12750%, with an average payload increase of 97.7%. While inference attacks can be made on the variable length BLE Advertising and Data PDUs, the Black BLE Advertising and Data PDUs mitigate payload length-based attacks because of their fixed length. Additionally, the performance impact of

processing maximum size PDUs can be offset by optimizing for Black PDUs of fixed length. There are efficiencies for header encryption (associated data) for the EAX mode (done once for same header). Finally, we note that the Access Address in the PHY layer is indicative of a connection between two devices. We propose a nonce/key pair for each node for generating an Access Address keystream, using AES-CTR. This keystream can be sent in clear to hide the Access Address by acting as a secured, changing address and mitigate eavesdropping and traffic analysis.

Table 3.1: Black PDU Payload Efficiency

PDU Type=0000, 0110, 0010, 0110		*B*A PDU	*Bytes gain*	*% increase*
Min	8 bytes	39 bytes	31 bytes	387.5%
Avg	23 bytes	39 bytes	16 bytes	69.6%
Max	39 bytes	39 bytes	0 bytes	0.0%
PDU Type=0001, 0011		*BA PDU*	*Bytes gain*	*% increase*
Min	14 bytes	39 bytes	25 bytes	178.6%
Avg	14 bytes	39 bytes	25 bytes	178.6%
Max	14 bytes	39 bytes	25 bytes	178.6%
PDU Type= 0101		*BA PDU*	*Bytes gain*	*% increase*
Min	14 bytes	39 bytes	25 bytes	178.6%
Avg	25 bytes	39 bytes	14 bytes	56.0%
Max	36 bytes	39 bytes	3 bytes	8.3%
PDU Type= Data		*BD PDU*	*Bytes gain*	*% increase*
Min	2 bytes	257 bytes	255 bytes	12750.0%
Avg	130 bytes	257 bytes	127 bytes	97.7%
Max	257 bytes	257 bytes	0 bytes	0.0%

3.2.4 Conclusions and Future Work

BLE-based IoT networks are engaged in mission-critical functions in industries and in applications such as medical devices and personal health monitoring. Securing the metadata for BLE Advertising and Data PDUs, by encrypting both the data and the metadata, at the Link layer and the Network layer, mitigates a range of attacks including eavesdropping, track and trace, packet injection, and packet modification. This Black Network method of

securing all aspects of communications within a network is done at the expense of symmetric key management, decreased routing efficiency and reduced payload efficiency. Future areas of research will focus on securing the Black frame by multiple methods that allow for a fine-grain approach to securing the metadata. An additional area of critical research is developing improved routing mechanisms, payload efficiency and extending Black Networks to non-IoT networks for end-to-end security.

The biggest vulnerability of IoT communications is the metadata [27]. Table 4.1 demonstrates that existing IoT communication protocols do not address metadata vulnerability. Black networks mitigate metadata vulnerability by encrypting the entire PDU with fixed-length packets of the maximum frame size.

3.3 Black IEEE 802.15.4

In this section, we present Black Networks for IoT devices. Black Networks secure the metadata and the payload within each layer. We specifically examine the IEEE 802.15.4 protocol in this section. The Black Network for the 802.15.4 Link layer communications by encrypting the metadata, and includes the cipher's initialization vector (IV) and encrypted metadata in the communicated frame. We similarly secure the metadata independently within the Network layer for protocols such as 6LowPAN, ZigBee and WirelessHART. The resulting 802.15.4 compatible frame, allows the intended recipient to correctly receive and decode the message while all other receiving nodes are unable to decode any data, including the sender and the receiver addresses.

With large networks of IoT nodes, routing becomes critical. We examine the impact of broadcast routing on the performance of Black Networks.

Black Networks mitigate a broad range of both passive and active attacks, due to the authenticated and secured communications at both the Link layer and the Network layer. Adversaries should not be able to determine the source, the destination, the frame sequence number or the replay counter [27].

Location information and communication patterns can be obtained from the metadata. Prior work in this area has been done by Conti et. al. [68], in wireless sensor networks. Source Location Privacy (SLP) allows the sender location to be hidden from adversaries. SLP

is achieved via multiple methods: Random Walk, Geographic routing and Network Layer Anonymity. Some secure routing mechanisms are Random Routing Scheme, Dummy Packet Injection Scheme and Anonymous Communication Scheme (ACS) [69], Anonymous Path Routing (APR) [70], Simple Anonymity Scheme [71], Destination Controlled Anonymous Routing Protocol for Sensor nets (DCARPS) [72] and Hashing Based Identity Randomization [73].

3.3.1 Black 802.15.4 Link Layer Frame

Figure 3.3 and Figure 3.4 show the IEEE 802.15.4 Link layer frame and its transformation to the Black Link Layer frame for the ZigBee and 6LoWPAN network layer packets respectively.

Flags are used to indicate a Black Link layer frame. This is the only metadata field left unencrypted. The Frame Control field is set to indicate a Black frame for IEEE 802.15.4. The initialization vector (IV) is used to synchronize the cryptographic engines for Link layer communication. Symmetric Link layer keys are used for secure communication and authentication. The IEEE 802.15.4 Black Link layer frame can be formed by encrypting the header and payload as a single block, using the Grain-128a authenticating cipher [74], resulting in an IEEE 802.15.4 compatible frame. An alternative method is to replace the header fields to be secured (all header fields except the Frame Control field) by an IV and a keystream. the resulting frames are both 802.15.4-compatible.

3.4 Black ZigBee

The IEEE 802.15.4 protocol defines a PHY and MAC sublayer, forms the basis for multiple higher layer protocols - ZigBee, 6LoWPAN and WirelessHART [9] [4] [63], being the most widely-used. The BLE communication protocol is based on a series of iterative standards Bluetooth 4.0 and above [13].

Our challenge, in IoT networks, is to mitigate internal and external threats, a range of active and passive attacks, and secure the communications per-hop and end-to-end. In this chapter, we focus on securing the IoT communications protocol, at each layer of the protocol stack. We assume that in wireless communications, the sender is always visible to an adversary. Our goal is to ensure that mission-critical IoT communications have built-in

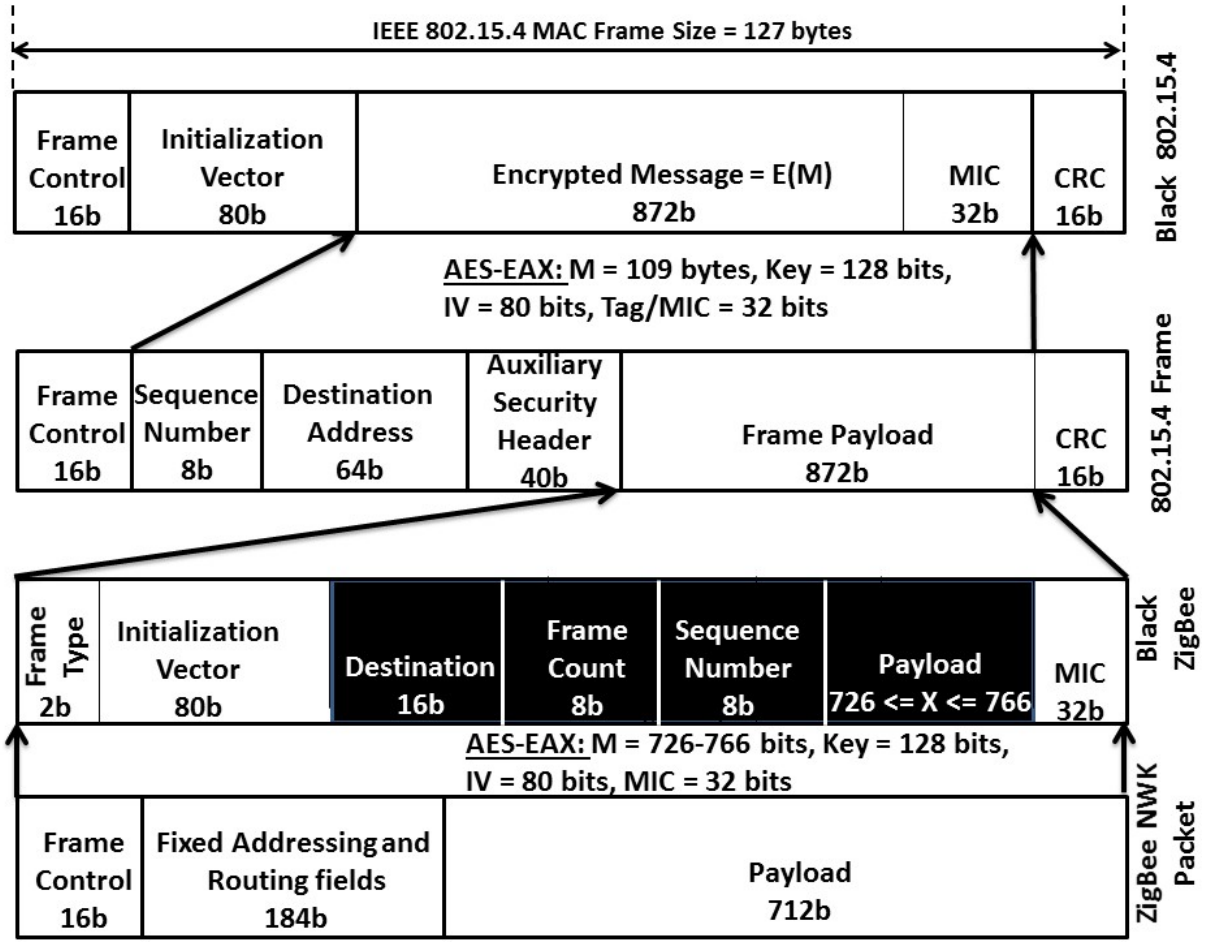


Figure 3.3: Black ZigBee packet and Black 802.15.4 frame

confidentiality, integrity, message authentication and privacy. The resulting communications, must be compatible with existing IoT protocols. To prevent inference and packet-length based attacks, we communicate using fixed length packets (the maximum size allowed by the IoT protocol - eg. 127 bytes in IEEE 802.15.4). Finally, our goal is to mitigate insider threats. A malicious node, or an intruder within the network (commonly referred to as an insider threat), must not be capable of deciphering a message that is not intended for it. This is achieved by allocating a unique symmetric key for each IoT node.

To achieve the above security objectives, we introduce Black Networks - where the meta-data AND the data are encrypted for every frame/packet/segment at each layer of the communications stack. AES in the EAX mode (or Grain-128a) is the preferred cipher (to

maintain payload efficiency). Securing the metadata leads to routing challenges, as the metadata contains source and destination information. Flooding and Broadcast over IoT networks, are ineffective, when the nodes sleep a majority of the time (a common energy-saving mechanism for IoT nodes).

The secured ZigBee NWK data PDU has an additional 112 bits of Auxiliary Header, which reduces the payload to a 528 bits-568 bits range. The 40-bit difference is the Auxiliary Security Header of the IEEE 802.15.4 frame. Therefore, if the 802.15.4 frame is also secured (recommended), then the secured ZigBee NWK Data PDU payload is 528 bits. If the 802.15.4 frame is not secured, then the secured ZigBee NWK Data PDU is 568 bits (Fig 3.3). Fig 3.3 also shows the transformation of an IEEE 802.15.4 frame to a Black frame [14], and the transformation of a ZigBee network PDU into a Black ZigBee packet.

The ZigBee network layer data packet header contains 16 bits of frame control information. The first subfield in the frame control field is Frame Type (data, control or command). The Frame Type reserved bits $b_0b_1 = 11$ indicate a Black ZigBee packet. Excepting the first 2 bits, the remainder of the packet is encrypted using an authenticating cipher, such as AES-EAX [67] (or Grain-128a [74]). An 80-bit Initialization Vector (IV) is included with each packet [75].

3.5 Black 6LoWPAN and Black IPv6

Black packets are fully encrypted, fixed length PDUs, at every layer of the communications protocol. Black packets [13] and Black networks [14] secure the metadata associated with the communications and mitigate a range of active and passive attacks. The PDU encryption is performed by an authenticating, stream-based cipher, such as Grain-128a [74] or AES in the EAX mode [67]. Figure 6.1 shows the transformation of an 802.15.4 Link layer frame, to a Black 802.15.4 frame, and a 6LoWPAN packet to a Black 6LoWPAN packet. The authenticating cipher used is the Grain-128a, with a key of 128 bits and an initialization vector (IV) of 128 bits. The only portion of the packet, that is not encrypted are the first 8 bits of the dispatch type and header = 11111111. This is a reserved value where $b_0b_1 = 11$ indicating a Black 6LoWPAN packet, and $b_2b_3b_4b_5b_6b_7 = 111111$, a header type of ESC, indicating an additional dispatch byte follows. To maintain privacy and security [40], the

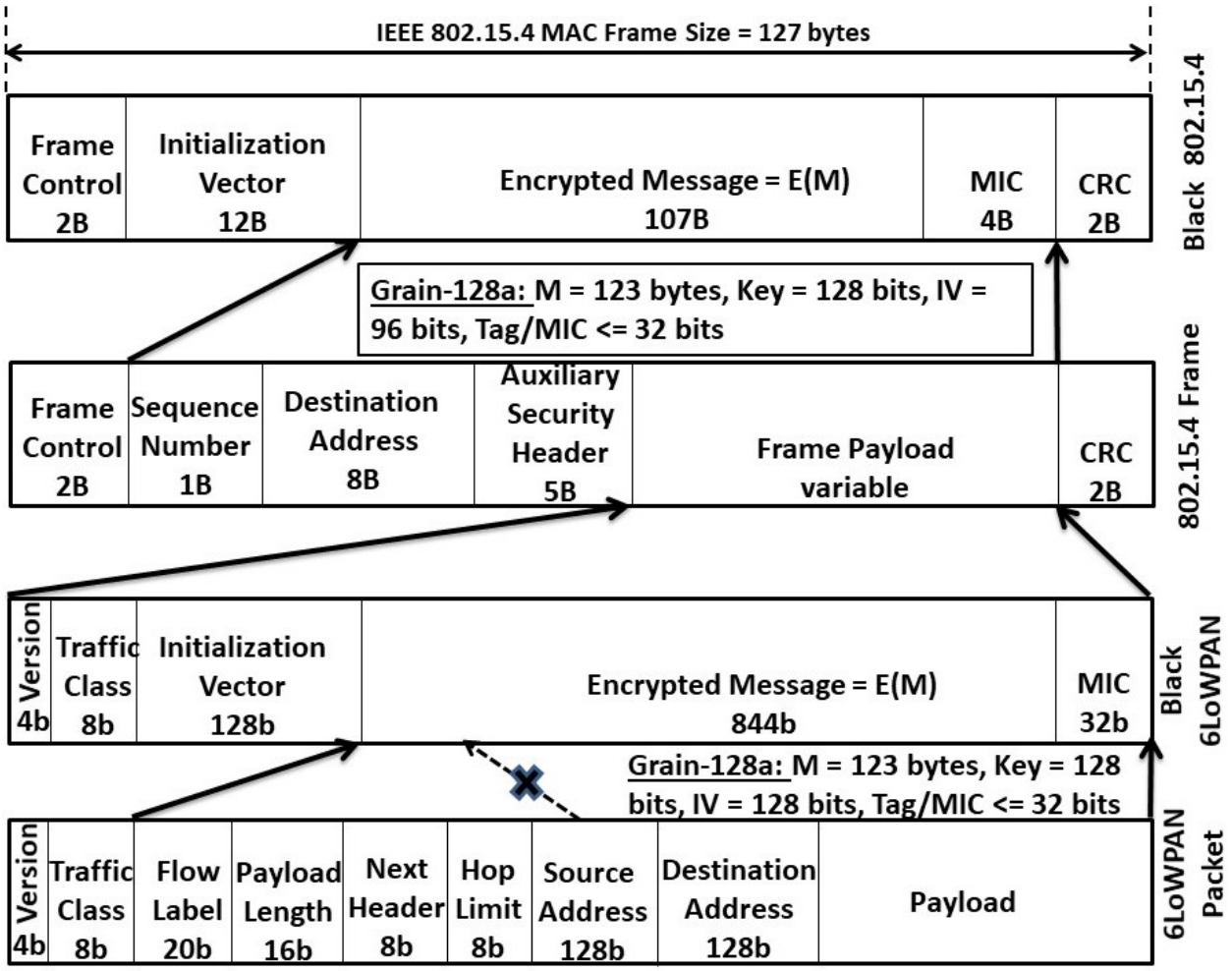


Figure 3.4: Black 6LoWPAN packet and Black 802.15.4 frame

IV replaces the Source and Destination address bits and is expanded to 96 bits. The Traffic Class bits indicate the Black Network layer packet (Black 6LoWPAN), and Grain-128a is used to encrypt the remainder of the payload and header. A symmetric key is shared between sender and receiver before communication begins. Flow label, payload length, next header, hop limit and destination address are encrypted along with payload and included as the Encrypted Message.

The Link layer transformation, from standard 802.15.4 to Black 802.15.4, is outlined in Section 3.3.

Figure 3.5 shows a similar transformation for an IEEE 802.11 frame [76] to a Black 802.11

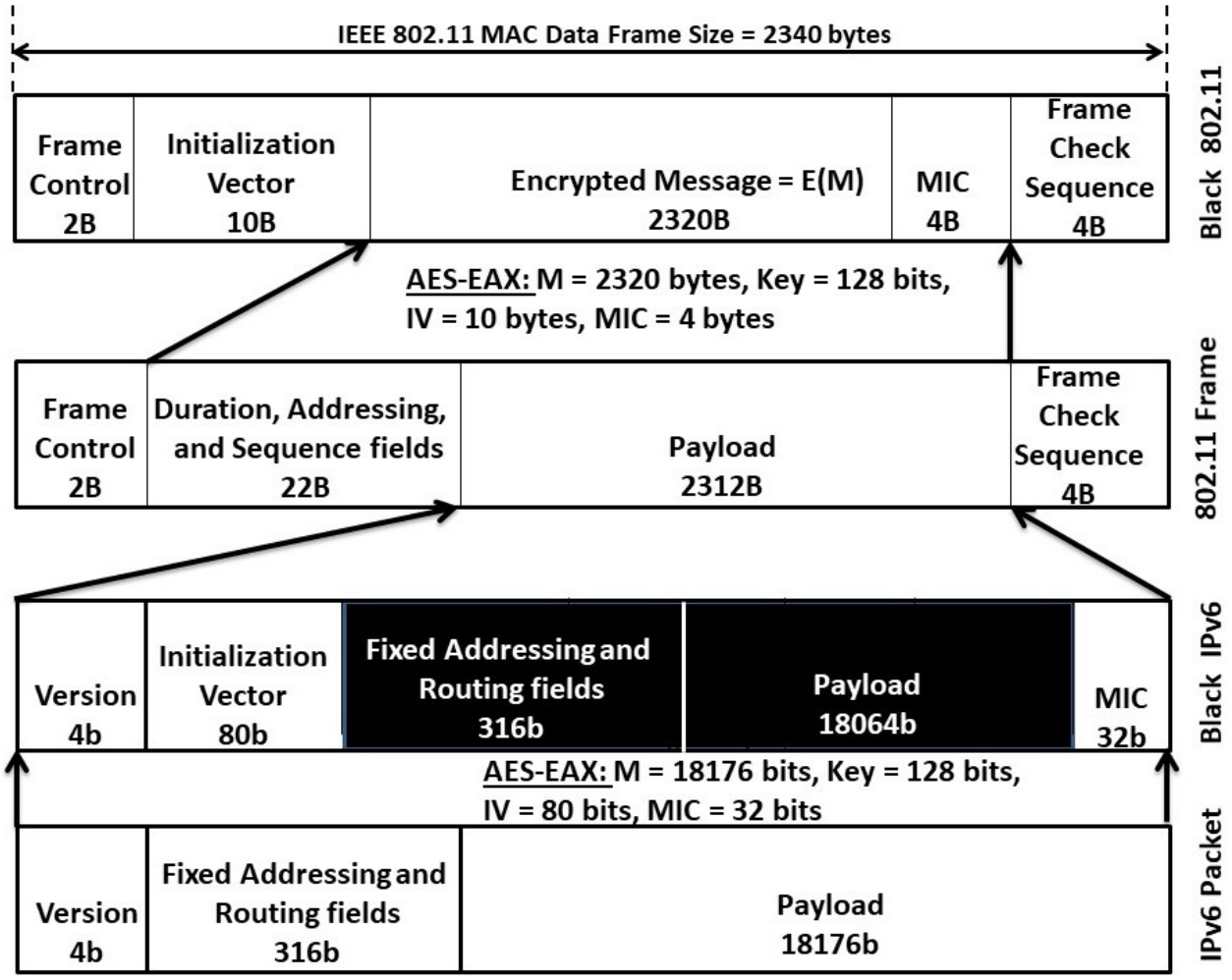


Figure 3.5: Black IPv6 packet and Black 802.11 frame

frame and an IPv6 packet [77] [78] [79] to a Black IPv6 packet. To demonstrate the flexibility in creating Black packets, we used the AES cipher, in the EAX mode, with a 128-bit key, and an IV of 80 bits (10 bytes). For an IP packet, the version field of 4-bits is either a 4 (for IPv4) or 6 (for IPv6). We use the reserved value of 1111 to denote a Black IPv6 packet. For the 802.11 frame, the first 2 bits of the 2-byte Frame Control field denotes the protocol version, and we use the reserved field of 11 to denote the Black 802.11 frame. In Fig. 3.4 and Fig. 3.5, the first 2 bits identify the frame as a Black frame, and are the only unencrypted portion of the frame metadata. The IV is in the clear, with the rest of the frame information encrypted.

Chapter 4

BLACK COMMUNICATIONS

In this chapter we present simple Black communications in Black networks. We present Flooding, Broadcast routing and Black Gateway communication using Black packets. With the metadata secured, and no additional components added to the network, Black networks may communicate with simple communications (instead of routing, which is presented in Chapter 6). We simulate the performance of all three communication methods and evaluate them against Shortest Path routing, measuring Overhead Traffic, Mean Wait and Mean Travel times. Other measures include payload efficiency and compute efficiency (payload efficiency compares the payload overhead associated with Black packets compared to regular packets; compute efficiency compares the encryption, decryption and MIC computations in Black networks vs. Shortest path routing). We present a simple star topology, the Black Gateway configuration, with performance equivalent to Shortest Path routing with much greater levels of security.

4.1 Introduction

The Internet of Things (IoT) is a general term that refers to the broad range of limited functionality devices and smart things that are (at least occasionally) connected to the Internet. IoT connected things have been identified to include large things such as automobiles and small things such as RFID (Radio Frequency Identification) enabled smart packages. This broad range of IoT devices are integrated into networks designed for monitoring and control that communicate via wireless ad-hoc communications. These monitoring and control networks provide critical automated functionality in a broad range of industries including healthcare (remote and personal diagnostics), building management (smart homes/offices), industrial control systems (automation), energy management (smart grids), transportation (smart cars), and environmental monitoring (air quality) [11].

As billions of IoT devices are deployed across multiple industries, the resulting IoT networks are increasingly carrying mission-critical data and control messages and operating in an automated fashion. The reliance upon these IoT networks for automated communications and control makes them a target for attackers that wish to monitor and disrupt, these network communications. The communications security for IoT networks is based upon the communication protocols used since the networks themselves often operate in uncontrolled or semi-controlled physical environments.

All of the common IoT communication protocols, including ZigBee [3], WirelessHART [9], 6LoWPAN [40] [58] [59] and Bluetooth Low Energy (BLE) [64] [65], have well-known security vulnerabilities, and none of them protect the metadata associated with the communication. The metadata is vulnerable in these protocols as it can reveal sensitive information such as the likely value of the sensor data being communicated or the particular control message to be executed.

IoT networks are vulnerable to a broad range of attacks. They include resource exhaustion attacks [55] and node capture attacks [80] [81]. Possible threats to IoT nodes include hardware trojans [82] and sensor side-channel threats [83]. Active and passive attacks on the communications of IoT networks are common and relatively easy to perform. All of the above-mentioned attacks constitute a multi-dimensional threat to IoT networks. Table 4.1 shows vulnerabilities across commonly used IoT communication protocols such as 802.15.4 [10] [15], ZigBee [26] [84], 6LoWPAN [54] [85], WirelessHART [62] [12] and BLE [86].

IoT communication protocols must provide security and privacy per-hop and end-to-end. The main motivation for Black Networks is metadata visibility and metadata attacks. Black Networks encrypt the metadata and the data of a frame/packet, using a stream-based cipher such as Grain-128a [74] or AES-EAX mode [67]. The resulting encrypted packet data units (PDUs) are called Black frames/packets. Encrypting the metadata, presents significant routing challenges in existing protocols. However, Flooding or Broadcast can be effectively used to enable Black communications in an IoT network. Black communication packets do not divulge source, destination, contents or any information that can be used for traffic analysis. The main contributions of this chapter are a review of the Black ZigBee and Black BLE packet design and the simulation and characterization of two communication

Table 4.1: IoT Communication Protocol Security Mechanisms do not address Metadata threats

Protocol	Security Services	Security Mechanisms	Security Vulnerabilities
IEEE	Integrity	MIC	Weak integrity at 16 bits
802.15.4	Replay	Frame Counter	Frame counters in the clear and can be exploited
	Privacy	None	Subject to metadata attacks
	Confidentiality	Encryption (AES-CCM* mode)	Acknowledgements are unencrypted and can be exploited. Default security is NULL. NO timed frame counters.
ZigBee	Confidentiality	Encryption	Trust Center is vulnerable. Network Keys can be extracted
	Traffic Analysis		Subject to metadata attacks
6LoWPAN	Confidentiality	Encryption	IPsec/IKE unsuitable for IoT
	Integrity	MIC	IPsec unsuitable for IoT networks
	Authentication	Node Authentication	Subject to device-based attacks
	Privacy		Subject to metadata attacks
	IP services		IP attacks (HELLO flood, sinkhole and selective-forwarding)
WirelessHART	Confidentiality	Encryption	Default security always ON.
	Integrity	MIC	
	Availability	Channel Hopping, Channel Blacklisting	Jamming
	Exhaustion	10ms time slot execution	Limits resource exhaustion, but does not eliminate
	Privacy		Possible metadata attacks
BLE	Confidentiality	Encryption	Key stolen during key exchange
	Integrity	CRC	CRC seed can be recovered
	Availability	Channel Hopping	Channels tracked via Access Address
	Privacy		Subject to metadata attacks

mechanisms (Flooding and Broadcast) for a mesh network and a Black Gateway star network using Black packets.

The remainder of this chapter is organized as follows: In Section 4.2, we review the security and vulnerabilities of IoT networks, and present the Black Networks approach with Black frames and Black packets compatible with the ZigBee and the Bluetooth Low Energy (BLE) protocols. In Section 4.3 we present Black network communications for Flooding, Broadcast and Gateway. In Section 4.4 we present our simulation model for traffic overhead and network delay. We analyze the security, performance impact and payload efficiency of Black communications in IoT networks in Section 4.5. We draw relevant conclusions and identify future areas of research in Section 4.6.

4.2 IoT Security Review

Table 4.1 shows vulnerabilities across commonly used IoT communication protocols 802.15.4 [10] [15], ZigBee [26] [84], 6LoWPAN [54] [85], WirelessHART [62] [12] and BLE [86].

IEEE 802.15.4 security is provided by the MAC sublayer [39]. The security services provided are: Data Confidentiality, Data Authenticity, and Replay protection, and the default is NO security [56] [57].

In ZigBee, the PHY and MAC sublayer are defined by IEEE 802.15.4 and the Network and Application layers are defined by the ZigBee specification. Security for the NWK (network) and APL (application) layers are provided by the Security Services Provider. A secured ZigBee network PDU includes a 14-byte Auxiliary Header to provide security specific information, such as frame counters for replay attacks, security level, key identified and nonce fields. A secured NWK packet is not always encrypted. The AES-CCM* mode has NO security and integrity protection only options, with no payload encryption (for IEEE 802.15.4 and ZigBee).

BLE security services are provided by the Controller (PHY and Link layers) providing AES-CCM encryption and the Host (upper layers) providing device authentication, encryption, message integrity, pairing and bonding (key management) [13].

The biggest vulnerability of IoT communications is the metadata [27]. Table 4.1 demonstrates that existing IoT communication protocols do not address metadata vulnerability. Black networks mitigate metadata vulnerability by encrypting the entire PDU with fixed-length packets of the maximum frame size.

The IEEE 802.15.4 protocol defines a PHY and MAC sublayer, forms the basis for multiple higher layer protocols - ZigBee, 6LoWPAN and WirelessHART [9] [4] [63], being the most widely-used. The BLE communication protocol is based on a series of iterative standards Bluetooth 4.0 and above [13].

Our challenge, in IoT networks, is to mitigate internal and external threats, a range of active and passive attacks, and secure the communications per-hop and end-to-end. In this paper, we focus on securing the IoT communications protocol, at each layer of the protocol stack. We assume that in wireless communications, the sender is always visible to an adversary. Our goal is to ensure that mission-critical IoT communications have built-in

confidentiality, integrity, message authentication and privacy. The resulting communications, must be compatible with existing IoT protocols. To prevent inference and packet-length based attacks, we communicate using fixed length packets (the maximum size allowed by the IoT protocol - eg. 127 bytes in IEEE 802.15.4). Finally, our goal is to mitigate insider threats. A malicious node, or an intruder within the network (commonly referred to as an insider threat), must not be capable of deciphering a message that is not intended for it. This is achieved by allocating a unique symmetric key for each IoT node.

To achieve the above security objectives, we introduce Black Networks - where the metadata AND the data are encrypted for every frame/packet/segment at each layer of the communications stack. AES in the EAX mode (or Grain-128a) is the preferred cipher (to maintain payload efficiency). Securing the metadata leads to routing challenges, as the metadata contains source and destination information. Flooding and Broadcast over IoT networks, are ineffective, when the nodes sleep a majority of the time (a common energy-saving mechanism for IoT nodes).

The secured ZigBee NWK data PDU has an additional 112 bits of Auxiliary Header, which reduces the payload to a 528 bits-568 bits range. The 40-bit difference is the Auxiliary Security Header of the IEEE 802.15.4 frame. Therefore, if the 802.15.4 frame is also secured (recommended), then the secured ZigBee NWK Data PDU payload is 528 bits. If the 802.15.4 frame is not secured, then the secured ZigBee NWK Data PDU is 568 bits (Fig 4.1). Fig 4.1 also shows the transformation of an IEEE 802.15.4 frame to a Black frame [14], and the transformation of a ZigBee network PDU into a Black ZigBee packet.

The ZigBee network layer data packet header contains 16 bits of frame control information. The first subfield in the frame control field is Frame Type (data, control or command). The Frame Type reserved bits $b_0 b_1 = 11$ indicate a Black ZigBee packet. Excepting the first 2 bits, the remainder of the packet is encrypted using an authenticating cipher, such as AES-EAX (or Grain-128a) [67] [74]. An 80-bit Initialization Vector (IV) is included with each packet [75].

Figure 4.2 shows the transformation of a BLE Data PDU into a Black BLE Data PDU (with the LLID = 00 to indicate a Black BLE Data PDU). Black BLE Data PDUs have a

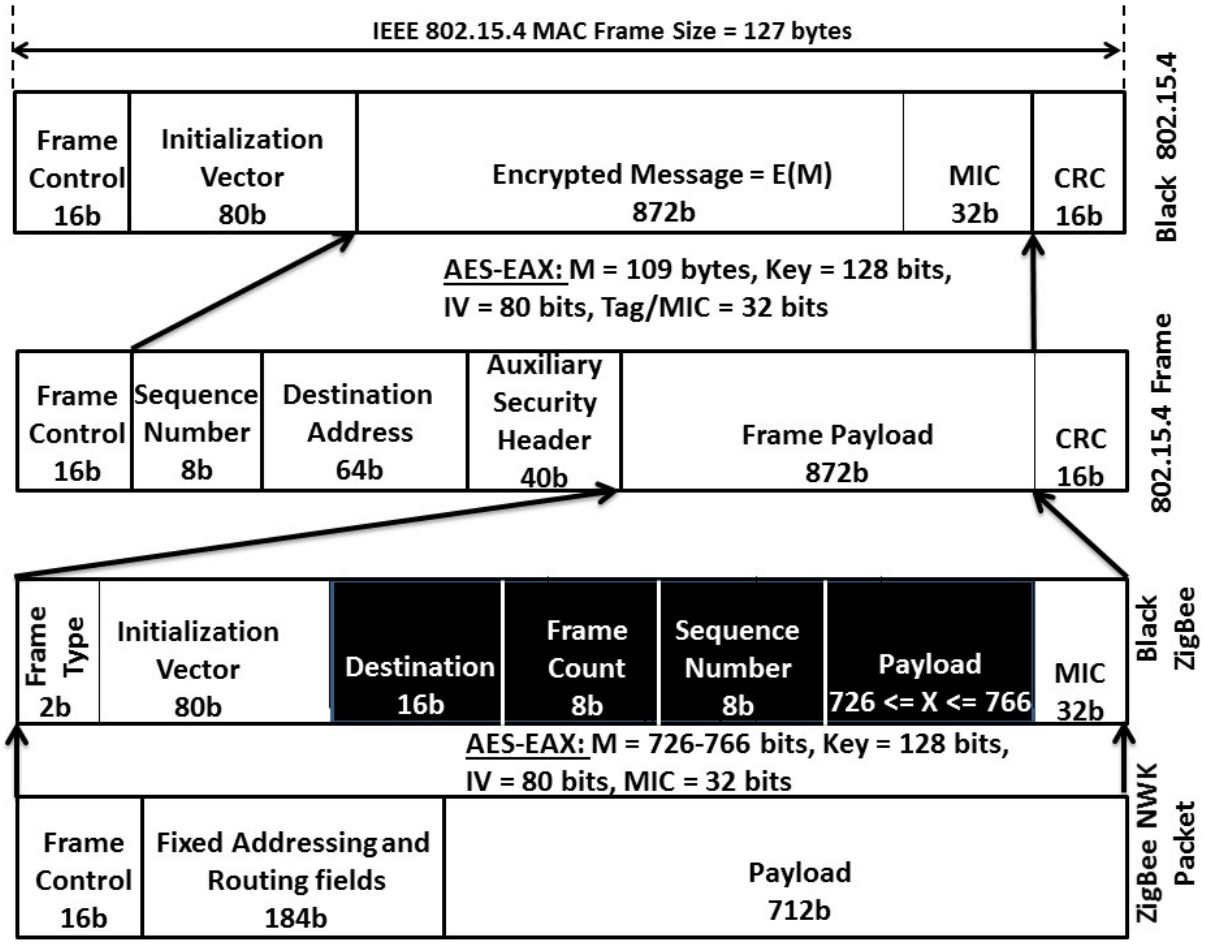


Figure 4.1: Black ZigBee packet and Black 802.15.4 frame

fixed size of 257 bytes with a payload of upto 251 bytes. Black BLE Advertising PDUs have a fixed size of 39 bytes with payloads ranging from 8 bytes to 39 bytes (with the PDU Type = 0111 to indicate a Black BLE Advertising PDU) [13].

4.3 Simple Black Network Communications

Black communications is a mechanism for delivering Black packets from source to destination, in an IoT network. Simple Black communications were evaluated for - Flooding, Broadcast and Gateway.

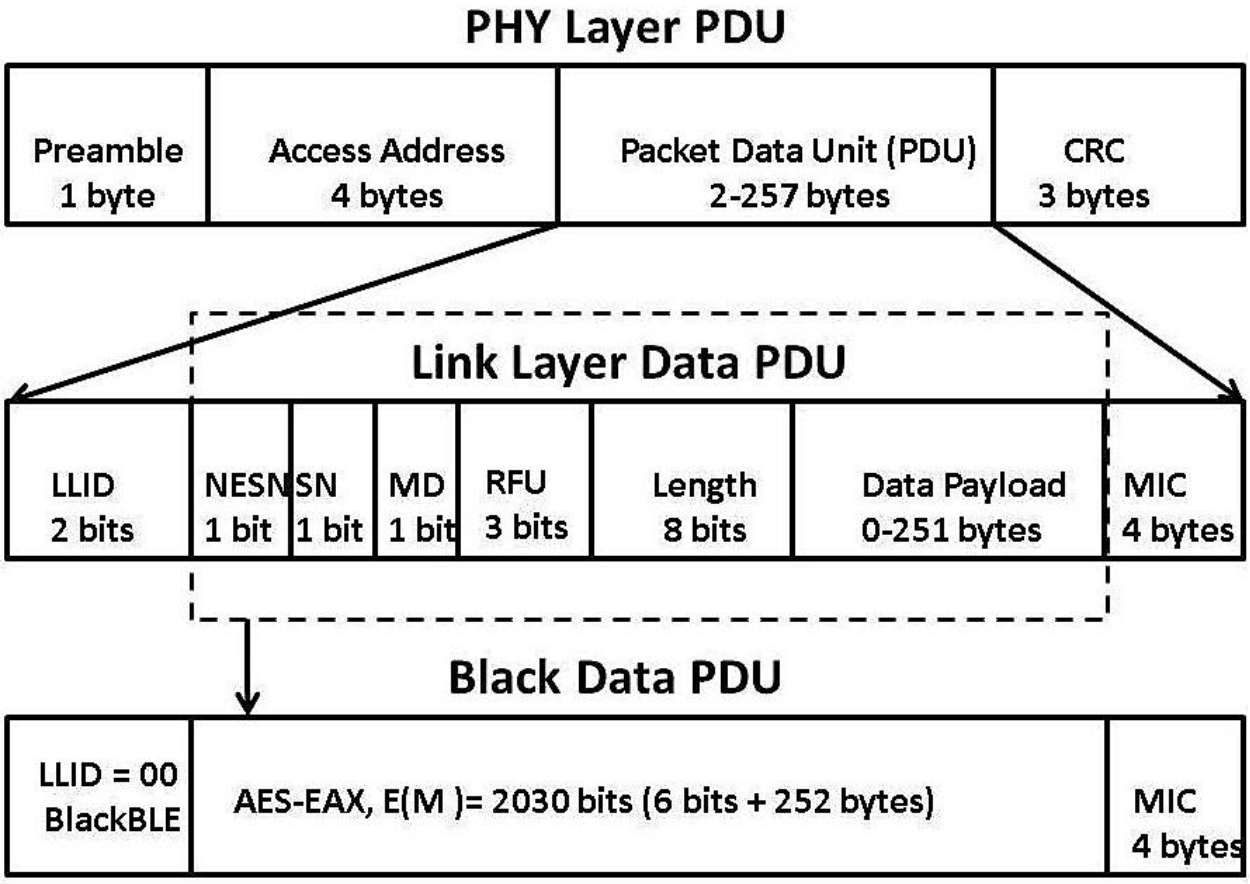


Figure 4.2: BLE Data PDU encryption to Black BLE Data PDU

4.3.1 Flooding

Flooding is a basic network routing algorithm which sends an incoming, or originating, packet to all outgoing routes, except for the one in which the packet arrived in. It means sending an incoming packet to all its neighbors. The process repeats until the destination is reached. For each node that gets the packet, a Message Integrity Check (MIC) determines if the Black packet is destined for the node. If the MIC fails, Flooding continues. If the MIC passes, the Black packet has reached its destination and that node does not continue re-transmission, but the other nodes do, until a hop count reaches zero, or the nodes run out of neighbors to re-transmit to.

4.3.2 Broadcast Communications

Broadcast refers to a node communicating to every other node in a network. A source communicates with a destination (along with every other network node) via a broadcast. The intended destination is able to pass the MIC and decrypt the Black packet, whereas the rest of the nodes are unable to pass the MIC and discard the packet.

Flooding and Broadcast have minimal intelligence for packet delivery in the network. This presents a problem, if all IoT node keys are unique. How does source create a Black packet that is readable by the destination, in large networks? Flooding and Broadcast have minimal intelligence in transmissions, and are inefficient for IoT networks where the nodes sleep a majority of the time (95-99%). If a packet arrives, when a node is in its sleep cycle, the packet is lost, as the node does not receive the packet, or retransmit it.

4.3.3 Black Gateway in a Star Network

A Black Gateway (BG) in a Star network configuration is a trusted third party (TTP) - that can process Black packets. The Black Gateway is directly connected to every network node. A star network configuration is one of the most commonly deployed topologies for IoT networks. The BG receives Black packets from the IoT nodes, and forwards these packets to their final destination. The BG manages ALL the unique, symmetric node keys, where k_I is the key for Node *I*. ALL traffic between the BG and IoT nodes are in Black packets. Black communications between Node S (source) and Node D (destination) occurs when Node S creates a Black packet using key k_S, and sends it to the BG. The BG does an exhaustive search on all keys until it passes the MIC. The BG decrypts the Black packet with key k_S and re-encrypts the Black packet using the destination key k_D of Node D. The BG sends the Black packet to Node D. When Node D, receives the Black packet, the unique symmetric node key (key k_D) decrypts the Black packet. The BG has no resource restrictions (either compute, memory or network). Even for thousands of nodes, an exhaustive search through the BG key space can be performed with negligible performance impact.

4.4 Black Network Simulations

We simulate Flooding, Broadcast and Gateway communications, using a Barabási graph, with a discrete-time network simulator which models Black network communications. Our

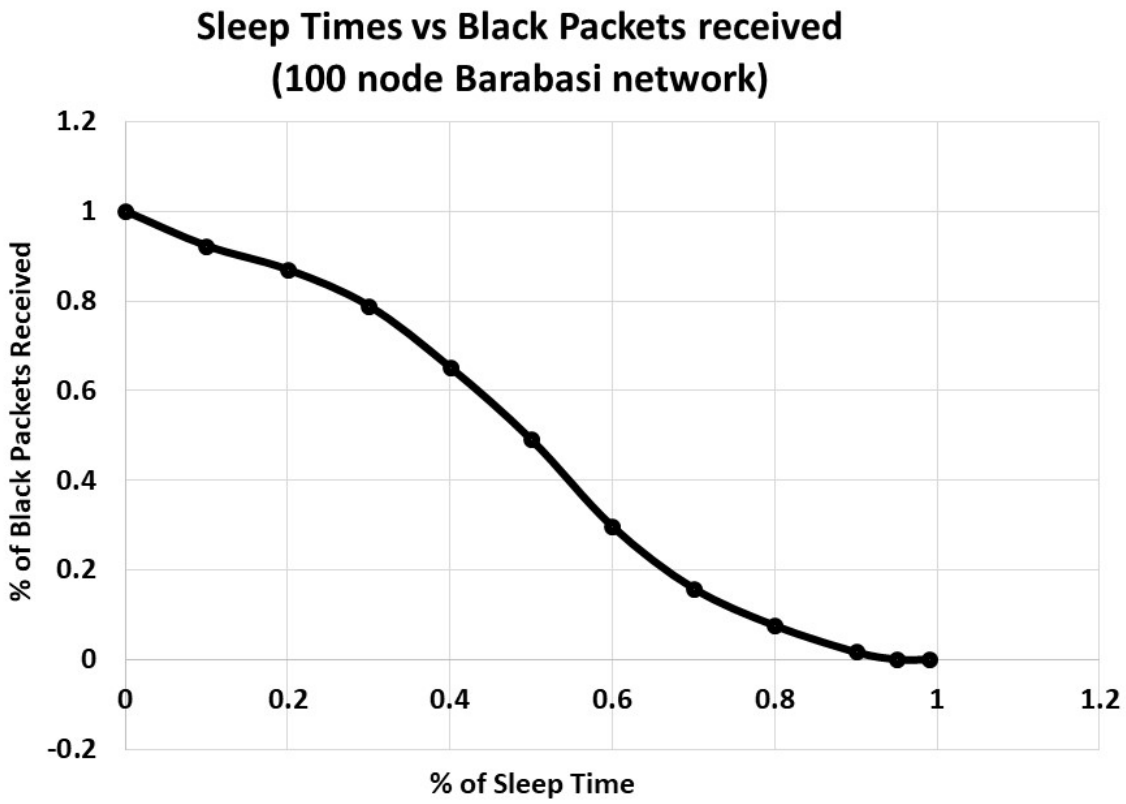


Figure 4.3: Black packet delivery probability to IoT sleeping nodes

simulator defines discrete time intervals such that any node sends a single packet per time interval. Medium access control concerns such as collisions and acknowledgments are assumed to be handled within time slots, and are not simulated. We restrict all network communications to Black packets. Traffic is randomly generated across the simulated Barabási graph network, and the inter-arrival times between new packets follow an exponential distribution with

$$T_t = e^{-\lambda * T_{(t-1)}}, \lambda = 2.0$$

Broadcast communications is simulated by sending all packets through a pre-computed arbitrary minimal spanning tree of the network. Since routing efficiency is spanning tree

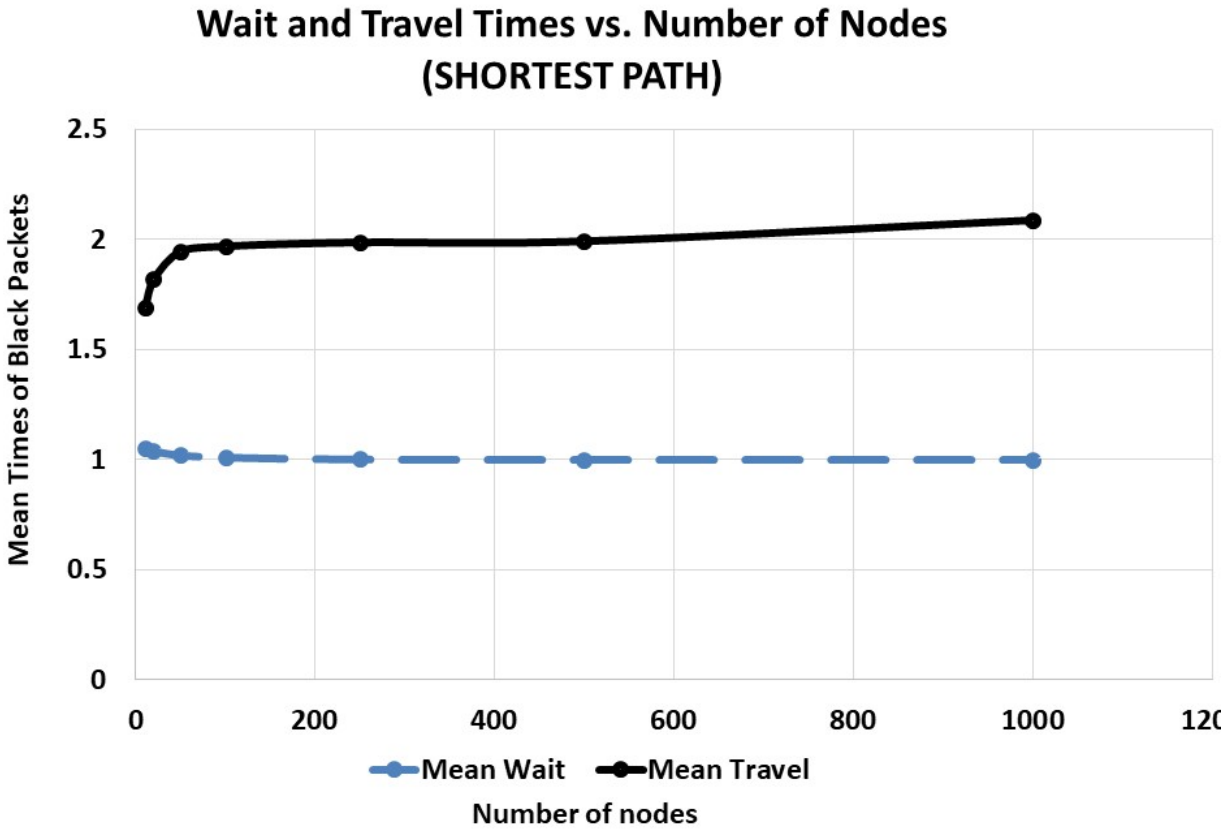


Figure 4.4: Shortest Path Simulations for Black Networks

dependent, a set of 10 spanning trees is generated, and results are averaged across 10 simulations, 1 for each potential spanning tree. Each simulation is run 10,000 times. Therefore each datapoint in the graph is a result of 100,000 runs of our discrete event simulator. Flooding is simulated by forwarding Black packets to all neighbors connected to the node generating the data, until the destination is found [87]. Nodes maintain a set of packets which they have flooded to avoid repeatedly flooding the same packet. Communications are simulated for source-destination pairs only with no multicast. Our discrete event simulator models a Barabási graph network with 10, 20, 50, 100, 250, 500, and 1000 nodes. In these IoT networks of increasing scale, we measure the following:

4.4.1 Wait time

The time interval between Black packet generation and transmission from the originating node. It provides a measure of the traffic in the IoT network, assuming a single FIFO queue in the originating node, that processes both the inter-nodal traffic (traffic arriving from other nodes), as well as originating traffic.

4.4.2 Travel time

The time taken for a Black packet to travel from source to destination, via intermediate nodes. It provides a measure of the network delay for the communication mechanism.

4.4.3 Traffic overhead

Traffic overhead constitutes of duplicate Black packets, for Flooding and Broadcast communications, and control packets for all types of communications. Traffic overhead is a performance indicator of network efficiency for each topology, communication mechanism and network scale. It is measured as the number of Black packets generated for each source-destination delivered Black packet.

Mean Wait Time, the Mean Travel Time and the Traffic overhead are simulated for Flooding, Broadcast and Gateway communications. For bench-marking, we simulate performance in a simple shortest-path routing. IoT networks have nodes that are in sleep cycles for a majority of the time. Packets arriving during the sleep cycle, do not get processed or re-transmitted. We simulate node reachability for Flooding in a 100-node IoT network where the nodes go into sleep cycles.

4.4.4 Node Reachability in Black networks

Figure 4.3 shows the delivery probability of Black packets, to destination nodes, when nodes go into sleep cycles to conserve power. Stringent power requirements often require IoT nodes to sleep 99% of the time. Simulations show that for a 100-node Barabási IoT network, there is a rapid decline in node reachability when the IoT node sleeps for more than 60% of the time. IoT networks are typically non-synchronized, and nodes have random sleep/wake cycles. If ALL IoT nodes were synchronized, to wake at the same time, then node reachability is guaranteed for our model IoT networks. For the remaining simulations

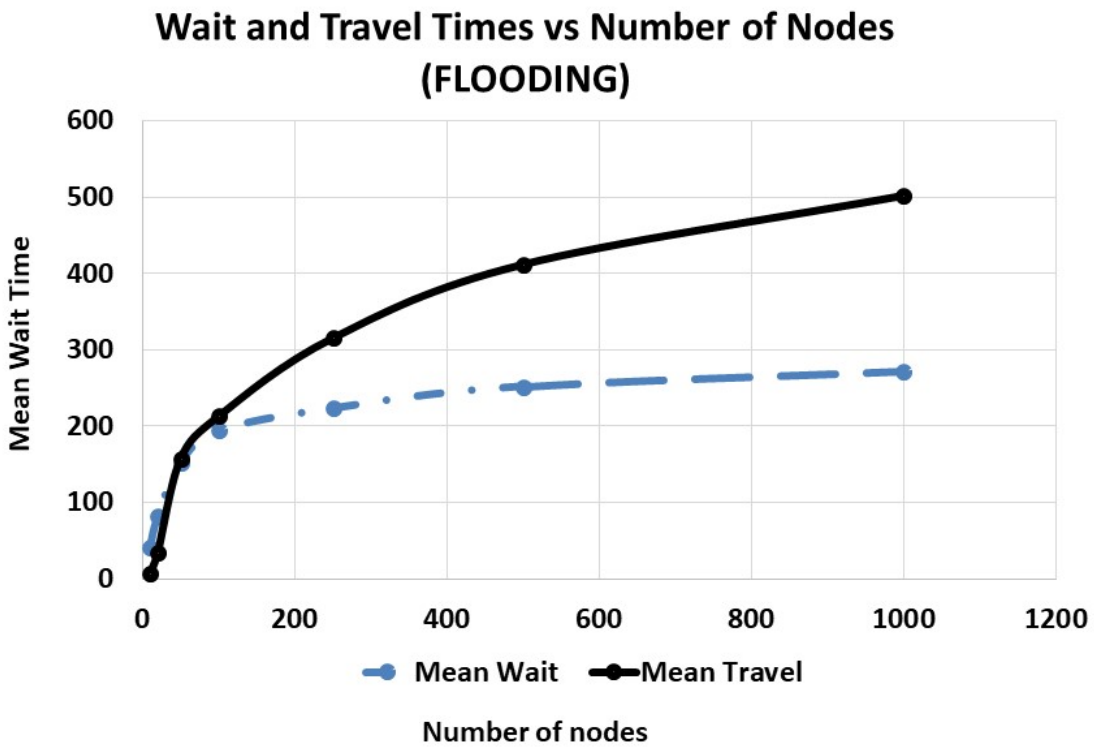


Figure 4.5: Flooding Simulations in Black Networks

of Flooding, Broadcast and Gateway, we assume that the nodes are awake all the time.

4.4.5 Simulations for Packet Delay and Congestion

We simulate Wait time, Travel time and Traffic overhead, for Flooding, Broadcast and Gateway communications, for a set of Black networks of 10 to 1000 nodes.

Figure 4.4 shows the simulations for Mean Wait time and Mean Travel time for IoT networks in ***Shortest Path routing***, which has no duplicate packet overhead and minimal control packet overhead. The Mean Wait Time is a constant at 1 time unit (TU), where Black packets are generated and sent from a network of any size. For smaller networks (less than 50 nodes), the Mean Travel Time is faster as is to be expected. As the number of nodes

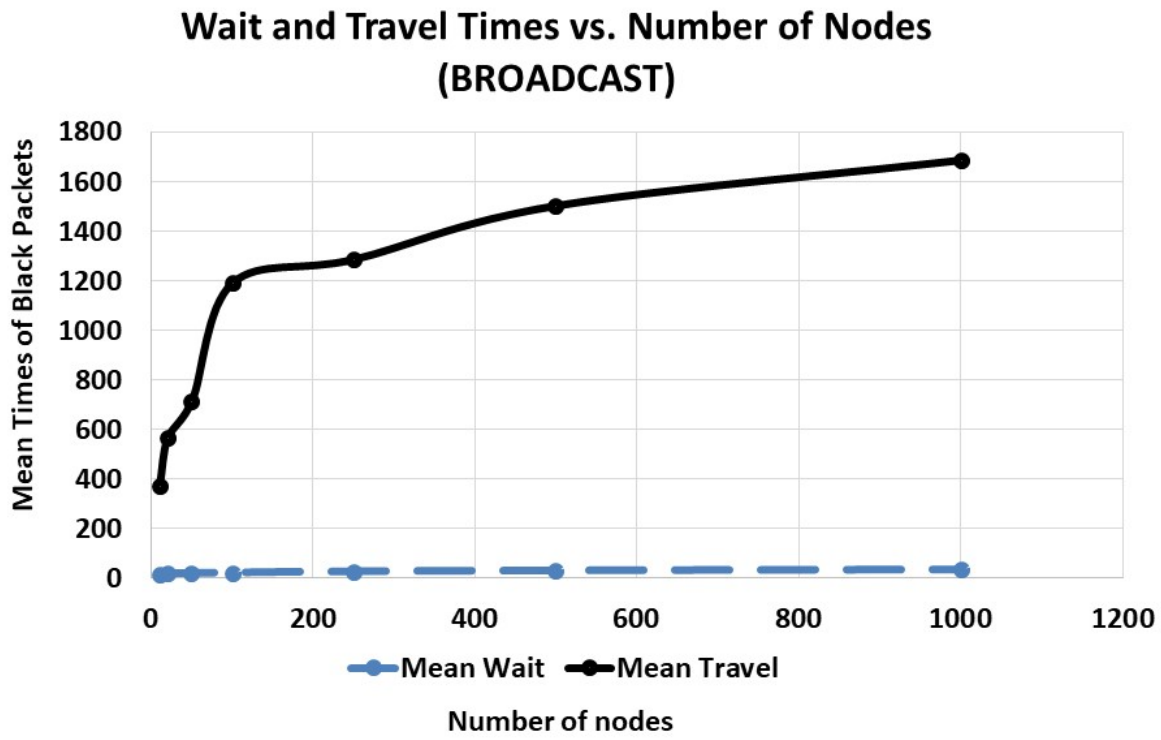


Figure 4.6: Broadcast Simulations in Black Networks

grows (from 100 to 800 nodes), there is mild congestion in the network and it takes 2 TUs to reach the destination. As nodes increase, there is slight further increase in Shortest Path congestion. For all practical purposes, Shortest Path Mean Wait Time is 1 TU and Mean Travel time is 2 TUs for our control set.

Figure 4.5 shows the Mean Wait and Mean Travel times for IoT Black networks using ***Flooding*** communications. We optimize Flooding by holding a Black packet, if it has been previously flooded. We observe that there is exponential growth in Mean Wait time from 10 to a 100 nodes. Between 100 to 1000 nodes, the Mean Wait Time increase is linear, but stable (from 213 TUs to 270 TUs). However, Mean Travel time continues to grow exponentially for all simulated Black networks. Flooding generates significant overhead (duplicate packets and

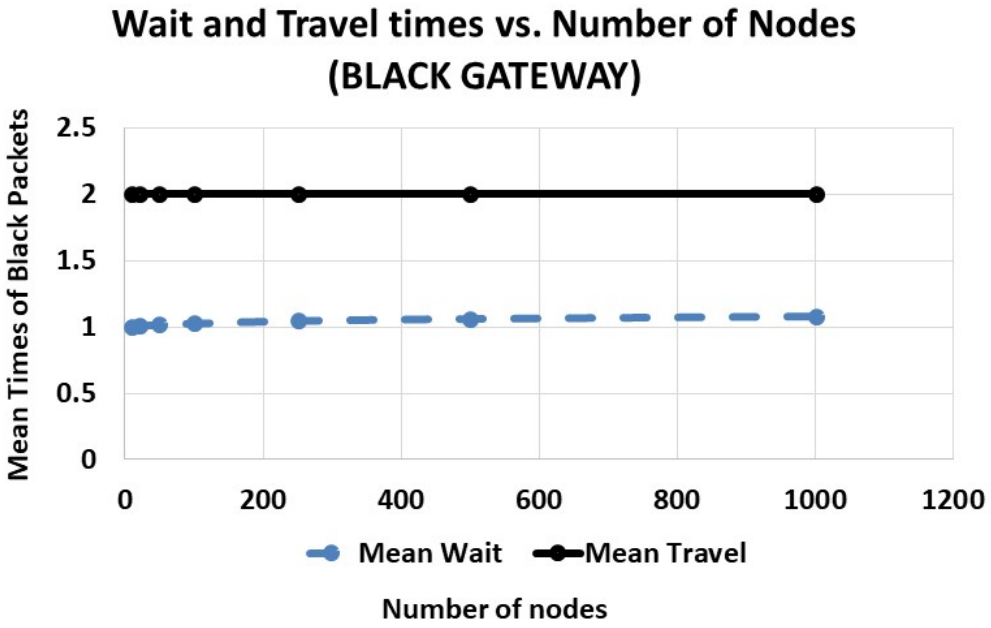


Figure 4.7: Gateway Simulations in Black Networks with Star Topology

control packets) for each delivered Black packet, leading to network congestion and delay.

Figure 4.6 shows the Mean Wait and Mean Travel times for ***Broadcast communications***. The Mean Wait time is negligible compared to the Mean Travel time. Each node sends the Black packet to ALL network nodes. The nodal traffic is an order of magnitude greater than the benchmark Shortest Path. Mean Wait time for Broadcast communications exhibits a slow linear increase as the number of nodes increases (from 14 to 34 TUs), but the Mean Travel time grows exponentially (from 372 to 1687 TUs), as nodes generate Black packets creating network delays for Black networks of increasing size.

Figure 4.7 displays the Mean Wait and Mean Travel times for the ***Black Gateway (BG)***. The BG Star Network is a 2 hop configuration for ALL source and destination pairs,

with the BG receiving and sending ALL Black packets. The Mean Wait time is a constant 1 TU. The Mean Travel time is also a constant at 2 TUs, regardless of the network size. The BG does not have any resource constraints or congestion delays. There is no duplicate traffic, and negligible control overhead, in Black networks of increasing size. Figure 4.8 displays the ***Traffic Overhead*** for each type of Black communications - Shortest Path, Flooding, Broadcast and Gateway. Shortest Path and BG have no duplicate Black packets and low control traffic. In Flooding, the traffic overhead, per delivered Black packet, is $O(n^2)$. Broadcast communications traffic overhead is lower by a constant, but the same order of magnitude as Flooding.

4.5 Evaluation and Analysis

Black networks provide enhanced security and privacy, while remaining compatible with existing IoT protocols. Black networks communicate using Black packets which encrypt the data AND the metadata. In Section 4.4, basic Black communications were simulated for Flooding, Broadcast and a Black Gateway. We analyze the results of Black communications in these networks.

4.5.1 How is Black Communications Different?

Black communications encrypt the entire packet (data and metadata) into a fixed length PDU (the length is limited by the Link layer maximum packet size - 127 bytes for IEEE 802.15.4 IoT protocols) for ALL communications. While such a format mitigates metadata, packet length, inference and insider attacks, a significant challenge exists in routing Black packets from source to destination. For Flooding and Broadcast in Black networks, the source node encrypts the Black packet using a symmetric key with the destination node, and forwards the Black packet. For resource-constrained IoT networks, it implies that the source and destination share a symmetric key. Public keys are not suitable for resource-constrained IoT networks. For a small number of nodes, it is feasible that each IoT node has a map of a unique key for each IoT network node. However, as the number of nodes increases, the symmetric key list, and its update to each node becomes prohibitively expensive in resource-constrained IoT networks. Moreover, holding keys in a resource-constrained node, has a significant security risk of node capture and other attacks. As the Black packet traverses

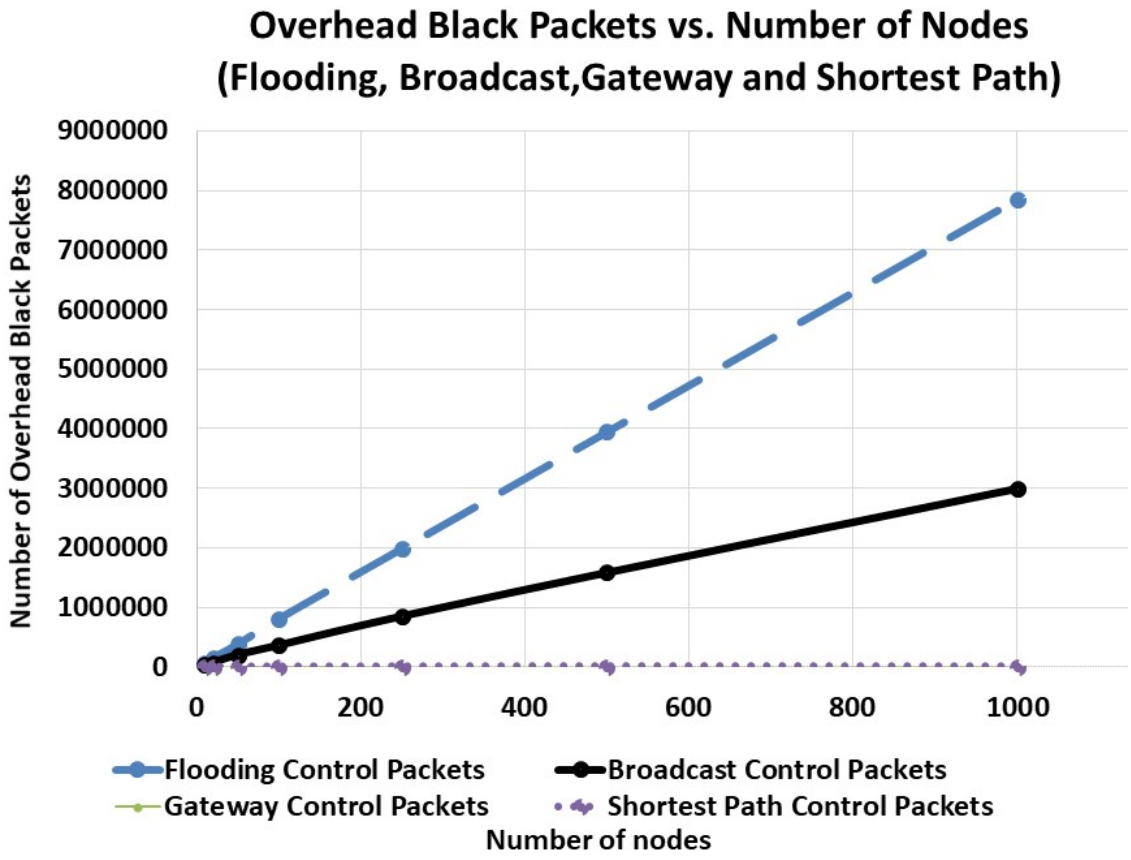


Figure 4.8: Traffic Overhead in Black Networks

through our Barabási network, there are two operations at each node: 1) A Message Integrity Check (MIC); 2) A forward or a decrypt. If the MIC passes, then the Black packet has reached its destination and is decrypted, else the Black packet is forwarded. Black networks are not limited to IoT, and may be applied to ALL communication protocols [76].

4.5.2 Performance Analysis

From the simulations in Section 4.4, we note that the most optimal network performance solution, for Black networks, is the Black Gateway (BG), in a Star configuration. The BG manages network keys and switches all the Black traffic from source to destination. There is a bounded path-limit of 2 hops for every source-destination path, as the BG is connected

to every node. Regardless of the number of nodes (we simulate for up to 1000 nodes), there is negligible overhead traffic and no duplicate packets (Fig. 4.7). The network performance characteristics are almost equivalent to the Shortest Path benchmark used (and slightly better because of a simpler network topology). For simple Black communications, there is no acknowledgement handling. A Star network configuration is a very commonly used topology for IoT networks.

The Shortest Path (SP) algorithm (our benchmark) is an optimal solution in networks as the most efficient path from source to destination. SP is efficient in Mean Travel (costing approximately 2 TUs from 50 to 1000 nodes) and Mean Wait (1 TU with negligible congestion and no duplicates) across multiple networks (Fig. 4.4).

Broadcast communications has the highest Mean Travel time which grows exponentially. In Broadcast communications Black packets take the spanning tree and not the shortest path, taking a more expensive path from source to destination. Simulations show that Broadcast communications are an order of magnitude longer than Flooding for smaller Black networks ($\leq$100), and over 2 orders of magnitude longer compared to BG and SP. For Mean Wait times, Flooding is an order of magnitude longer than Broadcast, as the number of nodes increase (>100 nodes, Fig. 4.5, Fig. 4.6).

Overhead traffic has a significant effect on Black network performance and consists of duplicate Black packets and control Black packets. The BG and SP configurations have negligible overhead traffic for delivered Black packets from source to destination. Flooding and Broadcast communications have significant traffic overhead (duplicate Black packets, upto 3 orders of magnitude greater than BG and SP) for delivered Black packets (Fig.4.8), which is the primary reason for exponential Mean Travel and Mean Wait times.

Table 4.2 shows a sample of actual numbers for Mean Travel, Mean Wait, and Overhead for Flooding, Broadcast, BG and SP communications for our median 100-node Barabási Black network simulation.

4.5.3 Security

Metadata attacks, with no mitigation mechanisms in IoT protocols (Table 4.1), are addressed with Black networks. Encrypting the metadata results in privacy between commu-

Table 4.2: Black Networks Simulation Numbers for 100 nodes

	Flooding	**Broadcast**	**BG**	**SP**
Mean Wait	194 TU	22 TU	1 TU	1 TU
Mean Travel	213 TU	1192 TU	2 TU	2 TU
Overhead	8.1x10^5p	3.6x10^5p	6065p	5851p
Black PDU	1E+n*M+1D	1E+N*M+1D	1E+1D	1E+n*M+1D
Other PDU	1E+1D	1E+N*D	1E+1D	1E+1D
ZigBee Payload		*Black ZigBee*	*Gain*	*% Gain*
Normal	712 b	766 b	54 b	7.6%
Secure	568 b	766 b	198 b	34.9.0%
BLE Payload		*Black BLE*	*Gain*	*% Gain*
Average	130 B	257 B	255 B	97.7%
Maximum	257 B	257 B	0 B	0.0%

n = Number of hops, E = Encryption, D = Decryption, N = Total nodes, M = MIC, p=packets, TU=time units, b=bits, B=bytes

nicating entities in Black networks, a security mechanism not implemented in IoT protocols. Black networks provides confidentiality, authentication (AES-EAX is an authenticating cipher) and privacy by default, at every level in the communications protocol. Frame counters which can be modified for replay attacks are encrypted in Black networks. Black networks are resilient to IP-based attacks which target 6LoWPAN networks, such as HELLO flood, sinkhole, Sybil and selective-forwarding. Black BLE does not require random address generation, resolution and lookup tables [13] like BLEs existing privacy feature. Inference and packet-length based attacks are mitigated in Black networks by using fixed-length Black packets for ALL communications. Insider threats are mitigated by each node having its own unique key. A malicious node will not have a usable key to communicate with the BG or

other nodes for insider attacks. Broadcast communications in Black networks do not divulge the destination, as destination addresses are secured and packets are sent to every IoT node.

4.5.4 Black Network Compute Efficiency

The Black Gateway (BG) is not resource-constrained, so we do not account for its compute operations. In the BG Star Network configuration, the source performs 1 encryption and the destination performs 1 decryption. For Flooding, Broadcast and Shortest Path, the source performs 1 encryption, and forwards the Black packet. The intermediate nodes perform a MIC. The destination will perform a MIC and 1 decryption (Table 4.2).

4.5.5 Black Packet Payload Efficiency

The ZigBee Secured NWK PDU has a 568 bit payload, reduced to 528 bits, with a Secure IEEE 802.15.4 frame. A regular ZigBee payload is 712 bits. In comparison, a Black ZigBee NWK data PDU has a 766 bit payload (see Fig 4.1). A Black ZigBee PDU carries a payload that is between 8% to 35% greater than a ZigBee PDU, with significantly improved security. Similarly, fixed-length Black BLE Data payloads can vary between 0-12750% larger, with the average Black BLE being 98% greater than BLE payloads [13] (Fig. 4.2, Table 4.2).

4.6 Conclusions and Future Work

IoT networks are increasingly carrying mission-critical data, and security for the Internet of Things (IoT) is of paramount importance, in the wake of multiple high-profile attacks. Black Networks secure IoT communications, by encrypting BOTH the data AND the metadata, in fixed-length packets, to mitigate a broad range of active, passive, packet length-based and insider attacks. Our analysis indicates that the improved security and privacy of Black IoT networks comes at a performance and compute cost overhead for simple Flooding and Broadcast communications, with no intelligence. With a Black Gateway in a star network topology, Black IoT networks closely match Shortest Path routing efficiencies. Therefore, Black Networks are a practical approach to securing metadata in IoT communications. Future research should extend to Black Routing in IoT Mesh networks and to non-IoT protocols. Mobility, High availability, Node Authentication, Key Management and Sleep times for IoT Black networks are open areas of research.

Part II

Black Routing

Chapter 5

NETWORK ARCHITECTURES

In Chap 3, we presented Black packets and Black packet design for various IoT communication protocols. In Chapter 4, we presented simple Black communications, using the Black packets, evaluating several measures of security and performance (overhead traffic, mean wait times, mean travel times, payload efficiency and compute efficiency). In this chapter we present the network support needed to enable Black routing in mesh networks. We present an SDN-based architecture for Black IoT networks, that guarantees node reachability, even when the nodes are asleep a majority of the time. Various configurations are presented and the simulated results are presented.

5.1 Introduction

We present Black SDN, a secure SDN IoT network architecture that utilizes an SDN controller as the trusted-third party in the Black Network. The primary goal of Black SDN is to secure communications by encrypting the header and the payload at the Network layer to mitigate a range of attacks, including traffic analysis/inference attacks. Header encryption causes routing challenges. We propose a simple broadcast routing, and a more efficient and secure SDN routing An SDN architecture improves IoT network security and efficiency for Black Networks. All approaches must consider asynchronous node 'sleep' and 'wake' cycles. We simulate Black SDN for IoT in star and mesh topologies and evaluate the results.

The major contributions of this chapter are: introduction of SDN-based IoT networks, the concepts and requirements for Black broadcast routing and Black Routing with a Trusted Third Party (TTP) and the SDN as the TTP.

The remainder of this chapter is organized as follows: We present Black SDN for IoT networks in Section 5.3, in three common configurations of star and mesh networks, with the SDN controller as the Trusted Third Party (TTP) for both broadcast and routing of

Black packets. We evaluate the security of Black SDN in Section 5.4 and analyze the results of our simulations. We draw the relevant conclusions and suggest future areas of research in Section 5.5.

5.2 SDN Controller: The Need for a Trusted Third Party

In Chapter 1, we outlined the security challenges, and vulnerabilities for IoT communication protocols - 802.15.4, 6LoWPAN, ZigBee and WirelessHART. SDN networks and the OpenFlow protocol [88] have similar challenges for IoT networks. We note that security contributions in the emerging field of SDN IoT networks are limited. In this section, we motivate and define the open problem of security for SDN IoT networks.

The general security problem we are addressing is: How does a packet get from node A to node B without an eavesdropper knowing that the packet went to node B?

The specific problem we are attempting to solve is: How does a packet get from node A to node B, *in an IoT network*, without an eavesdropper knowing that the packet went to node B? This translates to incorporating privacy within IoT network communications.

We resolve the privacy problem through Black Networks - where the data and the metadata of the frame, and packet, are secured at the Link and Network layers. This, however, presented a routing challenge in IoT networks, for peer-peer data transfer, where a packet may not reach its destination, if the intermediate nodes were asleep a majority of the time (which is a practical scenario).

The goal is to resolve the routing problem, with Black packets, in IoT networks. This means getting a Black packet from Node A to Node B, with intermediate nodes on their configured 'sleep'/'wake' schedule. The solution would have the added benefit of incorporating confidentiality, integrity and authentication for all communications. Additionally, it would mitigate a host of inference, traffic analysis, power depletion attacks and packet length-based attacks. As declared, it is not sufficient to receive the Black packet at its final destination, but also necessary to ensure that the receiving node (Node B) is unknown to an external observer.

We further outline our assumptions associated with the problem definition of routing in Black networks. They are:

- The operating environment is an IoT network consisting of low-power resource-constrained nodes in a mesh configuration. Heavy-duty protocols (such as IPsec, SSL/TLS) cannot be supported in this environment.
- When a node transmits, it is known. An adversary knows who transmitted data.
- A trusted third party (TTP) exists in the network, as an anchor, with a network topology view.
- Nodes operate in synchronous or asynchronous modes.
- TTP and nodes communicate via shared secret key

In Section 5.3, we present a solution for our problem definition with the above assumptions.

5.3 Black SDN for IoT Networks

Software Defined Networking (SDN) has been proposed to streamline network architecture, complexity and scalability [30]. SDN separates the control plane (signaling) from the data plane (media) in an IP network [6] [5], resulting in a scalable and very cost effective architecture. The routers (now called forwarding elements or switches) have minimal logic to forward data (and can be simple compute elements, without complex, expensive routing logic). The forwarding decisions are based on Flow Tables that are downloaded to the nodes by a centralized SDN controller, with a global network view. The controller communicates with the switches using an open, industry-defined, protocol called OpenFlow [89]. All other functions - protocols, middle-box functionality and network management and configuration reside in the SDN controller SDN security has been a major concern for potential adoptees. SDN security vulnerabilities are assessed to be within the seven areas in the network [90]. The OpenFlow standard describes basic TLS for controller-node security, but does not mandate it [91]. TLS, even with certificate-based authority, has well-known vulnerabilities. All of the scenarios have been proposed for large, broadband IP networks (enterprise, data centers and service providers).

IoT networks with an SDN architecture (Figure 5.1) have to contend with additional vulnerabilities of resource-constrained nodes, operating in a low-power WSN environment, with

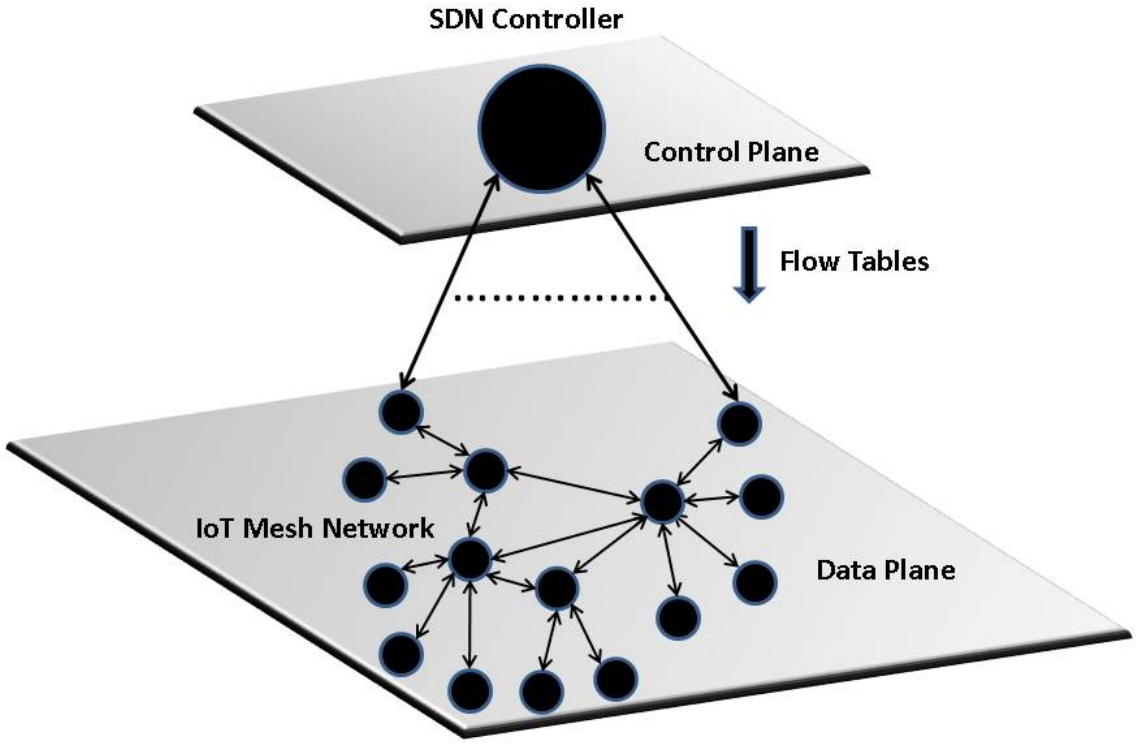


Figure 5.1: IoT Network with an SDN Controller

all the vulnerabilities outlined in Table 1.1. SDN for IoT is an architectural approach that incorporates an SDN controller in an IoT network. Resource-constrained IoT nodes cannot support a full SDN implementation - reserved for large, complex, broadband, IP networks. Adaptations of the SDN controller for IoT networks have been presented in [92] [31] [7] [93].

An architecture for SDN design to WSNs based on 802.15.4, with a simple flow table description, duty cycle handling and in-network data aggregation has been presented in [31]. An SDN protocol for WSNs, Sensor OpenFlow, based on the OpenFlow standard is presented by [7]. Sensor Openflow addresses some of the challenges with SDN applied to IoT such as in-situ data aggregation, simplicity of the flow tables, control plane communication between controller and forwarding element and minimizing the overhead of control channel traffic.

Both of these approaches consider a simple network with a single controller. However, [93], proposes SDN architecture for IoT networks within a complex domain (like smart cities), which has multiple IoT networks running heterogeneous mobile technologies. The resulting UbiFlow controller maintains partitions across multiple controllers to load-balance, guarantee performance, manage scalability and mobility. An SDN architecture for IoT, for heterogenous wireless networks with different classes of IoT traffic, on a single, layered IoT SDN controller is presented in [94].

The above SDN for IoT architectures and implementations have not focused on security. SDN for IoT security challenges are a combination of SDN, IoT and network security vulnerabilities. The security for SDN IoT networks are rudimentary and nascent, and highly secure SDN IoT Networks remains an open problem as defined in Section 5.2. In this section we present a highly secure Black SDN for IoT networks. This secure IoT network is enabled via an SDN controller (adapted for WSNs), and results in superior network performance, security and payload efficiency in star and mesh networks. We compare the results with non-SDN IoT networks.

The Black SDN for IoT consists of a star, or mesh, wireless IoT network that communicates with an IoT-adapted SDN controller. The SDN controller and the IoT nodes communicate via Black packets. An example of an SDN controller to IoT node *control* Black packet is shown in Figure 5.2. The fields are aligned for header fields, actions and logs/counters. Control Black packets from the IoT node to the SDN controller are identical in format. Header match fields could be Packet ID, Node ID and/or Network ID. The standard actions to act on a Black packet would be Forward, Drop, Modify (the data within the packet) or Sleep (for a given time period). The logs would include TTL (time-to-live) and Random (a small random value to forward Black packets, or rebroadcast them, to obfuscate the receiver). The Data field would contain neighbor lists, wake/sleep times and other parameters.

Using these minimal set of control parameters, we present and simulate three scenarios across topology (star or mesh), synchronization (synchronous or asynchronous) and transmission mode (broadcast or routing). In each case, we evaluate if the SDN controller is more effective for routing and security.

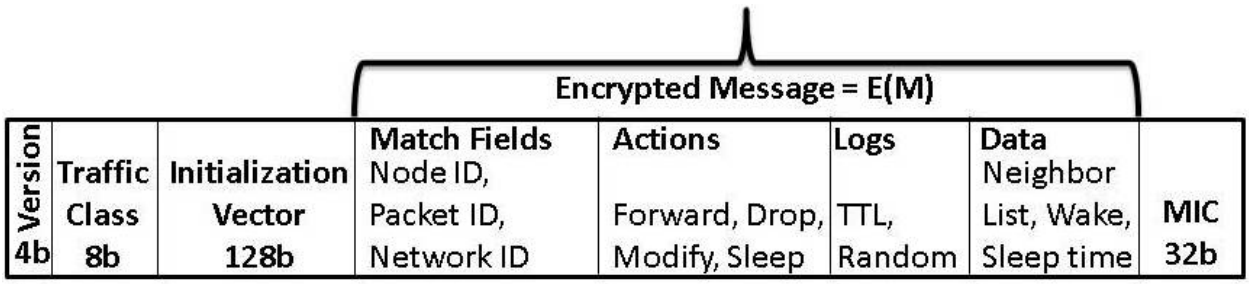


Figure 5.2: Black SDN for IoT Control Packet example

5.3.1 Scenario 1 - Broadcast on Star Network

Topology=Star, Sync = Y, Broadcast = Y, Controller = Y In this star topology, all nodes sleep and awake at the same time (are in sync), based on a controller initiated sleep/wake schedule and absolute time (clock). This is refreshed on a regular basis to eliminate timing drifts in the system. Assuming nodes are only within radio reach of controller (and not each other), inter-nodal communication occurs via SDN controller. Sending node broadcasts to controller, which in turn broadcasts to all nodes, including destination node. This ensures that the destination is obfuscated to an eavesdropper, even if the source is known (it is assumed that the transmitting source is known to an attacker). All nodes may re-broadcast the packet to further confuse the attacker as to whether the packet was accepted or rejected by receiver. In this scenario, the controller acts as a gateway.

If the nodes are within radio reach of each other, then a controller is not necessary. Nodes broadcast to each other, and then re-broadcast to mask the destination.

It must be noted that the overall system may not be secure for a small number of nodes (which is typical in a star network), and a statistical inference can be made on source and destination.

5.3.2 Scenario 2 - Synchronized Mesh Network

Topology=Mesh, Sync = Y, Route = Y, Controller = Y Like Scenario 1, in this mesh network all nodes are in controller-managed sync. The originator node, requests a route from the SDN controller, when it has to transmit. The SDN controller maps a route downloads

flow tables to the transmitting node and intermediate nodes. When nodes are in the wake cycle, the Black packet gets transmitted per hop.

The other option for routing the Black packet is via an onion router method [95] [25]. In this case, the SDN controller dynamically determines the next hop for the Black packet. At every wake cycle, the SDN controller having dynamically determined the next hop, downloads it to the current node storing the Black packet. This method is more secure and reliable than setting up an end-to-end path ahead of time. It also has a higher performance impact due to the control traffic generated during every wake cycle to all nodes (again to mask destination).

Figure 5.3 shows the simulated results of Black packet latency for Scenario 2, for a network path with upto 10 hops. Sleep times range from 0.5 ms to 10ms (approximately upto 95% sleep cycle).

5.3.3 Scenario 3 - Unsynchronized Mesh Networks

Topology=Mesh, Sync = N, Route = Y, Controller = Y Scenario 3 is the most challenging of all the scenarios. Sleep and wake cycles for the IoT nodes are not synchronized. Consequently, some nodes are asleep, while others are awake, in no particular order. Black packets transmitted to nodes that are sleeping, do not reach them. Consider Node A sending a Black packet to Node B. In this case, the SDN controller, based on its network map, downloads routes to the subset of nodes, that are adjacent to Node A and whose wake times overlap with Node A. Node A broadcasts the Black packet to these nodes and the process repeats until destination Node B. It is possible that NONE of Node A's adjacent nodes are awake during A's wake cycle. In which case, the SDN controller instructs Node A to sleep until the next hop awakes, and then the Black packet is transmitted. To eliminate such conditions, during operation, IoT network configurations should be managed accordingly. Node join requests must be initially populated with proper asymmetric cycle times, such that adjacent nodes have adequate overlapping awake times. This IoT network configuration should be done at the start when the nodes are joining the network. Figure 5.4 shows the simulated results of Black packet latency for Scenario 3, for a network path with upto 10 hops. Sleep times range from 1ms to 10ms. (approximately upto 90% sleep cycle, for

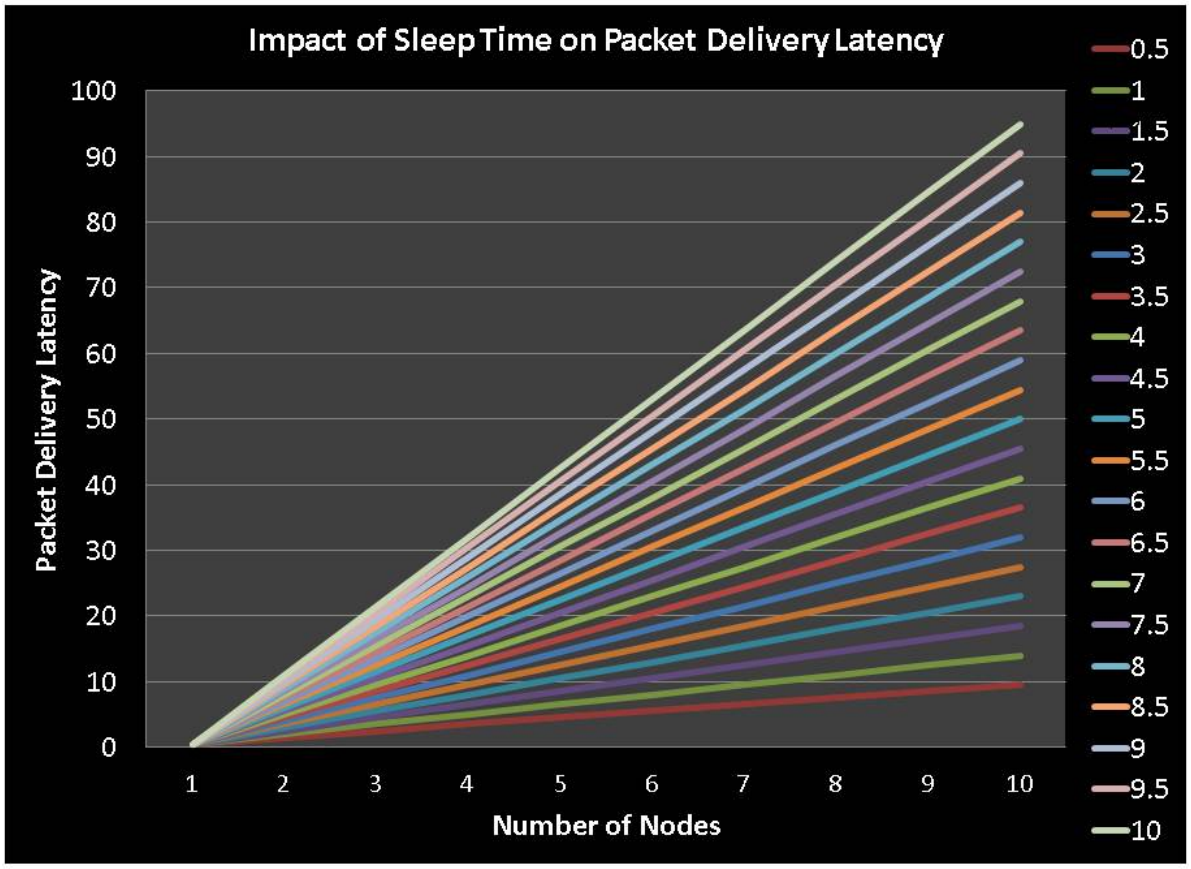


Figure 5.3: Scenario 2: Black SDN Packet Latency

asynchronous networks).

5.4 Evaluation and Analysis

Our motivation for presenting Black SDN for an IoT network is aimed towards enhanced security and network performance for mission-critical networks.

5.4.1 Network Performance

In Section 5.3 we demonstrate broadcast routing for Black Networks over multiple network topologies. Black Networks provide for a secure approach to communication by protecting each layer in the communication hierarchy - at at the expense of complicating the routing through the network. Simple broadcast provides for the most secure routing ap-

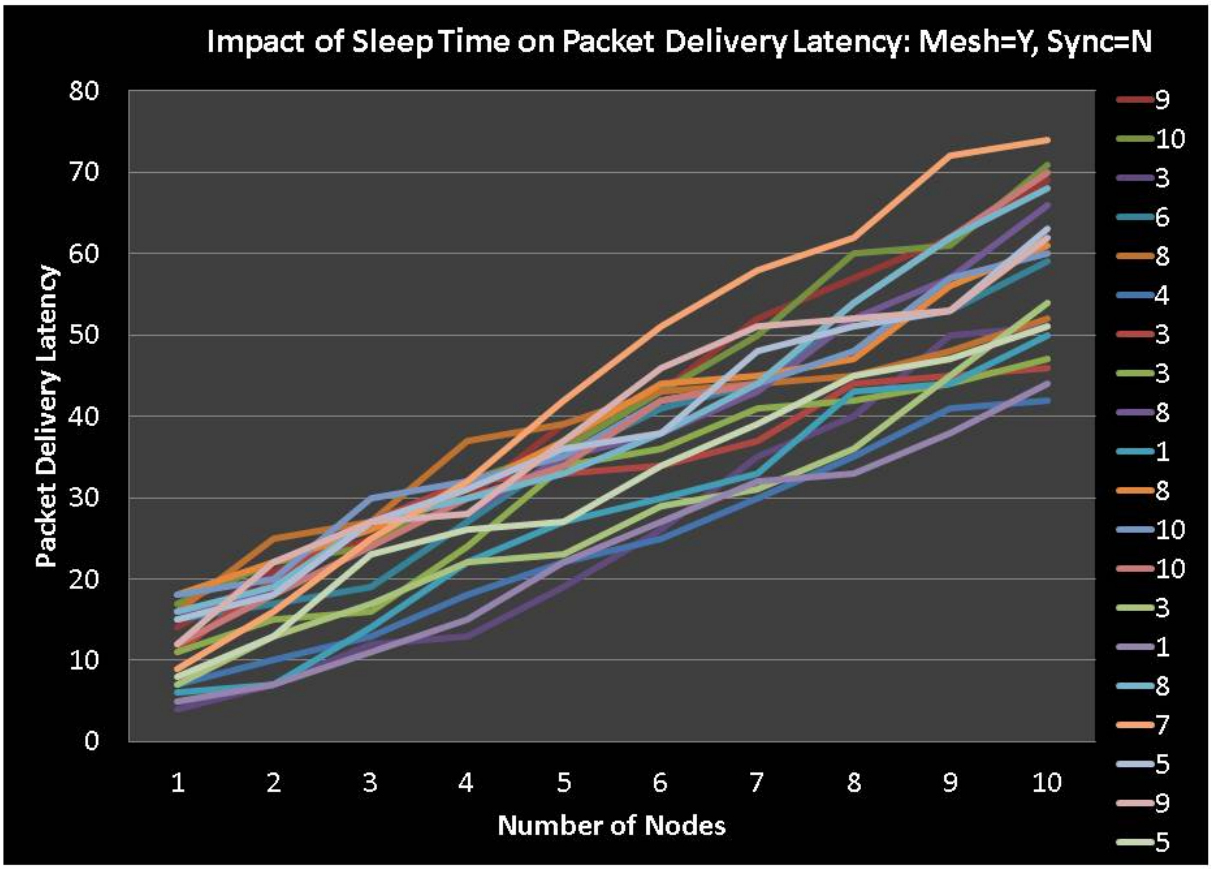


Figure 5.4: Scenario 2: Black SDN Packet Latency

proach, but it consumes significant energy across all nodes in the network. Furthermore, the network topology has a significant impact on the success of broadcast routing. Equal length paths between two nodes increase the likelihood of collisions, and limited numbers of paths between two nodes makes the network susceptible to becoming disjoint with both collisions and the use of sleep modes. Comparing with the Black SDN for IoT - we show that network performance is markedly better, as the SDN controller maintains the state of the network and its components. Black packet delivery - through either synchronization and sleep, reach their final destination, with latency, when nodes sleep a majority of the time. We note this for the star and the mesh topologies. One area of concern for Black SDN for IoT is the generation of control traffic as a result of increased communication between

the SDN controller and the IoT nodes. Further, the need to maintain anonymity results in additional messages being sent to obscure the recipient of the message. There is increased control messaging between SDN controller to IoT, for Scenario 2 (synchronous mesh), as the nodes wake and sleep at the same time. The possibility of message storms and the capacity handling ability of the SDN controller are areas of concern. With Scenario 3 (asynchronous mesh), the control messaging is lower, at the cost of increased latency in packet delivery.

5.4.2 Security

Black SDN provides a higher level of security than existing 802.15.4 protocols. Black SDN for IoT provides confidentiality, integrity, authentication and privacy. Table 5.1 compares the payload efficiency of IEEE 802.15.4 Link layer frame with an implicit nonce (nonce constructed out of header fields), to a Black frame. Black frames provide Privacy, Confidentiality, Integrity and Authenticity, by encrypting the header and 93 byte payload. We note that at higher IEEE 802.15.4 security levels (when both encryption and authentication are applied to the payload), Black frames provide equivalent, or better, payload capacities with a higher level of security (6% better vs. ENC-MIC-64 and 16% better vs. ENC-MIC-128). Unlike Black frames, the IEEE 802.15.4 options of no security, authentication only, and encryption only can lead to insecure implementations susceptible to a range of attacks as shown in Table 4.1. While inference attacks can be made on the 802.15.4 variable payload, the Black frame mitigates payload length-based attacks because of its fixed size.

Table 5.1: Payload efficiency of Black Frame.

Comparison of 802.15.4 Link layer with Black Frame		
Security Level	*802.15.4 Payload*	*Black Payload*
No Security	114 bytes	93 bytes
ENC-MIC-32	92 bytes	93 bytes
ENC-MIC-64	88 bytes	93 bytes
ENC-MIC-128	80 bytes	93 bytes
Encryption only	96 bytes	93 bytes

5.5 Conclusions and Future Research

Black SDN for IoT enables a secure Internet of Things. The Internet of Things will continue to grow and encompass all aspects of our lives. IEEE 802.15.4-based IoT devices will continue to play a significant part in IoT expansion. IoT devices are engaged in mission-critical functions in multiple industries. Current IoT protocols are vulnerable to a range of attacks including eavesdropping and packet injection attacks based upon the plain text metadata. Securing the communications between IoT devices, by encrypting both the data and the metadata, at the Link and Network layers prevents an additional range of attacks including eavesdropping, track and trace, packet injection, and packet modification attacks. A Black Network method of securing all data, provides for high security within a network, at the expense of symmetric key management, decreased network efficiency, and complicated routing. As networking paradigms shift to embrace Software Defined Networking (SDN) in enterprises and Service Providers, IoT networks will utilize the architecture to form the basis for a secure Internet of Things.

Simple broadcast provides for the most secure routing approach, but it consumes significant energy across all nodes in the network. Furthermore, the network topology has a significant impact on the success of broadcast routing. Equal length paths between two nodes increase the likelyhood of collisions, and limited numbers of paths between two nodes makes the network susceptible to becoming disjoint with both collisions and the use of sleep modes.

Future areas of research in Black Networks will focus on better routing mechanisms. These include developing sleep synchronization protocols that are appropriate for Black Networks in order to ensure packet delivery to all nodes. They also include routing for energy-efficient IoT nodes to minimize resource usage. Obfuscating the transmitting source, is an open problem for IoT network security. Another area of future research is to secure the Black Link layer frame by multiple methods that would allow for a fine-grain approach to securing the metadata such as, *a)* replacing the metadata fields by Grain-128a IV and a keystream, or *b)* using the AES-EAX mode and *c)* using a pre-shared IV to allow for better payload efficiency. Finally, extending Black Networks to non-IoT networks is needed, along with a standards initiative for secure IoT networks.

Chapter 6

BLACK ROUTING

In this chapter we combine Black packets and Black SDN for *routing* a Black packet from source to destination, in a mesh network. We present Black routing algorithms, for various configurations, and provide extensive simulations for performance and security analysis. We compare the performance of Black routing with Shortest Path routing.

6.1 Introduction

In the wake of several high profile and sophisticated network metadata attacks, there has been an increasing need for security and privacy in communication protocols [96] [97]. Metadata is being harvested at unprecedented levels, with new methods [98], for commercial gain and malicious intent [99], with the IoT being particularly vulnerable [100]. Specifically, resource-constrained IoT communication protocols, devices and networks, have been identified as an extremely vulnerable domain within deployed networks [101]. IoT communication protocol vulnerabilities are extensively documented [15]. The insecurity of the metadata remains the Achilles heel of communications security.

The popular IoT protocol, 6LoWPAN (IPv6 over Low Power Wireless Personal Area Networks), has a networking layer to adapt IPv6 for small packets of low-power PANs [102] [58], and a PHY and MAC sub-layer defined by IEEE 802.15.4 [39]. 6LoWPAN network attacks include resource exhaustion (where spurious messaging may be used to deplete energy and computational resources) [55], node capture (where the IoT node may be physically accessible and its data compromised) [80] [81], hardware trojans [82] and sensor side-channel threats (where information within the IoT node may be leaked via the sensors, or data gathered by the sensors can be tampered with) [83]. The standards-suggested security mechanism for 6LoWPAN, IPsec, cannot be applied to resource-constrained IoT networks due to heavy computational demands [103]. Lightweight IPsec versions [104] and end-to-end security [105]

for 6LoWPAN are proposed.

IPv6 (Internet Protocol version 6) is the communication protocol for the Internet. The IPv6 network layer (RFC 8200) [77] uses the PHY layer and MAC sublayer (IEEE 802.11) [106]. IPv6 security is provided by IPsec [107] whose support is mandatory with IPv6 deployment, but its use is not. IPv6 security vulnerabilities are a combination of: improper configuration of the complex IPsec suite, IPv4 attacks (such as application layer attacks, rogue devices, flooding) [108], IPv6 attacks (including auto configuration related attacks, metadata attacks such as neighbor discovery spoofing, router advertisement spoofing and redirects and privacy issues relating to address tracing and location inference for MobileIPv6 and insider threats within the local network) [109], implementation errors (in dual-stack systems vulnerabilities may be amplified with the combined use of IPv4 and IPv6), and IPsec vulnerabilities (e.g. authentication bypass via key reuse) [110]. In both IPv6 and 6LoWPAN, the metadata is not secured, and mechanisms proposed for securing the networking layer are incompatible with other protocols.

Black networks were introduced to mitigate metadata attacks [14]. Black networks encrypt both the metadata and the payload, in fixed length packets, at each layer in the protocol stack, thereby securing the packet from a range of passive and active attacks, both from outside attackers and insider attacks. The encryption of all metadata, including addressing information, prevents traditional routing approaches from moving the packet through the network. We present an SDN-based architecture to route Black packets from source to destination. The Black SDN Controller (BSDNC) uses a ciphertext-based packet forwarding mechanism to route the packets. The BSDNC is a trusted third party that performs the key management function, communicating with each node using unique, symmetric keys. While Black packets can provide confidentiality, integrity and privacy, in a communications protocol, an external observer can detect the communications and determine the communicating parties. To mitigate the threat of communicating parties discovery, we provide token-based node obscuring algorithm, hiding the source and destination communicating nodes. We simulate Black routing and node obscuring performance for our Black SDN configurations evaluating traffic overhead and network delay.

The main results of this chapter are: Black routing algorithms with an SDN architecture

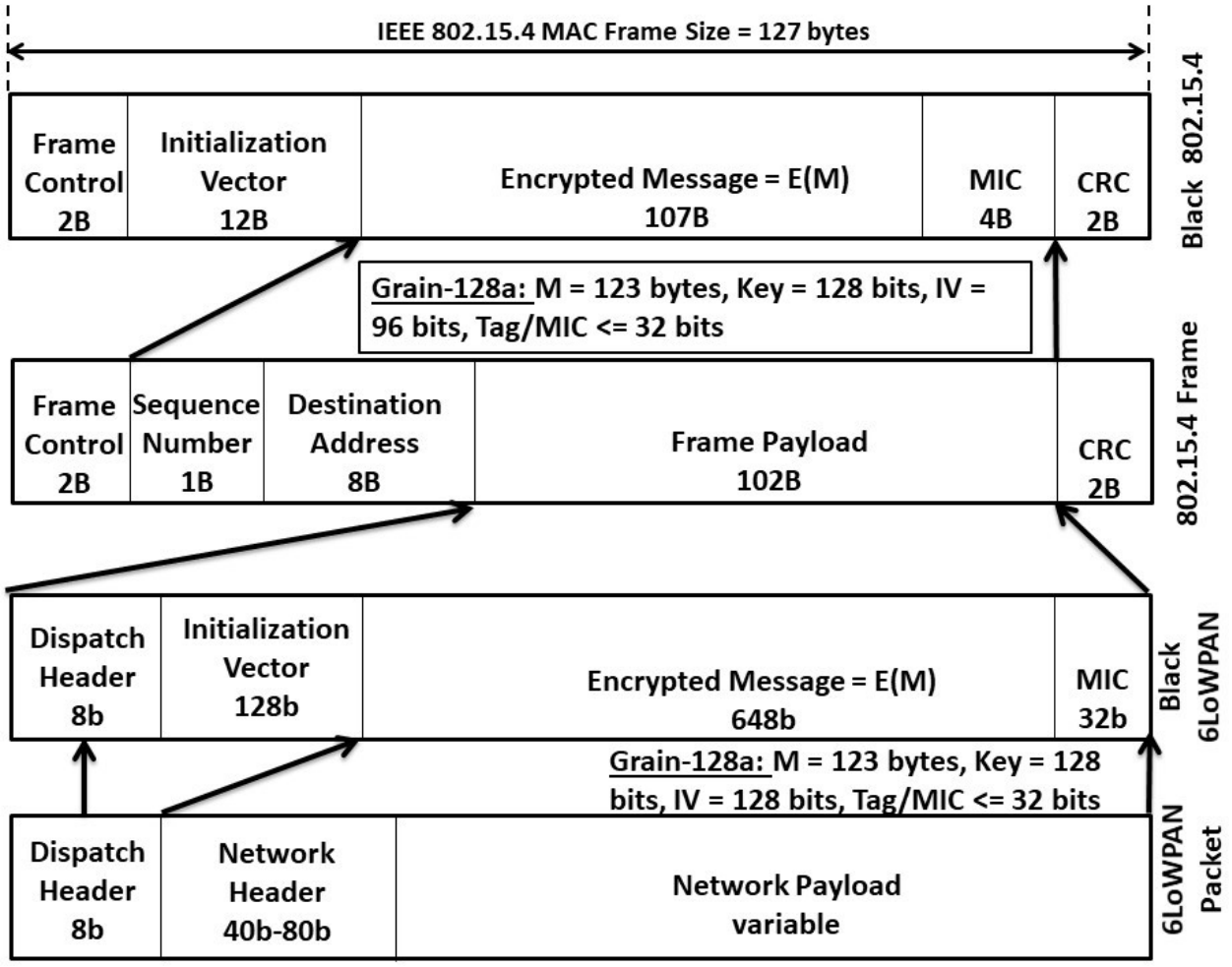


Figure 6.1: Black 6LoWPAN packet and Black 802.15.4 frame

in the star control and mesh control configurations and Black routing performance simulations.

The rest of this paper is organized as follows: In Section 6.2, we present Black packets for 6LowPAN and IPv6. Section 6.3 we present Black routing with an SDN-based architecture, with algorithms in a network. In Section 6.4 we present performance simulations for traffic overhead, network delay and congestion in Black routing. We analyze the security and performance impact of Black routing in Section 6.5, and draw relevant conclusions and identify future areas of research in Section 6.6.

6.2 Black Packets

Black packets are fixed length packets, that have both their payload and metadata encrypted at every layer of the communications protocol stack. Black packets [13] and Black networks [14] secure the metadata associated with the communications and mitigate a range of active and passive attacks. The encryption is performed by an authenticating, stream-based cipher, such as Grain-128a [74] or AES in the EAX mode [67]. Figure 6.1 shows the transformation of an 802.15.4 Link layer frame, to a Black 802.15.4 frame, and a 6LoWPAN packet to a Black 6LoWPAN packet, using the Grain-128a cipher, with a key of 128 bits and an initialization vector (IV) of 128 bits. The only portion of the packet, that is not encrypted are the first 8 bits of the dispatch type and header = 11111111. This is a reserved value where $b_0b_1 = 11$ indicating a Black 6LoWPAN packet, and $b_2b_3b_4b_5b_6b_7 = 111111$, a header type of ESC, indicating an additional dispatch byte follows. The Link layer transformation, from standard 802.15.4 to Black 802.15.4, is outlined in [14] [27].

Figure 6.2 shows a similar transformation for an IEEE 802.11 frame [76] to a Black 802.11 frame and an IPv6 packet [77] [78] [79] to a Black IPv6 packet. To demonstrate the flexibility in creating Black packets, we used the AES cipher, in the EAX mode, with a 128-bit key, and an IV of 80 bits (10 bytes). For an IP packet, the version field of 4-bits is either a 4 (for IPv4) or 6 (for IPv6). We use the reserved value of 1111 to denote a Black IPv6 packet. For the 802.11 frame, the first 2 bits of the 2-byte Frame Control field denotes the protocol version, and we use the reserved field of 11 to denote the Black 802.11 frame. In Fig. 6.1 and Fig. 6.2, the first 2 bits identify the frame as a Black frame, and are the only unencrypted portion of the frame metadata. The IV is in the clear, with the rest of the frame information encrypted.

6.3 Black Routing

Black Routing is the routing of Black packets in a (Black) network. Existing routing algorithms need metadata to route a packet, and that metadata is encrypted in a Black packet. Black routing utilizes an SDN architecture. SDN is a new networking paradigm that is catching on in enterprise networks, data centers and being evaluated in public networks [32], for broadband networks. SDN separates the control plane from the data plane,

allowing for network simplification, programmable networks, scalability and projected reduction in network costs [90]. The SDN architecture contains an SDN Controller, and a network of forwarding elements. Routing decisions, network configuration and node authentication are done by the SDN Controller. The forwarding elements have routing tables downloaded to them by the SDNC and forward incoming traffic accordingly. Open standards define the protocols that communicate between the SDNC and the forwarding elements [89] [88]. We adopt the SDN architecture with a Black SDN Controller (BSDNC) and Black control packets communicate with the network elements [14] [32] [7] [31]. Figure 6.3 shows two SDN configurations for a Black network. The Star Control configuration, is where the BSDNC

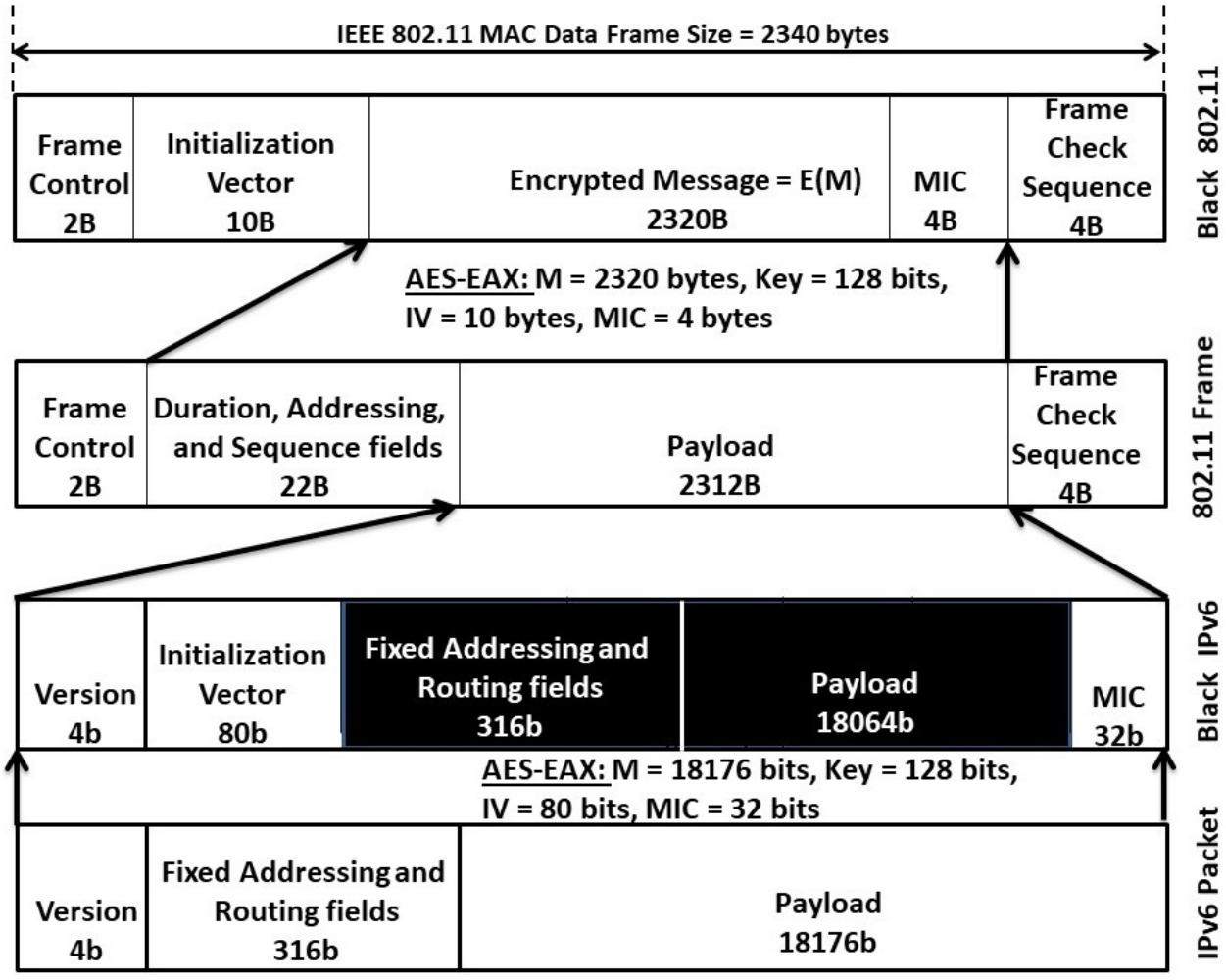


Figure 6.2: Black IPv6 packet and Black 802.11 frame

is connected directly to each network node. The Mesh Control configuration is where the BSDNC is directly connected to some of the network nodes.

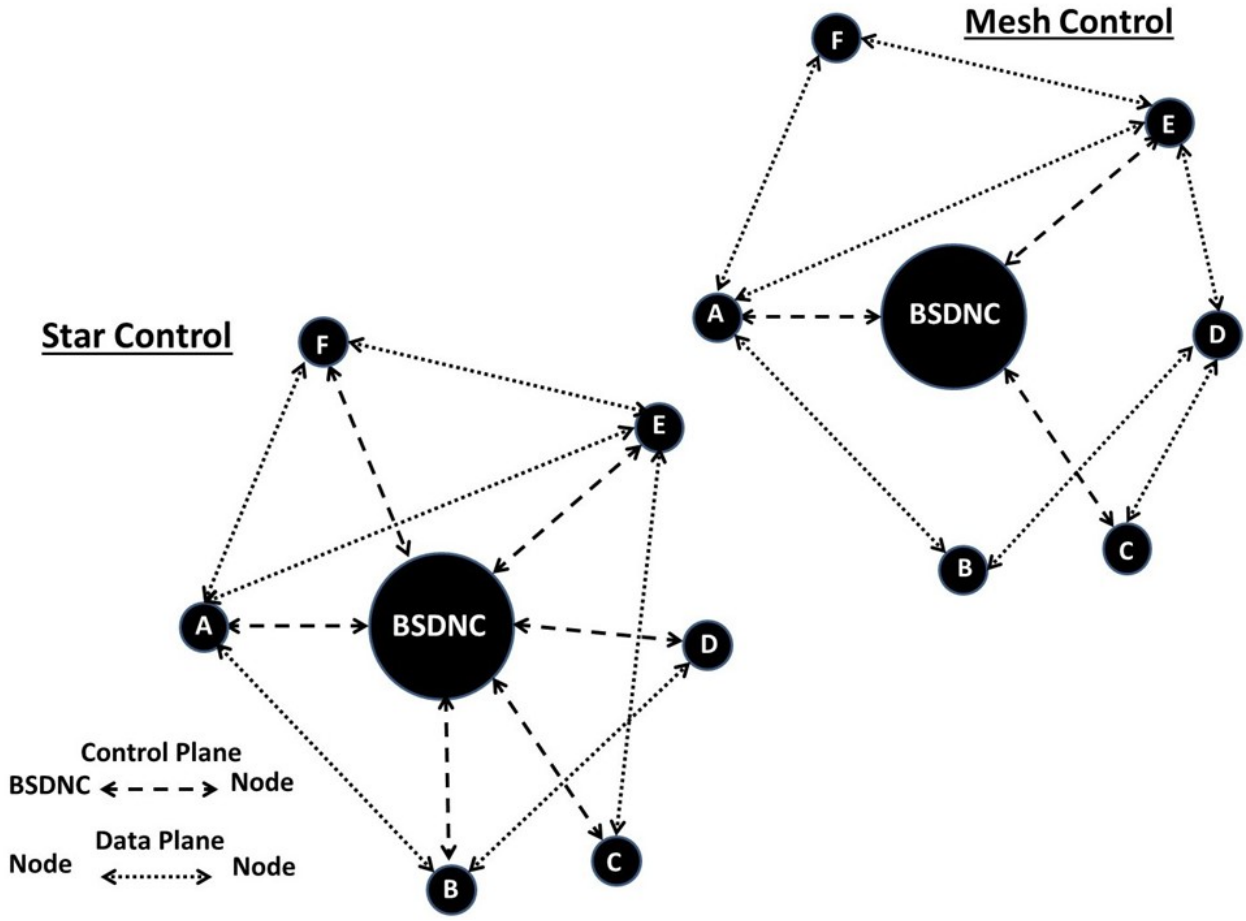


Figure 6.3: BSDNC configurations: Star and Mesh Control

Figure 6.3 shows two SDN configurations for a Black network. The 1-hop configuration (on the left), where the BSDNC is connected directly to each network node. The multi-hop configuration (right) is where the BSDNC is directly connected to some of the network nodes.

6.3.1 Star Control

In this configuration, consider Node A wants to communicate with Node D. Node A sends a Black control packet to the BSDNC with the address of Node A, the address of Node D, and the encrypted address of Node D. All encryption is done with the unique symmetric key

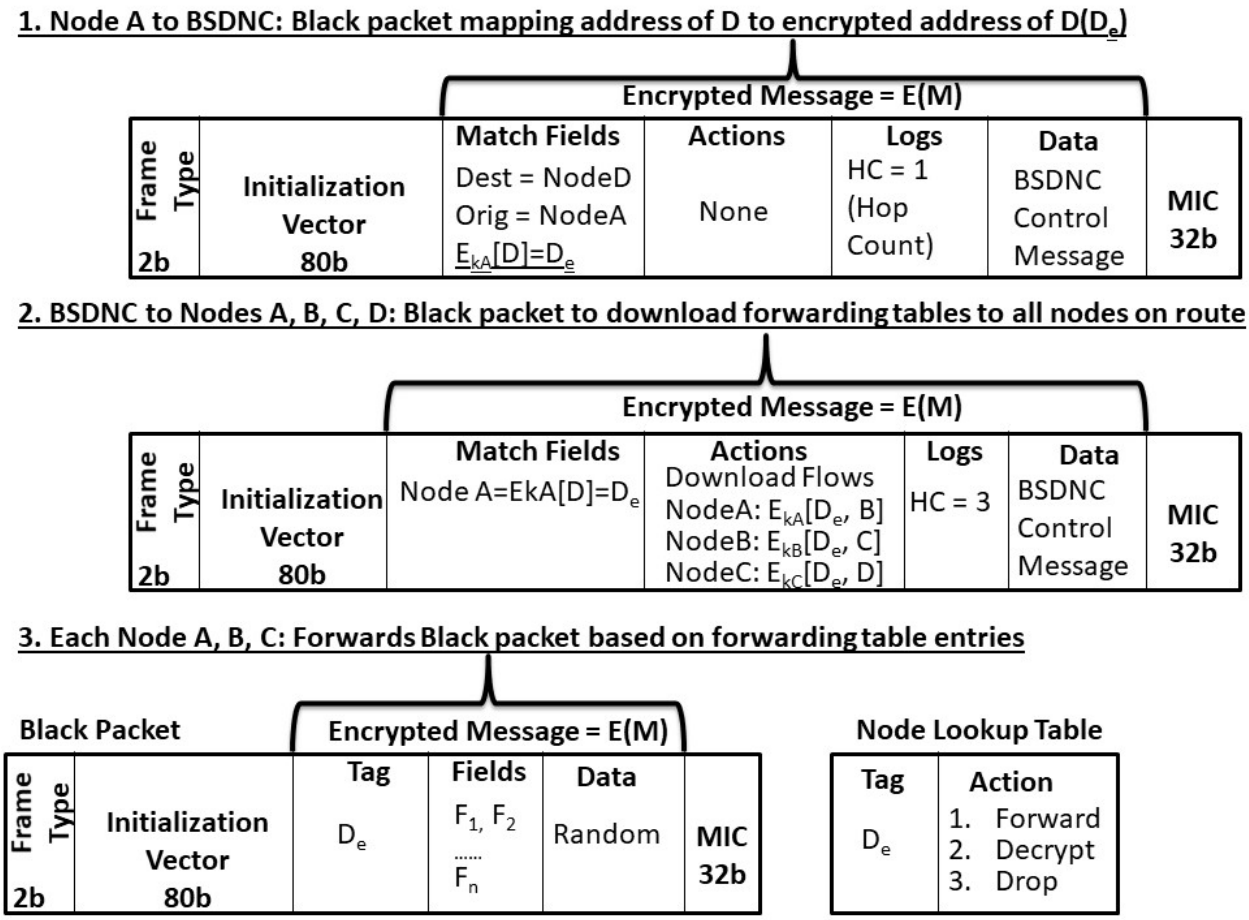


Figure 6.4: BSDNC messaging to nodes using Black control packets

of Node A (The BSDNC has unique symmetric keys for all nodes). The BSDNC maps the route from Node A to Node D, downloading routing entries with match fields to all nodes along the path, including Node A and Node D. The match fields indicate the next hop in the route, and in our case, is the string of bits corresponding to the encrypted address of Node D (denoted by $E_{kA}[D] = D_e$). Black data packets from Node A, check against match fields and forward to the next hop (Node B, Node C and Node D). When the Black data packet reaches Node D, it routes to itself, indicating that it has reached its destination. Black control packets, for each step, and routing table entries, are shown in Fig. 6.4. We present Algorithm 1 as a Black Routing algorithm for a BSDNC connected to every node.

Algorithm 1: Black Routing: Star Control to BSDNC

Result: Black Routing from NodeA to NodeD

Initialization; Source = NodeA; Destination = NodeD

Node A $\rightarrow$ BSDNC: $E_{kA[(A)}, (D), E_{kA(D)}]$

while *Node IN [A..D]* **do**

    BSDNC $\rightarrow$ NodeA..NodeD:

    Routing table match field = $\mathrm{E}_{k_A}(\mathrm{D})$

    Black packet to next hop based on match fields

end

6.3.2 Mesh Control

Let us consider a more complex scenario of the BSDNC connected to some of the nodes in the network - a likely scenario for ad-hoc, wireless, IoT networks. As the network initializes, nodes form a pre-determined paths to the BSDNC. Every node has a route to the BSDNC, whether it directly connected to the BSDNC or via a directly connected node. For Node A communicating with Node D, Node A sends the Black control packet $E_{kA}[(A), (D), E_{kA}(D)]$. Neither Node A nor Node D are directly connected to the BSDNC. All encryptions are done via the unique symmetric key of Node A (K_A). The pre-determined route sends the Black control packet to the BSDNC. The BSDNC brute forces the key, since the source of the Black control packet is unknown. In IoT network protocols, the address space is between 16 and 64 bits (Fig 6.1), and is computationally feasible. Algorithm 2 shows Black Routing with Mesh Control. We make several assumptions for the Black networks. We assume ALL nodal transmissions and receptions can be observed. The Black SDN architecture is a simplified for resource-constrained IoT networks, and applied to both wireless(6LoWPAN) and broadband(IPv6) networks. The BSDNC is not resource-constrained and is trusted. The BSDNC performs the key management function and holds unique symmetric keys for each node in the network. The BSDNC is capable of mapping routes upon request, as well as creating pre-configured routes between all nodes in the network. It must be noted that the Mesh Control configuration cannot be used for IPv6 Black networks. Brute forcing IPv6 addresses is not computationally feasible.

Algorithm 2: Black Routing: Mesh Control to BSDNC

```
Result: Black Packet routes from NodeA to NodeD
initialization
Neighbor list, paths to BSDNC downloaded to NodeA
while Dest NOT BSDNC do
    Forward E_kA[(A),(D),E_kA(D)] to Next Hop
    if BSDNC then
        Brute Force E_kA[(A),(D),E_kA(D)]
        Download forwarding tables to
            NodeA
            Intermediate Nodes
            NodeD
    else
end
```

6.4 Black Routing & Node Obscuring Simulations

We evaluate the performance of Black Routing using a discrete event simulator. Traffic is generated across the nodes, with Black packet inter-arrival times following an exponential distribution

$$T_t = e^{-\lambda * T_{(t-1)}}, \lambda = 2.0 \tag{6.1}$$

Simulator discrete time intervals per node is a single packet per time unit (TU). Collisions, acknowledgments and multicast are not simulated. All communications are in Black packets (control, data and tokens). We measure the *Mean Wait Time,* T_{MW}, *Mean Travel Time,* T_{MT} and the *Traffic Overhead*, T_{MW} is the time interval between Black packet generation and transmission at the source. We assume a single FIFO queue, processing both originating and inter-nodal traffic, to measure network traffic. T_{MT} measures network delay, the total time taken for a Black packet to travel from source to destination via intermediate nodes. Traffic overhead measures network efficiency for each configuration, and is the number of additional Black packets generated for each delivered Black data packet. We benchmark against Shortest Path Routing (SPR), which does performs traditional routing using standard IPv6 or 6LoWPAN packets.

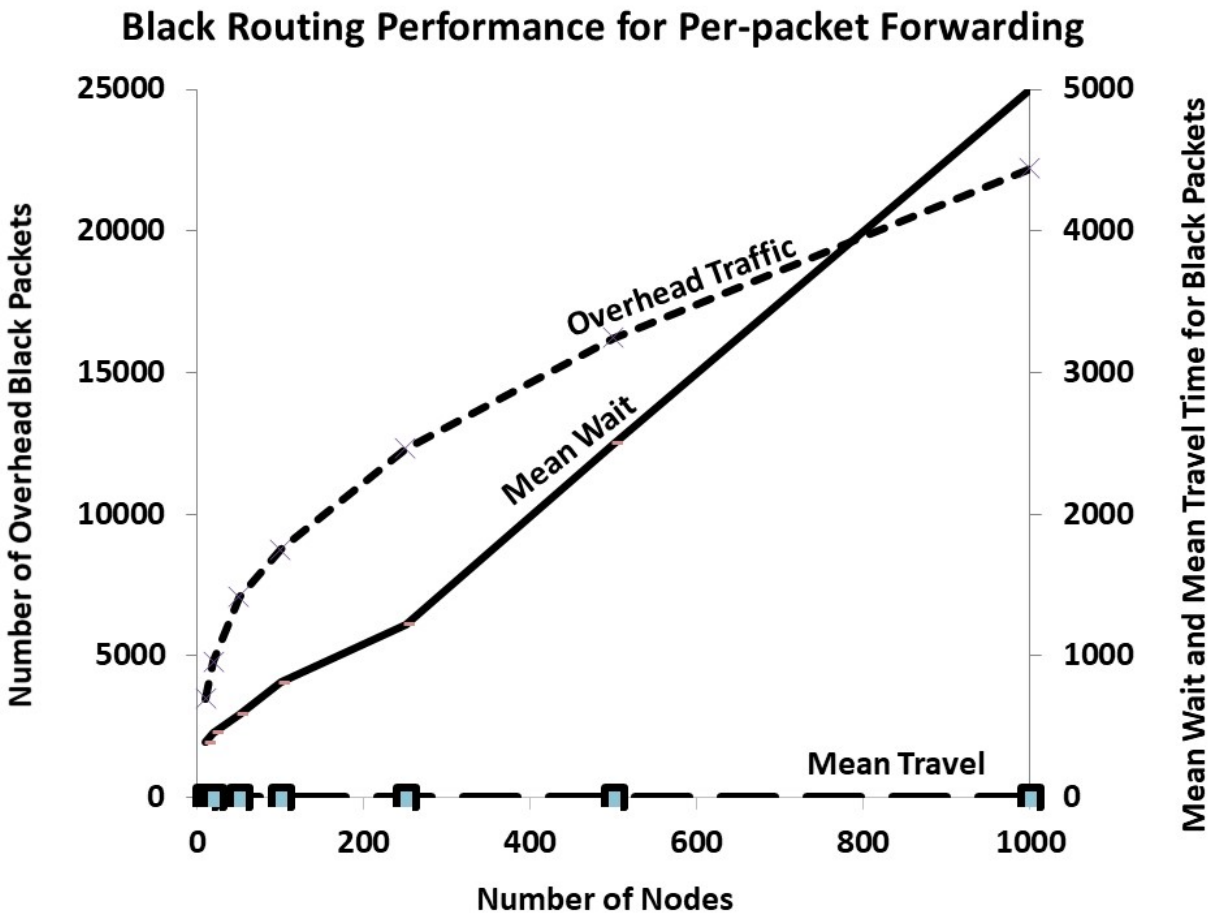


Figure 6.5: Black Routing performance in per-Black packet forwarding

6.4.1 Black Routing Simulations

The simulator models the 1-hop network with a BSDNC, in two configurations - synchronous (SBR) and asynchronous Black Routing (ABR). For each configuration, we model both ad-hoc and pre-defined routes (SBR-a, SBR-p, ABR-a, ABR-p in Table 4.2). In SBR, route requests and updates occur immediately for each generated Black packet to be forwarded from source to destination. In ABR, route requests and updates occur every 150 time units. Ad-hoc routes are requested from the BSDNC, for every Black packet, being sent from source to destination. Pre-defined routes are pre-configured routes in the BSDNC from source to destination for all nodes. The simulator models a Barabási graph network of increasing scale - 10, 20, 50, 100, 250, 500 and 1000 nodes. Each simulation is averaged over

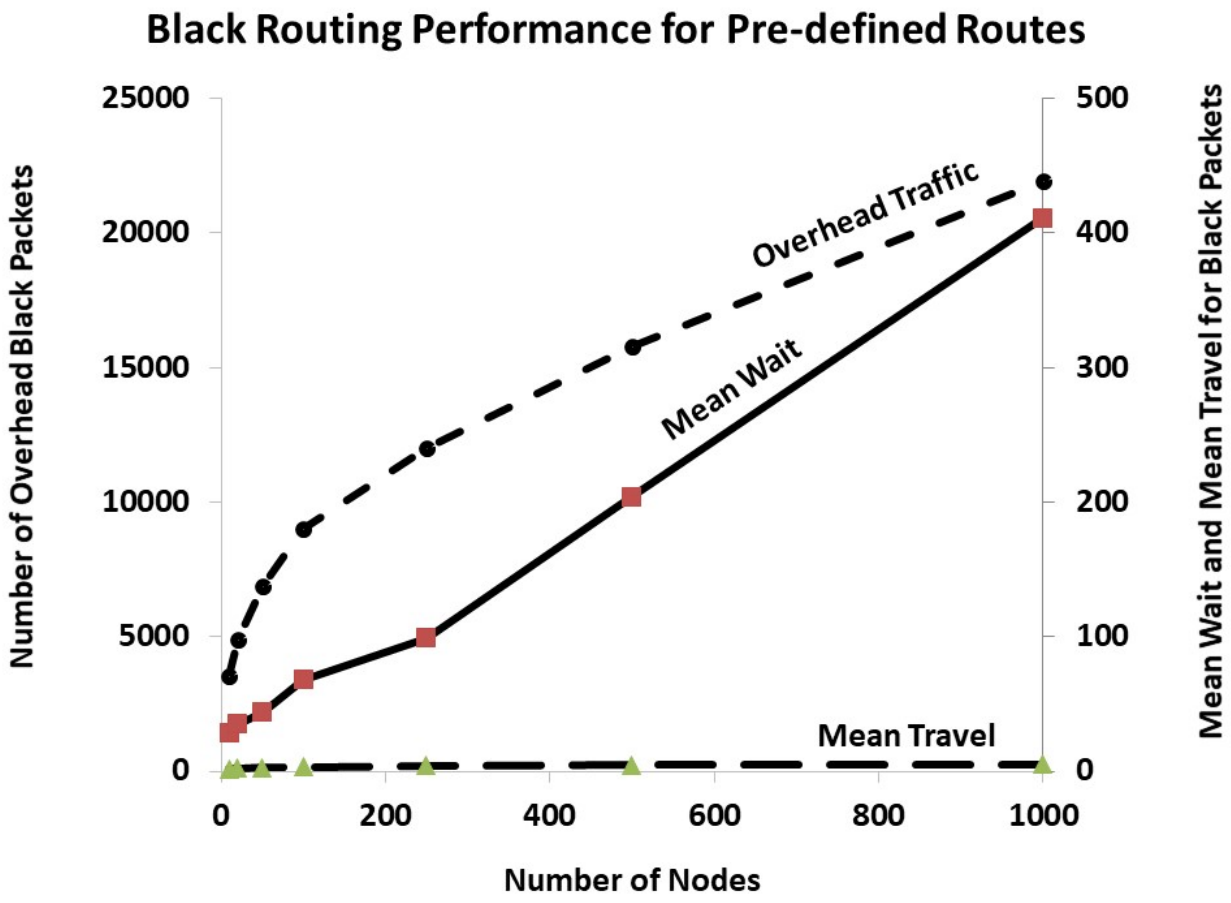


Figure 6.6: Black Routing performance with updates and pre-defined routes

10 random network layouts, with 1500 flows per network, for a total of 15000 data points per node set.

In Fig. 6.5, we display T_{MW}, T_{MT} and overhead traffic for SBR-a. Every Black packet from source to destination is forwarded based on the BSDNC downloading forwarding tables to all nodes the route. This straight-forward approach, of computing the route for every Black packet introduces higher overhead traffic and T_{MW}. Fig. 6.8 displays the performance indicators for ABR-p, where the forwarding table updates on pre-defined routes in the BSDNC occur every 150 TUs. The pre-computed approach and the sleep time, improves Black routing performance significantly. Results of all scenarios - SBR-a, SBR-p, ABR-a and ABR-p are listed in Table 4.2.

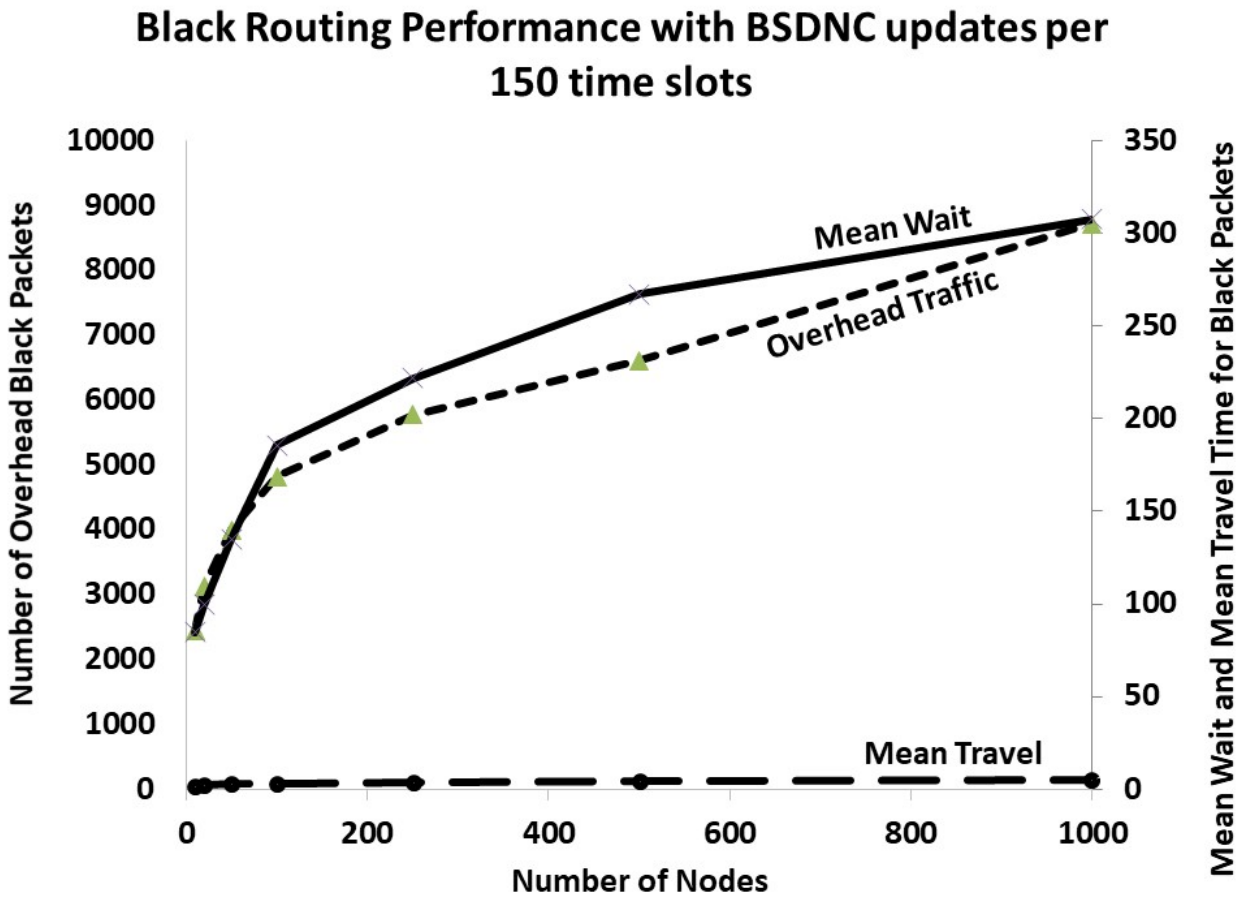


Figure 6.7: Black Routing performance with timed Flow refresh

6.5 Black Routing Analysis

Black routing uses an SDN architecture to route fixed-length Black packets. Black packets secure the communications metadata and data. They are compatible with the existing protocol in use (such as 6LoWPAN, IEEE802.15.4, ZigBee and IPv6) to provide privacy, confidentiality, integrity, authentication and node obscuring in network communications. Black routing mitigates a range of active, passive and insider attacks. We present Black Routing analysis for security and performance.

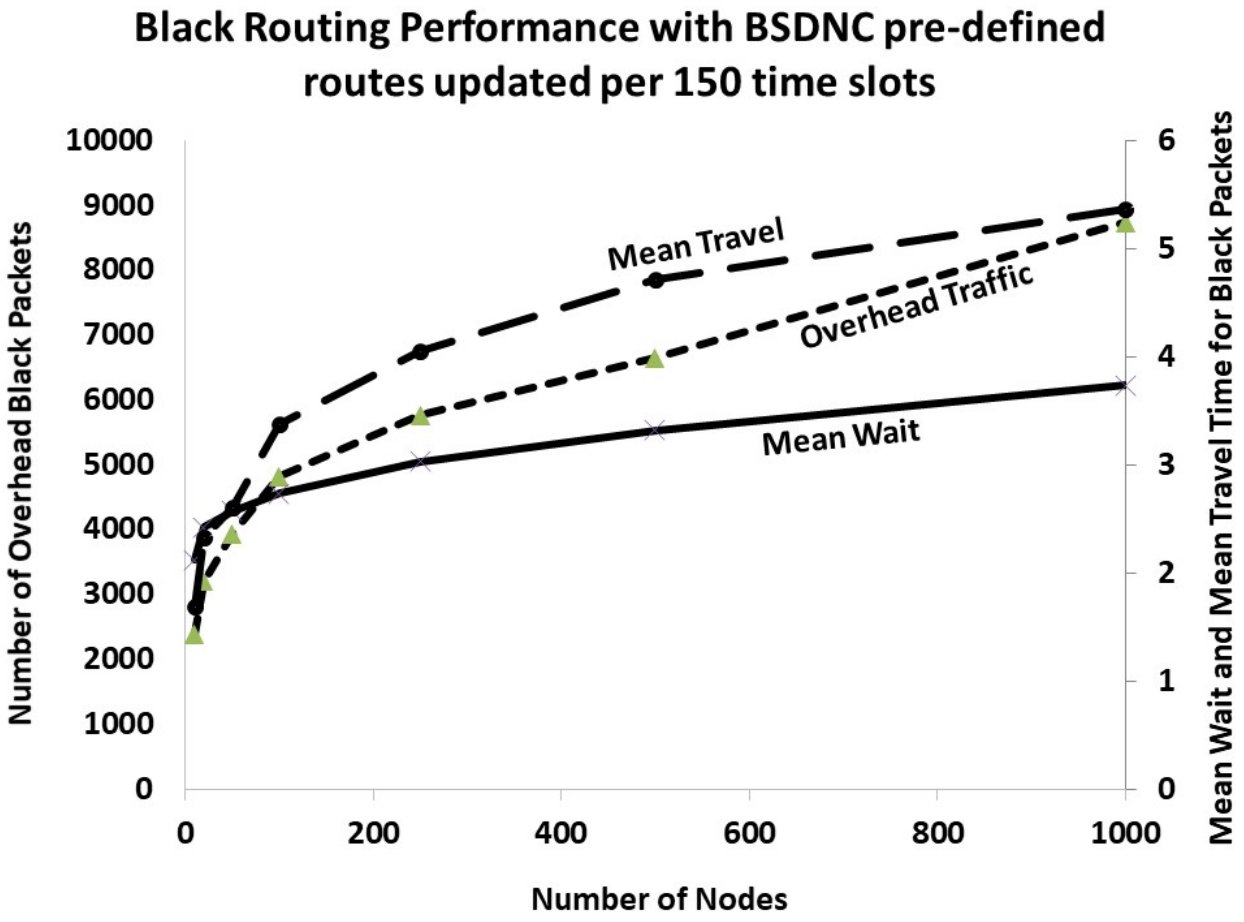


Figure 6.8: Black Routing performance with timed Flow refresh for Pre-determined routes

6.5.1 Security Analysis

Onion Routing (Tor) is the de-facto standard for anonymity and privacy in web-based, applications [95] [25]. A fixed route is selected by the Tor client. Intermediate nodes (called relays) have no knowledge of other nodes in the network, except for the node before, and the node after it. Tor uses public key cryptography and the source negotiates a session key with every successive hop. Black networks encrypt at every layer of the communications protocol (Tor secures TCP applications) and use symmetric keys, with no relay nodes, rendering them applicable for use both in IoT networks and broadband networks. Black routing is based on an SDN architecture and reaps the benefits of centralized network management, network scaling and reduced costs. Black routing is compatible with the existing protocol in

use and obscures the communicating nodes completely (including the exit nodes unlike Tor). Tor has been the subject of much analysis and several attacks on it have been presented including introduction of rogue relays [111], BGP hijacks [112] and perturbing the input traffic to observe changes in the output traffic [113], false resource advertising [114] and others [115]. While Tor remains the primary source of anonymous communication on the web, its vulnerabilities cause some Tor users to be compromised.

The 6LoWPAN IoT networking protocol uses the IEEE 802.15.4 PHY layer and MAC sublayer. IoT communications protocols are used in low-power, resource-constrained devices, where a variety of attacks are possible - including resource exhaustion attacks [55], sensor side channel attacks [83], hardware trojans [82], IP attacks [12] and vulnerabilities in the protocol [103] [105] and node capture attacks [54]. 6LoWPAN suggests the use of IPsec for network layer security, but does not mandate it. IPsec is complex, computation and resource intensive, for IoT protocols. The key management scheme associated with IPsec is IKEv2, and is computationally intensive for IoT systems - therefore key management for 6LoWPAN remains an open problem, with suggestions for an ECC-based approach by Raza, et. al [116], along with a compressed IPsec [104]. Black networks do not require additional security implementations for 6LoWPAN (or any other IoT protocol being transformed to its Black equivalent). Symmetric key management for Black routing is performed by the BSDNC, using Black control packets (Fig. 6.4). Insider threats are mitigated with unique symmetric keys at each node. Rogue nodes cannot communicate with legitimate nodes or with the BSDNC without a shared secret key. IP attacks are mitigated with fully encrypted packets and secured metadata. Resource exhaustion, traffic analysis and nodal attacks targeted towards the communicating nodes are mitigated by node obscuring. Inference and packet length-based attacks are mitigated by all Black packets being of fixed length.

6.5.2 Performance Analysis

Black routing performance is compared to Shortest Path Routing (SPR). The parameters of the simulation are described in Section 6.4. SBR-a and ABR-p are depicted graphically in Fig. 6.5 and Fig. 6.8. These are the 2 extreme cases of the entire data set of SBR-a, SBR-p, ABR-a, ABR-p and SPR shown in Table 6.1. SBR-a is the most expensive in terms

Table 6.1: Performance of Black Routing, Shortest Path Routing, and Node Obscuring (T_{MW}, T_{MT} (TUs), OT (packets))

# of Nodes	SBR-a			SBR-p			ABR-a			ABR-p			Shortest Path			NO-l		NO-g	
	T_{MW}	T_{MT}	OT	T_{MW}	T_{MT}	OT	T_{MW}	T_{MT}	OT	T_{MW}	T_{MT}	OT	T_{MW}	T_{MT}	OT	T_{MW}	T_{MT}	T_{MW}	T_{MT}
10	390	2.0	3497	28.2	1.7	3520	84.9	1.7	2446	2.1	1.7	2387	1.1	1.7	4815	7.1	5.0	3.2	2.6
50	587	2.9	7081	44.1	2.9	6891	135.2	2.9	3997	2.6	2.6	3935	1.0	2.0	5849	34.2	25.2	29.1	14.2
100	814	3.4	8749	68.1	3.4	9012	185.8	3.3	4818	2.7	3.4	4822	1.0	2.0	5851	64.4	49.8	43.1	20.8
500	2504	4.6	16241	204.0	4.6	15778	267.1	4.7	6612	3.3	4.7	6652	1.0	2.0	6207	236.5	248.9	90.4	44.7
1000	5004	5.3	22214	411.0	5.3	21951	307.7	5.3	8721	3.7	5.4	8742	1.0	2.1	6038	487.5	501.2	132.1	63.3

of overhead traffic and network delay for Black routing. In this configuration, for every Black packet to be sent from source to destination, there is a request to the BSDNC to set up a path between source and destination, with forwarding tables downloaded to intermediate nodes for each Black packet sent. As the number of nodes increases, OT increases exponentially compared to SPR, starting at 0.73 (3520 vs. 4815) of SPR at 10 nodes, increasing to 1.5 times at a 100 nodes ($\sim$8750 vs. $\sim$5850), to 3.7 times (22214 vs. 6038) at 1000 nodes. Network delay (T_{MW}) starts at 390 TUs, increases to 814 TUs and to 5004 TUs for 10, 100 and 1000 nodes respectively. For the simulated network configurations and the traffic flow, SPR T_{MW} does not experience network delay and remains constant at 1. In SBR-p, the routes are pre-defined for all source-destination pairs, and we note that T_{MW} reduces by an order of magnitude, compared to SBR-a (range: 390-5004 TUs vs. 28-411 TUs). For both SBR configurations, the traffic overhead (range: $\sim$3500 to $\sim$22000) and T_{MT} (range: 1.7-5.3 TUs) remain the same, for the simulated nodes. The rate of increase in T_{MT} for SPR is lower than Black routing (range: 1.7-2.1 TUs vs 1.7-5.4 TUs for all Black routing configurations).

Figure 6.8 shows the ABR-p performance. While T_{MT} (range: 2-5 TUs) and OT (range:$\sim$2500-8750 Black packets) are similar for both ad-hoc and pre-defined configs (Table 6.1), there is an order of magnitude reduction in T_{MW} from ad-hoc to pre-defined (range: 85-308 TUs vs. 2-4 TUs). Compared to SPR, ABR-p has between 0.5-1.5 times the OT, between 2-4 times the T_{MW} and between 1-2.6 times the T_{MT}, for the simulated range of nodes. The ABR-p configuration has the best Black routing performance characteristics of all simulated

configurations (SBR-a, SBR-p, ABR-a, ABR-p).

Across the four Black routing configurations, pre-configured routes (SBR-p and ABR-p) lead to a significant drop in T_{MW} when compared to the ad-hoc configurations (SBR-a vs. SBR-p, range: 390-5004 vs. 28-411 TUs; ABR-a vs. ABR-p, range: 85-308 vs. 2-4 TUs). Pre-computed routes need negligible setup time. The SBR configurations have similar (high) OT (range: 3497-22214 vs. 3520-21951 packets), based on the SBR update configuration. The ABR configurations maintain routing tables for 150 TUs, before updating them, leading to significantly lower OT and T_{MW} when compared with SBR configurations. Increasing the update time will improve the ABR-p performance characteristics when compared to SPR. For upto 20 nodes, OT for SBR is lower than SPR, and for upto 250 nodes, OT for ABR is lower than SPR. In comparison with SPR, we note that for upto 500 nodes, OT is better or equivalent for Black routing; for upto 50 nodes T_{MT} is equivalent in Black routing; and T_{MW} ranges between 2x and 3.7x for Black routing for the entire range (10-1000 nodes).

6.6 Conclusions and Future Research

Metadata is being collected at an unprecedented rate on users and traffic on the Internet and on the increasingly ubiquitous edge networks. The proliferation of IoT networks and their metadata vulnerabilities provide a huge attack surface. Black networks mitigate metadata attacks by securing both the metadata and the data; however, they require new algorithms to efficiently route Black packets while maintaining source and destination anonymity. We present several Black routing algorithms that use an SDN-based network architecture to route metadata-encrypted packets from source to destination while obscuring the communicating parties. Our simulations show that the improved security and privacy of Black routing comes with an overhead traffic and network delay cost that scales with the network size, but performance is comparable to basic Shortest Path routing for networks with less than 500 nodes. Black SDN networks are practical secure networks even with their security overhead.

Chapter 7

NODE OBSCURING

Black networks secure the metadata of networks communications. In the last 4 chapters we have started with Black packets, demonstrated simple Black communications with a star topology and Black Gateway, presented Black SDN and used that for Black routing. In these incremental steps we limit visibility only to network communications (packet data at each layer of the communications protocol). However, our assumptions include visibility of transmissions and receptions from, and to, the network nodes. To hide sending and receiving nodes from an external observer, we present node obscuring. We present two node obscuring configurations, and provide algorithms for each. We simulate the node obscuring configurations and compare to Black routing and Shortest Path routing.

7.1 Introduction

Hiding the metadata, is a necessary, but not sufficient, condition for communications privacy within a network - specially if data transmission by a node can be observed. The send and receive nodes must be obscured during communications [75]. We employ the concept of *subway* communications, using *tokens*. Tokens (trains) start at a Node 1, and traverse through ALL network nodes, to Node N. The tokens (empty Black packets) pick up data (passengers) from a source node (station), and drop them at their destination node (another station), and continue to the end of the line. The tokens pass through the data origination and destination points, pick up and drop off data, but the subway journey does not indicate the passenger pickup and drop-off points, thereby obscuring the source and destination. Fig. 7.1 shows the nXn grid network topology for node obscuring.

The simplest case for node obscuring is a single token traversing the grid, originating at Node 1, ending at Node N, sequentially through all the nodes (Fig. 7.1). Node P sends data to Node Q, where Node P is before Node Q, in the sequential path. When the token arrives

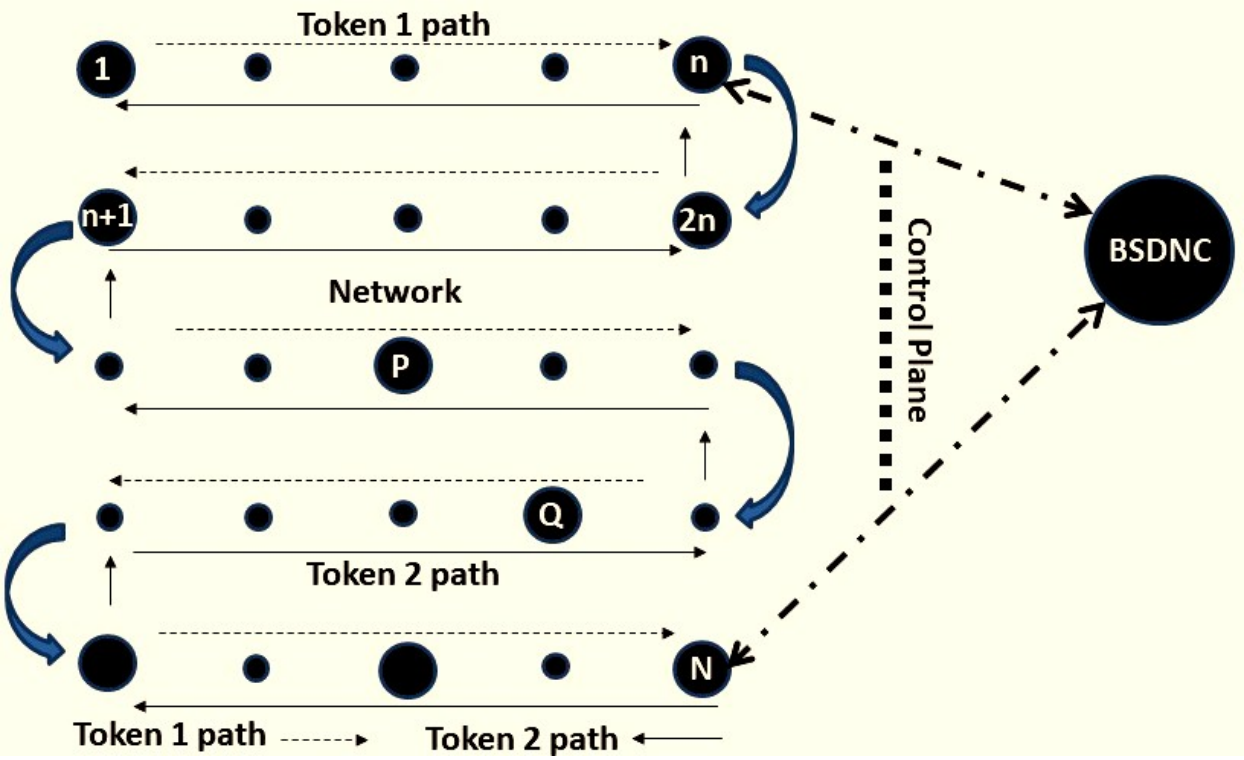


Figure 7.1: Node Obscuring in a Black Network

at Node P, the data is loaded onto the token. The token traverses the grid until it reaches Node Q, where the data is unloaded, and the token continues, to Node N. Once the token, reaches Node N, a new token is generated at Node 1, and it follows the same pattern. If Node Q, wishes to send data to Node P, then the token originates at Node N (and sequentially travels to Node 1). To further obscure the data transfer direction, tokens can travel from both Node 1 $\rightarrow$ Node N and Node N $\rightarrow$ Node 1, simultaneously. This is termed the *Node Obscuring-Linear, NO-l*. Obscurity is increased by generating more tokens. If tokens are generated at intervals of $T_i = T_{MW} + T_{MT}$, as a token travels to Node K_i, a new token arrives at Node K_{i-1}. There is a token at each node, when the first token reaches the final Node N. Additional tokens improve obscuring and the data transfer between communicating parties (Node P and Node Q). NO-l is simple and effective to implement, it is time-inefficient if the communicating nodes are spread far apart. Improved efficiency and bandwidth can be achieved by increasing the number of tokens and allocating them to sub-networks within the

Algorithm 3: Node Obscuring-Linear Algorithm, NO-l

Data: b_0=0 Empty token, b_0=1 Full token
Result: Communicating Nodes are Obscured
Init: BSDNC forms grid topology
Init: BSDNC configures communicating nodes
Init: BSDNC pre-shares symmetric keys
Request: Send data Node P → Node Q
    Node 1: initiate token(s) ;
while *NOT Node N* **do**
    if *(Node P) AND (b_0=0)* **then** Load token E_{kQ}[Data];
    if *(Node Q) AND (b_0=1)* **then** Unload token;
    Next node

network. The *Node Obscuring-Grid (NO-g)* model generates nXn tokens for each row and column of the Fig. 7.1 grid network. For data transfer from Node P → Node Q, Node P loads the data into the token traversing the column of Node P (T_{CP}), till the row of Node Q (R_Q). The data is then transferred to Node Q, by a token that traverses the row of Node Q (R_Q). Alternatively, the data may be transferred, via row token, from the row of Node P (R_P) to the column of Node Q (C_Q), and then to Node Q via column token (T_{CQ}) (Table 6.1).

Tokens are empty Black packets of the structure shown in Fig 6.4. The set of communicating nodes are pre-determined, their symmetric keys are pre-shared, and the transfer points (for NO-g) are pre-set during initial network configuration, by the BSDNC. Empty tokens are generated with the leading bit as b_0=*0* and traverses the nodes until it reaches Node P. Node P checks the leading bit (0 for empty token), loads data into the token, and encrypts the token with K_Q, and sets the leading bit to b_0=1 (indicating a loaded token). At each subsequent node, a MIC (Message Integrity Code) check is done to determine if the token is meant for that node. When the token arrives at Node Q, the MIC passes, the token is unloaded and decrypted with K_Q, and the token is forwarded to the next hop towards Node N. Algorithm 3 and Algorithm 4 present NO-l and NO-g. Both node obscuring configurations are pre-configured by the BSDNC, resulting in negligible control traffic to the BSDNC from the nodes, and no Overhead Traffic (OT). Tokens are generated at random or fixed intervals, by the nodes, and travel along a specified path.

Algorithm 4: Node Obscuring-Grid Algorithm, NO-g

```
Init: BSDNC determines communicating nodes and pre-shares keys;
Init: BSDNC determines, sets transfer nodes on R_P, C_Q ;
Init: tokens travel to and fro on all rows and columns;
Request: Send data Node P → Node Q
while NOT (Row n)OR(Column n) do
    Load T_RP@ Node P → C_Q; Load to T_CQ
    Unload T_CQ@ Node Q → Token continues
Tokens reach end of Row/Column and return to Node 1
```

Tokens are empty Black packets of the structure shown in Fig. 6.1 and Fig. 6.2. The BSDNC sends and receives Black heartbeat messages to and from all nodes, at regular intervals. When Node P needs to send data to Node Q, the heartbeat response includes $E_{kP}[(P),(Q),E_{kP(Q)}]$, to the BSDNC. The next heartbeat message from the BSDNC to Node 1, Node P and Node Q includes a session key K_{PQ}, and for Node 1 to initiate a token using the session key. The SDN match field for nodes P and Q is $E_{kP}(Q)$. The action fields for Node P and Node Q are *encrypt* and *decrypt* respectively. The token traverses the network from Node 1 to Node N, encrypting/loading the data onto the token at Node P and decrypting/unloading the data at Node Q. The operation is similar for both NO-l and NO-s.

7.1.1 Node Obscuring Simulations

Section 7.1 presents node obscuring using tokens and subway routes, using a *10x10* grid network topology. We simulate two configurations for node obscuring: The node obscuring linear (NO-l) and the node obscuring grid (NO-g) models. The results of T_{MW} and T_{MT} are shown in Table 6.1. We maintain the 15000 data points per node set. For NO-l we simulate tokens traveling sequentially through each node from *Node 1* to *Node N*, generated at the rate of

$$T = T_{MW} + T_{MT} \tag{7.1}$$

(Fig 7.1). This indicates a token at each node behind the first token. For NO-g, we simulate a single token starting from, and returning to, the first node in each row and column.

Fig 7.2 and Fig 7.3 graphs the performance of NO-l and NO-g. There is no overhead

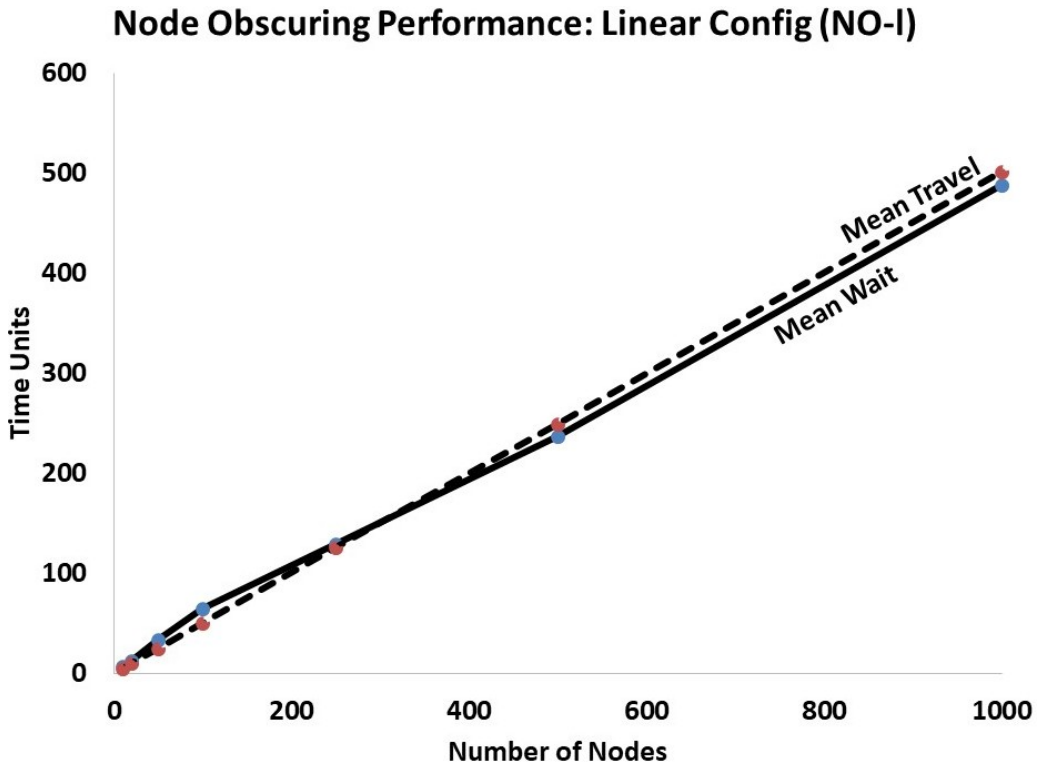


Figure 7.2: Node Obscuring-Linear (NO-l) token travel times

traffic in either case.

7.1.2 Node Obscuring Analysis

Figs. 7.2 and 7.3 display the mean wait time T_{MW} and the mean travel time T_{MT} for node obscuring algorithms NO-l and NO-g. Table 6.1 presents the simulation results for the node obscuring configurations. Since there is no communication with the BSDNC at runtime, there is no overhead traffic, OT for NO-l and NO-g. The NO-l T_{MW} and T_{MT} increase linearly with the number of nodes. This is consistent with the way the tokens are generated and the time it takes to traverse the network as shown in Equation (7.1). Dividing the grid network into subnets of subway routes (rows and columns) significantly reduces the T_{MW} and T_{MT} for NO-g, but remains much higher compared to the other Black routing algorithms. Further division of the network into smaller subnets will yield better node obscuring performance results, but may not be more secure, as obscuring becomes

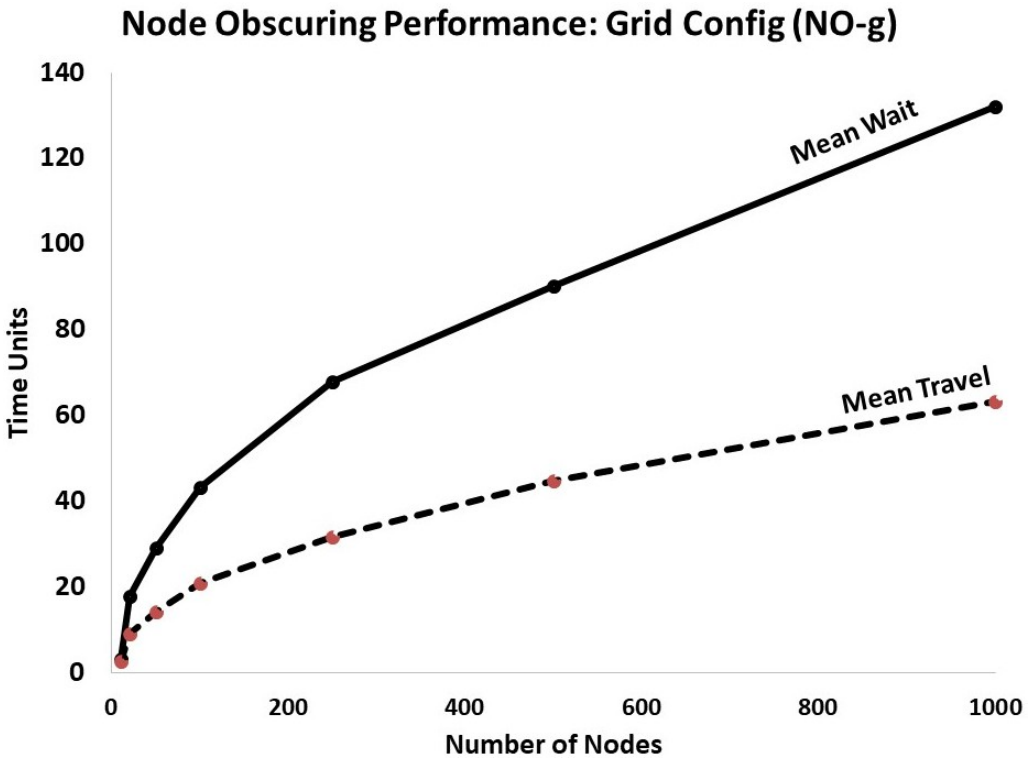


Figure 7.3: Node Obscuring-Subway (NO-g) token travel times

challenging with a smaller number of nodes. In summary, we note that the high T_{MW} and T_{MT} times for NO-l and NO-g are offset by no OT as compared to SPR and Black routing.

7.2 Conclusions and Future Research

Sustained traffic analysis over a localized IoT network can yield information about, and divulge the identities of, the communicating nodes, even in a Black network. Node obscuring techniques use subway communications and tokens to transmit data from origination to destination, and obscure the communication parties. We present two node obscuring algorithms, both of which tradeoff (higher) mean wait and mean travel times with higher bandwidth transfer and no overhead.

Part III

Black Networks in Secure Smart Cities

Chapter 8

INTRODUCTION TO SMART CITIES

Black networks can be applied to any industry that deploys IoT networks. The application domain proposed for Black networks is Smart Cities (in this dissertation). Smart cities deploy a complex array of IoT networks to provide citizen services. Smart cities are increasingly IoT-enabled. In this chapter, we introduce smart cities and their basic structure and services. We provide an appreciation for smart city vulnerability based on insecure IoT protocols.

8.1 Introduction

The Internet of Things (IoT) is growing pervasively around us. IoT refers to a system of small, smart objects that form low power, low duty cycle, ad-hoc, wireless mesh networks to monitor (sense) and transmit that information. IoT is found in healthcare (medical monitoring devices), electrical utilities (smart meters), physical security (wearable or wireless cameras), transportation (smart cars), industrial automation and controls and large composite systems like Smart Cities. IoT nodes usually powered by a small battery that lasts from months to a few years. This energy-efficient operation is possible as the nodes 'sleep' a majority of the time and 'awaken' to transmit small amounts of information. Given the size of these nodes, they have computational, memory, range of operation and energy constraints and must run efficient software protocols. A widely used base protocol for IoT is IEEE 802.15.4 LR-WPAN (Low Rate Wireless Personal Area Networks) [39]. 802.15.4 defines the Physical layer and the MAC-sublayer of the Link layer of the communications protocol. The network, transport and application layers are defined by protocols that are built on top of 802.15.4 such as 6LoWPAN [40], ZigBee [3] and WirelessHART [9]. Another commonly used high rate IoT protocol is Bluetooth Low Energy based on IEEE 802.15.1 WPAN.

A large scale application of such networks is in smart cities. Smart Cities incorporate

diverse, heterogenous, IoT networks, to automate and deliver enhanced citizen services. An example of a wide variety of smart city services, enabled by IoT devices and networks, in multiple domains (emergency services, nuclear power plant monitoring, environmental monitoring, healthcare, and other services) is illustrated in Fig. 8.1

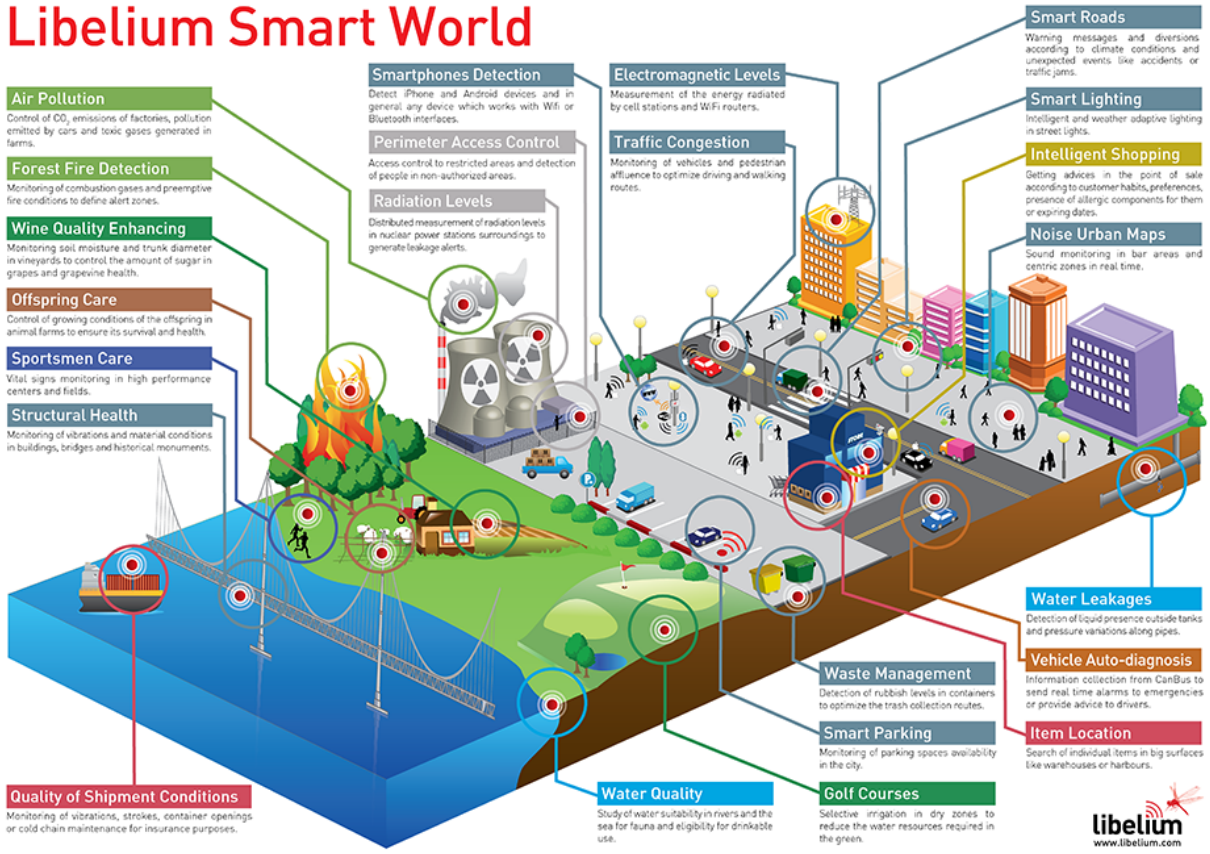


Figure 8.1: An IoT-enabled smart city. Source: Libelium [1]

8.2 Smart City Basics

Smart Cities have been proposed, and constructed, globally to provide enhanced citizen services and better city management. With over 50% of the world's population now in cities, with an accelerating trend, city managers proposed the use of Information and Communications Technologies (ICT) to modernize cities – creating Smart Cities [117]. Smart Cities are

large, complex, distributed, and continuous systems - with mission-critical data that must be secured end-to-end. The widespread use and adoption of IoT has resulted in IoT deployments within Smart Cities for critical services – such as critical infrastructure monitoring, water supply purification, pollution monitoring, street and traffic lighting. Smart Cities are increasingly becoming IoT-enabled and IoT dependent [118].

IoT networks can range from a single domain of multi-networks, to multi-networks across domains. A domain is a vertical - like healthcare, physical security, and/or energy. IoT nodes have limited computational power and memory and are carrying increasing amounts of mission-critical information, and the basic security mechanisms implemented within the above protocols are inadequate.

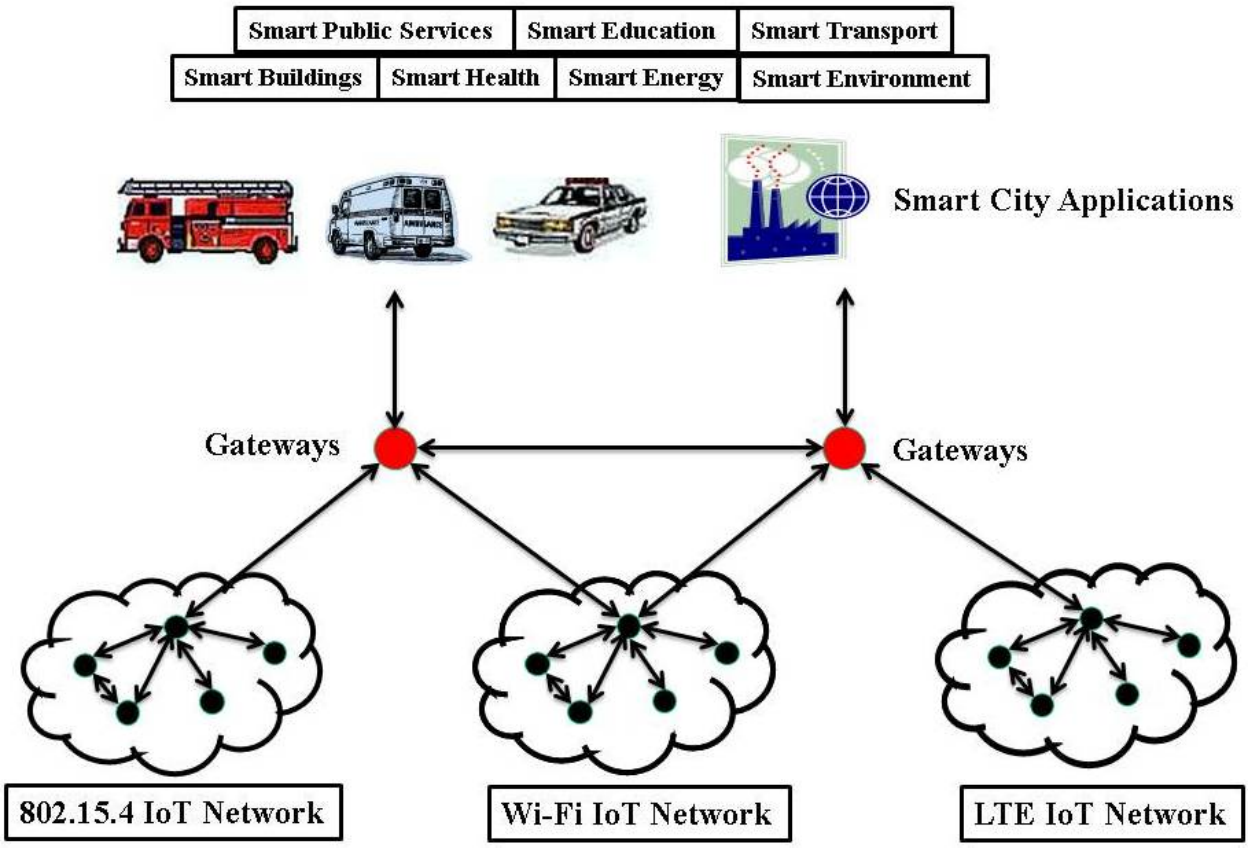


Figure 8.2: IoT Networks for various Smart City Functions

A domain, like healthcare, may have different networks - e.g. a body-area network, monitoring health parameters like temperature, heartbeat, oxygen levels, blood pressure for an individual; to monitoring a hospital of multiple patients, assets, and healthcare records. The traffic and its class of service varies widely across a domain or multiple domains. A cluster of sensors in close proximity results in a sensor network (often wireless) performing a single set of functions. Such networks will need a gateway to transmit the sensor data to a storage location (typically a cloud storage). Multiple such networks, geographically distributed, with their respective gateways, will transmit and store the data. Data analytics can performed on this stored data set. Multiple networks will result in multiple data sets of multiple types (ie patient data sets, to hospital data sets, in our example). Moreover, the wireless protocols that transmit data from sensors (to the gateway), will be different from, the protocols that will transmit the data from the gateway to the cloud storage. Networks of sensors will relay information for storage to the cloud. It can get very complicated, very quickly. Figure 8.2 illustrates IoT networks supporting multiple Smart City functions using heterogeneous wireless networks.

Our primary goal of designing IoT security for a Smart City is to ensure the security for ALL data that is in transit or at rest. At times we trade routing efficiency for higher security. Our precept is that the data being gathered, transmitted and stored is mission-critical. Chapter 9 presents a secure IoT framework, capable of generating, transmitting, storing, visualizing and actioning, mission-critical traffic across multiple IoT technologies.

Chapter 9

SECURE SMART CITIES

In this chapter we propose a secure IoT architecture for a Smart City. Secure Smart Cities are IoT enabled, and provide secure communications, advanced networking, authentication and external key management (i.e. key management outside of the IoT protocols key management services). Secure communications are provided via Black IoT networks. Advanced networking is via a software defined networks (SDNs), Authentication of all nodes is done by a Unified Registry. External key management is performed at each layer of the protocol stack.

9.1 Introduction

The security of Smart Cities is dependent upon the security of the underlying IoT protocols. These protocols have well-documented vulnerabilities, which make Smart Cities vulnerable to a wide range of attacks with possible catastrophic consequences. The security of IoT networks and its protocols has been examined in [12] [10] [54]. The main contribution of this paper is a secure IoT architecture for Smart Cities. The architecture contains Black Networks, SDN Controller as TTP, Unified Registry and Key Management. The security services extend beyond the basic security provided by IoT protocols to confidentiality, integrity, availability, privacy, identity management, authentication, authorization, and accounting - across heterogeneous IoT networks, across multiple device types, and for multiple Smart City functions. The security services provided mitigate the vulnerabilities of basic IoT networks, for mission-critical data, at the Link and Network layers. The secure IoT architecture for Smart Cities presented includes identity management, authentication, authorization, confidentiality, integrity, availability and privacy/anonymity to ensure mission-critical data is secured at the Link and Network layers [119].

The remainder of this chapter is organized as follows: in Section 9.2, we provide a

Security Overview for IoT networks, needed for Smart Cities. In Section 9.3 we present the building blocks for a secure IoT framework for a Smart City. An evaluation of the secure IoT framework is presented in Section 9.4. We draw relevant conclusions and suggest future areas of research in Section 9.5.

9.2 Security Overview

What security services must a Smart City IoT network provide? [120] There are several fundamental security services that are provided by a simple IoT network [121]: *access control, message integrity, message confidentiality, and replay protection.* These security services are designed into IoT protocols (such as IEEE 802.15.4), and provide basic protection. Each layer in the communication protocol stack should provide these security services. Additional security services, such as *routing integrity* and *routing assurance* should be provided at the Network layer, and *application security* at the Application Layer.

Access control services allows communications between authorized devices only. At the link layer, access control services prevents communications between authorized and unauthorized devices. Access control at the Link layer is the first, and lowest cost, layer of defense preventing unauthorized devices from accessing the network. Access control at the Network and higher layers prevents a device from accessing or using a resource for which it is not authorized. Access control at the Network and higher layers can be costly to implement but may protect individual resources and functions contained within a smart object or within the network itself. A *message integrity code* (MIC) may be included with each frame for both authentication and integrity. A MIC is a cryptographically secure digest of the message, or a portion thereof, that is typically computed using a secure hash function such as SHA-256 or SHA-3. A MIC may be used to protect the integrity of a frame, and it is perhaps the simplest approach to providing authentication at the Link layer. *Message confidentiality* ensures that the intended recipient gets the message. Message confidentiality is done by encrypting the message payload. The header information for the layer is not encrypted. The commonly used symmetric cipher for IoT nodes is AES (Advanced Encryption Standard). *Replay protection* services ensure that duplicate messages between authenticated parties are detected and dropped. A replay attack is simply the intentional retransmission of valid packets in an

attempt to either gain access to a resource or deny that resource to others.

Jamming (denial of service at the physical layer), node capture and a variety of resource exhaustion attacks are specific to IoT nodes because they are resource constrained and physically accessible. Power depletion attacks, where a device is forced to utilize all of its available energy to manage malicious communications or perform activities requested by an adversary, require explicit power management services in order to limit the consequences of the attack. Power depletion attacks have created specific security guidelines that are normally not considered in standard networks [55]. Node capture refers to an adversary directly accessing the device, either through physical access or electronic access, allowing the adversary to extract keys, inject messages, operate as an authenticated node and remove nodes from the network. Node capture can be mitigated by enforcing certain security requirements such as erasing secure key information when the node is disassociated from a network [54].

Additionally, inference attacks, traffic analysis, dictionary attacks, eavesdropping, packet injection and packet modification can be made based on the metadata associated with each frame and packet. Finally, popularly deployed IoT protocols, in Smart Cities, have vulnerabilities that risk mission-critical data. The 802.15.4 base protocol defaults to NO security unless security modes are explicitly requested by upper layers. ACKs in this protocol are not encrypted, leading to frame interception by an adversary, followed by a spurious ACK, resulting in frame loss with no retransmission. 802.15.4 does not use timed replay counters. An adversary can send a large number of intercepted frames, with large counters, thereby causing valid frames with smaller counters to be rejected [10]. The commonly deployed ZigBee protocol for IoT networks defines a single Trust Center (TC), in its security architecture, that is trusted by all nodes. The centralized nature of the TC, and the critical functions of key management and distributions that it performs, presents a significant vulnerability. When a ZigBee node is removed from a network, it still contains the Network Key, and data, that can be compromised [26]. WirelessHART protocol standards contain, but do not define, a Security Manager, which is expected to perform critical key management functions (generation, storage, renewal and revocation). This may lead to compliant, but insecure implementations [12]. Another commonly deployed IoT protocol, 6LoWPAN, uses the IPsec security architecture (for authentication and key management), which is highly resource and

computationally intensive for IoT applications (and impractical to deploy, and not mandated by the standard). Standard IP network threats also apply to 6LoWPAN [54] [85].

9.3 A Secure IoT Architecture for Smart Cities

What fundamental components are required to provide the security services for an IoT-based Smart City? [122]. Figure 9.1 shows the components of a secure Smart City IoT architecture, in the context of Smart City functions. Smart City IoT operate over heterogeneous networks, across multiple device types (Figure 8.2). In order to provide security services for mission-critical data, we assume that not all security can be embedded within the protocols, because of resource constraints. With these assumptions, we present four fundamental building blocks to provide a secure IoT architecture for a Smart City. They are:

- **Black Networks**: Data privacy, confidentiality, integrity and authentication
- **SDN (Software Defined Network) Controller**: Efficient and anonymous routing across IoT nodes that sleep upto 90% of the time.
- **Unified Registry**: Database of devices (sensors, gateways and nodes) and their attributes
- **Key Management**: A key management system for IoT networks.

9.3.1 Black Networks

A Black Network secures all data, including the metadata, associated with each frame or packet in an IoT protocol. Black Networks encrypt the payload and the metadata within an IoT protocol Link layer communications. For connectionless protocols, the cipher's initialization vector (IV), encrypted metadata and payload are including in every frame. For connection-oriented protocols, the IV is exchanged separately. Similarly, the metadata is independently secured in the Network layer. Encryption can be done via stream-oriented ciphers such as Grain128a [74], or standard AES ciphers in the EAX [67], or OFB modes. The resulting compatible frame, allows the intended recipient to correctly receive and decode the message, via a shared secret. Black Networks mitigate a broad range of both passive and

active attacks, and provide confidentiality, integrity and privacy in IoT networks due to the authenticated and secured communications at both the Link layer and the Network layer. However, encrypting the header creates routing challenges for IoT nodes which are asleep a majority of the time.

9.3.2 SDN (Software Defined Networking) Controller

Software Defined Networking (SDN) is a new routing paradigm that simplifies the routing function to packet forwarding while abstracting the control to an SDN controller [30]. Using SDN Controllers for wireless IoT networks, with a light flow-table mechanism is an emerging field. [7]. The primary motivation for an SDN Controller is to resolve the routing challenge presented in privacy preserving Black IoT networks. The problems to be solved: how does Node A send a packet to Node B, in an Black IoT network, without a) an adversary knowing the packet is destined for Node B b) traverse the IoT network where the nodes sleep a majority of the time. We propose two general methods to resolve this. The SDN controller, with an IoT network topology view, a sleep/wake timing view, can synchronize the nodes. Therefore, the SDN controller can deliver any Black packet, from Node A to Node B with flow tables, synchronizing the wake times for the intermediate nodes. Another approach is for the SDN Controller to create a random, dynamic route for the each hop based on Onion Routing [25]. The SDN Controller maintains a global IoT network view, manages sleep/wake cycles, along with other network states. The centralized SDN Controller improves availability of IoT networks and leads to a simplified network architecture.

9.3.3 Unified Registry

The concept of a Unified Registry is to consolidate the heterogenous technologies, addressing schemes and devices that make up IoT nodes. The concept can be extended to a Visiting Unified Registry for IoT nodes that are mobile, and cross networks. This is important from a security standpoint – a majority of IoT networks assume fixed nodes communicating using wireless technologies. In a Smart City environment, there are multiple wireless technologies in use (e.g. WiFi, LTE); there are multiple protocols in use (such as ZigBee, 6LoWPAN, WirelessHART, ISA100.11a, Bluetooth Low Energy) depending upon the domain; there are multiple addressing schemes (e.g. IPv6 128-bit addressing, Bluetooth

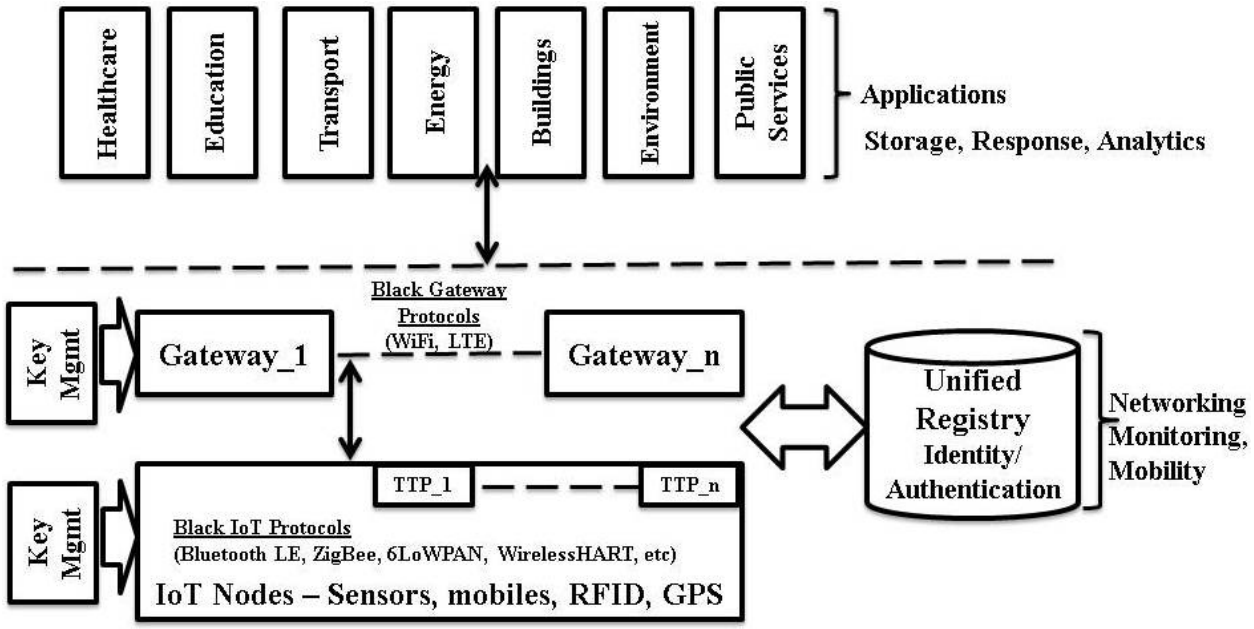


Figure 9.1: Components of a Secure IoT Architecture for Smart Cities

48-bit addressing, RFID addressing and E.164 (telephone numbering plan). All of these identities need a unified attribute set, for identity management, authentication, authorization and accounting. In addition, translations between wireless technologies, protocols and addressing schemes may have to be done, and the Unified registry facilitates the conversion.

9.3.4 Key Management

Resource-constrained IoT nodes across multiple protocols communicate by means of a shared key. Symmetric keys are used for simplicity and resource efficiency. Black Networks assume a shared secret. Key Management is a critical part of all security architectures and IoT nodes are vulnerable because of their limited resources. The highest vulnerability is at the time when an IoT node joins a network. In some cases keys are pre-flashed on the sensors – therefore the node is subject to being hijacked. In other cases, initial key exchange results in a MITM (man-in-the-middle) attack. Public key cryptographic methods are too resource-expensive for IoT nodes. In Section 9.2 we showed that multiple IoT protocols had vulnerabilities with key management. We propose external key management for each

layer of the communication protocol, instead of embedding this function in IoT protocols that are resource-constrained. Given multiple funtions, access technologies, protocols and node types, Smart Cities need a secure key management system for generating, distributing, storing, revocation and retiring keys for a wide variety of applications.

9.4 Evaluation of the Secure IoT Architecture

We evaluate our secure IoT Smart City architecture over each of the components. Table 9.1 shows the security services that each of the components adds to the basic security of IoT protocols. Starting with Black Networks, we note that simple broadcast routing of Black packets (where random nodes sleep between 0-90% of the time) is secure, but impractical as node reachability is very poor. A fully synchronized IoT network using an SDN Controller has more efficient routing for Black packets: a pre-determined route case where the Black packet is sent through nodes that are synchronized to sleep/wake to forward the packet; a dynamic node allocation for the next hop based on Onion Routing. The payload efficiency of Black frames and packets versus a regular Network layer packet needs to be performed for some popular IoT protocols. Black Networks always use the maximum length packet for communications, regardless of the actual payload size. This prevents packet-length based inference attacks. The performance of an IoT network with fixed length packets requires further analysis. We consider the combination of Black Networks and SDN Controller as a means to provide privacy and secure routing. The Unified Registry provides identity management and authentication during node join, as well as device id for key management. It further provides identity, authentication and authorization of services, if an IoT node, within the Smart City, is mobile. This allows for multiple device types to be monitored and authenticated. During IoT node failures, pre-authenticated nodes in the Unified Registry can be brought online to provide high-availability. A dedicated Key Management System eliminates vulnerabilities present in weak key management definitions, such as those described in Section 9.2. Embedding key management within the IoT protocol, provides for basic, symmetric key security services. For mission-critical data and services, we propose key management to be external and at each layer of the protocol stack. With these four components, we take a modular and practical approach to Smart City security, across multiple city functions.

Table 9.1: Secure IoT Smart City Architecture Services

IoT-based Smart City Security	
Security Component	*Security Services*
Black Networks	Confidentiality, Integrity, Privacy
SDN Controller	Secure Routing (Black packets), Availability
Unified Registry	Identity Management, Node Authentication, Authorization, Accounting, Availability and Mobility
Key Management	External Key Management

9.5 Conclusions and Future Research

Smart City IoT networks are increasingly widespread and carrying mission-critical data as they enable Smart Cities [123]. Smart City security is dependent on the underlying IoT protocols that have well-known vulnerabilities. Determined adversaries can launch co-ordinated cyber attacks on Smart Cities that have the potential for catastrophic damage. Our secure IoT Smart City architecture adds privacy (through Black Networks), identity management and authentication (by the Unified Registry), secure routing (via the SDN Controller) and a secure Key Management System. These four fundamental security architectural components can be deployed across all Smart City functions. Some areas of future research are detection of hardware Trojans in IoT nodes and extending Black Networks to Bluetooth Low Energy and WLANs.

Chapter 10

SECURE SMART CITY IoT SERVICES WITH DISTRIBUTED LEDGERS

The common smart city model is to have a centralized platform and physical location from where city services can be monitored and dispatched (a city operations center) [124]. While this centralized approach to smart cities is simple and efficient, it is risky, with minimal fault-tolerance and resilience [125]. From a cybersecurity perspective, such a centralized architecture, has potential for malfeasance and accidents. A centralized repository of smart city data and services can be breached and subject to loss of personal and private citizen data, or worse result in a catastrophic failure or shutdown of critical infrastructure. Is a resilient, secure, anonymous, distributed approach to smart city services possible? [126]

10.1 Introduction

Half of the worlds population resides in urban areas, meaning cities. Better opportunities, better healthcare and better facilities have contributed to the shift into urban areas [127]. City managers, having to deal with the influx of populations, a drive towards sustainable living have turned to ICT (Internet and Communications Technologies) to make cities 'smarter'. The proliferation of IoT (Internet of Things) devices and networks have led to IoT-enabled smart cities [118]. These smart cities also present a huge attack surface, due to IoT vulnerabilities [11]. Smart Cities face a threat of cyberattack that can disable critical infrastructure and citizen services [37].

Within a smart city context, IoT networks are heterogeneous, comprising of multiple technologies, hosting different applications. For secure smart cities, the communications must be secured at every layer of the communications protocol. Popular IoT communications protocols have several vulnerabilities [14], and existing standards may lack clear Key Management System (KMS) requirements and implementation mandates. Therefore external, automated key management is required to secure smart city IoT traffic, at every layer of

the communications protocol. Key Management is the most complex portion of a security solution, made even more vulnerable with large, heterogeneous deployments as in smart cities. KMS in IoT-enabled smart cities are based on the IoT communications protocol in use, such as IEEE 802.15.4 [39], ZigBee [26], 6LoWPAN [103], WirelessHART [12] and Bluetooth Low Energy [13] among others. Cloud-based KMS are a recent trend and may also be in use by smart cities. Public IoT networks, such as Ingenu, LoRaWAN and Sigfox will provide KMS within their own networks. All of the existing deployments are centralized KMS.

IoT network nodes are designated as fixed and immobile. However, a growing class of IoT devices are now mobile (such as personal health monitoring devices - that travel with the individual, and supply chain tracking). The nodes may roam between networks in a cluster, or between network clusters. In some cases, an IoT node may disassociate from a network (due to no activity, or extended sleep times), and rejoin after trigger event. All of these cases are equivalent: an IoT node is joining a network, and needs to be authenticated and authorized for services, within a reasonable time. Current IoT protocols not do not handle a roaming scenario, they only handle a node joining the network [128]. Again, IoT node authentication is a centralized function, based on existing standard protocols.

We propose a distributed model for both key management and mobile node authentication for smart city services using DLTs. Distributed ledger technologies (DLTs, or a Digital Ledger - DL) have been popularized by blockchains, used in the Bitcoin (and other) cryptocurrency. DLs are a shared, synchronized and distributed database, across a peer-to-peer network (of computers). The DL data structures, and associated consensus protocols remove the need for a centralized management control, by offloading that capability to network nodes. With no centralized control, many IoT communications protocol vulnerabilities are mitigated (e,g, the ZigBee protocol single Trust Center for key management in the network, or the WirelessHART Security Manager for key management that is not defined in the standards). The resulting decentralized key management and authentication services should be automated and enable a secure and private smart city.

The remainder of this chapter is organized as follows: In Section 10.2, we present related work of KMS and mobile node authentication, for IoT networks and smart cities. In Section 10.3 we review and distributed ledger technology (DLT). In Section 10.4 we present

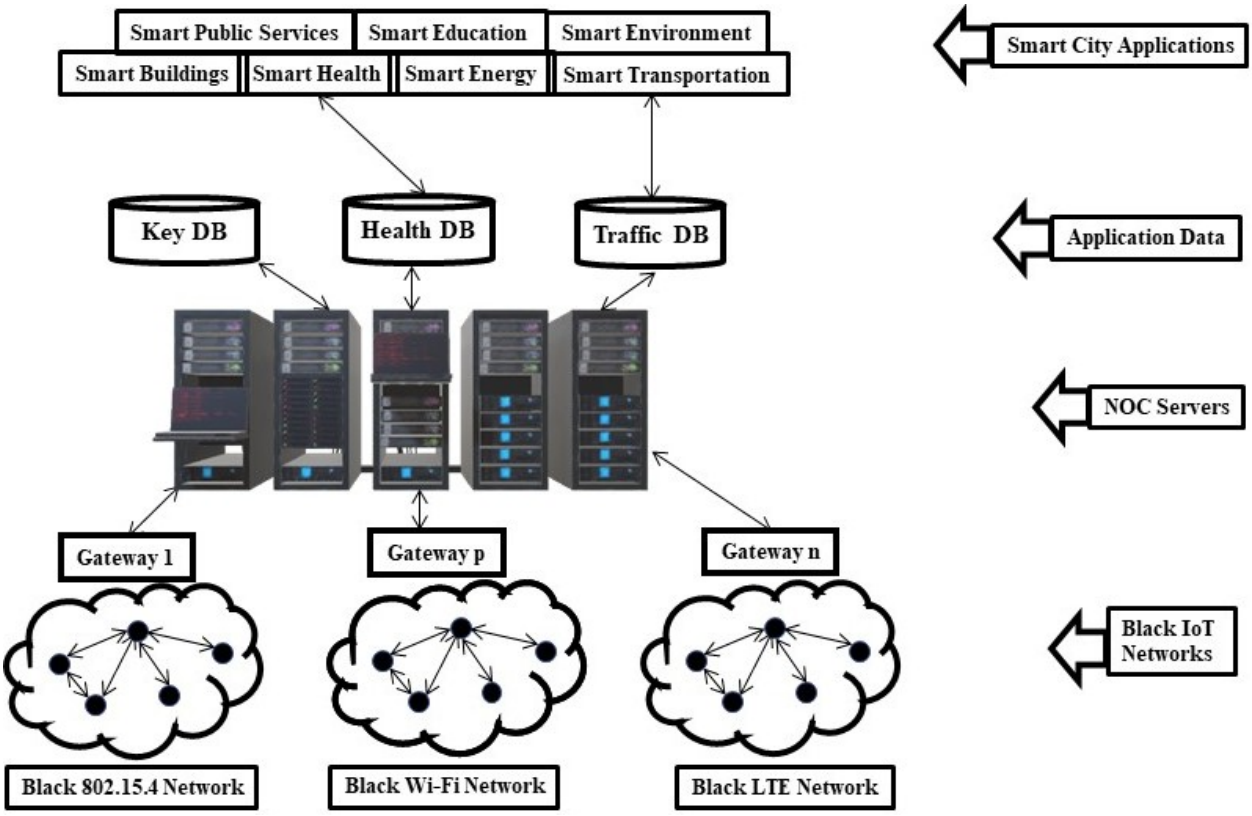


Figure 10.1: Smart City Centralized Applications

distributed KMS using DLTs for smart cities. In Section 10.5, we present the use of DLTs for distributed mobile node authentication, for IoT networks in smart cities. We analyze the threat models and security for DLT-based KMS and mobile node authentication for smart cities in Section 10.6, and draw relevant conclusions and identify future areas of research in Section 10.7.

10.2 Related Work in IoT and Smart Cities

Figure 10.1 illustrates an example of a common smart city deployment today. There are heterogeneous IoT networks (IEEE 802.15.4, and its commonly deployed upper layer protocols - ZigBee, 6LoWPAN, WirelessHART; WiFi - IEEE 802.11ah, and LTE-M (Long Term Evolution-Machine) or NB-IoT (Narrowband Internet of Things)) from public wireless providers, as examples), managed by communications protocol specific gateways. This means

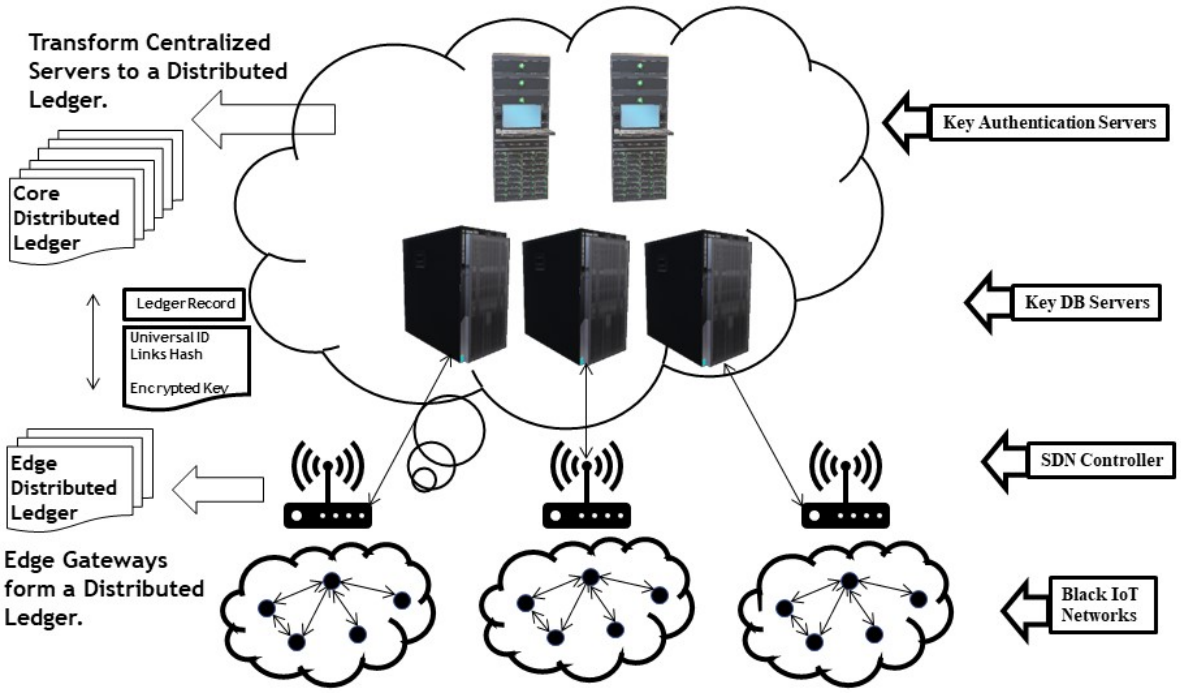


Figure 10.2: Key Management with Distributed Ledger in Smart Cities

the gateway for the LTE IoT network can handle LTE-M traffic and transfer it to the NOC (Network Operations Center) servers.

The IoT networks shown in Fig. 10.1 are Black networks. Black networks enable secure smart cities by securing both the data and the metadata in the communications protocol, which remaining compatible with the existing protocol [13]. Black networks use an SDN-based (Software Defined Networks) architecture to forward packets with encrypted metadata, using a ciphertext-based forwarding algorithm [14]. Black networks may also employ node obscuring mechanisms to hide the communicating nodes, while routing in a mesh network [101]. Securing the communications protocol is one way to ensure IoT-enabled secure smart cities. A secure smart city architecture may include other components such as Key Management, Unified registry (for authentication) and an SDN architecture (to enable Black networks) [11].

The gateways from multiple networks aggregate collected data and send it to NOC platform for storage and analysis. Smart City applications use this data and provide a broad

range of new and existing citizen services. Typically, the smart city applications also reside in the NOC. The applications are accessed, the data monitored and the citizen services dispatched, from a "smart city operations center" (SCOC). This centralized architecture is in common use for a practical and speedy deployment of services. However, this centralized configuration is susceptible to malicious and natural attacks. In the event of a natural disaster (though SCOCs are well secured against natural disasters and co-ordinate emergency dispatch), the access to smart city services may be lost. A sophisticated and well-funded nation state actor may mount an attack against the SCOC and data centers to disable smart city services and critical infrastructure [129] or generate false alarms that can lead to emergency services being overwhelmed [130]. Therefore, a decentralized approach is suggested. We look at two critical functions to decentralize for IoT-enabled secure smart cities - key management system and mobile node authentication.

10.2.1 External Key Management

In IEEE 802.15.4, key management is provided by the upper layers of the protocol. If keys are not provided or requested, then there is no default security or key management [10]. The KMS in ZigBee is performed by a software application called the Trust Center. The single Trust Center per network is a vulnerability, as is the requirements specifications for the Trust Center in the standards (which could lead to insecure implementations). The Trust Center link key is publicly known and maybe used for encrypting the network key (key for ALL ZigBee nodes), leading to an exploit [26]. 6LoWPAN uses IPsec for its security and key management, but does not mandate its use in the standards. IPsec is resource intensive (uses PKC) and impractical for use in IoT networks [103]. WirelessHART key management is performed by the Security Manager module, which is not defined in the WirelessHART standard, leading to compliant, but insecure implementations [12]. BLE key management is done at the host, via pairing and bonding, for communicating a shared secret, which can lead to meet-in-the-middle attacks, in addition to eavesdropping on advertising channels, track and trace and packet modification [13]. Key management vulnerabilities in IoT protocols have been extensively researched [26] [54] [12] [10], and the complexity of heterogeneous IoT systems in smart cities increases the risk of cyberattack [131].

Alternatively, key management may be provided by a cloud service, and offered by several cloud vendors. For complex heterogeneous and hybrid environments like smart cities, key management takes on additional complexity [132]. Such a service requires a secure channel from the cloud to the communicating parties (node, gateway). Khan [133] proposes a KMS in the cloud, as part of a security and privacy framework for smart cities. Eshenauer et. al [134] proposes a pre-shared distribution mechanism, for sensor networks, that uses a small set of pre-distributed keys (ring of keys), from a pre-generated pool of keys, for probabilistic key sharing between nodes of a random graph. Re-keying, revocation and key distribution are efficient [135]. Roman [136] extensively evaluates KMS for IoT, at both the network and the link layer, for multiple KMS frameworks, across public key and pre-shared key scenarios. Standards and Alliances are collaborating for a comprehensive and consistent KMS approach for smart cities that does not solely depend on vulnerable IoT protocols [137] [119]. A unifying theme is a centralized approach to key management, as well as identification of the limitations of public key exchange for IoT networks. IoT networks enabling smart cities are resource-constrained, therefore standardized PKCs (Public Key Cryptosystems) are unsuitable for IoT. Our goal is to provide a secure distributed and external KMS, for IoT, within a secure smart city, so as to mitigate centralized and IoT vulnerabilities.

10.2.2 Mobile Node Authentication

IoT networks contain nodes that are increasingly mobile, and require to be efficiently authenticated, when 'roaming', in both 'serving' networks and the mobile node's 'home' network (We borrow this nomenclature from mobile telephony - the home network is the network in which the node was originally provisioned and authenticated; the serving network is where the mobile IoT node has now joined; the IoT node is said to be 'roaming' or 'visiting' the serving network, since it is not originally part of that network). When joining a network for the first time, all nodes have to be authenticated.

IoT nodes have additional characteristics of sleep/wake cycles to preserve energy (as they are often powered by batteries). Such a combination of mobility and sleep cycles may cause the node to dissociate from the network - either because of extended sleep cycles, or because the node traveled to another network. In both cases the node has to be re-

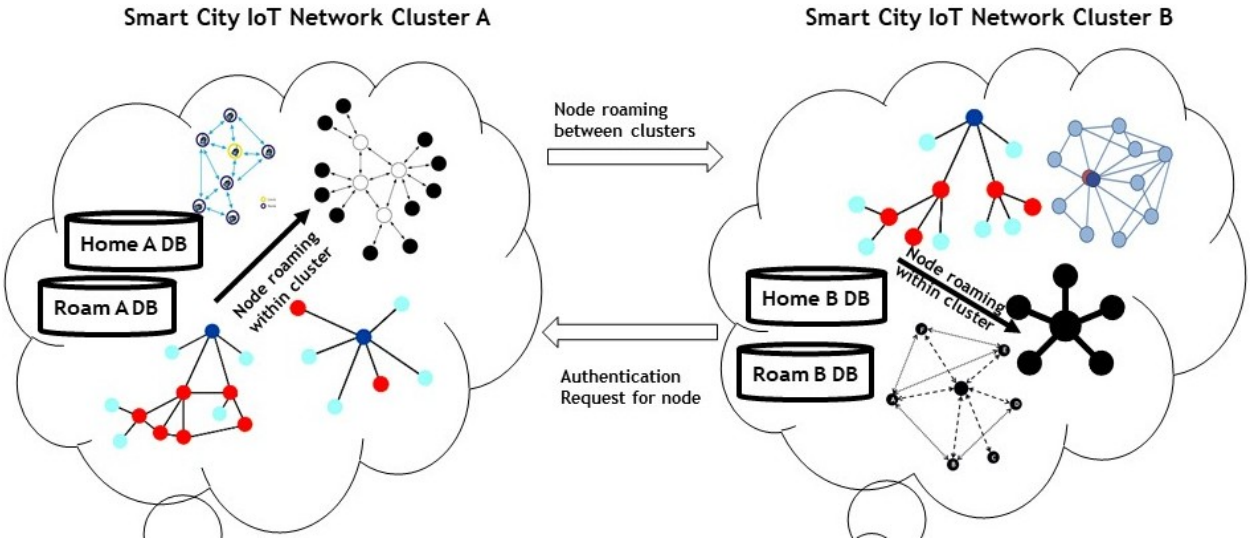


Figure 10.3: IoT mobile node roaming architecture: Intra-cluster and Inter-cluster

authenticated. With billions of nodes in deployment, and thousands within a given network, combined with the heterogeneous nature and low cost of IoT nodes, make a centralized authentication system cost-prohibitive and complex. Authentication mechanisms within IoT protocols do not manage, or account for mobility. As a result, node dissociating from ZigBee or 6LoWPAN networks contain the network key, which can be extracted and re-used by a malicious node [81] [54].

We propose a distributed authentication mechanism, using DLTs, key management and Black networks for secure authentication of mobile IoT nodes.

Figure 10.3 shows a network view of smart city IoT clusters, in which a mobile node is roaming between inter-cluster heterogeneous networks, and intra-cluster networks (depicted by arrows). Network clusters are a group of networks with similar functions, or a group of networks in a given geography. Once the mobile network roams into IoT network cluster B as shown in Figure 10.3, it requests to operate within a network in Cluster B, in which it has to be authenticated. Based on the request, which includes its home information (Cluster A), Node ID and a challenge. Cluster B, sends an authentication request to the Home A DB (database), with all of the received information. The Home A DB in Cluster A confirms the identity of the mobile IoT node and returns a challenge-response to the network in Cluster B

in which the mobile IoT node wants to join. The network in Cluster B in which the mobile node is roaming, then sends the challenge question to the mobile node, and upon receiving the correct response, updates the Roam B DB with the visiting node information, along with authorization of services.

Two databases (Home and Roam) are used to track mobile IoT nodes, within a cluster. Originally provisioned IoT nodes are listed in the Home DB, whereas roaming IoT nodes are listed in the Roam DB. These databases are centralized within a cluster, and all network joining requests will be checked against the these servers, for node authentication, authorization, accounting and privacy. Large volumes of mobile IoT nodes will cause network performance degradation. Services will be disrupted, in the event of a database failure.

10.3 Distributed Ledger Technologies (DLTs)

Distributed Ledger technologies (DLTs, or a Distributed Ledger - DL) have been popularized by Blockchain, used in the Bitcoin cryptocurrency. DLs are a shared, synchronized and distributed database, across a peer-to-peer network (of computers). The 'database' is a linked list with a Merkle-tree record structure, where the individual records cannot be changed, and the record updates, and/or new records are merely added to the linked list, providing a change history. The updates (across the network, to all nodes) are conducted by a consensus protocol. This flooding update approach, in a peer-to-peer network, precludes the need for a central authority, or a centralized database, and is the basis of DLT popularity in many industries (starting with the financial industry) [138]. There are four primary categorizations of DLTs: permissionless public, permissioned public, permissionless private and permissioned private. The public and private refer to who can read access the DL - systems where anyone can access the DL, are public, whereas consortiums/alliances of enterprises that prefer a closed group to access the DL, are private. Permissioned and permissionless refer to parties that can update the DL. Permissionless DLs are those in which anyone can form a node, run a consensus protocol and update the DL, whereas permissioned DLs allow have access control on who can update the DL. The consensus mechanism used is based on the category of DLT (e.g. permissionless public DLT uses the PoW (proof of work)-based protocol; while permissioned public DLTs use proof-of-stake consensus protocol). Other consensus

protocols include leader-based, voting-based, economy-based and virtual-voting-based [139]. IOTA is a DL platform designed specifically for the IoT, and enables machine-to-machine micropayment transfer [140]. The directed acyclic graph for storing the transactions is called Tangle. Markov Chain Monte Carlo algorithms are used to compute attach points for new transactions [141]. A literature review of DLTs (referred to as blockchains) for IoT use cases concludes that IoT networks are ill-suited for blockchains and DLTs, given resource restrictions in IoT networks. Therefore a specialized DLT architectures are needed to scale with IoT [142].

10.4 Distributed Key Management with DLTs

Figure 10.2 shows a traditional, centralized key management system (KMS), that would be commonly deployed in a smart city scenario. The SDN Controllers perform the KMS for the IoT networks that they manage. The Key Database Servers generate and store the keys. When keys are requested by an IoT node, the SDN controller checks if it has a pool to distribute keys from. If not, then it requests the Key DB Server for a set of keys (depending on the function). The Key DB may authenticate the request from the Key Authentication server, prior to releasing the key set. The secure SDN controller has a secure execution module within it for the keys. Any DDoS attacks on the links, the Key DB or the Key Authentication servers will cause this centralized configuration to fail. To distribute the risk, we put key rings (sets) [134], in a permissioned, private DL. The core functionality (designated by the functions inside the cloud, Fig. 10.2), would be called the Core DL. Likewise we employ the same principle to the SDN Controller layer, and call it the Edge DL. The Core DL and the Edge DL can communicate with each other, and exchange information and update each other. This portion of the communication can be performed using PKC - given the resources of SDN Controller and the servers. An example of a DL record is also shown in Fig. 10.2. The DL record has a standard format, except that a portion of it is encrypted (the actual keys are encrypted). In the event, an adversary gets past the system to the ledger, they would not be able to access the keys. What key is used to encrypt the portion of the DL? Who will unlock the records when keys are needed. The consensus protocol will allocate a timed decryption key based on agreement. With this distributed

system, the key distribution can be performed faster and closer to the point of requirement instead of sending all information back to the centralized servers. An example would be as follows: An IoT node requests a session key. It sends its request to the Edge DL. The Edge DL checks if it can satisfy the key request, by using a consensus protocol, amongst all the SDN Controllers. If a key set is available for the IoT node, then it is sent to the node. If a key set is not available for the IoT node, then the Edge DL communicates with the Core DL to obtain a key for the IoT node. The Core DL runs its consensus protocol, obtain the key set, and updates the Core DL and the Edge DL, and sends the key set to the Edge DL. The communication between the Core DL and the Edge DL is via PKC. The communication between the SDN Controllers, may also be PKC. However, between the SDN Controller and the IoT node the communication is symmetric key. While a DL-based architecture is very resilient (a DDoS attack will have to be initiated against ALL nodes of the Core and the Edge DL), the actual DL implementation, architecture and benefits will have to be understood as a further area of research.

10.5 Mobile Node Authentication with DLTs

Fig. 10.3 displays mobile IoT nodes roaming, within a cluster, and across clusters. A cluster is a region with multiple heterogeneous IoT networks. Each cluster manages its IoT networks and nodes via a Home database and a Roam database. The Home database contains the information of all provisioned IoT devices, within that cluster. The Roam database comprises of information of mobile IoT nodes that are roaming within that cluster (as in Home A DB and Roam A DB). The objective is to put the Home and Roam DBs into a DL. That way the DBs can be replicated and distributed for faster access, in the event of an authentication request. How is a DL faster and more reliable for authentication than a specialized central server? Firstly, the number of nodes per cluster will evolve to the thousands. With heterogeneous networks, multiple device types will have to be provisioned. This would be the Unified Registry conceptualized in [101]. Next there should be some roaming agreements between clusters - meaning there has to be a way for Cluster B to recognize a roaming IoT mobile node, and communicate with other Home and Roam DBs. Finally, at authentication, it should be known, if an IoT node simply disassociated from the

network or roamed to a new network. How would the network decide if a node was legitimate or malicious?

Based on our literature survey, the DL cannot be within the IoT nodes themselves, but must reside on the SDN Controller (SDNC/Gateway). If each SDNC manages the devices associated with it, then we have inherently distributed the Home and Roam DBs and provided a simplified solution. An SDNC failure limits the network outage to a small number of nodes only connected to that SDNC. Heterogeneous networks are not easily integrated into a single generic DB, and are difficult to manipulate subsequently. If the Home and Roam is on a DL, a simple consensus will determine if a mobile IoT node is roaming, or network disassociated. In the latter case, subsequent actions of removing the network key, from the disassociated node should result. Else an adversary can extract the key and introduce a malicious node (insider threat). Finally, a request needs to come to any one SDNC within a cluster to authenticate a roamer. The DL residing in the SDNC will have a list (or a partial list of all authenticated nodes). Roaming agreements are not needed, and a simple peer-to-peer request is made for authentication. As a precautionary measure, we make this DL permissioned and private. The DL resides on the SDNCs and provides a simple peer-to-peer authentication for a mobile IoT node. We reiterate, that much malicious activity occurs when there is no mutual authentication of parties - while the network authenticates the node, the node does not authenticate the network. We highly recommend that the mobile IoT node authenticate the network in which it is roaming. Malicious networks will draw mobile IoT nodes in and obtain critical information to follow up with a masquerade attack, if necessary.

10.6 Threat Models and Security Analysis

There is ongoing research on the applicability of DLTs to IoT networks. Kshetri [143] points out the centralized nature of IoT networks, along with possible threat models, such as IAM (Identity and Access Management), cloud availabilty and reliability and supply chain security. The identification and authentication of IoT devices may be stored in a DL. With billions of IoT nodes coming on-line, the centralized cloud model will lead to communication bottlenecks. Attacks and system failures on the cloud render the entire IoT network at risk,

and centralized cloud data store is susceptible to manipulation. The security of the supply chain is a generic use case with DLs, and can be applied to IoT security to ensure integrity during the history of the IoT hardware and software. These attributes may be used for mobile IoT node authentication. In general, the threat model for key management in IoT networks is the attack on a *network/master* key. In addition, an attack on an IoT node, or on IoT communications to extract keys. If an IoT node disassociates from a network, due to sleep cycles or the IoT node is captured, it contains the network key, which can be extracted. Malicious nodes can join the network and request keys. The pre-shared key in this case is standards-defined and is known to all 777 772E 6861 7274 636F 6D6D 2E6F 7267 [12].

Similarly for IoT mobile node authentication, a request to a network for joining maybe rejected, if the node cannot be authenticated by its home network. Outside of cellular IoT protocols, the mechanism for nodes to roam into another IoT network and request an obtain services does not exist. A malicious node could to roam into a network and request service giving the credentials of legitimate home network. We note that it is not sufficient for the mobile IoT node to be authenticated in the visting IoT network. The mobile IoT node must also authenticate the IoT network in which it is requesting services.

10.7 Conclusions and Future Research

The rapid growth of IoT-enabled smart cities have resulted in a centralized model. Smart City data is acquired via IoT networks and fed into a central platform for analysis and action. A central physical location serves as a Smart City Operations Center, into which data is fed and from which services are dispatched. This Smart City model is vulnerable to attack and accident. We propose a distributed model to deliver two Secure Smart City services - key management and mobile IoT node authentication. Distributed ledger technology (DLT) is used to deliver these services. IoT communication protocols have weak or unspecified key management systems, and do not handle node mobility and authentication in a visiting network. We propose a DLT-based architecture for key management, and to authenticate mobile IoT nodes joining another network. Our architecture allows secure key management, and secure roaming of IoT mobile nodes to other IoT networks.

Chapter 11

CONCLUSIONS

This dissertation proposes a novel approach to securing the metadata in communications protocols. The research for this approach targeted communications protocols for the Internet of Things (IoT), and applies them to the multidimensional domain of Smart Cities. IoT is becoming pervasive in our daily lives, across a multitude of industries, such as healthcare, transportation, energy and manufacturing. IoT provides a huge attack surface, is beginning to carry mission-critical data, and is coming to the mainstream with large-scale cyber attacks [144] using the IoT. IoT protocols have well-researched vulnerabilities that can be exploited (Table 1.1). A security survey of the most commonly used IoT communications protocols (IEEE 802.15.4, ZigBee, 6LoWPAN, WirelessHART and Bluetooth Low Energy) indicate that metadata is NOT secured for any of these IoT protocols, and is easy to exploit.

The dissertation presents Black packet designs for IoT communications protocols - for IEEE 802.15.4, ZigBee, 6LoWPAN, BLE and IPv6 (broadband protocol). The Black packet structure, for each communications protocol is presented and its compatibility with the existing IoT protocol in use (for example Black ZigBee is compatible with ZigBee) is shown. Black packets protect the metadata.

With the metadata secured, simple communications between IoT nodes in a network is presented. A network carrying Black packets is a Black network. A majority of IoT networks today work in the star network configuration, and we present a Black Gateway configuration, with the Black Gateway at the center of the star network, connected to all the nodes. We compare the Black Gateway configuration to Shortest Path routing, Broadcast routing and Flooding. Black Gateway communication has significantly higher security, prevents a broad range of active and passive attacks, with a performance equivalent to, or better than, Shortest Path routing. Flooding and Broadcast communications have significantly higher overhead traffic, Mean Wait and Mean Travel times in Black networks. Performance measurements

with Mean Wait, Mean Travel and Overhead Traffic, is conducted via extensive simulations, of multiple networks, with increasing number of nodes (10 to 1000 nodes, 15000 data points per node set).

With simple Black networks communications established and simulated, we turned to mesh networks and Black routing. How can a Black packet be routed? (i.e. How does one route a packet whose header is encrypted?). For that, we turned to an emerging routing paradigm - Software Defined Networks (SDN). In SDNs, a centralized open-API, SDN controller downloads routing information to a network of nodes. The nodes forward a packet according to the routing table entries and actions associated with it. We adapt a simplified version of SDN to a wireless IoT network and present a Black SDN architecture for routing Black packets (called Black routing). Using the Black SDN architecture, we demonstrate the feasibility of Black routing from source to destination, with encrypted metadata. The dissertation asserts guaranteed delivery, for IoT nodes with a sleep/wake cycle (as is common among IoT nodes for power savings), for two configurations: Star Control and Mesh Control.

With the network architecture to support Black routing established, we present Black routing algorithms for synchronous and asynchronous updates, with ad-hoc and pre-determined routes, using the Mesh Control SDN architecture. The dissertation presents extensive simulation results over a 1000-node Barabási network, using a simulation model with 10,000 data points per node set (10 networks, 1000 flows per network). On comparing with Shortest Path routing, we note that Black routing offers significant resistance to attacks, and comparable performance to Shortest Path routing, with trade-offs on overhead traffic, wait and travel times. E.g. Black routing overhead traffic is better for nodes $<$450, and can reach a maximum of 45% with a 1000-node network, when compared to Shortest Path routing.

We also note that securing the metadata is not sufficient to hide communicating parties, from an adversary conducting sustained traffic analysis, in a network. While Black networks hide all details of the communications, nodal transmissions and receptions can be observed, and inferences on the traffic type and communicating parties can be made. We present node obscuring to hide the communicating parties. We use the concept of *subway communications* and *tokens* to obscure the sender and the receiver. We present 2 approaches for node obscuring - a linear approach where tokens start from Node 1 and finish at Node N

(last node), and one where the network is divided into row and column subnets, with tokens within those subnets. We present algorithms for both node obscuring methods (called Node Obscuring-Linear, NO-l; and Node Obscuring-Grid, NO-g). While node obscuring offers a significant improvement in nodal communications security and privacy and incurs negligible overhead traffic, there is a tradeoff with high wait and travel times. We propose node obscuring algorithms with the reduction of wait and travel times as an open area of research.

Finally, we apply the Black networks as a use-case to the multi-dimensional domain of Smart Cities. Smart Cities are large, complex, diverse and non-stop systems, that are increasingly being enabled by the IoT ecosystem. IoT networks in smart cities will carry mission-critical data and controls. In the new era of cyberwarfare and cyberterror, adversaries are seeking to cripple cities by launching attacks on critical-infrastructure [37]. The dissertation presents a secure IoT architecture for smart cities, basing all IoT communications on Black networks with a Unified Registry for authentication and IoT Key Management to support heterogenous networks and devices. Most Smart Cities today are based in a centralized model. Data is gathered by IoT networks and sent to a central repository. The data is analyzed and services dispatched from a central location, typically a SCOC (Smart City Operations Center). This NOC-based centralized approach is vulnerable to accidents and to malicious attacks. Downing the SCOC can disable a Smart City, or lead to degradation of services. To further improve security and availability, this dissertation proposes a distributed architecture, based on distributed ledger technology (DLTs), to deliver the Secure Smart City services of key management and mobile node authentication.

BIBLIOGRAPHY

[1] Libelium. Libelium smart world [online]. http://www.libelium.com/libeliumworld/case-studies/. xiv, xvi, 23, 126

[2] Zach Shelby and Carsten Bormann. *6LoWPAN: The wireless embedded Internet*, volume 43. John Wiley & Sons, 2011. xiv, 33, 47, 50

[3] "zigbee specification". Technical report, ZigBee Alliance, Jan 2008. xiv, 22, 35, 72, 125

[4] Jianping Song, Song Han, Aloysius K Mok, Deji Chen, Mike Lucas, and Mark Nixon. WirelessHART: Applying wireless technology in real-time industrial process control. In *Real-Time and Embedded Technology and Applications Symposium, 2008. RTAS'08. IEEE*, pages 377–386. IEEE, 2008. xiv, 38, 39, 66, 74

[5] Thomas D Nadeau and Ken Gray. *SDN: Software Defined Networks*. O'Reilly Media, Inc., 2013. xiv, 45, 93

[6] Paul Goransson and Chuck Black. *Software Defined Networks: A Comprehensive Approach*. Elsevier, 2014. xiv, 45, 46, 93

[7] Tie Luo, Hwee-Pink Tan, and Tony QS Quek. Sensor openflow: Enabling software-defined wireless sensor networks. *Communications Letters, IEEE*, 16(11):1896–1899, 2012. xiv, 9, 46, 48, 49, 50, 94, 106, 133

[8] Raj Jain. Networking protocols for internet of things. *Coursework [Online]. http://www.cse.wustl.edu/~jain/cse570-13/ftp/m_19lpn.pdf*, 2013. xiv, 51

[9] HARTcomm.org. Wirelesshart [online]. http://en.hartcomm.org. xiv, 22, 56, 66, 72, 74, 125

[10] Naveen Sastry and David Wagner. Security considerations for IEEE 802.15. 4 networks. In *Proceedings of the 3rd ACM workshop on Wireless security*, pages 32–42. ACM, 2004. 1, 32, 72, 74, 129, 131, 141

[11] Shaibal Chakrabarty and Daniel W. Engels. A Secure IoT Architecture for Smart Cities. In *Consumer Communications and Networking Conference (IEEE CCNC), Oct 2015 IEEE Conference on*. IEEE, 2016. 1, 71, 137, 140

[12] Shahid Raza, Adriaan Slabbert, Thiemo Voigt, and Krister Landernas. Security considerations for the WirelessHART protocol. In *Emerging Technologies & Factory*

Automation, 2009. ETFA 2009. IEEE Conference on, pages 1–8. IEEE, 2009. 1, 39, 40, 72, 74, 115, 129, 131, 138, 141, 148

[13] Shaibal Chakrabarty and Daniel W. Engels. Black Networks for Bluetooth Low Energy. In *Consumer Electronics (ICCE), Jan 2016 IEEE International Conference on*. IEEE, 2016. 1, 66, 68, 74, 76, 88, 89, 105, 138, 140, 141

[14] Shaibal Chakrabarty, Daniel W. Engels, and Selina Thathapudi. Black SDN for the Internet of Things. In *Mobile Ad Hoc and Sensor Systems (IEEE MASS), Oct 2015 IEEE International Conference on*. IEEE, 2015. 2, 3, 5, 9, 10, 59, 60, 68, 75, 103, 105, 106, 137, 140

[15] Pedram Radmand, Alex Talevski, Stig Petersen, and Simon Carlsen. Taxonomy of wireless sensor network cyber security attacks in the oil and gas industries. In *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*, pages 949–957. IEEE, 2010. 3, 21, 22, 72, 74, 102

[16] Ying Dong, Tat Wing Chim, Victor OK Li, Siu-Ming Yiu, and CK Hui. Armr: anonymous routing protocol with multiple routes for communications in mobile ad hoc networks. *Ad Hoc Networks*, 7(8):1536–1550, 2009. 3

[17] Yanchao Zhang, Wei Liu, Wenjing Lou, and Yuguang Fang. Mask: anonymous on-demand routing in mobile ad hoc networks. *IEEE transactions on wireless communications*, 5(9), 2006. 3

[18] Lanjun Dang, Jie Xu, Hui Li, and Nan Dang. Dasr: distributed anonymous secure routing with good scalability for mobile ad hoc networks. In *Services Computing Conference (APSCC), 2010 IEEE Asia-Pacific*, pages 454–461. IEEE, 2010. 3

[19] Ronggong Song, Larry Korba, and George Yee. Anondsr: efficient anonymous dynamic source routing for mobile ad-hoc networks. 2005. 3

[20] Wei Liu and Ming Yu. Aasr: authenticated anonymous secure routing for manets in adversarial environments. *IEEE transactions on vehicular technology*, 63(9):4585–4593, 2014. 3

[21] Yang Qin, Dijiang Huang, and Vinayak Kandiah. Olar: On-demand lightweight anonymous routing in manets. In *Proc. Fourth Int'l Conf. Mobile Computing and Ubiquitous Networking (ICMU'08)*, pages 72–79. Citeseer, 2008. 3

[22] Denh Sy, Rex Chen, and Lichun Bao. Odar: On-demand anonymous routing in ad hoc networks. In *2006 IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, pages 267–276. IEEE, 2006. 3

[23] Zhiguo Wan, Kui Ren, and Ming Gu. Usor: An unobservable secure on-demand routing protocol for mobile ad hoc networks. *IEEE transactions on wireless communications*, 11(5):1922–1932, 2012. 3

[24] Muthumanickam Gunasekaran and Kandhasamy Premalatha. Teap: trust-enhanced anonymous on-demand routing protocol for mobile ad hoc networks. *IET Information Security*, 7(3):203–211, 2013. 3

[25] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. Technical report, DTIC, 2004. 4, 97, 114, 133

[26] Ender Yüksel, Hanne Riis Nielson, and Flemming Nielson. Zigbee-2007 security essentials. In *Proc. 13th Nordic Workshop on Secure IT-systems*, pages 65–82, 2008. 5, 34, 37, 72, 74, 131, 138, 141

[27] Ben Greenstein, Damon McCoy, Jeffrey Pang, Tadayoshi Kohno, Srinivasan Seshan, and David Wetherall. Improving wireless privacy with an identifier-free link layer protocol. In *Proceedings of the 6th international conference on Mobile systems, applications, and services*, pages 40–53. ACM, 2008. 5, 60, 65, 74, 105

[28] George Margelis, Robert Piechocki, Dritan Kaleshi, and Paul Thomas. Low throughput networks for the iot: Lessons learned from industrial implementations. In *Internet of Things (WF-IoT), 2015 IEEE 2nd World Forum on*, pages 181–186. IEEE, 2015. 6

[29] Usman Raza, Parag Kulkarni, and Mahesh Sooriyabandara. Low power wide area networks: An overview. *IEEE Communications Surveys & Tutorials*, 19(2):855–873, 2017. 6, 25, 27, 28

[30] Diego Kreutz, Fernando MV Ramos, PE Verissimo, C Esteve Rothenberg, Siamak Azodolmolky, and Steve Uhlig. Software-defined networking: A comprehensive survey. *proceedings of the IEEE*, 103(1):14–76, 2015. 9, 93, 133

[31] Salvatore Costanzo, Laura Galluccio, Giacomo Morabito, and Sergio Palazzo. Software Defined Wireless Networks: Unbridling SDNs. In *Software Defined Networking (EWSDN), 2012 European Workshop on*, pages 1–6. IEEE, 2012. 9, 94, 106

[32] Olivier Flauzac, Carlos Gonzalez, Abdelhak Hachani, and Florent Nolot. Sdn based architecture for iot and improvement of the security. In *Advanced Information Networking and Applications Workshops (WAINA), 2015 IEEE 29th International Conference on*, pages 688–693. IEEE, 2015. 9, 105, 106

[33] J Bradley, J Barbier, and D Handler. The internet of everything— a $ 19 trillion opportunity. *White Paper[Online].* *http://www.cisco.com/web/about/ac79/docs/innov/IoE_Economy.pdf*, 2013. 13, 21, 22

[34] Isaac Brown. A Detailed Breakdown of LPWAN Technologies and Providers. Technical report, Lux Research Inc. 13, 23, 26, 27

[35] UNEP DESAP. World urbanisation prospects: the 2014 revision, highlights (st/esa/ser. a/352). department of economic and social affairs. *Population Division, New York: United Nations*, 2014. 16

[36] Rida Khatoun and Sherali Zeadally. Smart cities: concepts, architectures, research opportunities. *Communications of the ACM*, 59(8):46–57, 2016. 16

[37] Cesar Cerrudo. An emerging us (and world) threat: Cities wide open to cyber attacks. *Securing Smart Cities*, 2015. 17, 137, 151

[38] John H Davies. *MSP430 microcontroller basics*. Elsevier, 2008. 21

[39] IEEE Standard for Local and metropolitan area networks Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs). IEEE Standards Association, IEEE Computer Society, June 2011. 22, 31, 32, 48, 74, 102, 125, 138

[40] C. Schumacher N. Kushalnagar, G. Montenegro. IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals. RFC 4919, RFC Editor, August 2007. 22, 32, 68, 72, 125

[41] IEEE 802.15.1-2005 - IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 15.1a: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Wireless Personal Area Networks (WPAN). IEEE Standards Association, IEEE Computer Society, June 2005. 22, 28

[42] Marco Centenaro, Lorenzo Vangelista, Andrea Zanella, and Michele Zorzi. Long-range communications in unlicensed bands: The rising stars in the iot and smart city scenarios. *IEEE Wireless Communications*, 23(5):60–67, 2016. 24

[43] Juan Carlos Zuniga and Benoit Ponsard. Sigfox system description. *LPWAN@ IETF97, Nov. 14th*, 2016. 25

[44] Link Labs. Low Power, Wide Area Networks: A Comprehensive Look. Technical report, Link Labs, June 2018. 25

[45] 802.15.4k-2013 - IEEE Standard for Local and metropolitan area networks– Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)–Amendment 5: Physical Layer Specifications for Low Energy, Critical Infrastructure Monitoring Networks. IEEE Standards Association, IEEE Computer Society, June 2013. 25, 28

[46] Aloÿs Augustin, Jiazi Yi, Thomas Clausen, and William Mark Townsley. A study of lora: Long range & low power networks for the internet of things. *Sensors*, 16(9):1466, 2016. 25

[47] Ferran Adelantado, Xavier Vilajosana, Pere Tuset-Peiro, Borja Martinez, Joan Melia-Segui, and Thomas Watteyne. Understanding the limits of lorawan. *IEEE Communications magazine*, 55(9):34–40, 2017. 26

[48] LoRa-Alliance-Technical-committee et al. Lorawan regional parameters. *LoRa Alliance, San Ramon, CA, USA, Tech. Rep. Version*, 1.1rB, 2018. 26

[49] GSMA (GSM Association). Mobile iot rollout report. https://www.gsma.com/iot/mobile-iot-commercial-launches/. 26

[50] Y-P Eric Wang, Xingqin Lin, Ansuman Adhikary, Asbjorn Grovlen, Yutao Sui, Yufei Blankenship, Johan Bergman, and Hazhir S Razaghi. A primer on 3gpp narrowband internet of things. *IEEE Communications Magazine*, 55(3):117–123, 2017. 27

[51] Jean-Paul Bardyn, Thierry Melly, Olivier Seller, and Nicolas Sornin. Iot: The era of lpwan is starting now. In *European Solid-State Circuits Conference, ESSCIRC Conference 2016: 42nd*, pages 25–30. IEEE, 2016. 27

[52] Mads Lauridsen, István Z Kovács, Preben Mogensen, Mads Sorensen, and Steffen Holst. Coverage and capacity analysis of lte-m and nb-iot in a rural area. In *Vehicular Technology Conference (VTC-Fall), 2016 IEEE 84th*, pages 1–5. IEEE, 2016. 27

[53] Rashmi Sharan Sinha, Yiqiao Wei, and Seung-Hoon Hwang. A survey on lpwa technology: Lora and nb-iot. *Ict Express*, 3(1):14–21, 2017. 27

[54] S Park, K Kim, W Haddad, Samita Chakrabarti, and Julien Laganier. IPv6 over low power WPAN security analysis. Technical report, IETF Internet Draft draft-6lowpan-security-analysis-05, 2011. 31, 33, 72, 74, 115, 129, 131, 132, 141, 143

[55] Frank Stajano and Ross Anderson. The resurrecting duckling: security issues for ubiquitous computing. *Computer*, 35(4):22–26, 2002. 31, 72, 102, 115, 131

[56] Yang Xiao, Hsiao-Hwa Chen, Bo Sun, Ruhai Wang, and Sakshi Sethi. MAC security and security overhead analysis in the IEEE 802.15. 4 wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 2006, 2006. 32, 74

[57] Jakob Jonsson. On the security of CTR+ CBC-MAC. In *selected Areas in Cryptography*, pages 76–93. Springer, 2003. 32, 39, 74

[58] P. Thubert J. Hui. Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks. RFC 6282, RFC Editor, September 2011. 32, 72, 102

[59] Z Shelby, S Chakrabarti, E Nordmark, and C Bormann. Neighbor discovery optimization for IPv6 over low-power wireless personal area networks (6LoWPANs). RFC 6775, RFC Editor, November 2012. 32, 72

[60] P Thubert, A Brandt, T Clausen, J Hui, R Kelsey, P Levis, K Pister, R Struik, and J Vasseur. RPL: IPv6 routing protocol for low power and lossy networks. *IETF draft[Online]. http://tools. ietf. org/html/draft-ietf-roll-rpl-19*, 2011. 33, 51, 52

[61] Paolo Baronti, Prashant Pillai, Vince WC Chook, Stefano Chessa, Alberto Gotta, and Y Fun Hu. Wireless sensor networks: A survey on the state of the art and the 802.15. 4 and zigbee standards. *Computer communications*, 30(7):1655–1695, 2007. 37

[62] Tomas Lennvall, Stefan Svensson, and Fredrik Hekland. A comparison of WirelessHART and ZigBee for industrial applications. In *IEEE International Workshop on Factory Communication Systems*, volume 2008, pages 85–88, 2008. 37, 72, 74

[63] Anna N Kim, Fredrik Hekland, Stig Petersen, and Paula Doyle. When HART goes wireless: Understanding and implementing the WirelessHART standard. In *Emerging Technologies and Factory Automation, 2008. ETFA 2008. IEEE International Conference on*, pages 899–907. IEEE, 2008. 39, 54, 66, 74

[64] BLUETOOTH SPECIFICATION Version 4.2. Technical Standard, Bluetooth SIG, December 2014. 41, 59, 72

[65] Carles Gomez, Joaquim Oller, and Josep Paradells. Overview and evaluation of Bluetooth Low Energy: An emerging low-power wireless technology. *Sensors*, 12(9):11734–11753, 2012. 41, 72

[66] Marc Mendonca, Bruno Astuto A Nunes, Xuan-Nam Nguyen, Katia Obraczka, and Thierry Turletti. A survey of software-defined networking: past, present, and future of programmable networks. *hal-00825087*, 2013. 44, 46

[67] M. Bellare, P. Rogaway, and D. Wagner. The EAX mode of operation. In *Fast Software Encryption*. Springer, 2004. 59, 68, 72, 75, 105, 132

[68] M. Conti, J. Willemsen, and B. Crispo. Providing Source Location Privacy in Wireless Sensor Networks: A Survey. *Communications Surveys Tutorials, IEEE*, 15(3):1238–1280, Third 2013. 65

[69] Xi Luo, Xu Ji, and Myong-Soon Park. Location Privacy against Traffic Analysis Attacks in Wireless Sensor Networks. In *Information Science and Applications (ICISA), 2010 International Conference on*, pages 1–6, April 2010. 66

[70] JR Jiang, JP Sheu, C. Tu, and JW Wu. An Anonymous Path Routing (APR) Protocol for Wireless Sensor Networks. *Journal of Information Science and Engineering*, 27(2):657–680, 2011. 66

[71] S. Misra and Guoliang Xue. SAS: A Simple Anonymity Scheme for Clustered Wireless Sensor Networks. In *Communications, 2006. ICC '06. IEEE International Conference on*, volume 8, pages 3414–3419, June 2006. 66

[72] Alireza A. Nezhad, Ali Miri, and Dimitris Makrakis. Location privacy and anonymity preserving routing for wireless sensor networks. *Computer Networks*, 52(18):3433 – 3452, 2008. 66

[73] L.A. Grieco, G. Boggia, S. Sicari, and P. Colombo. Secure wireless multimedia sensor networks: A survey. In *Mobile Ubiquitous Computing, Systems, Services and Technologies, 2009. UBICOMM '09. Third International Conference on*, pages 194–201, Oct 2009. 66

[74] Å. Grain-128a: A New Version of Grain-128 with Optional Authentication. 66, 68, 72, 75, 105, 132

[75] Shaibal Chakrabarty, Monica John, and Daniel W. Engels. Black Routing and Node Obscuring in IoT. In *Internet of Things (WF-IoT), Dec 2016 IEEE World Forum on*. IEEE, 2016. 68, 75, 118

[76] Nuno M Garcia, Mário M Freire, and Paulo P Monteiro. The ethernet frame payload size and its effect on ipv4 and ipv6 traffic. In *Information Networking, 2008. ICOIN 2008. International Conference on*, pages 1–5. IEEE, 2008. 69, 86, 105

[77] Robert Hinden. Internet protocol, version 6 (ipv6) specification. 2017. 70, 103, 105

[78] Atena Shiranzaei and Rafiqul Zaman Khan. Ipv6 security issues—a systematic review. In *Next-Generation Networks*, pages 41–49. Springer, 2018. 70, 105

[79] Sean McCreary and KC Claffy. Trends in wide area ip traffic patterns, 2000. 70, 105

[80] Adrian Perrig, John Stankovic, and David Wagner. Security in wireless sensor networks. *Communications of the ACM*, 47(6):53–57, 2004. 72, 102

[81] Mauro Conti, Roberto Di Pietro, Luigi Vincenzo Mancini, and Alessandro Mei. Emergent properties: detection of the node-capture attack in mobile wireless sensor networks. In *Proceedings of the first ACM conference on Wireless network security*, pages 214–219. ACM, 2008. 72, 102, 143

[82] Maryam Jalalitabar, Marco Valero, and Anu G Bourgeois. Demonstrating the threat of hardware trojans in wireless sensor networks. In *Computer Communication and Networks (ICCCN), 2015 24th International Conference on*, pages 1–8. IEEE, 2015. 72, 102, 115

[83] Venkatachalam Subramanian, Selcuk Uluagac, Hasan Cam, and Raheem Beyah. Examining the characteristics and implications of sensor side channels. In *Communications (ICC), 2013 IEEE International Conference on*, pages 2205–2210. IEEE, 2013. 72, 102, 115

[84] Rabia Riaz, Ki-Hyung Kim, and H Farooq Ahmed. Security analysis survey and framework design for ip connected lowpans. In *Autonomous Decentralized Systems, 2009. ISADS'09. International Symposium on*, pages 1–6. IEEE, 2009. 72, 74

[85] Anass Rghioui, Mohammed Bouhorma, and Abderrahim Benslimane. Analytical study of security aspects in 6lowpan networks. In *Information and Communication Technology for the Muslim World (ICT4M), 2013 5th International Conference on*. IEEE, 2013. 72, 74, 132

[86] Mike Ryan. Bluetooth: with low energy comes low security. In *Presented as part of the 7th USENIX Workshop on Offensive Technologies*, 2013. 72, 74

[87] A Rahman, W Olesinski, and P Gburzynski. Controlled flooding in wireless ad-hoc networks. In *Wireless Ad-Hoc Networks, International Workshop on*, pages 73–78. IEEE, 2004. 80

[88] Rowan Klöti, Vasileios Kotronis, and Paul Smith. Openflow: A security analysis. *Proc. Wkshp on Secure Network Protocols (NPSec). IEEE*, 2013. 92, 106

[89] OpenFlow Switch Consortium et al. Openflow switch specification version 1.0. 0, 2009. 93, 106

[90] Diego Kreutz, Fernando Ramos, and Paulo Verissimo. Towards secure and dependable software-defined networks. In *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*, pages 55–60. ACM, 2013. 93, 106

[91] Sandra Scott-Hayward, Gemma O'Callaghan, and Sakir Sezer. SDN security: A survey. In *Future Networks and Services (SDN4FNS), 2013 IEEE SDN for*, pages 1–7. IEEE, 2013. 93

[92] Ángel Leonardo Valdivieso Caraguay, Alberto Benito Peral, Lorena Isabel Barona López, and Luis Javier García Villalba. SDN: Evolution and Opportunities in the Development IoT Applications. *International Journal of Distributed Sensor Networks*, 2014, 2014. 94

[93] Di Wu, Dmitri I Arkhipov, Eskindir Asmare, Zhijing Qin, and Julie A McCann. UbiFlow: Mobility Management in Urban-scale Software Defined IoT. 94, 95

[94] Zhijing Qin, Grit Denker, Carlo Giannelli, Paolo Bellavista, and Nalini Venkatasubramanian. A Software Defined Networking Architecture for the Internet-of-Things. In *Network Operations and Management Symposium (NOMS), 2014 IEEE*, pages 1–9. IEEE, 2014. 95

[95] David Goldschlag, Michael Reed, and Paul Syverson. Onion routing. *Communications of the ACM*, 42(2):39–41, 1999. 97, 114

[96] Carl Herberger et al. 2016-2017 Global Application Network Security Report. Technical Report, Radware, Jan 2017. 102

[97] Stephen Farrell and Hannes Tschofenig. Pervasive monitoring is an attack. RFC 1654, RFC Editor, May 2014. 102

[98] Mohammad Lotfollahi, Ramin Shirali, Mahdi Jafari Siavoshani, and Mohammdsadegh Saberian. Deep packet: A novel approach for encrypted traffic classification using deep learning. *arXiv preprint arXiv:1709.02656*, 2017. 102

[99] Elizabeth Atkins. Spying on americans: At what point does the nsa's collection and searching of metadata violate the fourth amendment. *Wash. JL Tech. & Arts*, 10:51, 2014. 102

[100] Noah Apthorpe, Dillon Reisman, Srikanth Sundaresan, Arvind Narayanan, and Nick Feamster. Spying on the smart home: Privacy attacks and defenses on encrypted iot traffic. *arXiv preprint arXiv:1708.05044*, 2017. 102

[101] Shaibal Chakrabarty and Daniel W Engels. A secure iot architecture for smart cities. In *2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2016. 102, 140, 146

[102] Gabriel Montenegro, N Kushalnagar, J Hui, and D Culler. Rfc 4944. *Transmission of IPv6 packets over IEEE*, 802(4), 2007. 102

[103] Christine Hennebert and Jessye Dos Santos. Security protocols and privacy issues into 6lowpan stack: A synthesis. *IEEE Internet of Things Journal*, 1(5):384–398, 2014. 102, 115, 138, 141

[104] Shahid Raza, Simon Duquennoy, Tony Chung, Dogan Yazar, Thiemo Voigt, and Utz Roedig. Securing communication in 6lowpan with compressed ipsec. In *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on*, pages 1–8. IEEE, 2011. 102, 115

[105] Ghada Glissa and Aref Meddeb. 6lowpsec: An end-to-end security protocol for 6lowpan. *Ad Hoc Networks*, 82:100–112, 2019. 102, 115

[106] Eldad Perahia. Ieee 802.11 n development: History, process, and technology. *IEEE Communications Magazine*, 46(7), 2008. 103

[107] Sheila Frankel and Suresh Krishnan. IP security (ipsec) and Internet Key Exchange (IKE) Document Roadmap. RFC 6071, RFC Editor, Feb 2011. 103

[108] Carlos E Caicedo, James BD Joshi, and Summit R Tuladhar. Ipv6 security challenges. *Computer*, 42(2):36–42, 2009. 103

[109] Johanna Ullrich, Katharina Krombholz, Heidelinde Hobel, Adrian Dabrowski, and Edgar R Weippl. IPv6 Security:Attacks and Countermeasures in a Nutshell. 2014. 103

[110] Dennis Felsch, Martin Grothe, et al. The dangers of key reuse: practical attacks on ipsec ike. In *Proceedings of the 27th USENIX Conference on Security Symposium*, pages 567–583. USENIX Association, 2018. 103

[111] Masoud Akhoondi, Curtis Yu, and Harsha V Madhyastha. Lastor: A low-latency as-aware tor client. In *2012 IEEE Symposium on Security and Privacy*, pages 476–490. IEEE, 2012. 115

[112] Yixin Sun, Anne Edmundson, Laurent Vanbever, Oscar Li, Jennifer Rexford, Mung Chiang, and Prateek Mittal. Raptor: Routing attacks on privacy in tor. In *USENIX Security Symposium*, pages 271–286, 2015. 115

[113] Sambuddho Chakravarty, Marco V Barbera, Georgios Portokalidis, Michalis Polychronakis, and Angelos D Keromytis. On the effectiveness of traffic analysis against anonymity networks using flow records. In *International conference on passive and active network measurement*, pages 247–257. Springer, 2014. 115

[114] Kevin Bauer, Damon McCoy, Dirk Grunwald, Tadayoshi Kohno, and Douglas Sicker. Low-resource routing attacks against tor. In *Proceedings of the 2007 ACM workshop on Privacy in electronic society*, pages 11–20. ACM, 2007. 115

[115] Vasilis Pappas, Elias Athanasopoulos, Sotiris Ioannidis, and Evangelos P Markatos. Compromising anonymity using packet spinning. In *International Conference on Information Security*, pages 161–174. Springer, 2008. 115

[116] Shahid Raza, Thiemo Voigt, and Vilhelm Jutvik. Lightweight ikev2: a key management solution for both the compressed ipsec and the ieee 802.15. 4 security. In *Proceedings of the IETF workshop on smart object security*, volume 23. Citeseer, 2012. 115

[117] B Bowerman, J Braverman, J Taylor, H Todosow, and U Von Wimmersperg. The vision of a smart city. In *2nd International Life Extension Technology Workshop, Paris*, volume 28, 2000. 126

[118] Andrea Zanella, Nicola Bui, Angelo Castellani, Lorenzo Vangelista, and Michele Zorzi. Internet of Things for Smart Cities. *IEEE Internet of Things Journal*, 1(1):22–32, 2014. 127, 137

[119] Henrich C Pöhls, Vangelis Angelakis, Santiago Suppan, Kai Fischer, George Oikonomou, Elias Z Tragos, Rodrigo Diaz Rodriguez, and Theodoros Mouroutis. Rerum: Building a reliable iot upon privacy-and security-enabled smart objects. In *Wireless Communications and Networking Conference Workshops (WCNCW), 2014 IEEE*, pages 122–127. IEEE, 2014. 129, 142

[120] F Silva Ferraz and CA Guimaraes Ferraz. Smart city security issues: Depicting information security issues in the role of an urban environment. In *Utility and Cloud Computing (UCC), 2014 IEEE/ACM 7th International Conference on*, pages 842–847. IEEE, 2014. 130

[121] Hui Suo, Jiafu Wan, Caifeng Zou, and Jianqi Liu. Security in the internet of things: a review. In *Computer Science and Electronics Engineering (ICCSEE), 2012 international conference on*, volume 3, pages 648–651. IEEE, 2012. 130

[122] Elias Z Tragos, Vangelis Angelakis, Alexandras Fragkiadakis, David Gundlegard, Cosmin-Septimiu Nechifor, George Oikonomou, Henrich C Pohls, and Anastasius Gavras. Enabling reliable and secure iot-based smart city applications. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2014 IEEE International Conference on*, pages 111–116. IEEE, 2014. 132

[123] Kehua Su, Jie Li, and Hongbo Fu. Smart city and the applications. In *Electronics, Communications and Control (ICECC), 2011 International Conference on*, pages 1028–1031. IEEE, 2011. 136

[124] Edward Curry, Schahram Dustdar, Quan Z Sheng, and Amit Sheth. Smart cities–enabling services and applications, 2016. 137

[125] David Perez Abreu, Karima Velasquez, Marilia Curado, and Edmundo Monteiro. A resilient internet of things architecture for smart cities. *Annals of Telecommunications*, 72(1-2):19–30, 2017. 137

[126] Kamanashis Biswas and Vallipuram Muthukkumarasamy. Securing smart cities using blockchain technology. In *High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2016 IEEE 18th International Conference on*, pages 1392–1393. IEEE, 2016. 137

[127] Vito Albino, Umberto Berardi, and Rosa Maria Dangelico. Smart cities: Definitions, dimensions, performance, and initiatives. *Journal of Urban Technology*, 22(1):3–21, 2015. 137

[128] Sabrina Sicari, Alessandra Rizzardi, Luigi Alfredo Grieco, and Alberto Coen-Porisini. Security, privacy and trust in internet of things: The road ahead. *Computer networks*, 76:146–164, 2015. 138

[129] Jesus Pacheco and Salim Hariri. Iot security framework for smart cyber infrastructures. In *Foundations and Applications of Self* Systems, IEEE International Workshops on*, pages 242–247. IEEE, 2016. 141

[130] Matt Knight and Marc Newlin. Radio exploitation 101. 141

[131] A Bartoli, J Hernández-Serrano, M Soriano, M Dohler, A Kountouris, and D Barthel. Security and privacy in your smart city. In *Proceedings of the Barcelona smart cities congress*, volume 292, 2011. 141

[132] Ramaswamy Chandramouli, Michaela Iorga, and Santosh Chokhani. Cryptographic key management issues and challenges in cloud services. In *Secure Cloud Computing*, pages 1–30. Springer, 2014. 142

[133] Zaheer Khan, Zeeshan Pervez, and Abdul Ghafoor. Towards cloud based smart cities data security and privacy management. In *Utility and cloud computing (UCC), 2014 IEEE/ACM 7th international conference on*, pages 806–811. IEEE, 2014. 142

[134] Laurent Eschenauer and Virgil D Gligor. A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pages 41–47. ACM, 2002. 142, 145

[135] Osman Yagan. Performance of the eschenauer–gligor key distribution scheme under an on/off channel. *IEEE Transactions on Information Theory*, 58(6):3821–3835, 2012. 142

[136] Rodrigo Roman, Cristina Alcaraz, Javier Lopez, and Nicolas Sklavos. Key management systems for sensor networks in the context of the internet of things. *Computers & Electrical Engineering*, 37(2):147–159, 2011. 142

[137] Asma Elmangoush, Hakan Coskun, Sebastian Wahle, and Thomas Magedanz. Design aspects for a reference m2m communication platform for smart cities. In *Innovations in Information Technology (IIT), 2013 9th International Conference on*, pages 204–209. IEEE, 2013. 142

[138] Arvind Narayanan and Jeremy Clark. Bitcoin's academic pedigree. *Communications of the ACM*, 60(12):36–45, 2017. 144

[139] S Voshmgir and V Kalinov. Blockchain: A beginners guide. blockchainhub, 2017. 145

[140] Daniel Burkhardt, Maximilian Werling, and Heiner Lasi. Distributed ledger. In *2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, pages 1–9. IEEE, 2018. 145

[141] S Popov. The tangle, iota whitepaper, 2018. 145

[142] Marco Conoscenti, Antonio Vetro, and Juan Carlos De Martin. Blockchain for the internet of things: A systematic literature review. In *Computer Systems and Applications (AICCSA), 2016 IEEE/ACS 13th International Conference of*, pages 1–6. IEEE, 2016. 145

[143] Nir Kshetri. Can blockchain strengthen the internet of things? *IT Professional*, 19(4):68–72, 2017. 147

[144] Allison Nixon, John Costello, and Z Wilkholm. An after-action analysis of the mirai botnet attacks on dyn, 2016. 149

[145] Adel S Elmaghraby and Michael M Losavio. Cyber security challenges in smart cities: Safety, security and privacy. *Journal of advanced research*, 5(4):491–497, 2014.

[146] Christopher Gaffney and Cerianne Robertson. Smarter than smart: Rio de janeiro's flawed emergence as a smart city. *Journal of Urban Technology*, 25(3):47–64, 2018.

[147] Mohamed Tahar Hammi, Badis Hammi, Patrick Bellot, and Ahmed Serhrouchni. Bubbles of trust: a decentralized blockchain-based authentication system for iot. *Computers & Security*, 78:126–142, 2018.

[148] Roben Castagna Lunardi, Regio Antonio Michelin, Charles Varlei Neu, and Avelino Francisco Zorzo. Distributed access control on iot ledger-based architecture. In *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*, pages 1–7. IEEE, 2018.

[149] Steve Deering and Robert Hinden. Internet protocol, version 6 (ipv6) specification. Technical report, 2017.

[150] P Atik and W Rick. A complete guide on ipv6 attack and defense, 2012.

[151] Abdur Rahim Choudhary and Alan Sekelsky. Securing ipv6 network infrastructure: A new security model. In *Technologies for Homeland Security (HST), 2010 IEEE International Conference on*, pages 500–506. IEEE, 2010.

[152] Antonio Marcos Alberti and Dhananjay Singh. Internet of things: Perspectives, challenges and opportunities. In *International Workshop on Telecommunications (IWT 2013)*, 2013.

[153] ZigBee Specification. Zigbee alliance. *URL: http://www.zigbee.org*, 558, 2006.

[154] James A Jerkins. Motivating a market or regulatory solution to iot insecurity with the mirai botnet code. In *Computing and Communication Workshop and Conference (CCWC), 2017 IEEE 7th Annual*, pages 1–5. IEEE, 2017.

[155] Alf Helge Omre and Steven Keeping. Bluetooth low energy: wireless connectivity for medical monitoring. *Journal of diabetes science and technology*, 4(2):457–463, 2010.

[156] Ovidiu Vermesan and Peter Friess. *Internet of things: converging technologies for smart environments and integrated ecosystems*. River Publishers, 2013.

[157] Nathalie Omnes, Marc Bouillon, Gael Fromentoux, and Olivier Le Grand. A programmable and virtualized network & it infrastructure for the internet of things: How can nfv & sdn help for facing the upcoming challenges. In *Intelligence in Next Generation Networks (ICIN), 2015 18th International Conference on*, pages 64–69. IEEE, 2015.

[158] Kshira Sagar Sahoo, Bibhudatta Sahoo, and Abinas Panda. A secured sdn framework for iot. 2015.

[159] CP Vandana. Security improvement in iot based on software defined networking (sdn).

[160] Akram Hakiri, Pascal Berthou, Aniruddha Gokhale, and Slim Abdellatif. "publish/subscribe-enabled software defined networking for efficient and scalable iot communications". *Communications Magazine, IEEE*, 53(9):48–54, 2015.

[161] A.A.; Baras J.S.; Moustakides G.V. Radosavac, S.; Cardenas. Detecting IEEE 802.11 MAC layer misbehavior in ad hoc networks: Robust strategies against individual and colluding attackers. () [On the electrodynamics of moving bodies]. *Journal of Computer Security*, (15):103–128, 2007.

[162] Shahnaz Saleem, Sana Ullah, and Kyung Sup Kwak. A study of IEEE 802.15. 4 security framework for wireless body area networks. *Sensors*, 11(2):1383–1395, 2011.

[163] Jon T. Adams. An Introduction to IEEE STD 802.15.4. 2005.

[164] Donald E. Knuth. *Fundamental Algorithms*, chapter 1.2. Addison-Wesley, 1973.

[165] Yakov Rekhter and Tony Li. A Border Gateway Protocol 4 (BGP-4). RFC 1654, RFC Editor, July 1995.

[166] Linus Wallgren, Shahid Raza, and Thiemo Voigt. Routing attacks and countermeasures in the rpl-based internet of things. *International Journal of Distributed Sensor Networks*, 2013, 2013.

[167] Alireza A Nezhad, Ali Miri, and Dimitris Makrakis. Location privacy and anonymity preserving routing for wireless sensor networks. *Computer Networks*, 52(18):3433–3452, 2008.

[168] Stephan Olariu, Mohamed Eltoweissy, and Mohamed Younis. Answer: autonomous wireless sensor network. In *Proceedings of the 1st ACM international workshop on Quality of service & security in wireless and mobile networks*, pages 88–95. ACM, 2005.

[169] Jalal Al-Muhtadi, Roy Campbell, Apu Kapadia, M Dennis Mickunas, and Seung Yi. Routing through the mist: privacy preserving communication in ubiquitous computing environments. In *Distributed Computing Systems, 2002. Proceedings. 22nd International Conference on*, pages 74–83. IEEE, 2002.

[170] Fahmida Aseez and Sheena Mathew. Hierarchical partition-based anonymous routing protocol (hpar) in manet for efficient and secure transmission. *arXiv preprint arXiv:1605.02860*, 2016.

[171] Karim El Defrawy and Gene Tsudik. Alarm: anonymous location-aided routing in suspicious manets. *IEEE Transactions on Mobile Computing*, 10(9):1345–1358, 2011.

[172] Jiejun Kong and Xiaoyan Hong. Anodr: anonymous on demand routing with untraceable routes for mobile ad-hoc networks. In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 291–302. ACM, 2003.

[173] Panayiotis Kotzanikolaou, George Chatzisofroniou, and Mike Burmester. Broadcast anonymous routing (bar): scalable real-time anonymous communication. *International Journal of Information Security*, pages 1–14, 2016.

[174] Li Zhuang, Feng Zhou, Ben Y Zhao, and Antony Rowstron. Cashmere: Resilient anonymous routing. In *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2*, pages 301–314. USENIX Association, 2005.

[175] Apala Ray, Johan Akerberg, Mikael Gidlund, and M Bjorkman. Initial key distribution for industrial wireless sensor networks. In *Industrial Technology (ICIT), 2013 IEEE International Conference on*, pages 1309–1314. IEEE, 2013.

[176] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer networks*, 52(12):2292–2330, 2008.

[177] Amjad Soomro and Dave Cavalcanti. Opportunities and challenges in using wpan and wlan technologies in medical environments [accepted from open call]. *Communications Magazine, IEEE*, 45(2):114–122, 2007.

[178] Steven D Baker and David H Hoglund. Medical-grade, mission-critical wireless networks [designing an enterprise mobility solution in the healthcare environment]. *Engineering in Medicine and Biology Magazine, IEEE*, 27(2):86–95, 2008.

[179] Qiwei Zhang, Andre BJ Kokkeler, and Gerard JM Smit. Cognitive radio for emergency networks. 2006.

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

[181] G Piro, G Boggia, and LA Grieco. A standard compliant security framework for ieee 802.15. 4 networks. In *Internet of Things (WF-IoT), 2014 IEEE World Forum on*, pages 27–30. IEEE, 2014.

[182] Paul Syverson, R Dingledine, and N Mathewson. Tor: the second-generation onion router. In *Usenix Security*, 2004.

[183] Damon McCoy, Kevin Bauer, Dirk Grunwald, Tadayoshi Kohno, and Douglas Sicker. Shining light in dark places: Understanding the tor network. In *International Symposium on Privacy Enhancing Technologies Symposium*, pages 63–76. Springer, 2008.

[184] Kevin Bauer, Damon McCoy, Dirk Grunwald, Tadayoshi Kohno, and Douglas Sicker. Low-resource routing attacks against tor. In *Proceedings of the 2007 ACM workshop on Privacy in electronic society*, pages 11–20. ACM, 2007.

[185] Jing Deng, Richard Han, and Shivakant Mishra. Countermeasures against traffic analysis attacks in wireless sensor networks. In *First International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM'05)*, pages 113–126. IEEE, 2005.

[186] Joan Feigenbaum, Aaron Johnson, and Paul Syverson. Probabilistic analysis of onion routing in a black-box model. *ACM Transactions on Information and System Security (TISSEC)*, 15(3):14, 2012.

[187] Steven J Murdoch and George Danezis. Low-cost traffic analysis of tor. In *2005 IEEE Symposium on Security and Privacy (S&P'05)*, pages 183–195. IEEE, 2005.

[188] David Goldschlag, Michael Reed, and Paul Syverson. Onion routing. *Communications of the ACM*, 42(2):39–41, 1999.

[189] Michael G Reed, Paul F Syverson, and David M Goldschlag. Anonymous connections and onion routing. *IEEE Journal on Selected areas in Communications*, 16(4):482–494, 1998.

[190] Paul Syverson, Gene Tsudik, Michael Reed, and Carl Landwehr. Towards an analysis of onion routing security. In *Designing Privacy Enhancing Technologies*, pages 96–114. Springer, 2001.

[191] David M Goldschlag, Michael G Reed, and Paul F Syverson. Hiding routing information. In *International Workshop on Information Hiding*, pages 137–150. Springer, 1996.

[192] David L Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–90, 1981.

[193] Security Architecture for Open Systems Interconnection for CCITT Applications. Recommendation X.800, ITU-T, 1991.

[194] Sakir Sezer, Sandra Scott-Hayward, Pushpinder-Kaur Chouhan, Barbara Fraser, David Lake, Jim Finnegan, Niel Viljoen, Marc Miller, and Navneet Rao. Are we ready for sdn? implementation challenges for software-defined networks. *Communications Magazine, IEEE*, 51(7):36–43, 2013.

[195] Seungwon Shin and Guofei Gu. Attacking software-defined networks: A first feasibility study. In *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*, pages 165–166. ACM, 2013.

[196] Aashima Gagandeep and Pawan Kumar. Analysis of different security attacks in manets on protocol stack a-review. *International Journal of Engineering and Advanced Technology (IJEAT)*, 1(5):269–75, 2012.

[197] Pradip M Jawandhiya, Mangesh M Ghonge, MS Ali, and JS Deshpande. A survey of mobile ad hoc network attacks. *International Journal of Engineering Science and Technology*, 2(9):4063–4071, 2010.

[198] Jean-François Raymond. Traffic analysis: Protocols, attacks, design issues, and open problems. In *Designing Privacy Enhancing Technologies*, pages 10–29. Springer, 2001.

[199] D Karuppaiah. Traffic analysis using multicast routing for mobile ad hoc network. *Traffic*, 5(3), 2016.

[200] R Radha and R Subhashini. A novel method for traffic analysis in manets. 2015.

[201] Zheng Wang and Jon Crowcroft. Analysis of shortest-path routing algorithms in a dynamic network environment. *ACM SIGCOMM Computer Communication Review*, 22(2):63–71, 1992.

[202] Justin A Boyan and Michael L Littman. Packet routing in dynamically changing networks: A reinforcement learning approach. *Advances in neural information processing systems*, pages 671–671, 1994.

[203] Pandurang Kamat, Yanyong Zhang, Wade Trappe, and Celal Ozturk. Enhancing source-location privacy in sensor network routing. In *25th IEEE International Conference on Distributed Computing Systems (ICDCS'05)*, pages 599–608. IEEE, 2005.

[204] Shaibal Chakrabarty, Daniel W Engels, and Selina Thathapudi. Black sdn for the internet of things. In *Mobile Ad Hoc and Sensor Systems (MASS), 2015 IEEE 12th International Conference on*, pages 190–198. IEEE, 2015.

[205] Ed Callaway, Paul Gorday, Lance Hester, Jose A Gutierrez, Marco Naeve, Bob Heile, and Venkat Bahl. Home networking with ieee 802. 15. 4: a developing standard for low-rate wireless personal area networks. *IEEE Communications magazine*, 40(8):70–77, 2002.

[206] Martin Agren, Martin Hell, Thomas Johansson, and Willi Meier. Grain-128a: a new version of grain-128 with optional authentication. *International Journal of Wireless and Mobile Computing*, 5(1):48–59, 2011.

[207] Bruno Astuto A Nunes, Marc Mendonca, Xuan-Nam Nguyen, Katia Obraczka, and Thierry Turletti. A survey of software-defined networking: Past, present, and future of programmable networks. *IEEE Communications Surveys & Tutorials*, 16(3):1617–1634, 2014.

[208] Hyojoon Kim and Nick Feamster. Improving network management with software defined networking. *IEEE Communications Magazine*, 51(2):114–119, 2013.

[209] Stefaan Seys and Bart Preneel. Arm: Anonymous routing protocol for mobile ad hoc networks. *International Journal of Wireless and Mobile Computing*, 3(3):145–155, 2009.

[210] Charles E Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (dsdv) for mobile computers. In *ACM SIGCOMM computer communication review*, volume 24, pages 234–244. ACM, 1994.

[211] Guoyou He. Destination-sequenced distance vector (dsdv) protocol. *Networking Laboratory, Helsinki University of Technology*, pages 1–9, 2002.

[212] Bo Zhu, Zhiguo Wan, Mohan S Kankanhalli, Feng Bao, and Robert H Deng. Anonymous secure routing in mobile ad-hoc networks. In *Local Computer Networks, 2004. 29th Annual IEEE International Conference on*, pages 102–108. IEEE, 2004.

[213] Zygmunt J Haas. A new routing protocol for the reconfigurable wireless networks. In *Universal Personal Communications Record, 1997. Conference Record., 1997 IEEE 6th International Conference on*, volume 2, pages 562–566. IEEE, 1997.

[214] Charles Perkins, Elizabeth Belding-Royer, and Samir Das. Ad hoc on-demand distance vector (aodv) routing. Technical report, 2003.

[215] Ian D Chakeres and Elizabeth M Belding-Royer. Aodv routing protocol implementation design. In *Distributed Computing Systems Workshops, 2004. Proceedings. 24th International Conference on*, pages 698–703. IEEE, 2004.

[216] Sanjay Raghunath Deshpande. Arbitration protocol for peer-to-peer communication in synchronous systems, August 13 2002. US Patent 6,434,638.

[217] Liu Yang, Markus Jakobsson, and Susanne Wetzel. Discount anonymous on demand routing for mobile ad hoc networks. In *Securecomm and Workshops, 2006*, pages 1–10. IEEE, 2006.

[218] Anuj K Gupta, Harsh Sadawarti, and Anil K Verma. Performance analysis of aodv, dsr & tora routing protocols. *International Journal of Engineering and Technology*, 2(2):226, 2010.

[219] Mehran Abolhasan, Tadeusz Wysocki, and Eryk Dutkiewicz. A review of routing protocols for mobile ad hoc networks. *Ad hoc networks*, 2(1):1–22, 2004.

[220] Azzedine Boukerche. Performance evaluation of routing protocols for ad hoc wireless networks. *Mobile Networks and Applications*, 9(4):333–342, 2004.

[221] Xiaoxin Wu and Bharat Bhargava. Ao2p: Ad hoc on-demand position-based private routing protocol. *IEEE Transactions on Mobile Computing*, 4(4):335–348, 2005.

[222] Shima Mohseni, Rosilah Hassan, Ahmed Patel, and Rozilawati Razali. Comparative review study of reactive and proactive routing protocols in manets. In *4th IEEE International Conference on Digital Ecosystems and Technologies*, pages 304–309. IEEE, 2010.

[223] Chao-Chin Chou, C-c Jay Kuo, Kshirasagar Naik, et al. An efficient anonymous communication protocol for peer-to-peer applications over mobile ad-hoc networks. *IEEE Journal on selected areas in communications*, 25(1):192–203, 2007.

[224] Adam M Gustafson, Edwards Allen, Scott Givan, Daniel Smith, James C Carrington, and Kristin D Kasschau. Asrp: the arabidopsis small rna project database. *Nucleic Acids Research*, 33(suppl 1):D637–D640, 2005.

[225] Kazuya Sakai, Min-Te Sun, Wei-Shinn Ku, Jie Wu, and Faisal S Alanazi. An analysis of onion-based anonymous routing for delay tolerant networks. In *Proc. of the 36th IEEE International Conference on Distributed Computing Systems (ICDCS 2016)*, 2016.

[226] Michael G Reed, Paul F Syverson, and David M Goldschlag. Anonymous connections and onion routing. *IEEE Journal on Selected areas in Communications*, 16(4):482–494, 1998.

[227] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. Technical report, DTIC Document, 2004.

[228] Michael G Reed, Paul F Syverson, and David M Goldschlag. Proxies for anonymous routing. In *Computer Security Applications Conference, 1996., 12th Annual*, pages 95–104. IEEE, 1996.

[229] Karim El Defrawy and Gene Tsudik. Prism: Privacy-friendly routing in suspicious manets (and vanets). In *Network Protocols, 2008. ICNP 2008. IEEE International Conference on*, pages 258–267. IEEE, 2008.

[230] Steven J Murdoch and George Danezis. Low-cost traffic analysis of tor. In *2005 IEEE Symposium on Security and Privacy (S&P'05)*, pages 183–195. IEEE, 2005.

[231] Robert W Floyd. Algorithm 97: shortest path. *Communications of the ACM*, 5(6):345, 1962.

[232] Donald B Johnson. A note on dijkstra's shortest path algorithm. *Journal of the ACM (JACM)*, 20(3):385–388, 1973.

[233] S Skiena. Dijkstra's algorithm. *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica, Reading, MA: Addison-Wesley*, pages 225–227, 1990.

[234] Jie Wu and Hailan Li. On calculating connected dominating set for efficient routing in ad hoc wireless networks. In *Proceedings of the 3rd international workshop on Discrete algorithms and methods for mobile computing and communications*, pages 7–14. ACM, 1999.

[235] C-K Toh. Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks. *IEEE communications Magazine*, 39(6):138–147, 2001.

[236] David B Johnson and David A Maltz. Dynamic source routing in ad hoc wireless networks. In *Mobile computing*, pages 153–181. Springer, 1996.

[237] Richard Bellman. On a routing problem. *Quarterly of applied mathematics*, pages 87–90, 1958.

[238] Maxim Likhachev, Geoffrey J Gordon, and Sebastian Thrun. Ara*: Anytime a* with provable bounds on sub-optimality. In *Advances in Neural Information Processing Systems*, page None, 2003.

[239] Charles E Leiserson and Tao B Schardl. A work-efficient parallel breadth-first search algorithm (or how to cope with the nondeterminism of reducers). In *Proceedings of the twenty-second annual ACM symposium on Parallelism in algorithms and architectures*, pages 303–314. ACM, 2010.

[240] Imad Aad, Claude Castelluccia, and Jean-Pierre Huubaux. Packet coding for strong anonymity in ad hoc networks. In *Securecomm and Workshops, 2006*, pages 1–10. IEEE, 2006.

[241] Zhou Zhi and Yow Kin Choong. Anonymizing geographic ad hoc routing for preserving location privacy. In *25th IEEE International Conference on Distributed Computing Systems Workshops*, pages 646–651. IEEE, 2005.

[242] Fraser Cadger, Kevin Curran, Jose Santos, and Sandra Moffett. A survey of geographical routing in wireless ad-hoc networks. *IEEE Communications Surveys & Tutorials*, 15(2):621–653, 2013.

[243] M Rahman, Masahiro Mambo, Atsuo Inomata, and Eiji Okamoto. An anonymous on-demand position-based routing in mobile ad hoc networks. In *International Symposium on Applications and the Internet (SAINT'06)*, pages 7–pp. IEEE, 2006.

[244] Manel Guerrero Zapata. Secure ad hoc on-demand distance vector routing. *ACM SIGMOBILE Mobile Computing and Communications Review*, 6(3):106–107, 2002.

[245] Yih-Chun Hu, Adrian Perrig, and David B Johnson. Ariadne: A secure on-demand routing protocol for ad hoc networks. *Wireless networks*, 11(1-2):21–38, 2005.

[246] Yih-Chun Hu, David B Johnson, and Adrian Perrig. Sead: Secure efficient distance vector routing for mobile wireless ad hoc networks. *Ad hoc networks*, 1(1):175–192, 2003.

[247] Kimaya Sanzgiri, Bridget Dahill, Brian Neil Levine, Clay Shields, and Elizabeth M Belding-Royer. A secure routing protocol for ad hoc networks. In *Network Protocols, 2002. Proceedings. 10th IEEE International Conference on*, pages 78–87. IEEE, 2002.

[248] Sencun Zhu, Sanjeev Setia, Sushil Jajodia, and Peng Ning. An interleaved hop-by-hop authentication scheme for filtering of injected false data in sensor networks. In *Security and privacy, 2004. Proceedings. 2004 IEEE symposium on*, pages 259–271. IEEE, 2004.

[249] Yi Yang, Xinran Wang, Sencun Zhu, and Guohong Cao. Sdap: A secure hop-by-hop data aggregation protocol for sensor networks. *ACM Transactions on Information and System Security (TISSEC)*, 11(4):18, 2008.

[250] Haowen Chan and Adrian Perrig. Security and privacy in sensor networks. *Computer*, 36(10):103–105, 2003.

[251] Wenrui Zhao and Mostafa H Ammar. Message ferrying: Proactive routing in highly-partitioned wireless ad hoc networks. In *Distributed Computing Systems, 2003. FTDCS 2003. Proceedings. The Ninth IEEE Workshop on Future Trends of*, pages 308–314. IEEE, 2003.

[252] Consolee Mbarushimana and Alireza Shahrabi. Comparative study of reactive and proactive routing protocols performance in mobile ad hoc networks. In *Advanced Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on*, volume 2, pages 679–684. IEEE, 2007.

[253] Shima Mohseni, Rosilah Hassan, Ahmed Patel, and Rozilawati Razali. Comparative review study of reactive and proactive routing protocols in manets. In *4th IEEE International Conference on Digital Ecosystems and Technologies*, pages 304–309. IEEE, 2010.

[254] Charles E Perkins, Elizabeth M Royer, Samir R Das, and Mahesh K Marina. Performance comparison of two on-demand routing protocols for ad hoc networks. *IEEE Personal communications*, 8(1):16–28, 2001.

[255] Baruch Awerbuch, David Holmer, Cristina Nita-Rotaru, and Herbert Rubens. An on-demand secure routing protocol resilient to byzantine failures. In *Proceedings of the 1st ACM workshop on Wireless security*, pages 21–30. ACM, 2002.

[256] Lidong Zhou and Zygmunt J Haas. Securing ad hoc networks. *IEEE network*, 13(6):24–30, 1999.

[257] Shahid Raza, Adriaan Slabbert, Thiemo Voigt, and Krister Landernäs. Security considerations for the wirelesshart protocol. In *2009 IEEE Conference on Emerging Technologies & Factory Automation*, pages 1–8. IEEE, 2009.

[258] Adrian Perrig, John Stankovic, and David Wagner. Security in wireless sensor networks. *Communications of the ACM*, 47(6):53–57, 2004.

[259] Mauro Conti, Roberto Di Pietro, Luigi Vincenzo Mancini, and Alessandro Mei. Emergent properties: detection of the node-capture attack in mobile wireless sensor networks. In *Proceedings of the first ACM conference on Wireless network security*, pages 214–219. ACM, 2008.

[260] Maryam Jalalitabar, Marco Valero, and Anu G Bourgeois. Demonstrating the threat of hardware trojans in wireless sensor networks. In *2015 24th International Conference on Computer Communication and Networks (ICCCN)*, pages 1–8. IEEE, 2015.

[261] Mihir Bellare, Phillip Rogaway, and David Wagner. The eax mode of operation. In *International Workshop on Fast Software Encryption*, pages 389–407. Springer, 2004.

[262] Tie Luo, Hwee-Pink Tan, and Tony QS Quek. Sensor openflow: Enabling software-defined wireless sensor networks. *IEEE Communications Letters*, 16(11):1896–1899, 2012.

[263] Salvatore Costanzo, Laura Galluccio, Giacomo Morabito, and Sergio Palazzo. Software defined wireless networks: Unbridling sdns. In *2012 European Workshop on Software Defined Networking*, pages 1–6. IEEE, 2012.

[264] Nathalie Omnes, Marc Bouillon, Gael Fromentoux, and Olivier Le Grand. A programmable and virtualized network & it infrastructure for the internet of things: How can nfv & sdn help for facing the upcoming challenges. In *Intelligence in Next Generation Networks (ICIN), 2015 18th International Conference on*, pages 64–69. IEEE, 2015.

[265] Lain-Jinn Hwang, Shiann-Tsong Sheu, Yun-Yen Shih, and Yen-Chieh Cheng. Grouping strategy for solving hidden node problem in ieee 802.15. 4 lr-wpan. In *First International Conference on Wireless Internet (WICON'05)*, pages 26–32. IEEE, 2005.

[266] Scott Corson and Joseph Macker. Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation considerations. Technical report, 1998.

[267] Joe Touch, Yu-Shun Wang, and Lars Eggert. Use of ipsec transport mode for dynamic routing. *ISI*, 2004.

[268] Stephen Kent and Randall Atkinson. Rfc 2401: Security architecture for the internet protocol, november 1998. *Obsoletes RFC1825 [Atk95a]. Status: PROPOSED STANDARD.*

[269] Stephen Kent and Randall Atkinson. Rfc 2402: Ip authentication header. 1998.

[270] Kent, Stephen and Atkinson, R. RFC 2406,. *Encapsulating Security Protocol*, 1998.

[271] S Kent and IP Authentication Header. Rfc 4302. *IETF, December*, 2005.

[272] Stephen Kent. Ip encapsulating security payload (esp). 2005.

[273] Stephen Kent and Karen Seo. Security architecture for the internet protocol. Technical report, 2005.

[274] Dan Harkins and Dave Carrel. The internet key exchange (ike). Technical report, 1998.

[275] EC Kaufman. Rfc 4306: Internet key exchange (ikev2) protocol, 2005.

[276] S Frankel and S Krishnan. Ip security (ipsec) and internet key exchange (ike) document roadmap. Technical report, 2011.

[277] Azzedine Boukerche, Khalil El-Khatib, Li Xu, and Larry Korba. Sdar: a secure distributed anonymous routing protocol for wireless and mobile ad hoc networks. In *Local Computer Networks, 2004. 29th Annual IEEE International Conference on*, pages 618–624. IEEE, 2004.

[278] Charles Perkins, Elizabeth Belding-Royer, and Samir Das. Ad hoc on-demand distance vector (aodv) routing. Technical report, 2003.

[279] Shree Murthy and Jose Joaquin Garcia-Luna-Aceves. An efficient routing protocol for wireless networks. *Mobile Networks and applications*, 1(2):183–197, 1996.

[280] Daniel A Schult and P Swart. Exploring network structure, dynamics, and function using networkx. In *Proceedings of the 7th Python in Science Conferences (SciPy 2008)*, volume 2008, pages 11–16, 2008.

[281] Morris Dworkin. Recommendation for block cipher modes of operation. methods and techniques. Technical report.

[282] NIST FIPS Pub. 197: Advanced encryption standard (aes). Technical report, 2001.

[283] Jorge Arturo Pardiñas-Mir and Bruno Salgues. Design considerations for a wireless sensor network architecture attached to a cognitive training system for the elder. *[Online].http:// projet-village.fr/WiCom2013.pdf*, 2013.

[284] Donald Knuth. Knuth: Computers and typesetting.

[285] Chris Karlof and David Wagner. Secure routing in wireless sensor networks: Attacks and countermeasures. In *Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on*, pages 113–127. IEEE, 2003.

[286] Andrew J Wixted, Peter Kinnaird, Hadi Larijani, Alan Tait, Ali Ahmadinia, and Niall Strachan. Evaluation of lora and lorawan for wireless sensor networks. In *SENSORS, 2016 IEEE*, pages 1–3. IEEE, 2016.

[287] N Ahmed, H Rahman, and Md I Hussain. A comparison of 802.11ah and 802.15.4 for iot. *ICT Express*, 2(3):100–102, 2016.

[288] Rapeepat Ratasuk, Benny Vejlgaard, Nitin Mangalvedhe, and Amitava Ghosh. Nb-iot system for m2m communication. In *Wireless Communications and Networking Conference (WCNC), 2016 IEEE*, pages 1–5. IEEE, 2016.

[289] Kais Mekki, Eddy Bajic, Frederic Chaxel, and Fernand Meyer. A comparative study of lpwan technologies for large-scale iot deployment. *ICT Express*, 2018.

[290] Harith Dawood. Ipv6 security vulnerabilities. *International Journal of Information Security Science*, 1(4):100–105, 2012.

[291] Rodney Thayer, Naganand Doraswamy, and Rob Glenn. Ip security document roadmap. Technical report, 1998.