

A Clean Slate Approach to Secure Wireless Networking

Jonathan Ponniah

Texas A&M University
jonathan.ponniah@gmail.com

Yih-Chun Hu

University of Illinois at Urbana-Champaign
yihchun@illinois.edu

P. R. Kumar

Texas A&M University
prk@tamu.edu

now

the essence of knowledge

Boston — Delft



Documents

Document Sharing Hub

Foundations and Trends[®] in Networking

Published, sold and distributed by:

now Publishers Inc.
PO Box 1024
Hanover, MA 02339
United States
Tel. +1-781-985-4510
www.nowpublishers.com
sales@nowpublishers.com

Outside North America:

now Publishers Inc.
PO Box 179
2600 AD Delft
The Netherlands
Tel. +31-6-51115274

The preferred citation for this publication is

J. Ponniah, Y.-C. Hu, and P. R. Kumar. *A Clean Slate Approach to Secure Wireless Networking*. Foundations and Trends[®] in Networking, vol. 9, no. 1, pp. 1–105, 2014.

This Foundations and Trends[®] issue was typeset in L^AT_EX using a class file designed by Neal Parikh. Printed on acid-free paper.

ISBN: 9781680830491

© 2015 J. Ponniah, Y.-C. Hu, and P. R. Kumar

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, mechanical, photocopying, recording or otherwise, without prior written permission of the publishers.

Photocopying. In the USA: This journal is registered at the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923. Authorization to photocopy items for internal or personal use, or the internal or personal use of specific clients, is granted by now Publishers Inc for users registered with the Copyright Clearance Center (CCC). The 'services' for users can be found on the internet at: www.copyright.com

For those organizations that have been granted a photocopy license, a separate system of payment has been arranged. Authorization does not extend to other kinds of copying, such as that for general distribution, for advertising or promotional purposes, for creating new collective works, or for resale. In the rest of the world: Permission to photocopy must be obtained from the copyright owner. Please apply to now Publishers Inc., PO Box 1024, Hanover, MA 02339, USA; Tel. +1 781 871 0245; www.nowpublishers.com; sales@nowpublishers.com

now Publishers Inc. has an exclusive license to publish this material worldwide. Permission to use this content must be obtained from the copyright license holder. Please apply to now Publishers, PO Box 179, 2600 AD Delft, The Netherlands, www.nowpublishers.com; e-mail: sales@nowpublishers.com

Foundations and Trends® in Networking

Volume 9, Issue 1, 2014

Editorial Board

Editor-in-Chief

Anthony Ephremides
University of Maryland
United States

Editors

François Baccelli
University of Texas, Austin

Victor Bahl
Microsoft Research

Helmut Bölcskei
ETH Zurich

J.J. Garcia-Luna Aceves
UC Santa Cruz

Andrea Goldsmith
Stanford University

Roch Guerin
Washington University in Saint Louis

Bruce Hajek
UIUC

Jean-Pierre Hubaux
EPFL

Frank Kelly
University of Cambridge

P.R. Kumar
Texas A&M University

Steven Low
Caltech

Eytan Modiano
MIT

Keith Ross
Polytechnic Institute of NYU

Henning Schulzrinne
Columbia University

Mani Srivastava
UCLA

Leandros Tassioulas
Yale University

Lang Tong
Cornell University

Ozan Tonguz
Carnegie Mellon University

Don Towsley
University of Massachusetts, Amherst

Nitin Vaidya
UIUC

Pravin Varaiya
UC Berkeley

Roy Yates
Rutgers University

Raymond Yeung
Chinese University of Hong Kong

Editorial Scope

Topics

Foundations and Trends® in Networking publishes survey and tutorial articles in the following topics:

- Modeling and analysis of:
 - Ad hoc wireless networks
 - Sensor networks
 - Optical networks
 - Local area networks
 - Satellite and hybrid networks
 - Cellular networks
 - Internet and web services
- Protocols and cross-layer design
- Network coding
- Energy-efficiency incentives/pricing/utility-based
- Games (co-operative or not)
- Security
- Scalability
- Topology
- Control/Graph-theoretic models
- Dynamics and asymptotic behavior of networks

Information for Librarians

Foundations and Trends® in Networking, 2014, Volume 9, 4 issues. ISSN paper version 1554-057X. ISSN online version 1554-0588. Also available as a combined paper and online subscription.

A Clean Slate Approach to Secure Wireless Networking

Jonathan Ponniah
Texas A&M University
jonathan.ponniah@gmail.com

Yih-Chun Hu
University of Illinois at Urbana-Champaign
yihchun@illinois.edu

P. R. Kumar
Texas A&M University
prk@tamu.edu

Contents

1	Introduction	2
2	The Model	13
2.1	The Model Assumptions	18
3	A Game Between Protocols and Adversarial Strategies	21
3.1	A Game Between Protocols and Adversarial Strategies . . .	24
4	Closed Synchronized Networks	28
4.1	Overview	29
4.2	The Neighbor Discovery Phase	32
4.3	The Network Discovery Phase	33
4.4	The Scheduling Phase	34
4.5	The Data Transfer Phase	34
4.6	The Verification Phase	35
4.7	Steady State Behavior	35
4.8	Feasibility of Protocol and Optimality Proof	36
4.9	Concluding Remarks	48
5	Secure Clock Synchronization	50
5.1	Previous Work	52
5.2	A Motivating Example: The Man-In-The-Middle Attack . . .	53

5.3	The Clock Model	54
5.4	Impossibility Results in Clock Synchronization	55
5.5	Clock Synchronization in a Cycle	58
5.6	The Consistency Check	59
5.7	Main Result	63
5.8	Application to General Networks	75
5.9	Concluding Remarks	76
6	Closed Unsynchronized Networks	78
6.1	Model Assumptions	81
6.2	An Orthogonal MAC code	82
6.3	Forming a Rudimentary Synchronized Network	88
6.4	Controlling the Overhead	94
6.5	Conclusions	97
7	Conclusion	98
	Acknowledgements	100
	References	101

Abstract

The design of secure protocols for wireless ad-hoc networks is an important problem in communication systems research. A seemingly fundamental limitation of the design process is that any new protocol only addresses the vulnerabilities detected in its predecessors, leaving the remaining vulnerabilities unaffected. Hence, the design process amounts to an arms race between more sophisticated attacks and protocol fixes. To change this situation, a framework is needed for secure protocol design that offers provable performance and security guarantees against all possible attacks on the network.

This monograph proposes such a framework, contingent on some underlying model assumptions. The framework consists of a game defined between protocols and adversarial strategies in which the adversarial strategy is selected after the protocol has been revealed to all of the nodes. Each choice of protocol and adversarial strategy results in a payoff that corresponds to the functionality retained by the network, despite the adversarial activity. The design imperative is to choose the protocol that maximizes this payoff.

Two scenarios are considered: networks in which the nodes are initially synchronized and unsynchronized respectively. In each scenario, a protocol is described and three results are proved. First, the protocol is max-min optimal with respect to the payoff. The max-min payoff is the best that can be achieved because the protocol is always known to the adversarial nodes before the adversarial strategy is chosen. Second, the protocol is min-max optimal; there is a Nash equilibrium in the space of protocols and adversarial strategies. By implication, the adversarial nodes gain no strategic advantage from knowing protocol a priori. Finally, the adversarial nodes are effectively confined to one of two behavior modes: either jam or conform to the protocol, neither of which can be prevented by any protocol.

J. Ponniah, Y.-C. Hu, and P. R. Kumar. *A Clean Slate Approach to Secure Wireless Networking*. Foundations and Trends® in Networking, vol. 9, no. 1, pp. 1–105, 2014.

DOI: 10.1561/13000000037.

1

Introduction

A communication network is a set of terminal nodes (sources and destinations), intermediate nodes, and telecommunication links through which certain pairs of nodes are directly connected. The purpose of such a network is to route information generated by the source terminal nodes in the form of bits, via a connecting path of links, to the corresponding destination terminal nodes. These bits are transmitted electromagnetically across each link in the connecting path.

In wireless networks, as opposed to wired networks, the communication medium is simultaneously shared by all terminal nodes; the signals transmitted at one terminal can potentially reach all other terminals in the network. As a result, the signals of two terminals that transmit simultaneously may interfere with each other at the corresponding destination nodes, preventing either node from decoding the intended message. An *independent set* is a set of simultaneously activated links in which such mutual interference does not occur, and is determined by the physical properties of the wireless channel.

In order for messages to be reliably transmitted between any source-destination pair, either a centralized controller or the nodes themselves must know or discover the set of links in the network as well as the

independent sets of links, and by extension, the routes connecting each source-destination pair, and the joint selection of transmit power levels that correspond to an independent set. Furthermore, the centralized controller or the nodes themselves must decide on a desired throughput vector between all source-destination pairs and allocate the network resources to support this desired throughput.

Wireless *ad-hoc* networks, the focus of this monograph, are wireless networks that carry out these tasks without the aid of a centralized controller. Instead the nodes themselves must jointly determine a schedule of transmissions and receptions, consistent with the independent sets in the network, without a priori knowledge of the network topology. In addition, the nodes may be half-duplex in the sense that they cannot simultaneously transmit and receive, thus complicating the process by which the nodes can even acquire the minimal knowledge to form a rudimentary network.

A set of instructions given to each node, known as a *protocol*, enables the nodes as a collective to form a reliable network in which data can be transported. A *legitimate node*, by definition, follows the protocol exactly. An *adversarial node*, on the other hand, can behave arbitrarily or with malicious intent to prevent the legitimate nodes from forming a network. In this monograph, we confront the challenge of designing protocols for wireless ad-hoc networks infiltrated with adversarial nodes. Part of the challenge in designing such protocols is that many of the protocols already in widespread use predate contemporary demands for security, and are instead tailored for performance and low complexity.

We will provide a general description of the kinds of protocols that have been developed over the past few decades before focusing on protocols that specifically address issues of security.

Typically, protocols are either categorized as *reactive* or *pro-active*. Reactive (or “on-demand”) protocols determine a route from a source to a destination, if it is not already known, only after a source requests it. On the other hand, table-driven or proactive protocols maintain tables of all the known routes in the network, and ensure at considerable overhead, that these tables are periodically updated. Consequently, data packets can be routed more quickly since the routes are known a pri-

ori. The choice of a pro-active or re-active protocol reflects a trade-off between faster routing times and larger overhead, and is based on the computational power available in the network and the quality of service demanded by the users.

Another way of categorizing protocols is by “link-state” or “distance-vector routing”. A link-state protocol enables each node to retain a view of the entire network topology, whereas a distance-vector routing protocol allows each node to monitor its distance (in hops) from every destination node. The latter requires less overhead but is more sensitive to topological changes that alter the routes.

In general, a protocol caters to the specific computational limitations and environmental constraints in which the network operates. As a result, some protocols best fall into hybrid categories such as pro-active link state or on-demand distance vector routing. The diverse range of protocols available today is the outcome of extensive research and development to tailor protocols for every environmental contingency.

Within this larger set of protocols is a smaller subset developed explicitly with security and the possibility in mind that some of the nodes, being adversarial, may not cooperate with protocol. One particular strategy to deal with adversarial nodes is to add redundancy by dispersing essential activity throughout the network. To that end, there are protocols that replicate computational tasks [1] [2] [8] [9] [14] [15] [37] [40] [41], and storage tasks [3] [5] [7] [13] [35]. However, there is an adversarial counter-strategy that negates the defense offered by redundancy; the adversarial nodes could adopt multiple identities so as to artificially inflate their numbers. This attack (affectionately named “Sybil” after a fictional character suffering from multiple personality disorder) defeats any network that lacks a trusted central authority, regardless of the internal mechanisms employed to self-authenticate member identities.

A protocol strategy, to “counter” this adversarial counter-strategy, requires reliable methods for authenticating the authorship and content of network communication; in other words, encryption. Methods of encryption fall into one of two categories: symmetric, wherein the

users share a single key, or asymmetric wherein the source has a private key and the end-users have a public key. Symmetric encryption is more computationally efficient but is unable to prevent a malicious node that possess the key from forging packets or impersonating a user. Asymmetric encryption does not share this deficiency, but is computationally more complex than its symmetric counterpart by several orders of magnitude.

Encryption is also important because a small number of adversarial nodes can severely disrupt the network by forging or tampering with the control packets in a routing protocol. Some of the symptoms of such tampering include routing loops in which packets traverse a cycle and never reach their destination, black holes in which packets are dropped, or grey holes in which packets are selectively dropped, gratuitous detours which are unnecessarily long and/or circuitous routes when shorter routes are available, and artificial partitions of the network. These types of attacks, in which an adversarial node can disproportionately affect the network operation, are referred to as denial-of-service attacks.

To protect the network from Sybil and other denial-of-service attacks, some protocols incorporate specific encryption mechanisms into their operations. Examples include secure pro-active routing protocols with asymmetric encryption [30] [21] [23], [39], secure on-demand routing protocols with asymmetric encryption [44] [42] [38], secure pro-active routing protocols with symmetric encryption [4] [16] [43], and secure routing protocols with symmetric encryption for limited topologies such as between nodes and a base station [32], or between communicating routers [17].

This variety is partly due to the computational complexity of asymmetric encryption. Networks that use asymmetric encryption, but have limited computational capabilities are also vulnerable to a denial of service attack in which malicious nodes flood the network with bogus encrypted packets that are computationally expensive to decode. The resulting dilemma is whether to use symmetric encryption and accept the risk of packet tampering and forgery, or use asymmetric encryption and expose the network to a denial of service attack.

The TESLA broadcast authentication protocol [31] attempts to resolve this dilemma by using symmetric encryption in conjunction with a loosely synchronized network to periodically release new keys and ensure freshness. The protocol aims to provide the security inherent in asymmetric encryption but retain the smaller complexity of symmetric encryption. Two routing protocols that incorporate TESLA authentication methods are Ariadne [20], a secure on-demand routing protocol, and SEAD [18], a secure pro-active routing protocol.

Finally, some protocols attempt to secure the routing mechanism without recourse to encryption. For example, the Watchdog and Pathrater protocol [29] maintains a blacklist of nodes that appear to misbehave and penalizes them in the algorithm for determining routes. However, the protocol must strike a careful balance when penalizing nodes, because some of the nodes in the blacklist might be victims of false reporting by malicious nodes. To account for this possibility, the protocol also includes a process that allows nodes to be rehabilitated after they have been blacklisted.

Despite the efforts to secure networks via encryption and other means, structural vulnerabilities in certain protocol categories can still escape detection. The rushing attack [19] exposes such a vulnerability in the way route requests are forwarded in most source-routing protocols including those hardened with encryption. The proposed fix, called RAP (Rushing Attack Prevention) uses a modified version of DSR (Dynamic Source Routing) in which route requests from a source are stored and buffered, and one request from the buffer is randomly forwarded while the rest are discarded. The wormhole attack [6] is another example of an attack on a structural vulnerability. In this attack, an adversarial node creates an artificial link between two legitimate nodes in an attempt to control a substantial fraction of the network traffic. A patch for this attack uses what are called “packet leashes” to prevent packets from travelling too far from their transmitter.

It is also possible for an adversarial strategy to combine these attacks so as to increase their destructive effect. For instance, an adversarial node could hypothetically create a wormhole to tunnel a route request, accelerating a rushing attack, while simultaneously flooding

other routes with bogus encrypted packets to slow down the arrival of competing route requests.

There is an underlying trend in this survey of protocols for wireless ad-hoc networks and secure protocols specifically. That trend can be described as an arms race between attacks and fixes; in every protocol is a hidden vulnerability that once discovered leads to the development of a patch for that attack. Over time, the attacks become more sophisticated and the patches more elaborate. At no point in this process is it possible to provide any security guarantees or know whether there are vulnerabilities that have yet to be discovered.

The purpose of this monograph is to develop a system-theoretical approach to secure protocol design that provides provable and comprehensive security guarantees, where any features of a protocol that reduce complexity or maximize the network throughput do not come the expense of security. The desired approach is one of security first and performance second, which, as apparent from the survey, is the reverse of the current approach.

The efforts to design secure wireless ad-hoc networks are part of a deeper and more pervasive engineering challenge; designing complex systems that are secure. A system refers to a set of “interacting or interconnected components that together form an integrated whole”. Wireless ad-hoc networks are but one example of complex systems; other examples include economic systems, military systems, software systems, and power systems.

All of these systems are designed to carry out some system-specific function. In the case of wireless ad-hoc networks, the function is clear; to facilitate the exchange of information between all source-destination node pairs. A secure system, by definition retains its functionality even if some of the individual components of which it is composed, are compromised or attacked. Note that the term “security”, when applied to communication systems, can also be used in another context; to protect the privacy of information exchanged between a source and destination. We will limit our analysis exclusively to the former definition and not the latter. By this definition, a secure wireless ad-hoc network is one in which the legitimate source-destination nodes in the network are

able to reliably exchange data despite the non-cooperative or hostile behavior of the adversarial nodes. Examples of such behavior include all of the attacks described in the survey, in addition to less subtle acts of non-compliance; the adversarial nodes could choose to drop packets, advertise a wrong hop-count, jam, refuse to acknowledge a neighbor, lie about the local topology, in short do anything to undermine the operation of the network.

Naturally, there is a great deal of public interest and research invested in making complex systems secure. The philosophy that best describes the current approach towards secure system design is “defense-in-depth”; compartmentalize the system into more manageable subsystems, and defend each individual subsystem in a way that ensures the system functionality is retained even if other subsystems fail. This approach is understandable, given the near impossibility of comprehensively identifying all possible failure modes in a large complex system.

However, as evident in the application to wireless ad-hoc networks, the defense-in-depth philosophy has some inherent weaknesses. Namely, the possible existence of structural vulnerabilities that underly individual subsystems and the difficulty of completely anticipating, even within a subsystem, all of the ways in which an attack can occur.

In this monograph we propose a system-theoretic approach to secure protocol design in which comprehensive, provable security and performance guarantees can be made, contingent on some underlying model assumptions. As long as the assumptions hold, the guarantees retain their validity; to attack the protocol one must attack the underlying assumptions, and to harden the protocol to these types of attacks, the corresponding assumptions must increase in generality.

In a system-theoretic approach the notions of network “functionality” and “security” (the network functionality “retained” despite adversarial activity), must both be defined quantitatively. We will propose the following game-theoretic framework that addresses these requirements.

First the protocol is announced to all nodes, both legitimate and adversarial, since the adversarial nodes are not known to the protocol or the legitimate nodes a priori. Next, the adversarial nodes choose a

strategy q_p specifically tailored to the protocol p . At time $t = 0$ with respect to some global reference clock, each node turns on and proceeds to execute either p or q_p depending on whether the node is legitimate or adversarial respectively. Over a fixed operating lifetime, the source-destination node pairs exchange some number (possibly zero) of data packets. Let x denote the resulting throughput vector of all source-destination pairs in the network. We will assume that the “functionality” of the network is measured by some utility function $U(x)$ of x , where $U(x)$ reflects the priority that one source-destination pair in x is given relative to the others.

The “payoff” of this zero sum game, $J(p, q_p)$, is a quantitative measure of the “functionality retained” by the system, in this case a wireless ad-hoc network operating according to protocol p , in the aftermath of the adversarial strategy q_p . There are many ways of defining $J(p, q_p)$ but in general, one should expect $J(p, q_p)$ to depend on the legitimate source-destination pairs in x , the effective throughput attained by these source-destination pairs, and the utility function $U(x)$.

The problem of secure protocol design in this context is to choose the protocol $\tilde{p}$ that achieves the max-min payoff in the zero sum game defined by $J(p, q_p)$. In other words, given a priori knowledge of the protocol p , the adversarial nodes, in the worst case, choose a strategy $\tilde{q}_p$ to minimize the payoff $J(p, q_p)$ over all other adversarial strategies. The best any protocol can do, is maximize the worst-case payoff. That is, the best payoff that can be achieved over all adversarial strategies is:

$$\max_{\text{protocols } p} \min_{\text{attacks } q_p} J(p, q_p). \quad (1.1)$$

The max-min payoff in (1.1) is the best that any protocol can achieve no matter how the payoff function $J(p, q_p)$ is defined; the adversarial nodes always know the protocol a priori and can always choose a strategy that minimizes the payoff for a specific protocol. Hence a protocol that achieves (1.1) is proveably secure in the sense that the max-min payoff is guaranteed regardless of what the adversarial nodes do, and no protocol is guaranteed of doing better. For these reasons, the game-theoretic framework addresses the deficiencies in the defense-

in-depth approach to securing complex systems; namely, by allowing a quantitative measure of system security in the form of a payoff function, and the possibility of comprehensive, proveable security guarantees in the form of a protocol that is max-min optimal with respect to the payoff. These guarantees however, are contingent on the model assumptions that decide the payoff $J(p, q_p)$ for a specific choice of protocol p and adversarial strategy q_p .

The challenge then, is to find the protocol p that is max-min optimal with respect to $J(p, q_p)$. This problem is non-trivial because the strategy space of protocols and adversarial strategies is generally undefined. In the chapters to follow we will prove the following main results. First, we will describe a protocol that achieves the max-min payoff in (1.1) for a specific payoff $J(p, q_p)$ and a specific set of model assumptions. Next we will show that the protocol does even better and achieve the following payoff:

$$\min_{\text{attacks } q} \max_{\text{protocols } p} J(p, q). \quad (1.2)$$

Since the min-max is always greater than or equal to the max-min, and the max-min is the best any protocol can achieve, it follows that this game has a Nash equilibrium. In other words, there is a saddle-point in the optimization over protocols and adversarial strategies. The implication of (1.2) is that the attackers gain no advantage (in terms of reducing the payoff) by knowing the protocol beforehand; the payoff in (1.2) is the payoff achieved if the adversarial strategy is announced before the protocol.

Finally, let $\mathcal{Q}$ denote the set of strategies in which the adversarial nodes are restricted to either jamming or conforming to the protocol, actions which no protocol can prevent from occurring (in Chapter 3, we will explain why an adversarial node may choose to cooperate with the protocol). The following payoff is achievable:

$$\min_{\text{attacks } q \in \mathcal{Q}} \max_{\text{protocols } p} J(p, q). \quad (1.3)$$

We will describe a protocol that achieves (1.1), (1.2), and (1.3) for two different sets of model assumptions, increasing in generality, that concern the pre-existing level of clock synchronization in the network.

There are several reasons why clock synchronization specifically, and the notion of time more generally, is relevant to the discussion of wireless ad-hoc network security. First, the notion of time is essential for evaluating the effective throughput vector x , that is, the total number of data bits exchanged for each source-destination pair over the operating lifetime. The effective throughput is an important metric of the functionality, and by extension, security, of a wireless ad-hoc network.

In addition, a common reference time is also useful for distributed half-duplex nodes to coordinate their activity and/or jointly execute a common schedule; recall that in the wireless setting, two active links cause mutual interference if they don't belong to the same independent set. Moreover, even if all other nodes are silent, a pair of adjacent, half-duplex nodes will be prevented from exchanging information if both nodes are always simultaneously in transmit mode; a half-duplex node, by definition, is unable to simultaneously transmit and receive messages. For these reasons, clock synchronization is especially pertinent in the operation of wireless ad-hoc networks. However, achieving clock synchronization in a network with adversarial nodes is no trivial task; secure clock synchronization is an active area of research in distributed systems theory.

To simplify the presentation, we will first describe a protocol that is proveably secure under the following assumptions: the nodes turn “on” simultaneously and the local clocks at each node tick at the same rate. The resulting network is *closed* in the sense that the network once formed, does not expand to include new nodes, since no nodes turn “on” after the primordial birth. The nodes are also synchronized, since by assumption the local clocks are not subject to either an offset or a skew. Later, we will modify the protocol and show that similar security guarantees can be made for closed networks with unsynchronized, relatively affine local clocks. These results also appear in [33].

In Chapter 2 we describe the model assumptions on which the results are predicated. The assumptions describe valid modes of communication, topological constraints, encryption capabilities, and the parameters of the local clocks. The goal in this chapter is to chose model

assumptions that impose the minimal necessary conditions needed to provide any security guarantees.

In Chapter 3 we define a specific payoff function $J(p, q_p)$ in terms of any utility function $U(x)$. There are many ways in which $J(p, q_p)$ can be defined, but ultimately the payoff should correspond to the functionality “retained” by the network despite adversarial activity. The results in this monograph apply only for the payoff $J(p, q_p)$ defined in Chapter 3. In Chapter 4 we describe a protocol that operates under the assumption each node is synchronized and the nodes are born simultaneously. We will show that protocol achieves the payoff defined in (1.1), (1.2), and (1.3). In Chapter 5 we consider the model in which the local clocks are relatively affine. We will discuss the difficulties in synchronizing affine clocks when the network is infiltrated with adversarial nodes. Finally, in Chapter 6 we revisit Chapter 4 and show that the results in Chapter 4 hold under the assumption of relatively affine, unsynchronized local clocks. We will offer our conclusions and propose areas for further research in Chapter 7.

2

The Model

The security guarantees described in the previous chapter are contingent on four underlying model assumptions that describe the capabilities of the legitimate nodes and constraints on the network topology. This chapter develops concepts and definitions so we can state these assumptions precisely.

The model has two objectives: to capture the essential features of wireless operation; and impose the minimal, necessary conditions required for a network infiltrated with hostile nodes to function reliably.

First we explain the general setting, before describing specific features of the model. There are n processors with radio transceivers, or “nodes”, placed arbitrarily in some topographical domain. We will defer for now from discussing the processor hardware or the domain in which they are placed. The nodes are pre-programmed with a set of instructions, called p , that describe how a node should behave at every instant in time in response to its internal and external stimuli. We can assume that the instruction set, or *protocol* p , enables the collection of nodes to form a communication network, by regulating the local activity of each individual node.

Some of these nodes are by definition, *legitimate*, in that they follow the protocol exactly as specified, without caveat or exception. The legitimate nodes are *half-duplex*; they cannot simultaneously transmit and receive but instead at any time instant must choose between one of the two modes, transmit or receive. The nodes that are not legitimate, are by definition, *adversarial*; they are not obliged to follow the protocol, but are free to behave in any way they so choose or deem fit. The adversarial nodes are able to fully cooperate with each other and coordinate their actions over backchannels of infinite bandwidth. Moreover, the adversarial nodes know which nodes are legitimate and adversarial a priori, whereas the legitimate nodes do not. The adversarial nodes, after studying p , collectively decide on a new set of instructions q_p . The nodes, both legitimate and adversarial, then start and run their respective programs p and q_p for a predetermined amount of time.

The interaction between the protocol and the adversarial nodes yields an effective throughput between each source-destination pair during every time interval. In the next chapter, we define the utility accrued to the network based on the effective throughput over each such interval; here, we will clarify how the protocol and adversarial nodes interact to yield the resulting throughput in the first place.

The mechanism by which the legitimate nodes communicate through the physical medium is a *modulation scheme*. A legitimate *modulation scheme* m_i employed by node i corresponds to a single message from a larger set, and specifies a symbol in a signal constellation, the associated transmit power level, and an encoding rate (based on an encoding scheme such as LDPC or Reed-Solomon codes). In other words, a modulation scheme describes the physical action of a legitimate node at the interface of the wireless medium.

Whenever a modulation scheme is used, there are two possible outcomes of interest: the intended destination could either decode or fail to decode the corresponding message. We denote the transmission rate of the modulation scheme m_i in the event of a successful reception, by $r(m_i)$. We refer to $r(m_i)$ as the *intended* transmission rate, and distinguish $r(m_i)$ from the *actual* transmission rate which may be zero if the receiver fails to decode the associated message.

In addition to the modulation schemes just described, an adversarial node could also choose to emit random noise into the wireless channel to disrupt the transmissions of a neighboring node. This activity, which we refer to as *jamming*, is constrained only by the finite transmit power of the adversarial node.

A *concurrent transmission vector* c , is an $n \times (n - 1)$ vector of modulation schemes in which each element of the vector is a source modulation scheme for a corresponding one-hop source-destination pair. We use the term “one-hop” to indicate direct communication between the source and destination without the assistance of intermediate nodes; we will address multi-path routing later. The dimension of c is equal to the number of possible one-hop source-destination pairs in a network of n nodes.

Equivalently, a concurrent transmission vector can also be thought of as a set of edges, in which each edge e corresponds to a source-destination pair and the weight of an edge is equal to the rate of the corresponding modulation scheme.

A concurrent transmission vector c is *feasible* if every destination in the network decodes the message specified by the corresponding modulation scheme in c . There are several scenarios to consider. First, suppose the network is composed entirely of legitimate nodes. In this scenario, the feasibility of a concurrent transmission vector is determined exclusively by the physical properties of the channel; due to interference, the relative proximity of one source-destination pair to another may constrain the modulation schemes that can be simultaneously used by both pairs. Next, suppose the network also includes adversarial nodes and that these adversarial nodes can communicate via a backchannel of infinite bandwidth. In this scenario, the adversarial nodes enlarge the set of concurrent transmission vectors that are feasible. For example, consider a non-feasible concurrent transmission vector c in which the edges assigned non-zero rate consist of legitimate nodes, and replace these nodes with those that are adversarial. Now c can be “artificially” made feasible; the adversarial nodes simply communicate through the backchannel of infinite bandwidth while simulating the assigned modulation schemes over the wireless medium. Finally, suppose

the edges of non-zero weight are incident to both adversarial and legitimate nodes; the former can help the latter by relaying messages over the backchannel to adversarial nodes that neighbor the intended destinations. These adversarial nodes then replay the messages to the corresponding destinations over the wireless medium. Our definition of a feasible concurrent transmission vector includes all of these scenarios.

Let $\mathcal{F}$ denote the set of concurrent transmission vectors that are feasible due to the properties of the wireless channel as well as those that can be made “artificially” feasible with the assistance of the adversarial nodes.

We say a feasible concurrent transmission vector $c \in \mathcal{F}$ is *disabled* if the adversarial nodes prevent at least one message in c from being received at the intended destination. There are several ways in which this can occur. First, the adversarial nodes could choose to ignore their assigned modulation schemes in c and jam instead. To make this notion more precise, let G denote the set of legitimate (or good) nodes and B denote the set of adversarial (or bad) nodes. We can decompose the concurrent transmission vector into the following subvectors $c := c_{GG}c_{GB}c_{BG}c_{BB}$ where the subscript pair denotes the identity of the source and destination node respectively as legitimate or adversarial. For example c_{GB} denotes the vector of modulation schemes corresponding to edges in which the source is legitimate and the destination is adversarial. Now suppose the adversarial nodes choose to deploy a different set of modulation schemes $c'_{BG}c'_{BB}$ than those assigned $c_{BG}c_{BB}$. Note that $c'_{BG}c'_{BB}$ can include both jamming as well as the legitimate modulation schemes described earlier. Following this deployment, the concurrent transmission vector $c := c_{GG}c_{GB}c_{BG}c_{BB}$ is changed to $c' := c_{GG}c_{GB}c'_{BG}c'_{BB}$. Formally, we say that c is *actively disabled* if c' is not feasible (i.e., $c' \notin \mathcal{F}$), or any adversarial source or destination node in $c_{GB}c'_{BG}c'_{BB}$ refrains from transmitting or receiving a message prescribed in c . Informally, c is actively disabled if the adversarial nodes, by jamming, prevent a message from reaching its intended destination. In addition, an “artificially” feasible concurrent transmission vector can be disabled if the adversarial nodes decline to help the message reach the intended destination.

We say that an edge $e \in c$ is *actively disabled* if the corresponding message was not received by the destination following the deployment of c' . It follows from the definition of feasibility that a concurrent transmission vector c is actively disabled if at least one edge $e \in c$ is actively disabled.

Let $\mathcal{D}$ denote the set of concurrent transmission vectors that can be actively disabled. From the definition of an actively disabled concurrent transmission vector, the set $\mathcal{D}$ is protocol independent; if $c \in \mathcal{D}$ then c can be actively disabled regardless of the protocol that deploys c .

As we have stated, the set $\mathcal{D}$ essentially captures those concurrent transmission vectors that can be disabled by jamming. However, there are more sophisticated protocol-dependent strategies that the adversarial nodes can use to disable specific concurrent transmission vectors.

In order for a protocol to activate a feasible concurrent transmission vector $c \in \mathcal{F}$ at time t all of the nodes that correspond to non-zero rate edges must simultaneously activate at t . More precisely, the protocol must inform each node of the incident edge to activate and the time (with respect to the local clock) at which the edge is to be activated. This knowledge is only valid if the legitimate nodes share some degree of consensus with respect to a common reference time, network topology, and schedule. In a distributed network such knowledge is not granted a priori but must be acquired during the course of the network operation.

We say a feasible concurrent transmission vector $c \in \mathcal{F}$ is *passively disabled* at time t if the adversarial nodes prevent the legitimate nodes from arriving at the minimally required consensus. The types of attacks that disrupt consensus, qualitatively speaking, tend to be more subtle and unpredictable than straightforward jamming by adversarial nodes. Typically, these attacks occur when the adversarial nodes lie or disseminate false information about the network, the packets they have received, or the time on their clocks. Moreover, these attacks unlike jamming, are also protocol specific; those protocols that use naive or simplistic mechanisms to arrive at a consensus are more vulnerable to sabotage than those that do not.

2.1 The Model Assumptions

We are now ready to define the four model assumptions on which the results in this monograph are contingent. Let $\mathcal{M}$ denote the set of legitimate modulation schemes available to each node.

(M1) The set $\mathcal{M}$ is finite and known to all legitimate nodes a priori.

A straightforward consequence of assumption (M1) is that the set of concurrent transmission vectors composed of legitimate modulation schemes is also finite and known a priori. In other words, let $\mathcal{C}$ denote the set of concurrent transmission vectors $c := (m_1, \dots, m_{n(n-1)})$ where $m_j \in \mathcal{M}$ and each index j corresponds to one of the $n(n-1)$ source-destination pairs (or edges). It follows from (M1) that $\mathcal{C}$ is finite and known to all legitimate nodes a priori.

(M2) The legitimate nodes are connected in the following sense. There exists a graph G on the network in which each edge in the graph belongs to a concurrent transmission vector in $\mathcal{C}$. The subgraph L of legitimate nodes in G is connected and each edge in L belongs to a concurrent transmission vector that cannot be actively disabled. That is, for each edge $e \in L$ there exists some c such that $e \in c$ and $c \in \mathcal{F} \setminus \mathcal{D}$.

We leave open the possibility in (M2) that the edges in L can be *passively disabled* if the protocol in use is vulnerable to the types of attacks that disrupt consensus. Nevertheless, (M2) still makes strong assumptions about the network topology induced by the wireless channel; namely, that an edge in L never fails to deliver the messages prescribed to it. This assumption doesn't accurately model the effects of channel fading or probabilistic packet loss. We will leave such improvements for future work.

Although the assumption (that the underlying connected component of legitimate nodes suffers no packet loss) is strong, it is a well established result of distributed systems theory that a network must have some level of connectivity to guarantee consensus. In this respect, (M2) is minimally necessary.

(M3) Each node is given a private encryption key an identity certificate from a trusted authority. Encrypted messages cannot be tampered with or forged.

This assumption accurately models an ongoing trend in computational processing power; a trend in which computational processing power is continuously increasing. As a result, tasks such as asymmetric encryption which previously were too computationally expensive to perform efficiently can now be carried out on most wireless platforms. In conjunction with (M3) we will also assume that the nodes have infinite computational processing power.

The final assumption concerns the properties of the local clocks at each node. The notion of time will be an important component of our analysis; without it we cannot evaluate throughput, and without a measure of throughput we cannot evaluate network performance. A common reference time is also necessary to achieve some degree of coordination between distributed legitimate nodes.

We will consider two different variants of the assumption, each of which will be addressed in Chapter 4 and Chapters 5 and 6 respectively.

(M4) In Chapter 4 we assume the nodes are born simultaneously, and the local clocks start when the nodes turn on and tick at the same rate. That is, the local clocks are perfectly synchronized.

In Chapters 5 and 6 we modify the assumption again, so that the clocks do not tick at the same but are relatively affine with bounded relative skew. More precisely, let t be the time according to the local clock of node j , and let $\tau^i(t)$ be the time of the local clock at node i with respect to t . We assume that $\tau^i(t) := a_{ij}t + b_{ij}$ where a_{ij} and b_{ij} are constants that denote the relative skew and offset between node i and j respectively. Furthermore, we assume $\frac{1}{a_{max}} \leq a_{ij} \leq a_{max}$ for some $a_{max} \geq 1$ where a_{max} is known to all nodes a priori. The assumption that the local clocks are relatively affine, or remain within an affine envelope of each other is standard in the secure clock synchronization literature and models the observed behavior of clock oscillators in different environmental conditions. A bound on the relative skew, along with a priori knowledge of the bound are minimally necessary to make any performance guarantees. Without the bound, two local clocks can potentially be infinitely far apart, and therefore incapable of any interaction.

These four assumptions (M1)-(M4) form the basis of the analysis in this monograph. We will make comprehensive security and performance guarantees that can only be invalidated if the assumptions themselves are subverted. Examples of such subversion include packet loss on an edge in the underlying connected component of legitimate nodes in (M2), or breaking the encryption of a legitimate node in (M3). Conversely, to strengthen the guarantees, the assumptions must be loosened or generalized. It is our contention that this method of secure protocol design is a more effective way of ensuring security than an arm's race between protocols and attacker strategies.

In the next chapter we will quantify the concepts of network functionality and network security, and address the issue of designing protocols with security and performance guarantees.

3

A Game Between Protocols and Adversarial Strategies

The previous chapter described a model that specifies, to some extent, how the actions of the adversarial nodes affect the throughput between each source-destination pair in the network. In this chapter, we construct a zero-sum game in which the pay-off for a specific choice of protocol and corresponding adversarial strategy depends on the resulting throughput between legitimate source-destination pairs. The objective in the remaining chapters of this monograph will be to find the protocol that achieves the *optimal* payoff for this game over all protocols and adversarial strategies.

In this chapter it will be shown that the optimal protocol is max-min optimal with respect to the pay-off function; by definition the max-min pay-off is achievable regardless of the adversarial strategy. Hence this game-theoretic framework for secure protocol design offers comprehensive security and performances guarantees, contingent on the model assumptions described in the previous chapter.

We start by introducing a quantitative measure of the functionality of a network, namely, a utility function $U(x)$ of the network throughput x . The utility function measures the utility or benefit accrued to the network if the throughput vector between all source-destination pairs

is equal to x . Suppose $x := (x_1, \dots, x_{n(n-1)})$. Two common utility functions that appear in the literature are the minimum throughput, that is, $U(x) := \min_i x_i$ and the average throughput, $U(x) := \frac{1}{n(n-1)} \sum_i x_i$. The analysis to follow applies to any choice of utility function $U(x)$ that is continuous in the components of x .

First we will restate some definitions from the previous chapter. A *modulation scheme* m is defined as a mode of communication between a source-destination pair consisting of a symbol in a signal constellation, a transmit power level, and an encoding rate. The combined selection of these parameters corresponds to an effective transmission rate, denoted by $r(m)$. Alternatively, a modulation scheme m can be associated with a single weighted edge e in the network graph, where the edge weight is equal to the effective transmission rate of the corresponding modulation scheme.

A *concurrent transmission vector* $c := (m_1, \dots, m_{n(n-1)})$ is by definition, a vector of modulation schemes, where each element in the vector corresponds to a source-destination pair. In a network with n nodes, a concurrent transmission vector consists of $n(n-1)$ elements; the total number of possible source-destination pairs. A concurrent transmission vector is equivalent to a set of $n(n-1)$ weighted edges, where each edge e represents one of the source-destination pairs, and the associated weight denotes the effective transmission rate of the corresponding modulation scheme.

Finally, a *feasible* concurrent transmission vector is defined as a concurrent transmission vector in which the destination of every source-destination pair receives the transmitted message, contingent on the cooperation and/or assistance of adversarial nodes in the network.

Now suppose for the moment that all nodes in the network are legitimate. Given a set of feasible concurrent transmission vectors $\mathcal{F} := \{c_i, i = 1, \dots, N\}$, the classical scheduling problem is to find the schedule or time-share of concurrent transmission vectors that maximizes the network utility:

$$\max U(x) \tag{3.1}$$

subject to the constraints

$$\sum_{i=1}^N \gamma_i \leq 1, \quad (3.2)$$

$$\sum_{i=1}^N \gamma_i r(c_i) = z, \quad (3.3)$$

$$\sum_{p:(n,m) \in p} y_p \leq z_{(n,m)}, \quad (3.4)$$

$$\sum_{p:p \in (s_i, d_i)} y_p = x_{(s_i, d_i)}. \quad (3.5)$$

The first constraint (3.2) is a time-sharing constraint; each γ_i is the fraction of the operating lifetime allocated to the concurrent transmission vector c_i . The constraint in (3.2) simply states that the total fraction of time allocated to all concurrent transmission vectors cannot exceed one.

For some edge e and concurrent transmission vector c_i , let $r(c_i)$ denote a rate vector, where each element in $r(c_i)$ is the effective rate of the corresponding modulation scheme in c_i and $r(c_i(e))$ is the effective rate of the modulation scheme corresponding to edge e in c_i . Let $z(e)$ denote the link capacity of edge e ; the effective transmission rate of edge e for a specific schedule $\{\gamma_i, i = 1, \dots, N\}$, evaluated over the operating lifetime. The link capacity of edge e is the time-averaged weight of that edge over all concurrent transmission vectors; $z(e) := \sum_i \gamma_i r(c_i(e))$. If z denotes a vector of all $n(n-1)$ edges in the network, the corresponding link capacity constraint is given by (3.3).

It follows that the total throughput of all routes that traverse edge e cannot exceed the link capacity of the edge $z(e)$; hence the throughput constraint (3.4).

The total throughput accrued by a single source-destination pair is the combined throughput over all routes that connect this pair, giving (3.5).

The solution to (3.1) is difficult to solve for general utility functions U , but reduces to a convex program when U is convex. In this monograph we do not address the computational difficulties of evaluating (3.1). Instead the focus is on deriving an equivalent notion of the

network functionality, when some of the source-destination pairs in the network are adversarial. More precisely, we would like to measure the functionality retained by the network when the adversarial nodes behave subversively, and then determine a protocol that is optimal with respect to this measure. The remainder of this chapter will be devoted to this effort.

3.1 A Game Between Protocols and Adversarial Strategies

Given the model and setting described in the previous chapter, we define the following zero-sum game between protocols and adversarial strategies. A protocol p is announced to all nodes, legitimate and adversarial. The protocol is designed to create a functioning network out of the distributed collection, wherein legitimate source-destination pairs can communicate reliably. The adversarial nodes choose a strategy q_p tailored to protocol p designed to subvert the network operation. We say the pay-off for this game, $J(p, q_p)$, qualitatively speaking, is the functionality retained by network given protocol p and adversarial strategy q_p . Before defining $J(p, q_p)$ quantitatively, we state an important property about the optimal protocol for any pay-off function $J(\cdot)$; the best possible pay-off that can be achieved by a protocol is:

$$\max_{\text{protocols } p} \min_{\text{attacker strategies } q_p} J(p, q_p). \quad (3.6)$$

This property simply follows from the setup of the game; the adversarial nodes, knowing the protocol a priori, are able to choose the protocol-specific strategy that minimizes the payoff. The best we can do is choose a protocol that maximizes the worst case pay-off.

We will propose a $J(p, q_p)$ that is a quantitative measure of the functionality retained by the network given protocol p and adversarial strategy q_p . Since the pay-off function is not obvious, we will start with what appears to be a reasonable choice for $J(p, q_p)$ and successively modify this choice appropriately.

Let us start by setting $J(p, q_p)$ equal to $U(x)$; the utility accrued to the network from operating at a throughput vector x . This pay-off function may seem reasonable, but it suffers from the following drawback; the throughput vector x includes both legitimate and adversarial

source-destination pairs. Any measure of the functionality retained by the network should ideally exclude the throughput and service time allocated to adversarial nodes.

Let L denote the set of legitimate nodes, and x_L denote the throughput vector x restricted to the source-destination pairs composed of legitimate nodes. Now, set $J(p, q_p)$ equal to $U(x_L)$. Here we run into another problem; as per the setting described in Chapter 2, the legitimate nodes do not know a priori which nodes are legitimate and which nodes are adversarial. Moreover, an adversarial node that chooses to conform to perfectly conform to the protocol is indistinguishable from a legitimate node, from an external point of view. Therefore, unless the adversarial nodes betray their identity with overtly malicious behavior, no protocol is assured of maximizing $U(x_L)$.

Furthermore, it turns out that the adversarial nodes may have an incentive to conform to the protocol; such a strategy may be a far more effective means of reducing the throughput to the legitimate source-destination pairs than any subversive activity. For example, let $U(x) = \min_i x_i$ and consider a network of three nodes, two of which are legitimate and one adversarial, where the node that is adversarial is located far away from the nodes that are legitimate. An adversarial node that chooses to jam will have limited impact on the channel between the legitimate nodes. On the other hand, an adversarial node that cooperates with the protocol will draw a disproportionate share of the service time, since the link incident to the adversarial node is relatively weaker than the link between the legitimate nodes. This type of attack, known as the partial-deafness attack, can significantly reduce the service time to the legitimate nodes. Without a priori knowledge of the network topology, this kind of attack cannot be stopped.

Since $U(x_L)$ in general cannot be optimized by any protocol, let L' denote the set of nodes, legitimate and possibly adversarial, that conform perfectly to the protocol over the entire operating lifetime and let $J(p, q_p)$ be $U(x_{L'})$. This pay-off, though more tractable than its predecessor, is still difficult to optimize because the legitimate nodes may not be able to identify L' ; some adversarial nodes may be able to selectively conform to the protocol and still evade detection.

We will define a pay-off function that approximates $U(x_{L'})$ and in subsequent chapters describe protocols that are max-min optimal with respect to this pay-off. At the time of this writing, a protocol solution to the game in which the pay-off is $U(x_{L'})$ remains an open problem.

In defining $J(p, q_p)$ we will take advantage of model assumption (M2) described in the previous chapter to restrict x to those source-destination pairs that are not overtly malicious.

Given a protocol p and adversarial strategy q_p let $\mathcal{D}_{p,q}(t)$ denote the set of concurrent transmission vectors disabled (both actively and passively) at time t . Let $\mathcal{F}$ denote the set of feasible concurrent transmission vectors. It should be emphasized that the protocol does not know q , $\mathcal{D}_{p,q}(t)$, or $\mathcal{F}$ a priori. Now consider the graph $G(t)$ induced from the set of all legitimate nodes and the edges in $\mathcal{F}$ that have not been disabled. That is, the edges in $\mathcal{F} \setminus \mathcal{D}_{p,q}(t)$. Note that the graph $G(t)$ may include adversarial nodes that conform, or selectively conform to the protocol for reasons discussed earlier. In addition, the legitimate nodes in $G(t)$ may not necessarily be connected; $\mathcal{D}_{p,q}(t)$ includes concurrent transmission vectors that are both actively *and* passively disabled. Both terms are defined in the previous chapter but an important distinction is that the legitimate nodes are only connected, per assumption (M2), in the sense that the underlying subgraph is composed of edges that cannot be actively disabled. The set of edges that can be passively disabled is protocol-specific, and possibly includes edges in this underlying subgraph.

Since there are a finite number of concurrent transmission vectors and those that are disabled remain disabled for the remainder of the operating lifetime, it follows that $G(t)$ maps to a finite sequence of graphs $\{G_i\}$. Let α_i denote the fraction of the operating lifetime in which the i th graph G_i is active, and let x_{G_i} denote the effective throughput vector of all source-destination pairs in G_i evaluated over this fraction of the operating lifetime.

We will define the pay-off function as the time-averaged utility accrued to each subgraph G_i in the execution of p and q_p :

$$J(p, q_p) := \sum_i \alpha_i U(x_{G_i}). \quad (3.7)$$

We make some observations about $J(p, q_p)$ as defined in (3.7). First of all, the pay-off excludes from the throughput vector x , all nodes that are overtly adversarial; those adversarial nodes that remove themselves from the underlying graph in which the legitimate nodes are connected. As a result, the pay-off $J(p, q_p)$ to some extent reflects the functionality retained by the network in the face of adversarial activity. However, an adversarial node that selectively conforms to the protocol, by disabling some concurrent transmission vectors but enabling others, can still remain in the graph of legitimate nodes provided it is connected to the graph by at least one edge. For this reason $J(p, q_p)$ is an imperfect approximation of $U(x_{L'})$.

One other deficiency of the payoff defined in (3.7) is that the adversarial nodes are not able to fully exploit their knowledge of the identities of the legitimate nodes; the utility is evaluated over subgraphs G_i that may include adversarial nodes. Therefore, the protocol is not penalized for allocating service time to adversarial nodes in the subgraphs $\{G_i\}$.

This being a first attempt to formalize secure protocol design as a game-theory problem, we will address these important issues in future work. In the next chapter, we will describe a protocol that achieves the min-max pay-off under the model assumption of perfect clock synchronization.