Security and Cooperation in Wireless Networks

Tutorial at ACM MobiCom/MobiHoc 2007

Jean-Pierre Hubaux and Panos Papadimitratos

{firstname.lastname}@epfl.ch
Security and Cooperation in Wireless Networks

Part 1: New Wireless Networks and New Challenges

Jean-Pierre Hubaux

jean-pierre.hubaux@epfl.ch
Security and Cooperation in Wireless Networks

Thwarting Malicious and Selfish Behavior in the Age of Ubiquitous Computing

Levente Buttyan and Jean-Pierre Hubaux

With contributions from N. Ben Salem, M. Cagalj, S. Capkun, M. Felegyhazi, T. Holczer, P. Papadimitratos, P. Schaffer, and M. Raya

http://secowinet.epfl.ch
Security and Cooperation in Wireless Networks

1. Introduction
2. Thwarting malicious behavior
3. Thwarting selfish behavior
The Internet: something went wrong

Network deployment

Observation of new misdeeds (malicious or selfish)

Install security patches (anti-virus, anti-spam, anti-spyware, anti-phishing, firewalls,…)

“The Internet is Broken”
⇒ NSF FIND, GENI, etc.
Where is this going?


The Economist, April 28, 2007

What if tomorrow’s wireless networks are even more unsafe than today’s Internet?
Upcoming wireless networks

• New kinds of networks
  – Personal communications
    • Small operators, community networks
    • Cellular operators in shared spectrum
    • Mesh networks
    • Hybrid ad hoc networks (also called “Multi-hop cellular networks”)
    • “Autonomous” ad hoc networks
    • Personal area networks
  – Vehicular networks
  – Sensor and RFID networks
  – …

• New wireless communication technologies
  – Cognitive radios
  – MIMO
  – Ultra Wide Band
  – Directional antennas
  – …
Upcoming wireless networks

• New kinds of networks
  – Personal communications
    • Small operators, community networks
    • Cellular operators in shared spectrum
    • Mesh networks
    • Hybrid ad hoc networks (also called “Multi-hop cellular networks”)
    • “Autonomous” ad hoc networks
    • Personal area networks
  – Vehicular networks
  – Sensor and RFID networks
  – …

• New wireless communication technologies
  – Cognitive radios
  – MIMO
  – Ultra Wide Band
  – Directional antennas
  – …
Community networks

Example: service reciprocation in community networks

- Incentive technique based on proof of contribution


- Distributed solution:
  E. Pantelis, A. Frangoudis, and G. Polyzos
  Stimulating Participation in Wireless Community Networks
  INFOCOM 2006
Mesh Networks
Mesh Networks: node compromise
Mesh Networks: jamming

More on mesh networks:
• IEEE Wireless Communications, Special Issue on Wireless Mesh Networking, Vol. 13 No 2, April 2006
Vehicular networks: why?

- Combat the awful side-effects of road traffic
  - In the EU, around 40’000 people die yearly on the roads; more than 1.5 millions are injured
  - Traffic jams generate a tremendous waste of time and of fuel
- Most of these problems can be solved by providing appropriate *information* to the driver or to the vehicle
Example of attack: Generate “intelligent collisions”

- All carmakers are working on vehicular comm.
- Vehicular networks will probably be the largest incarnation of mobile ad hoc networks

For more information:
http://ivc.epfl.ch
http://www.sevecom.org
Sensor networks

Vulnerabilities:
• Theft ➔ reverse engineered and compromised, replicated
• Limited capabilities ➔ risk of DoS attack, restriction on cryptographic primitives to be used
• Deployment can be random ➔ pre-configuration is difficult
• Unattended ➔ some sensors can be maliciously moved around
RFID

- RFID = Radio-Frequency Identification

- RFID system elements
  - RFID tag + RFID reader + back-end database

- RFID tag = microchip + RF antenna
  - microchip stores data (few hundred bits)
  - Active tags
    - have their own battery → expensive
  - Passive tags
    - powered up by the reader’s signal
    - reflect the RF signal of the reader modulated with stored data
Trends and challenges in wireless networks

• From centralized to distributed to self-organized ➔ Security architectures must be redesigned
• Increasing programmability of the devices ➔ increasing risk of attacks and of greedy behavior
• Growing number of tiny, embedded devices ➔ Growing vulnerability, new attacks
• From single-hopping to multi-hopping ➔ Increasing “security distance” between devices and infrastructure, increased temptation for selfish behavior
• Miniaturization of devices ➔ Limited capabilities
• Pervasiveness ➔ Growing privacy concerns

... Yet, mobility and wireless can facilitate certain security mechanisms
Grand Research Challenge

Prevent ubiquitous computing from becoming a pervasive nightmare
Reasons to trust organizations and individuals

• Moral values
  – Culture + education, fear of bad reputation

• Experience about a given party
  – Based on previous interactions

• Rule enforcement organization
  – Police or spectrum regulator

• Usual behavior
  – Based on statistical observation

• Rule enforcement mechanisms
  – Prevent malicious behavior (by appropriate security mechanisms) and encourage cooperative behavior

Will lose relevance

Scalability challenge

Can be misleading
## Upcoming wireless networks vs. mechanisms

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Small operators, community networks</td>
<td>X X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Cellular operators in shared spectrum</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mesh networks</td>
<td>X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hybrid ad hoc networks</td>
<td>X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Self-organized ad hoc networks</td>
<td>X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vehicular networks</td>
<td>X X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sensor networks</td>
<td>X</td>
<td>?</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>RFID networks</td>
<td>X</td>
<td>?</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Security**
- **Cooperation**
Security and Cooperation in Wireless Networks

1. Introduction

2. Thwarting **malice**: security mechanisms
   2.1 Naming and addressing
   2.2 Establishment of security associations
   2.3 Secure neighbor discovery
   2.4 Secure routing in multi-hop wireless networks
   2.5 Privacy protection
   2.6 Secure positioning

3. Thwarting **selfishness**: behavior enforcement
   3.0 Brief introduction to game theory
   3.1 Enforcing fair bandwidth sharing at the MAC layer
   3.2 Enforcing packet forwarding
   3.3 Wireless operators in a shared spectrum
   3.4 Secure protocols for behavior enforcement
2.1 Naming and addressing

• Typical attacks:
  – Sybil: the same node has multiple identities
  – Replication: the attacker captures a node and replicates it
    several nodes share the same identity

• Distributed protection technique in IPv6: Cryptographically Generated Addresses (T. Aura, 2003; RFC 3972)

For higher security (hash function output beyond 64 bits), hash extension can be used

IPv6 address

2.2 Pairwise key establishment in sensor networks

1. Initialization
   - Key reservoir (k keys)
   - $m << k$ keys in each sensor ("key ring of the node")

2. Deployment
   - Probability for any 2 nodes to have a common key:
     \[ p = 1 - \frac{((k - m)!)^2}{k!(k - 2m)!} \]
Probability for two sensors to have a common key

Eschenauer and Gligor, *ACM CCS 2002*

See also:
- Karlof, Sastry, Wagner: TinySec, *Sensys 2004*
2.3 Securing Neighbor Discovery: Thwarting Wormholes

- Routing protocols will choose routes that contain wormhole links
  - typically those routes appear to be shorter
  - Many of the routes (e.g., discovered by flooding based routing protocols such as DSR and Ariadne) will go through the wormhole
- The adversary can then monitor traffic or drop packets (DoS)
Wormholes are not specific to ad hoc networks

Hu, Perrig, and Johnson
Packet leashes: a defense against wormhole attacks in wireless networks
INFOCOM 2003
2.4 Secure routing in wireless ad hoc networks

Exchange of messages in Dynamic Source Routing (DSR):

- Routing disruption attacks
  - routing loop
  - black hole / gray hole
  - partition
  - detour
  - wormhole

- Resource consumption attacks
  - injecting extra data packets in the network
  - injecting extra control packets in the network
Operation of Ariadne illustrated

A → *: [req, A, H, MAC$_{KAH}$, (), ()]
E → *: [req, A, H, h(E|MAC$_{KAH}$), (E), (MAC$_{KE,i}$)]
F → *: [req, A, H, h(F|h(E|MAC$_{KAH}$)), (E, F), (MAC$_{KE,i}$, MAC$_{KF,i}$)]

H → F: [rep, H, A, (E, F), (MAC$_{KE,i}$, MAC$_{KF,i}$), MAC$_{KHA}$, ()]
F → E: [rep, H, A, (E, F), (MAC$_{KE,i}$', MAC$_{KF,i}$'), MAC$_{KHA}$, (K$_{F,i}$)]
E → A: [rep, H, A, (E, F), (MAC$_{KE,i}$, MAC$_{KF,i}$), MAC$_{KHA}$, (K$_{F,i}$, K$_{E,i}$)]
Secure route discovery with the Secure Routing Protocol (SRP)

Route Request (RREQ): \( S, T, Q_{SEQ}, Q_{ID}, MAC(K_{S,T}, S, T, Q_{SEQ}, Q_{ID}) \)
(1) \( S \) broadcasts \( RREQ \);
(2) \( V_1 \) broadcasts \( RREQ, V_1 \);
(3) \( V_2 \) broadcasts \( RREQ, V_1, V_2 \);
(4) \( V_3 \) broadcasts \( RREQ, V_1, V_2, V_3 \);

Route Reply (RREP): \( Q_{ID}, T, V_3, V_2, V_1, S, \)
\( MAC(K_{S,T}, Q_{ID}, Q_{SEQ}, T, V_3, V_2, V_1, S) \)
(5) \( T \rightarrow V_3 : RREP \);
(6) \( V_3 \rightarrow V_2 : RREP \);
(7) \( V_2 \rightarrow V_1 : RREP \);
(8) \( V_1 \rightarrow S : RREP \);

\( Q_{SEQ} \): Query Sequence Number
\( Q_{ID} \): Query Identifier
More on secure routing

- **Secure Route Discovery**
  - *Sangrizi, Dahill, Levine, Shields, and Royer:* ARAN, Nov. 2002
  - *Zapata and Asokan:* S-AODV, Sept. 2002

  All above proposals are difficult to assess
  - *G. Ács, L. Buttyán, and I. Vajda:* Provably Secure On-demand Source Routing

- **Secure Data Communication**
  - *Papadimitratos and Haas:* Secure Single Path (SSP) and Secure Multi-path (SMT) protocols,

- **Cross-layer attacks**
  - *Aad, Hubaux, Knightly:* Jellyfish attacks, 2004
2.5 Privacy: the case of RFID

- RFID = Radio-Frequency Identification

- RFID system elements
  - RFID tag + RFID reader + back-end database

- RFID tag = microchip + RF antenna
  - microchip stores data (few hundred bits)
  - Active tags
    • have their own battery → expensive
  - Passive tags
    • powered up by the reader’s signal
    • reflect the RF signal of the reader modulated with stored data
RFID privacy problems

- RFID tags respond to reader’s query automatically, without authenticating the reader
  - clandestine scanning of tags is a plausible threat

- Two particular problems:
  1. **Inventorying**: a reader can silently determine what objects a person is carrying
     - books
     - medicaments
     - banknotes
     - underwear
     - ...
  2. **Tracking**: set of readers can determine where a given person is located
     - tags emit fixed unique identifiers
     - even if tag response is not unique it is possible to track a set of particular tags

2.6 Secure positioning

a) Node displacement

b) Wormhole

c) Malicious distance enlargement

d) Dissemination of false position and distance information

http://www.syssec.ethz.ch/research/spot

v - honest node
m - malicious node
c - compromised node
Security and Cooperation in Wireless Ad Hoc Networks

1. Introduction
2. Thwarting *malice*: security mechanisms
   2.1 Naming and addressing
   2.2 Establishment of security associations
   2.3 Secure neighbor discovery
   2.4 Secure routing in multi-hop wireless networks
   2.5 Privacy protection
   2.6 Secure positioning
3. Thwarting *selfishness*: behavior enforcement
   3.0 Brief introduction to game theory
   3.1 Enforcing fair bandwidth sharing at the MAC layer
   3.2 Enforcing packet forwarding
   3.3 Wireless operators in a shared spectrum
   3.4 Secure protocols for behavior enforcement
3.0 Brief introduction to Game Theory

• Discipline aiming at modeling situations in which actors have to make decisions which have mutual, possibly conflicting, consequences
• Classical applications: economics, but also politics and biology
• Example: should a company invest in a new plant, or enter a new market, considering that the competition could make similar moves?
• Most widespread kind of game: non-cooperative (meaning that the players do not attempt to find an agreement about their possible moves)
Example 1: The Forwarder’s Dilemma
From a problem to a game

• Users controlling the devices are *rational* (or *selfish*): they try to maximize their benefit

• Game formulation: $G = (P, S, U)$
  – $P$: set of players
  – $S$: set of strategy functions
  – $U$: set of utility functions

• **Strategic-form** representation

<table>
<thead>
<tr>
<th></th>
<th>Forward</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forward</td>
<td>$(1-c, 1-c)$</td>
<td>$(-c, 1)$</td>
</tr>
<tr>
<td>Drop</td>
<td>$(1, -c)$</td>
<td>$(0, 0)$</td>
</tr>
</tbody>
</table>

• Reward for packet reaching the destination: 1
• Cost of packet forwarding: $c$ ($0 < c << 1$)
Solving the Forwarder’s Dilemma (1/2)

**Strict dominance:** strictly best strategy, for any strategy of the other player(s)

Strategy $s_i$ strictly dominates if

$$u_i(s_i', s_{-i}) < u_i(s_i, s_{-i}), \forall s_{-i} \in S_{-i}, \forall s_i' \in S_i$$

where: $u_i \in U$ utility function of player $i$

$s_{-i} \in S_{-i}$ strategies of all players except player $i$

In Example 1, strategy Drop *strictly dominates* strategy Forward

<table>
<thead>
<tr>
<th></th>
<th><strong>Forward</strong></th>
<th><strong>Drop</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Forward</strong></td>
<td>(1-c, 1-c)</td>
<td>(-c, 1)</td>
</tr>
<tr>
<td><strong>Drop</strong></td>
<td>(1, -c)</td>
<td>(0, 0)</td>
</tr>
</tbody>
</table>
Solution by iterative strict dominance:

<table>
<thead>
<tr>
<th></th>
<th>Forward</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blue</td>
<td>(1-c, 1-c)</td>
<td>(1, -c)</td>
</tr>
<tr>
<td>Green</td>
<td>(-c, 1)</td>
<td>(0, 0)</td>
</tr>
</tbody>
</table>

Drop strictly dominates Forward

But Forward would result in a better outcome
Nash equilibrium

Nash Equilibrium: no player can increase his utility by deviating unilaterally

The Forwarder’s Dilemma

<table>
<thead>
<tr>
<th></th>
<th>Forward</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blue</td>
<td>$(1-c, 1-c)$</td>
<td>$(-c, 1)$</td>
</tr>
<tr>
<td>Green</td>
<td>$(1, -c)$</td>
<td>$(0, 0)$</td>
</tr>
</tbody>
</table>

(Drop, Drop) is the **only** Nash equilibrium of this game
Example 2: The Multiple Access game

Reward for successful transmission: 1
Cost of transmission: c (0 < c << 1)

There is no strictly dominating strategy
There are two Nash equilibria
More on game theory

Pareto-optimality
A strategy profile is Pareto-optimal if the payoff of a player cannot be increased without decreasing the payoff of another player.

Properties of Nash equilibria to be investigated:
- uniqueness
- efficiency (Pareto-optimality)
- emergence (dynamic games, agreements)

Promising area of application in wireless networks: cognitive radios
Security and Cooperation in Wireless Networks

1. Introduction

2. Thwarting malice: security mechanisms
   2.1 Naming and addressing
   2.2 Establishment of security associations
   2.3 Secure neighbor discovery
   2.4 Secure routing in multi-hop wireless networks
   2.5 Privacy protection
   2.6 Secure positioning

3. Thwarting selfishness: behavior enforcement
   3.0 Brief introduction to game theory
   3.1 Enforcing fair bandwidth sharing at the MAC layer
   3.2 Enforcing packet forwarding
   3.3 Wireless operators in a shared spectrum
   3.4 Secure protocols for behavior enforcement
3.1 Enforcing fair bandwidth sharing at the MAC layer

The access point is *trusted*

- Kyasanur and Vaidya, *DSN 2003*
- [http://domino.epfl.ch](http://domino.epfl.ch)
- Cagalj et al., *Infocom 2005* (game theory model for CSMA/CA ad hoc networks)
3.2 Enforcing packet forwarding

Usually, the devices are assumed to be cooperative. But what if they are not, and there is no incentive to cooperate?

• V. Srinivasan, P. Nuggehalli, C. Chiasserini, and R. Rao, Infocom 2003, IEEE TWC 2005
Modeling packet forwarding as a game

Player: node

Strategy: cooperation level

Payoff of node i: proportion of packets sent by node i reaching their destination
3.3 Games between wireless operators
Multi-domain sensor networks

• Typical cooperation: help in packet forwarding
• Can cooperation emerge spontaneously in multi-domain sensor networks based solely on the self-interest of the sensor operators?
3.3 Border games of cellular operators (1/3)
3.3 Border games of cellular operators (2/3)

- Two CDMA operators: A and B
- Adjust the pilot signals
- Power control game (no power cost):
  - players = operators
  - strategies = pilot powers
  - payoffs = attracted users (best SINR)

Signal-to-interference-plus-noise ratio

$$SINR_{A\nu}^{\text{pilot}} = \frac{G_p^{\text{pilot}} \cdot P_A \cdot d_{A\nu}^{-\alpha}}{N_0 \cdot W + I_{\text{own}}^{\text{pilot}} + I_{\text{other}}^{\text{pilot}}}$$

Own-cell interference

$$I_{\text{own}}^{\text{pilot}} = \varsigma \cdot d_{A\nu}^{-\alpha} \left( \sum_{w \in \mathcal{M}_A} T_{Aw} \right)$$

Other-to-own-cell interference

$$I_{\text{other}}^{\text{pilot}} = \eta \cdot d_{B\nu}^{-\alpha} \left( P_B + \sum_{w \in \mathcal{M}_B} T_{Bw} \right)$$

where:

- $G_p^{\text{pilot}}$ – pilot processing gain
- $P_A^{\text{p}}$ – pilot signal power of BS A
- $d_{A\nu}^{-\alpha}$ – path loss between A and v
- $\varsigma$ – own-cell interference factor
- $\eta$ – other-to-own-cell interference factor
- $T_{Aw}$ – traffic signal power assigned to w by BS A
- $\mathcal{M}_A$ – set of users attached to BS A
3.3 Border games of cellular operators (3/3)

- Unique and Pareto-optimal Nash equilibrium
- Higher pilot power than in the standard $P^s = 2W$
- 10 users in total

Extended game with power costs = Prisoner’s Dilemma

<table>
<thead>
<tr>
<th>Player A</th>
<th>$P^s$</th>
<th>$P^*$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_A$</td>
<td>$U + \Delta - C^*, U - \Delta$</td>
<td>$U - C^<em>, U - C^</em>$</td>
</tr>
<tr>
<td>Player B</td>
<td>$U, U$</td>
<td>$U - \Delta, U + \Delta - C^*$</td>
</tr>
</tbody>
</table>

where:

- $U$ – fair payoff (half of the users)
- $\Delta$ – payoff difference by selfish behavior
- $C^*$ - cost for higher pilot power
3.4 Secure protocols for behavior enforcement

- Self-organized ad hoc network
- Investigation of both routing and packet forwarding


*Mobicom 2005*
Who is malicious? Who is selfish?

There is no watertight boundary between malice and selfishness

¬ Both security and game theory approaches can be useful
From discrete to continuous

Warfare-inspired Manichaeism:

0
Bad guys (they)
Attacker

1
Good guys (we)
System (or country) to be defended

The more subtle case of commercial applications:

0
Undesirable behavior

1
Desirable behavior

• Security often needs incentives
• Incentives usually must be secured
### Book structure (1/2)

#### Upcoming wireless networks

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Small operators, community networks</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Cellular operators in shared spectrum</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mesh networks</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hybrid ad hoc networks</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Self-organized ad hoc networks</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Sensor networks</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>?</td>
<td>X</td>
<td>?</td>
<td></td>
<td>?</td>
</tr>
<tr>
<td>RFID networks</td>
<td>X</td>
<td>?</td>
<td>X</td>
<td>X</td>
<td></td>
<td>?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Part I

- Small operators, community networks
- Cellular operators in shared spectrum
- Mesh networks
- Hybrid ad hoc networks
- Self-organized ad hoc networks
- Vehicular networks
- Sensor networks
- RFID networks

#### Part II

- Naming and addressing
- Security associations
- Securing neighbor discovery
- Privacy
- Enforcing fair MAC
- Enforcing Pkt Fwing greedy op.
- Discouraging greedy op.

#### Part III

- Behavior enforc.
Book structure (2/2)

Security

1. Existing networks
2. Upcoming networks
3. Trust
4. Naming and addressing
5. Security associations
6. Secure neighbor discovery
7. Secure routing
8. Privacy protection

Appendix A: Security and crypto

Cooperation

1. Existing networks
2. Upcoming networks
3. Trust

Appendix B: Game theory

11. Operators in shared spectrum
10. Selfishness in PKT FWing
9. Selfishness at MAC layer

12. Behavior enforcement
Conclusion

- Upcoming wireless networks bring formidable challenges in terms of security and cooperation
- The proper treatment requires a thorough understanding of upcoming wireless networks, of security, and of game theory

Slides available at http://secowinet.epfl.ch
Security and Cooperation in Wireless Networks

Part 2: Thwarting Malicious Behavior

Panos Papadimitratos
panos.papadimitratos@epfl.ch
Outline

- Topics
  - Security Association Establishment
  - Secure Neighbor Discovery
  - Secure Route Discovery
  - Secure Data Communication
  - Privacy Enhancing Technologies for Vehicular Communication Systems
Security Association Establishment
Problem statement

- Establishing secure communication channels between devices

Alice-Bob secure channel

Carol-Dave secure channel
Problem statement (cont’d)

- Security requirements
  - Authentication
  - Integrity
  - Confidentiality
  - Non-repudiation
  - ...

Alice → Bob
Problem statement (cont’d)

• Security mechanisms
  – Message Authentication Codes (MACs)
  – Digital signatures
  – Encryption/decryption
  – Passwords
  – ...

• Cryptography
  – Asymmetric key
  – Symmetric key
Problem statement (cont’d)

• Enable secure communication
  – Uni-directional
  – Bi-directional

• Issues to consider
  – Long- or short- term?
  – What fraction of the system nodes?
  – Is there a trusted third party?
  – ...

Public-key approach

- **Pro:** Any-to-any secure communication
- **Con:** Need to bind public keys to identities

Alice
Identity: A
Public key: $K_A$
Private key: $k_A$

Bob
Identity: B
Public key: $K_B$
Private key: $k_B$
Public-key approach (cont’d)

Alice

(1) \( n_A, A, \text{ text, } \text{Sig}_A(n_A,A,\text{text}) \)

Bob

(2) \( E_A(\text{sec-text},B), \text{Sig}_B(A,n_A,E_A(\text{sec-text},B)) \)

- Secure communication example
  - Message (1): signed with \( k_A \); \( n_A \) is a nonce
  - Message (2): sec-text and B encrypted with \( K_A \); A, \( n_A \), and ciphertext signed with \( k_B \)
  - Note: In practice, different keys are used for signing/verifying and encrypting/decrypting
Public-key approach (cont’d)

- Certification Authority (CA)
  - Trusted Third Party
  - Known $K_{CA}$
  - $\text{Cert}_{CA}(A, K_A)$: CA signature on the identity, public key, and other information (e.g., lifetime)

Alice

Identity: A
Public key: $K_A$
Private key: $k_A$
$\text{Cert}_{CA}(A, K_A)$

Bob

Identity: B
Public key: $K_B$
Private key: $k_B$
$\text{Cert}_{CA}(B, K_B)$
Using a CA

• Largely independent of communication
  – Users obtain certificates over the wire-line network
  – Certificates are installed at wireless devices and the corresponding keys used to secure wireless communication

• Examples specific to wireless networks
Using a CA (cont’d)

- **Wireless local-area (e.g., campus-wide) networks**
  - CA locally administered
  - IEEE 802.11 devices communicate securely with access points

- **Tactical networks**
  - CA operated by the corresponding government department
  - Keys and certificates installed at wireless-enabled devices
  - Hierarchical network organization

- **Vehicular Communication (VC) Systems**
  - More detailed look next
CA example: Vehicular Networks

- Authorities

Swiss Automobile Services

Vaud

Bern

Zurich

Ticino
CA example: Vehicular networks (cont’d)

- **Authorities**
  - Hierarchical organization
  - ‘Forest’ with cross-certification

![](image-url)
Public key cryptography - Practical aspects

- There is no single trusted authority
  - Nodes belonging to different administrative domains will in general be associated and execute security protocols

- PK cryptography is feasible even in low-end mobile platforms, but it is costly
  - Processing
  - Energy consumption
  - Delays
  - Transmission overhead
Symmetric key establishment

• PK cryptography
  – Moderated use recommended
  – Examples:
    • Session keys
    • Shared symmetric key establishment

• Key agreement
  – Both nodes contribute to the shared symmetric key

• Key transport
  – One of the nodes ‘chooses’ the shared symmetric key
Key agreement

- Authenticated Diffie-Hellman protocol
  - $g$ publicly known parameter; $G$ a multiplicative group
  - $A$ selects a random number $r_A$ in $G$; it calculates $t_A = g^{r_A}$
  - $B$ selects a random number $r_B$ in $G$; it calculates $t_B = g^{r_B}$

\[
\begin{align*}
  & (1) \ t_A, \ E_B(A, n_A) \\
  & (2) \ t_B, \ E_A(n_B), \ MAC(K_0, t_A, t_B, B, A) \\
  & (3) \ MAC(K_0, t_A, t_B, A, B)
\end{align*}
\]

\[
K_{AB} = (t_B)^{r_A} = g^{r_Br_A} = (t_A)^{r_B}
\]

H. Krawczyk, “SKEME: A versatile secure key exchange mechanism for Internet,” NDSS’96
Key transport

(1) $n_A$, $B$, $E_B(A, K_{AB})$, $\text{Sig}_A(n_A, B, E_B(A, K_{AB}))$

(2) $n_B$, $A$, $n_A$, $E_A(B, K_{BA})$, $\text{Sig}_B(n_B, A, n_A, E_A(B, K_{BA}))$

(3) $n_B$, $B$, $\text{Sig}_A(n_B, B)$

$K_{AB} = f(K_{AB}, K_{BA})$

X.509 three-pass key transport protocol
Hash chains

• Cryptographic hash or one-way function
  – $h : \{0,1\}^* \rightarrow \{0,1\}^n$
  – Input: Arbitrary length
  – Output: Fixed length $n$

• Required properties
  – **Collision resistance**: it is computationally infeasible to find two distinct inputs, $x$, $y$, which hash to a common value $h(x) = h(y)$
  – **Pre-image resistance**: given a specific hash-value $z$, it is computationally infeasible to find an input $x$ such that $h(x) = z$
  – **2nd pre-image resistance**: given $x$ and $h(x)$ it is computationally infeasible to find a second input $y \neq x$ such that $h(y) = h(x)$
  – **Low computational cost**: given $h$ and an input $x$, $h(x)$ is easy to compute.
Hash Chains (cont’d)

- Pick a random number $r$
- Generate $k$ elements by hashing $r$ successively $k$ times

$$h^k(r) \leftarrow h^{k-1}(r) \leftarrow \cdots \leftarrow h^3(r) \leftarrow h^2(r) = h(h(r)) \leftarrow h(r)$$

$$H_0 \leftarrow H_1 \leftarrow \cdots \leftarrow H_{k-3} \leftarrow H_{k-2} \leftarrow H_{k-1}$$

- $H_0$ is the hash chain anchor
- The remaining $k-1$ elements can be used for authentication
Bootstrapping a hash chain

Alice

- Alice must ‘commit’ to the hash chain anchor
- Each $B_i$ node validates the commitment (signature) and stores $H_0$
- Alice can then utilize the hash chain elements

$\text{Sig}_A(H_0, A, \text{text}), H_0, A, \text{text}, \text{Cert}_{\text{CA}}(K_A, A)$
Using a hash chain

- Chain elements as authenticators, e.g., to transmit “yes” / “no”
  - “Yes” chain
    \[ H_0 \leftarrow H_1 \leftarrow \cdots \leftarrow H_{k-3} \leftarrow H_{k-2} \leftarrow H_{k-1} \]
  - “No” chain
    \[ G_0 \leftarrow G_1 \leftarrow \cdots \leftarrow G_{k-3} \leftarrow G_{k-2} \leftarrow G_{k-1} \]
- Sender: ‘Reveal’ elements in this order
- Use \( G_i \) or \( H_i \) to authenticate a “no” or “yes”
- Receiver: For the \( i \)–th message from Alice, verify that \( h^i(H_i) = H_0 \) or \( h^i(G_i) = G_0 \)

Using a hash chain (cont’d)

- Chain elements as symmetric keys

\[ H_0 \leftarrow H_1 \leftarrow \cdots \leftarrow H_{k-3} \leftarrow H_{k-2} \leftarrow H_{k-1} \]

Time \( T_i \): \( m_i = A, \text{text}_i, \text{MAC}(H_i, A, \text{text}_i) \)

Time \( T_{i+j} \): Release \( H_i \)

- Synchronized clocks at sender and receiver
- Sender release keys (e.g., flooding them across the network) at specific intervals
- *A posteriori* validation at the receiver: reject messages not generated sufficiently close to the release time


A. Perrig et al., "Efficient and secure source authentication for multicast," NDSS ’01
Recap: Public key enabled security

• **Advantages**
  – Any-to-any secure communication
  – Basis for bootstrapping symmetric key primitives

• **Disadvantages**
  – Processing and communication overhead
  – Setting up a certification authority

• **Comment**
  – Methods discussed so far are rather ‘agnostic’ to the underlying network technology
What if no CA is available?

- **Main challenge:** *Man-in-the-Middle* attacks
What if no CA is available? (cont’d)

- Can we leverage on characteristics of the network or the mobile application?
- **Observation 1:** Wireless, mobile devices are used by human beings, who can assist the security association establishment
- **Observation 2:** Wireless communication possible only within a very short range or within a line of sight can imply that no other device is present (**caution!**)
Leveraging on the users

- **Password-based key establishment**
  - $\pi$: shared password
  - $g_\pi = (h(\pi))^2 \mod q$
  - $q$ publicly known parameter
  - $A, B$ select random numbers $x, y$ respectively

\\[ w_A = g_x^\pi \mod q \]
\\[ w_B = g_y^\pi \mod q \]
\\[ z = w_x^B \mod q \]

Abort if $z$ is not in range $[2, q-2]$
Otherwise, $z$ is a large shared secret

Leveraging on the users (cont’d)

- **Password-based key establishment**
  - $h$: hash function
  - Once $z$ is established, A and B prove to each other they know the same $z$
  - A and B can then derive a session key from $z$

\[
o_A = h(04 \parallel w_A \parallel w_B \parallel z \parallel g_\pi)
\]

\[
o_B = h(03 \parallel w_A \parallel w_B \parallel z \parallel g_\pi)
\]

\[
o_3 = h(03 \parallel w_A \parallel w_B \parallel z \parallel g_\pi)
\]

\[
o_4 = h(04 \parallel w_A \parallel w_B \parallel z \parallel g_\pi)
\]

Abort if $o_B \neq o_3$ or $o_A \neq o_4$
Leveraging on the user (cont’d)

• The user verifies that the keys generated at the two devices are identical
• Visual and audible hashes


Leveraging on the wireless link

• ‘Off-line’ local channels
  – One example: infra-red
    • D. Balfanz, D. Smetters, P. Stewart, and H. Wong, “Talking to strangers: Authentication in ad-hoc wireless networks,” NDSS’02
  – Exchange information over the local channel that allows you to authenticate over the wireless radio channel

• Caution: System and protocol design must ensure that it is indeed impossible for the attacker to interfere actively with the communication over the local channel
  – For example, the attacker must be unable to act as an ‘invisible’ relay
Leveraging on the network

• Mobility
  – Users meeting each other, e.g., at a conference, can set up symmetric keys or exchange public keys
    • Chapter 5 and S. Capkun, J-P. Hubaux, and L. Buttyan, “Mobility helps security,” ACM Mobihoc’03
  – More generally, a mobile device can be interested in obtaining public keys of other devices in proximity, e.g., within a few hops
    • Example later in secure routing
  – Point of caution: communication pattern
Summary

• One-to-one, one-to-many, one-to-all, any-to-any secure communication

• Need for protocols that allow dynamic establishment of security associations
  – Public-key cryptography
  – Symmetric-key cryptography
  – Leveraging on the communication and computing environment characteristics

• Various communication patterns
  – Duration, number of communicating devices, direction of communication

• Additional readings
  – Sensor network key distribution (Chapter 5)
Secure Neighbor Discovery
• Node discovery
  – A node discovers other nodes it can *directly* communicate with
Problem statement (cont’d)

- R: nominal communication range
- Caution: A, B are neighbors if and only if they can communicate directly
• B is neighbor of A if and only if it can receive directly from A
• Link (A,B) is up ⇔ B is neighbor of A
• Consider the case with different nominal communication ranges, e.g., $R_A$, $R_B$; then (A,B) may be up while (B,A) is down.
Neighbor discovery

• Neighbor discovery is a building block for other system functionality
  – Communication
  – Access control
  – Physical access control

• Examples
  – First step before routing
  – Connection to an wireless LAN access point
  – Radio Frequency Identification (RFID) reader controlling a door
Neighbor discovery (cont’d)

• Simple, widely used solution, but **not** secure
• Easy to mislead B that A is its neighbor when this is not the case
Attacking neighbor discovery

- Single adversary appears as multiple neighbors

```
B: Neighbor List = {A, C, ..., Z}
```

- “Hello, I’m A”
- “Hello, I’m C”
- “Hello, I’m Z”

- Single adversary appears as multiple neighbors
Securing neighbor discovery

- A first attempt
  - Authenticate “Hello” messages
- The adversary can record signed “Hello” messages and transmit (replay) them later

A: “Hello, I’m A”, Sig\(_A\) (“Hello, I’m A”), Cert\(_{CA}(K_A, A)\)

B: (1) Validate Cert\(_{CA}()\)
(2) Validate Sig\(_A()\)
(3) Add A to neighbor iff (1), (2) are successful
Securing neighbor discovery (cont’d)

(1) $n_B, B$

(2) $A, n_A, n_B, B, \text{Sig}_A(A, n_A, n_B, B), \text{Cert}_{CA}(K_A, A)$

• A second attempt
  – Message authenticity and replay protection
    • $n_A, n_B$ are nonces
  – Bob essentially ‘challenges’ Alice to provide a ‘hello’ message
Attacking neighbor discovery (cont’d)

- “Relay” or “Wormhole” Attack
  - Simply relay any message, without any modification

```
A

"Hello B, I’m A"

M

"Hello B, I’m A"

B: Anyone there?

"B: Anyone there?"

B: Neighbor List = {A}
```
Attacking neighbor discovery (cont’d)

- Long-range relay / wormhole
  - The attacker relays messages across large distances

M1 \rightarrow \text{“Hello, I’m A”} \rightarrow \text{out-of-band or private channel} \rightarrow M2

A \rightarrow \text{“Hello, I’m A”} \rightarrow B

B: Neighbor List = \{A\}
**Attack implications**

- **Network access control**
  - The attacker ‘assists’ access
  - But it has control over the nodes’ communication
Attack implications (cont’d)

- **Routing**
  - The attacker creates a ‘link’ and ‘provides’ shortest routes
  - Attracted traffic is under the control of the adversary
Attack implications (cont’d)

• Physical access control
  – RFID based access control
  – Attacker close to the owner of the access-granting RFID tag; relays signals from and to her accomplice, who obtains access

Z. Kfir and A. Wool, “Picking virtual pockets using relay attacks on contact-less smartcard,” SECURECOMM ’05
Securing neighbor discovery (cont’d)

• A third attempt
  – Geographical packet leashes
    • Nodes are aware of their location in a secure manner
    • Loosely synchronized clocks
    • Sender adds coordinates to each packet
    • Receiver checks if sender is within range
  – Temporal packet leashes
    • Nodes have tightly synchronized clocks
    • Sender (A) adds a timestamp to each packet
    • Receiver (B) estimates its distance from the sender based on the elapsed time, \( t_{\text{prop}} = t_{\text{receiveB}} - t_{\text{sendA}} \)
    • \( \text{Dist}(A,B) < ct_{\text{prop}} \)
      – \( c \) is the speed of light
      – ‘Ignore’ the clock drift

**Observation:** *Physical proximity does not* necessarily imply correct nodes are able to communicate directly

**No** protocol using time-of-flight measurements can distinguish the two situations

Securing neighbor discovery (cont’d)

- Location-aware nodes (securely)
- Estimate neighbor distance in two ways
  - Based on the time-of-flight (ToF)
  - Based on the location information (LOC)
- Compare the two distance estimates

\[
\text{B: } \text{Dist}(\text{LOC}_A, \text{LOC}_B) = \text{Dist}_{\text{Estimate}}(t_{sendA}, t_{receiveB})
\]

B: “Add A to neighbor list”
Securing neighbor discovery (cont’d)

- **Secure Neighbor Discovery**: exchange location information, and compare ToF and LOC based distance estimates

   \[ \text{Dist}(\text{LOC}_A, \text{LOC}_B) < \text{Dist}_{\text{Estimate}}(\text{TS}_A, t_{\text{receive}_B}) \]

   B: “Do NOT add A to neighbor list”

Summary

• Secure Neighbor Discovery
  – Solution for
  – Hard problem; solution is not easy to implement in practice
  – Prerequisite for secure networking protocols and system security
  – Additional reading
    • Other methods, surveyed in [Poturalski-Papadimitratos-Hubaux] report: Using distance bounding, directional antennas, knowledge of topology, properties of the radio signal
    • Centralized visual and statistical wormhole detection (Chapter 6)
Secure Route Discovery
Multi-hop routing

- Wireless multi-hop connectivity
Multi-hop Routing (cont’d)

- (Multi-hop) Connectivity graph
Multi-hop routing (cont’d)

- Stage 0: neighbor discovery
- Stage 1: route discovery

Route: Sequence of nodes (and edges); for simplicity: (A, G, E)
Multi-hop routing (cont’d)

- Explicit route discovery
  - Fully, clearly expressed and readily observable route returned by the routing protocol

Route to E: (A, C, F, E)

Route to A: (E, F, C, A) (Possibly)
Multi-hop routing (cont’d)

- Implicit route discovery
  - Distributed computation that returns a tuple of the form (current node, relay node, destination node)
Multi-hop routing (cont’d)

• Basic route discovery
  – Explicit or implicit, providing only the structure of the route

• Augmented route discovery
  – Need a function that assigns labels to links, denoted as link metrics
    • For a link \((V_1, V_2)\), metric \(m_{1,2}\)
  – Route metric: a function that is the aggregate of the route link metrics
    • For a route \((V_0, V_1, \ldots, V_n)\), route metric \(g(m_{0,1}, m_{1,2}, \ldots, m_{n-1,n})\)
Multi-hop routing (cont’d)

Input

\( S, T \in N \)

(Secure) Routing Protocol

Output

An \((S,T)\)-route and

(i) Explicit:

\[ m_{0,1}, m_{1,2}, \ldots, m_{n-1,n} \]

(ii) Implicit:

\[ g(m_{0,1}, m_{1,2}, \ldots, m_{n-1,n}) \]

• Input: source, \( S \), and destination, \( T \), nodes

• Output: an \((S-T)\)-route (of \( n \) links) and
  – The link labels (metrics) or
  – The route metric
Multi-hop routing (cont’d)

- Example of route discovery
  - Reactive routing protocol
Attacking route discovery (cont’d)

- Impersonation of the destination, for example, in any reactive routing protocol

**RREQ: “A is looking for H”**

**RREP: “I am H”**
• Modification of the route links, for example, in DSR
Attacking route discovery (cont’d)

• Abuse of the routing caching mechanism, for example, in DSR

Route Cache:
...
\{A, D, Y\}
\{A, D, Y, H\}
...

RREP:
Route = \{X, D, Y, H\}
Attacking route discovery

- Disrupting a link state routing protocol, for example, in OLSR
Attacking route discovery (cont’d)

- Disrupting distance vector routing, for example, in AODV
Attacking route discovery (cont’d)

- Disrupting distance vector routing, for example, in DSDV
Attacking route discovery (cont’d)

• Caution: none of the above-mentioned protocols (DSR, AODV, DSDV, OLSR) was designed with security in mind

• Many possible ways to attack the route discovery

• Outcome of attacks
  – Control communication
    • Become part of utilized routes
    • Monopolize resources
  – Disrupt communication
    • Degrade or deny
Secure route discovery requirements

• What do we need a secure routing protocol to do?

• Network model
  – Capture the system characteristics
    • For example, dynamically changing topology

• Specification
  – Define the properties of any candidate secure routing protocol independently of its functionality
Requirements

• We are interested in protocols that discover routes with the following two properties:

1. **Loop-freedom**: an (S,T)-route is loop-free when it has no repetitions of nodes.

2. **Freshness**: an (S,T)-route is fresh with respect to a \((t_1,t_2)\) interval if each of the route’s constituent links is up at some point during the \((t_1,t_2)\).

• Loop-freedom and freshness are relevant for both explicit and implicit route discovery, and both basic and augmented protocols.

Secure Routing Protocol (SRP)

- Explicit basic route discovery
- Observation
  - It is hard to ‘know’ all nodes in the network, i.e., establish associations with all of them
  - Often infeasible and very costly
  - Especially in ‘open’ networks
- SRP assumptions
  - Secure neighbor discovery
  - Hop-by-hop authentication of all control traffic
  - End nodes (source, destination) ‘know’ each other
    - Can set up security associations

Route Request (RREQ):
\[ S, T, Q_{SEQ}, Q_{ID}, MAC(K_{S,T}, S, T, Q_{SEQ}, Q_{ID}) \]

1. \( S \) broadcasts \( RREQ \);
2. \( V_1 \) broadcasts \( RREQ, \{ V_1 \} \);
3. \( V_2 \) broadcasts \( RREQ, \{ V_1, V_2 \} \);
4. \( V_3 \) broadcasts \( RREQ, \{ V_1, V_2, V_3 \} \);
SRP (cont’d)

Route Reply (RREP):

\[ Q_{ID}, \{ T, V_3, V_2, V_1, S \}, \]
\[ MAC(K_{S,T}, Q_{ID}, Q_{SEQ}, T, V_3, V_2, V_1, S) \]

5. \( T \rightarrow V_3 : RREP \);
6. \( V_3 \rightarrow V_2 : RREP \);
7. \( V_2 \rightarrow V_1 : RREP \);
8. \( V_1 \rightarrow S : RREP \);
SRP (cont’d)

• Route requests verifiably reach destination
  – Intermediate node replies disabled
  – Aggressive caching of routing information disabled
• Route replies must trace back the paths traversed by route requests
• Intermediate nodes are not authenticated at the end nodes
• Dual route request identifier
  – $Q_{ID}$: random, used by the intermediate nodes
  – $Q_{SEQ}$: sequence number, used by the destination
  – The adversary cannot launch a “sequence number” attack
SRP (cont’d)

• Crucial to operate on top a secure neighbor discovery protocol

• Neighbor Lookup Protocol (NLP)
  – Secure neighbor discovery
  – Establish security associations between neighbors
  – Identify control traffic injected by each neighbor
  – Prevent attacks that misuse network addresses
    • IP spoofing
    • Use of multiple identities
    • MAC spoofing
  – DoS protection

• Efficient mechanisms to discard spurious/corrupted traffic at intermediate nodes
  – Replies relayed only if neighbors had previously forwarded the corresponding request
SRP (cont’d)

• Routes discovered by SRP in the presence of independent adversaries are fresh
  – $t_1$ is the point in time at which S transmitted a RREQ for T, and $t_2$ is the point at which S received the corresponding RREP

• In the presence of colluding adversaries SRP discovers ‘weakly fresh’ routes
  – A sequence of links, in general different than those in the discovered route were up at some point in $(t_1, t_2)$
Secure Link State Protocol (SLSP)

- **Secure Neighbor Discovery**
  - Correct nodes discover only actual neighbors
- **Periodic Link State Update (LSU) advertisements**
  - Nodes distribute their discovered neighbors within an extended neighborhood, the *zone*
  - LSUs are signed
- **Link state accepted *iff* reported by both incident nodes**
- **Nodes distribute their public key throughout the zone**
- **SLSP can adjust its scope with different zone radii**

P. Papadimitratos and Z.J. Haas, "Secure Link State Routing for Mobile Ad Hoc Networks," WSAAN’03
SLSP (cont’d)

- SLSP can adjust its scope, with different zone radii
- It can operate locally, combined with another global route discovery, or network-wide
• Keep the LSU propagation within the zone
  – Use a hash chain mechanism
  – \( \text{zone\_radius} = X_R = h^R(x_0) \)
  – \( \text{hops\_traversed} = X_1 = h(x_0), \ T TL = R-1 \)
  – After \( i \) hops (\( i=R-T TL \)), relay packet if:
    • \( i < R \), and
    • \( h^{R-i}(\text{hops\_traversed}) == \text{zone\_radius} \)
  – \( \text{hops\_traversed} = H(\text{hops\_traversed}) \)

• Same idea can be applied in reactive routing, to perform an expanding ring search
Authenticating intermediate nodes

- Source knows all nodes in the network
- All nodes know any source and destination node (especially in the case of reactive protocols)
- Overall, all nodes know all nodes, or equivalently have security associations established before any route discovery
- Hard to achieve, yet what if? For example, in small or closed networks
Ariadne

- Secures DSR, adding authentication of RREQ and RREP messages by each intermediate node that relays and modifies them
- All-to-all security associations
- Use of different cryptographic primitives
  - Signatures, Message Authentication Codes, and TESLA

Ariadne (cont’d)

- Operation across a route \((S, F_1, F_2, D)\) with MACs
- If TESLA is used, the delayed authentication (for key disclosure) becomes part of the route discovery delay

| \(S\) | \(h_S = MAC_{SD}(rreq, S, D, id)\) |
| \(S \rightarrow \ast\) | \((rreq, S, D, id, h_S, [\ ], [\ ])\) |
| \(F_1\) | \(h_{F_1} = H(F_1, h_S)\) |
| \(F_1 \rightarrow \ast\) | \((rreq, S, D, id, h_{F_1}, [F_1], [mac_{F_1}] )\) |
| \(F_2\) | \(h_{F_2} = H(F_2, h_{F_1})\) |
| \(F_2 \rightarrow \ast\) | \((rreq, S, D, id, h_{F_2}, [F_1, F_2], [mac_{F_1}, mac_{F_2}] )\) |
| \(D \rightarrow F_2\) | \((rrep, D, S, [F_1, F_2], mac_D)\) |
| \(F_2 \rightarrow F_1\) | \((rrep, D, S, [F_1, F_2], mac_D)\) |
| \(F_1 \rightarrow S\) | \((rrep, D, S, [F_1, F_2], mac_D)\) |

Protocol operation as in Fig. 7.6 (p.202) of SeCoWiNet book
EndairA

- All-to-all security associations, digital signatures
- **Novelty**: intermediate nodes sign only the RREP
- Withstands provably attacks and reduces overhead with respect to Ariadne

\[
\begin{align*}
  S \rightarrow * & : (rreq, S, D, id, []) \\
  F_1 \rightarrow * & : (rreq, S, D, id, [F_1]) \\
  F_2 \rightarrow * & : (rreq, S, D, id, [F_1, F_2]) \\
  D \rightarrow F_2 & : (rrep, S, D, id, [F_1, F_2], \{\text{sig}_D\}) \\
  F_2 \rightarrow F_1 & : (rrep, S, D, id, [F_1, F_2], \{\text{sig}_D, \text{sig}_{F_2}\}) \\
  F_1 \rightarrow S & : (rrep, S, D, id, [F_1, F_2], \{\text{sig}_D, \text{sig}_{F_2}, \text{sig}_{F_1}\})
\end{align*}
\]

Protocol operation as in Fig. 7.8 (p.206) of SeCoWiNet book

Augmented Discovery: Requirement

- Let $l_{i,i+1} \in M$ be the actual link metric for each link of a discovered $(S,T)$-route and $g(l_{0,1}, ..., l_{n-1,n})$ the actual route metric.
- The metric estimated (by a protocol) for link $(V_i, V_{i+1})$ is $m_{i,i+1}$.

(3) **Accuracy:** an $(S,T)$-route is accurate with respect to a route metric $g$ and a constant $\Delta_{good} \geq 0$ if:

$$| g(m_{0,1}, ..., m_{n-1,n}) - g(l_{0,1}, ..., l_{n-1,n}) | < \Delta_{good}$$

- Accuracy is relevant only to augmented, explicit or implicit, route discovery.
Quality-of-Service Aware Discovery

- QoS-SRP: Secure QoS-aware routing
- Nodes estimate metrics for their incident links
  - For link \((V_i, V_{i+1})\), node \(V_i\) calculates \(m_{i,i+1}^i\) and \(V_{i+1}\) calculates \(m_{i,i+1}^{i+1}\)
  - For some \(\varepsilon > 0\), \(|m_{i,i+1}^i - m_{i,i+1}^{i+1}| < \varepsilon\)
  - \(\varepsilon\) is a protocol-selectable and metric-specific threshold that allows for metric calculation inaccuracies
- \(\tilde{\delta} \geq 0\) is the maximum metric calculation error by a correct node

P. Papadimitratos and Z.J. Haas, "Secure Route Discovery for QoS-Aware Routing in Ad Hoc Networks," Sarnoff '05
**QoS-SRP**

**Route Request (RREQ):**

\[ S, T, Q_{SEQ}, Q_{ID}, MAC(K_{S,T}, S, T, Q_{SEQ}, Q_{ID}) \]

1. **S** broadcasts **RREQ**;
2. **V₁** broadcasts **RREQ**, \{ \(V₁\), \{ \(m^1_{s,1}\) \} \};
3. **V₂** broadcasts **RREQ**, \{ \(V₁, V₂\), \{ \(m^1_{s,1}, m^2_{1,2}\) \} \};
4. **V₃** broadcasts **RREQ**, \{ \(V₁, V₂, V₃\), \{ \(m^1_{s,1}, m^2_{1,2}, m^3_{2,3}\) \} \};
QoS-SRP (cont’d)

Route Reply (RREP):

1. \( T \rightarrow V_3 \) : RREP;
2. \( V_3 \rightarrow V_2 \) : RREP;
3. \( V_2 \rightarrow V_1 \) : RREP;
4. \( V_1 \rightarrow S \) : RREP;
QoS-SRP (cont’d)

- Metric types
  - \( \Delta_{\text{good}}^{\text{add}}, \ g_{\text{add}} \left( m_{0,1}^1, \ldots, m_{n-1,n}^n \right) = \sum_{i=0}^{n-1} m_{i,i+1}^{i+1} \)
  - If \( m_{i,i+1}^{i+1} > 0 \), \( g \left( m_{0,1}^1, \ldots, m_{n-1,n}^n \right) = \prod_{i=0}^{n-1} m_{i,i+1}^{i+1} \)

  can be written as \( g_{\text{add}} \left( \bar{m}_{0,1}^1, \ldots, \bar{m}_{n-1,n}^n \right) \)

  where \( \bar{m}_{i,i+1}^{i+1} = \log(m_{i,i+1}^{i+1}) \), for \( 0 \leq i \leq n - 1 \)

- \( \Delta_{\text{good}}^{\text{max}}, \ g_{\text{max}} \left( m_{0,1}^1, \ldots, m_{n-1,n}^n \right) = \max_{0 \leq i \leq n-1} \left\{ m_{i,i+1}^{i+1} \right\} \)
- \( \Delta_{\text{good}}^{\text{min}}, \ g_{\text{min}} \left( m_{0,1}^1, \ldots, m_{n-1,n}^n \right) = \min_{0 \leq i \leq n-1} \left\{ m_{i,i+1}^{i+1} \right\} \)
QoS-SRP (cont’d)

• Routes discovered by SRP in the presence of independent adversaries are accurate, with respect to (i) $g_{\text{add}}$ and $\Delta_{\text{good}}^{\text{add}} = n^2 \varepsilon + n \tilde{\delta}$ (ii) $g_{\text{max}}$ and $\Delta_{\text{good}}^{\text{max}} = n \varepsilon + \tilde{\delta}$, and (iii) $g_{\text{min}}$ and $\Delta_{\text{good}}^{\text{min}} = n \varepsilon + \delta$, with $n$ the number of route links, $\varepsilon > 0$ the maximum allowable difference between $m_{i,i+1}^i$ and $m_{i,i+1}^{i+1}$, and $\tilde{\delta} \geq 0$ the maximum error for a metric calculation by a correct node.
Attacking route discovery (cont’d)

- Adversary acting as a relay, ‘creating’ Byzantine links
- Secure neighbor discovery and hop-by-hop authentication can defeat this attack
Multiple Colluding Attackers

- $M_1$ and $M_3$ are seemingly correct to their neighbors, but they ‘omit’ protocol functionality when handling packets from $M_2$.
- Example: $M_2$ relays RREQ and RREP packets without appearing in the route discovery.
Attacking Routing - Revisited

- **Tunneling Attack**
  - Two colluding attackers: $M_1$, $M_2$
  - $M_1$ encapsulates control traffic and forwards to $M_2$ and vice versa
  - Attackers seemingly follow the protocol with respect to their neighbors

Summary

• Route discovery is vulnerable
• Secure route discovery specification
  – Loop freedom
  – Freshness
  – Accuracy
• Protocols relying on different trust assumptions
• Securing basic and augmented route discovery in open, dynamic networks
• Colluding adversarial nodes can subvert any route discovery protocol; ‘tunneling attack’
• Additional reading
  – Chapter 7: More secure routing protocols, including sensor network protocols
Secure Data Communication
Problem statement

• **Goal:**
  – Reliable and low-delay data delivery in the presence of attackers that disrupt the data communication

• **Solution:**
  – Detect and avoid compromised and failing routes
  – Tolerate malicious and benign faults
    • In general, hard to distinguish in highly dynamic networking environments
Data Communication

Route to E

Route to A
Data Communication (cont’d)

Message for E
Data Communication (cont’d)
Data Communication (cont’d)

• How can an attacker be part of a route?
  – Make the route appear ‘preferable’ (shorter in hops, delay, or any other metric)
  – Other routing protocol-specific attacks (e.g., ‘rushing’)
  – Do nothing that disrupts the secure route discovery

• Consider
  – An ideal secure routing protocol, ensuring loop-free, fresh, and accurate routes against any possible attack
  – All nodes on the discovered route authenticated

• Still, the attacker can deny communication, dropping packets

• Worse even, the attacker can choose to hit when it hurts the most
Data Communication (cont’d)

• What is the impact of the adversary that ‘lies low’ and disrupts only the data communication?

![Graph showing the relationship between attacker strength and reliability.](image)

- 50% of the network nodes attacking
- 35% message delivery

Reliability

Attacker Strength
Securing Data Communication

- Use multiple routes
Securing Data Communication (cont’d)

- Disperse data

\[
\text{Original message} = \begin{cases} 
1 \\
2 \\
\vdots \\
m-1 \\
m \\
n-2 \\
n-3 \\
n
\end{cases}
\]

Introduce redundancy to the original message

Introduce redundancy to the original message
Securing Data Communication (cont’d)

- Disperse data

Reconstruct message if any \textit{m-out-of-n} pieces are intact
Securing Data Communication (cont’d)

• Transmit simultaneously across the routes

Sending $n=3$
E needs $m=2$

Received $m$ pieces!
Securing Data Communication (cont’d)

- Get feedback

Tell A which pieces were intact
Securing Data Communication (cont’d)

- Reliable and Real-Time Communication in Hostile Environments
  
  - Secure Routing Only
  - Secure Routing + Secure Data Communication

![Graph showing reliability vs. attacker strength]

- 93% message delivery without retransmissions
- 35% message delivery
- 50% of the network nodes are attacking
Securing Data Communication (cont’d)

Bandwidth For Security

Redundancy

Average delay for 100% message delivery

Message delivery without retransmissions

Delay

Reliability

1.2 s

0.4 s

1

Redundancy

3.5

1

Redundancy

3.5

82%

93%

Average delay for 100% message delivery

Message delivery without retransmissions
Securing Data Communication (cont’d)

- Secure Message Transmission (SMT) protocol
  - Dispersion of the transmitted data
  - Simultaneous usage of multiple node-disjoint routes
  - Data integrity and origin authentication
  - End-to-end secure and robust feedback
  - Adaptation to the network conditions

- Secure Single Path (SSP) protocol
  - Discovery and utilization of a single route
  - End-to-end security and feedback
SMT Operation

• The Active Path Set (APS)
  – Maintain a (partial) view of the network topology
  – Construct a set of node disjoint routes (per destination)
  – Routes remain in the APS until deemed non-operational

• Multi-path operation
  – Select the APS routes to transmit a dispersed message
  – Route selection attributes
    • Path rating
    • Probability of path survival
    • Overall probability of successful message delivery
  – Assign each message piece to one of the selected routes
SMT Operation (cont’d)

• Example: Transmission of a single message

Source

Destination

Dispersed Message

Dispersed ACK

ACK

timer

Re-transmit

Time
SMT Operation (cont’d)

• Secure and robust end-to-end feedback
  – Dispersed and returned over multiple routes
  – Informs on the successfully received pieces
  – Allows the correlation of successfully received pieces with data routes
  – Provides “safe” information for the adaptation of the protocol operation
SMT Operation (cont’d)

• Adapt to the network conditions
  – Detect non-operational routes
  – Switch to alternate (new) routes
  – Adapt the protocol configuration
    • Number of routes
    • Transmission redundancy
    • Route selection
    • Additional route discovery
• Path rating mechanism
  – Each route is associated to a rating $r_s \in [r_s^{thr}, r_s^{max}]$
  – Update $r_s$ for each transmission across the route
  – For each delivered piece, $r_s$ is increased by a constant $\beta$
  – For each lost piece, $r_s$ is decreased by a constant $\alpha$
  – The route is discarded when its rating reaches $r_s^{thr}$

$$r_s(i) = \begin{cases} 
  \max \{r_s(i-1) - \alpha, r_s^{thr}\}, & \text{if a piece is lost} \\
  \min \{r_s(i-1) + \beta, r_s^{max}\}, & \text{if a piece is received} 
\end{cases}$$
SMT Operation (cont’d)

- Robustness to arbitrary attack patterns
  - Bounded fraction of data the adversary can drop (Bandwidth Loss (BWL)) before the compromised route is detected
    \[ BWL \leq \frac{\beta}{\alpha + \beta} \]
  - Non-operational routes are promptly discarded
  - Route re-instatement after transient data loss

SMT Operation (cont’d)

• What is the appropriate choice for $\alpha, \beta$?
  – The attack pattern is not known in advance
  – The faster a non-operational route is discarded the better
  – Not discarding a route after a transient packet loss is preferable

• One criterion
  – Min-Max Regret

SMT Operation (cont’d)

- Selection of $\alpha, \beta$
SMT Operation (cont’d)

- Selection of $\alpha, \beta$
SMT Operation (cont’d)

- Is the route rating sufficient to maintain reliable communication? What about mobility?
- The higher the route age is, the more likely it is to break
- \( t \) : current route age of the i-th route in APS
- \( p_i(t) \) : probability of survival of the route during a piece transmission (delay \( d \))
- Estimate this from route lifetime samples (periods of time from the discovery till the route removal from the APS)

\[
\hat{p}_i(t) = \begin{cases} 
\frac{S - 1}{S}, & \text{if } t + d < \tau_1 \\
\frac{S - j}{S}, & \text{for } j \text{ such that: } \tau_j \leq t + d < \tau_{j+1} \\
\frac{1}{S}, & \text{if } t + d > \tau_D
\end{cases}
\]
SMT Operation (cont’d)

- Example of the estimated probability of path survival, based on collected data
- FA: Fraction of adversaries present in the network
SMT Operation (cont’d)

• Determine the appropriate message dispersion
  – To achieve the sought end-to-end reliability, \( P_{GOAL} \) while minimizing
    • The transmission redundancy: \( P_{GOAL} - r_{\min} \)
    • The number of utilized paths: \( P_{GOAL} - N_{\min} \)
  – To achieve a redundancy goal while maximizing the end-to-end reliability: \( r_{GOAL} \)
## Performance Evaluation

<table>
<thead>
<tr>
<th><strong>Nodes</strong></th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fraction of Adversaries</strong></td>
<td>10%, 20%, 30%, 40%, or 50% of the network</td>
</tr>
<tr>
<td><strong>Measurements</strong></td>
<td>50 randomly seeded runs for each point</td>
</tr>
<tr>
<td><strong>Security Bindings</strong></td>
<td>Single destination per source</td>
</tr>
<tr>
<td><strong>Simulated time</strong></td>
<td>300 sec</td>
</tr>
<tr>
<td><strong>Mobility</strong></td>
<td>Random waypoint; Pause times: 0, 20, 40, 60, 100, 150, 200, 250 seconds</td>
</tr>
<tr>
<td><strong>Load</strong></td>
<td>3, 7, 15, 20 CBR flows, Data payload: 512 Bytes Rates: 4, 10, 15, 20, 25, and 30 packets/sec</td>
</tr>
<tr>
<td><strong>Coverage Area</strong></td>
<td>1000m-by-1000m</td>
</tr>
<tr>
<td><strong>PHY/MAC</strong></td>
<td>802.11, DCF, 2 and 5.5 Mbps, 300m</td>
</tr>
<tr>
<td><strong>Tool</strong></td>
<td>OPNET</td>
</tr>
</tbody>
</table>
Performance Evaluation (cont’d)

- Secure Message Transmission (SMT) protocol
- Secure Single Path (SSP) protocol
- Secure route discovery for both protocols
  - Explicit, basic
    - Reactive, Proactive
    - SRP, SLSP
- Attack pattern
  - Full compliance with the route discovery
  - Discarding in–transit data packets
Performance Evaluation (cont’d)

SMT-LS: SMT with a Link State Protocol

Message Delivery Fraction

Message Delay

- $\text{Retry}_{\text{MAX}} = 0$
- $\text{Retry}_{\text{MAX}} = 3$
- $\text{Retry}_{\text{MAX}} = 100$
Performance Evaluation (cont’d)

Message Delivery Fraction

SMT-RRD: SMT with SRP

Message Delay

\[ \text{MDF} \]

\[ \text{Delay} \text{(sec)} \]
Performance Evaluation (cont’d)

Routing Overhead  
Transmission Overhead

SMT-RRD: SMT with SRP
Performance Evaluation (cont’d)

SSP-RRD: SSP with SRP

Message Delivery Fraction

Message Delay

Retry_{MAX} = 0
Retry_{MAX} = 3
Retry_{MAX} = 100
Performance Evaluation (cont’d)

Routing Overhead

Transmission Overhead

SSP-RRD: SSP with SRP
Performance Evaluation (cont’d)

Transmission Redundancy

Average number of sent pieces (N)  Average number of required pieces (M)
Performance Evaluation (cont’d)

Impact of mobility; SMT-RRD

Message Delivery Fraction

Message Delay

Impact of mobility; SMT-RRD
Throughput – no flow control

Throughput - SMT-RRD with TCP

Impact of Load and interaction with TCP
Performance Evaluation (cont’d)

Impact of Load and SMT interaction with TCP

Message delay – no flow control

Message delay - SMT-RRD with TCP

Impact of Load and SMT interaction with TCP
Summary

• Secure data communication is critical
  – Secure routing protocols are vulnerable
  – As long as attackers can place themselves on utilized routes, they can degrade or deny communication
  – The only answer is to assess whether data are delivered, and avoid non-operational routes

• Secure data communication is practical
  – Low-delay, low-jitter, and highly reliable; essentially, real-time
  – Flexible
  – Low overhead
  – End-to-end
  – Effective against any data-dropping pattern
Privacy Enhancing Technologies for Vehicular Communication (VC) Systems
VC System (cont’d)

- High rate broadcast communication
- VANET-only (e.g., safety) and TCP/IP communication
Security for VC

- Focus: Communication
- Main objectives
  - Identity and Cryptographic Key Management
  - Privacy Enhancing Technologies (PET)
  - Secure Communication
- Requirements
  - Authentication, integrity, non-repudiation, access control, confidentiality, availability
  - Privacy
  - Liability identification

Secure VC

• Authorities
  – Trusted entities
  – Issuing and managing identities and credentials

• Network nodes
  – Vehicles
    • Public
    • Private
  – Road-side units

• Users
Secure VC (cont’d)

- System Illustration
Secure VC (cont’d)

Abstract view of a vehicle in a (secure) vehicular communications system
Secure VC (cont’d)

• Node Identity
  – Unique identity \( V \)
  – Integration of pre-VC and VC-specific identifiers

• Node Keys
  – Public / private key pair \( K_V, k_V \)

• Node Credentials
  – Certificate \( Cert_{\chi} \{ K_V, A_V \} \)
  – \( A_V \): attributes of node \( V \)

• Long-term identification
Secure VC (cont’d)

- Secure Communication
  - Single- and Multi-hop
  - Vehicle to vehicle
  - Vehicle to infrastructure

- Digital signatures more appropriate tool
  - Any-to-any communication; e.g., broadcast, geo-cast
  - High mobility

- Relatively simple networking protocols ‘shift’ the security focus to the application
Secure VC (cont’d)

- Secure Communication (cont’d)

<table>
<thead>
<tr>
<th>Payload</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location: ((x_V, y_V, z_V))</td>
</tr>
<tr>
<td>Signature with (k_V)</td>
</tr>
</tbody>
</table>

\[ Cert_X\{K_V A_V\} \]

Warning: Accident at \((x, y, z)\)

Vehicle V

Vehicle U
Problem statement

• Frequent, high-rate vehicle-to-vehicle and vehicle-to-infrastructure communication
  – Periodic, triggered, dependent on network characteristics (e.g., density)
  – Example: a vehicle transmits safety messages every 100 to 300 milliseconds
  – Safety messages include vehicle-specific information; e.g., its coordinates

• Communication cannot be regulated or controlled by the node/user
  – Safety messaging will be essentially an ‘always-on’ application
Problem statement (cont’d)

• Vehicle-originating wireless transmissions are particularly easy to eavesdrop
  – Data link very similar to a widely adopted technology: IEEE 802.11p
  – Very large and increasing numbers of 802.11 access points already deployed
  – Road-side infrastructure deployed for other services could be subverted into acting as an eavesdropper

• Linking messages to the transmitting vehicle and inferring private information about its passengers
Problem statement (cont’d)

1. A at location \((x_1, y_1, z_1)\) at time \(t_1\)
   - A communicates with B
   - Eavesdropper

2. A refuels at time \(t_2\) and location \((x_2, y_2, z_2)\)

3. A enters the parking lot at time \(t_3\)
   - A downloads data from server X
Problem statement (cont’d)

• What are we after?
  – At least the same degree of privacy achieved nowadays, before the advent of vehicular communications
  – Combination of strong security and privacy-enhancing technologies
  – Ideally, anonymous and authentic communications, but:
    • High processing and communication overhead
    • Often, messages from the same vehicle should be linkable
  – Requirement: messages generated by a given vehicle can be linked at most over a protocol-selectable period of time
    • The shorter this period, the harder to track a vehicle becomes
PET for VC (cont’d)

- **Pseudonym**: Remove all identifying information from certificate
- Equip vehicles with multiple pseudonyms
  - Alternate among pseudonyms over time (and space)
  - Sign message with the private key corresponding to pseudonym
  - Append current pseudonym to signed message
PET for VC (cont’d)

- PET system setup

Authority X  
Long-term Identification

Vehicle V

Authority A
Pseudonym Provider

PSNYM_1, …, PSNYM_k
Set of pseudonyms for V
PET for VC (cont’d)

- PET system setup (cont’d)
- Multiple pseudonym providers

Organization 1  Organization 2  ...  Organization n
V-PNYM-1  V-PNYM-2  ...  V-PNYM-n

Vehicle V
PET for VC (cont’d)

- **Pseudonym format**

<table>
<thead>
<tr>
<th>PSNYM-Provider ID</th>
<th>PSNYM Lifetime</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Public Key $K_i$</td>
</tr>
<tr>
<td></td>
<td>PSNYM-Provider Signature</td>
</tr>
</tbody>
</table>

- **Supplying vehicles with pseudonyms**
  - Sufficient in number
  - Periodic ‘refills’
Secure VC Building Blocks (cont’d)

- Pseudonym Change Mechanism

**Inputs:**
- Vehicle Location
- Vehicle Clock
- Recipient(s) / (Verifier(s))

**Output:**
Use PSNYM_i for period $[t_i, t_{i+1}]$

- One pseudonym per day (?)
- One per transaction (?)
PET for VC (cont’d)

- Other vehicle network identifiers: e.g., IP and MAC addresses
- Change addresses along with pseudonyms
- Maintain addresses only when necessary, but encapsulate
PET for VC (cont’d)

- Credentials Management

  ‘Re-filling’ with or obtaining new credentials

  ‘Re-filling’ with or obtaining new credentials
PET for VC (cont’d)

• Pseudonym Resolution

Pseudonymous Communication Transcript

“Vehicle V generated the transcript”
PET for VC (cont’d)

• Challenge
  – Managing a pseudonymous authentication system is cumbersome
    • Preload large numbers of pseudonyms or obtain them on-the-fly
    • Costly computations at the side of the pseudonym provider
    • Costly wireless communication to obtain pseudonyms
    • Need reliable access to the pseudonym provider

• Solution
  – On-board generation of pseudonyms
Summary

• Security and privacy-enhancing mechanisms are a prerequisite for the VC systems deployment

• Securing VC systems is a complex yet ‘real’ problem that attracts the attention of the community

• Opportunity: Awareness and joint efforts in industry and academia
Acknowledgements

• Thanks to Marcin Poturalski and George Theodorakopoulos for their feedback on Part 2 of this tutorial
Questions?

panos.papadimitratos@epfl.ch
Appendix B: A tutorial on game theory for wireless networks

- static games;
- dynamic games;
- repeated games;
- strict and weak dominance;
- Nash equilibrium;
- Pareto optimality;
- Subgame perfection;
...
Chapter outline

B.1 Introduction
B.2 Static games
B.3 Dynamic games
B.4 Repeated games
Brief introduction to Game Theory

- Discipline aiming at modeling situations in which actors have to make decisions which have mutual, **possibly conflicting**, consequences
- Classical applications: **economics**, but also politics and biology
- Example: should a company invest in a new plant, or enter a new market, considering that the **competition may** make similar moves?
- Most widespread kind of game: **non-cooperative** (meaning that the players do not attempt to find an agreement about their possible moves)
Classification of games

<table>
<thead>
<tr>
<th>Non-cooperative</th>
<th>Cooperative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>Dynamic (repeated)</td>
</tr>
<tr>
<td>Strategic-form</td>
<td>Extensive-form</td>
</tr>
<tr>
<td>Perfect information</td>
<td>Imperfect information</td>
</tr>
<tr>
<td>Complete information</td>
<td>Incomplete information</td>
</tr>
</tbody>
</table>

Perfect info: each player knows the identity of other players and, for each of them, the payoff resulting of each strategy.

Complete info: each player can observe the action of each other player.
Cooperation in self-organized wireless networks

Usually, the devices are assumed to be cooperative. But what if they are not?
Chapter outline

B.1 Introduction
B.2 Static games
B.3 Dynamic games
B.4 Repeated games
Example 1: The Forwarder’s Dilemma
From a problem to a game

- users controlling the devices are *rational* = try to maximize their benefit
- game formulation: \( G = (P, S, U) \)
  - \( P \): set of players
  - \( S \): set of strategy functions
  - \( U \): set of utility functions

- **strategic-form** representation
  - Reward for packet reaching the destination: 1
  - Cost of packet forwarding: \( c \) (\( 0 < c << 1 \))

<table>
<thead>
<tr>
<th></th>
<th>Forward</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forward</td>
<td>((1-c, 1-c))</td>
<td>((-c, 1))</td>
</tr>
<tr>
<td>Drop</td>
<td>((1, -c))</td>
<td>((0, 0))</td>
</tr>
</tbody>
</table>
Solving the Forwarder’s Dilemma (1/2)

**Strict dominance**: strictly best strategy, for any strategy of the other player(s)

Strategy $s_i$ strictly dominates if

$$u_i(s_i', s_{-i}) < u_i(s_i, s_{-i}), \forall s_{-i} \in S_{-i}, \forall s_i' \in S_i$$

where: $u_i \in U$ utility function of player $i$

$s_{-i} \in S_{-i}$ strategies of all players except player $i$

In Example 1, strategy Drop *strictly dominates* strategy Forward

<table>
<thead>
<tr>
<th></th>
<th>Forward</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forward</td>
<td>$(1-c, 1-c)$</td>
<td>$(-c, 1)$</td>
</tr>
<tr>
<td>Drop</td>
<td>$(1, -c)$</td>
<td>$(0, 0)$</td>
</tr>
</tbody>
</table>
Solving the Forwarder’s Dilemma (2/2)

Solution by iterative strict dominance:

<table>
<thead>
<tr>
<th></th>
<th>Forward</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blue</td>
<td>(1-c, 1-c)</td>
<td>(1, -c)</td>
</tr>
<tr>
<td>Green</td>
<td>(-c, 1)</td>
<td>(0, 0)</td>
</tr>
</tbody>
</table>

Result: Tragedy of the commons! (Hardin, 1968)

Drop *strictly dominates* Forward

BUT

Forward would result in a *better outcome*
Example 2: The Joint Packet Forwarding Game

- Reward for packet reaching the destination: 1
- Cost of packet forwarding: \( c \) (\( 0 < c << 1 \))

<table>
<thead>
<tr>
<th></th>
<th>Blue Forward</th>
<th>Blue Drop</th>
<th>Green Forward</th>
<th>Green Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>((1-c, 1-c))</td>
<td>((0, 0))</td>
<td>((-c, 0))</td>
<td>((0, 0))</td>
</tr>
</tbody>
</table>

No strictly dominated strategies!
Weak dominance

Weak dominance: strictly better strategy for at least one opponent strategy

Strategy $s'_i$ is weakly dominated by strategy $s_i$ if

$$u_i(s'_i, s_{-i}) \leq u_i(s_i, s_{-i}), \forall s_{-i} \in S_{-i}$$

with strict inequality for at least one $s_{-i}$

Iterative weak dominance

The result of the iterative weak dominance is not unique in general!

<table>
<thead>
<tr>
<th></th>
<th>Forward</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Blue</strong></td>
<td>$(1-c, 1-c)$</td>
<td>$(-c, 0)$</td>
</tr>
<tr>
<td><strong>Green</strong></td>
<td>$(0, 0)$</td>
<td>$(0, 0)$</td>
</tr>
</tbody>
</table>
# Nash equilibrium (1/2)

**Nash Equilibrium**: no player can increase its utility by deviating unilaterally

**E1: The Forwarder’s Dilemma**

<table>
<thead>
<tr>
<th></th>
<th>Forward</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Blue</strong></td>
<td>$(\frac{1-c}{2}, \frac{1-c}{2})$</td>
<td>$(\frac{-c}{2}, 1)$</td>
</tr>
<tr>
<td><strong>Green</strong></td>
<td>$(\frac{1-c}{2}, \frac{1-c}{2})$</td>
<td>$(0, 0)$</td>
</tr>
</tbody>
</table>

**E2: The Joint Packet Forwarding game**

<table>
<thead>
<tr>
<th></th>
<th>Forward</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Blue</strong></td>
<td>$(1-c, 1-c)$</td>
<td>$(0, 0)$</td>
</tr>
<tr>
<td><strong>Green</strong></td>
<td>$(1-c, 1-c)$</td>
<td>$(0, 0)$</td>
</tr>
</tbody>
</table>
Strategy profile $s^*$ constitutes a **Nash equilibrium** if, for each player $i$,

$$u_i(s_i^*, s_{-i}^*) \geq u_i(s_i, s_{-i}^*), \forall s_i \in S_i$$

where: $u_i \in U$ utility function of player $i$

$s_i \in S_i$ strategy of player $i$

The **best response** of player $i$ to the profile of strategies $s_{-i}$ is a strategy $s_i$ such that:

$$b_i(s_{-i}) = \arg\max_{s_i \in S_i} u_i(s_i, s_{-i})$$

**Nash Equilibrium = Mutual best responses**

**Caution!** Many games have more than one Nash equilibrium
Example 3: The Multiple Access game

<table>
<thead>
<tr>
<th></th>
<th>Blue</th>
<th>Green</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quiet</td>
<td>(0, 0)</td>
<td>(0, 1-c)</td>
</tr>
<tr>
<td>Transmit</td>
<td>(1-c, 0)</td>
<td>(-c, -c)</td>
</tr>
</tbody>
</table>

Reward for successful transmission: 1
Cost of transmission: c (0 < c << 1)

There is no strictly dominating strategy
There are two Nash equilibria
Mixed strategy Nash equilibrium

\( p: \) probability of transmit for Blue
\( q: \) probability of transmit for Green

\[ u_{\text{blue}} = p(1-q)(1-c) - pqc = p(1-c-q) \]
\[ u_{\text{green}} = q(1-c-p) \]

Objectives
- Blue: choose \( p \) to maximize \( u_{\text{blue}} \)
- Green: choose \( q \) to maximize \( u_{\text{green}} \)

\( p = 1-c, \ q = 1-c \)

is a Nash equilibrium
Example 4: The Jamming game

Transmitter:
- Reward for successful transmission: 1
- Loss for jammed transmission: -1

Jammer:
- Reward for successful jamming: 1
- Loss for missed jamming: -1

Two channels: C₁ and C₂

<table>
<thead>
<tr>
<th></th>
<th>C₁</th>
<th>C₂</th>
</tr>
</thead>
<tbody>
<tr>
<td>C₁</td>
<td>(-1, 1)</td>
<td>(1, -1)</td>
</tr>
<tr>
<td>C₂</td>
<td>(1, -1)</td>
<td>(-1, 1)</td>
</tr>
</tbody>
</table>

There is no pure-strategy Nash equilibrium

\[ p = \frac{1}{2}, \quad q = \frac{1}{2} \] is a Nash equilibrium

\[ p \]: probability of transmit on C₁ for Blue
\[ q \]: probability of transmit on C₁ for Green
Theorem by Nash, 1950

**Theorem:**
Every finite strategic-form game has a mixed-strategy Nash equilibrium.
### Efficiency of Nash equilibria

**E2: The Joint Packet Forwarding game**

<table>
<thead>
<tr>
<th></th>
<th>Forward</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Blue</strong></td>
<td>(1-c, 1-c)</td>
<td>(0, 0)</td>
</tr>
<tr>
<td><strong>Green</strong></td>
<td>(-c, 0)</td>
<td>(0, 0)</td>
</tr>
</tbody>
</table>

**How to choose between several Nash equilibria?**

**Pareto-optimality**: A strategy profile is Pareto-optimal if it is not possible to increase the payoff of any player without decreasing the payoff of another player.
How to study Nash equilibria?

Properties of Nash equilibria to investigate:
- existence
- uniqueness
- efficiency (Pareto-optimality)
- emergence (dynamic games, agreements)
B.1 Introduction
B.2 Static games
B.3 Dynamic games
B.4 Repeated games
Extensive-form games

- usually to model sequential decisions
- game represented by a tree
- Example 3 modified: the **Sequential** Multiple Access game:
  Blue plays first, then Green plays.

![Diagram](image)

Reward for successful transmission: 1

Cost of transmission: c

\(0 < c << 1\)
Strategies in dynamic games

- The strategy defines the moves for a player for every node in the game, even for those nodes that are not reached if the strategy is played.

strategies for Blue: T, Q

strategies for Green: TT, TQ, QT and QQ

If they have to decide independently: three Nash equilibria

(T,QT), (T,QQ) and (Q,TT)
Backward induction

- Solve the game by reducing from the final stage
- Eliminates Nash equilibria that are *incredible threats*

incredible threat: (Q, TT)

\[
\begin{array}{c|cc}
& T & Q \\
\hline
T & (1-c,0) & (0,1-c) \\
\end{array}
\]

\[
\begin{array}{c}
\hline
& (0,0) \\
\end{array}
\]

\[
\begin{array}{c|cc}
& T & Q \\
\hline
(-c,-c) & (1-c,0) & (0,1-c) \\
\end{array}
\]

\[
\begin{array}{c}
\hline
& (0,0) \\
\end{array}
\]
Subgame perfection

- Extends the notion of Nash equilibrium

**One-deviation property:** A strategy $s_i$ conforms to the *one-deviation property* if there does not exist any node of the tree, in which a player $i$ can gain by deviating from $s_i$ and apply it otherwise.

**Subgame perfect equilibrium:** A strategy profile $s$ constitutes a subgame perfect equilibrium if the one-deviation property holds for every strategy $s_i$ in $s$.

Finding subgame perfect equilibria using backward induction

Subgame perfect equilibria: $(T, QT)$ and $(T, QT)$

![Diagram of a game tree with strategies and payoffs]

- Blue
  - $T$: (1-c,0)
  - $Q$: (0,1-c)

- Green
  - $T$: (-c,-c)
  - $Q$: (0,0)
Chapter outline

B.1 Introduction
B.2 Static games
B.3 Dynamic games
B.4 Repeated games
Repeated games

- repeated interaction between the players (in stages)
- move: decision in one interaction
- strategy: defines how to choose the next move, given the previous moves
- history: the ordered set of moves in previous stages
  - most prominent games are history-1 games (players consider only the previous stage)
- initial move: the first move with no history
- finite-horizon vs. infinite-horizon games
- stages denoted by $t$ (or $k$)
Utilities: Objectives in the repeated game

- finite-horizon vs. infinite-horizon games
- myopic vs. long-sighted repeated game

myopic: \( \overline{u}_i = u_i(t + 1) \)

long-sighted finite: \( \overline{u}_i = \sum_{t=0}^{T} u_i(t) \)

long-sighted infinite: \( \overline{u}_i = \sum_{t=0}^{\infty} u_i(t) \)

utility with discounting: \( \overline{u}_i = \sum_{t=0}^{\infty} u_i(t) \cdot \omega^t \)

\( 0 < \omega \leq 1 \) is the discounting factor
Strategies in the repeated game

- usually, history-1 strategies, based on different inputs:
  - others’ behavior: \( m_i(t+1) = s_i[m_{-i}(t)] \)
  - others’ and own behavior: \( m_i(t+1) = s_i[m_i(t), m_{-i}(t)] \)
  - utility: \( m_i(t+1) = s_i[u_i(t)] \)

Example strategies in the Forwarder’s Dilemma:

<table>
<thead>
<tr>
<th>Blue (t)</th>
<th>initial move</th>
<th>F</th>
<th>D</th>
<th>strategy name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green (t+1)</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>AllC</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>F</td>
<td>D</td>
<td>Tit-For-Tat (TFT)</td>
</tr>
<tr>
<td></td>
<td>D</td>
<td>D</td>
<td>D</td>
<td>AllID</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>D</td>
<td>F</td>
<td>Anti-TFT</td>
</tr>
</tbody>
</table>
The Repeated Forwarder’s Dilemma

<table>
<thead>
<tr>
<th></th>
<th>Forward</th>
<th>Drop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forward</td>
<td>$(1-c, 1-c)$</td>
<td>$(-c, 1)$</td>
</tr>
<tr>
<td>Drop</td>
<td>$(1, -c)$</td>
<td>$(0, 0)$</td>
</tr>
</tbody>
</table>

Blue

Green

stage payoff
Analysis of the Repeated Forwarder’s Dilemma (1/3)

infinite game with discounting:  
\[ \bar{u}_i = \sum_{t=0}^{\infty} u_i(t) \cdot \omega^t \]

<table>
<thead>
<tr>
<th>Blue strategy</th>
<th>Green strategy</th>
<th>Blue utility</th>
<th>Green utility</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllID</td>
<td>AllID</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>AllID</td>
<td>TFT</td>
<td>1</td>
<td>-c</td>
</tr>
<tr>
<td>AllID</td>
<td>AllIC</td>
<td>(1/(1-\omega))</td>
<td>-c/(1-\omega)</td>
</tr>
<tr>
<td>AllIC</td>
<td>AllIC</td>
<td>(1-c)/(1-\omega)</td>
<td>(1-c)/(1-\omega)</td>
</tr>
<tr>
<td>AllIC</td>
<td>TFT</td>
<td>(1-c)/(1-\omega)</td>
<td>(1-c)/(1-\omega)</td>
</tr>
<tr>
<td>TFT</td>
<td>TFT</td>
<td>(1-c)/(1-\omega)</td>
<td>(1-c)/(1-\omega)</td>
</tr>
</tbody>
</table>
### Analysis of the Repeated Forwarder’s Dilemma (2/3)

<table>
<thead>
<tr>
<th>Blue strategy</th>
<th>Green strategy</th>
<th>Blue utility</th>
<th>Green utility</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllD</td>
<td>AllD</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>AllD</td>
<td>TFT</td>
<td>1</td>
<td>-c</td>
</tr>
<tr>
<td>AllD</td>
<td>AllC</td>
<td>(\frac{1}{1-\omega})</td>
<td>-(\frac{c}{1-\omega})</td>
</tr>
<tr>
<td>AllC</td>
<td>AllC</td>
<td>(\frac{(1-c)}{(1-\omega)})</td>
<td>(\frac{(1-c)}{(1-\omega)})</td>
</tr>
<tr>
<td>AllC</td>
<td>TFT</td>
<td>(\frac{(1-c)}{(1-\omega)})</td>
<td>(\frac{(1-c)}{(1-\omega)})</td>
</tr>
<tr>
<td>TFT</td>
<td>TFT</td>
<td>(\frac{(1-c)}{(1-\omega)})</td>
<td>(\frac{(1-c)}{(1-\omega)})</td>
</tr>
</tbody>
</table>

- AllC receives a high payoff with itself and TFT, but
- AllD exploits AllC
- AllD performs poor with itself
- TFT performs well with AllC and itself, and
- TFT retaliates the defection of AllD

**TFT is the best strategy if \(\omega\) is high!**
Analysis of the Repeated Forwarder’s Dilemma (3/3)

<table>
<thead>
<tr>
<th>Blue strategy</th>
<th>Green strategy</th>
<th>Blue utility</th>
<th>Green utility</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllD</td>
<td>AllD</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TFT</td>
<td>TFT</td>
<td>((1-c)/(1-\omega))</td>
<td>((1-c)/(1-\omega))</td>
</tr>
</tbody>
</table>

**Theorem:** In the Repeated Forwarder’s Dilemma, if both players play AllD, it is a Nash equilibrium.

**Theorem:** In the Repeated Forwarder’s Dilemma, both players playing TFT is a Nash equilibrium as well.

The Nash equilibrium \(s_{\text{Blue}} = \text{TFT}\) and \(s_{\text{Green}} = \text{TFT}\) is Pareto-optimal (but \(s_{\text{Blue}} = \text{AllD}\) and \(s_{\text{Green}} = \text{AllD}\) is not)!
Experiment: Tournament by Axelrod, 1984

- any strategy can be submitted (history-X)
- strategies play the Repeated Prisoner’s Dilemma (Repeated Forwarder’s Dilemma) in pairs
- number of rounds is finite but unknown
  - TFT was the winner
- second round: TFT was the winner again
Discussion on game theory

- Rationality
- Utility function and cost
- Pricing and mechanism design (to promote desirable solutions)
- Infinite-horizon games and discounting
- Reputation
- Cooperative games
- Imperfect / incomplete information
Who is malicious? Who is selfish?

Harm everyone: viruses,…

Selective harm: DoS,…

Cyber-gangster: phishing attacks, trojan horses,…

Big brother

Spammer

Greedy operator

Selfish mobile station

➡️ Both security and game theory backgrounds are useful in many cases!!
Conclusions

- Game theory can help modeling greedy behavior in wireless networks
- Discipline still in its infancy
- Alternative solutions
  - Ignore the problem
  - Build protocols in tamper-resistant hardware
Chapter 10: Selfishness in packet forwarding
Introduction

- the operation of multi-hop wireless networks requires the nodes to forward data packets on behalf of other nodes
- however, such cooperative behavior has no direct benefit for the forwarding node, and it consumes valuable resources (battery)
- hence, the nodes may tend to behave selfishly and deny cooperation
- if many nodes defect, then the operation of the entire network is jeopardized
- questions:
  - What are the conditions for the emergence of cooperation in packet forwarding?
  - Can it emerge spontaneously or should it be stimulated by some external mechanism?
Modeling packet forwarding as a game

**Players:** nodes

**Strategy:**
- cooperation level

<table>
<thead>
<tr>
<th>time slot:</th>
<th>0</th>
<th>1</th>
<th>t</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p_C(0) )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( p_C(1) )</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( p_C(t) )</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Payoff** (of node i):
- proportion of packets sent by node i reaching their destination
Cost function

Cost for forwarder $f_j$:

$$\eta_{f_j}(r,t) = -T_s(r) \cdot c \cdot \hat{\tau}_j(r,t)$$

where:

- $T_s(r)$ – traffic sent by source $s$ on route $r$
- $c$ – unit cost of forwarding

Example:

$$\hat{\tau}_C(r,t) = \prod_{k \in \{E,C\}} p_{f_k}(t) = p_E(t) \cdot p_C(t)$$

$$\eta_C(r,t) = -T_A(r) \cdot c \cdot \hat{\tau}_j(r,t)$$

Normalized throughput at forwarder $f_j$:

$$\hat{\tau}_j(r,t) = \prod_{k=1}^{j} p_{f_k}(t)$$

where:

- $r$ – route on which $f_k$ is a forwarder
- $t$ – time slot
- $f_k$ – forwarders on route $r$
- $p_{f_k}$ – cooperation level of forwarder $f_k$

Security and Cooperation in Wireless Networks
Appendix B: A tutorial on game theory for wireless networks

10.1 Game theoretic model of packet forwarding
Utility function

Experienced throughput:

\[ \tau(r,t) = T_s(r) \cdot \prod_{k=1}^{l} p_{f_k}(t) \]

where:
- \( s \) – source
- \( r \) – route on which \( s \) is a source
- \( t \) – time slot
- \( f_k \) – forwarders for \( s \)
- \( p_{f_k} \) – cooperation level of forwarder \( f_k \)

Example:

\[ \tau(r,t) = T_A(r) \cdot p_E(t) \cdot p_C(t) \]
Total payoff

Payoff = Utility - Cost

\[ \pi_i(t) = \sum_{q \in S_i(t)} u_i(\tau(q, t)) + \sum_{r \in F_i(t)} \eta_i(r, t) \]

where:  
\( S_i(t) \) – set of routes on which \( i \) is a source  
\( F_i(t) \) – set of routes on which \( i \) is a forwarder

The goal of each node is to maximize its total payoff over the game

\[ \max \bar{\pi}_i = \sum_{t=0}^{\infty} \pi_i(t) \cdot \omega^t \]

where: \( \omega \) – discounting factor  
\( t \) – time

Example:

<table>
<thead>
<tr>
<th>Payoff:</th>
<th>( \pi_A(0) )</th>
<th>( \pi_A(1) \cdot \omega )</th>
<th>( \pi_A(t) \cdot \omega^t )</th>
</tr>
</thead>
<tbody>
<tr>
<td>time slot:</td>
<td>0</td>
<td>1</td>
<td>( t )</td>
</tr>
</tbody>
</table>
Representation of the nodes as players

Node $i$ is playing against the rest of the network (represented by the box denoted by $A_{-i}$)

Strategy function for node $i$:

$$p_i(t) = \sigma(\tau(r, t \mid t - 1) \mid r \in S_i(t - 1))$$

where:

- $\tau(r, t)$ – experienced throughput
- $S_i$ – set of routes on which $i$ is a source
Examples of strategies

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Initial cooperation level</th>
<th>Function $\sigma_i(y_i) = x_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllD (always defect)</td>
<td>0</td>
<td>$\sigma_i(y_i) = 0$</td>
</tr>
<tr>
<td>AllC (always cooperate)</td>
<td>1</td>
<td>$\sigma_i(y_i) = 1$</td>
</tr>
<tr>
<td>TFT (Tit-For-Tat)</td>
<td>1</td>
<td>$\sigma_i(y_i) = y_i$</td>
</tr>
</tbody>
</table>

where $y_i$ stands for the input

- **non-reactive strategies**:  
  the output of the strategy function is independent of the input (example: AllD and AllC)

- **reactive strategies**:  
  the output of the strategy function depends on the input (example: TFT)
**Concept of dependency graph**

**dependency:** the benefit of each source is dependent on the behavior of its forwarders

(a) routes

(b) dependency graph

dependency loop
Analytical Results (1/2)

**Theorem 1:** If node $i$ does not have any dependency loops, then its best strategy is AllD.

**Theorem 2:** If node $i$ has only non-reactive dependency loops, then its best strategy is AllD.

**Corollary 1:** If every node plays AllD, it is a Nash-equilibrium.
Theorem 3 (simplified): Assuming that node $i$ is a forwarder, its behavior will be cooperative only if it has a dependency loop with each of its sources.

Corollary 2: If Theorem 3 holds for every node, it is a Nash-equilibrium.

Example in which Corollary 2 holds:

![Network Diagram]

![Dependency Graph]
Classification of scenarios

D: Set of scenarios, in which every node playing AllD is a Nash equilibrium

C: Set of scenarios, in which a Nash equilibrium based on cooperation is not excluded by Theorem 1

C2: Set of scenarios, in which cooperation is based on the conditions expressed in Corollary 2
### Simulation settings

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of nodes</td>
<td>100, 150, 200</td>
</tr>
<tr>
<td>Distribution of the nodes</td>
<td>random uniform</td>
</tr>
<tr>
<td>Area type</td>
<td>torus</td>
</tr>
<tr>
<td>Area size</td>
<td>1500x1500m, 1850x1850m, 2150x2150m</td>
</tr>
<tr>
<td>Radio range</td>
<td>200 m</td>
</tr>
<tr>
<td>Number of routes originating at each node</td>
<td>1-10</td>
</tr>
<tr>
<td>Route selection</td>
<td>shortest path</td>
</tr>
<tr>
<td>Number of simulation runs</td>
<td>1000</td>
</tr>
</tbody>
</table>
Simulation results

![Graph showing simulation results](image)

- **Proportion of scenarios, where all nodes have a dependency loop**
- **Number of routes originating at each node**

Legend:
- 100 nodes
- 150 nodes
- 200 nodes
Summary

- **Analytical results:**
  - If everyone drops all packets, it is a Nash-equilibrium
  - *In theory*, given some conditions, a cooperative Nash-equilibrium can exist (i.e., each forwarder forwards all packets)

- **Simulation results:**
  - *In practice*, the conditions for cooperative Nash-equilibria are very restrictive: the likelihood that the conditions for cooperation hold for every node is extremely small

- **Consequences:**
  - Cooperation cannot be taken for granted
  - Mechanisms that stimulate cooperation are necessary
    - incentives based on virtual currency
    - reputation systems
Chapter 11: Wireless operators in shared spectrum

(multi-domain sensor networks)

border games in cellular networks
Introduction

- spectrum licenses do not regulate access over national borders
- adjust pilot power to attract more users

Is there an incentive for operators to apply competitive pilot power control?
System model (1/2)

Network:
- cellular networks using CDMA
  - channels defined by orthogonal codes
- two operators: A and B
- one base station each
- pilot signal power control

Users:
- roaming users
- users uniformly distributed
- select the best quality BS
- selection based signal-to-interference-plus-noise ratio ($SINR$)
System model (2/2)

pilot signal $SINR$:

$$SINR_{i\text{v}}^{pilot} = \frac{G_p^{pilot} \cdot P_i \cdot g_{iv}}{N_0 \cdot \mathbb{W} + I_{\text{own}}^{pilot} + I_{\text{other}}^{pilot}}$$

$$I_{\text{own}}^{pilot} = \varsigma \cdot g_{iv} \cdot \left( \sum_{w \in \mathcal{M}_i} T_{iw} \right)$$

$$I_{\text{other}}^{pilot} = \eta \cdot \sum_{j \neq i} g_{jv} \cdot \left( P_j + \sum_{w \in \mathcal{M}_i} T_{iw} \right)$$

traffic signal $SINR$:

$$SINR_{i\text{v}}^{tr} = \frac{G_p^{tr} \cdot T_{iv} \cdot g_{iv}}{N_0 \cdot \mathbb{W} + I_{\text{own}}^{tr} + I_{\text{other}}^{tr}}$$

$$I_{\text{own}}^{tr} = \varsigma \cdot g_{iv} \cdot \left( P_i + \sum_{w \neq v, w \in \mathcal{M}_i} T_{iw} \right)$$

$$I_{\text{other}}^{tr} = I_{\text{other}}^{pilot}$$

$P_i$ – pilot power of $i$

$G_p^{pilot}$ – processing gain for the pilot signal

$g_{iv}$ – channel gain between BS $i$ and user $v$

$N_0$ – noise energy per symbol

$\mathbb{W}$ – available bandwidth

$I_{\text{own}}^{pilot}$ – own-cell interference affecting the pilot signal

$\varsigma$ – own-cell interference factor

$T_{iv}$ – traffic power between BS $i$ and user $v$

$\mathcal{M}_i$ – set of users attached to BS $i$

$\eta$ – other-to-own-cell interference factor

11.2 Border games in cellular networks

11.2.1 Model
Game-theoretic model

- **Power Control Game, \( G_{PC} \)**
  - players → networks operators (BSs), A and B
  - strategy → pilot signal power, \( 0W < P_i < 10W, i = \{A, B\} \)
  - standard power, \( P^S = 2W \)
  - payoff → profit, \( u_i = \sum_{v \in M_i} \theta_v \) where \( \theta_v \) is the expected income serving user \( v \)
  - normalized payoff difference:
    \[
    \Delta_i = \max_{s_i} \left( u_i(s_i, P^S) - u_i(P^S, P^S) \right) / u_i(P^S, P^S)
    \]
## Simulation settings

### Parameter | Value
---|---
simulation area size | 1 km² (250 m, 500 m) and (750 m, 500 m)
BS positions | (750 m, 500 m)
user distribution | random uniform
number of simulations | 500
default path loss exponent, \( \alpha \) | 4
BS max power | 43 dBm = 20 W
BS max load | 40 dBm = 10 W
BS standard power, \( P^s \) | 33 dBm = 2 W
BS min power | 20 dBm = 0.1 W
power control step size, \( P_{step} \) | 0.1 W
orthogonality factor, \( \zeta \) | 0.4
other-to-own-cell interference factor, \( \eta \) | 0.4
user traffic types:
- audio, \( R^{tr} = 12.2 \) kbps
- video, \( R^{tr} = 144 \) kbps
- data, \( R^{tr} = 384 \) kbps

required CIR (audio, video, data):
-20 dB, -12.8 dB, -9 dB
expected incomes (\( \theta_{audio}, \theta_{video}, \theta_{data} \)):
10, 20, 50 CHF/month
Is there a game?

- only A is strategic (B uses $P_B = P^S$)
- 10 data users
- path loss exponent, $\alpha = 2$
When both operators are strategic

- 10 data users
- path loss exponent, $\alpha = 4$
Nash equilibria

10 data users

100 data users
Efficiency (1/2)

- 10 data users

![Graph showing payoffs and Nash equilibrium in cellular networks](image)
Efficiency (2/2)

- 100 data users

![Graph showing payoff vs. payoff gap with markers for payoffs, Pareto-optimal, standard, and Nash equilibrium.]
Convergence to NE (1/2)

- convergence based on better-response dynamics
- convergence step: 2 W
Convergence to NE (2/2)

- convergence step: 0.1 W
Conclusion on border games

- not only individual nodes may exhibit selfish behavior, but operators can be selfish too
- example: adjusting pilot power to attract more users at national borders
- the problem can be modeled as a game between the operators
  - the game has an efficient Nash equilibrium
  - there’s a simple convergence algorithm that drives the system into the Nash equilibrium