Security of Wireless Networks
Lecture 7

Srdjan Ćapkun
Department of Computer Science
ETH Zurich
Physical Layer Security

Broadcast Jamming-Resistant Communication
– keys, some keys, no keys –
Broadcast Communication

Broadcast communication

- One sender, many receivers
- Open system
  - New receivers may join, receivers may withdraw
  - Any receiver can listen (in contrast to multicast)

Examples:
- radio (audio) broadcast (AM, FM, ...)
- navigation signals: satellite-based (GPS), terrestrial (LORAN)
Attacks on Broadcast Communication

For pairwise (unicast) communication we only consider *external* (*outsider*) *attackers*:

- $A$ and $B$ are mutually trusted
- Attacker uses only public information

Broadcast communication:

- High and unknown number of receivers
- Receivers are potentially untrusted and may be colluding
- We need to consider *external attackers* and *internal* (*insider*) *attackers* (can be more efficient)
- Group keys?
External Attackers on SS Techniques

**External attacker**

- Does *not* know the spreading code / hopping sequence
- Partial-band attacker can still jam. Example: FHSS

\[ p_j = \text{Probability that the packet is jammed} \]
\[ = 1 - (1 - c_j/c)^{n_j} \]

- \( c \) = # frequency channels
- \( c_j \) = # channels the jammer jams
- \( n_j \) = # jamming cycles per packet (given by min. jamming period, packet length, and jammer capabilities)

Typical computation of jamming probability via the inverse
Internal Attackers on SS Techniques

Internal attacker

- Legitimate receiver: can decode the broadcast signal, i.e. knows the used spreading code and its synchronization
- Can *misuse the spreading code and synchronization* for jamming to disable other receivers to get the signal
- *Group keys do not prevent this attack!* We need a better solution!
Anti-jamming Broadcast

Problem: Base station (BS) needs to broadcast an (authenticated / confidential) message to a large number of receivers in an anti-jamming manner

Desirable properties:
• Detect / prevent jamming
• Support a flexible number of receivers
• Tolerate a certain fraction of malicious receivers

Some solutions based on keys shared between sender and receivers:
1. Desmedt et al.: FHSS-based – each receiver listens to a subset of frequencies on which the sender transmits
2. Chiang, Hu: DSSS-based – codes assigned to each receiver
Broadcast Anti-jamming Systems [Desmedt et al.] - I

Broadcast anti-jamming based on frequency hopping (FHSS)

Coding method provides protection against malicious receivers

- Base station transmits the same signal simultaneously on multiple frequencies
- Each receiver listens to a subset of these frequencies at a given time
- \textit{Threshold scheme}: provides protection against up to $j-I$ colluding receivers

Based on secret information
Public Channel Allocation Table

- Defines the subset of channels where each receiver $R_i$ is listening
- Known to every receiver
- $j-1$ receivers do not cover all channels of any other receiver
- Set coverage problem

| Channel | BS | R1 | R2 | R3 | R4 | ...
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Secret Frequency Allocation Table

- The actual frequencies are secret
- Created and updated via a pseudo-noise generator

| Channel | 1     | 2     | 3     | 4     | 5     | 6     | 7     | 8     | 9     | ...
|---------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-----|
| Frequency (in GHz) | 2.437 | 2.462 | 2.417 | 2.442 | 2.447 | 2.457 | 2.412 | 2.422 | 2.432 | ...

[Snapshot of the frequency allocation table, the complete table is only known to the base station]
Broadcast Anti-jamming Systems [Desmedt et al.] - III

System Description:

- Channels \( C = \{c_1, c_2, \ldots, c_m\} \)
- Receivers \( R = \{R_1, R_2, \ldots, R_l\} \)
- Subsets of channels \( CR = \{C_1, C_2, \ldots, C_l\} \)

**Theorem:** If \( |C_i| \geq 1 + (j - 1)d \) for all \( 1 \leq i \leq l \) and \( |C_i \cap C_k| \leq d \) for all \( i \neq k \), then \((C, CR)\) is a Broadcast Anti-Jamming System.

Sufficient but not necessary condition

**Example:** \( C = \{1,2,3,4,5,6\}, \ R = \{B_1,B_2,B_3,B_4\}, \ CR= \{\{1,2\}, \{2,3\}, \{4,5\}, \{5,6\}\} \)

- Resistant to \( j = 3 \) jammers, i.e., \( j - 1 = 2 \)
- \( m = 6, \ l = 4, \ |C_i \cap C_k| \leq d = 1 \)
- Yet \( |C_i| = 2 \), not the required \( |C_i| \geq 1 + (j - 1)d = 3 \)
The Desmedt broadcast anti-jamming system works if

- the group of colluders consists of \( j - 1 \) or fewer members and hence each receiver is always left with at least one free (\( = \) unjammed) channel
- the assigned frequencies can be distributed over a broad, non-continuous frequency band

However, this scheme requires secret information to be shared between the base station and each participating receiver \( \rightarrow \) multicast solution
Dynamic Jamming Mitigation

Broadcast anti-jamming based on DSSS

Counteract jamming by using a balanced binary key tree

- Each node corresponds to a spreading code
- Each user $N_i$ is assigned to a leaf and knows all codes on the path from the root

The base station transmits on ...

- a disjoint cover of codes, i.e., all users can decode using exactly one code
- a set of test codes

If a user receives a message on a test code but not on the corresponding detectable code, it reports jamming
Dynamic Jamming Mitigation [Chiang and Hu] – II

Jamming detection and mitigation

Detection and mitigation rely on feedback

Splitting and reforming the tree allows the transmitter to send each transmission on \( \leq 2j+1 \) codes, where \( j \) is the (expected upper) number of jammers (details omitted)
Dynamic Jamming Mitigation [Chiang and Hu] – III

Requires highly flexible base station (sending and receiving on a potentially large number of codes) and feedback channels

- Not applicable to unidirectional broadcast

Requires secrets to be shared between the base station and the receivers

- Each receiver knows the codes on its path to the root but no other codes
- Number of required secrets grows with the number of receivers
Looking back ...

Introduction to broadcast systems

Group keys are not a solution against jamming

Two solutions based on secrets shared between the base station and the receivers:

- FH-based by Desmedt et. al
- DSSS-based by Chiang et. al

*Can we achieve jamming-resistant communication without shared secrets?*
Physical Layer Security

Broadcast Anti-Jamming Techniques
Without Shared Secrets
Anti-jamming Broadcast *Without Shared Keys*

**Problem:** BS needs to broadcast an (authenticated) message to a large number of *unknown/untrusted receivers* in an *anti-jamming manner.*

Applications: alarm broadcast, navigation signals, etc ... 

But ...  
- Anti-Jamming communication relies on shared secret keys  
- In anti-jamming broadcast we cannot rely on shared keys (unknown/untrusted receivers)  
- The prior schemes (Desmedt, Chiang) do not work for unknown receivers  
- Public-key crypto does not help
Anti-Jamming Key Establishment

Problem:
$A$ and $B$ want to establish a shared secret key in the presence of a jammer $J$

Assumptions:
$A$ and $B$ do not share any secrets
The clocks of $A$ and $B$ are loosely synchronized $O(s)$
Each node has a public/private key pair and a certificate binding its identity to the public key
$CA$ (Certification Authority) is trusted by all nodes; it may be off-line or unreachable by the nodes at the time of communication
Anti-Jamming / Key-establishment Dependency

Key establishment depends on jamming-resistant communication

Common anti-jamming techniques require a shared secret key (code)

Leads to an anti-jamming/key-establishment dependency cycle
Two Solutions: UFH and UDSSS

Basic idea:
- If you cannot coordinate the sender and the receiver – Don’t!
- Sender uses random hopping sequences / spreading codes unknown to the receiver (public set)

Two solutions:
- Uncoordinated Frequency Hopping Spread Spectrum (UFH)
- Uncoordinated Direct Sequence Spread Spectrum (UDSSS)

Rationale:
- The attacker cannot predict which channels will be used (neither can the receiver)
- Equivalent to FH in jamming protection (but not in throughput)
- Throughput can be improved by using broadband receivers \( (c_t, c_r) \)
Attacker Model

- Attacker goal: to prevent communication!
- Attacker actions: *Jam, Insert, Modify*
- Attacker types: Responsive, Sweep, Random, ...
- Attacker strength (channels/time to jam/sense): $c_s/t_s$, $c_j/t_j$
- Power to insert, jam, and overshadow: $P_t$, $P_j$, and $P_o$

\[ c_t P_t + c_j P_j + c_o P_o \leq P_T \]

- $P_T$: total signal strength that attacker $J$ can achieve at the receiver $B$

- Given the number of frequency channels on which the attacker inserts ($c_t$), jams ($c_j$), and overshadows ($c_o$),
Uncoordinated Frequency Hopping (transmitter)

1. Fragmentation

2. Fragment linking (protects against insertion)

3. Packet Encoding (ECC) (protects against jamming)

4. Repeated transmission
Uncoordinated Frequency Hopping (receiver)

1. Receiving packets
   \[ f_1: m_1 \rightarrow m_3 \]
   \[ f_2: m_2 \rightarrow m_1 \]

2. Packet decoding

3. Ordering and linking

4. Message reassembly and signature verification
   \[ M := m, \text{sig}(m), \ldots \]
Problem: Fragments are not individually authenticated (pollution attacks)

Signature verification at each candidate message (after reassembly)

In the best case, $I=1 \ldots$ (depends on attacker’s # of channels, power \ldots)

but $\ell$ is large; $\ell = \frac{\text{message size}}{\text{slot size}}$ ($>20$)

Result: Attacker performs a DoS attack on the logical level instead on the physical
Security analysis: Fragment linking

**Problem:** Fragments are not individually authenticated (pollution attacks)

**Solution:** Cryptographically link fragments (no reliance on shared key)

to achieve *message integrity*

### Hash linking

\[ m_i := \text{id}||i||l||M_i||h_{i+1}||\ldots||h_{i+\alpha} \]

### One-way Accumulators

\[ m_i := \text{id}||i||l||M_i||w_i \]

**Min 1 hash**

**1 witness**

### Short signatures

\[ m_i := K_M||i||l||M_i||\text{Sig}_{K_M}^{-1}(K_M||i||l||M_i) \]

**1 short signature**
**Gain:** Instead of \((I+1)^t\) signature verifications, reduction to \((I+1)^\ell\) hash/acum/signature verifications + \((I+1)\) signature verifications

Signatures and accumulators better than hash linking

**Possible extensions:**
- Use linking with erasure codes, e.g., Fountain codes.
- Reconstruct the message from any \(k\) fragments.
Security analysis: Packet Encoding

Defined by the jamming resistance $\rho$ and coding rate $r_c$

- Packet transmission time:
- #channels that the attacker can (blindly) jam during the transmission:
- #channels that the attacker can scan during the transmission:
- #channels that the attacker can block during the transmission

$$t_m = |m| B / r_c$$

$$n_j := \frac{t_m}{\rho t_m + t_j}$$

$$n_s := \frac{t_m - \rho t_m - t_j}{t_s}$$

$$c_b = n_j c_j + n_s c_s$$

$$p_j = \frac{c_b}{c}$$
Performance Results

• Optimal # of channels ($c^* = 2c_b$)

Relative throughput w.r.t. coordinated FH

• Some results ($c = 200$, 1MBit/s, 1600 hops/s, ECC signature, $|M|=2176$, $\ell=13$)
  • Throughput: 1000x slower than FH
  • Latency: 2 – 100s (different attacker strengths)
Two Solutions: UFH and UDSSS

Basic idea:
- If you cannot coordinate the sender and the receiver – Don’t!
- Sender uses random hopping sequences / spreading codes unknown to the receiver (public set)

Two solutions:
- Uncoordinated Frequency Hopping Spread Spectrum (UFH)
- Uncoordinated Direct Sequence Spread Spectrum (UDSSS)

Rationale:
- The attacker cannot predict which spreading codes are used by the sender (neither can the receiver)
- UDSSS has reduced latency compared to DSSS
- Throughput can be improved by using parallelization
Uncoordinated Direct Sequence Spread Spectrum

**DSSS**
- Transmission chain
- Spread-Spectrum Code
- RF Out
- RF Link
- Receive chain
- Spread-Spectrum Code
- RF In
- Data Out

**UDSSS**
- Transmission chain
- Spread-Spectrum Code
- RF Out
- RF Link
- Public Set of Spreading Sequences $c_1, c_2, \ldots, c_n$
- Receive chain
- Spread-Spectrum Code
- RF In
- Data Out

As the pulses in time get shorter, the frequency bandwidth gets larger.
Uncoordinated Direct Sequence Spread Spectrum

- Public set $C$ of spreading sequences

Sender randomly selects sequence $c_s \in C$ to spread message $M$

Receivers record signal and despread $M$ by applying sequences from $C$ using a trial-and-error method

**UDSSS**

![Diagram of UDSSS system](image-url)
Uncoordinated Direct Sequence Spread Spectrum

- **Message repetitions**, due to
  - lacking synchronization between sender and receivers
  - the possibility of successful jamming attacks

![Diagram showing message repetitions and timing](image)
Uncoordinated Direct Sequence Spread Spectrum

- Code set $C$ composed of $n$ code sequences
- Each code sequence is composed of $\ell$ spreading codes containing $N$ chips
  - E.g., $N = 100$ chips → $20$ dB processing gain
  - Auto-correlation and cross-correlation properties
  - Successful despreading requires to hit the correct spreading sequence and the correct synchronization
UDSSS: Optimization

- **Idea**: Use UDSSS to transmit the spreading key only.
- **Trick**: First transmit message $M$ using a random spreading code $K$, then transmit the spreading code $K$ using UDSSS.

UDSSS: Example Application

- For positioning and/or time-synchronization
- Requirements:
  - signals from three to four different base stations
  - precise time-stamping of signal reception

- UDSSS provides:
  - anti-jamming transmission of multiple signals in parallel
  - precise time-stamping of signal reception (despite delayed recovery) & updated time-stamps in each transmitted message
  - anti-spoofing protection of authenticated messages
Summary

- Anti-jamming – key-establishment circular dependency
- Broadcast anti-jamming problem
- UDSSS and UFH
- New attacker models
- Applications
Physical Layer Security

Application of (Broadcast) Anti-Jamming Techniques to Key Establishment
Applications for Shared Keys in Wireless Networks

- Secret keys are required / used for:
  - Communication techniques (DSSS, Frequency Hopping)
  - Encryption of messages
  - Integrity protection of messages (MACs = Message Authentication Codes)
  - Authentication / authorized access
  - ...

[www.answers.com]
The Problem with Key Establishment

Key establishment is a challenge

**Pre-sharing Symmetric Keys**

- A Trusted Third Party (TTP) pre-loads the keys
- Efficient (+)
- Suffers from network dynamics problems (−):
  - new nodes joining, key revocation, key compromise

**Key Establishment**

- Based on public-key (asymmetric) cryptography
- Prominent examples: RSA, Diffie-Hellman (DH)
  - Based on computational hardness of the factorization (RSA) or discrete logarithm (DH) problem
- Requires reliable communication
DH Key Establishment

- Nodes A and B do not share any secrets, but possess certificates of their public keys
- Authenticated Diffie-Hellman Protocol (using signatures)

\[
g^a \mod p \\
g^b \mod p, \, \text{sig}_B(g^b, g^a) \\
\text{sig}_A(g^a, g^b) \\
K_{AB} = (g^b)^a \mod p \\
K_{AB} = (g^a)^b \mod p
\]

\[
\text{trusted Third Party } = \text{Certification Authority (CA)}
\]

- Conventional SS-Techniques cannot be used for the communication due to the missing shared secret
Anti-jamming / Key-establishment dependency

- Key establishment (e.g. using DH) depends on jamming-resistant communication

- Common anti-jamming techniques require a shared secret key (code)

- Leads to an anti-jamming/key-establishment dependency cycle

- Key idea: break the dependency cycle by using **Uncoordinated Frequency Hopping (UFH)**
ECC-based Station-to-Station Diffie-Hellman

- **Plies on elliptic curve** $E(F_q)$, $CA =$ Certification Authority
- $PK_A = A$'s public key, $\text{Sig}_A = A$'s signature, $r_A P = A$'s key contribution

Elliptic Curve Cryptography (ECC) enables to reduce the key length while maintaining the level of security

- E.g., 128-bit security level [NIST] → 256 bit prime fields on elliptic curves and 512 bit keys (vs. 3072-bit key for RSA)

Use UFH to transmit the messages
What to remember?

- What are broadcast systems?
- Applications for broadcast
- Approaches for enabling jamming-resistant broadcast despite internal attackers
- Jamming-resistant communication without shared secrets
- Anti-jamming/Key-establishment dependency
References


• Desmedt, Safavi-Naini, Wang, Charnes, Pieprzyk: Broadcast Anti-jamming Systems, International Conference on Networks (ICON) 1999


• Strasser, Pöpper, Capkun, Cagalj: Jamming-resistant Key Establishment using Uncoordinated Frequency Hopping, IEEE Symposium on Security and Privacy, 2008

• Strasser, Pöpper, Capkun: Efficient Uncoordinated FHSS Anti-Jamming Communication, ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), 2009

• Pöpper, Strasser, Capkun: Jamming-resistant Broadcast Communication without Shared Keys, Usenix Security, 2009

Signal Manipulations on the Physical Layer
Introductory Example: Alarm System

- Risk: Alarm messages can be *jammed*
- Solution: Use a protocol with *regular messages* and *energy detection*
- But: Can a burglar modify/remove signals with low energy?
Introductory Example: Alarm System

- This example is about the *compromise of communication links*

- How can an attacker circumvent energy detectors and still impact the signal at the receiver?
Attacker models

  - The omnipresent Dolev-Yao (DY) attacker can *eavesdrop*, *modify*, *replay*, and *delete* messages, *insert* own messages, and also instantaneously transfer information.

- This model is widely accepted for the security analysis of networks. Is it also appropriate for wireless transmissions?
  - Can the attacker indeed modify and delete messages transmitted by wireless signals (without being detected)?

- Which types of attacks on wireless transmissions exist and how feasible are these (low-power) attacks?
Classification of Wireless Attacks

\[ m_{A} \]  

\[ m' \]  

\[ m_{\text{received}} = m_{A} + m' \neq m_{A} \]

<table>
<thead>
<tr>
<th>Technique</th>
<th>Goal</th>
<th>Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symbol Modification</td>
<td>change</td>
<td>Bit Flipping</td>
</tr>
<tr>
<td></td>
<td>block</td>
<td>Preamble Corruption</td>
</tr>
<tr>
<td>Signal Strength Modification</td>
<td>change</td>
<td>Amplification</td>
</tr>
<tr>
<td></td>
<td>change</td>
<td>Attenuation</td>
</tr>
<tr>
<td></td>
<td>block</td>
<td>Annihilation</td>
</tr>
<tr>
<td>Overshadowing</td>
<td>change</td>
<td>Msg. Overshadowing</td>
</tr>
<tr>
<td></td>
<td>block</td>
<td>Noise Jamming</td>
</tr>
</tbody>
</table>
## Classification of Wireless Attacks

<table>
<thead>
<tr>
<th>Message layer effect</th>
<th>Signal layer effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deletion</td>
<td>Signal annihilation</td>
</tr>
<tr>
<td>Insertion</td>
<td>Noise jamming</td>
</tr>
<tr>
<td>Modification</td>
<td>Signal creation / replay</td>
</tr>
<tr>
<td>-</td>
<td>Symbol flipping</td>
</tr>
<tr>
<td>-</td>
<td>Overshadowing</td>
</tr>
<tr>
<td>-</td>
<td>Amplification</td>
</tr>
<tr>
<td>-</td>
<td>Attenuation</td>
</tr>
</tbody>
</table>

**Focus in this lecture:**

- $S = $ sender
- $R = $ receiver
- $A = $ attacker
Symbol Flipping for QPSK (e.g., 802.11b) Signals

I-channel

Q-channel

Received signal

[http://www.complextoreal.com]
Symbol Flipping for QPSK (e.g., 802.11b) Signals

• What are the required characteristics of the attacker’s flipping signal to achieve symbol flipping at the receiver?
  • Power
  • Frequency alignment (frequency of the carrier signal)
  • Carrier phase offset (phase offset of the carrier signal)
  • Baseband offset (bit offset of the carrier signal)

• What is the influence of the SNR (signal-to-noise ratio)?

• Possible way of investigation: Simulation (e.g., in Matlab)
  1. generate own signals
  2. record wireless signals (e.g., from transmissions of a software radio)
Carrier Phase Offset vs. Baseband Offset

- Carrier phase alignment and baseband alignment: Which offset of the attacker's signal with respect to the sender's signal at the receiver is required for successful symbol flipping?

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
</table>

Digital data

Modulated signal at \( R \)

Attacker's signal at \( R \)

carrier phase offset

baseband offset
Selected Simulation Results for Symbol Flipping

- **Carrier phase alignment** of the attacker's signal is the most critical factor.

- **Tolerance** of carrier phase offset for flipping *specific bits*: 13% (for two bit flips) to 25% (for one bit flip)
Results for Symbol Flipping

- **Deterministic symbol flipping** requires carrier phase alignment (and sufficient baseband alignment, frequency alignment, and power)

- **Random symbol flipping** does not require carrier phase alignment and can result in low-power message deletion and modification at the receiver
  - Will not be identified by energy detectors

- Without protection, symbol flipping can result in low-power spoofing attacks and manipulated messages
Signal Annihilation for QPSK (802.11b) Signals

- Experimental setup for evaluating *signal annihilation* attacks:

  - Changing the antennas position varies the arrival time of the attacker's replay signal (different propagation delay)
  - Carrier phase alignment matters
Results for Signal Annihilation (and Attenuation)

- High attenuation (or even annihilation at the receiver) achieved by destructive interference of the attacker’s signal with the original transmission at the receiver
- Highest attenuation for carrier phase shifts of $\pi$
Implications: The Alarm Scenario

- A solution based on energy detectors and regular message transmissions may not be enough
- Use/create channel randomness, carefully pick modulation scheme, use message integrity checks, ...
Conclusion: Important Aspects

• How can protocols (message transmissions) be attacked on the physical layer?
  • Jamming, symbol flipping, signal annihilation, attenuation and amplification, delay, ...

• What are conditions for these attacks to be successful?
  • Receiver sensitivity and environmental characteristics (SNR) also matter

• Take-home message: Viable security solutions must also defend against attacks on the underlying modulation scheme and physical layer
Further Reading

- Christina Pöpper, Nils Ole Tippenhauer, Boris Danev, Srdjan Capkun: “Investigation of Signal and Message Manipulations on the Wireless Channel”  
  In Proceedings of the European Symposium on Research in Computer Security (ESORICS), 2011
Acknowledgments

- Christina Pöpper