# Lightweight Protocol for Trusted Spontaneous Communication

Przemysław Błaśkiewicz, Marek Klonowski, Mirosław Kutyłowski, Piotr Syga

Wrocław University of Technology, Poland

INTRUST 2014, Beijing

# Ubiquitous Communication

# Model assumptions



## Transmission model

1. a single shared radio channel
2. a sink node and many sender nodes
3. one-hop network - the sink receives signals from each sender

## Activity model

1. unpredictable who and when will attempt to send data to the sink node
2. each communication is a stream of (encrypted) bits
3. length of a stream is unpredictable as well

# Traditional Approach to Multi-party Communication

# Self-organization

## Steps

1. estimating the number of communicating parties
2. leader election or initialization
3. assigning channel for exclusive use

# Layered approach



## Traditional approach

1. on low layers take care of physical problems of conflicts in transmission
2. on a higher layer reliable bit transmission
3. on top of that encryption

## Idea

- bring encryption on the lowest level
- skip conflict resolution - replace by Bloom filters

# Conflict Resolution



## Cai-Wang Scheme

- based on random experiment
- choose $r \in [0, t]$ at random
- monitor the channel at time $r$ and detect if there is carrier signal
- if there is not, then start sending carrier signal at time $r + \delta$
  ($\delta$ comes from technical limitations)

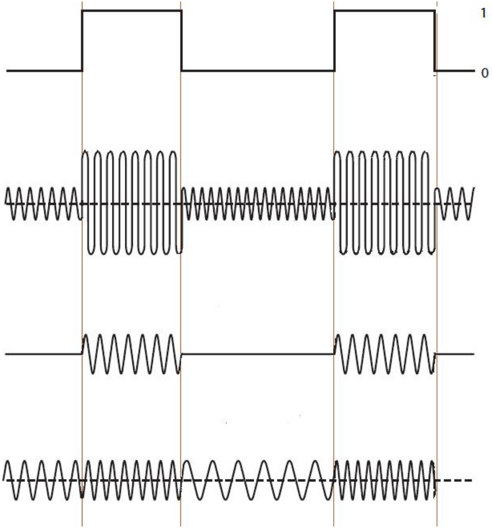
- possible extension network initialization – assigning consecutive numbers to all stations willing to transmit
- initialization not much useful if the situation changes dynamically

# Beeping Communication Model

# Information encoding - physical level

## amplitude modulation, frequency modulation, ...

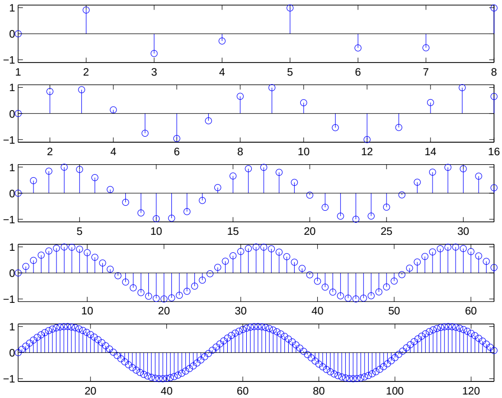




# Information decoding
## sampling

Lightweight Communication

- Ubiquitous communication
- Traditional approach
- Beeping model
- Bloom Filters
- Low layer encryption
- Properties




Frequency of the carrier signal is much higher than channel throughput

# Beeping model

## States of the communication channel

1. silence
2. beep

# Beeping model

## States of the communication channel

1. silence
2. beep

## Properties

- A beep is any activity at a given channel (frequency) above the natural noise level.
- no properties like amplitude, etc. are taken into consideration
- robust to signal interferences:
  - noise+noise $\rightarrow$ noise
  - noise+silence $\rightarrow$ noise
  - silence+silence $\rightarrow$ silence

# Bloom Filters

# Bloom Filter data structure



## Goal

- given a universe $U$ of objects, cardinality of $U$ relatively high
- a small number of elements to be stored in the filter

## Straightforward approach

- a list of items from $U$
- each item specified via a binary code
- total length for $k$ items:

$$k \log |U|$$

Disadvantages:

- requires synchronization between the parties creating the list
- error prone

# Bloom Filter
## basic technique



### Bloom Filter

a bit array $\mathcal{B}$ of length $m$, initially: all-zeroes

### Insert operation

in order to include the element $a$, we put 1 for positions $i_1, \ldots, i_k$ where

$$i_1 = H_1(a) \bmod m, \ldots, i_k = H_k(a) \bmod m$$

and $H_1 \ldots H_m$ are independent hash functions.

# Bloom Filter

## Membership check:

for $j = 1, \ldots, k$ check whether

$$\mathcal{B}[H_j(a) \bmod m] = 1$$

- If $\forall_{j \in \{1,\ldots,k\}}\ \mathcal{B}[H_j(a) \bmod m] = 1$, then potentially $a \in \mathcal{B}$.
- If $\exists_{j \in \{1,\ldots,k\}}\ \mathcal{B}[H_j(a) \bmod m] = 0$, then $a \notin \mathcal{B}$.

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| representation of $\mathcal{B}$: | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| positions with 1 for $a \in \mathcal{B}$: | 1 | | | 1 | | | 1 | | | |

# Bloom Filter

## Membership check:

for $j = 1, \ldots, k$ check whether

$$\mathcal{B}[H_j(a) \bmod m] = 1$$

- If $\forall_{j \in \{1,\ldots,k\}}\ \mathcal{B}[H_j(a) \bmod m] = 1$, then potentially $a \in \mathcal{B}$.
- If $\exists_{j \in \{1,\ldots,k\}}\ \mathcal{B}[H_j(a) \bmod m] = 0$, then $a \notin \mathcal{B}$.

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| representation of $\mathcal{B}$: | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| positions with 1 for $b \notin \mathcal{B}$: | | | | 1 | 1 | | | | | 1 |

# Bloom filter



## Advantages

- each insertion consists of operations of the form

$$\mathcal{B}(i) := 1$$

- inserting into the filter $\mathcal{B}$ **can be done in parallel**
- **no collisions**

## Disadvantages

- false positives are possible

# Bloom filter

## inserting multiple elements in parallel



| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| element *a*: | | 1 | | | | | 1 | | | | | | 1 | | | |
| element *b*: | | | | | | 1 | 1 | | | | 1 | | | | | |
| element *c*: | | | | 1 | 1 | | | | | | | | | | | 1 |
| filter state: | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |

# Low Layer Encryption

# Bloom filters - idea and problems



## Idea

- beeping model, carrier signal understood as 1
- the nodes encode information via Bloom filters
- very unlikely that two signals with the same frequency cancel each other

## Advantages

- no coordination necessary, the stations unaware of each other
- impossible to cancel a beep (unless full jamming)

## Problems

- no common clock
- the stations come and go $\Rightarrow$ configuring not much useful

# Slots

## Time slots

loosely synchronized:

- time divided into slots separated by guarding periods
- adjusting to the slots: listen, adjust the clock shift to the beeps heard

## Remarks

- a small drift of clocks not a problem
- no information exchange between the stations is necessary – adjusting on the physical level

# Sparse encoding



## $r$-sparse encoding

- a single bit encoded
- the sender and the receiver share a secret $K$
- a window consisting of $r$ slots
- with the secret $K$ two slots determined: one for 0, and one for 1

| | (a) 2-sparse | (b) 2-sparse | (c) 5-sparse |
|---|---|---|---|
| encoding bit 0: | 0 1 | 1 0 | 0 1 0 0 0 |
| encoding bit 1: | 1 0 | 0 1 | 0 0 0 0 1 |

# Setup

## Parameters

- $k$ - maximal number of stations for which we have quality guarantees

# Setup



## Parameters

- $k$ - maximal number of stations for which we have quality guarantees
- for each node $A$ a dynamic pseudonym $ID_A$ and a key $\mathcal{K}_A$ shared with the receiver

# Setup

## Parameters

- $k$ - maximal number of stations for which we have quality guarantees
- for each node $A$ a dynamic pseudonym $ID_A$ and a key $\mathcal{K}_A$ shared with the receiver
- $r > 2k$, $r$-sparse encoding to be used

# Setup



## Parameters

- $k$ - maximal number of stations for which we have quality guarantees
- for each node $A$ a dynamic pseudonym $ID_A$ and a key $\mathcal{K}_A$ shared with the receiver
- $r > 2k$, $r$-sparse encoding to be used
- selection of the $r$-sparse coding for the $i$th bit transmitted by $ID_A$ according to $\mathcal{H}(ID_A, \mathcal{K}_A, i)$

# Example encoding
## stream sent by $ID_A$



4-sparse encodings:

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | | 0 | 1 | | 0 | | 1 | | | 0 | | | 0 | 1 |

$0 \to$ beep at 3, $1 \to$ beep at 1

$0 \to$ beep at 2, $1 \to$ beep at 0

$0 \to$ beep at 3, $1 \to$ beep at 0

$0 \to$ beep at 2, $1 \to$ beep at 3

encoded message 1001:

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | ■ | | | | | ■ | | | | | ■ | | | | ■ |

Exemplary encoding of message 1001 using 4-scare encoding
"$\to$" represents choosing encoding via $\mathcal{H}_r(ID_A, \mathcal{K}_A, i)$.

# Transmission

## Preamble

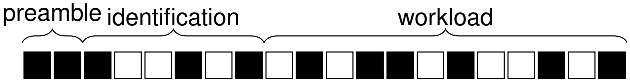
signaling the start of a transmission, consists of $r$ consecutive beeps

## Identification part

presenting the current pseudonym of the sender, $ID_{\text{sender}}$
each of the $m$ bits of sent separately encoded by $r$-sparse code

## Workload part

transmitting message $M$ where each bit is repeated $l$ times. each copy of each bit encoded separately using independently chosen $r$-sparse encoding.



# Decoding and decryption



## Transmission detection

- the sink continuously monitors the channel
- a sequence of $r$ slots of beeps treated as a preamble
- it triggers identification phase

## Identification phase

- inspect for each possible $ID_A$ the next $m \cdot r$ slots
- check if there are beeps on all positions with 1 as indicated by $ID_A$
- if yes, then a separate virtual channel opened

# Decoding and decryption



**Decoding for virtual channel for a node $A$:**

decoding each bit separately (a bit encoded $l$ times with $r$-sparse encodings

- **single bit:** decoded as $b$ if
  - beeps occur at all $l$ positions where the node $ID_A$ is supposed to beep for value $b$
  - on at least one position for the bit $1-b$ there is no beep

  $b$ appended to the decoded virtual channel for $ID_A$
- **unknown:** for both $b=0$ and $b=1$ the beeps occur at all positions where node $ID_A$ is supposed to beep for $b$ append the mark "?" to the decoded contents of the virtual channel for $ID_A$
- **failure:** other cases
  conclusion: node $ID_A$ is not transmitting, close the virtual channel corresponding to $ID_A$

# Evolution of Identifiers

**update**

After transmission the node updates its identifier as follows: $ID_A := \mathcal{H}(\mathcal{K}_A, ID_A)$.

# Properties of Low Layer Encryption

# Analysis

**Goal: Theorem**

Every transmitted message is decodable by the sink with probability at least $1 - \varepsilon$ regardless other nodes' transmission starting times.

- assumption: at any time at most $k$ stations can transmit. Moreover, $r > 2k$.
- If nobody is sending a preamble, then in each block of $r$ consecutive slots there is at least one empty slot.

**Lemma**

Consider a transmission of a single bit using $r$-sparse coding. If no other node transmits its preamble, then the probability that the bit is **not** ambiguous is at least

$$(1 - 1/r)^{k-1} > (1 - 1/r)^{r/2} \geq \frac{1}{2}$$

So after repeating $l$ times . . .

**Theorem**

For

$$m = 2k + \log(|\mathcal{A}|) + \lceil \log \tfrac{1}{\delta} \rceil$$

probability that the decoding procedure returns a pseudonym of a node $A'$ that has **not** transmitted the preamble at the considered time $t$

**is smaller than** $\delta$,

provided that $A'$ either has not transmitted its preamble at time $t'$ where $|t' - t| \leq r$.

# Data confidentiality

## Dynamic mixing

- the bits of the ciphertexts are mixed together – additional problem for cryptanalysis
- if the number of “?” bits is limited, trial decryptions with exhaustive search may recover the plaintexts anyway

# Unlinkability



## ID's

- the ID's are transmitted in clear
- however each ID for a single transmission
- linking ID's requires knowledge of the secret key

# Conclusions

one can start an ad hoc shared communication channel with encrypted data with no coordination between stations

Bloom filters can serve as a direct method for encoding information – instead of multi-layer wrapping and encoding

silence is also a message – *green computing* paradigm

the proposed encoding method is just an example – probably a lot of optimization possible

# Thanks for your attention!

## Contact data

1. Miroslaw.Kutylowski@pwr.edu.pl
2. http://kutylowski.im.pwr.edu.pl