

# Secure Ranging (with focus on 802.15.4z)

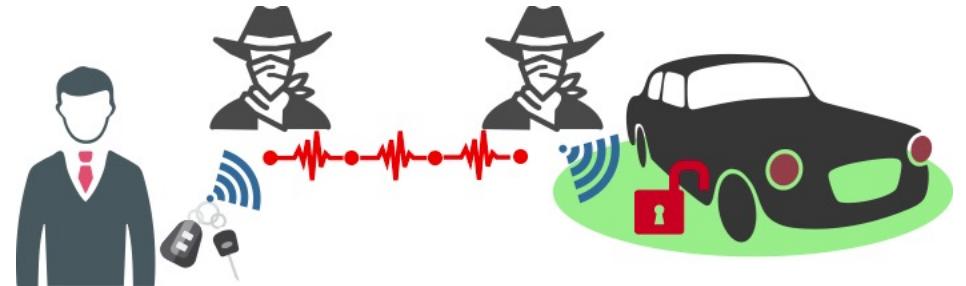
Srdjan Čapkun

**ETH** zürich



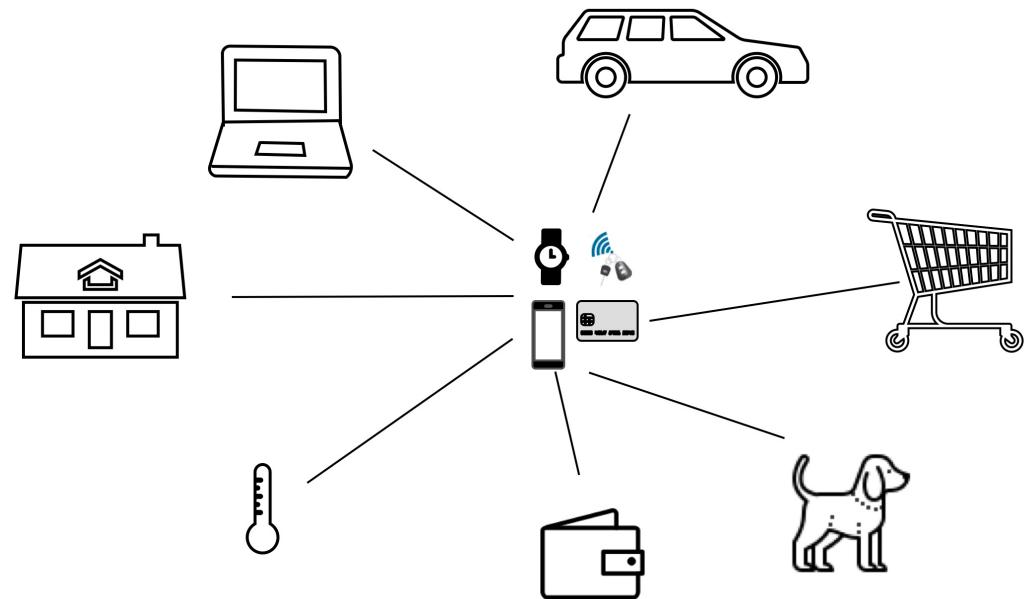
# Secure Distance Measurement

Problem: prevent manipulation of measured distance.

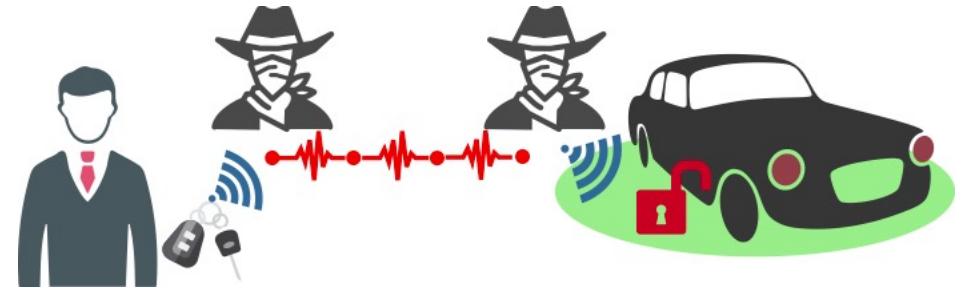


## Broader Application Space:

- Access Control (e.g., PKES)
- Payments
- Tracking
- Automation



# Secure Ranging



- We need ranging that is  
**(provably) secure from logical and physical layer attacks**
- Security should not depend on the context or attacker abilities
- Should bind distance to identity
- Main focus: (also in 802.15.4z) prevention of distance reduction attacks

# (Secure) Ranging Technologies

## NON-Time-of-Flight:

- RSSI (e.g., WiFi, Bluetooth, 802.15.4, NFC / RFID ) – Insecure
- Phase (multi-carrier) (e.g., Atmel AT86RF233) – Insecure
- FMCW (Frequency-Modulated Continuous-Wave) – Insecure

## Time-of-Flight:

- Chirp Spread Spectrum (802.15.4 CSS, ISO/IEC 24730-5) – Insecure
- Ultra Wide Band (UWB) IEEE 802.15.4z – proposed for secure ranging  
[https://standards.ieee.org/standard/802\\_15\\_4z-2020.html](https://standards.ieee.org/standard/802_15_4z-2020.html)
- WiFi IEEE 802.11az, 5G - first efforts to secure OFDM-based ranging

# IEEE 802.15.4z: Two Modes

Low Rate Pulse (**LRP**) mode:

- PKES: Passive Keyless Entry and Start (VW, Daimler, ...)
- Microchip (ATA8352/50), Renesas, 3dB
- Open, simple security design



High Rate Pulse (**HRP**) mode:

- PKES (BMW), iPhone/Airtags, Samsung/Smartags
- Apple (U1), NXP (SR040/150), Qorvo (DWM3000), 3dB (HRP+LRP, 2022)
- Closed/Proprietary security design



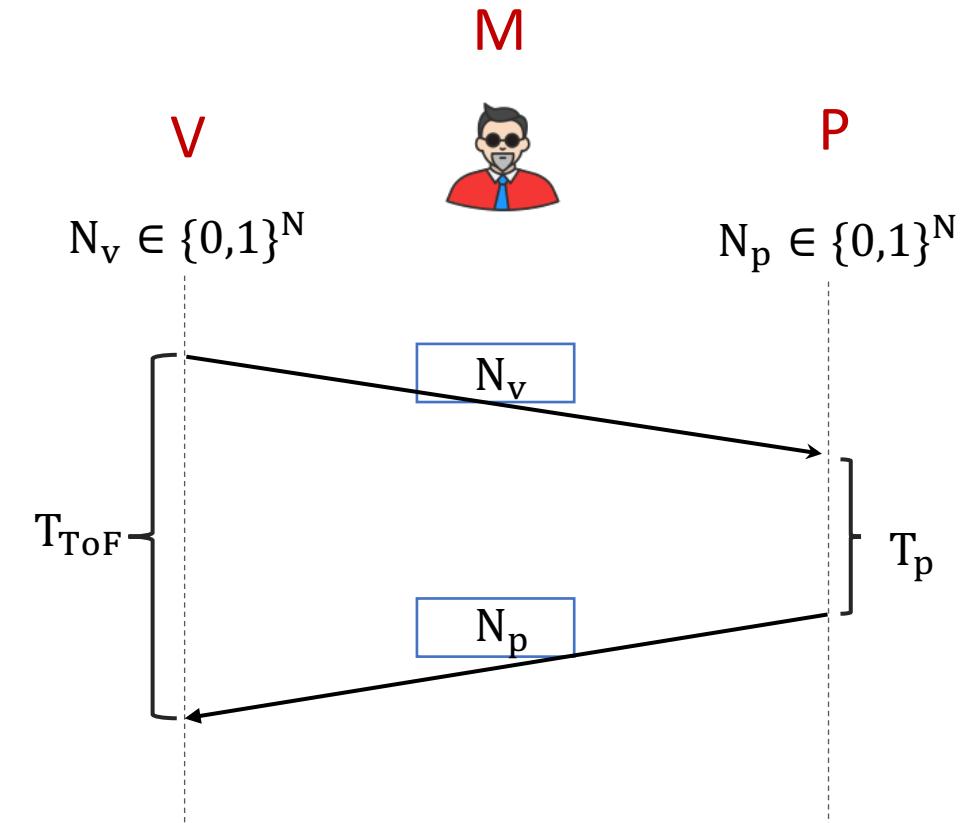
# This Lecture

- Main Challenges in Securing (UWB) Ranging:
  - Logical and Physical Layer
  - HRP vs LRP Security Design
- Practical Distance Reduction Attacks on **HRP** ranging:  
iPhone/Airtags (U1), NXP (SR040/150), Qorvo (DWM3000)

# Main Challenges in Securing (UWB) Ranging

# Secure Ranging Background: Logical Layer

- Distance reduction prevented using challenge-response protocols with time measurements (i.e., distance-bounding)
- V and P share a secret key unknown to M
- V and P exchange messages that are (in part) unpredictable to M



# Secure Ranging Background: Logical Layer

- STS cannot be predicted by M  
*(before it is sent)*

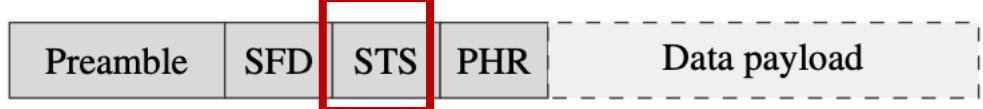
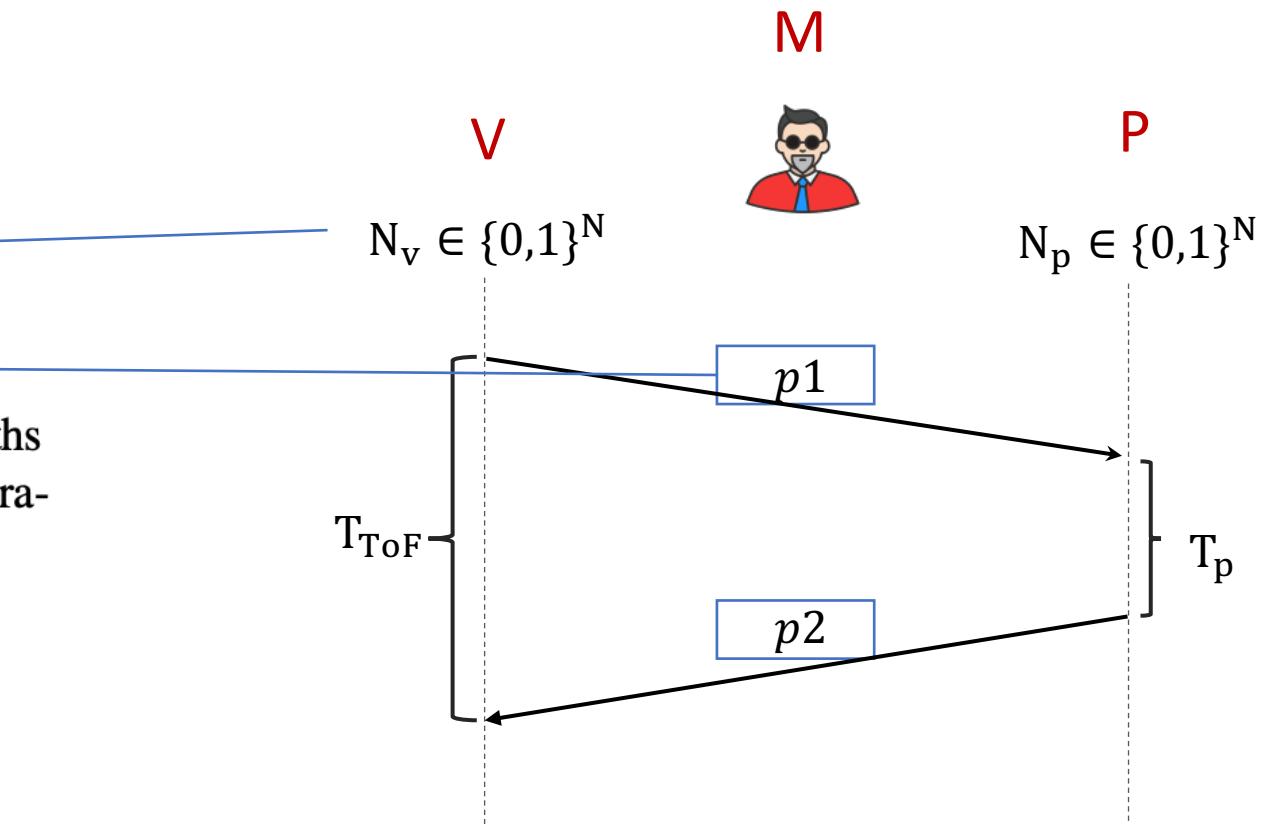


Figure 4: Example format of an HRP packet [4]. The lengths of the different parts and their order depend on the configuration.

Physical Layer header (PHR)  
Start-of-frame delimiter (SFD)  
Scrambled Timestamp Sequence (STS)

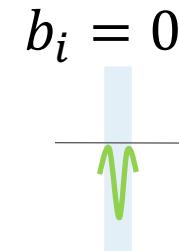
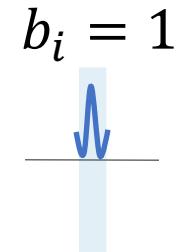


This design prevents  
logical layer attacks.

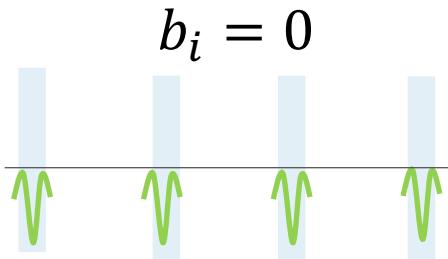
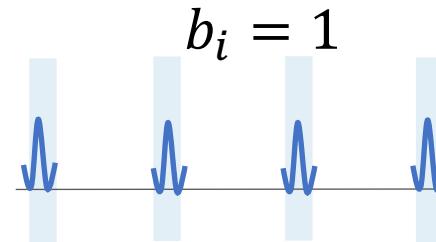
But ranging also needs  
to be protected against **physical**  
**layer attacks.**

# UWB Symbols (example)

Bit represented by one UWB pulse



Bit represented by multiple  
UWB pulses.



Example Demodulation

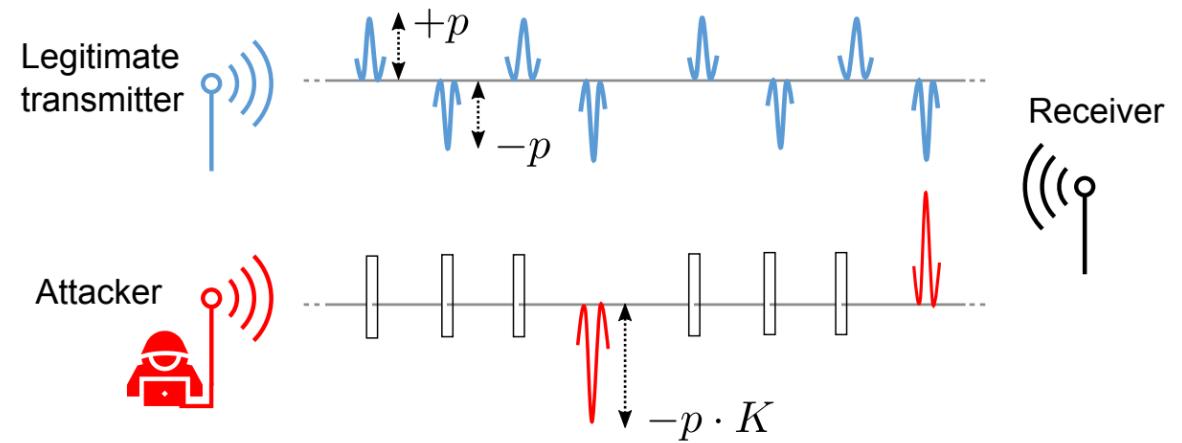
$$\sum \left( \text{blue waveform} + \text{blue waveform} + \text{blue waveform} + \text{blue waveform} \right) \longrightarrow b_i = 1$$

$$\sum \left( \text{green waveform} + \text{green waveform} + \text{green waveform} + \text{green waveform} \right) \longrightarrow b_i = 0$$

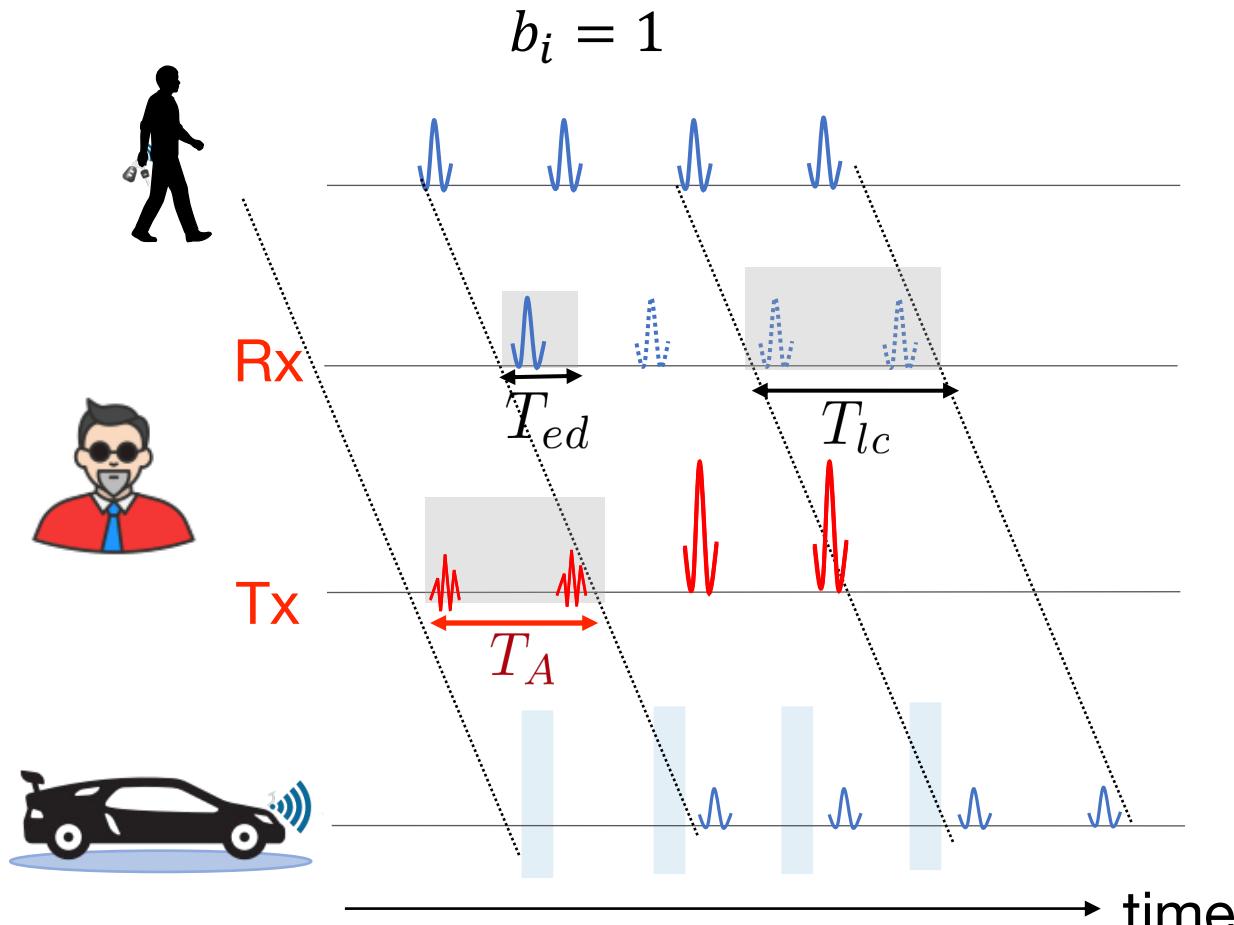
Longer symbols increase the range and robustness to noise and multipath.

# UWB Security: Attacks on ToA Estimation

- Early Detect / Late Commit
- Cicada
- Cicada+
- Adaptive Injection
- Ghost Peak
- ...?



# Problem: Distance **Shortening** Attacks



Early-detect/late-commit (ED/LC) Attack

Steps to insert an earlier path

- Send noise in time  $T_A$
- Learn shape of the symbol in time  $T_{ed}$
- Commit correct symbol in time  $T_{lc}$

$$\Sigma ( \text{noise} + \text{noise} + \text{noise} + \text{noise} ) \rightarrow b_i = 1 \text{ Correct Bit}$$

# Problem: Distance **Shortening** Attacks

$$b_i = 1$$

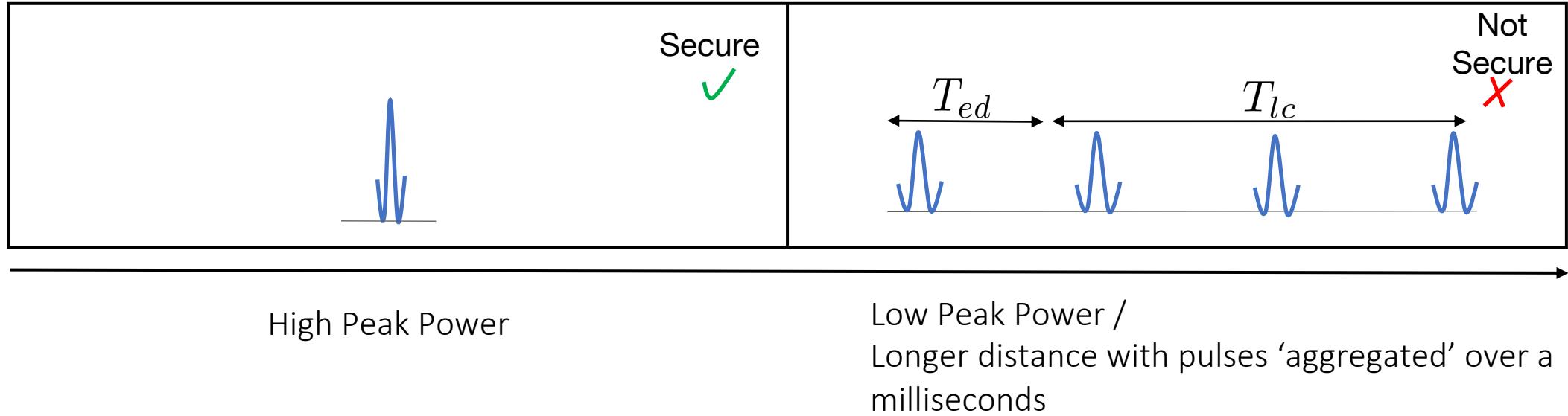


To prevent distance shortening attacks we need to encode the message such that the receiver can check if the Time of Arrival estimation was manipulated

(Message Time of Arrival Codes)

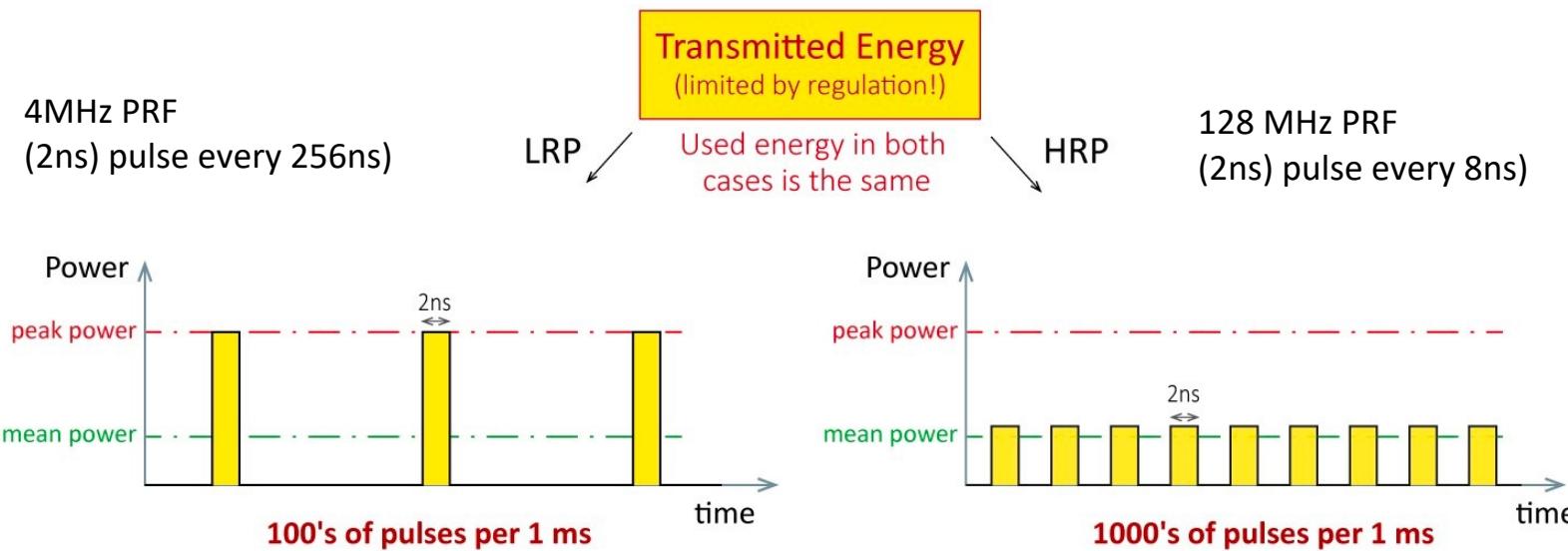
Early-detect/late-commit (ED/LC) Attack

# Performance/Security Tradeoff?



802.15.4z was introduced in 2020 to enable secure ranging while providing both performance and robustness to attacks. Does 4z protect Time of Arrival?

# IEEE 802.15.4z: LRP and HRP mode



- 1-2ns wide pulses
- > 500MHz bandwidth
- Frequencies 6-9GHz
- Power spectral density  $< -41.3 \text{ dBm/MHz}$  avg. over 1ms
- Power in 50 MHz bandwidth around peak frequency limited to 0 dBm

## Low Rate Pulse (LRP)

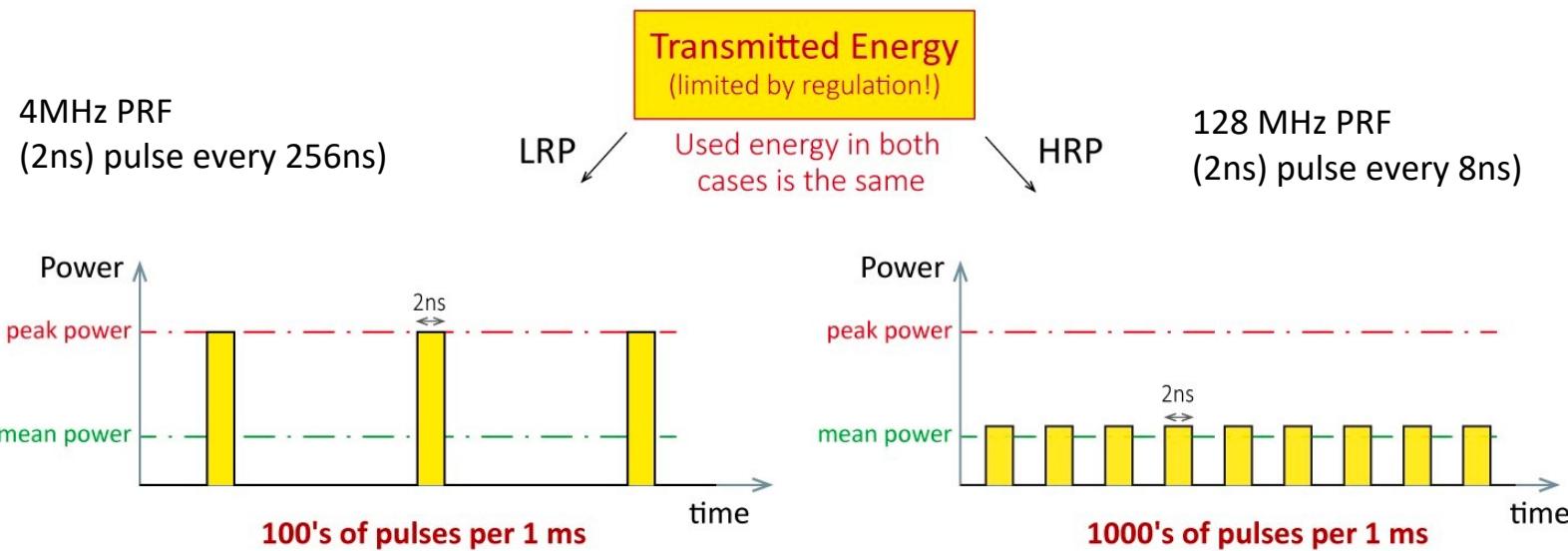
- Modulation: Binary Frequency Shift Keying (BFSK)
- Single ranging requires few (e.g. 100) pulses
- **Bit represented by single or multiple pulses**
- Non-coherent
- More power efficient than HRP

## High Rate Pulse (HRP)

- Modulation: BPSK (secure ranging) and BPM+BPSK (data)
- Single ranging requires 1000s to 10000s pulses
- **Each bit represented by multiple pulses**
- **Cannot use single pulse mode (energy too low)**
- Coherent

# Security of 802.15.4z LRP Mode

# LRP vs HRP



- 1-2ns wide pulses
- > 500MHz bandwidth
- Frequencies 6-9GHz
- Power spectral density  $< -41.3 \text{ dBm/MHz}$  avg. over 1ms
- Power in 50 MHz bandwidth around peak frequency limited to 0 dBm

## Low Rate Pulse (LRP)

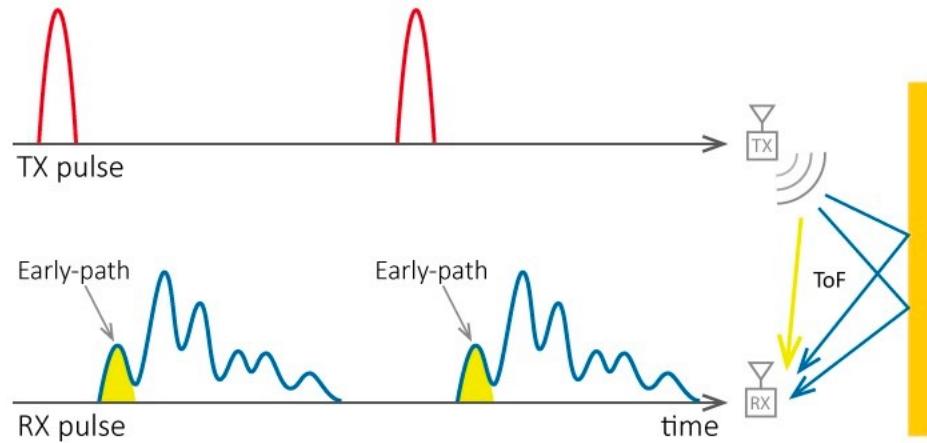
- Modulation: Binary Frequency Shift Keying (BFSK)
- Single ranging requires few (e.g. 100) pulses
- Bit represented by single or multiple pulses
- Non-coherent
- More power efficient than HRP

## High Rate Pulse (HRP)

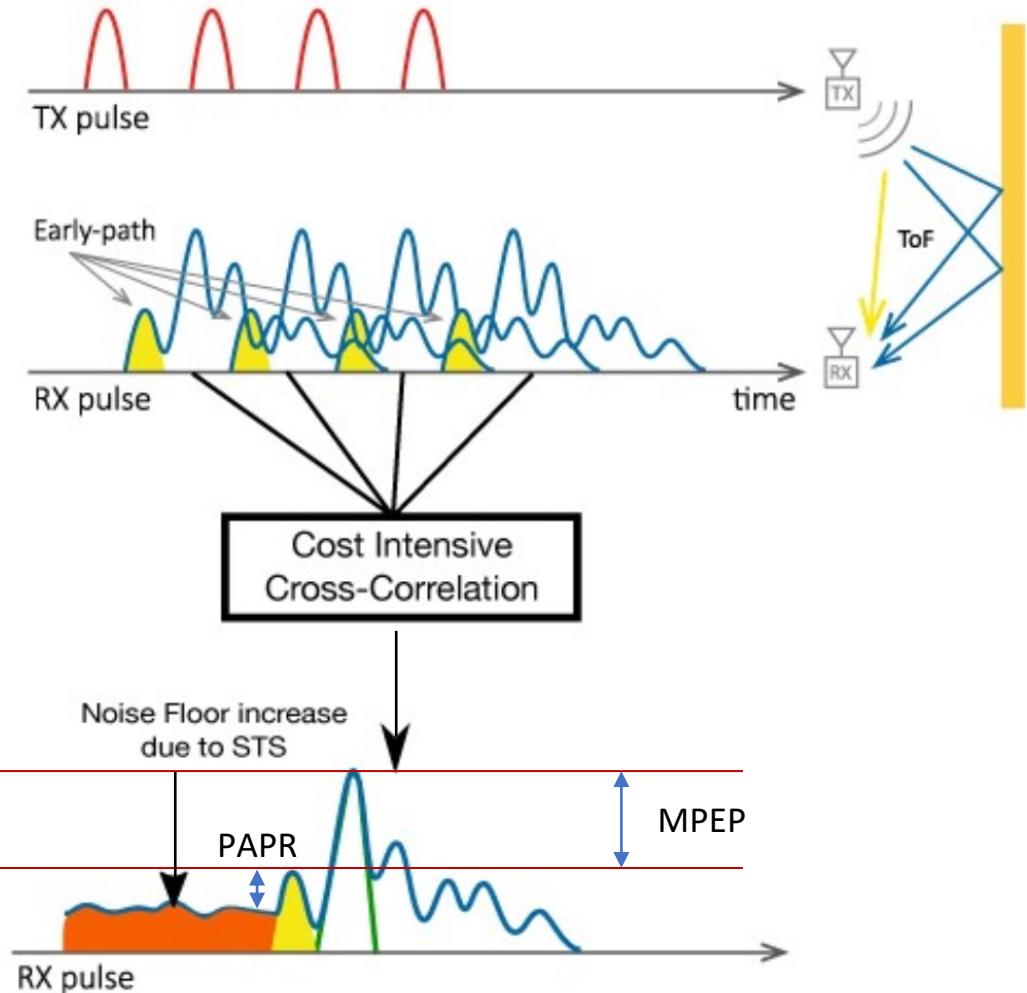
- Modulation: BPSK (secure ranging) and BPM+BPSK (data)
- Single ranging requires 1000s to 10000s pulses
- Each bit represented by multiple pulses
- Cannot use single pulse mode (energy too low)
- Coherent

# LRP vs HRP

LRP 4MHz PRF (1 (2ns) pulse every 256ns)



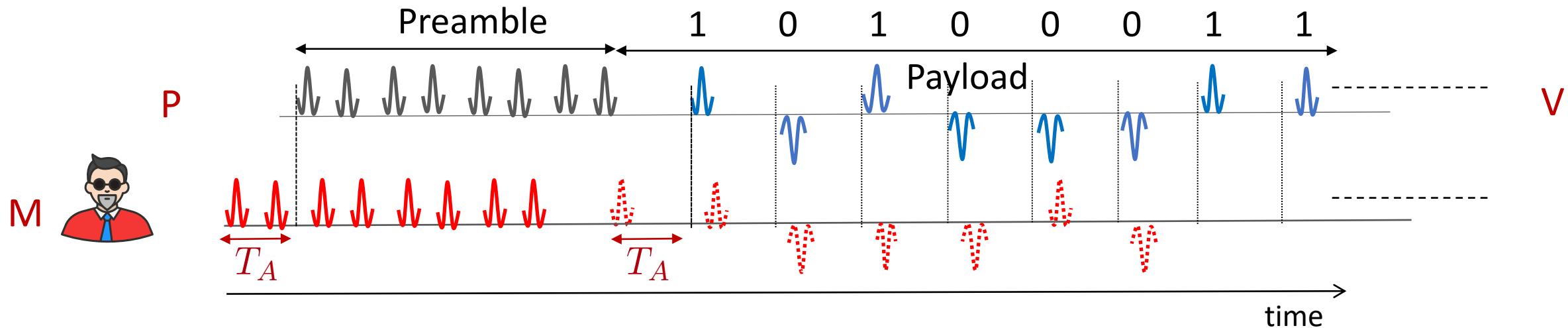
HRP STS 128 MHz PRF (1 (2ns) pulse every 8ns)



In LRP pulses are stronger than in HRP

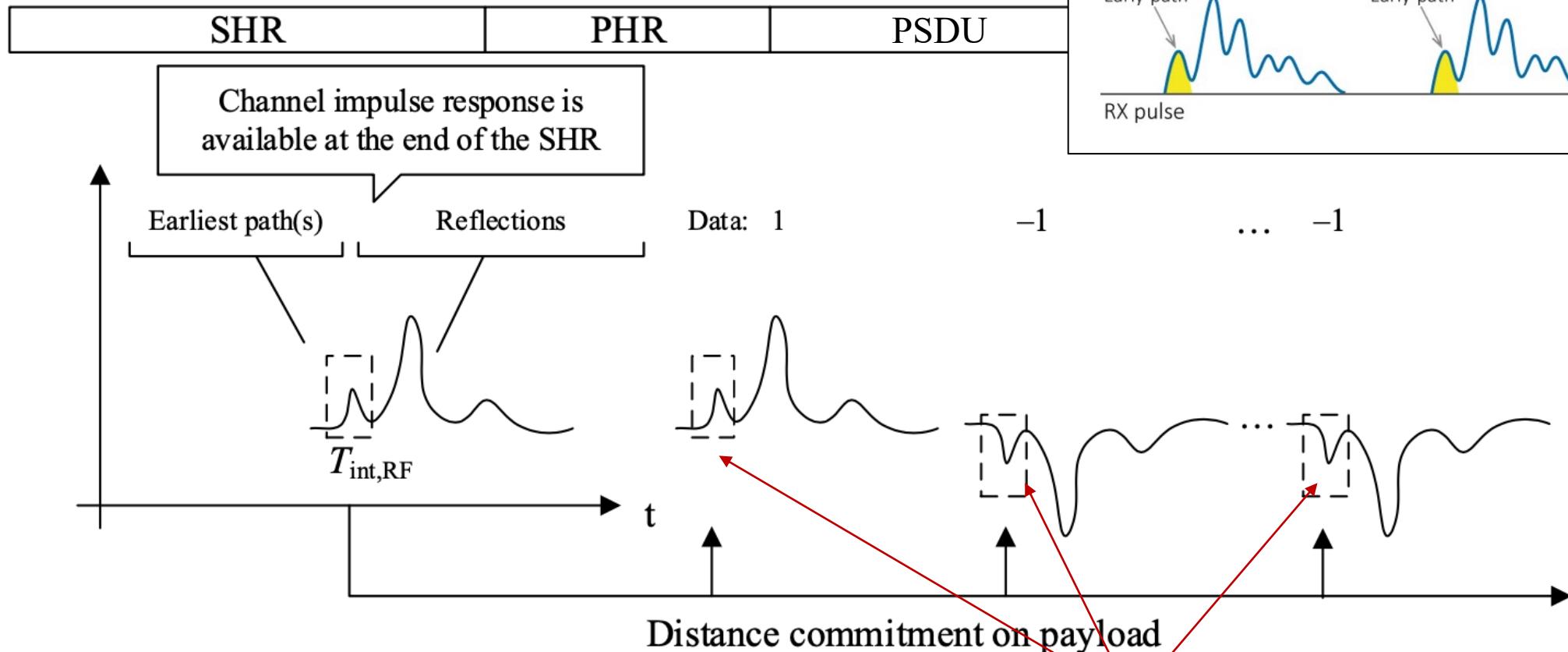
In LRP pulses are spaced more apart and less susceptible to inter-pulse-interference  
=> Less noise => better reliability and security

# LRP Security: Single-Pulse Per Bit Secure Ranging

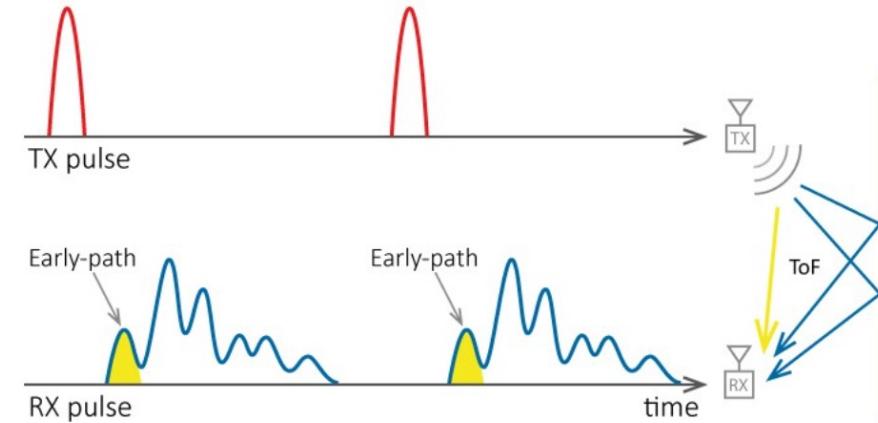


- Step 1. ToA measured over a public preamble (distance commitment)
- Step 2. ToA verified over random payload bits (receiver decodes each pulse into one bit), attacker needs to guess every pulse/bit.

# LRP Security

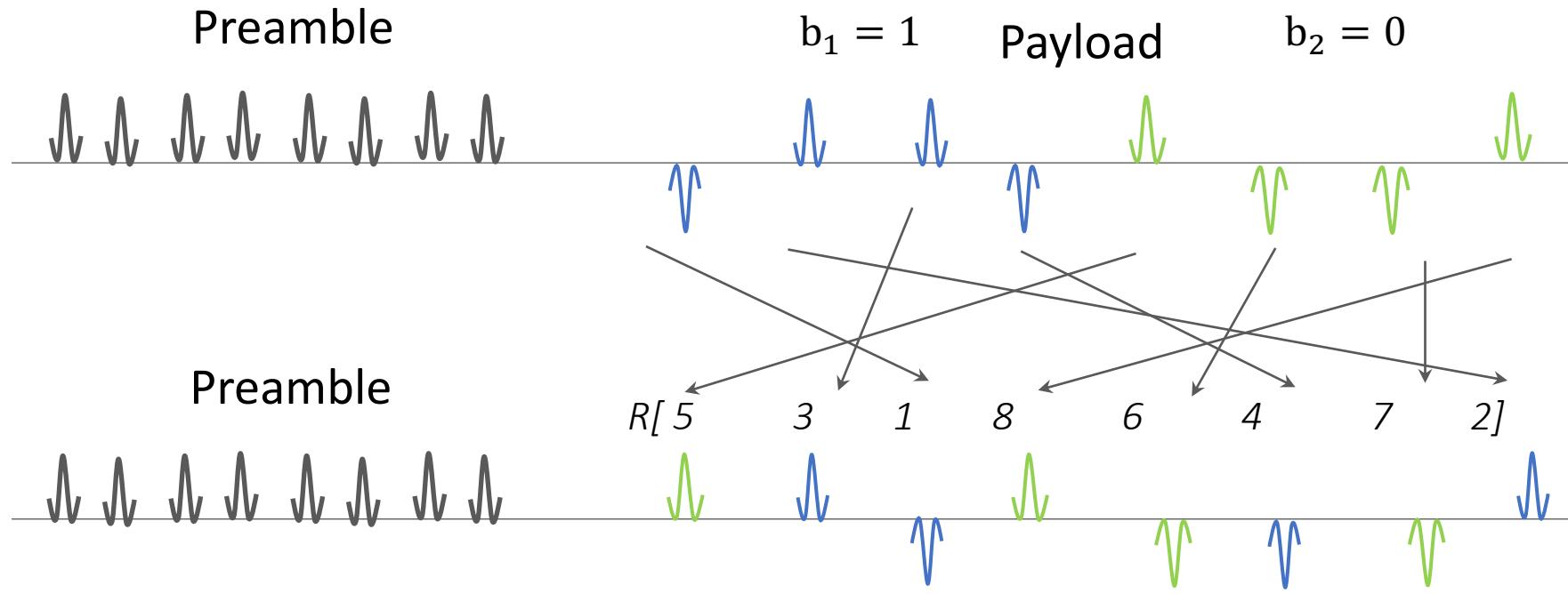


LRP



Attacker needs to create a correct pulse polarity at each of these point in time. i.e., attacker needs to guess.

# LRP Security: Multi-Pulse Per Bit Secure Ranging

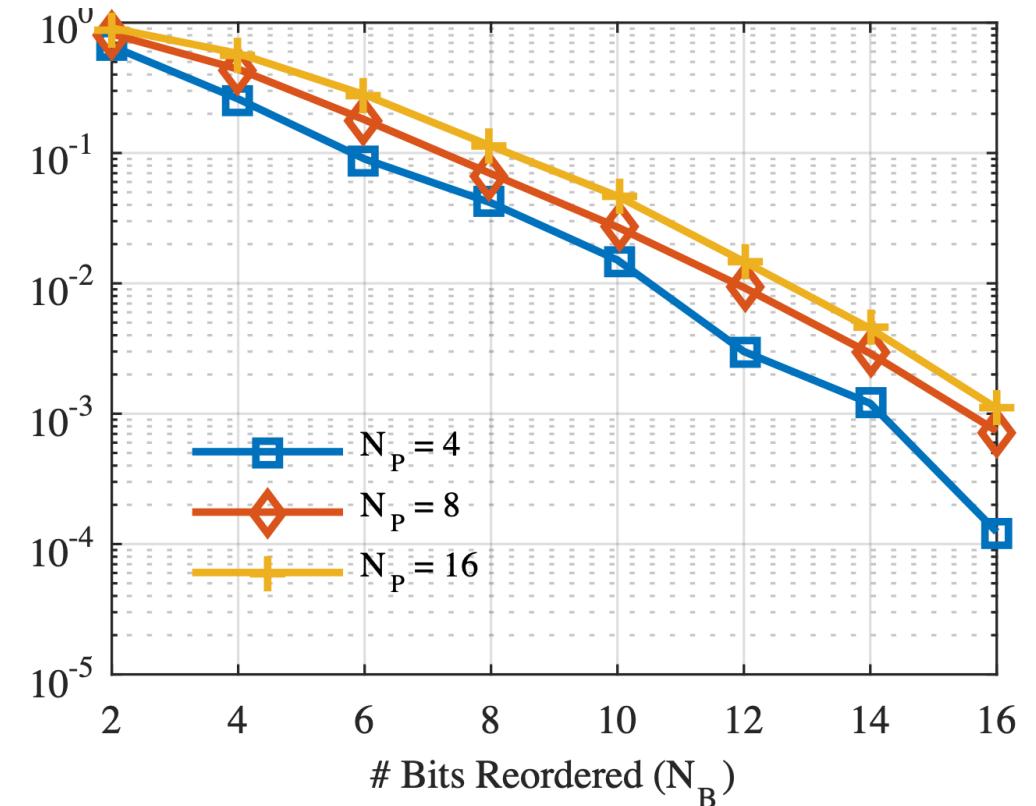


- For cases where individual pulses not visible to receiver
- UWB with Pulse Reordering (UWB-PR):  
shape, start/end time of symbols is unpredictable => robustness to attacks

# LRP Security: Probability of Attacker Success

- Single-Pulse per bit  
Probability of attacker's success:  $2^{-|\text{payload}|}$  ( $2^{-32}, 2^{-64}, 2^{-128}, \dots$ )

- Multi-Pulse per bit  
(LRP with Pulse Reordering) →
- *LRP is open, simple and comes with explicit security guarantees.*

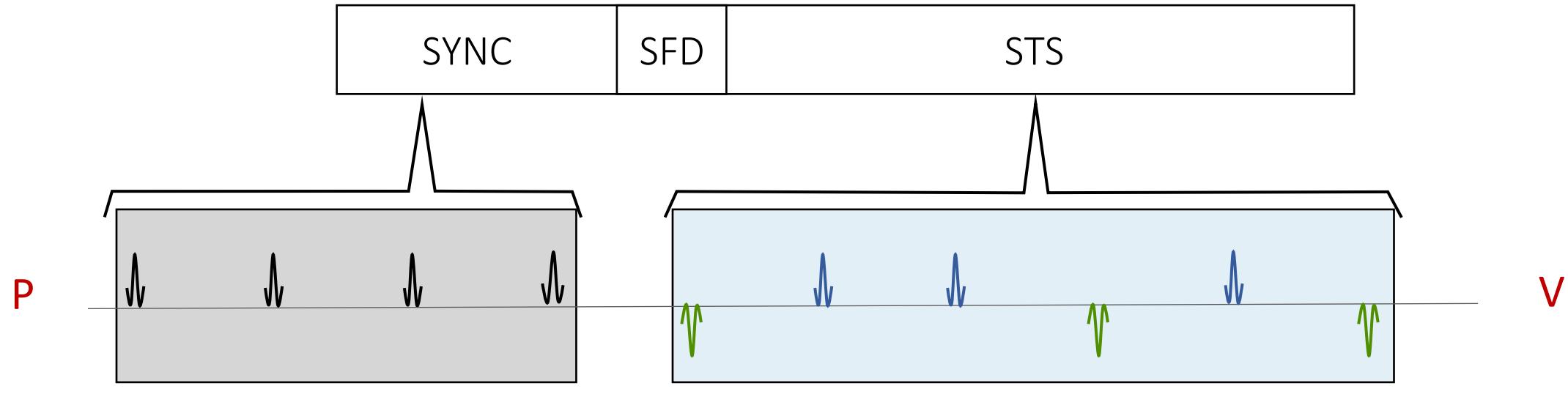


=>

In LRP receivers can decode individual pulses  
*(|secure payload| = 32,64,128,... pulses; 1 pulse per bit)*

In HRP receivers **cannot** decode individual pulses  
*(STS is 4092 pulses long, bits(?)*)

# HRP Security: Scrambled Timestamp Sequence



public preamble

Scrambled Timestamp Sequence: random pulses  
(known to P and V, hard to predict for M)

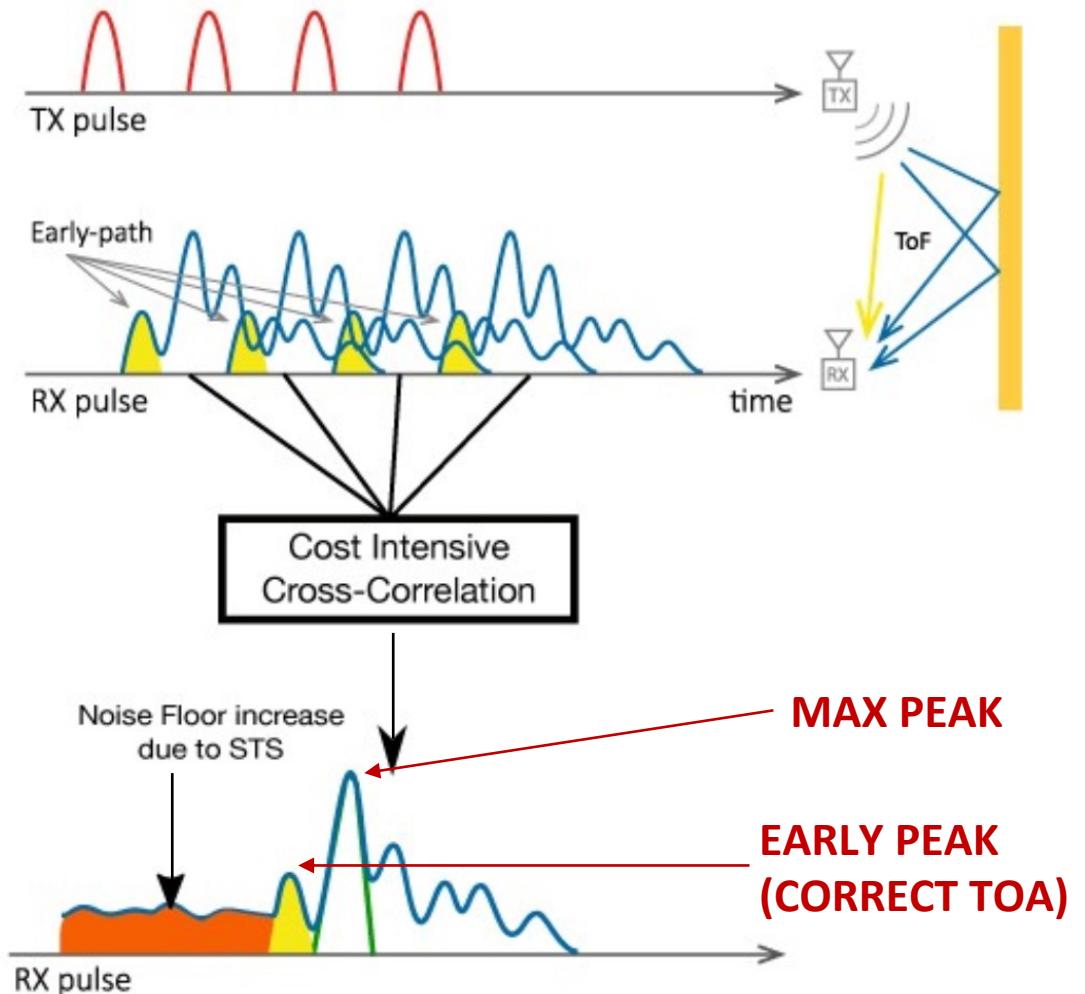
Receiver correlates the received STS with the local STS copy.  
Autocorrelation peak determines the Time of Arrival.

# HRP Security: Time of Arrival Estimation

- Max autocorrelation peak is not the correct ToA
- Receivers look for Early Path (leading edge / correct ToA)
- How to identify Early Path?  
How weaker than Max Peak and stronger than Avg Noise should it be?

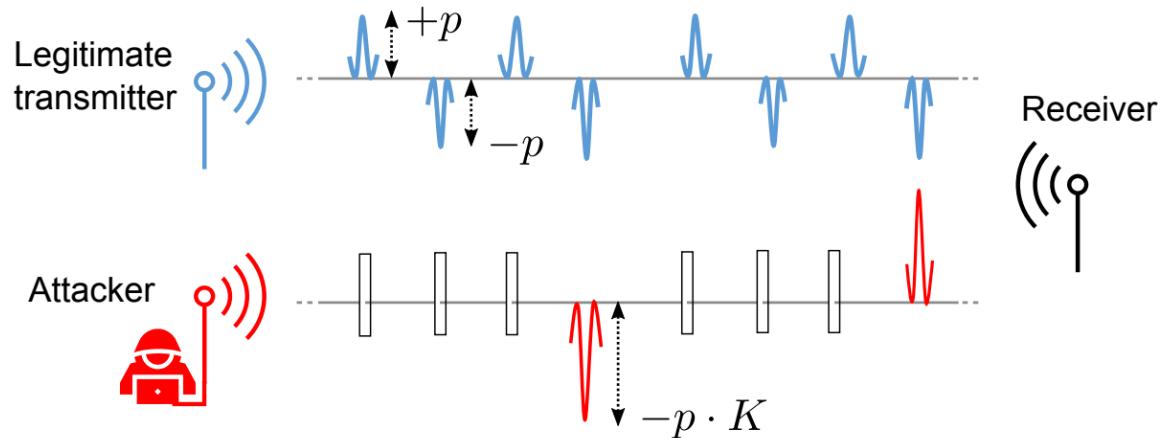
*Reliability / Security Tradeoff?*

HRP STS



# HRP Security: Attacks on ToA Estimation

- Early Detect / Late Commit
- Cicada
- Cicada+
- Adaptive Injection
- Ghost Peak
- ...?



**Goal of the attacker:**  
**create a FAKE EARLY PEAK despite not knowing STS**  
**(remember: receiver doesn't 'see' individual pulses)**

# HRP Security: Attacks on ToA Estimation

- Ghost-Peak Attack shows that physical-layer attacks on HRP are a real threat.
- Apple/NXP/Qorvo use hidden ToA estimation algorithms making it hard to evaluate if their HRP ranging systems are truly secure. These algorithms should be public and comprehensively evaluated.
- Like with ciphers/signatures/MACs security should rely only on the unpredictability of the STS, not on the secrecy of the algorithms.

# HRP Security: How To Move Forward?

- Using **autocorrelation** for secure ToA estimation is not well understood.
- Currently, we cannot fully answer the following questions:  
*“If STS is 4092 pulses long, what is the probability that the attacker creates a fake early path (i.e., reduces the distance)?”*  
*How many bits of security does a 4092 pulse long STS provide?*
- Initial analysis can be found in:

P. Leu, M. Singh, M. Roeschlin, K. Paterson, S. Čapkun  
*Message Time of Arrival Codes: A Fundamental Primitive for Secure Distance Measurement*, in IEEE Symposium on Security and Privacy (S&P), 2020

M. Singh, M. Röschlín, E. Zalzala, P. Leu, S. Čapkun  
*Security Analysis of IEEE 802.15.4z/HRP UWB Time-of-Flight*  
In ACM WiSec 2021

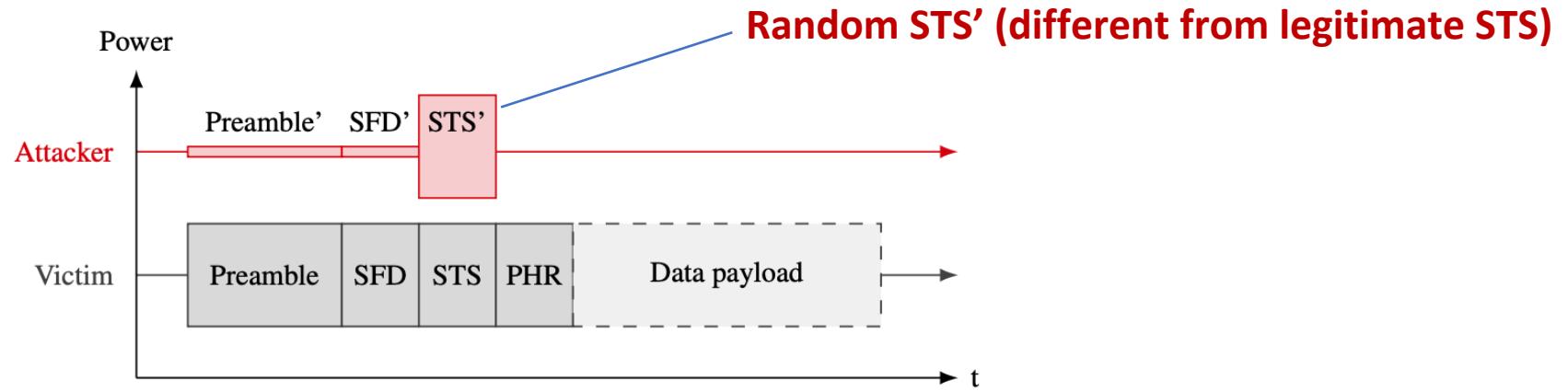
# Ghost-Peak: Practical Distance Reduction Attacks on HRP UWB Ranging

Patrick Leu<sup>1,\*</sup>, Giovanni Camurati<sup>1,\*</sup>, Alexander Heinrich<sup>2</sup>, Marc Roeschlin<sup>1</sup>, Claudio Anliker<sup>1</sup>,  
Mathias Hollick<sup>2</sup>, Srdjan Čapkun<sup>1</sup>, and Jiska Classen<sup>2</sup>



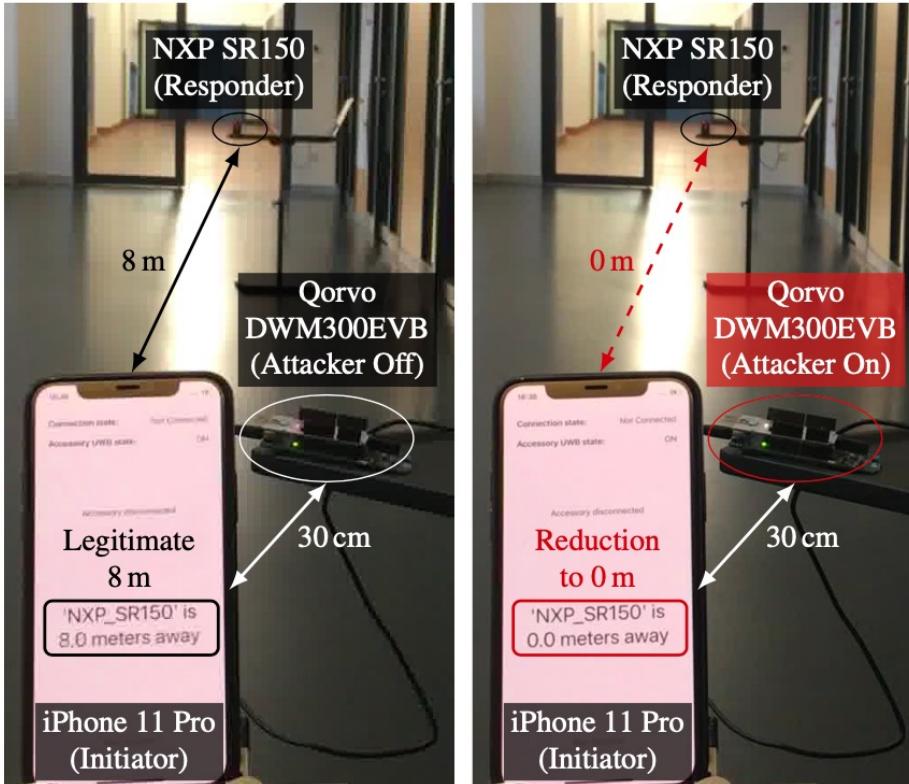
[www.securepositioning.com/ghost-peak](http://www.securepositioning.com/ghost-peak)

# HRP Security: Ghost Peak Attacks

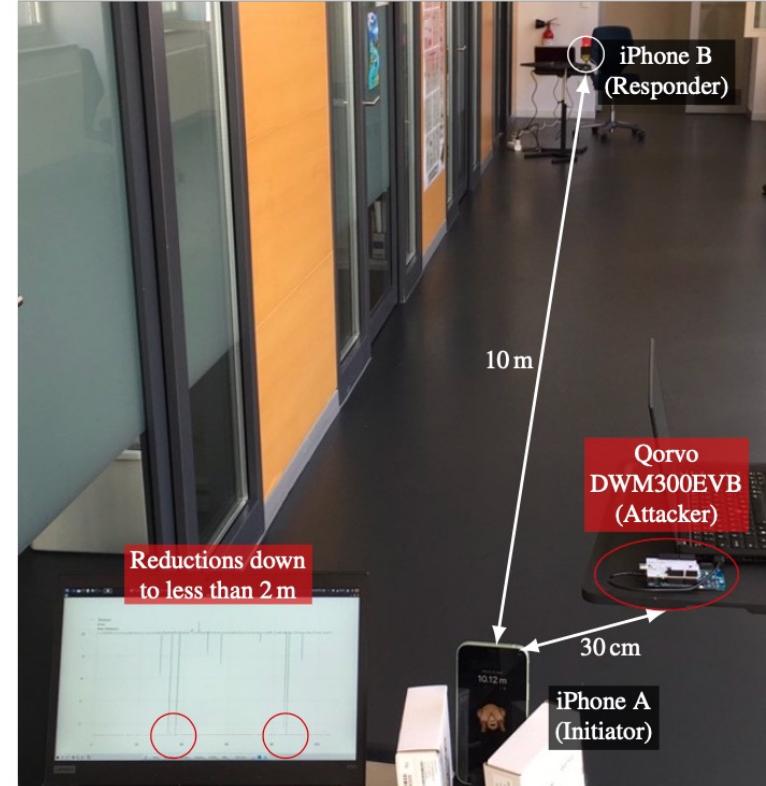


- **Main idea:** inject a packet in-synch ( $\mu$ s) with legitimate transmission, but with varying power of Preamble', SFD', **STS'** (randomly generated)
- **Attack assumptions:**
  - No knowledge of keys shared between victim devices
  - No ability to predict scrambled timing sequence (STS)
- **Attack setup:** inexpensive (USD 65) commercial off-the-shelf device

# HRP Security: Ghost Peak Attacks



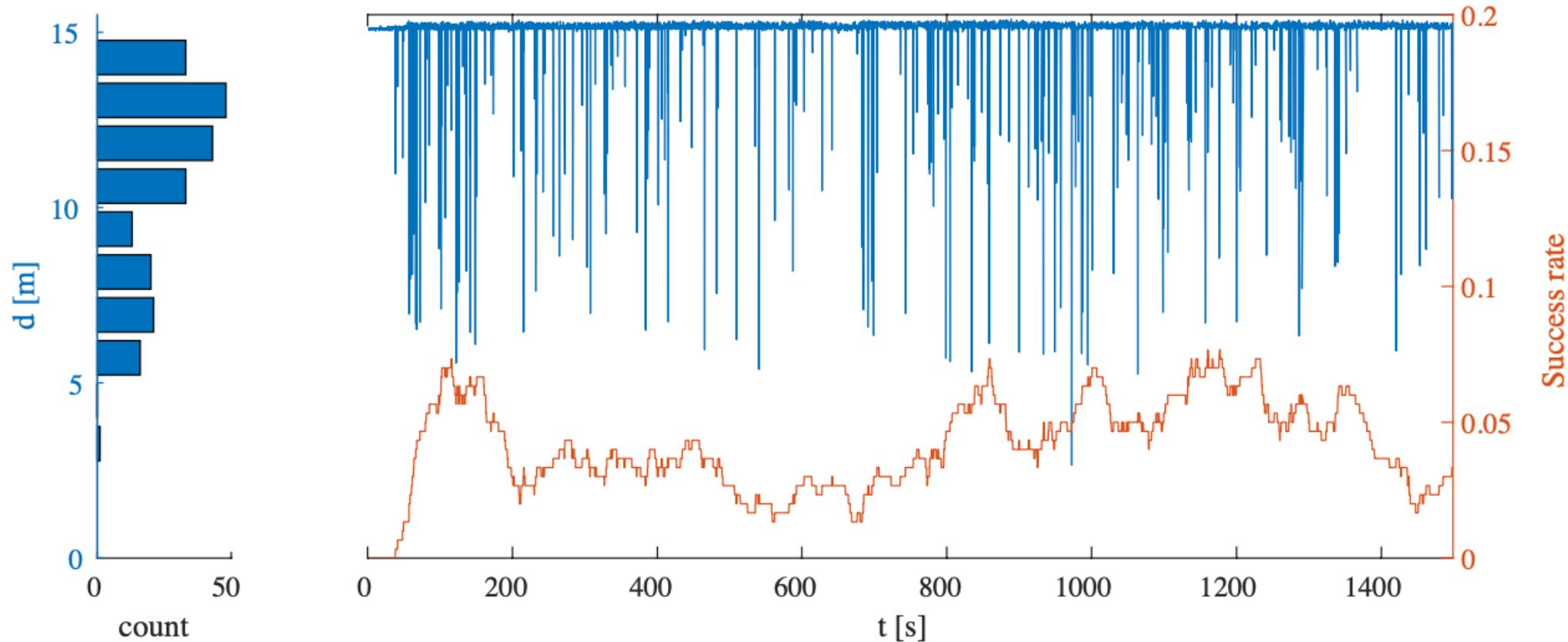
(a) iPhone (initiator, victim) + NXP SR150 (responder): reduction from 8 m to 0 m visible on the screen of the iPhone.



(b) iPhone + iPhone: reduction from 10 m to less than 2 m visible in the raw measurements logs.

Figure 9: Two concrete examples of distance reduction attacks.

# HRP Security: Ghost Peak Attacks



# HRP Security: Ghost Peak Attacks

Table 2: Overview of attack scenarios and results.

Scenario	Primary Victim (V1)	Secondary Victim (V2)	Maximum Reduction	Success Rate
Handoff Music	iPhone (Apple U1)	HomePod mini (Apple U1)	9.01 m	2.10 %
Nearby Interaction	iPhone (Apple U1)	iPhone (Apple U1)	12.45 m	4.08 %
AirTag	AirTag (Apple U1)	iPhone (Apple U1)	9.09 m	4.25 %
NXP Initiator	iPhone (Apple U1)	Tag (NXP SR040)	4.80 m	1.87 %
NXP Responder	iPhone (Apple U1)	Tag (NXP SR150)	9.68 m	2.15 %
Qorvo	iPhone (Apple U1)	Tag (Qorvo DWM3000)	8.13 m	3.09 %

# HRP Security: Ghost Peak Attacks

The screenshot shows a web browser displaying the NXP website for the Trimension SR040. The URL in the address bar is [nxp.com/products/wireless/secure-ultra-wideband-uwb/trimension-sr040-reliableuwb-solution-for-iot:SR040](https://nxp.com/products/wireless/secure-ultra-wideband-uwb/trimension-sr040-reliableuwb-solution-for-iot:SR040). The page features the NXP logo and a navigation menu with links to PRODUCTS, APPLICATIONS, DESIGN, SUPPORT, and COMPANY. Below the menu, a breadcrumb trail shows the path: Home / Wireless / Secure Ultra-Wideband (UWB) / Trimension SR040. The main title is "Trimension™ SR040: Reliable UWB Solution for IoT". A horizontal navigation bar below the title includes links for OVERVIEW, DOCUMENTATION, TOOLS & SOFTWARE, BUY/PARAMETRICS, and PACKAGE/QU. On the left, there's a "Jump To" section with links to Overview & Features, Development Boards and Designs, and Target Applications. The central content area starts with an "Overview" section, which describes the device as optimized for low-power operation and designed for use in coin cell battery-operated IoT devices like UWB trackers and tags. It mentions the integrated FiRa™ MAC by NXP and its compatibility with Bluetooth® Low Energy. The right side of the page lists several features, each preceded by a red bullet point.

**Trimension™ SR040: Reliable UWB Solution for IoT**

OVERVIEW DOCUMENTATION TOOLS & SOFTWARE BUY/PARAMETRICS PACKAGE/QU

**Jump To**

- Overview & Features
- Development Boards and Designs
- Target Applications

**Overview**

Optimized for low-power operation and reducing the need for external components, Trimension SR040 is designed for use in coin cell battery-operated IoT devices, such as UWB trackers and tags. The integrated FiRa™ MAC by NXP supports interoperability and fast time-to-market. Trimension SR040 can be integrated with Bluetooth® Low Energy or other connectivity controllers in one device. Being used as a Time Difference of Arrival (TDoA) tag, they can send blink packets only.

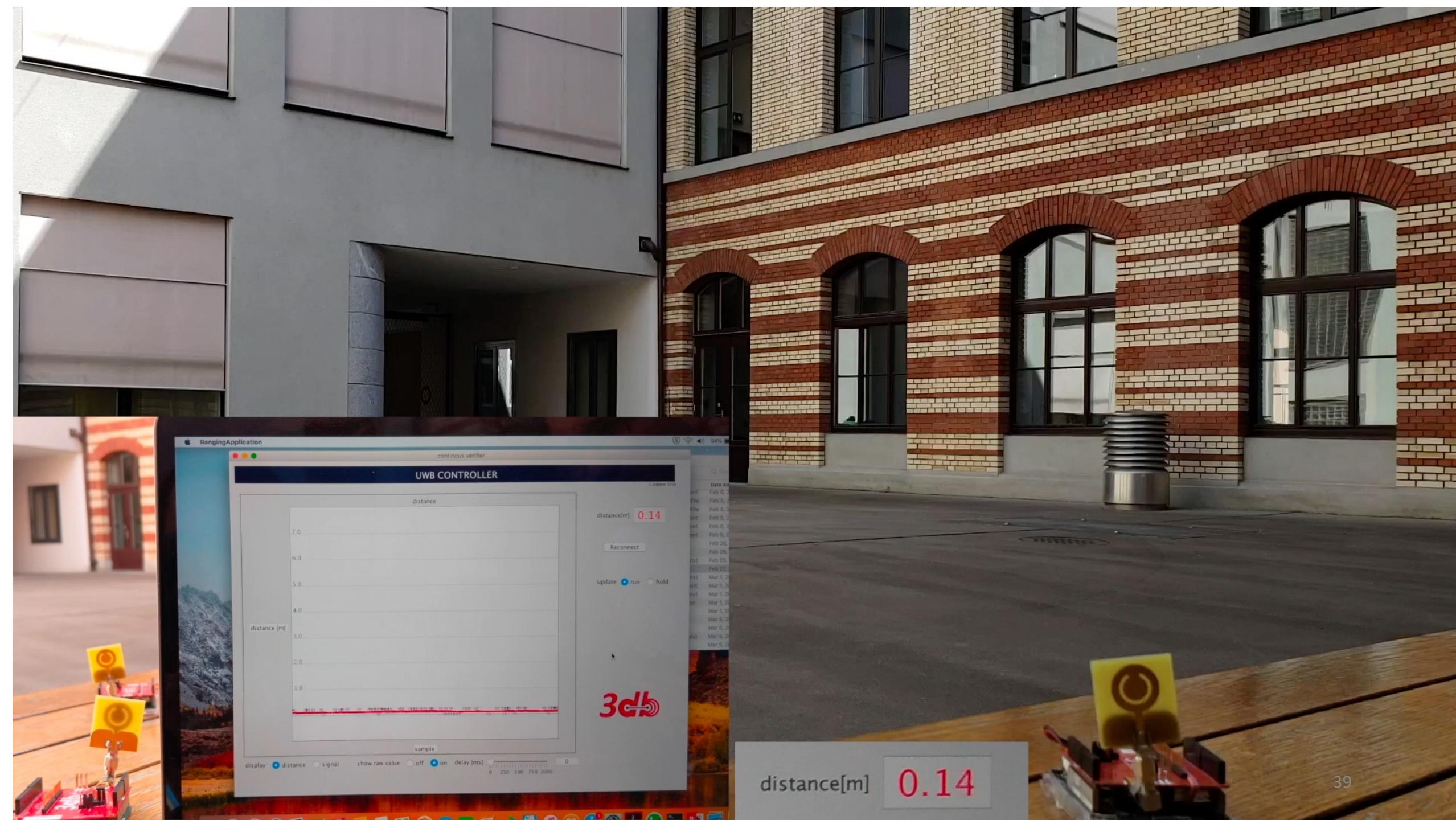
**Features**

- Specialized pa
- Download-free in embedded fl
- Optimized low-
- Integrated Tx/F
- In accordance
- IEEE 802.15.4

# Conclusion

# Conclusion

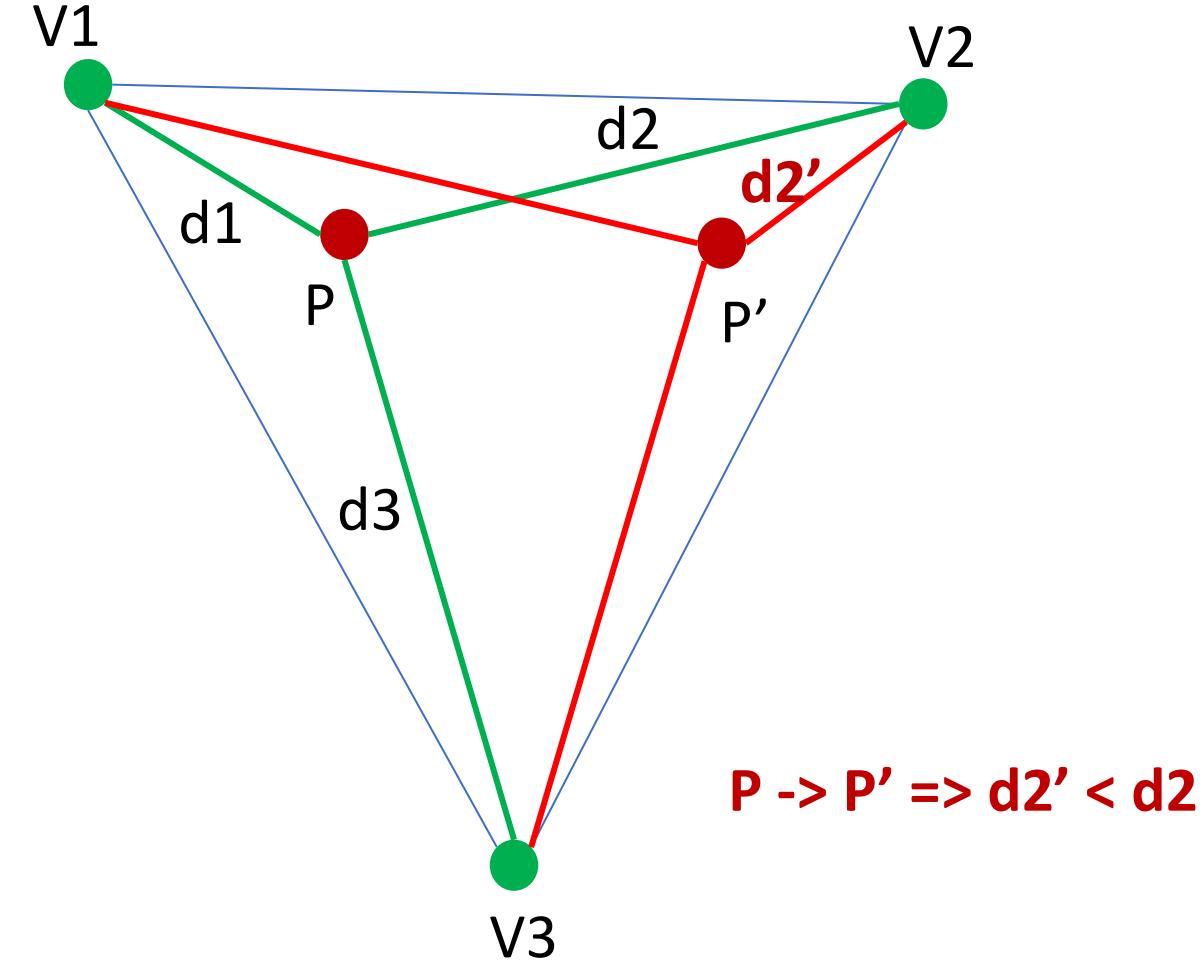
- UWB provides a strong basis for secure ranging. No other radio technology is likely to provide stronger guarantees.
- 802.15.4z **HRP**:
  - HRP security algorithms used by Apple/NXP/Qorvo are not public
  - Autocorrelation seems used for (secure) ToA estimation: this is not well understood
  - More transparency needed: Ghost Peak attack show that physical layer attacks on HRP are possible and need to be addressed.
- 802.15.4z **LRP**:
  - Security design is open and simple.
  - Does not seem to suffer from the same issues like HRP.



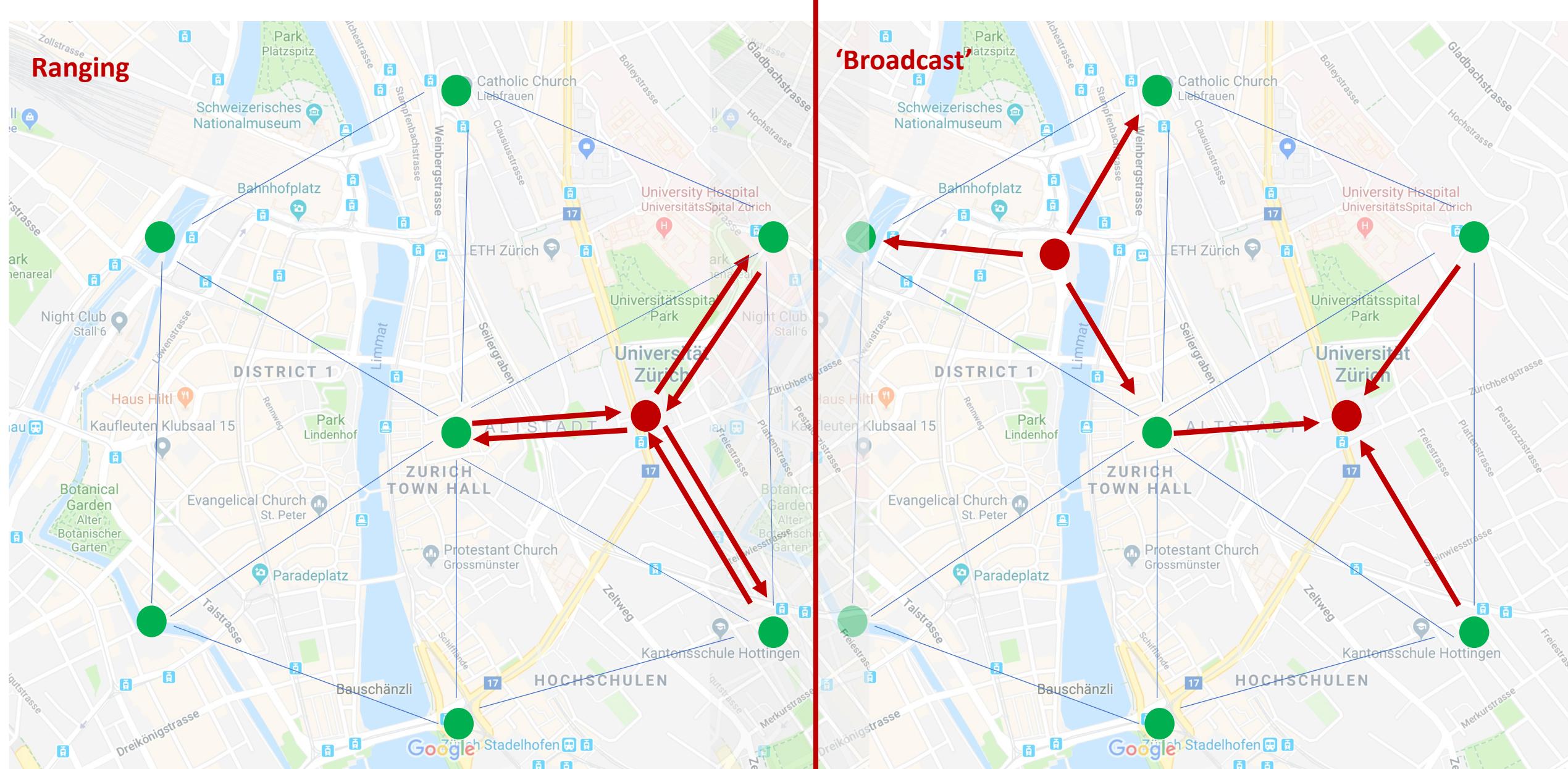
# Secure Positioning

# Bi-Directional Ranging and Verifiable Multilateration

- Bi-Directional ranging offers hope.
- Round-trip Time-of-Flight Protocols can *prevent distance shortening attacks.*
- If distance shortening is prevented, we can build a secure positioning system based on *Verifiable Multilateration within verification triangles.*



# Securing Positioning with Secure Ranging



# Replay / Relays on Broadcast Techniques

Broadcast techniques are subject  
relay/replay/delay-based spoofing attacks.  
Hard to fully prevent.

**Relay/replay attacks on GNSS signals**

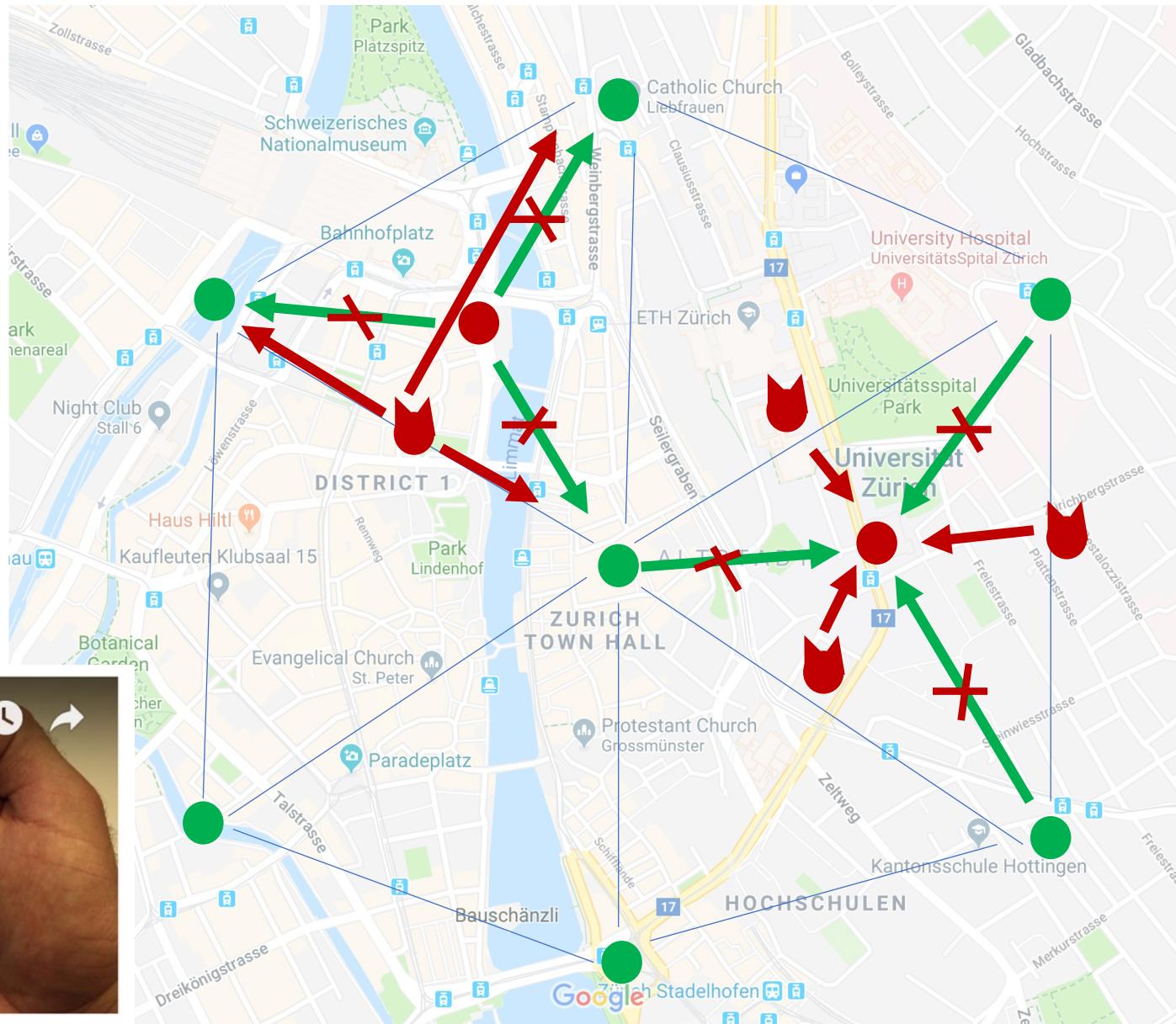
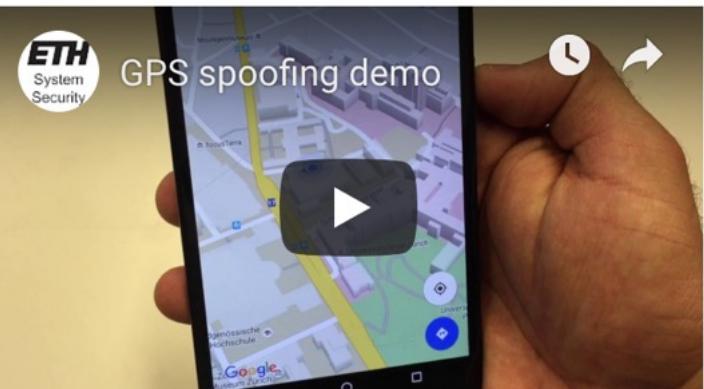
Authors: Malte Lenhart, Marco Spanghero, Panagiotis Papadimitratos Au

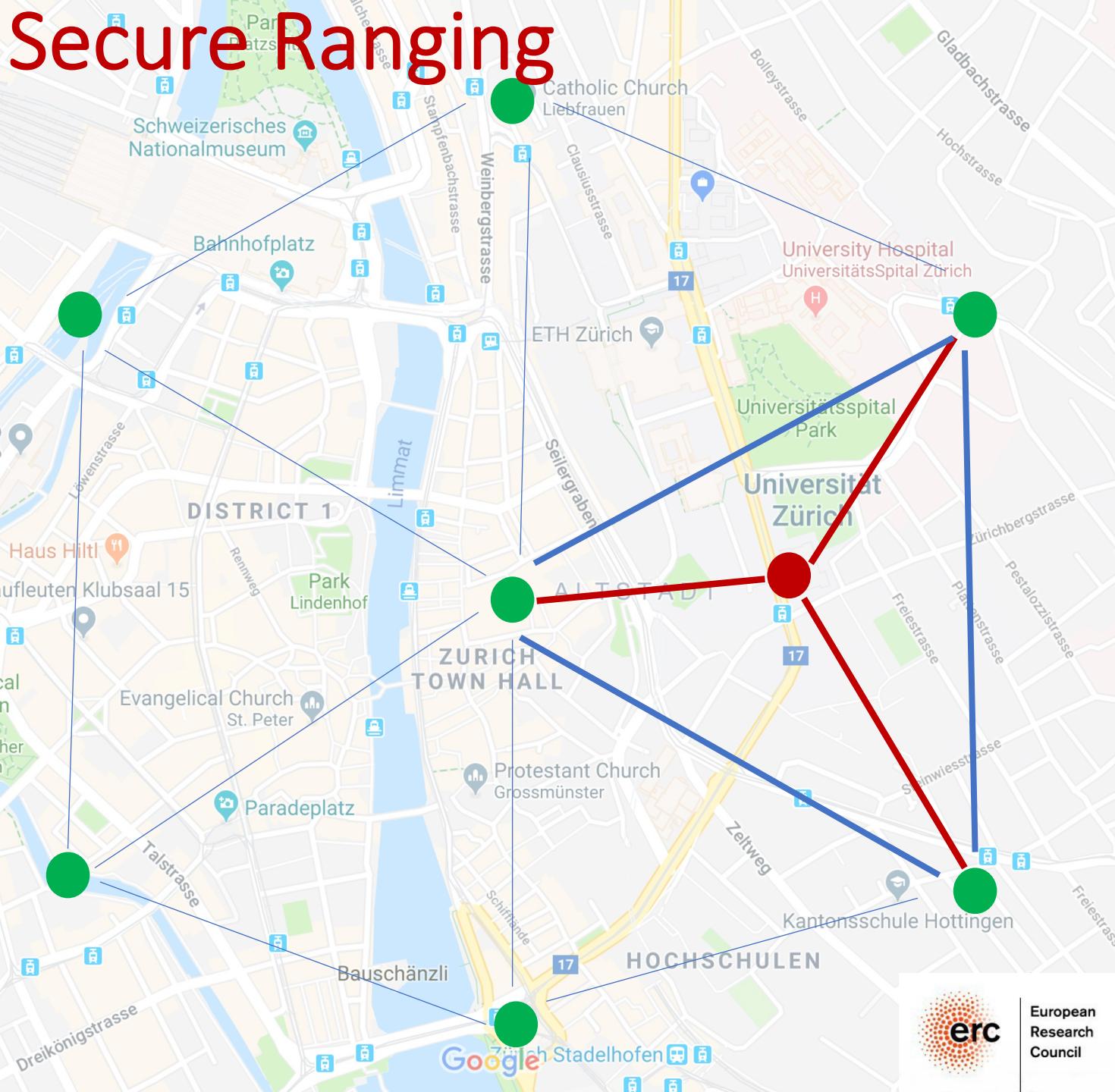
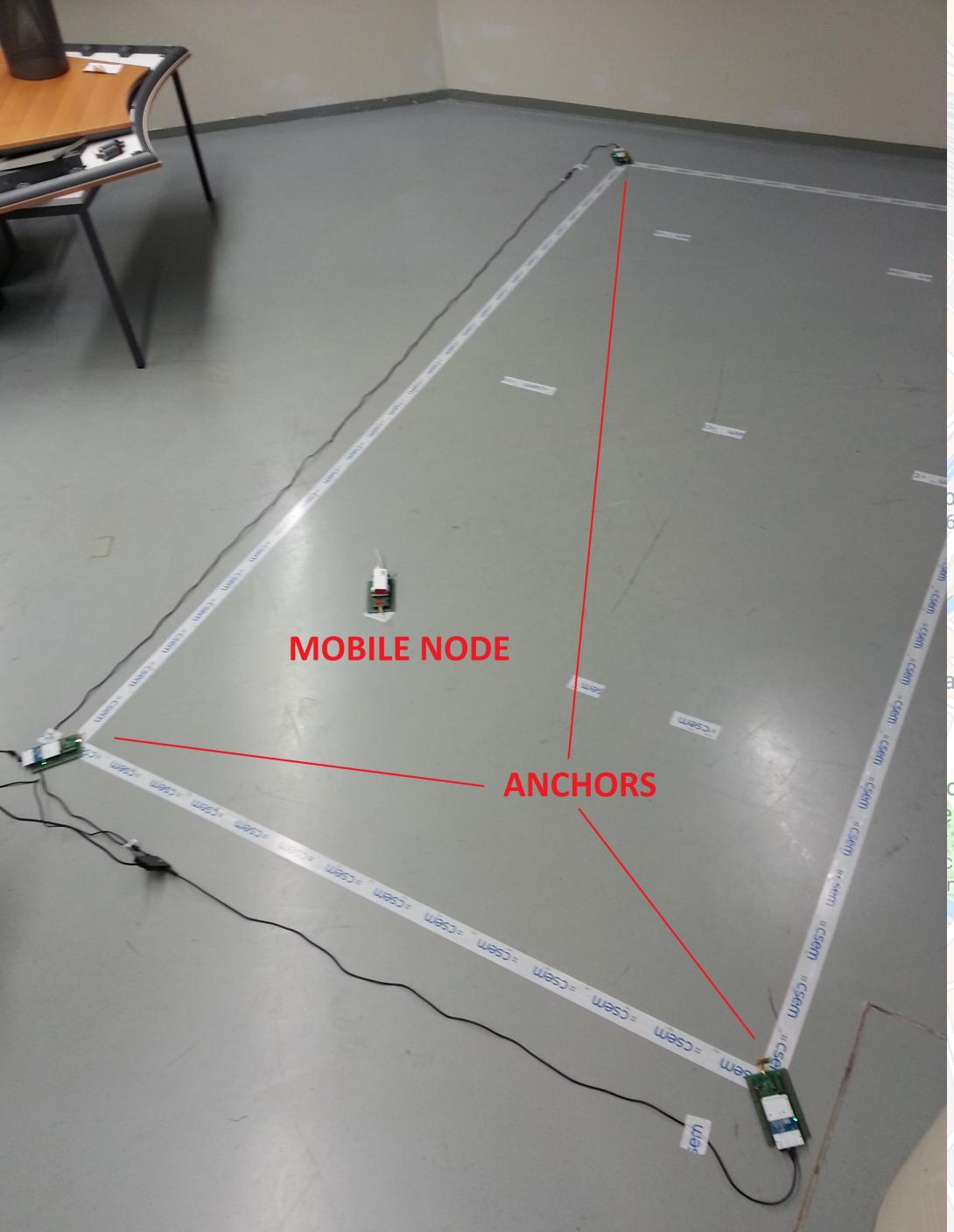
**Investigation of Multi-device Location Spoofing Attacks on Air Traffic Control and Possible Countermeasures**

Daniel Moser, Dept. of Computer Science, ETH Zürich, Switzerland; Patrick Leu, Dept. of Computer Science, ETH Zürich, Switzerland; Vincent Lenders, armasuisse Thun, Switzerland; Aanjan Ranganathan, Dept. of Computer Science, ETH Zürich, Switzerland; Fabio Ricciato, Faculty of Computer and Information Science, University Ljubljana, Slovenia; Srdjan Capkun, Dept. of Computer Science, ETH Zürich, Switzerland

**BBC NEWS**  
**Technology**  
**Researchers use spoofing to 'hack' into a flying drone**  
29 June 2012  
American researchers took control of a flying drone by "hacking" into its GPS system - acting on a \$1,000 (£640) dare from the US Department of Homeland Security (DHS).  
Drones are mostly used for military operations

**GPS spoofing demo**





# Verifiable Multilateration

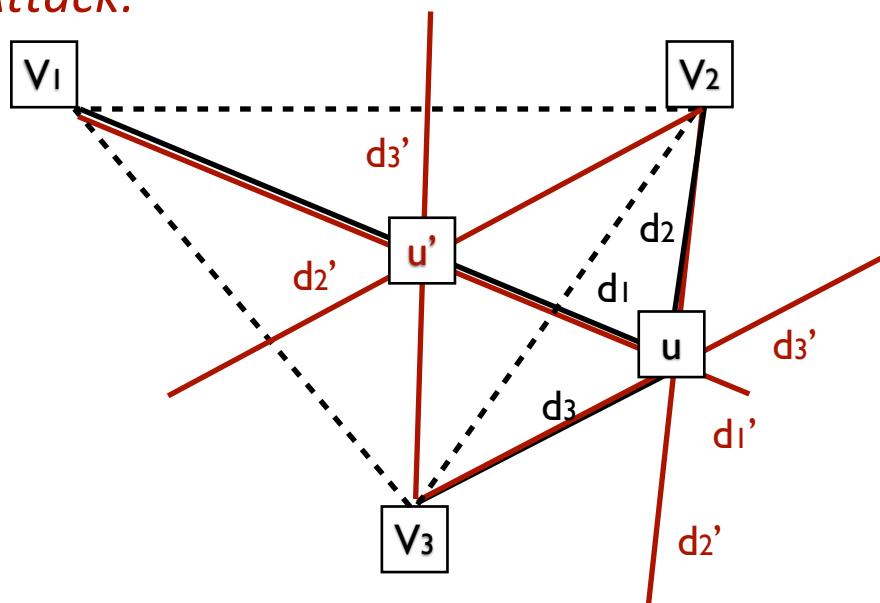
The algorithm and the errors:

- Need to be careful how the position is computed!
- Example: *Minimum Mean Square Estimate (MMSE)*

Let  $f_i(x'_u, y'_u) = db_i - \sqrt{(x_i - x'_u)^2 + (y_i - y'_u)^2}$

The position of  $u$  is obtained by minimizing  
 $F(x'_u, y'_u) = \sum_{v_i \in \mathcal{T}} f_i^2(x'_u, y'_u)$   
over all estimates of  $u$

- *Attack:*



# Verifiable Multilateration

## Verifiable Multilateration Algorithm

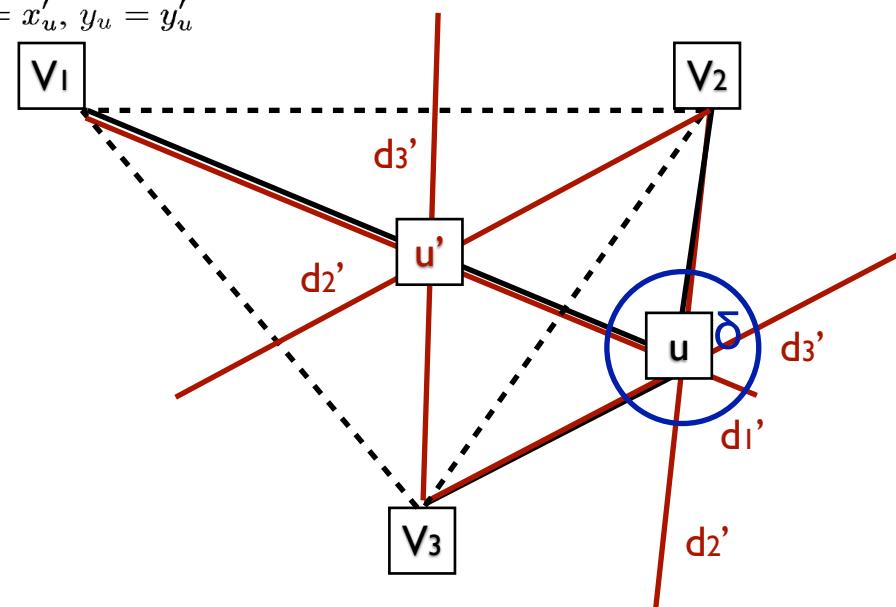
$\mathcal{T} = \emptyset$ ; set of verification triangles enclosing  $u$

$\mathcal{V} = \{v_1, \dots, v_n\}$ ; set of verifiers in the power range of  $u$

1 For all  $v_i \in \mathcal{V}$ , perform distance bounding  
from  $v_i$  to  $u$  and obtain  $db_i$

2 With all  $v_i \in \mathcal{V}$ , compute the estimate  $(x'_u, y'_u)$  of the position  
by MMSE

3 If for all  $v_i \in \mathcal{V}$ ,  $|db_i - \sqrt{(x_i - x'_u)^2 + (y_i - y'_u)^2}| \leq \delta$  then  
for all  $(v_i, v_j, v_k) \in \mathcal{V}^3$ , if  $(x'_u, y'_u) \in \Delta(v_i, v_j, v_k)$   
then  $\mathcal{T} = \mathcal{T} \cup (v_i, v_j, v_k)$   
if  $|\mathcal{T}| > 0$  then position is accepted and  $x_u = x'_u, y_u = y'_u$   
else the position is rejected  
else the position is rejected

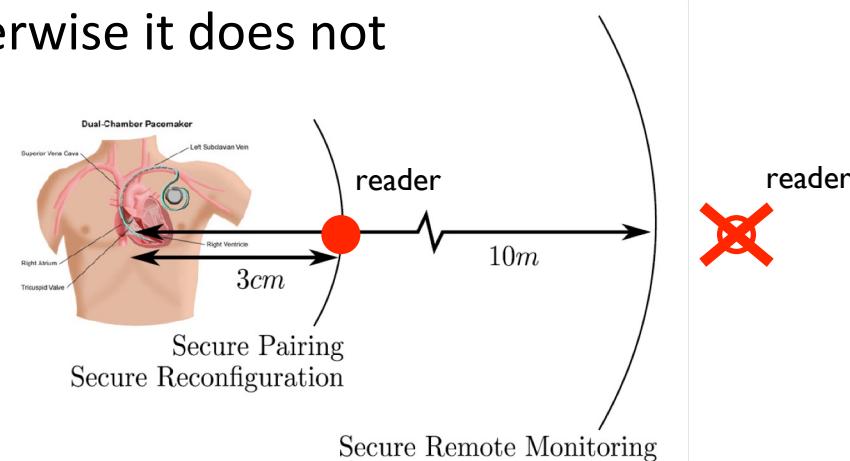


*Untrusted Prover (Distance Bounding)*

# Secure Proximity Verification

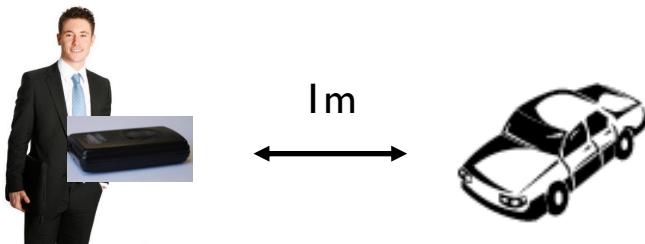
One (untrusted) device wants to *prove to be close* to another device.

- e.g., if a reader is close to the pacemaker, it gets access, otherwise it does not

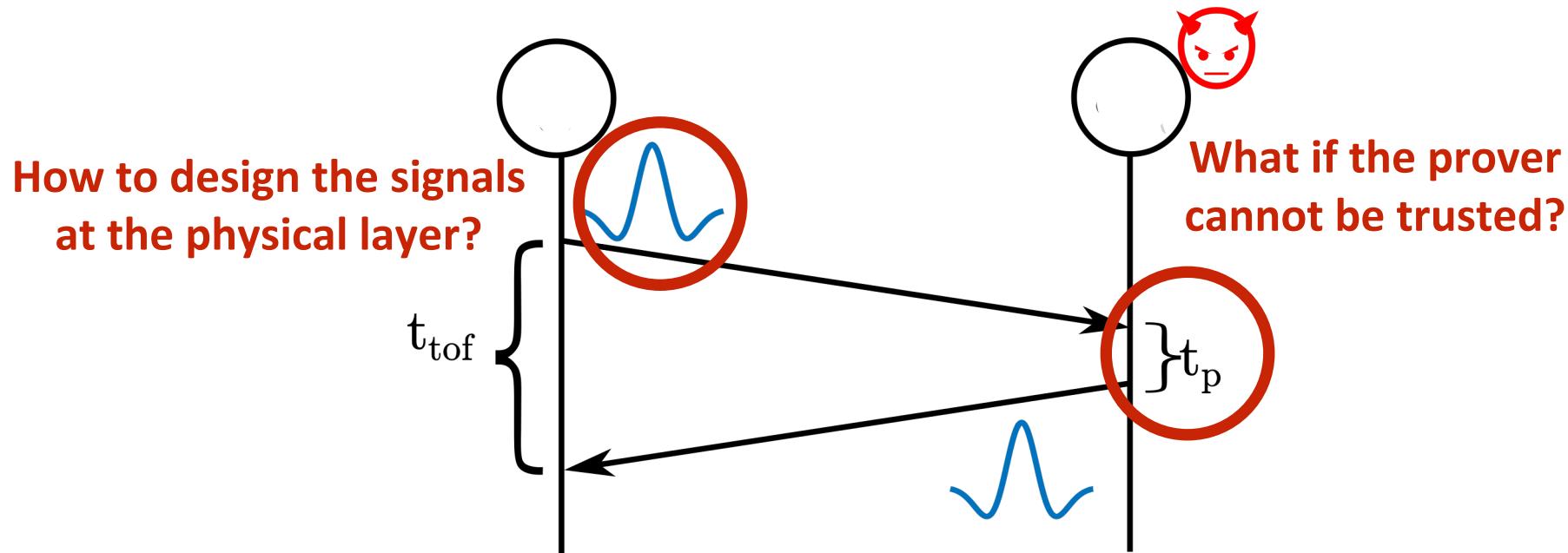


Two devices want to *verify if they are indeed close*.

- e.g., a car and a key want to verify if they are physically close



# Estimating Proximity using Time of Flight



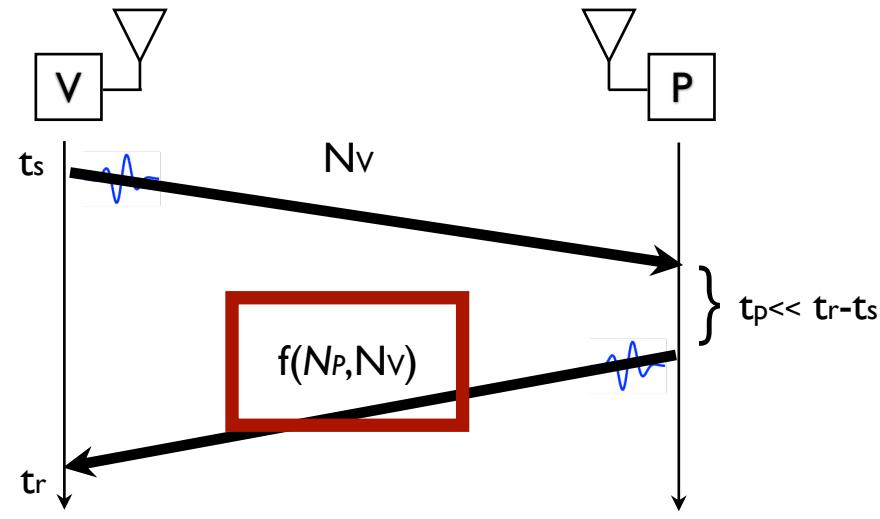
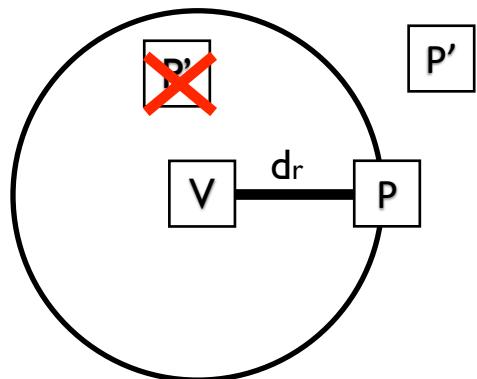
$$d = c * (t_{\text{tof}} - t_p) / 2$$

Can an attacker reduce time?

Manipulating time is harder than changing signal strength or phase  
BUT...

# Distance Bounding [BrandsChaum93]

## Basic Idea

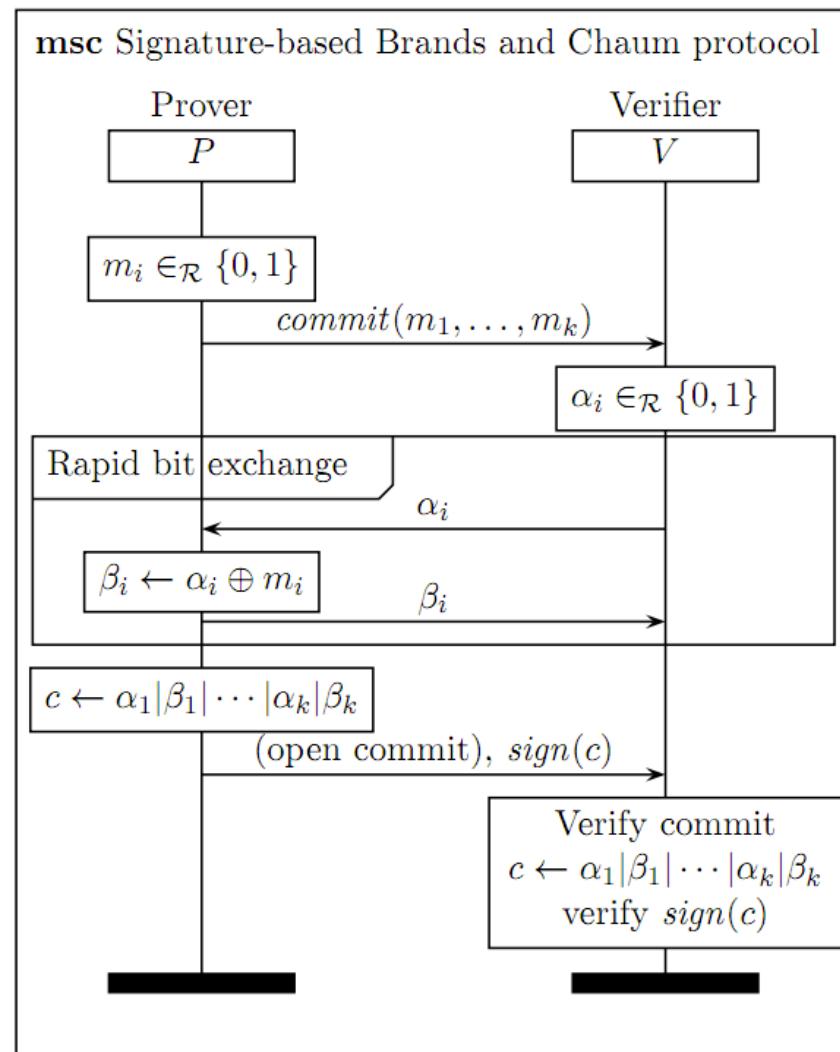


$$d = (t_r - t_s - t_p)c/2$$

Property:

Measured distance  $d$  should be an *upper bound* on the true distance  $d_r$  between  $V$  and  $P$ .

# Distance Bounding [BrandsChaum93]



## 2.1 Fraud Definitions

A distance-bounding protocol is a process whereby a party (known as *verifier*) is assured (i) of the identity of a second party (known as *prover*) and (ii) that the prover is located in his close vicinity (known as *neighborhood*). Four frauds against distance-bounding are usually considered, *impersonation*, *distance*, *mafia*, and *terrorist* frauds [2], which are introduced below.

*Impersonation.* An *impersonation fraud* is an attack where an adversary acting alone purports to be a legitimate prover.

*Distance Fraud.* A *distance fraud* is an attack where a dishonest prover purports to be in the neighborhood of the verifier. He cheats without help of other entities located in the neighborhood.

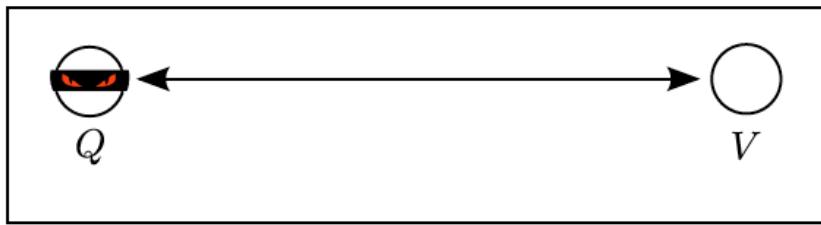
*Mafia Fraud.* A *mafia fraud* is an attack where an adversary defeats a distance-bounding protocol using a man-in-the-middle between the verifier and an honest prover located outside the neighborhood.

*Terrorist Fraud.* A *terrorist fraud* is an attack where an adversary defeats a distance-bounding protocol using a man-in-the-middle between the verifier and a dishonest prover located outside of the neighborhood under the following circumstances. The dishonest prover actively helps the adversary to maximize her current attack success probability but without giving her any advantage for future man-in-the-middle attacks. (In such attacks, the man-in-the-middle (MiM) would attempt to pass the distance-bounding protocol as a valid prover/tag that the MiM does not represent/possess.)

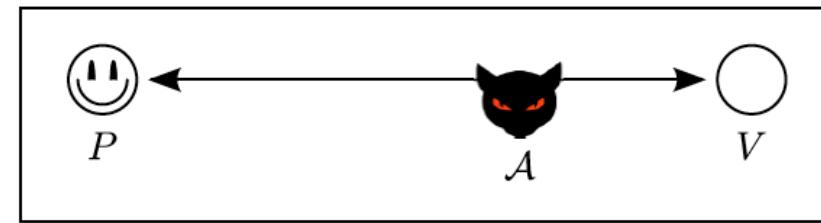
Note that protocols that are known to suffer from a key-recovery attack are not analyzed in this article. This includes Tu and Piramuthu's protocol [71] whose flaws are discussed in References [50, 56], Reid et al.'s protocol [64] broken in References [5, 53], and Hitomi whose vulnerabilities are described in Reference [66]. While Reference [9] points out a key recovery attack on Bussard and Bagga's protocol [18], this protocol is kept in this analysis, because the attacks presented in Reference [9] could be applied to other protocols and designers must be aware of their existence to avoid them. Note also that the length of the long-term secret keys of the parties, the length of the signatures (when appropriate), and the length of the nonces are assumed to be large enough, such that exhaustive search and replay attack are not relevant. Finally, the pseudo random functions used in the protocols are assumed to be without design flaws, i.e., no trapdoor pseudo random functions, like those discussed in Reference [13].

Another type of fraud, known as *distance hijacking*, has recently been introduced in Reference [26]. The fraud considers a dishonest prover who aims to convince a verifier that he is located within the verifier's neighborhood, abusing for that some other provers who are indeed in the verifier's neighborhood. For example, a dishonest prover can reach his goal by hijacking the fast phase of a distance-bounding protocol executed between an honest (closer) prover and the

# Distance Fraud



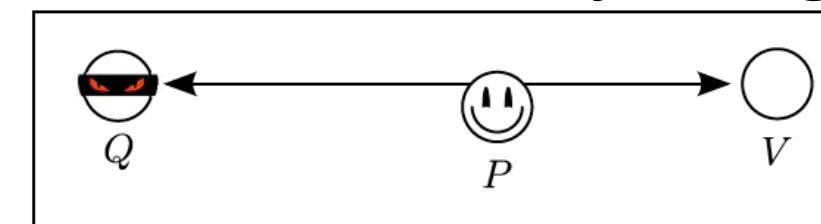
# Mafia Fraud



# Terrorist Fraud



# Distance Hijacking



# Distance Hijacking Attack

<https://eprint.iacr.org/2011/129.pdf>

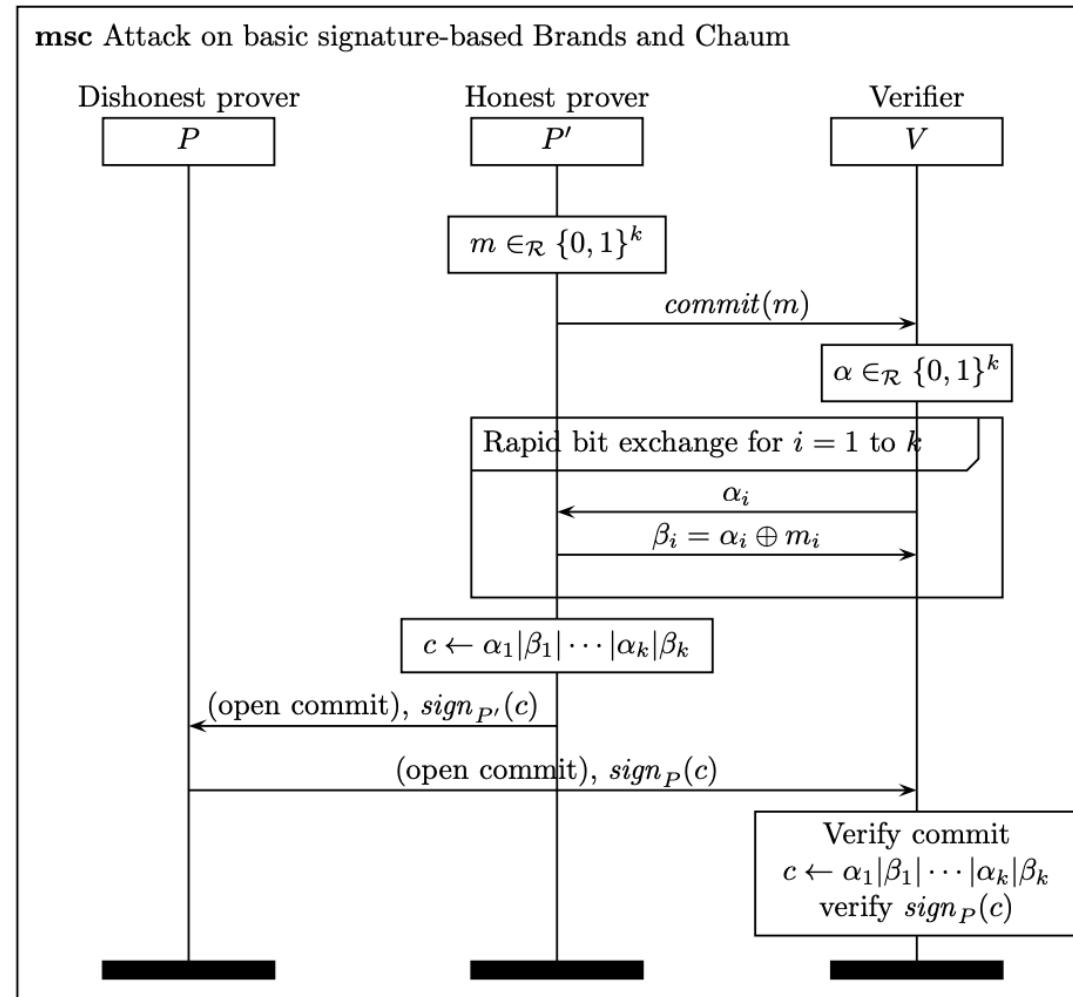
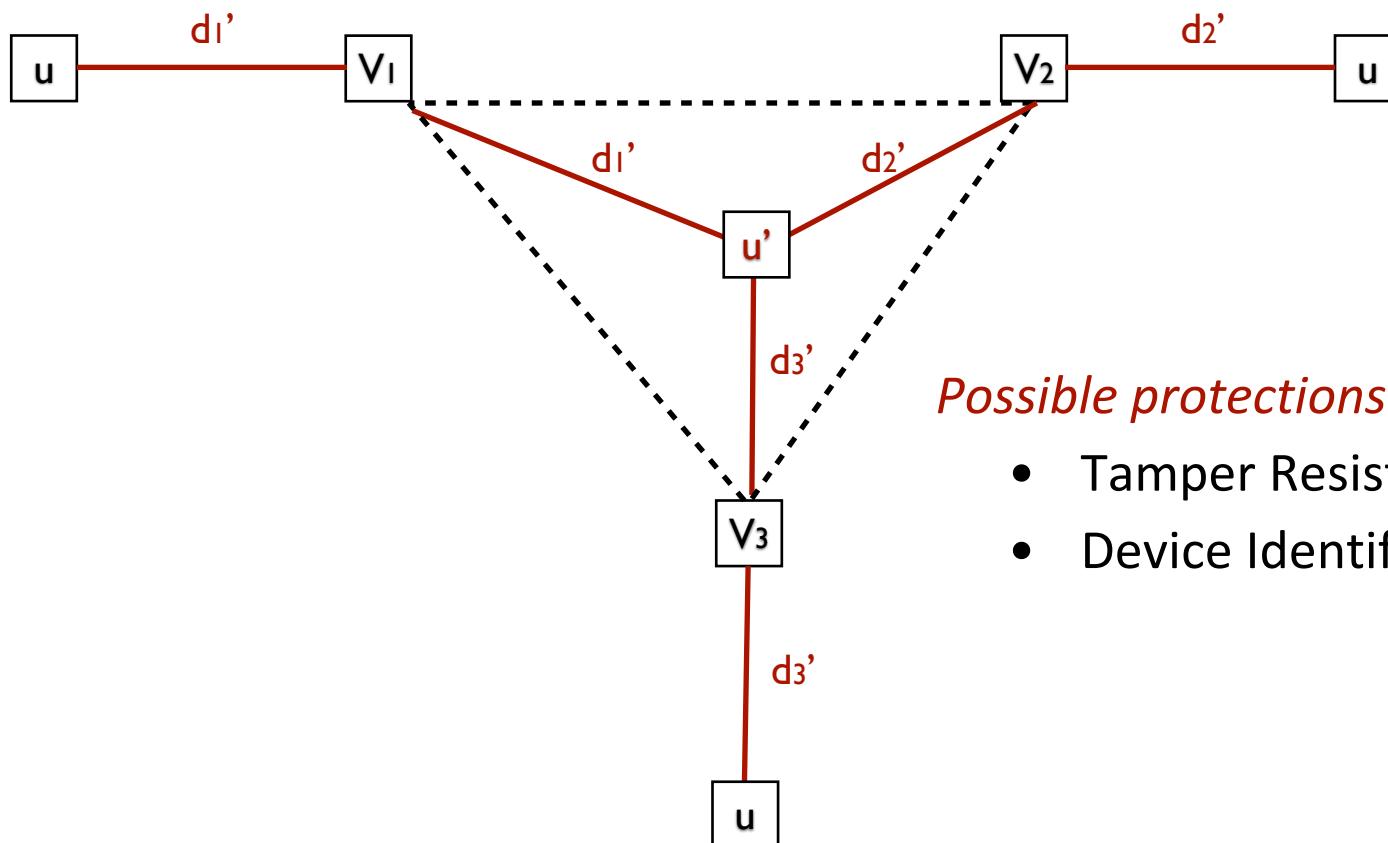


Figure 2: Distance Hijacking attack on basic signature-based Brands and Chaum.  $\text{sign}_P$  and  $\text{sign}_{P'}$  denote the signatures with the signature keys of  $P$  and  $P'$ , respectively.

# Verifiable Multilateration

Collusion attacks (only with untrusted prover under location verification)

- *Attack:*



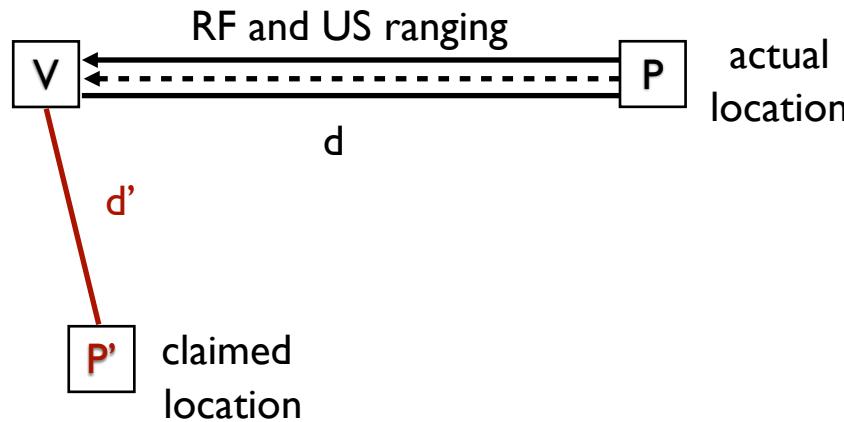
*Possible protections:*

- Tamper Resistance
- Device Identification

# Location Verification using Hidden and Mobile Stations (*Verifiers*)

The basic idea:

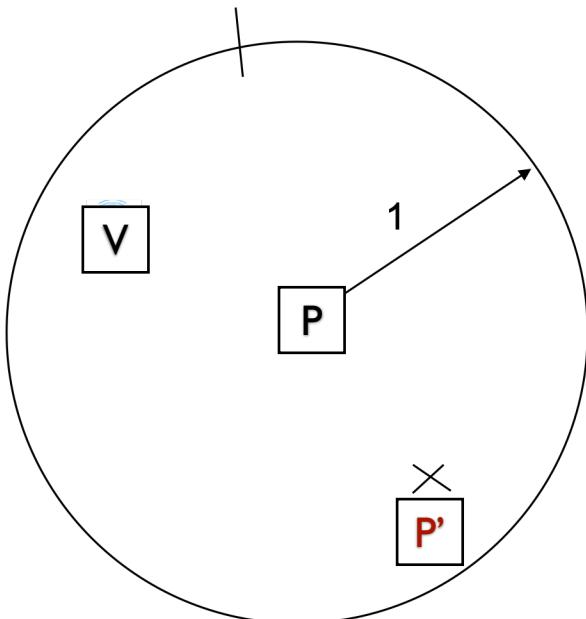
- *If the prover does not know where the verifiers are, it doesn't know how to cheat.*



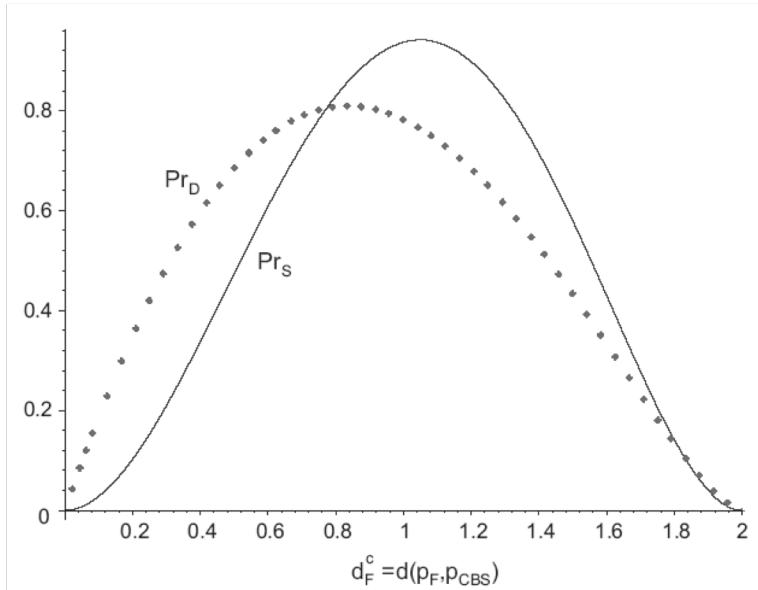
$$p(\text{successful cheating}) = p(d - d' \leq \Delta)$$

where  $\Delta$  **is the ranging/localization accuracy**

# Location Verification using Hidden and Mobile Stations (*Verifiers*)



Observation 1:



not all distances are equally likely

- Not all locations are equally easy to fake (center is the ‘easiest’).
- *Problems if the attacker knows where verifiers cannot be.*

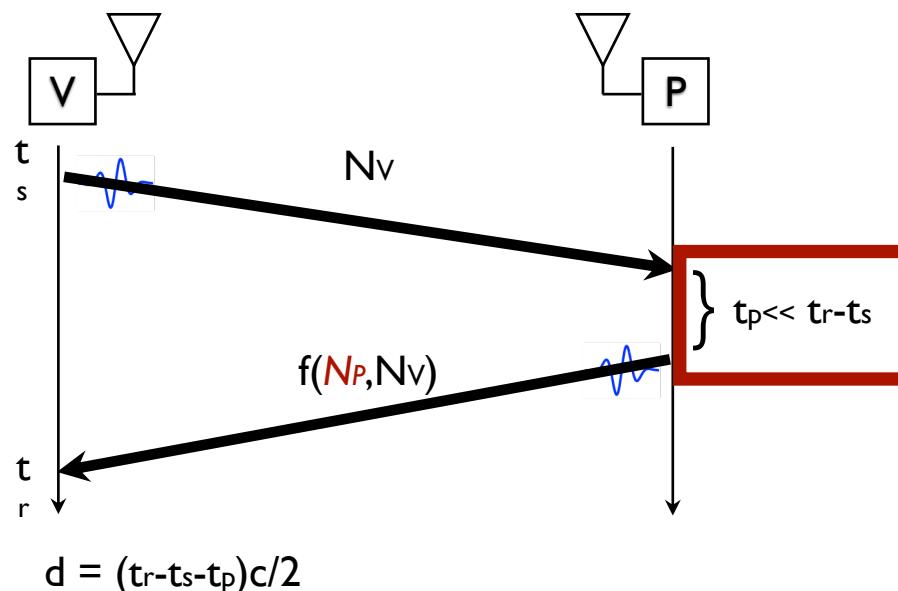
# Implementation

# Distance Bounding: $f()$ and $t_p$

Provers should **quickly receive  $N_v$ , compute  $f(N_v, N_p)$  and send  $f(N_v, N_p)$**

- The verifier estimates prover's processing =  $t_p$
- If attacker's processing = 0 then he **can cheat by  $t_p/2$**
- Thus ideally  $t_p=0$ s, in most applications  $t_p=1-2$ ns (15-30cm)
- $t_p$  needs to be **stable and short**

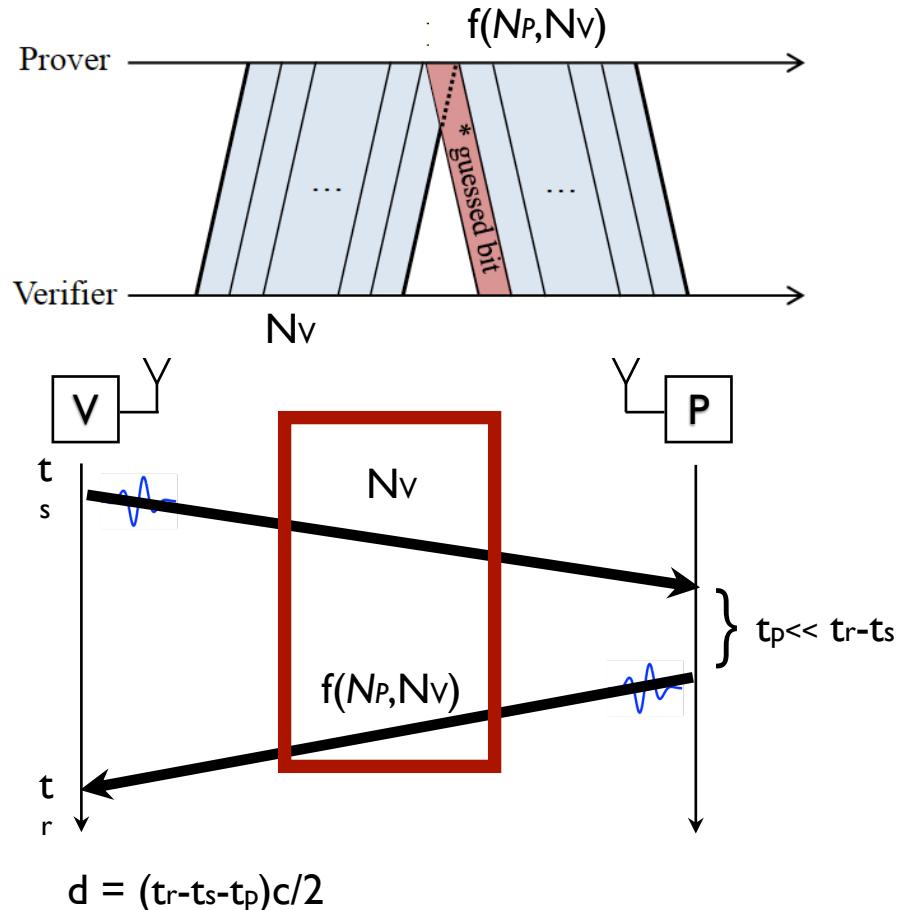
*Main assumption: we do not control the prover*



# Distance Bounding: $N_v$

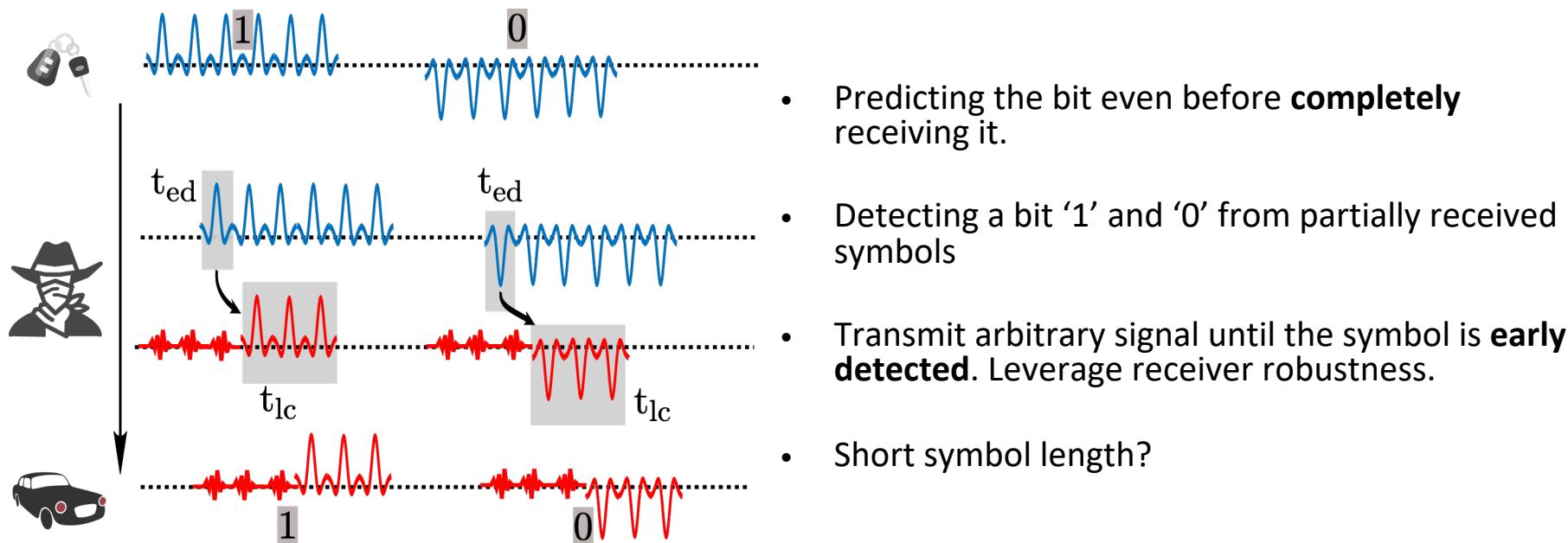
$N_v$  and  $f(N_p, N_v)$  should be “short” in the # of bits [HankeKuhn]

- short compared to the required accuracy / security



# Distance Bounding: *symbols*

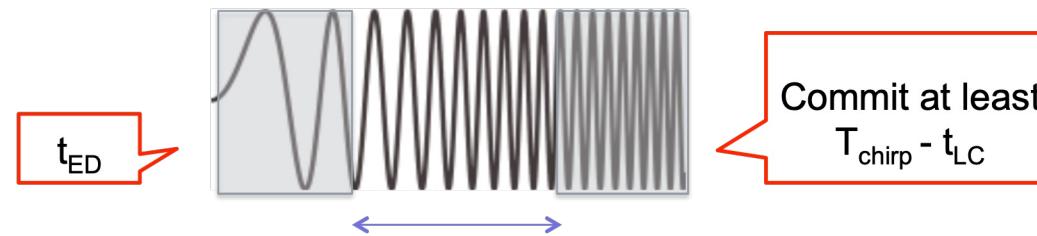
## Early detect and late commit attacks



# Distance Bounding: *symbols*

*Chirp SS ranging (802.15.4) systems strongly affected*

- long symbol lengths allow for simple ED and LC attacks
- Early Detection
- Late Commit



$$t_{GAIN} = t_{LC} - t_{ED} - t_{HW}$$

$$D = c \times t_{GAIN}$$

# Realization of RF Distance Bounding: *Processing Function $f(N_v, N_p)$*

$f(N_v, N_p)$  is computed by the prover:

- takes as input  $N_v$  (received from the verifier)
- takes as input  $N_p$  (locally generated by the prover)
- Should allow that the prover: *receives  $N_v$ , computes and outputs  $f(N_v, N_p)$  in a short time (few ns)*

***DB protocols in the literature:***

[BethDesmedt]  $\text{sign}(N_v); h(N_v); \text{mac}(N_v); E(N_v); \dots \Rightarrow t_p \gg ns$

[BrandsChaum, CapkunInfocom05, ...] *XOR*  $\Rightarrow t_p = ?$

[HanckeKuhn, TippenhauerESORICS09, ...] *bit comparison*  $\Rightarrow t_p = ?$

> 20 proposed protocols, not one was *fully* implemented

*Can the proposed DB protocols be realized?*

# Realization of RF Distance Bounding: Processing Function $f(N_v, N_p)$

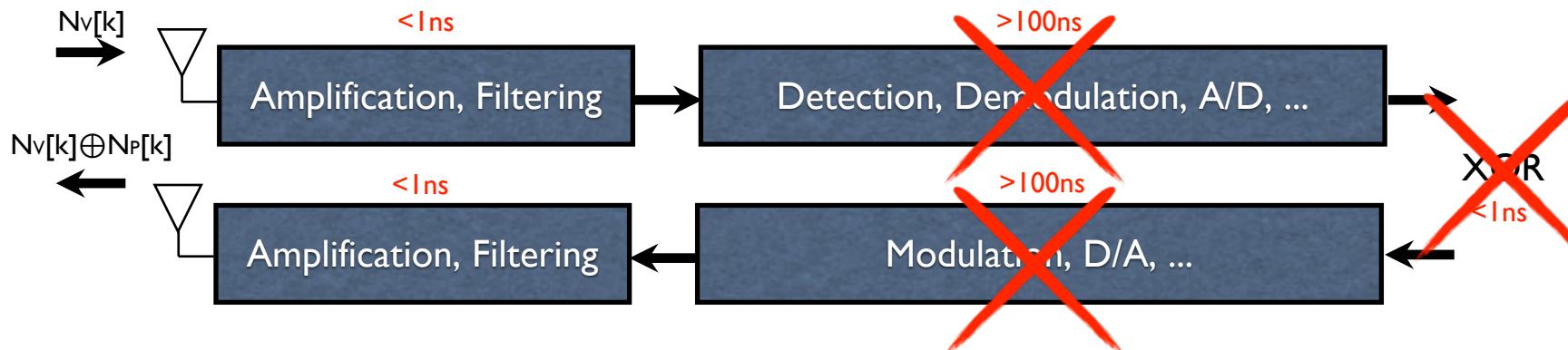
[BethDesmedt] sign(); h(); mac(); E(); ... =>  $t_p \gg ns$

[BrandsChaum, ...] **XOR** =>  $t_p = ?$  ( $nx100ns$  ?)

[HanckeKuhn, ...] **bit comparison** =>  $t_p = ?$  ( $nx100ns$  ?)

[RasmussenSec09, ...] **CRCS (analog modulation)** =>  $t_p < 1ns$

... > 20 proposed protocols



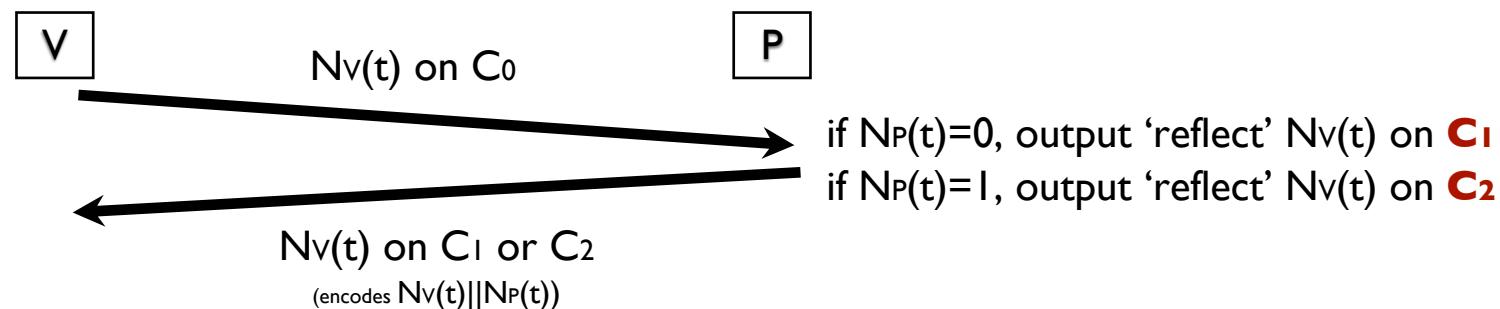
*Can we use functions that don't require interpretation (demodulation)  $N_v$  ?*

# A new Function: CRCS

This approach: *Challenge Reflection with Channel Selection*

- Prover does not interpret  $Nv$
- All *time-critical* processing is done in *analog*
- Verifier does “all the work”

Main idea ( $C_0, C_1, C_2$  are channels)

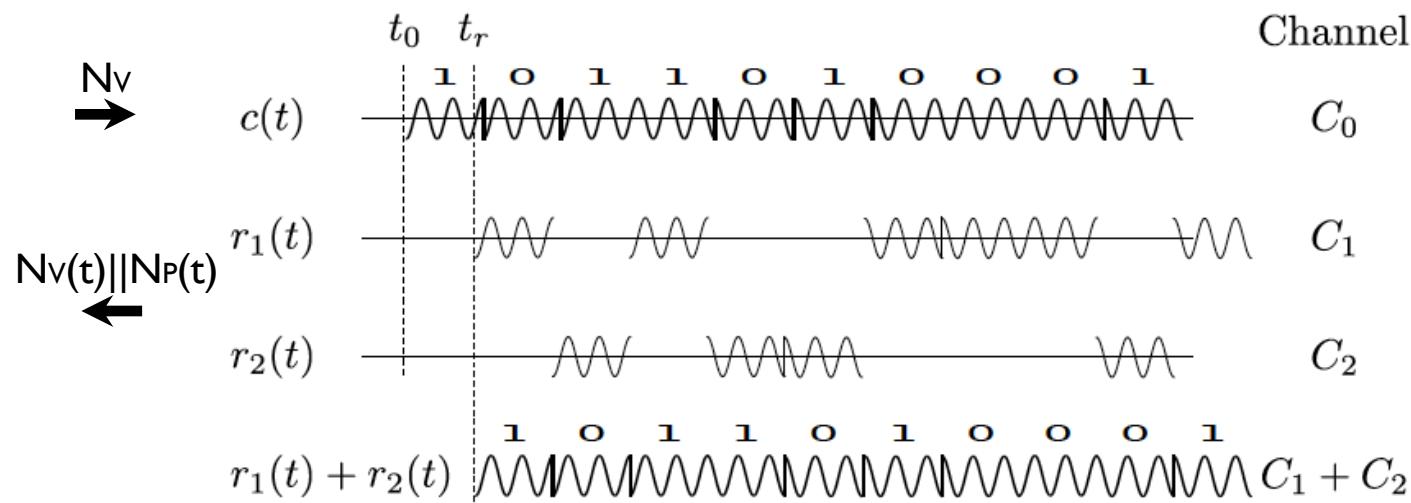


# A new Function: CRCS

This approach: *Challenge Reflection with Channel Selection*

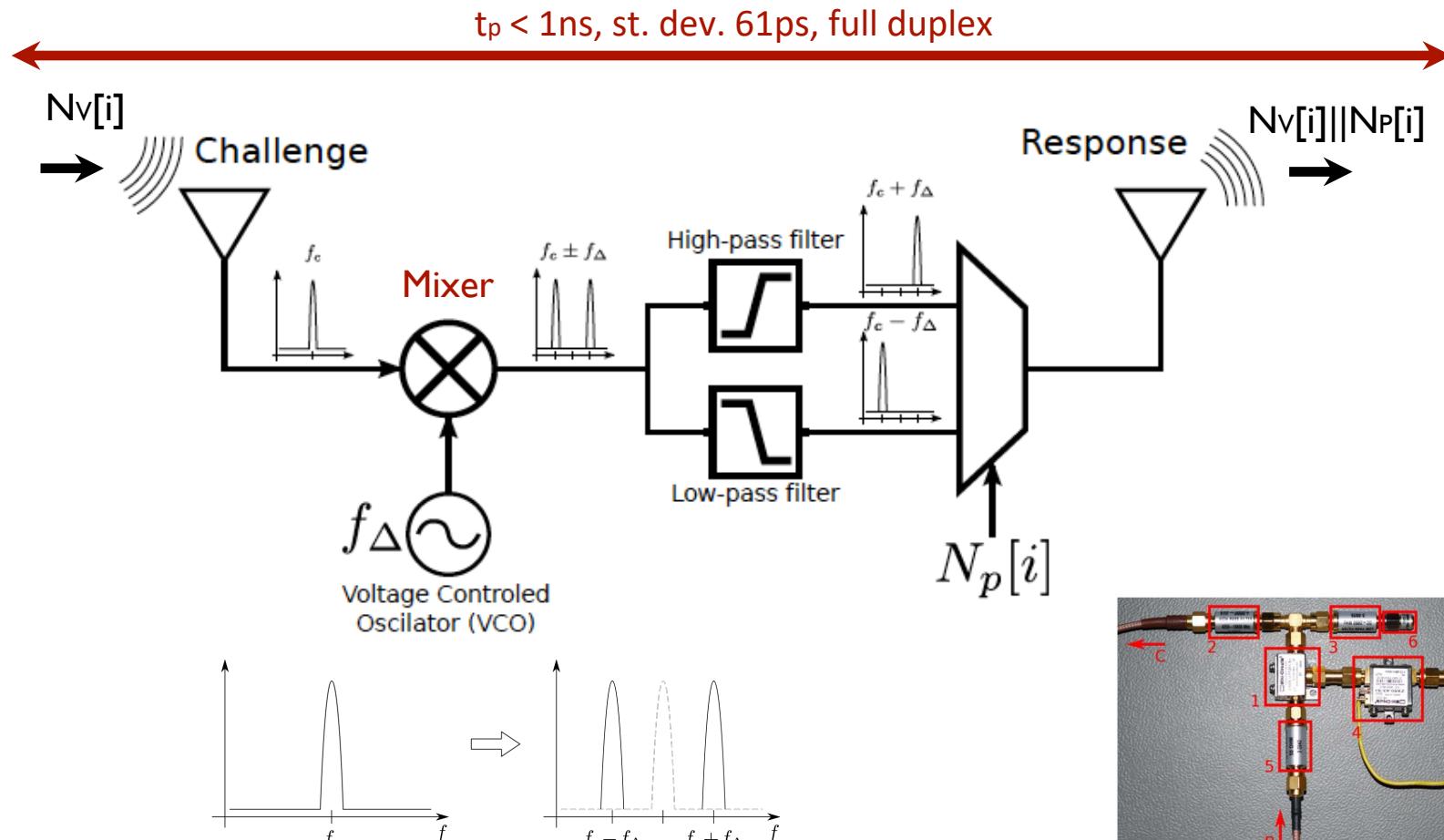
- Prover does not interpret Nv
- All *time-critical* processing is done in *analog*
- Verifier does “all the work”

Main idea ( $C_0, C_1, C_2$  are channels)

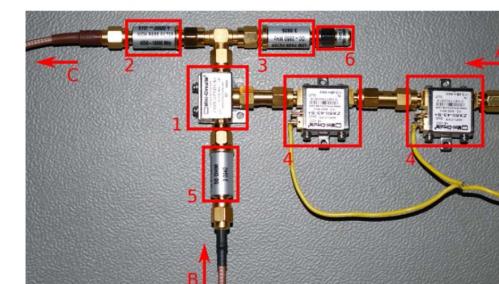


# A new Function: CRCS

## Implementation of CRCS



Mixer up+down converts the input signal



[www.zisc.ethz.ch](http://www.zisc.ethz.ch)

## Secure Positioning

Home

Security of GNSS

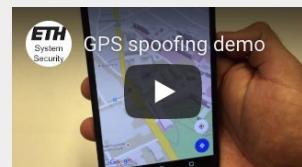
Secure Proximity Verification

Secure Positioning

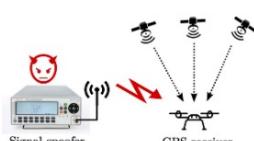


### About us

With the development of new location-based services and the expected deployment of cyber-physical systems (e.g., autonomous cars and drones) the reliance on location and time information in critical applications will only increase. Today's positioning systems are vulnerable to location spoofing by which devices can cheat on their own positions or can manipulate the measured positions of other devices. Proximity-based access control systems are insecure against man-in-the-middle relay attacks (e.g., passive keyless entry system in automobiles, contactless access and payment cards). This problem cannot be fixed by a simple upgrade - existing positioning systems rely on legacy distance measurement techniques and protocols that were designed without security considerations or with security as an after-thought. For more than a decade, we in the [System Security Group at ETH Zurich](#) have been designing secure technologies to address these issues. [More details >>](#)



### Securing Satellite-based Navigation Systems



Today, it is possible to spoof a Global Satellite Navigation System (e.g., GPS, Galileo) receiver to any

### Secure Positioning



There is a clear need to develop a new positioning infrastructure that is secure and private by design. Such an infrastructure should provide resilience to

### Secure Proximity Verification



Proximity information is ubiquitously used in a wide variety of systems. For example, contactless access tokens that use proximity information are prevalent