Data-Plane Signaling in Cellular IoT: Attacks and Defense

Zhaowei Tan, Boyan Ding, Jinghao Zhao, Yunqi Guo, Songwu Lu

Cellular IoT Networks

C-IoT: Standardized Low-Power Wide-Area Networks



Wide Coverage

Low Power

Flexibility

Cellular IoT Networks

C-IoT: Standardized Low-Power Wide-Area Networks

Wide Coverage

Low Power

Flexibility

Anywhere, anytime Internet services through cellular infrastructure

Cellular IoT Networks

C-IoT: Standardized Low-Power Wide-Area Networks

Wide Coverage

Low Power

Flexibility

Extended power-saving techniques for extended battery life

Cellular IoT Networks

C-IoT: Standardized Low-Power Wide-Area Networks

Wide Coverage

Low Power

Flexibility

Category M1 (Cat-M) and Narrowband IoT (NB-IoT) for different use cases

Cellular IoT Networks

C-IoT: Standardized Low-Power Wide-Area Networks

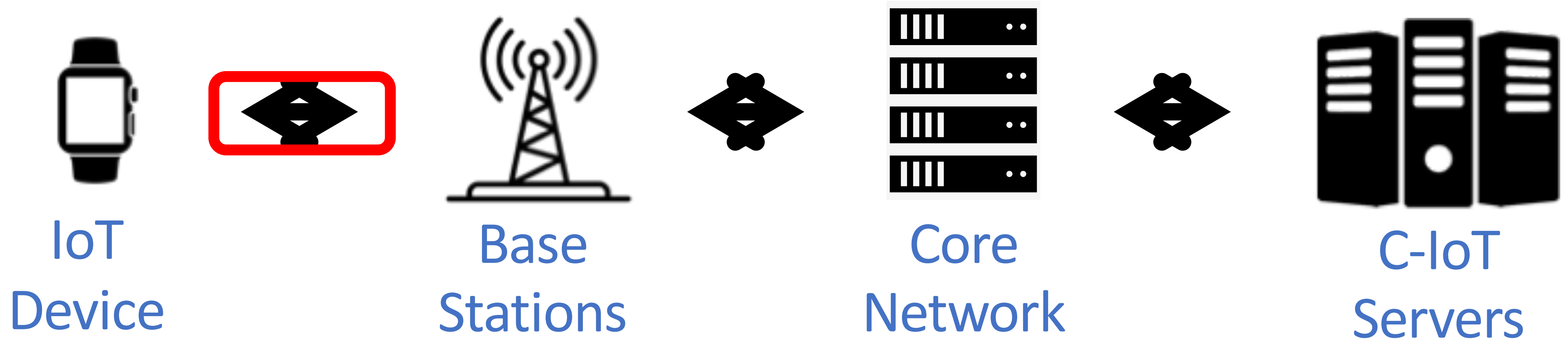
Wide Coverage

Low Power

Flexibility

However, what about its security?

Focus of C-IoT Security in This Work



- We consider threats in radio access network
- We assume the attacker cannot compromise device or any key

Security Measures for C-IoT RAN

Mutual authentication establishes security context on both sides

– All subsequent data packets/control-plane signaling are protected

Is an attack still feasible *after* mutual authentication?

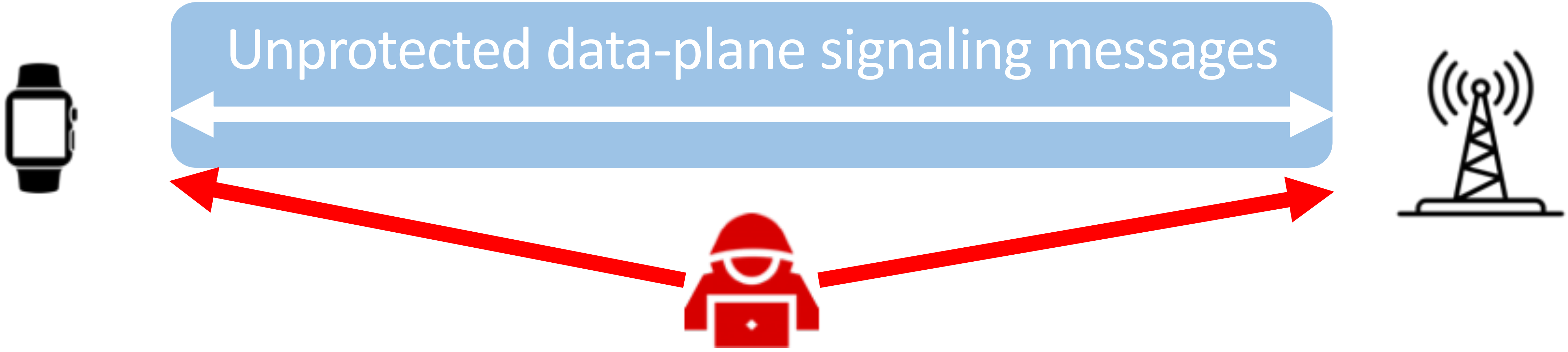
Yes, C-IoT Data-Plane is Still Vulnerable

Data-plane sub-layers also contain signaling messages

- They facilitate data transfer, e.g., provide power control, scheduling, etc.

Vulnerability: Data-plane signaling is neither encrypted nor integrity protected *after mutual authentication*

Vulnerability in Data-Plane Signaling



Data-plane signaling forgery

Vulnerability: Data-plane signaling is neither encrypted nor integrity protected *after mutual authentication*

Outline

The remaining of the talk:

1. Can forged data-plane signaling appear legitimate?
2. How to incur serious damage with forged signaling?
3. Is it possible to eliminate this vulnerability?

Forge data-plane signaling

- What are the challenges?
- How can an attacker address them?

How to convince the receiver?

The forged signaling must pass the checks at both PHY and MAC Protocols

PHY: receiver decodes the signals with the assigned parameters

Attacker: Modulate the signaling with correct parameters

MAC: base station schedules resource blocks (RB) for each device

Attacker: Forge the signaling in the scheduled RB

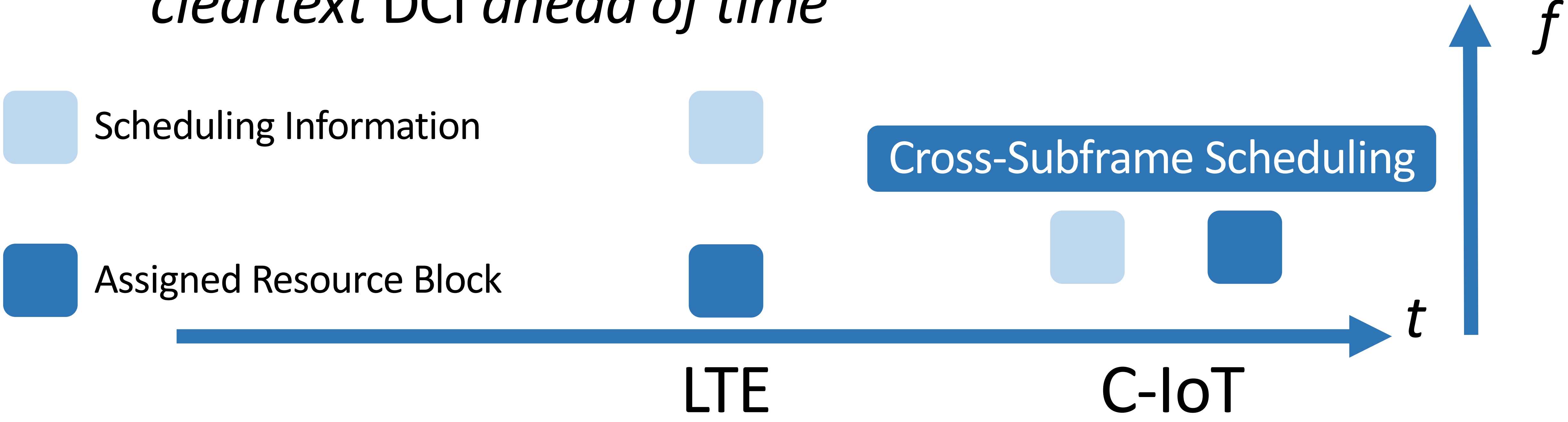
Challenge 1: Forging Data-Plane Signaling with Correct Encoding

- All necessary parameters can be inferred from DCI messages and broadcast messages
  - Parameters learned from broadcast: reference signal config, etc.
  - Parameters learned from unicast DCI: modulation, MCS, etc.

Vulnerability: DCI and broadcast messages are transmitted *in cleartext* over-the-air

Challenge 2: Send Forged Messages at Correct Frequency/Timing

- Unlike LTE, the authorized RBs can be inferred in *cleartext* DCI *ahead of time*



Challenge 2: Send Forged Messages at Correct Frequency/Timing

- Unlike LTE, the authorized RBs can be inferred in *cleartext* DCI *ahead of time*
- An attacker can decode scheduling info to calculate the assigned RBs based on 3GPP standard

Vulnerability: Scheduling can be inferred from cleartext DCI ahead of time due to cross-subframe scheduling

Other Technical Requirements

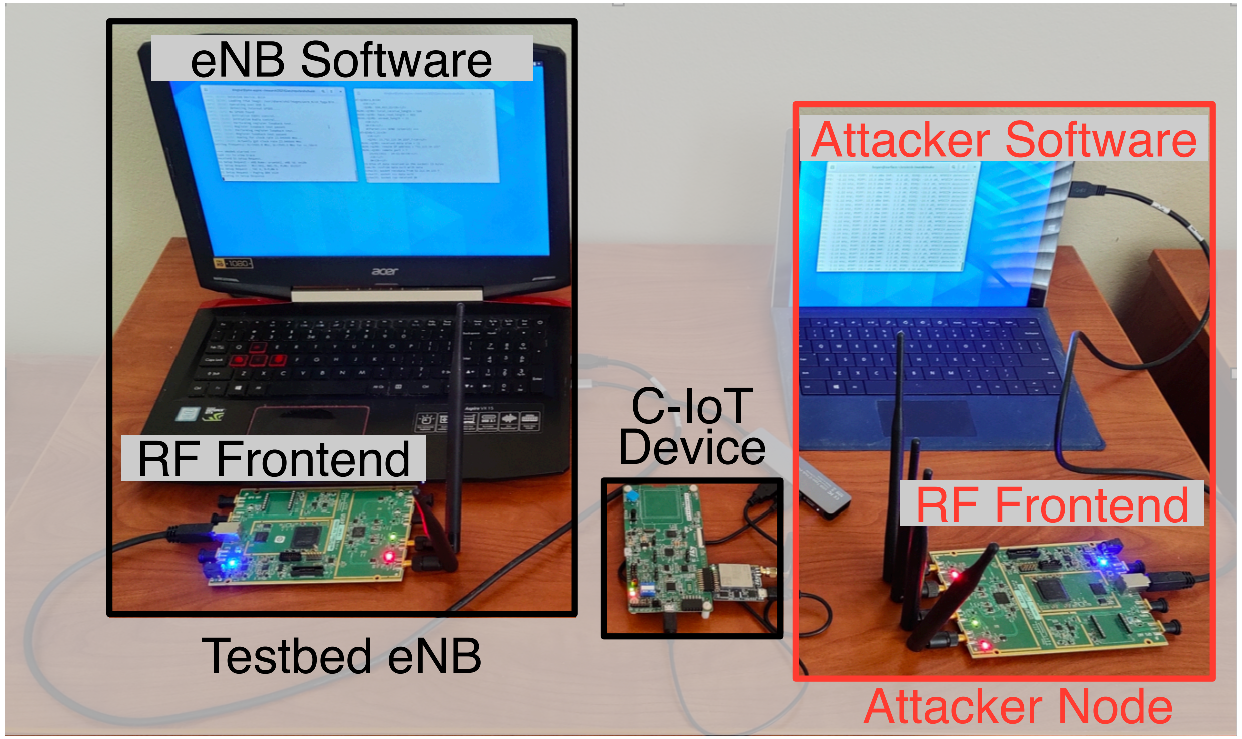
- Overshadow data from the authentic sender
  - Use capture effect with stronger signal strength
- Use correct physical-layer identifier
- Synchronization with the receiver
- Tackle Carrier frequency offset (CFO) and Sampling frequency offset (SFO)

- Please refer to our paper for details

Testbed for Attack Validation

The attacks are validated in our C-IoT testbed

- Standard-compliant C-IoT network (r14)
- Commercial off-the-shelf device
- USRP-based attacker



Attack Validation Results

We verify the successful forgery by checking logs on both server side and device side (with MobileInsight)

High success rate for both uplink and downlink

Relative Power	3dB	5dB	7dB
DL	40.3%	75.8%	99.9%
UL	41.2%	70.3%	99.8%

Attacks with forged signaling

- What are the data plane signaling that we could forge?
- How to cause beyond-simple-DoS damages?

Overview of the Attacks

- We design 6 attacks with the forged data-plane signaling
  - 3 single-layer, 3 cross-layer attacks
  - Each attack carefully determines the forgery content and context
  - Beyond simple DoS damages

Radio Resource Draining

Prolonged Packet Delivery

Flexible Throughput Limiting

Packet Delivery Loop

Device Localization

Connection Reset

Overview of the Attacks

- We design 6 attacks with the forged data-plane signaling
  - 3 single-layer, 3 cross-layer attacks
  - Each attack carefully determines the forgery content and context
  - Beyond simple DoS damages

Radio Resource Draining

Prolonged Packet Delivery

Flexible Throughput Limiting

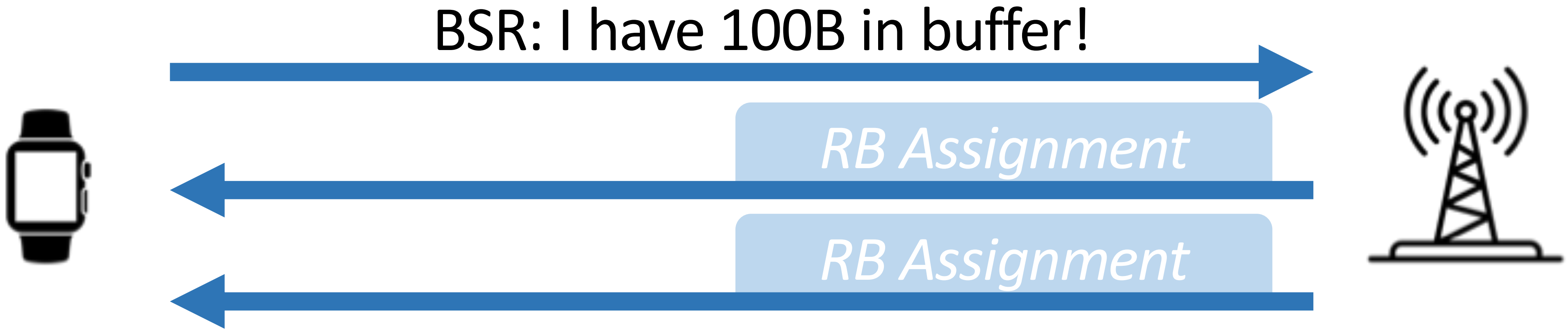
Packet Delivery Loop

Device Localization

Connection Reset

Radio Resource Draining with Buffer Status Report (BSR)

BSR: A message from device to network that requests for UL resource specified in its value

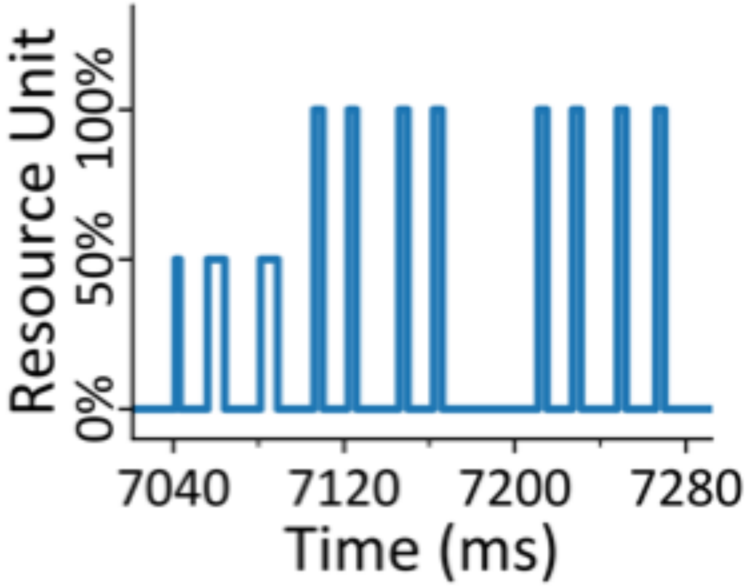


Radio Resource Draining with Buffer Status Report (BSR)

BSR: A message from device to network that requests for UL resource specified in its value

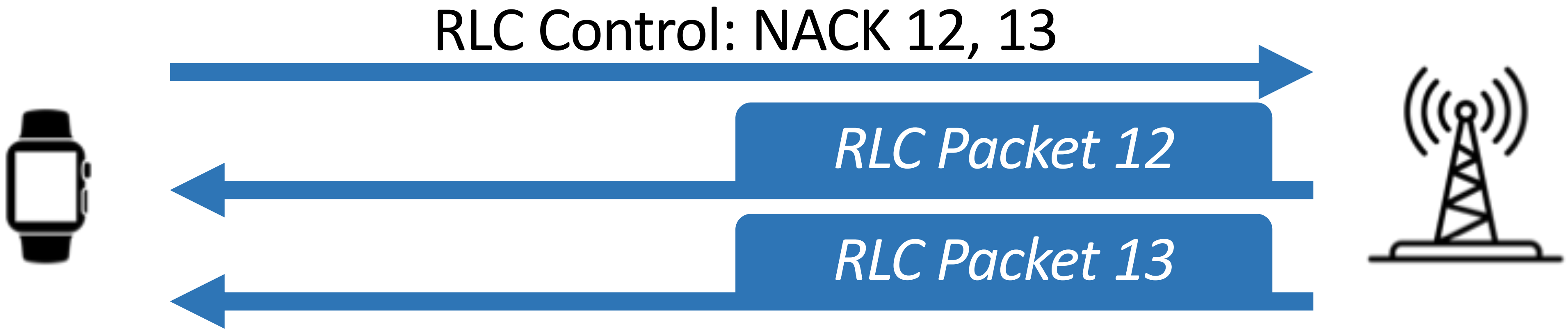
Attack: The attacker forges a BSR with large value

Damage: The BS schedules its limited C-IoT uplink resource to the attacker, blocking all other users' access



Packet Delivery Loop with RLC Control

RLC Control: A message that acknowledges or negative-acknowledges data specified with sequence number

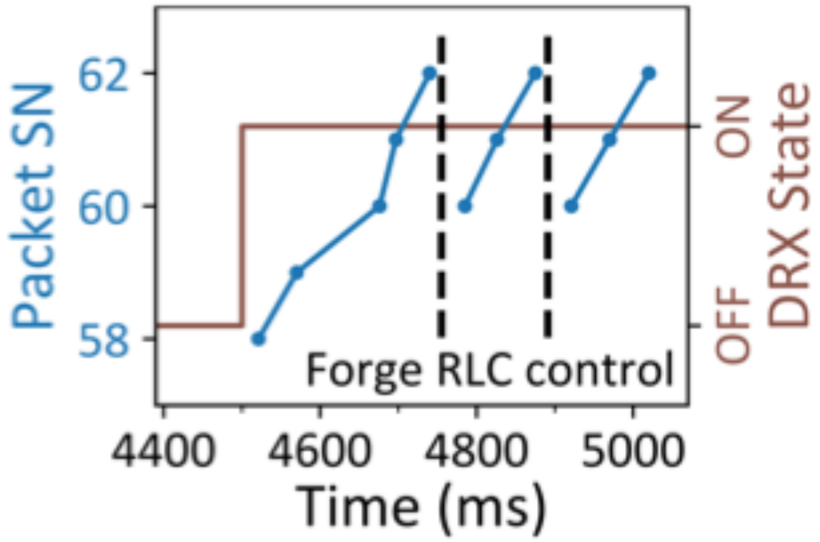


Packet Delivery Loop with RLC Control

RLC Control: A message that acknowledges or negative-acknowledges data specified with sequence number

Attack: The attacker forges RLC control with NACK

Damage: The victim consumes energy but cannot send or receive new data



Defense solution

- How to design a low-overhead solution without excessive cross-layer interactions?

Solution Idea for Protection

- The straightforward way to protect data-plane signaling is to encrypt and integrity protect it
- Generate keystream in MAC to prevent key-reuse

Challenge: No unique sequence number at MAC to generate the demanded keystream

Time-Based Protection with Low Overhead

Idea: use synchronized time clocks (1ms granularity) as parameter for securely generating keystream

Evaluate the Defense Solution

We prototype the solution in the testbed

Small Overhead!

- Reuse the proven EIA/EEA algorithm
- 3.6% amortized processing overhead
- 4B extra data for each signaling

Summary

- C-IoT is still vulnerable even after mutual authentication
  - The data-plane signaling is not well-protected
- We design attacks that can forge data-plane signaling and cause various attack damages
- Time-based defense to combat the threats

Thank you!