Bluetooth Security

Securing Wireless Networks COMP4337/9337

Never Stand Still

Dr. Hailun Tan
Postdoc fellow

School of Computer Science and Engineering, UNSW

Roadmap

- Introduction
- Features
- Security Issues
- Case Study – packet sniff
- Conclusions

Introduction

- Open wireless protocol for exchanging data over short distances from fixed and mobile devices, creating personal area network.
- A reliable wireless protocol for voice and data transmission

Bluetooth Evolution

- Bluetooth Special Interest Group (SIG)
- Founded in Spring 1998
- By Ericsson, Intel, IBM, Nokia, Toshiba
- Now more than 2,000 organizations have joined the SIG

Roadmap

- Introduction
- Features
- Security Issues
- Case Study – packet sniff
- Conclusions

Features

- Bluetooth-enabled devices can automatically locate each other
- Topology is established on a temporary and random basis
- Up to eight Bluetooth devices may be networked together in a master-slave relationship to form a piconet

Features (Cont.)

- One is master, which controls and sets up the network (piconet)
- Two or more piconet interconnected to form a scatter net
- Only one master for each piconet
- A device can't be masters for two piconet
- The slave of one piconet can be the master of another piconet

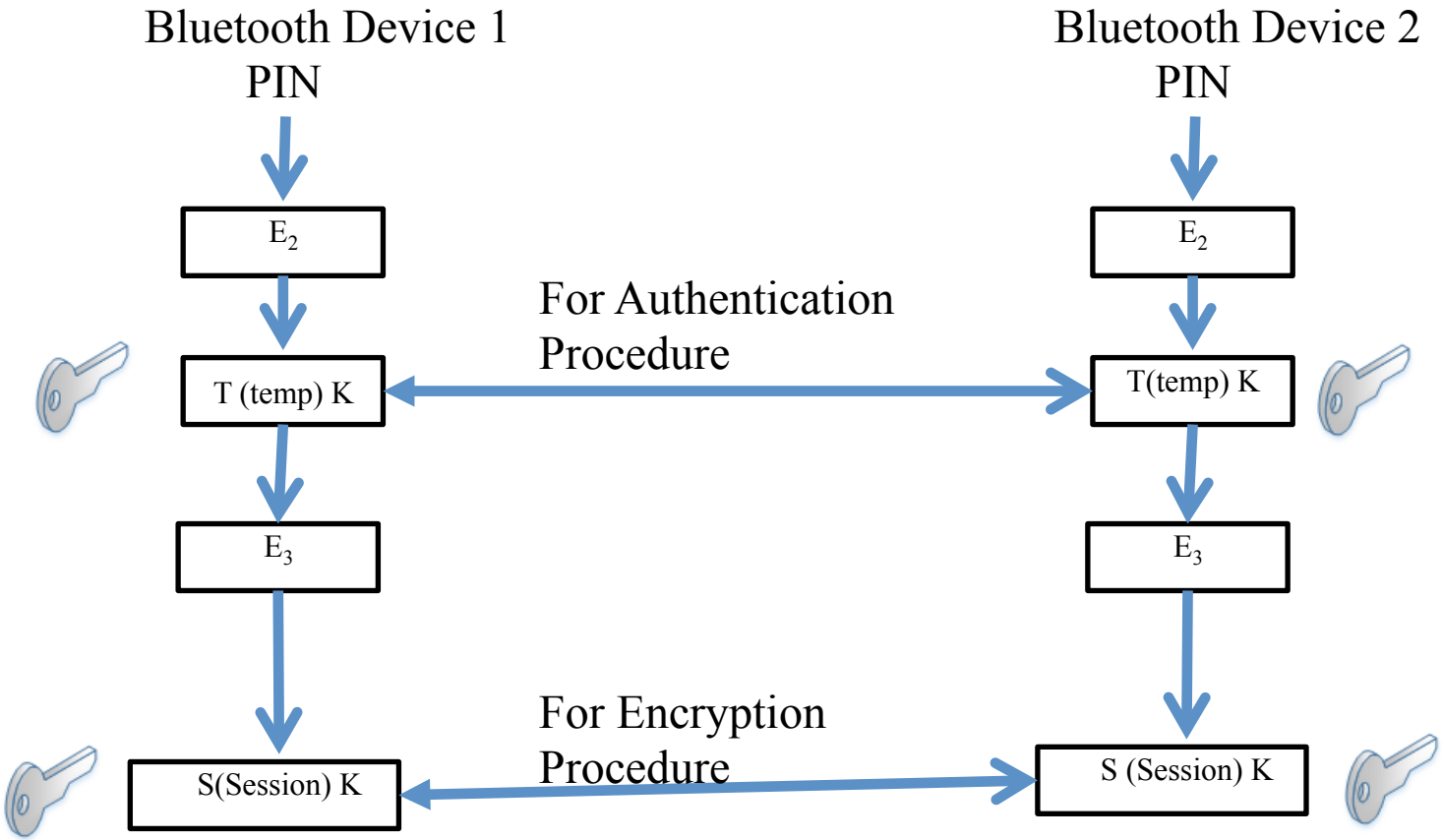
Roadmap

- Introduction
- Features
- Security Issues
- Case Study – packet sniff
- Conclusions

Security Issues

- Authenticity: Are you the device you claim you are?
  - Impersonation
- Confidentiality: Is the exchanged data only available to the intended devices?
  - Packet sniffing
- Authorisation: Are only the intended devices accessing the specified data and control?
  - Prerequisite: authenticity and confidentiality

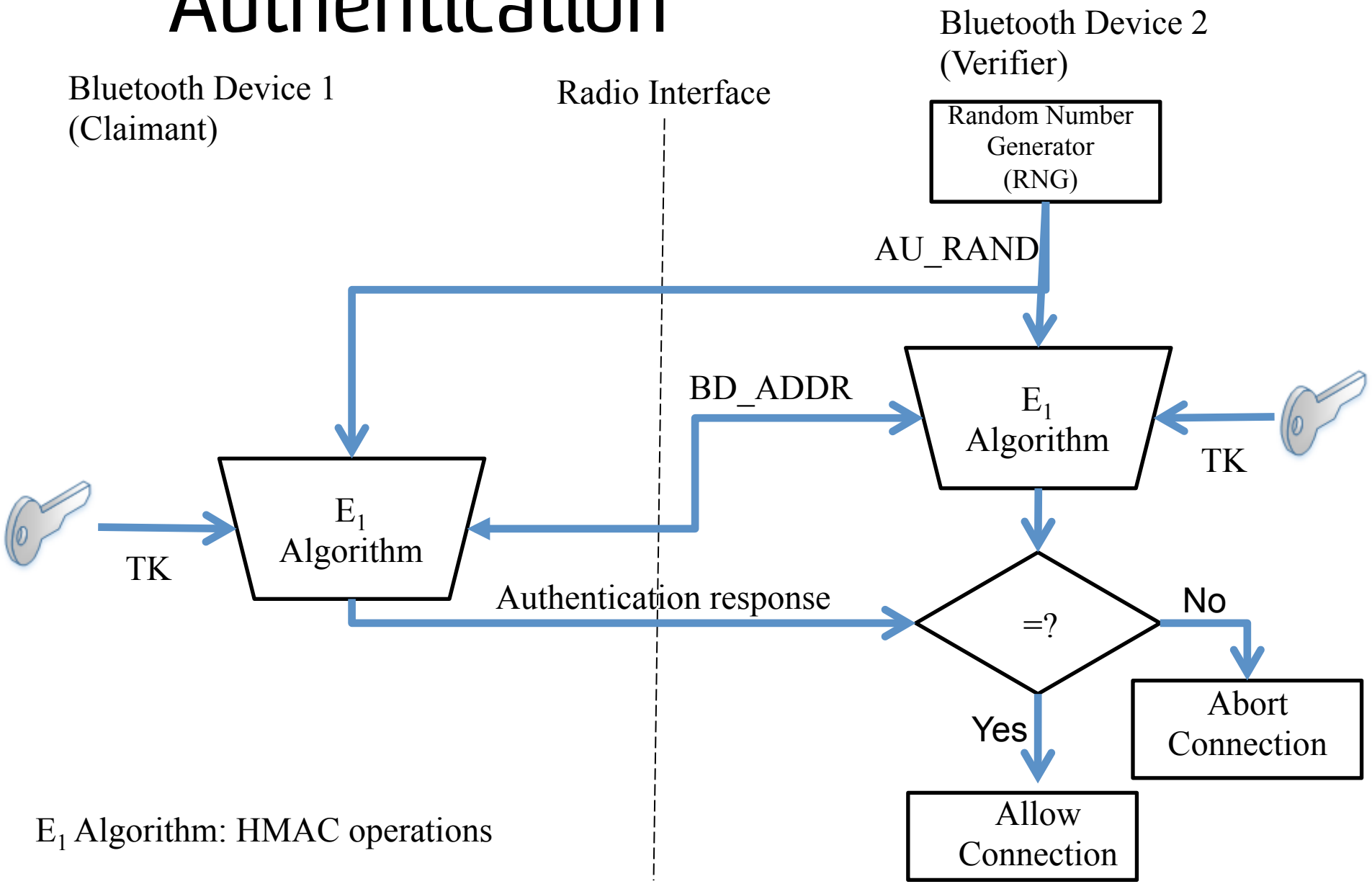
Temporary Key Generation



E_1: HMAC algorithm
E_2: it would be discussed in case study
E_3: it is described in slide 13.

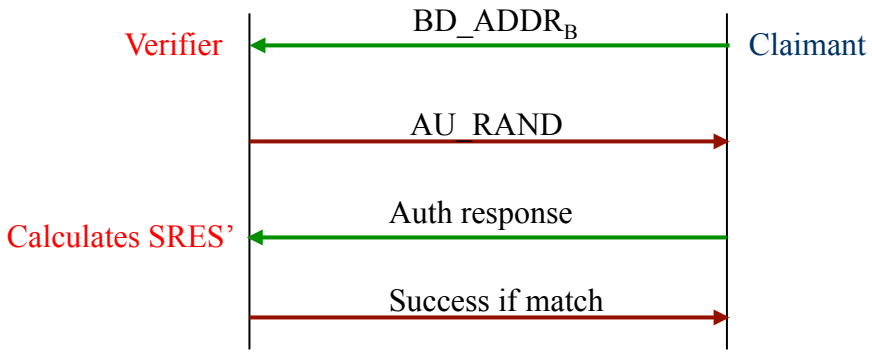
-

Authentication



E_1 Algorithm: HMAC operations

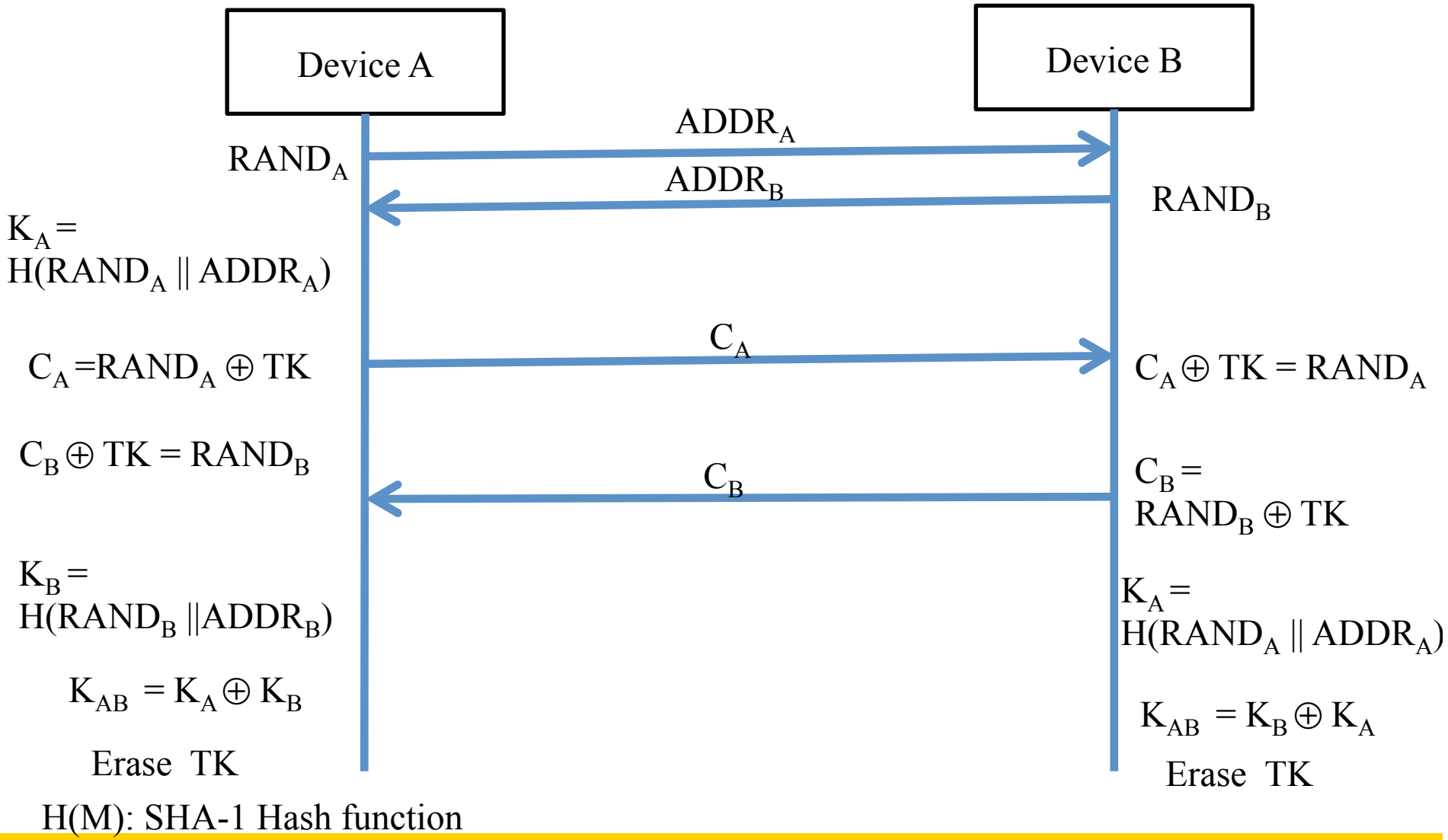
Authentication Summary



Authentication Process

Parameter	Length	Secrecy parameter
Device Address	48 Bits	Public
Random Challenge	128 Bits	Public
Authentication(Auth) Response	32 Bits	Public
Temporary Key	128 Bits	Secret

Session Key Generation



Is Channel Hopping Secure?

- Channel Hopping (Bluetooth Smart only) – Both communication parties would hop to a different wireless channel per packet in a fixed channel hopping increment.
- Adversary could not achieve data by monitoring one wireless channel only.
- We will study its vulnerability soon.

Roadmap

- Introduction
- Features
- Security Issues
- Case Study – packet sniff
- Conclusions

Case study: Bluetooth Low Energy (BTLE)

- Introduced in Bluetooth 4.0 (2010)
- New modulation and link layer for low power devices
  - Incompatible with classic Bluetooth devices
  - PHY and link layer different (no channel hopping in classic Bluetooth)
  - High-level protocols reused (L2CAP, ATT)

BTLE applications

- High end smart phones
- Sports/fitness devices
- Door locks
- Upcoming medical devices (e.g., blood glucose monitor)

BTLE Protocol review

GATT (Generic Attribute Profile) – how to discover and provide services based on ATT
ATT (Attribute protocol) – how to discover/read/write attributes on a peer device
L2CAP (Logical Link Control and Adaptation Protocol) – packet segmentation and reassemble
Link Layer
Physical Layer

- We will focus on Link layer and Physical layer security only.
- Other layers are similar to their counterparts in wired networks.

Physical Layer

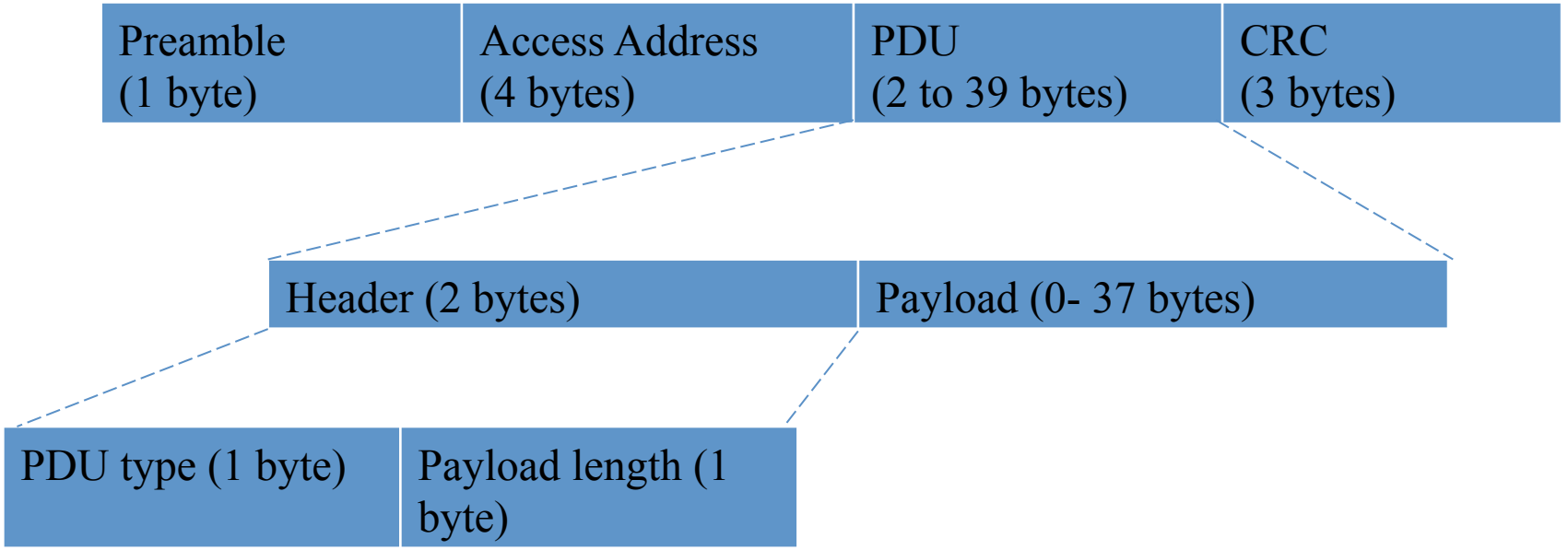
- Physical layer: channels for hopping (40 available channels in 2.4Ghz)
  - Advertising: 3 channels
  - Data: 37 channels

Channel Hopping

- Hop along 37 data channels
- One data packet per channel
- Next channel = current channel + hop increment (mod 37)
- Time between hops: hop interval, it is the duration when both communication parties stays in one channel. It is equal to one Round Trip Time + channel switch latency.

3→10→17→24→31→1→8→15→…(hop increment = 7)

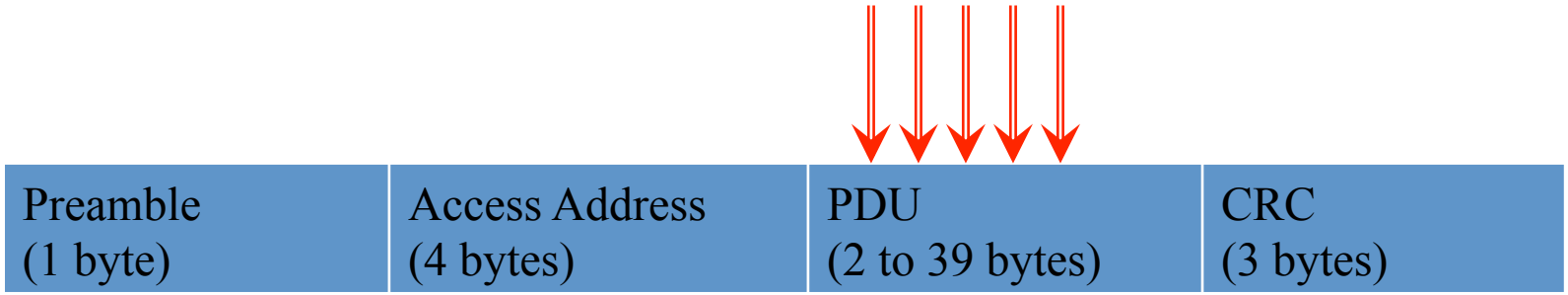
Link Layer



- Only PDU encrypted which creates security vulnerability
  - packet sniff to break the confidentiality in PDU.

Encryption and MACs

- Encrypts and MACs PDU section
- AES-CCM algorithm
- AES-CCM is secure but the key exchange protocol is weak!



Packet Sniff Process

- Configuration
  - Set modulation parameters to match BTLE (e.g., set to the same frequency, 2.4GHz)
  - Tune to proper channel – it needs to know the channel hopping pattern.
    - Hop Increment
    - Hop interval
    - Both can be sniffed from connection packet or recovery in promiscuous mode

What Information do we need?

Preamble (1 byte)	Access Address (4 bytes)	PDU (2 to 39 bytes)	CRC (3 bytes)

- Access Address (AA)
  - Advertising: Fixed 0x8E89BED6
  - Connection: Actual device address
- Channel Information:
  - Hop interval
  - Hop increment

Where to get this info: Connection packet!

- easy if you get the starting packets

Promiscuous mode

- What if I missed the connection packets?
  - Capture a number of data packets.
  - Perform the pattern search (promiscuous mode) to recover access addresses, hop interval and hop increment values (easily done!)
- Crack the session key to decrypt PDU

Recovery of Access Address

Preamble (1 byte)	Access Address (4 bytes)	PDU (2 to 39 bytes)	CRC (3 bytes)

What we know: Preamble (01010101)

What we have: Sea of bits

What we want: Access Address

10001110111101010101 → likely preamble!

10011100000100011001.. -> part of AA

100011001…100011101 →32 bit complete of AA! After that, PDU!

Recovery of Access Address (Cont.)

- A preamble is “01010101” but “01010101” is not always a preamble.
- CRC is here to help (for attacker)!
- Attacker could use CRC (after PDU) to verify the access address and PDU.
  - If CRC passes, the access address is correct. Otherwise, the “01010101” is false positive for preamble.

Recovery of Hop Interval

- Observation: 37 is a prime

- Sit on one data channel and wait for two consecutive packets. Measure the time difference.

$$\Delta t/37 = \text{hop interval}$$

Recovery of Hop Increment

- Start on data channel 0, jump to data channel 1 when a packet arrives.
- We know hop interval, we can calculate how many channels have been hopped between channel 0 and 1.
  - Δt/hop interval = channel hops

Calculate Hop Increment

$$\text{HopIncrement} * \text{channel hops} \equiv 1 \pmod{37}$$

$$\text{HopIncrement} \equiv \text{channel hops}^{-1} \pmod{37}$$

Apply Fermat's little theorem : $a^{p-1} \equiv 1 \pmod{p}$,

$$\text{HopIncrement} \equiv \text{channel hops}^{37-2} \pmod{37}$$

Sniff summary

- Connections packets
- Promiscuous mode: recovery of
  - Access Address
  - Hop Interval
  - Hop Increment

Custom Key exchange protocol

- Three pairing methods
  - Just Works™
  - 6-digit PIN
  - 00B
  - "None of these key pairing methods provide protection against a passive eavesdropper" – Bluetooth Core Spec

Cracking Temporary Key

- Temporary Key (TK) = AES (PIN, AES(PIN, rand XOR p1) XOR p2) (E_2 in slide 10)

Green – transmitted in plaintext

Red – wanted to know

PIN: integer between 0 and 999,999

JustWork™ is always 0!

Cracking the PIN

Total Time to crack:

< 1 second

Subsequent Key crack

PIN → STK

STK → LTK

LTK → Session key!

- Every key is known.
- Attacker can learn about PDU
- Attacker can inject the packets in the networks with the session key!

Roadmap

- Introduction
- Features
- Security Issues
- Case Study – packet sniff
- Conclusions

Conclusions

- Bluetooth has some security mechanisms
- Bluetooth is not secure. There exist loopholes and they are easy to exploit.
- Security in Bluetooth is yet to be improved.

Thank you!

Any Questions?