Opening Doors and Stealing Cars: Bluetooth LE Link Layer Relay Attacks

Sultan Qasim Khan

October 27, 2022 – Hardwear.io NL 2022

About me

- Principal Security Consultant at NCC Group, one of the largest security consultancies in the world
- Part of Hardware and Embedded Systems practice
- Based in Waterloo, Ontario, Canada
- Creator of Sniffle, a Bluetooth 5 LE sniffer
- Bluetooth geek, auto enthusiast, and general embedded systems nerd

Proximity Authentication

What is it and what are the risks?

Proximity Authentication

- Purpose of such systems is to authenticate users when they are nearby and in possession of a trusted object
- Achieved through wireless communications between the trusted object and the device that is being unlocked/logged into/started
- Automotive keyless entry is the most common and obvious application of such technology
- Similar technology used for many other applications too, such as smart locks, building access control systems, and some laptops and mobile phones

Distance Measurement

- RSSI-based ranging
- Time-of-Flight (ToF) based ranging
- Multi-carrier phase-based ranging
  - Measure distance based on phase difference between tones transmitted at two or more different carrier frequencies
- Angle-of-Arrival (AoA) measurement
  - Determines AoA by examining phase difference in a single signal received by multiple antennas
- Multi-receiver triangulation
  - Any of the above methods can be used with multiple receivers to triangulate the source direction, better estimate distance, and identify discrepancies

Distance Measurement Attacks

- Replay attacks
  - Replay previously recorded authentication signals from nearby, if authentication scheme is not replay-resistant
- Relay attacks
  - Bridge communications between devices to fool them into thinking they are close to each other
- Responding from nearby to time-of-flight pings that are not part of a cryptographic challenge-response scheme
- Retransmitting ranging signals with shifted phase to defeat multi-carrier phase-based ranging
- Time-of-flight signal confusion (Early Detect/Late Commit, Ghost Peak)

Bluetooth LE Proximity Authentication

- Infers proximity based on the ability to maintain a BLE connection and communicate over BLE while remaining within reasonable latency bounds
- Used by many products
    - Cars: Tesla and many other major brands
    - Smart locks: Weiser/Kwikset Kevo and many others
    - Mobile phones: Android Trusted Devices
    - PCs: Chromebook Smart Lock and others
    - Home alarm systems, commercial building access control systems, hospital patient tracking systems, and countless other products

BLE Proximity Authentication Issues

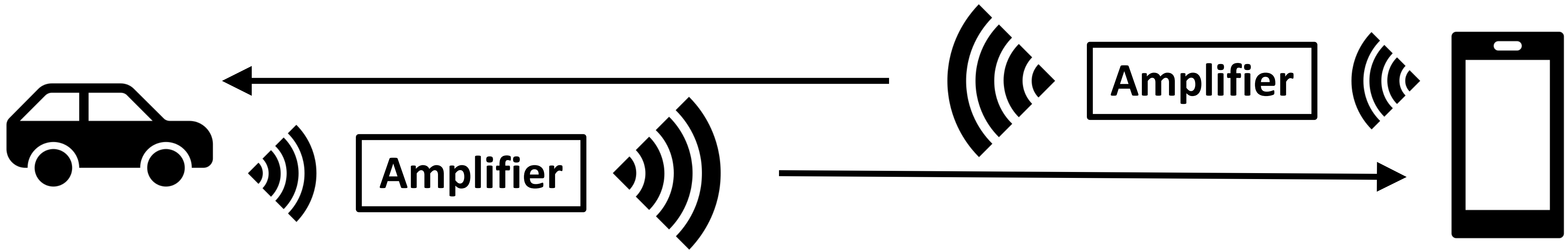
- Standard Bluetooth LE (currently) does not provide any means for time-of-flight measurement or multi-carrier phase-based ranging
- Distance must be determined by RSSI and/or AoA triangulation
- Vulnerable to relay attacks
- Bluetooth SIG states that: "*two devices can be spoofed into assuming that the other device is close*" and BLE proximity auth "*should not be used as the only protection of valuable assets*" (Proximity Profile v1.0.1, page 17)
- Companies use BLE for proximity authentication despite known security risks because it's ubiquitous and convenient

Relay Attacks

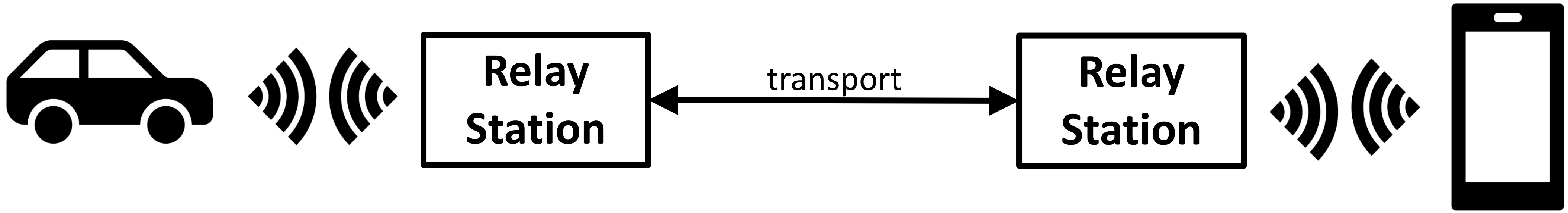
Attack variations and how they work

Relay Attack Types

- Signal amplification



- Transport and retransmit



Analog Signal Amplification

- Benefits:
  - Simple hardware
  - Ultra-low latency
  - Communications patterns and response timing remains unmodified, except for very subtle latency added due to the speed of light
- Drawbacks:
  - Limited relay distance
  - Protocol timing requirements may be violated at long distances
  - Handling of bidirectional communications on the same frequency can get complicated
  - May be detected by RSSI or AoA triangulation systems

Analog Transport and Retransmit

- Analog signals received on one frequency, transported (over a different carrier), and retransmitted at the other end back at the original frequency
- Benefits:
  - Ultra-low latency
  - Communications patterns and response timing remains unmodified, except for very subtle latency added due to the speed of light
  - Circumvents RSSI and AoA triangulation defenses
- Drawbacks:
  - More complex hardware than signal amplification
  - Analog transport still has distance limits
  - Protocol timing requirements may be violated at long distances
  - Handling of bidirectional communications on the same frequency can get complicated

Digital Transport and Retransmit

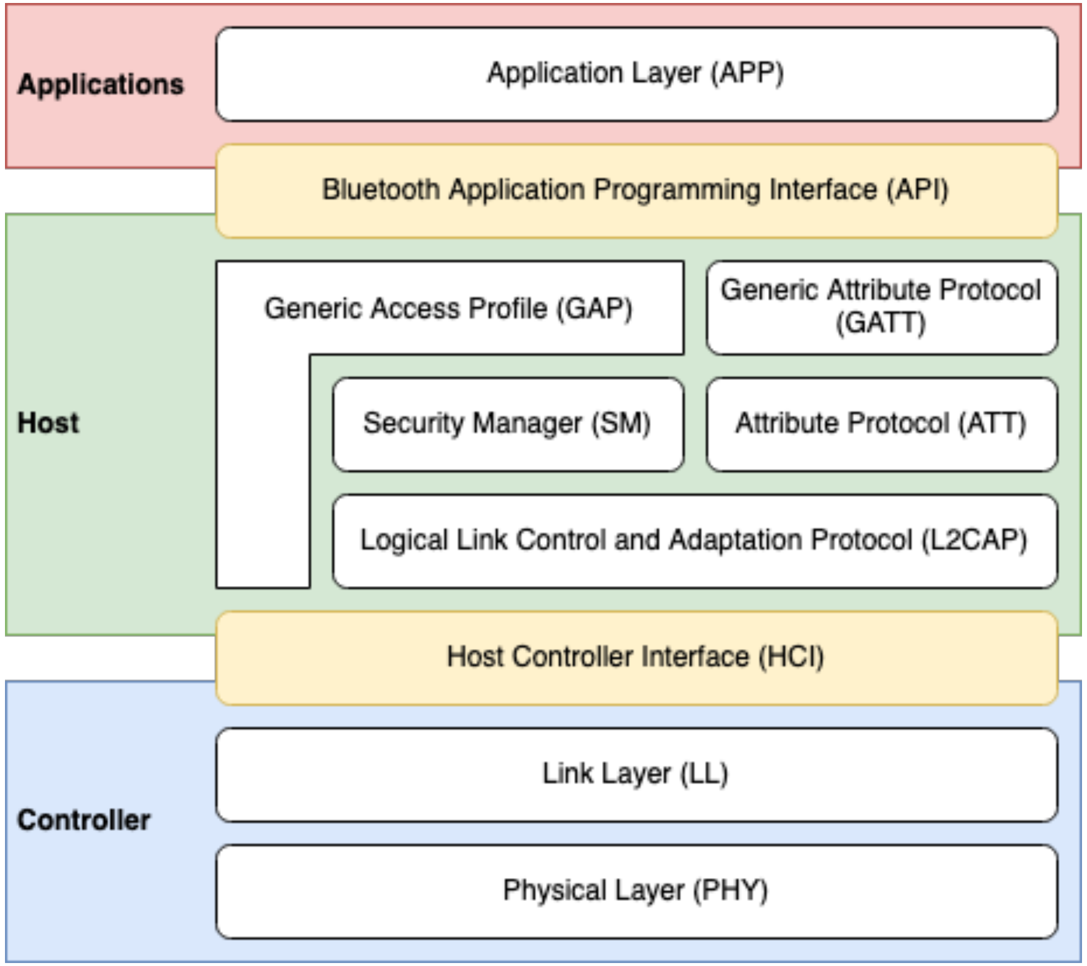
- Forwarding digital packets/frames, rather than raw analog signals
- Can be done at any layer of the protocol stack, but with different limitations at each layer depending on the protocol being relayed
- Benefits:
  - Distance only limited by latency bounds (internet transport possible)
  - Low bandwidth requirements
  - Can also perform MITM-style packet injection, modification, or dropping
- Drawbacks:
  - Higher latency
  - Response timing and device behavior may not precisely match unrelayed operation

Types of BLE Relay Attacks

Different ways to bridge BLE connections

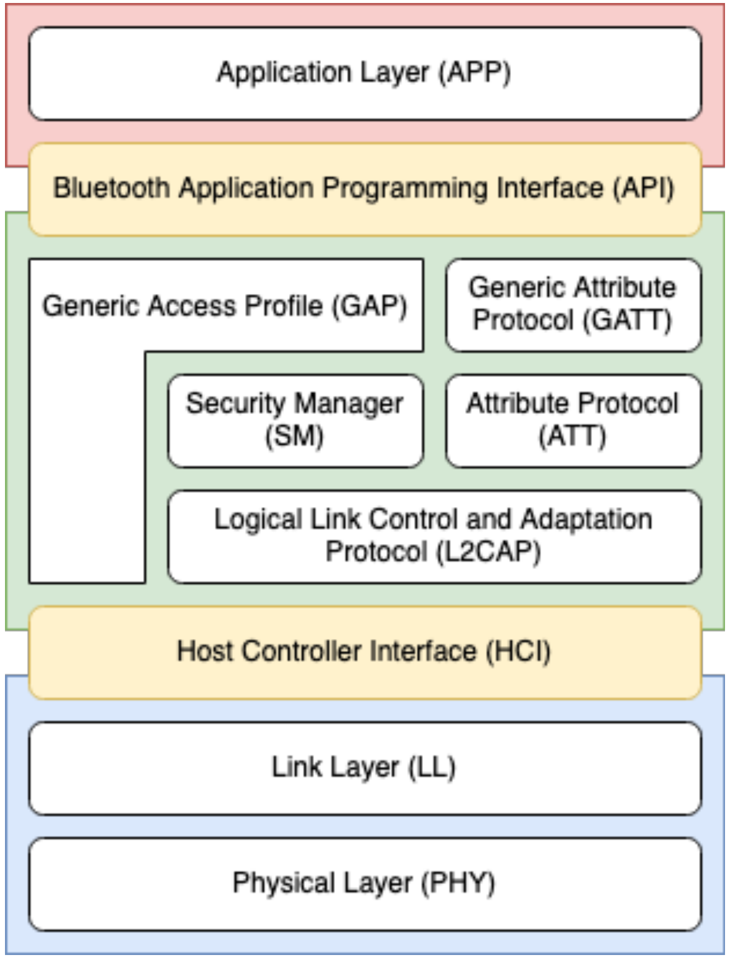
Bluetooth LE Stack

- Communications at and above the link layer can optionally be encrypted and decrypted by the controller
- L2CAP provides multiplexing of higher-level protocols, packet segmentation and reassembly, and QoS management
- GATT structures application communications into services containing characteristics that can be read, written to, and/or provide notifications
- GAP defines procedures for advertising, establishing connections, pairing/bonding


Applications
Application Layer (APP)
Bluetooth Application Programming Interface (API)
Host
Generic Access Profile (GAP)
Generic Attribute Protocol (GATT)
Security Manager (SM)
Attribute Protocol (ATT)
Logical Link Control and Adaptation Protocol (L2CAP)
Host Controller Interface (HCI)
Controller
Link Layer (LL)
Physical Layer (PHY)

Prior BLE Relay Attacks

- GATT relays
  - Forwarding GATT requests and responses, above standard Bluetooth stack
  - First shown in 2016
  - Gattacker by Slawomir Jasek
  - Btlejuice by Damien Cauquil
- HCI relay: Mirage by Romain Cayre
  - Forwarding of HCI messages from standard Bluetooth controllers, bypassing standard Bluetooth stacks and allowing relaying of non-GATT data
- Analog (PHY) relay (Staat et al., 2022)
  - Analog amplification of entire 2.4 GHz band with direction switching through transmit detection


Application Layer (APP)
Bluetooth Application Programming Interface (API)
Generic Access Profile (GAP)
Generic Attribute Protocol (GATT)
Security Manager (SM)
Attribute Protocol (ATT)
Logical Link Control and Adaptation Protocol (L2CAP)
Host Controller Interface (HCI)
Link Layer (LL)
Physical Layer (PHY)

Limitations of Prior BLE Relays

- GATT relays
  - Clearly detectable GATT response latency
  - No support for link layer encryption with unknown LTK
- HCI relay
  - No support for link layer encryption with unknown LTK
  - Detectable added latency
- Analog relay
  - Distance limits
- We can combine many of the benefits of HCI and analog relays while avoiding their drawbacks by relaying PDUs at the layer in between: the link layer

Link Layer Summary

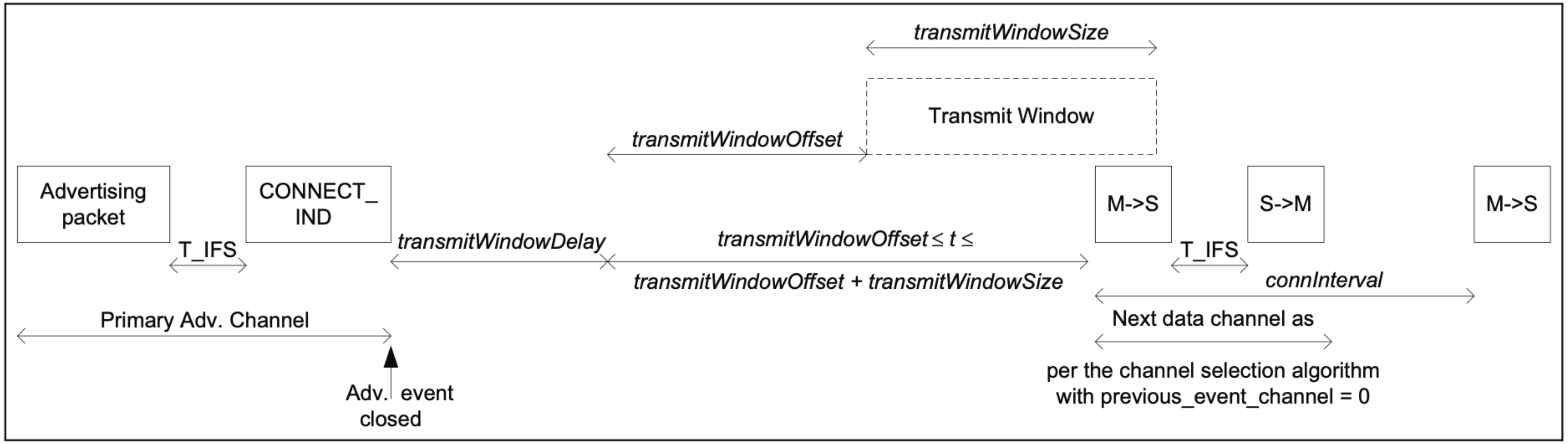
How does the BLE link layer work? How can it be relayed?

BLE Physical Layer

- Operates within 2.4 GHz unlicensed band
  - Divided into 40 channels, 37 data and 3 primary advertising
- Uses Gaussian Frequency Shift Keying (GFSK)
- Time Division Multiple Access (TDMA) for bidirectional communications and coexistence with other devices
- Three PHY modes supported in Bluetooth 5
  - Original 1M mode
  - New fast 2M mode
  - New long range error correcting coded PHY

BLE Link Layer Operation

- Connections frequency hop over 37 data channels in a predictable sequence
- Time duplexed bidirectional comms, with alternating Master->Slave and Slave->Master frames during connection events
- 150 us inter-frame separation (T_IFS) between alternating frame directions



Bluetooth Core Specification v5.2, page 2984

BLE Link Layer Operation (Contd.)

- Frames contain sequence numbers for acknowledgement, with retransmission occurring for frames that fail to be acknowledged
- Connection events end when either both sides declare no more data to transmit, master decides not to transmit, or timeout occurs
- After a connection event ends, no more data can be sent until the next event
- Next connection event begins after connection interval elapses
- Switch to next channel in hop sequence for next connection event

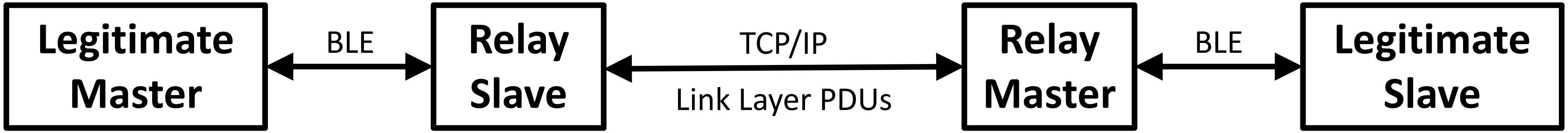
Link Layer Relaying

How does it work? What can it do?

Why Relay at the Link Layer?

- A link layer relay forwards link layer PDUs
- Digital relay
  - Distance only bounded by transport latency
- Allows relaying connections with link layer encryptions
- Allows control over certain low level timing parameters to help reduce or mask latency
- Allows injection or selective dropping of link layer PDUs (with some limitations for encrypted connections)
- Circumvents traditional Bluetooth controller firmware and Bluetooth stacks, aiding latency reduction

Link Layer BLE Relaying Process



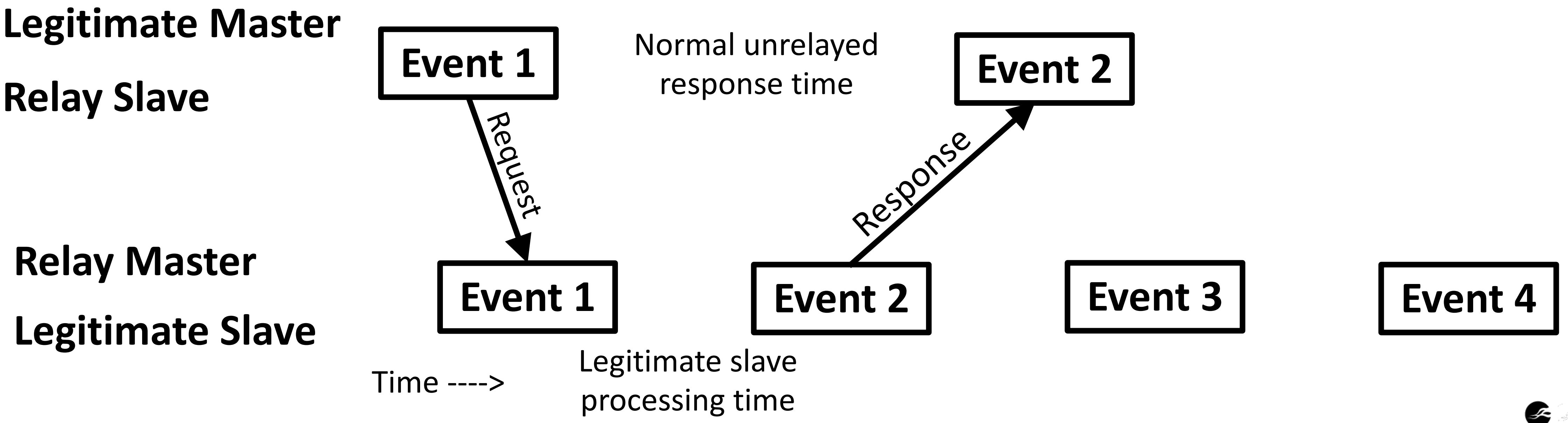
1. Relay master captures advertisements of legitimate slave
2. Relay slave mimics advertisements of legitimate slave
3. Legitimate master connects to relay slave
4. Relay slave notifies relay master of incoming connection
5. Relay master connects to legitimate slave
6. Relay master and slave forward link layer PDUs

Latency and Timing Control

- Latency can be reduced or masked through control of connection event Interval and transmit window offset
- Link layer relays can fully control these parameters for the relay master, and influence these parameters for the relay slave (through LL_CONNECTION_PARAM_REQ)
- Default connection intervals: 30 ms for iOS, 48.75 ms for Android
- Changes to the connection interval are not directly visible at the application layer on iOS or Android

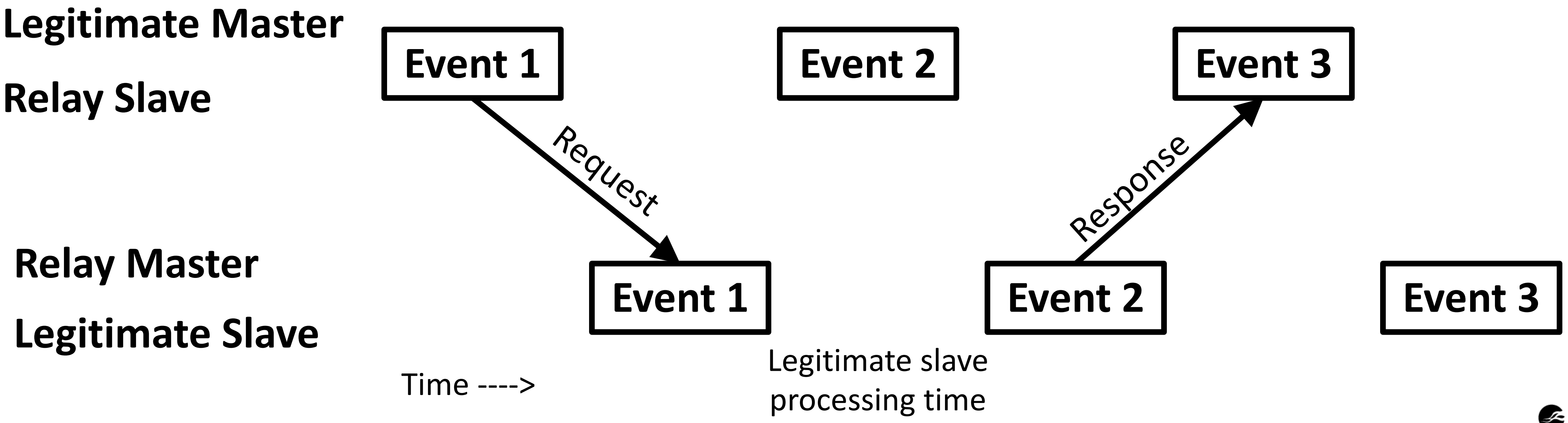
Latency Reduction Techniques

- Selecting a fast connection interval (such as the 7.5 ms minimum) can make relayed latency less than unrelayed operation
- Setting up unequal connection intervals between the two sides of the relay can allow timing identical to unrelayed operation, as shown below



Connection Event Staggering

- If connection intervals cannot be modified, staggering connection events between relay master and relay slave can at least limit added round trip latency to a single connection event
- Added latency is within the range of normal timing variation



Dealing with Encryption

Why does link layer encryption make relay attacks more difficult? How can we overcome these difficulties?

BLE Link Layer Encryption

- AES-CCM encryption (counter mode with CBC-MAC)
- Unique session key derived for each connection using nonces from both master and slave devices and previously negotiated Long Term Key (LTK)
- Counters to prevent replay, injection, or reordering of packets
- Both data and control PDUs are encrypted
- While there have been issues with pairing mechanisms in the past, the actual encryption and authentication scheme used by BLE has withstood the test of time and is generally considered solid
- Empty PDUs are not encrypted and don’t influence packet counters
- Sequence numbers for acknowledgement and retransmit are also not encrypted or authenticated

Difficulties from Link Layer Encryption

- Link layer control PDUs (that indicate connection event timing changes, PHY mode changes, and channel map changes) are encrypted, so you don’t know when, on which channel, and which PHY mode to listen or transmit
    - Thus past digital BLE relays have been unable to relay connections with (unknown keyed) link layer encryption
- Can’t inject or drop PDUs once link layer encryption is active
- Unequal connection intervals between the two sides of the relay would cause instants in control PDUs to go out of sync, and we can’t modify or correct instants in encrypted/authenticated PDUs

Relaying Encrypted Connections

- Empty PDU injection and retransmission will not affect encryption
- We can figure out connection parameters through a combination of traffic analysis, heuristics, observation of past unrelayed behaviour, and some link layer trickery
- Connections start off unencrypted, and there is a short window where we can control/influence connection intervals and event timing before encryption starts
- If we quickly switch both sides of the relay to a quicker connection interval, instants in control PDUs won’t go out of sync, so no control PDU dropping or modification would be needed

Figuring Out Connection Parameters

- Encrypted connection update PDUs can be uniquely identified by their LLID field and length in the unencrypted link layer data header
- Prior research by Stas Mekinulashvili [1] found that new connection parameters almost always take effect 6 or 7 connection events from the time of the update PDU being sent
- PHY update PDUs are almost universally a switch from 1M to 2M PHY
- When the channel map changes, sniff all the channels to figure out the new map by seeing which are used by the legitimate master
    - Hop sequence for channels not excluded from map remains unchanged, and when combined with retransmit functionality, risk of losing connection during channel map recovery is low

[1] Mekinulashvili, Stanislav. Sniffing encrypted BLE traffic after changing connection parameters, using low-cost hardware that captures only one channel at a time. May 27, 2021. https://digikogu.taltech.ee/en/Item/b0dfa43a-f1a9-4946-b108-d0baeb79f806

Figuring Out Connection Event Timing

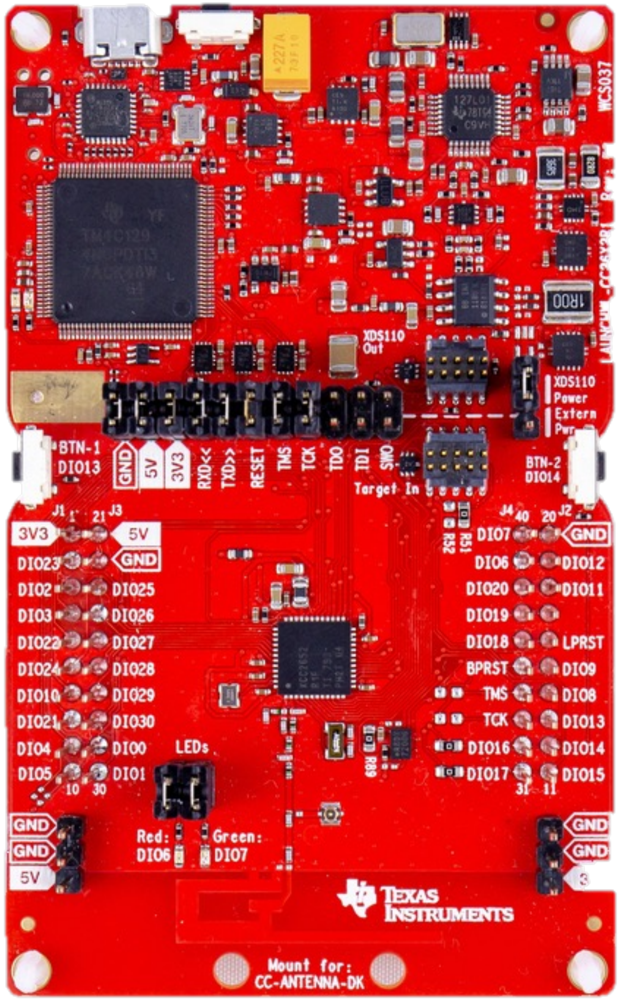
- Relay slave can immediately hop to next channel when last connection event ends, and just wait to see when legitimate master transmits to figure out the new connection interval
- Trickier for relay master, as it must transmit at the right time before the legitimate slave, however there is a solution:
  - Changes to connection intervals tend to be consistent/predictable
  - Changes to WinOffset tend to be random, but there are a limited number of possible discrete integer offsets
  - Relay master can transmit an empty PDU at every possible WinOffset, and see at which offset it gets a response from the legitimate slave

Putting It All Together

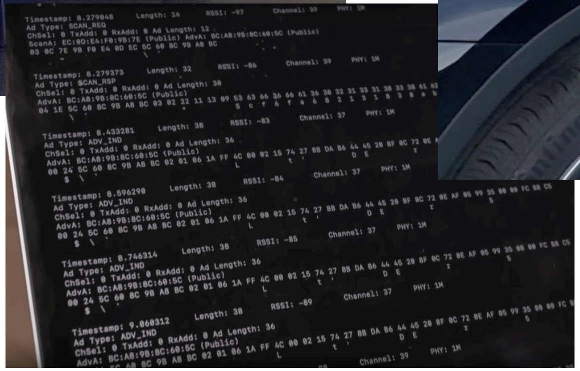
Implementing the BLE link layer relay, ways to defend against such attacks, and things to keep in mind

Sniffle Relay

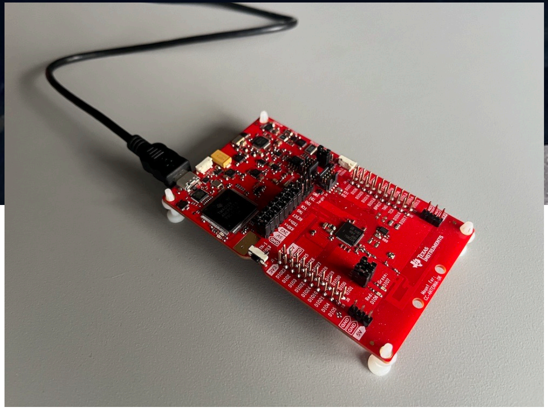
- Extends the Sniffle firmware and software to perform link layer relaying of BLE
- Uses the same Texas Instruments CC2652/CC1352 hardware as Sniffle, my Bluetooth 5 sniffer
- Implements the previously described techniques for latency minimization and working with link layer encryption
- We are not releasing the tool to the public at this time due to the risk of misuse



Demos



```
Timestamp: 8.279048    Length: 14    RSSI: -97    Channel: 39    PHY: 1M
Ad Type: SCAN_REQ
ChSel: 0 TxAdd: 0 RxAdd: 0 Ad Length: 12
ScanA: EC:0D:E4:F0:9B:7E (Public) AdvA: BC:A8:9B:8C:60:5C (Public)
03 0C 7E 9B F0 E4 0D EC 5C 60 8C 9B A8 BC
   ~                \

Timestamp: 8.279373    Length: 32    RSSI: -86    Channel: 39    PHY: 1M
Ad Type: SCAN_RSP
ChSel: 0 TxAdd: 0 RxAdd: 0 Ad Length: 30
AdvA: BC:A8:9B:8C:60:5C (Public)
04 1E 5C 60 8C 9B A8 BC 03 02 22 11 13 09 53 63 66 36 66 61 36 38 32 31 33 31 38 33 38 61 62
      \                 "                 S  c  f  6  f  a  6  8  2  1  3  1  8  3  8  a  b

Timestamp: 8.433281    Length: 38    RSSI: -83    Channel: 37    PHY: 1M
Ad Type: ADV_IND
ChSel: 0 TxAdd: 0 RxAdd: 0 Ad Length: 36
AdvA: BC:A8:9B:8C:60:5C (Public)
00 24 5C 60 8C 9B A8 BC 02 01 06 1A FF 4C 00 02 15 74 27 8B DA B6 44 45 20 8F 0C 72 0E A
   $  \                                L              t  '           D  E           r

Timestamp: 8.596290    Length: 38    RSSI: -84    Channel: 37    PHY: 1M
Ad Type: ADV_IND
ChSel: 0 TxAdd: 0 RxAdd: 0 Ad Length: 36
AdvA: BC:A8:9B:8C:60:5C (Public)
00 24 5C 60 8C 9B A8 BC 02 01 06 1A FF 4C 00 02 15 74 27 8B DA B6 44 45 20 8F 0C 72 0E AF 05 99 35 00 00 FC B8 C5
   $  \                                L              t  '           D  E           r                 5

Timestamp: 8.746314    Length: 38    RSSI: -85    Channel: 37    PHY: 1M
Ad Type: ADV_IND
ChSel: 0 TxAdd: 0 RxAdd: 0 Ad Length: 36
AdvA: BC:A8:9B:8C:60:5C (Public)
00 24 5C 60 8C 9B A8 BC 02 01 06 1A FF 4C 00 02 15 74 27 8B DA B6 44 45 20 8F 0C 72 0E AF 05 99 35 00 00 FC B8 C5
   $  \                                L              t  '           D  E           r                 5

Timestamp: 9.060312    Length: 38    RSSI: -89    Channel: 37    PHY: 1M
Ad Type: ADV_IND
ChSel: 0 TxAdd: 0 RxAdd: 0 Ad Length: 36
AdvA: BC:A8:9B:8C:60:5C (Public)
   5C 60 8C 9B A8 BC 02 01 06 1A FF 4C 00 02 15 74 27 8B DA B6 44 45 20 8F 0C 72 0E AF 05 99 35 00 00 FC
                                    L              t  '           D  E           r                 5
                                                                        Channel: 39    PHY: 1M
```



Mitigation Approaches

- Requiring user interaction to authorize unlock
- Secure ranging
  - Combines time-of-flight measurement with cryptographic challenge/response
  - Several radio standards can implement it, including Ultra Wide Band (802.15.4z)
  - Note that secure ranging is not bulletproof – attacks against it such as Early Detect/Late Commit and Ghost Peak exist
- Checking GPS location
  - May not always be available, may have issues with speed and power consumption
- Disabling proximity authentication functionality when phone or key fob has been stationary for a while according to accelerometer

Key Takeaways

- BLE relay attacks are practical and easy to carry out with low-cost hardware
- RSSI alone should not be relied upon as proof of proximity
- BLE link layer encryption does not prevent relay attacks
- The latency of relayed BLE communications can be made to match unrelayed communications
- The current version of the Bluetooth specification does not provide suitable mechanisms for secure ranging
- It is possible to combine BLE data transfer with secure ranging using a different protocol (such as UWB), as done by CCC Digital Key 3.0

Questions?