BLURtooth: Exploiting Cross-Transport Key Derivation in Bluetooth Classic and Bluetooth Low Energy

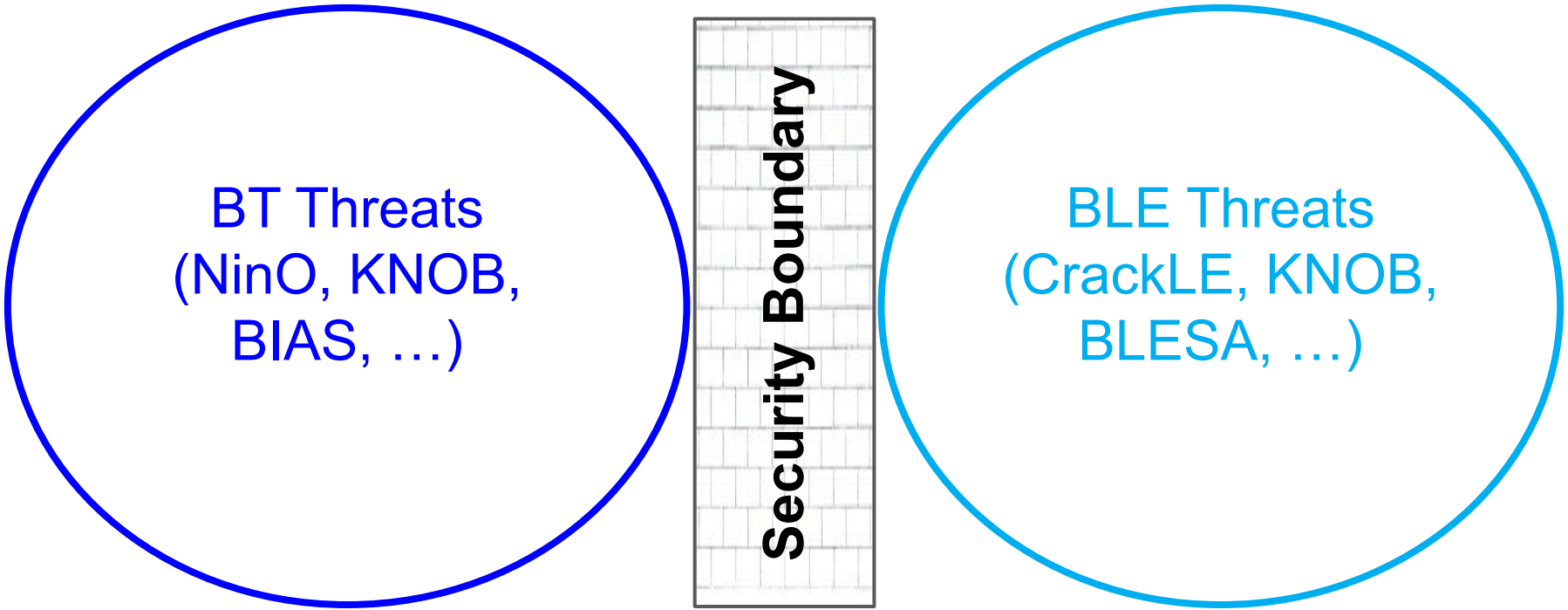
ACM AsiaCCS'22

Daniele Antonioli (EURECOM and EPFL)
Nils Ole Tippenhauer (CISPA)
Kasper Rasmussen (University of Oxford)
Mathias Payer (EPFL)

Bluetooth is a Pervasive Wireless Technology

- Bluetooth Classic (BT)
  - High throughput services
- Bluetooth Low Energy (BLE)
  - Ultra low power services
- Bluetooth standard (v5.3)
  - One vulnerability in the standard
  - Billions of exploitable devices

BT and BLE Security Are Considered Separately



We Blur the Security Boundary abusing CTKD

BT Threats
(NinO, KNOB,
BIAS, …)

Security Boundary

BLE Threats
(CrackLE, KNOB,
BLESA, …)

We perform Cross-Transport Attacks on BT and BLE

BT Threats
(NinO, KNOB
BIAS, …)

NEW: BT-BLE
Cross-Transport
Threats (BLUR)

E Threats
LE, KNOB,
ESA, …)

Contributions

- CTKD is a **novel** and **cross-transport** attack surface
- Uncover **four vulnerabilities** in the CTKD **specification**
- Develop **four cross-transport (BLUR) attacks**
  - Cross-transport Impersonation, MitM, unintended sessions
- **Conduct** the BLUR attacks on actual devices
  - Exploit 16 devices (14 chips, Bluetooth 4.1, 4.2, 5.0, 5.1, 5.2)
- **Fix** the BLUR attacks
  - Unlike the mitigation in the Bluetooth standard

Device Discovery and Pairing Initialization

Central

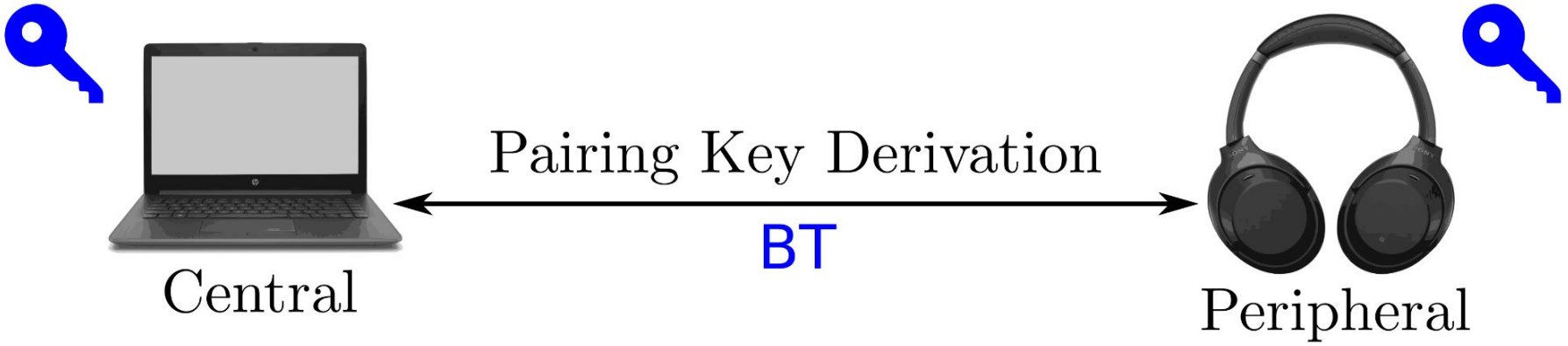
Peripheral

Victims support BT, BLE and CTKD.
They start pairing over BT

Pairing Feature Exchange

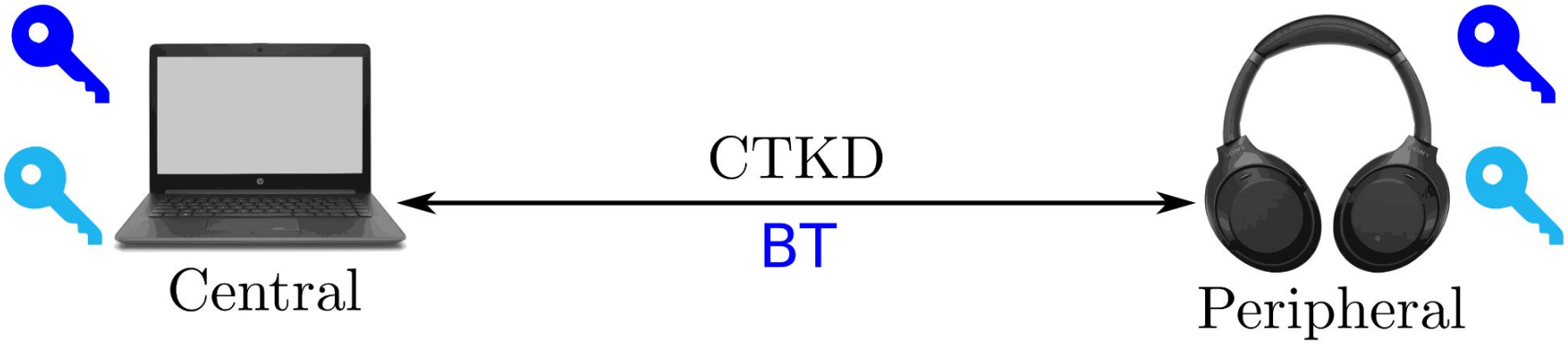
Strongest security mode. E.g. Secure Connections, MitM, CTKD, and Input-Output support.

BT Pairing Key Derivation and Authentication



BT pairing key derivation via ECDH. Strongest authentication available (Numeric Comparison)

BLE Pairing Key Cross-Transport Key Derivation (CTKD)

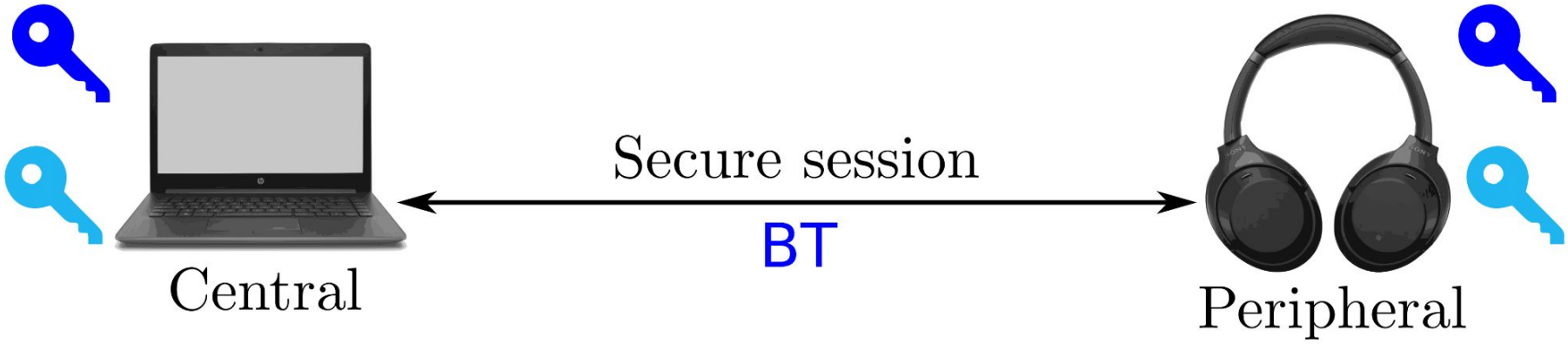


Derives K_{BLE} from K_{BT}, no BLE packets

Crosses the BT/BLE security boundary CTKD

Introduced for **usability,** no security evaluation (2014, v4.2)

Pairing Completed and Secure Sessions Establishment



Devices can start a BLE secure session **without** having to pair over BLE

Attacker Model

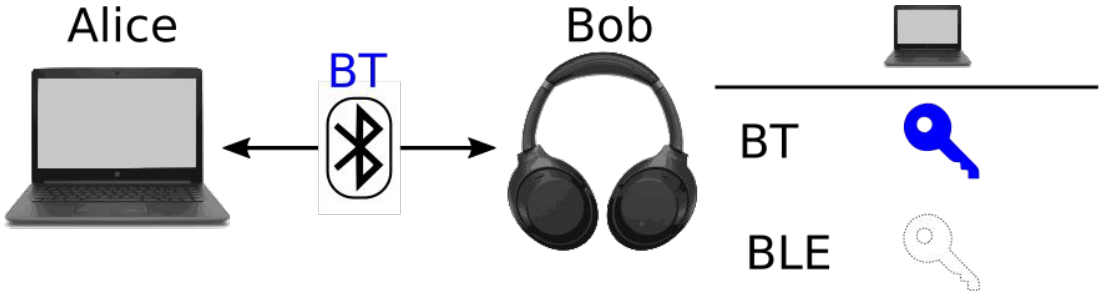
Charlie, attacker in Bluetooth range
Goals: **Cross-transport Impersonation, MitM, unintended sessions**

BLUR Attacks: Summary

1. **Cross-transport central impersonation**
2. **Cross-transport peripheral impersonation**
3. **Cross-transport MitM**
4. **Cross-transport unintended session**

NOTE: attacks as standard-compliant as they exploit CTKD’s specification

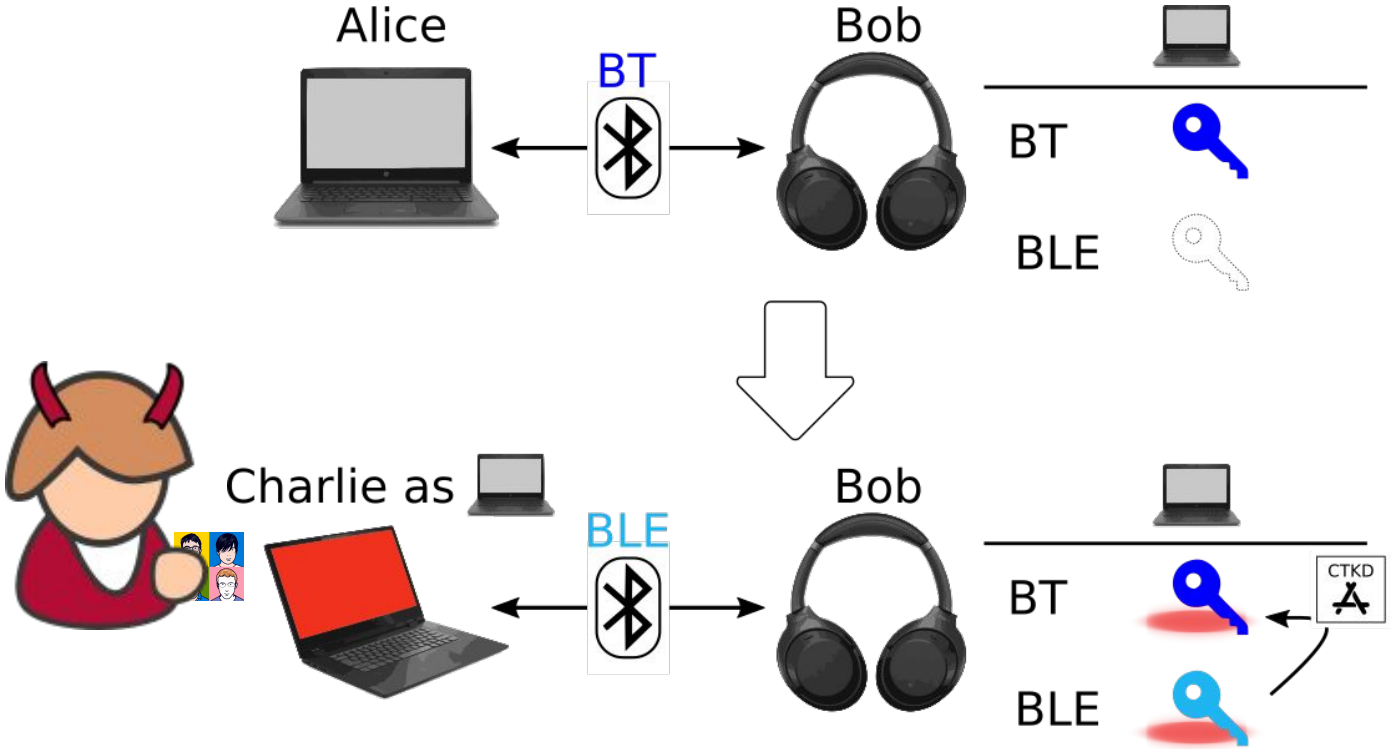
BLUR Attacks: Cross-Transport Central Impersonation



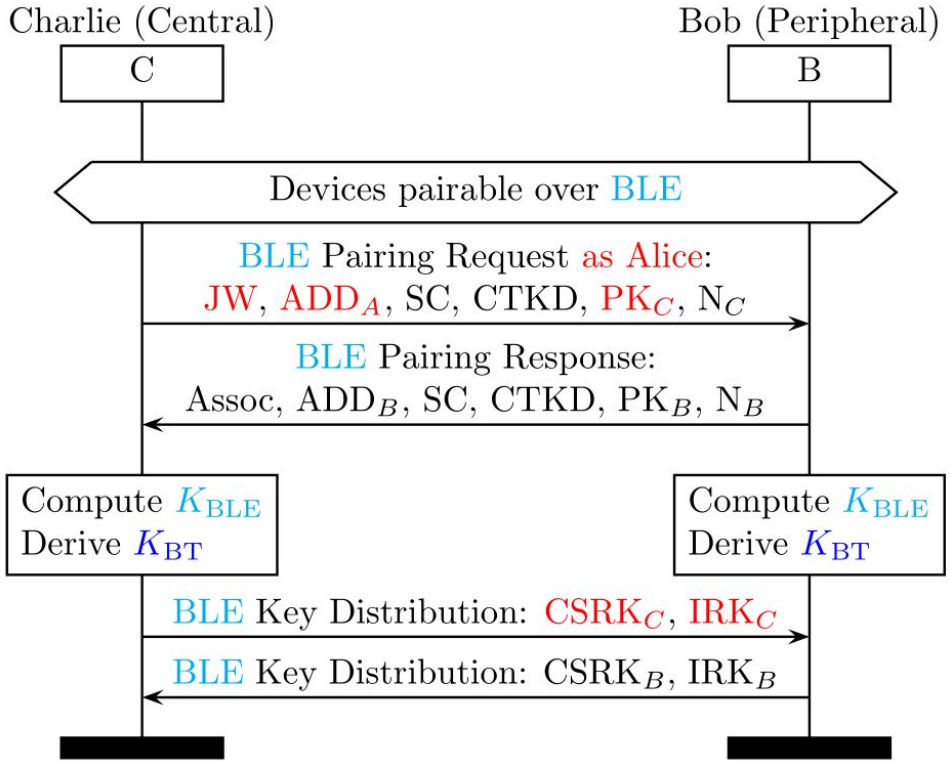
What happens if Charlie tries to pair over BLE with Bob while impersonating Alice?

NEW: Cross-transport Central Impersonation

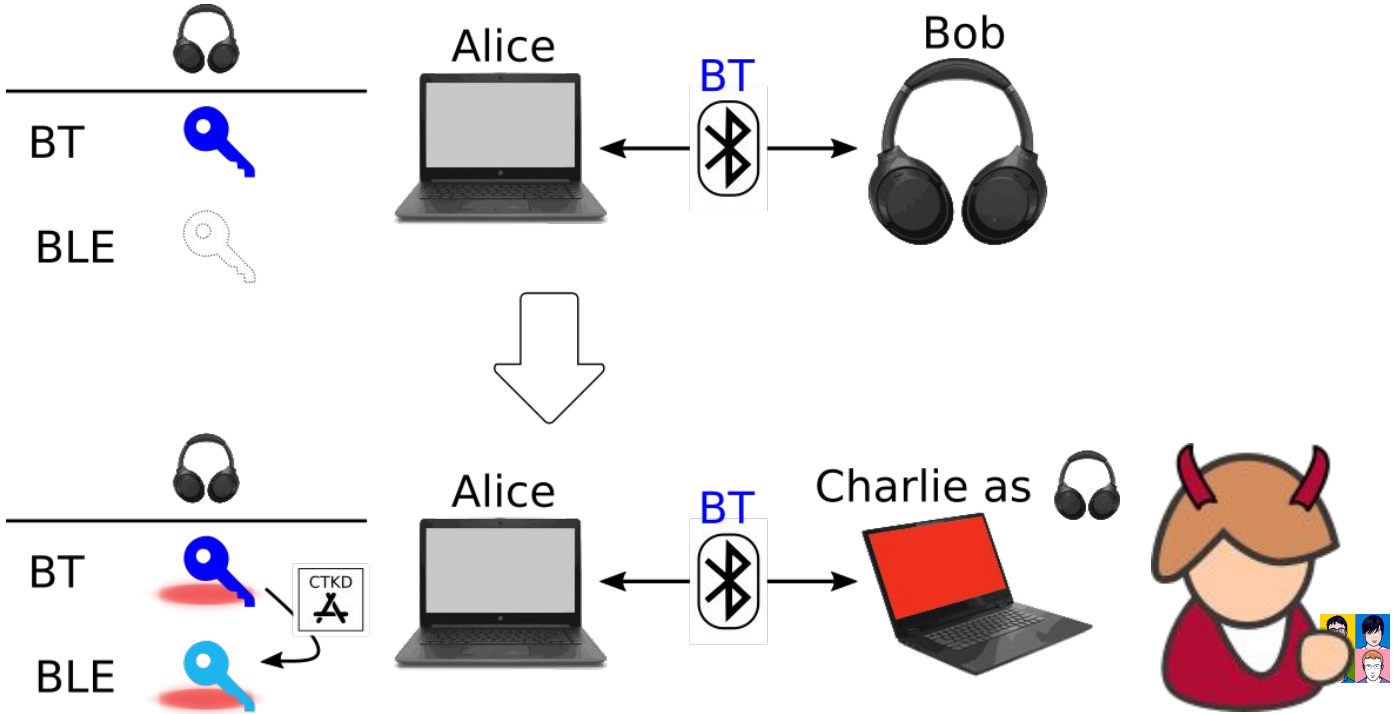
BLUR Attacks: Cross-Transport Central Impersonation



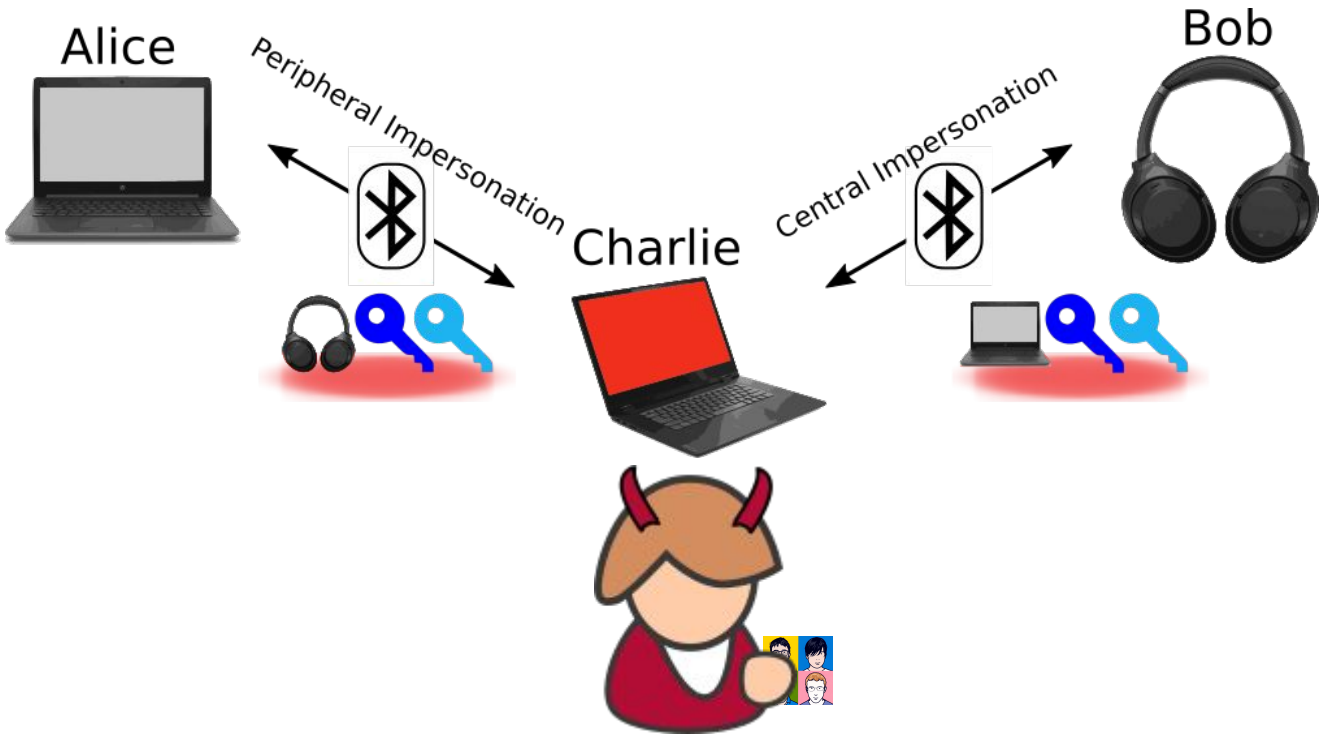
BLUR Attacks: Cross-Transport Central Impersonation (2)



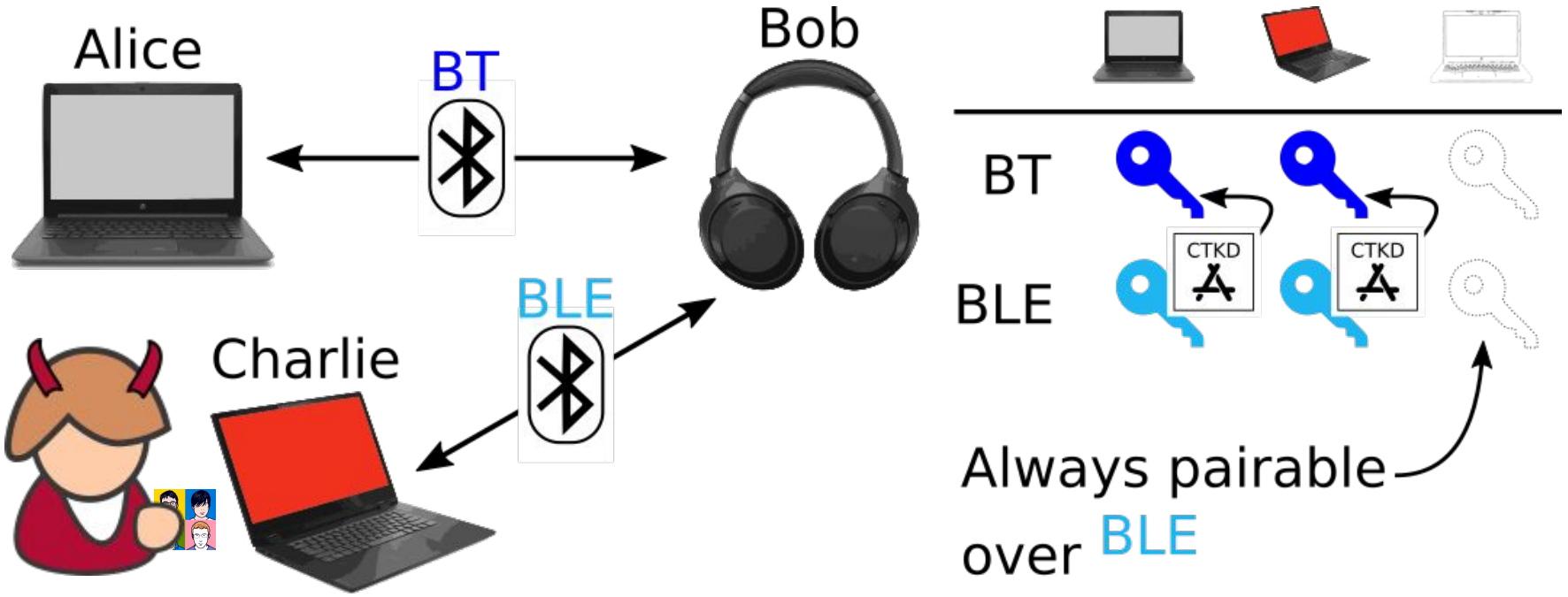
BLUR Attacks: Cross-Transport Peripheral Impersonation



BLUR Attacks: Cross-Transport MitM



BLUR Attacks: Cross-Transport Unintended Session



Evaluation: Exploiting 16 devices (14 unique chips)

Device			Chip		Bluetooth	BLUR Attack			
Producer	Model	OS	Producer	Model	Version	Role	MI/SI	MitM	US
Cypress	CYW920819EVB-02	Proprietary	Cypress	CYW20819	5.0	Peripheral	✓	✓	✓
Dell	Latitude 7390	Win 10 PRO	Intel	8265	4.2	Peripheral	✓	✓	✓
Google	Pixel 2	Android	Qualcomm	SDM835	5.0	Peripheral	✓	✓	✓
Google	Pixel 4	Android	Qualcomm	702	5.0	Peripheral	✓	✓	✓
Lenovo	X1 (3rd gen)	Linux	Intel	7265	4.2	Peripheral	✓	✓	✓
Lenovo	X1 (7th gen)	Linux	Intel	9560	5.1	Peripheral	✓	✓	✓
Samsung	Galaxy A40	Android	Samsung	Exynos 7904	5.0	Peripheral	✓	✓	✓
Samsung	Galaxy A51	Android	Samsung	Exynos 9611	5.0	Peripheral	✓	✓	✓
Samsung	Galaxy A90	Android	Qualcomm	SDM855	5.0	Peripheral	✓	✓	✓
Samsung	Galaxy S10	Android	Broadcom	BCM4375	5.0	Peripheral	✓	✓	✓
Samsung	Galaxy S10e	Android	Broadcom	BCM4375	5.0	Peripheral	✓	✓	✓
Samsung	Galaxy S20	Android	Broadcom	BCM4375	5.0	Peripheral	✓	✓	✓
Xiaomi	Mi 10T Lite	Android	Qualcomm	9312	5.1	Peripheral	✓	✓	✓
Xiaomi	Mi 11	Android	Qualcomm	10765	5.2	Peripheral	✓	✓	✓
Sony	WH-1000XM3	Proprietary	CSR	12414	4.2	Central	✓	✓	✓
Sony	WH-CH700N	Proprietary	CSR	12942	4.1†	Central	✓	✓	✓

BLUR Attacks Root Causes: Issues with CTKD

- Device always pairable over BT and BLE
  - Attacker pairs on unused transports (impersonating someone)
- Cross-transport key tampering
  - Attacker writes, overwrites, and steals BT/BLE keys
- Cross-transport association mismatch
  - Attacker downgrades association (when necessary)
- Cross-transport roles mismatch
  - Attacker pairs mixing roles (e.g., BLE Central, BT Peripheral)

Our Countermeasures

- Disable key overwriting via CTKD, unless user consent
  - Prevent key overwriting via CTKD
  - We implemented and tested it on Linux
- Disable BT/BLE pairability if not needed, provide a pairing UI
  - Prevent an attacker from pairing on unused transports

Fix in the Bluetooth standard 5.1+ is not effective

From the standard: *“While performing CTKD derivation, if the key for the other transport already exists, then the devices shall not overwrite that existing key with a key that is weaker in either* ***strength*** *or* ***MITM protection****”*

- Bluetooth 4.2 and 5.0 are not covered despite being popular versions
- BLUR key write and unintended session attacks not covered
- BLUR key overwrite attacks do not require to downgrade key’s strength and MitM protection

Conclusion and Q&A

- CTKD is a **novel** and **cross-transport** attack surface
- Uncover **four vulnerabilities** in the CTKD **specification**
- Develop **four cross-transport (BLUR) attacks**
  - Cross-transport Impersonation, MitM, unintended sessions
- **Conduct** the BLUR attacks on actual devices
  - Exploit 16 devices (14 chips, Bluetooth 4.1, 4.2, 5.0, 5.1, 5.2)
- **Fix** the BLUR attacks
  - Unlike the mitigation in the Bluetooth standard
- **Links**: paper, slides, video, code, website