BlueMirror: Reflections on Bluetooth Pairing and Provisioning Protocols

Tristan Claverie, José Lopes Esteves

Agence nationale de la sécurité des systèmes d'information

May 27, 2021

Presentation

ANSSI, Wireless Security Laboratory

- Electromagnetic Security (TEMPEST, IEMI)
- Wireless protocols
- Signal processing
- Simulations, measures, electromagnetism
- Embedded systems

Presentation

ANSSI, Wireless Security Laboratory

- Electromagnetic Security (TEMPEST, IEMI)
- Wireless protocols
- Signal processing
- Simulations, measures, electromagnetism
- Embedded systems

Tristan Claverie

- Wireless protocol security
- Internet of Things
- DVB, Bluetooth LE, Classic, Mesh, LoRaWAN
- Software-Defined Radio

Outline of the presentation

1. Introduction to Bluetooth Classic, LE, Mesh

Presentation of Bluetooth technologies

Bluetooth Classic (BT)

- Standardised in 1999
- Communication protocol
- 2+ devices communicate together
- Spec : *Bluetooth Core Specification*

Use cases :

- Cars, Smartphones
- Audio devices

Bluetooth Low Energy (BLE)

- Standardised in 2010
- Communication protocol
- 2 devices communicate together
- Spec : *Bluetooth Core Specification*

Use cases :

- Smartphones
- Smart* (watches, bands. . .)
- Medical devices

Bluetooth Mesh (BM)

- Standardised in 2017
- Uses BLE PHY/LNK layers
- Network of devices communicate together
- Several applications (light, sensors. . .) in a Network.
- Spec : *Bluetooth Mesh {Model, Profile} Specification*

Use cases :

- Connected homes

Security in Bluetooth technologies

BT / BLE security goals

- Confidentiality
- Integrity
- Authenticity (opt.)

Security in Bluetooth technologies

BT / BLE security goals

- Confidentiality
- Integrity
- Authenticity (opt.)

BM security goals

- Confidentiality
- Integrity
- Authenticity (opt.)
- Segregation of applications inside a network

Security in Bluetooth technologies

BT / BLE security goals

- Confidentiality
- Integrity
- Authenticity (opt.)

BM security goals

- Confidentiality
- Integrity
- Authenticity (opt.)
- Segregation of applications inside a network

Symmetric secrets :

- EncKey - protect communication between two devices (LK, LTK, . . .)

Security in Bluetooth technologies

BT / BLE security goals

- Confidentiality
- Integrity
- Authenticity (opt.)

Symmetric secrets :

- EncKey - protect communication between two devices (LK, LTK, . . .)

BM security goals

- Confidentiality
- Integrity
- Authenticity (opt.)
- Segregation of applications inside a network

Symmetric secrets :

- NetKey - communicate on the network
- AppKey - send/receive applicative data
- DevKey - device configuration

Security in Bluetooth technologies

BT / BLE security goals

- Confidentiality
- Integrity
- Authenticity (opt.)

Symmetric secrets :

- EncKey - protect communication between two devices (LK, LTK, . . .)

BM security goals

- Confidentiality
- Integrity
- Authenticity (opt.)
- Segregation of applications inside a network

Symmetric secrets :

- NetKey - communicate on the network
- AppKey - send/receive applicative data
- DevKey - device configuration

=> A Key agreement protocol is used to exchange those symmetric secrets

Key agreement in Bluetooth

BT / BLE : **Pairing**

- Happens between an Initiator and a Responder
- Used when two devices have no previously shared secret
- At the end of the procedure, both devices share EncKey
- May be authenticated
- Several Pairing protocols exist, not the same between BT/BLE

Key agreement in Bluetooth

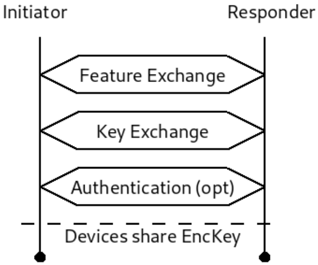
BT / BLE : **Pairing**

- Happens between an Initiator and a Responder
- Used when two devices have no previously shared secret
- At the end of the procedure, both devices share EncKey
- May be authenticated
- Several Pairing protocols exist, not the same between BT/BLE

BM : **Provisioning**

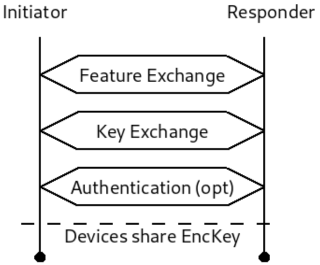
- Happens between a Provisioner and a Device
- Used when a device wants to join a Network
- At the end of the procedure, the Device receives NetKey and derives DevKey.
- May be authenticated
- Several Provisioning protocols exist

Pairing :

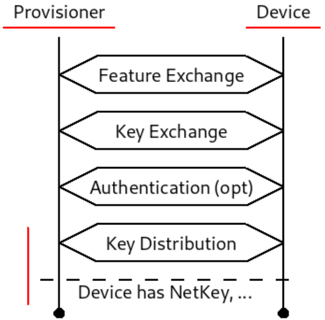

Initiator
Responder
Feature Exchange
Key Exchange
Authentication (opt)
Devices share EncKey

Pairing/Provisioning protocol : high-level view

Pairing :



Provisioning :



12 shades of Pairing

Pairing method depends on : supported version, user interaction.

Technology	BT		BLE	
Pairing Mode	**Legacy**	**Secure Simple Pairing**	**Legacy Pairing**	**LE Secure Pairing**
Pairing Method	PIN Pairing	JustWorks	JustWorks	JustWorks
		Passkey Entry	Passkey Entry	Passkey Entry
		Numeric Comparison	Out of Band	Numeric Comparison
		Out of Band		Out of Band

- BLE : Legacy/Secure are **different** protocols => Legacy JW $\neq$ Secure JW
- BLE/BT : SSP and LESP are the **same** protocols => SSP JW $\approx$ LESP JW

8 kinds of Provisioning

Provisioning depends on :

- How the key exchange is performed (in-band, out of band)
- How authentication data is exchanged (no authentication, input data, output data, static data)
- No specific names for the 8 variants of the Provisioning protocol.

In-band ; No auth	Out of Band ; No auth
In-band ; Input	Out of Band ; Input
In-band ; Output	Out of Band ; Output
In-band ; Static	Out of Band ; Static

Classifying Bluetooth key agreement protocols

At a high-level, all Bluetooth key agreement fall into one of three categories :

- Unauthenticated : key agreement is not authenticated
- Authenticated : key agreement is authenticated
- Out of Band : security properties come from an unspecified communication channel

Technology	BT		BLE		BM	
Pairing Mode	Legacy	Secure Simple Pairing	Legacy Pairing	LE Secure Pairing	N/A	N/A
Pairing/ Provisioning Method	PIN Pairing	JustWorks	JustWorks	JustWorks	In-band ; No auth.	Out of Band ; No auth.
		Passkey Entry	Passkey Entry	Passkey Entry	In-band ; Input	Out of Band ; Input
		Numeric Comparison	Out of Band	Numeric Comparison	In-band ; Output	Out of Band ; Output
		Out of Band		Out of Band	In-band ; Static	Out of Band ; Static

2. Scope of the study

State of the Art

Passive attacks: [SW05], [Rya13]

Active attacks: [Lin08], [Ros13], [BN19], [vTPFG21], [ATR20b]

Technology	BT		BLE		BM	
Pairing Mode	Legacy	Secure Simple Pairing	Legacy Pairing	LE Secure Pairing	N/A	N/A
Pairing/ Provisioning Method	PIN Pairing	JustWorks	JustWorks	JustWorks	In-band ; No auth.	Out of Band ; No auth.
		Passkey Entry	Passkey Entry	Passkey Entry	In-band ; Input	Out of Band ; Input
		Numeric Comparison	Out of Band	Numeric Comparison	In-band ; Output	Out of Band ; Output
		Out of Band		Out of Band	In-band ; Static	Out of Band ; Static

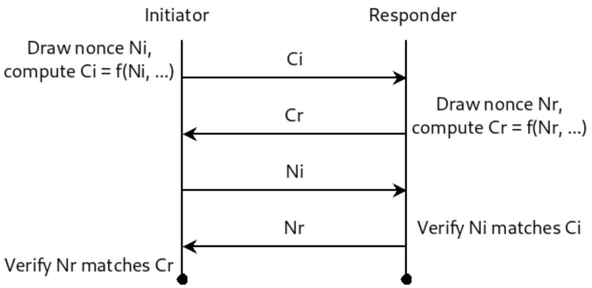
Study

Goal : Study authenticated Bluetooth protocols

Means : Reflection attacks

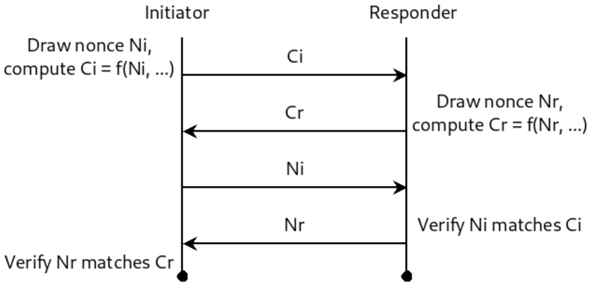
Technology	BT		BLE		BM	
Pairing Mode	Legacy	Secure Simple Pairing	Legacy Pairing	LE Secure Pairing	N/A	N/A
Pairing/ Provisioning Method	PIN Pairing					
		Passkey Entry	Passkey Entry	Passkey Entry	In-band ; Input	
		Numeric Comparison		Numeric Comparison	In-band ; Output	

Building block in Bluetooth authentication protocols : commitment protocol

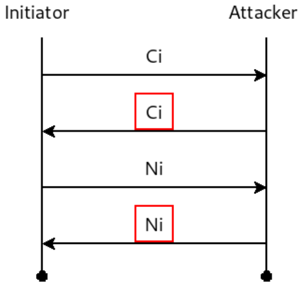


Reflection attacks : concept

Building block in Bluetooth authentication protocols : commitment protocol



Example of a reflection attack



Reflection attacks : impact

Goals :

- Complete authentication protocol, do not retrieve encryption key
- Complete authentication protocol, retrieve encryption key

In the literature :

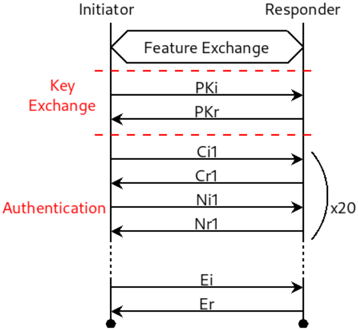
- Reflection in TLS 1.3 PSK mode, no encryption key at the end [DG19]
- Theoretical reflection in a BT security protocol, no encryption key at the end [ATR20a]

=> Easy to patch in implementations, but should be made impossible by good protocols.

3. Results

Secure Passkey Entry

Used for BT SSP, BLE SP
One device displays a passkey, user inputs in on the other.

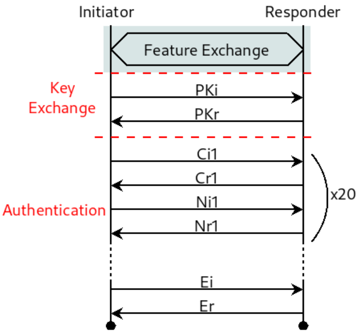


Passkey is 20 bits long

Secure Passkey Entry

Used for BT SSP, BLE SP

One device displays a passkey, user inputs in on the other.

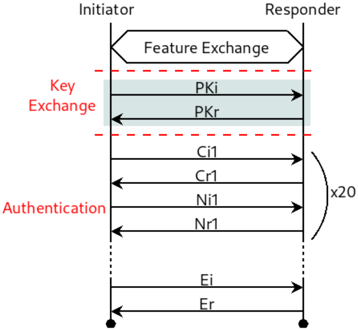


Passkey is 20 bits long

1. Feature Exchange

Secure Passkey Entry

Used for BT SSP, BLE SP
One device displays a passkey, user inputs in on the other.

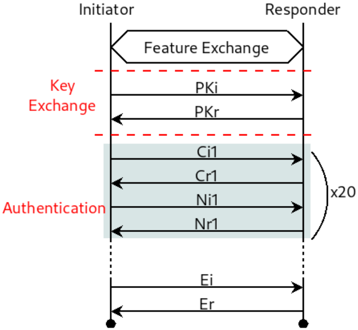


Passkey is 20 bits long

1. Feature Exchange
2. Diffie-Hellman key exchange

Secure Passkey Entry

Used for BT SSP, BLE SP
One device displays a passkey, user inputs in on the other.

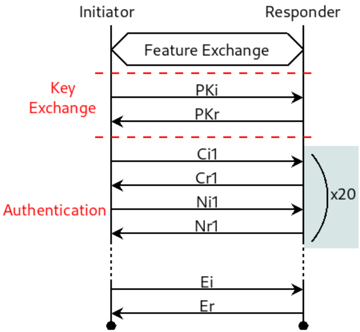


Passkey is 20 bits long

1. Feature Exchange
2. Diffie-Hellman key exchange
3. Commitment protocol uses 1 bit of the passkey

Secure Passkey Entry

Used for BT SSP, BLE SP
One device displays a passkey, user inputs in on the other.

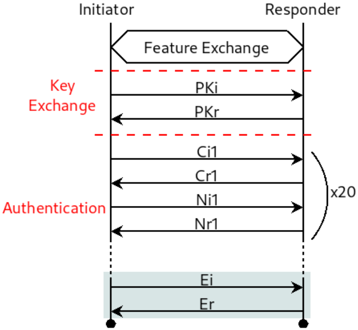


Passkey is 20 bits long

1. Feature Exchange
2. Diffie-Hellman key exchange
3. Commitment protocol uses 1 bit of the passkey
4. 20 rounds of commitments

Secure Passkey Entry

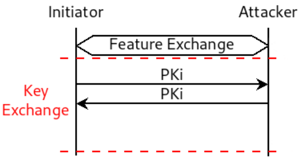
Used for BT SSP, BLE SP
One device displays a passkey, user inputs in on the other.



Passkey is 20 bits long

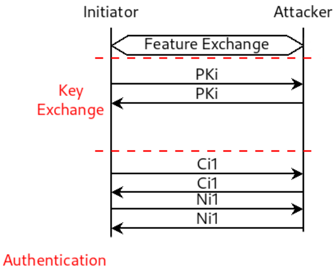
1. Feature Exchange
2. Diffie-Hellman key exchange
3. Commitment protocol uses 1 bit of the passkey
4. 20 rounds of commitments
5. Final exchange of messages

Secure Passkey Entry : Impersonation



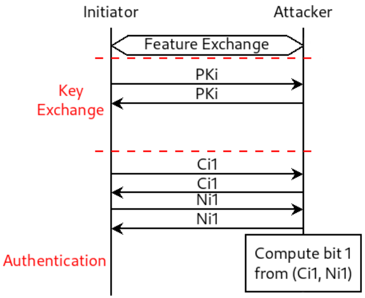
1. Reflect Initiator's public key, then all rounds

Secure Passkey Entry : Impersonation



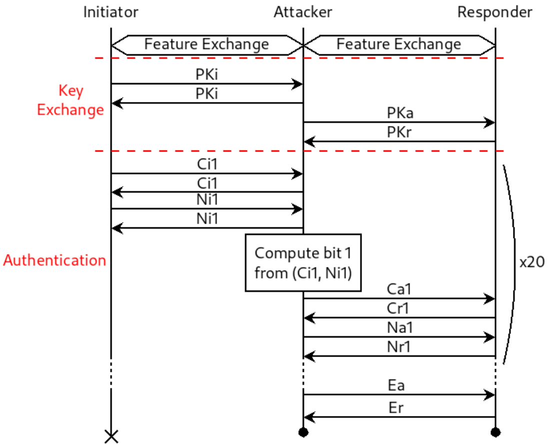
1. Reflect Initiator's public key, then all rounds

Secure Passkey Entry : Impersonation



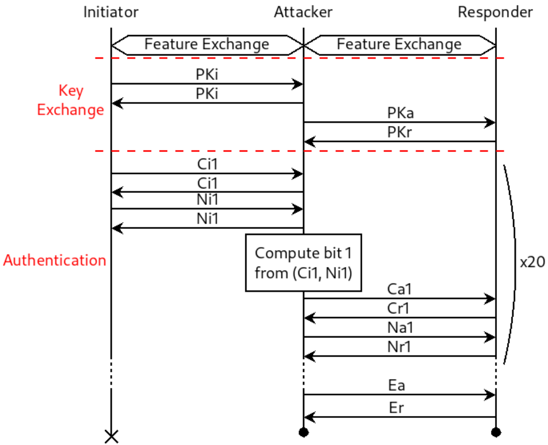
1. Reflect Initiator's public key, then all rounds
2. => Attacker can learn the passkey : retrieve p_k from (Cx_k, Nx_k)(Lindell, 2008 [Lin08])

Secure Passkey Entry : Impersonation



1. Reflect Initiator's public key, then all rounds
2. => Attacker can learn the passkey : retrieve p_k from (Cx_k, Nx_k)(Lindell, 2008 [Lin08])
3. => Use the passkey to authenticate to the legitimate responder

Secure Passkey Entry : Impersonation



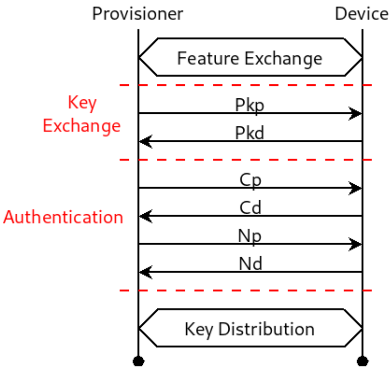
1. Reflect Initiator's public key, then all rounds
2. => Attacker can learn the passkey : retrieve p_k from (Cx_k, Nx_k)(Lindell, 2008 [Lin08])
3. => Use the passkey to authenticate to the legitimate responder

- Attacker ends impersonating Initiator, with EncKey
- Works in BT SSP, BLE SP
- Initiator has failed Pairing

Details and variants in the proceedings

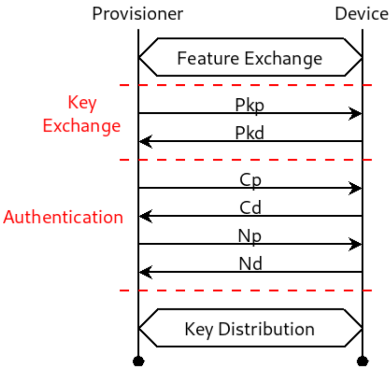
Provisioning protocol

Authenticated Provisioning : Key exchange is performed in-band ; one device outputs AuthData and the user inputs it on the other end.



Provisioning protocol

Authenticated Provisioning : Key exchange is performed in-band ; one device outputs AuthData and the user inputs it on the other end.



- AuthData is padded into *AuthValue*.
- *AuthValue*, nonces and confirmations are **16 bytes** long.

Commitment protocol :

$$CK = f(DHKey, FeatureExchange)$$

$$Cp = \text{AES-CMAC}_{CK}(Np || AuthValue)$$

$$Cd = \text{AES-CMAC}_{CK}(Nd || AuthValue)$$

- Trivial reflection attack (cf. proceedings)
- Cryptographic misuse !

Provisioning : Cryptographic misuse

Problem : CMAC mode is **not pre-image resistant** => with known key, one block of plaintext leaks.
AES-CMAC : RFC4493

$$CK_1 = f(CK)$$
$$C = \text{AES-CMAC}_{CK}(N || AuthValue)$$
$$C = \text{AES}_{CK}(\text{AES}_{CK}(N) \oplus CK_1 \oplus AuthValue)$$

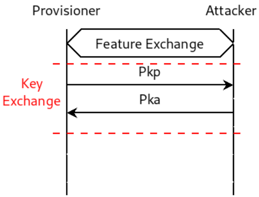
Retrieve $AuthValue$ with (CK, N, C) :

$$AuthValue = \text{AES}^{-1}_{CK}(C) \oplus \text{AES}_{CK}(N) \oplus CK_1$$

Retrieve N with $(CK, AuthValue, C)$:

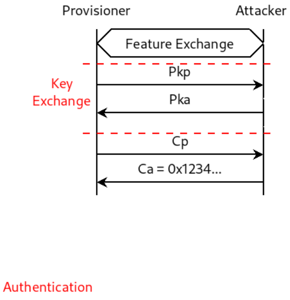
$$N = \text{AES}^{-1}_{CK}(\text{AES}^{-1}_{CK}(C) \oplus CK_1 \oplus AuthValue)$$

Provisioning : Attack



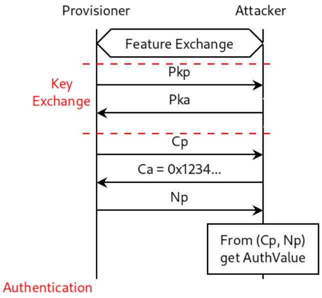
1. Send public key

Provisioning : Attack



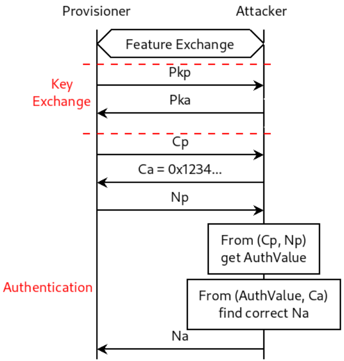
1. Send public key
2. Send random confirmation

Provisioning : Attack



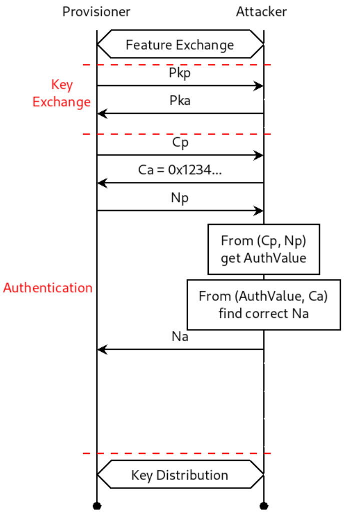
1. Send public key
2. Send random confirmation
3. Retrieve AuthValue

Provisioning : Attack



1. Send public key
2. Send random confirmation
3. Retrieve AuthValue
4. Craft nonce

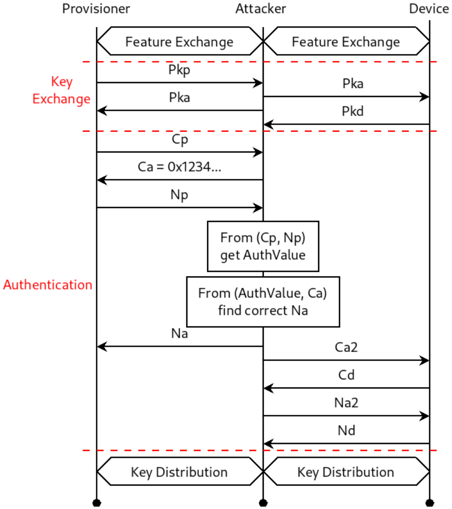
Provisioning : Attack



Impersonation :

- Gains NetKey, may get AppKey(s)
- Legitimate Device couldn't join the Network

Provisioning : Attack



Impersonation :

- Gains NetKey, may get AppKey(s)
- Legitimate Device couldn't join the Network

MitM :

- Gain DevKey of the legitimate device
- Legitimate device appears to have joined the network
- Not patchable at the implementation level => specification update

Context of presented results

Secure Passkey Entry

Before :

- If passkey is perfectly random, no problem [Lin08]

This work :

- If passkey is perfectly random, problems remain

Context of presented results

Secure Passkey Entry

Before :

- If passkey is perfectly random, no problem [Lin08]

This work :

- If passkey is perfectly random, problems remain

Mesh

Before :

- No analysis of Provisioning protocol

Related :

- Malleable commitment in BLE Legacy Passkey Entry => Authentication is broken [Ros13]

This work :

- Malleable commitment in BM Provisioning => Authentication is broken

Synthesis

- In total, 7 attacks discovered
- Results were validated experimentally on real-world implementations
- Responsible disclosure to Bluetooth SIG in September, 2020 => 6 CVEs allocated

Attack	Technology			Security	Attacker position	Key recovered	Impact	Target	Test	Weakness	CVE
	BT	BLE	BM								
BLE-A		X		Legacy	Spoofer	No	Impersonation	Initiator	Complete	Reflection	No
BT-A	X			Legacy	Spoofer	Yes	Impersonation	Initiator	Partial	Reflection	2020-26555
PE-A2	X	X		Secure	MitM	Yes	Impersonation	Responder	Complete	Reflection	2020-26558
PE-A1	X	X		Secure	Spoofer	No	Impersonation	Initiator	Partial	Reflection	No
M-A1			X	Secure	Spoofer	Yes	Impersonation	Provisioner	Complete	Reflection	2020-26560
M-A2			X	Secure	Spoofer	Yes	Impersonation	Provisioner	Complete	Crypto	2020-26557
					MitM		MitM	Both			
M-A3			X	Secure	Spoofer	Yes	Impersonation	Provisioner	Complete	Crypto	2020-26556 2020-26559
					MitM		MitM	Both			

Synthesis

- Authenticated key agreements

Technology	BT		BLE		BM	
Pairing Mode	**Legacy**	**Secure Simple Pairing**	**Legacy Pairing**	**LE Secure Pairing**	**N/A**	**N/A**
Pairing/ Provisioning Method	PIN Pairing	JustWorks	JustWorks	JustWorks	In-band ; No auth.	Out of Band ; No auth.
		Passkey Entry	Passkey Entry	Passkey Entry	In-band ; Input	Out of Band ; Input
		Numeric Comparison	Out of Band	Numeric Comparison	In-band ; Output	Out of Band ; Output
		Out of Band		Out of Band	In-band ; Static	Out of Band ; Static

Synthesis

- Authenticated key agreements
- Secure key agreements according to the specification [Blu19a, Blu19b]

Technology	BT		BLE		BM	
Pairing Mode	Legacy	Secure Simple Pairing	Legacy Pairing	LE Secure Pairing	N/A	N/A
Pairing/ Provisioning Method	PIN Pairing	JustWorks	JustWorks	JustWorks	In-band ; No auth.	Out of Band ; No auth.
		Passkey Entry	Passkey Entry	Passkey Entry	In-band ; Input	Out of Band ; Input
		Numeric Comparison	Out of Band	Numeric Comparison	In-band ; Output	Out of Band ; Output
		Out of Band		Out of Band	In-band ; Static	Out of Band ; Static

Synthesis

- Authenticated key agreements (blue square)
- Secure key agreements according to the specification [Blu19a, Blu19b] (black circle)
- Successfully attacked key agreements in this study (red circle)

Technology	BT		BLE		BM	
Pairing Mode	Legacy	Secure Simple Pairing	Legacy Pairing	LE Secure Pairing	N/A	N/A
Pairing/ Provisioning Method	PIN Pairing	JustWorks	JustWorks	JustWorks	In-band ; No auth.	Out of Band ; No auth.
		Passkey Entry	Passkey Entry	Passkey Entry	In-band ; Input	Out of Band ; Input
		Numeric Comparison	Out of Band	Numeric Comparison	In-band ; Output	Out of Band ; Output
		Out of Band		Out of Band	In-band ; Static	Out of Band ; Static

4. Conclusion

Conclusion

- Very informative cases of real-world reflection attacks, with key retrieval
- Numeric Comparison appears (again) to be the most resistant Pairing method
- Most of the problems we found (reflection attacks) can be patched in implementations ; some will require a redesign
- Three out of three Bluetooth technologies required complete redesign of initial key agreements protocols

- Bluetooth retrocompatibility may pose new problems in BM
- Don't rely on Bluetooth built-in security
- If you have to, pair/provision devices in controlled environments (e.g. Faraday cage)

Questions ?

Contact

- tristan.claverie@ssi.gouv.fr

References

[ATR20a] Daniele Antonioli, Nils Ole Tippenhauer, and Kasper Rasmussen, *BIAS : Bluetooth Impersonation AttackS*, Proceedings of the IEEE Symposium on Security and Privacy (S&P), May 2020.

[ATR20b] ______, *Key Negotiation Downgrade Attacks on Bluetooth and Bluetooth Low Energy*, ACM Trans. Priv. Secur. **23** (2020), no. 3.

[Blu19a] Bluetooth SIG, *Bluetooth core specification*, 12 2019, v5.2.

[Blu19b] Bluetooth SIG, *Mesh profile bluetooth specification*, 01 2019, v1.0.1.

[BN19] Eli Biham and Lior Neumann, *Breaking the Bluetooth Pairing – The Fixed Coordinate Invalid Curve Attack*, Cryptology ePrint Archive, Report 2019/1043, 2019, `https://eprint.iacr.org/2019/1043`.

[DG19] Nir Drucker and Shay Gueron, *Selfie : Reflections on TLS 1.3 with PSK*, Cryptology ePrint Archive, Report 2019/347, 2019, `https://eprint.iacr.org/2019/347`.

References (cont.)

[Lin08] Andrew Y Lindell, *Attacks on the Pairing Protocol of Bluetooth v2.1*, https://www.blackhat.com/presentations/bh-usa-08/Lindell/BH_US_08_Lindell_Bluetooth_2.1_New_Vulnerabilities.pdf, June 2008, BlackHat USA, p. 10.

[Ros13] Tomas Rosa, *Bypassing Passkey Authentication in Bluetooth Low Energy*, Cryptology ePrint Archive, Report 2013/309, 2013, https://eprint.iacr.org/2013/309.

[vTPFG21] M. von Tschirschnitz, L. Peuckert, F. Franzen, and J. Grossklags, *Method Confusion Attack on Bluetooth Pairing*, 2021 2021 IEEE Symposium on Security and Privacy (SP) (Los Alamitos, CA, USA), IEEE Computer Society, may 2021, pp. 213–228.