

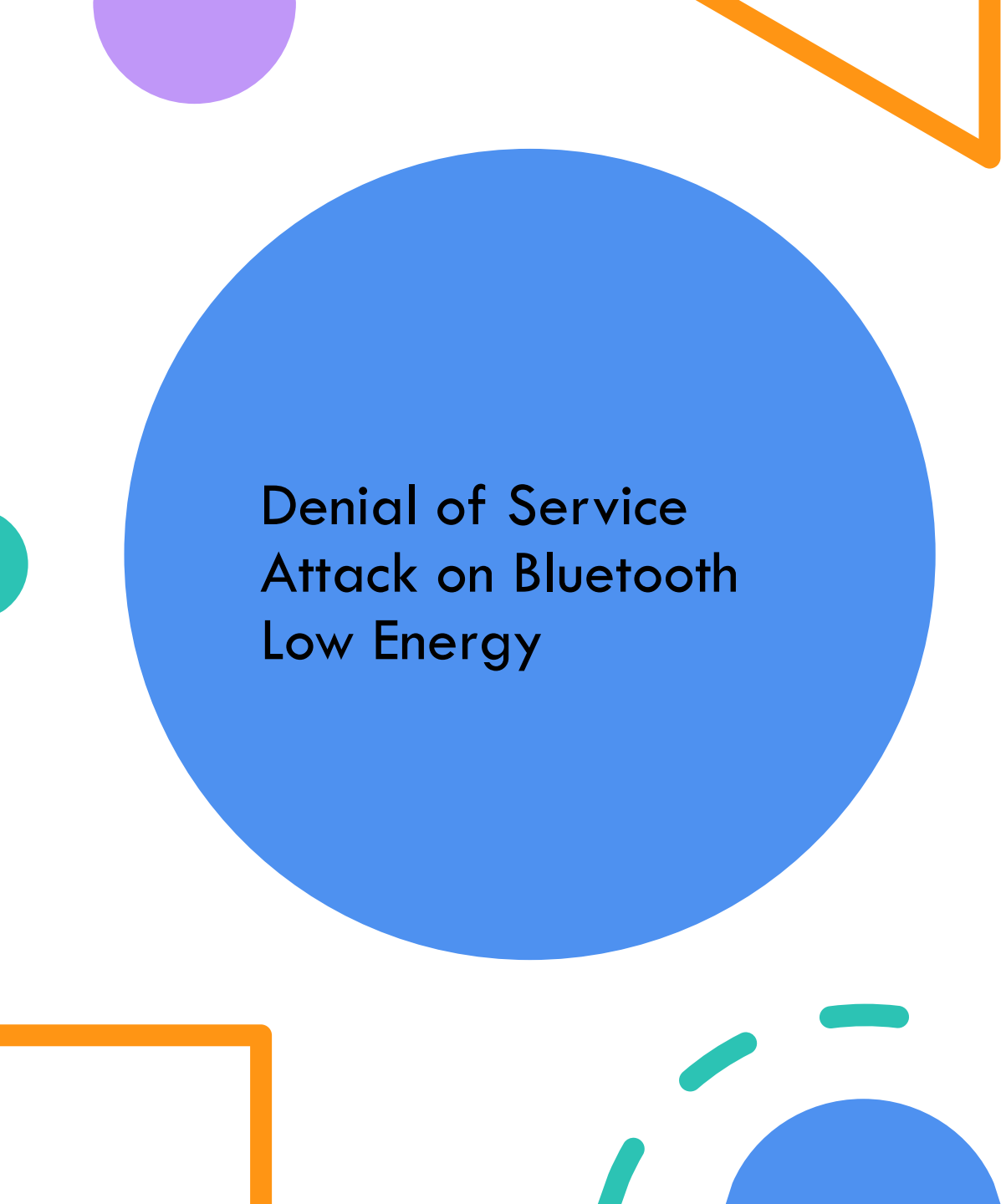


DSR

Paper Presentation

Joao Quinta





Denial of Service Attack on Bluetooth Low Energy

Auteur:

Peter Gullberg

Présentation de l'article

Introduction

Bluetooth Low Energy (BLE)

- Indépendant de Bluetooth classique
- BLE est une plus récente technologie

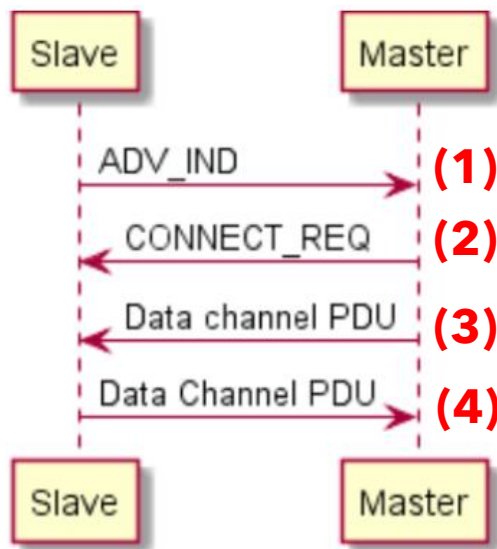
Avantages

- Consommation d'énergie très basse → On peut avoir un appareil qui utilise une petite batterie pendant des années
- Est présent sur quasiment tous les nouveaux appareils portables → Occupe une place physique de 2,5mm x 2,5mm

On peut facilement intégrer de la communication sans fils dans tout appareil

Analyse de l'étape de connexion

- Dans ce travail de recherche, le but est d'attaquer le processus de connexion entre deux appareils qui utilisent BLE
- Pour le faire nous devons d'abord introduire le processus de connexion



Il existe 40 chaînes disponibles, 37 pour transfert de données, 3 pour la connexion

(1) Le «slave» utilise la chaîne 37, 38 ou 39 pour envoyer «ADV_IND»

On y trouve des infos pour qu'on puisse le recontacter

- Adresse «slave»

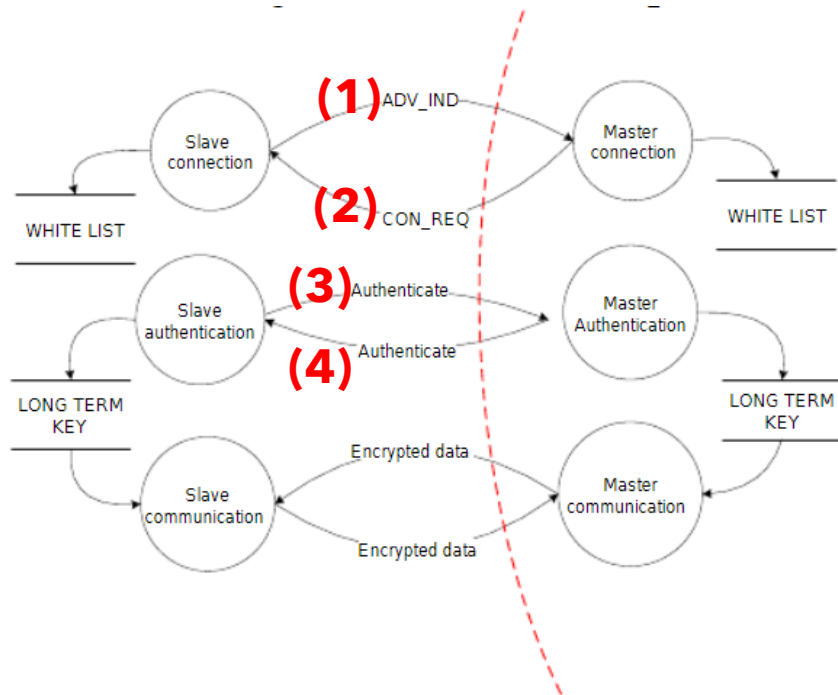
Le «master» qui écoute sur 1 des 3 chaînes reçoit le message de (1)

(2) Le «master» répond avec «CONNECT_REQ»

dans ce payload on trouve les informations qui vont définir la suite de la communication

- Chaîne à utiliser
- Période d'attente
- Adresse «master»

Analyse de l'étape de connexion



- Les étapes **(3)** et **(4)** sont les étapes d'authentification
- Dans l'étape **(2)** le «slave» peut vérifier la «white list» pour authentifier le «master», sauf si le master figure d'une adresse privé



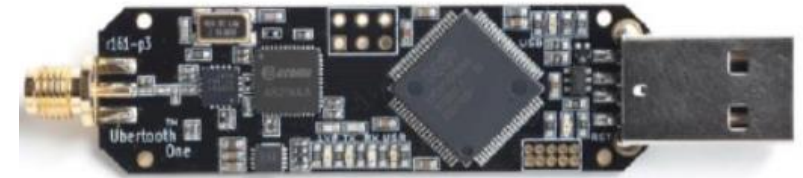
Les chips BLE n'ont pas la capacité de lire cette adresse privé en «real time» donc ils vont en étape **(3)** même sans connaître l'adresse du «master»

Idée d'attaque

- Créer un faux «master» qui est à l'écoute de «slave»
- Quand il en détecte grâce au message **(1)** «ADV_IND»
- On envoie un message **(2)** «CONNECT_REQ», avec une adresse privé
- Résultat attendu:
 - Le «slave» change de chaîne et attend la suite de la communication **(3)**, qui n'aura jamais lieu
- Quelques détails supplémentaires:
 - Le «slave» reste en étape **(3)** pendant un temps défini en étape **(2)** * 6
 - Le «slave» initialise la connexion **(1)** dans 1 des 3 chaînes disponibles à la fois, comme ça il est sûr d'éventuellement trouver un «master»

Création de l'attaque

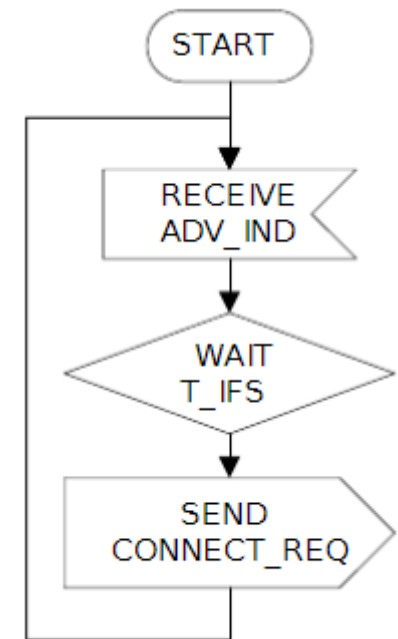
- D'abord il faut avoir un appareil physique qui sera notre «master» malicieux
 - On achète «The Ubertooth One Hardware»



- Ensuite il faut créer «l'exploit», ici se trouve l'étape plus technique, il faut modifier quelques fonctions du code qui est open source
- Le «slave» est un appareil tout à fait normal, capable de se connecter via BLE
- Pour surveiller l'attaque, on a un outils supplémentaire, un simple «sniffer» qui est tout simplement à l'écoute

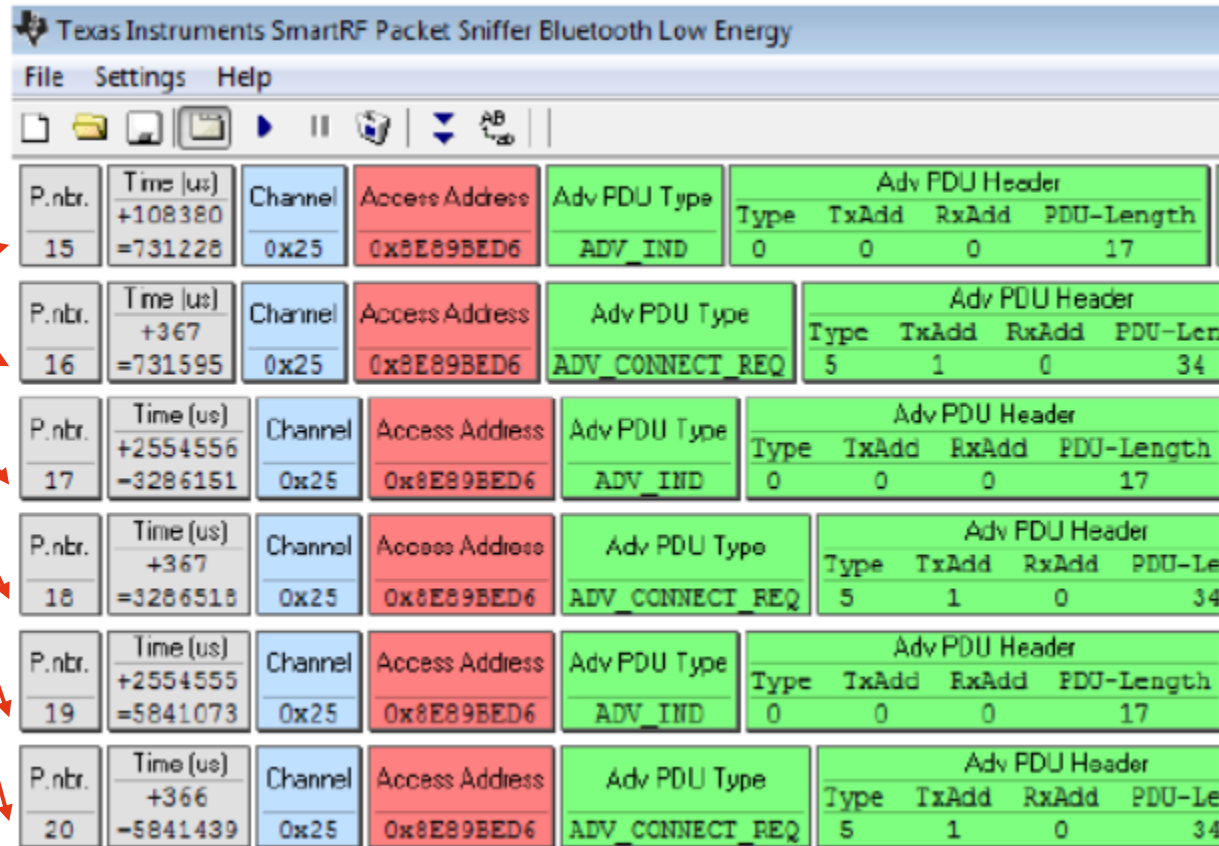
Mise en pratique

- L'attaque est lancée pendant 1 minute
- Le «slave» enverra le message de connexion toutes les 100 ms
- Le faux «master répond au «slave» **(2)** après 400ms, dans le but de le faire attendre le plus possible
- Le faux «master» indiquera au «slave» que le temps d'attente en étape **(3)** est de 400ms
- Le «slave» attend $6 \times 400\text{ms}$, pour l'étape **(3)**, pour un total de 2400ms ou 2,4s
- Notre sniffer qui est à l'écoute enregistre les données suivantes



Résultat

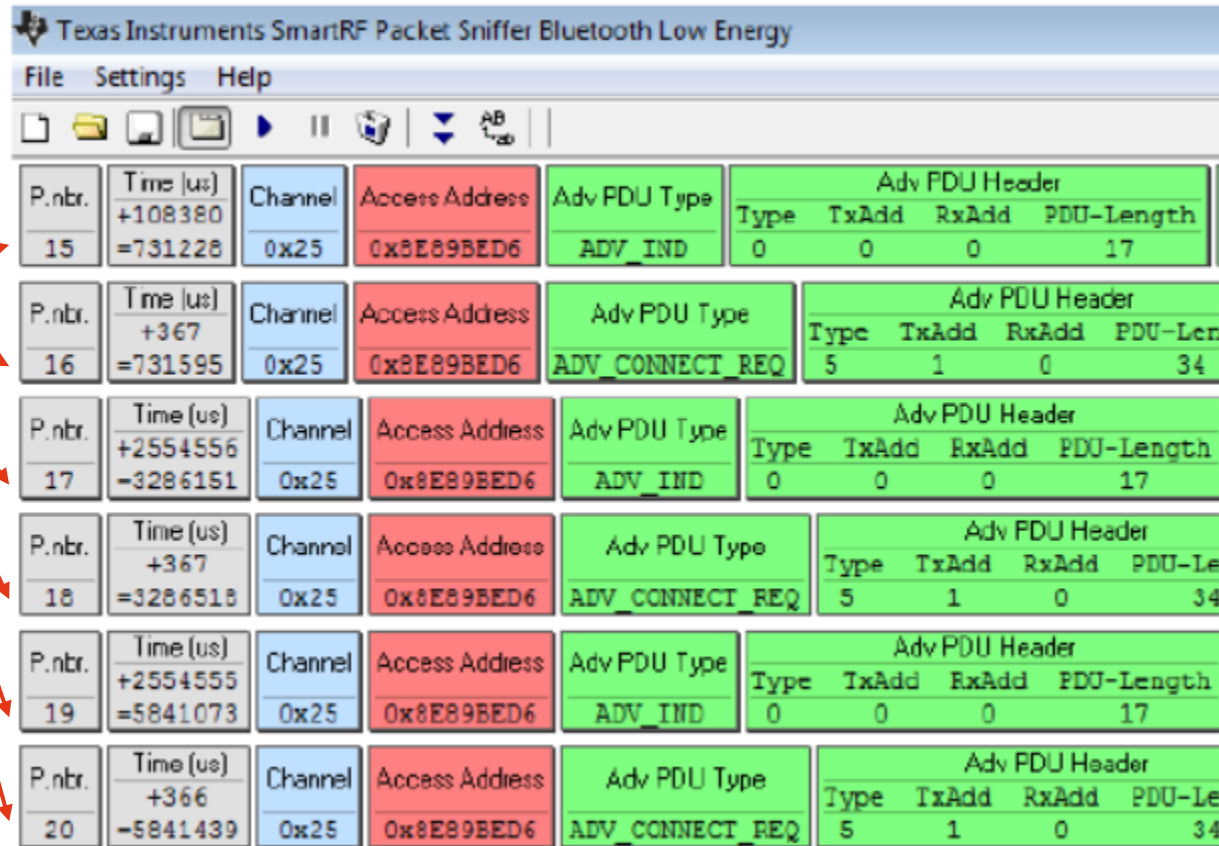
Numéro de paquet



Texas Instruments SmartRF Packet Sniffer Bluetooth Low Energy							
File Settings Help							
P.nbr.	Time (us)	Channel	Access Address	Adv PDU Type	Adv PDU Header		
					Type	TxAdd	RxAdd PDU-Length
15	+108380 =731228	0x25	0x8E89BED6	ADV_IND	0	0	0 17
16	+367 =731595	0x25	0x8E89BED6	ADV_CONNECT_REQ	5	1	0 34
17	+2554556 =3286151	0x25	0x8E89BED6	ADV_IND	0	0	0 17
18	+367 =3286518	0x25	0x8E89BED6	ADV_CONNECT_REQ	5	1	0 34
19	+2554555 =5841073	0x25	0x8E89BED6	ADV_IND	0	0	0 17
20	+366 =5841439	0x25	0x8E89BED6	ADV_CONNECT_REQ	5	1	0 34

Résultat

Numéro de paquet



Texas Instruments SmartRF Packet Sniffer Bluetooth Low Energy							
File Settings Help							
P.nbr.	Time (us)	Channel	Access Address	Adv PDU Type	Adv PDU Header		
					Type	TxAdd	RxAdd PDU-Length
15	+108380 =731228	0x25	0x8E89BED6	ADV_IND	0	0	0 17
16	+367 =731595	0x25	0x8E89BED6	ADV_CONNECT_REQ	5	1	0 34
17	+2554556 =3286151	0x25	0x8E89BED6	ADV_IND	0	0	0 17
18	+367 =3286518	0x25	0x8E89BED6	ADV_CONNECT_REQ	5	1	0 34
19	+2554555 =5841073	0x25	0x8E89BED6	ADV_IND	0	0	0 17
20	+366 =5841439	0x25	0x8E89BED6	ADV_CONNECT_REQ	5	1	0 34

Chaîne (0x25 = 37)

Résultat

Numéro de paquet

Type de packet

Texas Instruments SmartRF Packet Sniffer Bluetooth Low Energy							
File Settings Help							
P.nbr.	Time (us)	Channel	Access Address	Adv PDU Type	Adv PDU Header		
					Type	TxAdd	RxAdd PDU-Length
15	+108380 =731228	0x25	0x8E89BED6	ADV_IND	0	0	0 17
16	+367 =731595	0x25	0x8E89BED6	ADV_CONNECT_REQ	5	1	0 34
17	+2554556 =3286151	0x25	0x8E89BED6	ADV_IND	0	0	0 17
18	+367 =3286518	0x25	0x8E89BED6	ADV_CONNECT_REQ	5	1	0 34
19	+2554555 =5841073	0x25	0x8E89BED6	ADV_IND	0	0	0 17
20	+366 =5841439	0x25	0x8E89BED6	ADV_CONNECT_REQ	5	1	0 34

Chaîne (0x25 = 37)

Résultat

Temps écoulé entre packets

Numéro de paquet

Type de packet

P.nbr.	Time (us)	Channel	Access Address	Adv PDU Type	Adv PDU Header
					Type TxAdd RxAdd PDU-Length
15	+108380 =731228	0x25	0x8E89BED6	ADV_IND	0 0 0 17
16	+367 =731595	0x25	0x8E89BED6	ADV_CONNECT_REQ	5 1 0 34
17	+2554556 =3286151	0x25	0x8E89BED6	ADV_IND	0 0 0 17
18	+367 =3286518	0x25	0x8E89BED6	ADV_CONNECT_REQ	5 1 0 34
19	+2554555 =5841073	0x25	0x8E89BED6	ADV_IND	0 0 0 17
20	+366 =5841439	0x25	0x8E89BED6	ADV_CONNECT_REQ	5 1 0 34

Chaîne (0x25 = 37)

Discussion des résultats et de notre attaque

- L'attaque mise en place demande des très bonnes capacités dans le domaine pour la créer
- L'attaque mise en place demande très peu de connaissances à exécuter après sa création, le matériel nécessaire n'est pas difficile à trouver
- Le score de l'attaque est de 18 (moyen) en utilisant la «common criteria attack evaluation methodology»

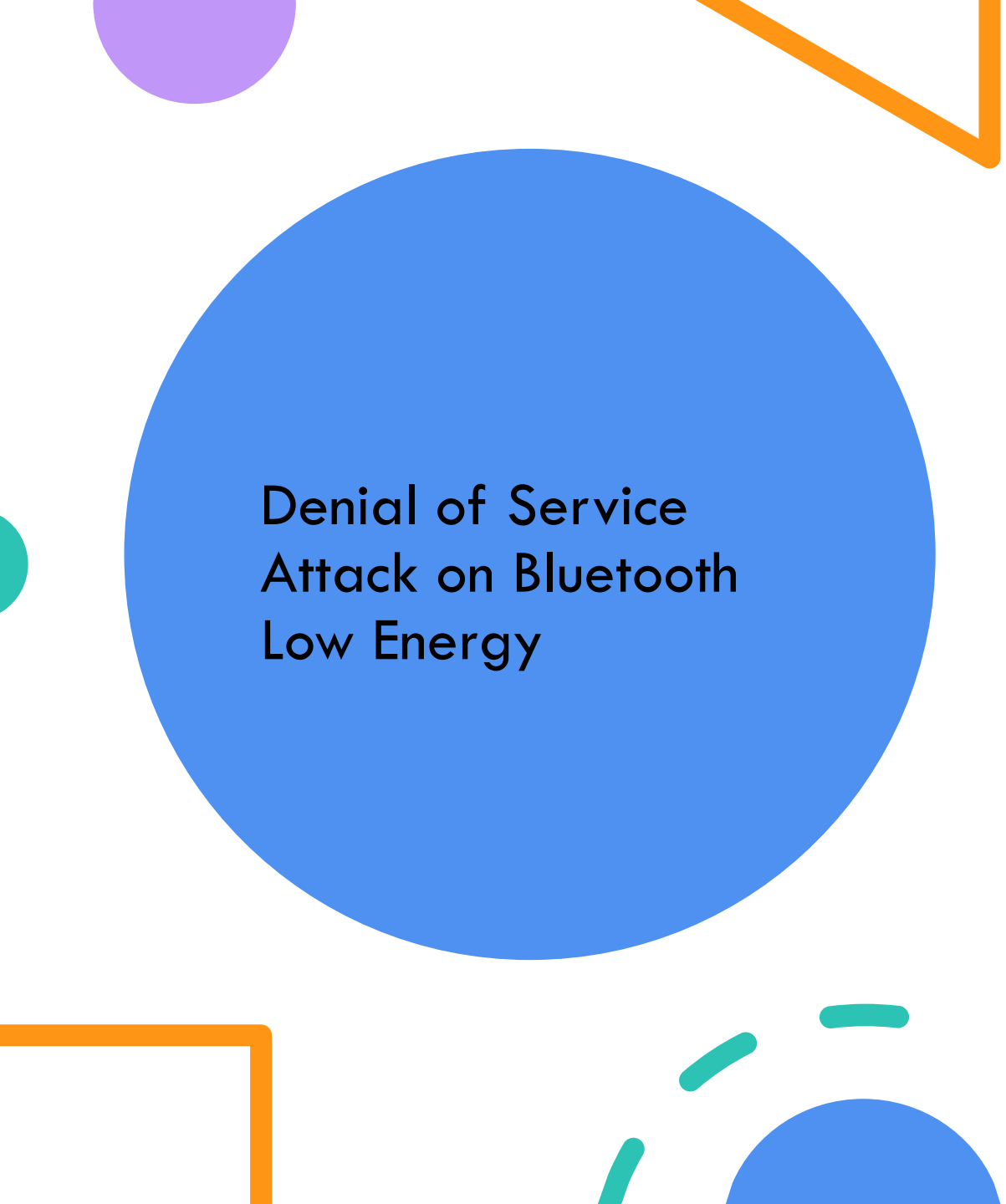
Elapsed time	Expertise	Knowledge of the TOE	Access to the TOE	Equipment
0: <0.5 hour	0: Layman	0: None	0: <0.5 hour	0: None
3: <1 day	2: Proficient	2: Public	4: <1 day	2: Standard
5: < 1 month	4: Expert	4: Sensitive	6: < 1 month	4: Specialised
8: > 1 month		6: Very sensitive	9: > 1 month	6: Bespoke

- L'attaque fait exactement ce qu'on veut, le «slave» est à la recherche d'une connexion, et n'y parvient pas

➡ On bloque le «slave» avec succès, il ne peut rien faire

Travail à faire

- Notre scénario n'est pas réaliste, car il n'existe pas un vrai «master»
- Si un vrai «master» existait il sera à l'écoute sur 1 des 3 chaînes, alors qu'on peut facilement déployer 3 faux «master», 1 par chaîne de connexion
- Dans ce scénario le «slave» a un 1 chance sur 3 de tomber sur la chaîne avec le bon master
- Le faux master peut tout de même être capable de cloquer la connexion si on arrive a «jam» le signal
- La possibilité de «jam», est une fonction de:
 1. Distance entre le vrai «master» et le «slave»
 2. Force de signal du vrai «master»
 3. Distance entre le faux «master» et le «slave»
 4. Force de signal du faux «master»
- La réussite de l'attaque sera donc plus compliqué




Denial of Service Attack on Bluetooth Low Energy

Auteur:
Peter Gullberg

Analyse de l'article



Présentation de l'article

- Titre: Denial of Service Attack on Bluetooth Low Energy
 - Auteur: Peter Gullberg
 - Date de publication: Septembre 2016
 - Type de d'article: proof of concept, identification et exploitation d'un problème
 - Publié par: researchgate.net
 - Citations: 6 fois
- 

Abstract



- 182 mots (informatif)
- Voix active
- Tout le processus est décrit dans l'abstract
- Il explique pourquoi son article est important
- La méthode de recherche est définie
- Les résultats et les implications sont mentionnés

Abstract

Bluetooth Low Energy is a promising technology for wireless communication. The main benefits are that it is energy efficient and is slowly becoming ubiquitous. We can expect that the technology will be used in many demanding applications.

This raises the question whether Bluetooth Low Energy is suitable for products and services that require high resilience, robustness and availability.

In this paper we focus on the availability aspects in the connection setup of Bluetooth Low Energy. We explore an attack path that allows us to do a denial of service attack on the connection setup mechanism. We refine the attack scenario and implement an exploit using the Project Ubertooth, an open source platform for Bluetooth experimentation. We then characterize the attack vector using the Common Criteria attack evaluation methodology.

Our result indicates that it is possible to successfully mount a denial of service attack that blocks the connection setup on Bluetooth Low Energy using standard off-the-shelf components. The consequence of this exploit helps us bring awareness that Bluetooth Low Energy may not guarantee availability when an attacker has the motivation and vicinity access.

Reasons for writing

Problem

Methodology

Results

Implications

Article

- Bien structuré
 1. Introduction
 2. Bluetooth Low Energy Background
 3. Related work
 4. Threat modeling
 5. The exploit
 6. Exploitation
 7. Analysis
 8. Discussion
 9. Conclusions and Future work

Article

(+)

- Le processus de connexion du protocole BLE est expliqué en détail: chaque packet de communication est décrit, jusqu'à la granularité de chaque octet
- Le processus de la création de l'exploit est bien détaillé: il nous informe même quels fonctions ont du être modifiées

(-)

- N'explique pas si Bluetooth classique peut être exploité de la même façon ? Si non, Pourquoi