南京郵電大學
Nanjing University of Posts and Telecommunications

无线个域网安全-低功耗蓝牙LE

汇报人：梁鹏远
汇报日期：5.6

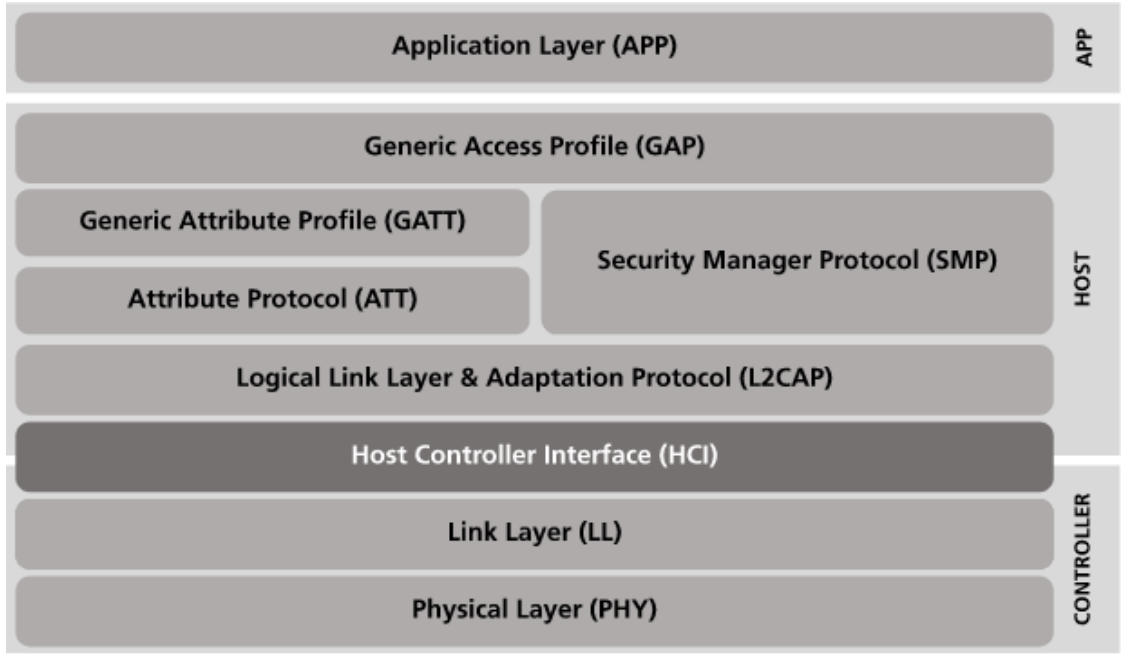
1.引言

蓝牙是一种在2.4GHz工业、科学和医疗（ISM）频段运行的短距离无线协议集。蓝牙规范分为两个主要部分。一部分称为蓝牙BR/EDR，也被称为经典蓝牙，另一部分是蓝牙低功耗（BLE），它是在4.0版本中新增的。两者几乎是完全独立的协议。BLE旨在用于低功耗设备，或者换句话说，用于电源有限且计算和存储能力有限的设备。由于这些设备应在不更换电池的情况下尽可能长时间运行，因此需要像BLE这样的特殊通信协议。其应用领域十分广泛，包括对安全性和可靠性要求较高的领域，例如电子锁、报警系统、过程监控或医疗设备。

这些设备通常通过BLE由智能手机或笔记本电脑进行控制和监控。由于是无线接口，BLE接口特别容易受到潜在攻击。攻击者无需对设备进行物理接触，并且在实施攻击时被发现的风险较低。潜在的攻击目标包括窃听、拒绝服务（DoS）、伪装、注入消息、部分或完全接管连接、跟踪和定位。蓝牙规范针对这些威胁提供了安全措施，引入了多种设备配对方案、可选的连接加密和身份验证，或地址随机化。

2. BLE协议栈

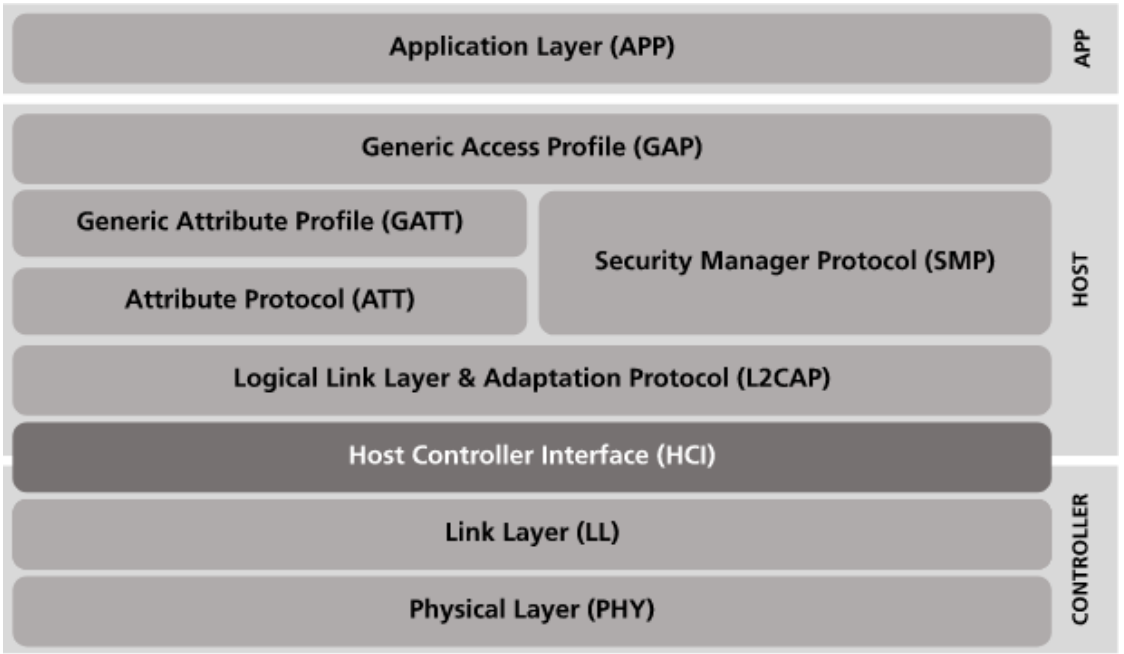
BLE协议栈主要划分为控制器和主机两部分。两部分之间通过主机控制器接口协议（Host Controller Interface, HCI）进行通信，该协议能够将主机的操作转化成HCI指令传给控制器。



控制器部分由物理层（PHY）、链路层（Link Layer, LL）和HCI层组成。主机部分由逻辑链路控制及自适应协议层（Logical Link Control and Adaptation Protocol, L2CAP）、安全管理层（Security Manage Protocol, SMP）、属性协议层（Attribute, ATT）、通用属性协议（Generic Attribute Profile, GATT）和通用访问配置文件层（Generic Access Profile, GAP）组成，上层可以调用下层提供的函数来实现需要的功能。

2. BLE协议栈

协议栈中最上层的GAP用于配置BLE设备的可发现性、广播数据内容和与安全相关的参数，协议栈中其它层需要从GAP中获取配置信息和初始化参数。使用GAP定义的参数建立连接后，便可通过GATT进行双方数据的存取，该协议定义了客户端如何对服务器端的数据进行读写，服务器端如何向客户端发送通知（Notify）和指示（Indication）等。GATT进而使用ATT层提供的属性数据结构、属性类型和权限来描述需要传输的数据[36]，若连接需要安全性保护，则使用SMP提供的加密认证算法和密钥交换方式，数据通过L2CAP和链路层封装由物理层发送。



3. BLE安全机制

蓝牙低功耗（BLE）旨在支持广泛的应用场景，这些场景可能具有相当不同的安全需求和限制，例如设备能力或用户界面。为了满足这些需求，BLE在安全机制的选择和组合方式上非常灵活。其选项范围从完全不使用安全机制，到支持中间人攻击（MITM）防护的密钥交换以及加密和认证的通信。支持的安全功能被组织成安全模式和安全级别，下面将从配对和绑定功能对其进行介绍。

Overview of BLE security modes.

Mode, Level		Pairing security	Message security	Since version
1,	1	No pairing	No security	≥ 4.0
	2	Unauthenticated legacy	Encryption + MAC	≥ 4.0
	3	Authenticated legacy	Encryption + MAC	≥ 4.0
	4	Authenticated LE-SC	Encryption + MAC	≥ 4.2
2,	1	unauthenticated	GATT layer data signing	≥ 4.0
	2	Authenticated	GATT layer data signing	≥ 4.0
3,	1	No pairing	No security	≥ 5.2
	2	Unauthenticated	Encryption + MAC	≥ 5.2
	3	Authenticated	Encryption + MAC	≥ 5.2

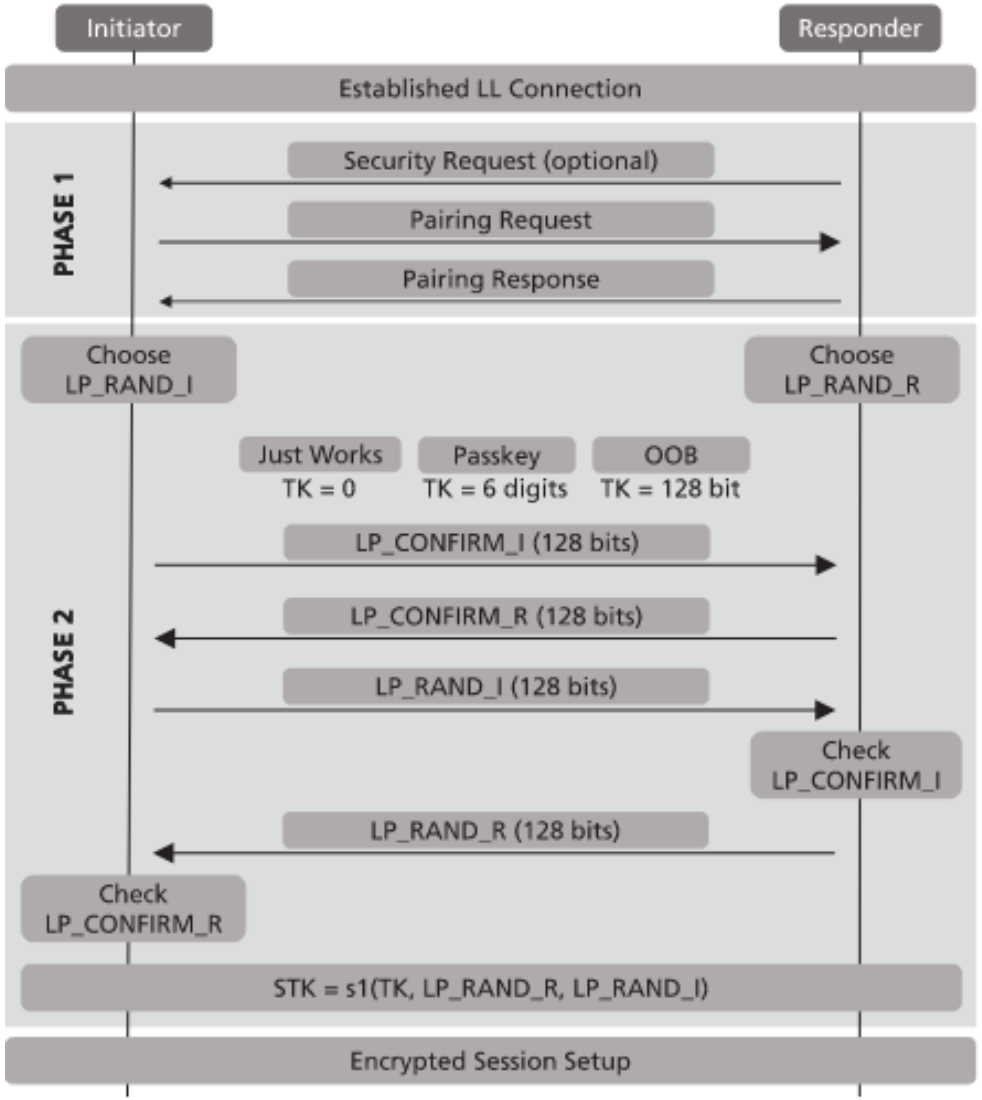
3. BLE安全机制

下表是BLE与普通蓝牙的安全机制对比

对比项目	普通蓝牙	BLE
配对方式	传统配对方式多样，包括 Just Works、Passkey Entry、Out of Band 等。Just Works 无需用户交互，但安全性低；Passkey Entry 要求用户输入 6 位数字 PIN 码，安全性中等但存在暴力破解风险；Out of Band 利用 NFC 等替代通道交换认证信息，安全性高，不过需设备支持额外通信通道。	在蓝牙 v4.2 之前，传统配对是 BLE 中唯一可用的配对方法，存在安全隐患。蓝牙 v4.2 引入了 LE 安全连接，使用椭圆曲线 Diffie-Hellman（ECDH）加密技术生成公钥 - 私钥对，配对过程更安全，且新增了 Numeric Comparison 等用户交互方式，Passkey Entry 也采用逐位验证算法，增大了中间人攻击的难度。
加密算法	BR/EDR 传统配对使用基于 SAFER + 的 E21 和 E22 算法，设备鉴权使用基于 SAFER + 的 E1 算法，加密使用源于 Massey-Rueppel 算法的 E0 算法，加密强度相对较低，且对加密消息完整性没有明确规定，CRC 提供的完整性保护有限，易被伪造。	LE 传统配对使用 AES-CCM 加密，LE 安全连接进一步提升了加密强度和消息完整性验证能力，使用 128 位加密密钥，并支持基于 AES-CCM 加密通信，还使用 AES-CMAC 替换了旧的 HMAC 算法，使消息完整性验证更可靠。
密钥管理	密钥基础为链路密钥，长度为 56-128 位可变，密钥管理相对复杂，设备间共享链路密钥，一旦密钥泄露，整个通信的安全性将受到威胁。	密钥基础为长期密钥（LTK），长度固定为 128 位，密钥管理更高效，且引入了身份解析密钥（IRK）和连接签名密钥（CSRK）等，分别用于保护设备隐私和确保数据完整性。
身份验证	主要依赖配对过程中的 PIN 码验证等方式，安全性相对较低，且在某些配对模式下无法有效防御中间人攻击。	LE 安全连接中，设备通过交换公钥和使用 ECDH 密钥协商技术，结合配对方法验证对等设备的真实性，如 Passkey Entry 中输入的密钥、OOB 中交换的认证信息以及 Numeric Comparison 中用户手动检查和确认的值等，身份验证更可靠、更安全。
数据保护	BR/EDR 对加密消息完整性没有规定，尽管 CRC 提供了一定的完整性保护，但由于其可被轻易伪造因而不被视为提供了加密完整性，数据在传输过程中可能被篡改而不易被察觉。	LE 安全连接不仅对数据进行加密，还通过更强的消息完整性验证机制，如 AES-CCM 和 AES-CMAC 算法，确保数据在传输过程中不被篡改，有效保护了数据的完整性和保密性。

4. BLE配对过程

配对过程是为了在两个BLE设备之间建立和分发密钥，以便加密未来的连接或对交换的数据进行签名。因此，除了安全模式1、级别1（“无安全模式”）之外，此过程是每种安全配置的强制性要求。配对包括右图所示的两个强制阶段，以及一个可选的第三阶段，称为“绑定”



4. BLE配对过程

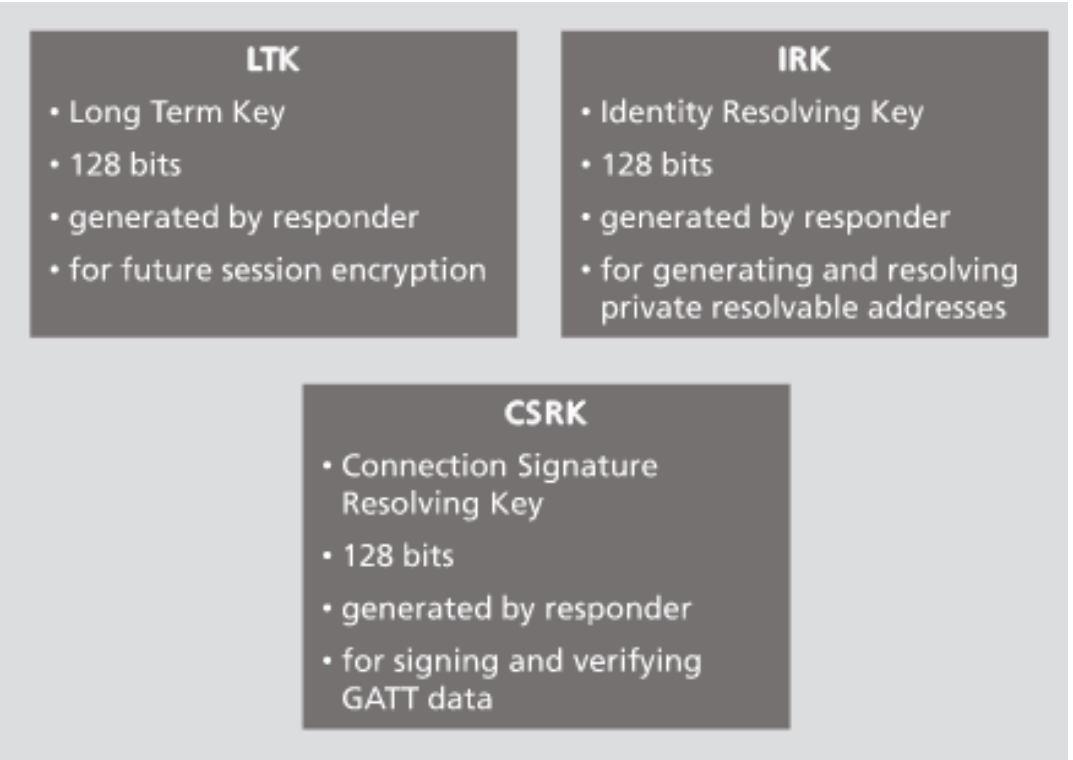
在第一阶段，两个设备交换配对功能，例如它们的输入/输出能力，即是否有显示功能或者是输入功能。

实际的密钥建立是在第二阶段完成的。根据参与设备的输入/输出能力，“输入密码”过程会略有不同。

如果一个设备具有显示功能，而另一个设备具有输入功能，那么具有显示功能的设备会生成一个随机的6位数字密码（范围为“000000”到“999999”）。用户需要在另一个设备的键盘上输入此密码。如果两个设备都只有“仅键盘”功能，则用户必须确保在两个设备上输入的密码相同，并且尽可能随机。这个最多20位的数字密码会被零填充到128位值，并用作临时密钥（TK）。如果通过带外通信进行关联，则TK是一个通过第二个传输通道（例如近场通信，NFC）交换的128位值。对于未经认证的配对，会使用“仅工作模式”（即just work 模式），其中TK直接设置为“0”。

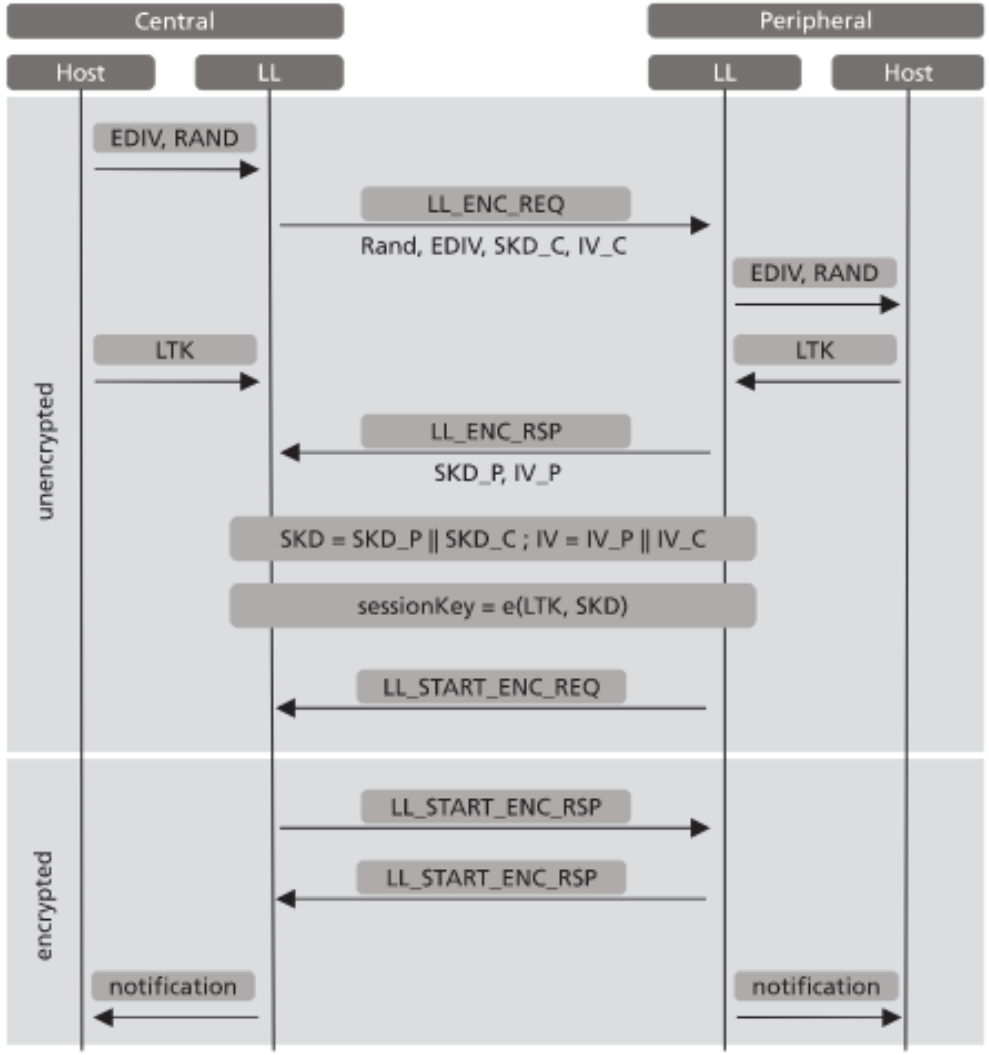
5. BLE绑定过程

"绑定"将在配对第二阶段完成后执行，利用基于短期密钥（STK）建立的加密蓝牙连接进行。在此过程中，设备还可选择性地分发用于解析私有设备地址的身份解析密钥（IRK）以及连接签名解析密钥（CSRK）。CSRK可在未加密的低功耗蓝牙（BLE）连接中使用，通过属性协议PDU在ATT层发送经认证的数据，接收方设备可验证这些数据确系来自声称的发送设备。密钥分发完成后，基于STK的加密蓝牙连接将终止，并立即依照后续加密通信过程使用新获取的LTK重新建立连接。LTK，IRK，CSRK都是由外围设备生成的。



6. 加密会话过程

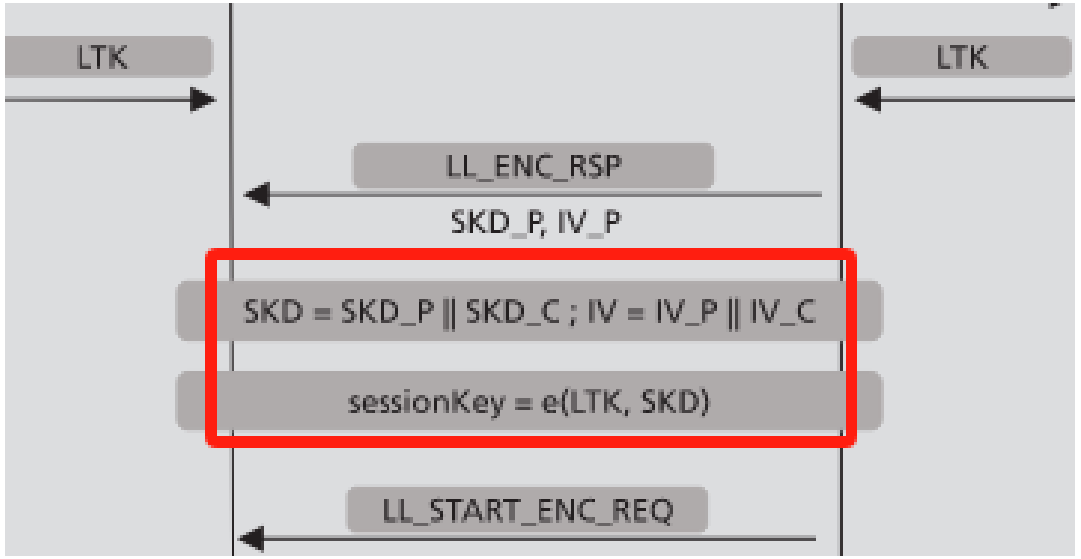
配对成功完成后，加密密钥（STK 或 LTK）已交换。该密钥可用于加密 BLE 连接，执行加密会话设置过程如图 4 所示。如果之前已进行绑定，EDIV 和 Rand 值已作为 LTK 的特定连接标识符交换，以恢复 LTK。在这种情况下，中心设备向外围设备发送这两个值，以允许外围设备从其数据库中选择正确的 LTK。如果仅进行配对，则不会存在 EDIV 和 Rand 值，因为 STK/LTK 未存储。这种情况下，相应的 PDU 字段将被设为零。



6. 加密会话过程

除上述两个值外，两台设备还生成并交换新的会话密钥分散器（SKD_C 和 SKD_P，128 位随机数）以及初始化向量（IV_C 和 IV_P，64 位随机数），使用链路层加密请求 PDU（LL_ENC_REQ）。每个设备将随机数连接如下：

- SKD = SKD_P // SKD_C
- IV = IV_P // IV_C



6. 加密会话过程

下一步，通过 AES128 算法使用会话密钥分散器（SKD）作为明文输入，STK/LTK 作为密钥来计算会话密钥。输出的会话密钥用于加密新会话。最后，执行三步握手，使用 AES-CCM 算法指示加密 BLE 连接的开始。加密过程如下：

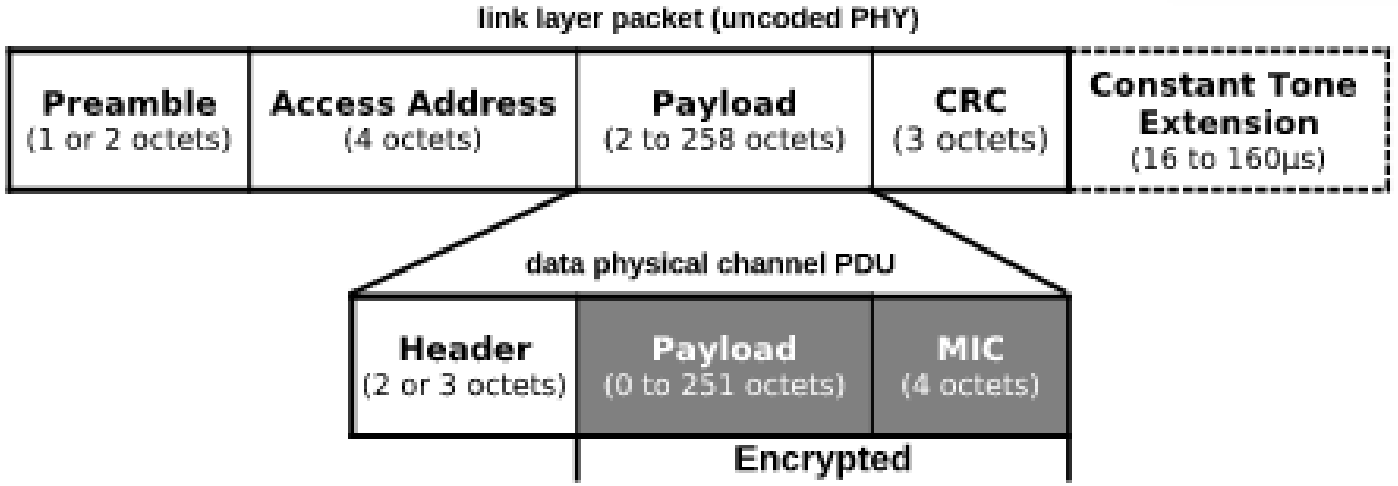
1. 生成认证码（MIC），覆盖数据物理信道 PDU 的有效载荷及其首字节的报头。根据规范，该消息认证码被定义为消息完整性代码（MIC），避免与媒体访问控制（MAC）缩写混淆。

2. 首字节报头作为额外认证数据（AAD）提供给 AES-CCM 算法，并将报头的某些位掩码为零。

3. 对有效载荷和 MIC 进行加密。

6. 加密会话过程

下图展示了 BLE 数据包中受加密保护的部分以及仍以明文传输的部分。未加密的部分所泄露的信息非常有限，主要是前导码、访问地址（AA）和 CRC 不包含敏感信息的字段。



7. BLE漏洞挖掘方法

理论上，除了使用“无安全模式”的设备可能会遭受中间人攻击之外，采用BLE协议的设备在进行通信的过程中应该是很安全的， 但是在实际的实现的过程中，BLE设备还是存在很多漏洞，漏洞挖掘的方式如下表所示。下面将主要介绍一种针对BLE协议的基于状态机学习的模糊测试方法。

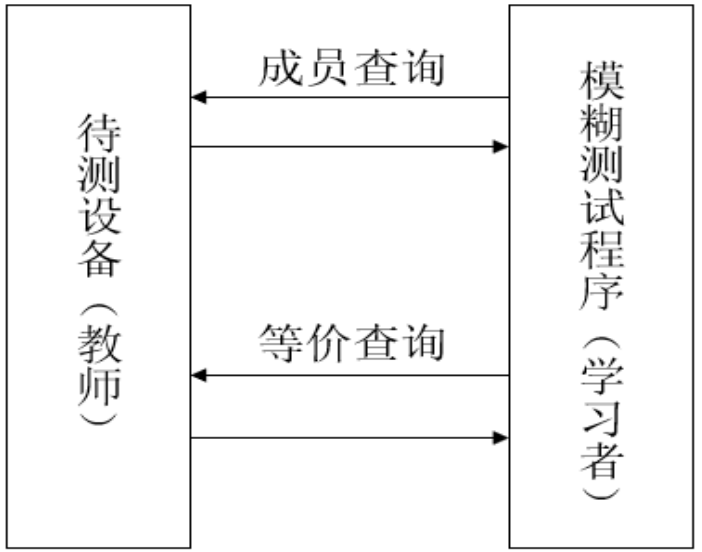
漏洞挖掘方法	说明
源码审计	分析 BLE 协议栈和应用程序的源代码，查找安全漏洞，如缓冲区溢出、整数溢出、格式化字符串漏洞等
模糊测试	使用模糊测试工具生成随机或变异的数据，作为输入发送给 BLE 设备或协议栈，观察是否引发崩溃或异常行为
协议分析	深入研究 BLE 协议规范，分析设备之间的通信数据包，查找协议实现中的漏洞或差异
中间人攻击模拟	搭建中间人攻击环境，截获并篡改 BLE 设备之间的通信数据，测试设备的安全机制是否能够检测和抵御攻击
固件逆向分析	提取 BLE 设备的固件镜像，使用逆向工程工具分析其代码逻辑和程序结构，查找潜在的漏洞
基于模拟器的挖掘	使用模拟器模拟 BLE 设备的运行环境，加载目标固件并进行动态调试和测试，监测固件的运行状态和行为

7. BLE漏洞挖掘方法

首先是状态机学习，

不同 BLE 设备之间的实现存在差异性，使用预先设定好的静态状态模型不能准确反映 BLE 设备的行为差异性，因此需要通过待测设备的消息应答类型来区分不同的状态，并且记录下不同消息序列能够触发的所有状态。

具体通过L*主动学习的方式来推断待测设备的状态机。将BLE的模型学习分为连接准备和配对两个阶段。I1 = [CONNECT_REQ, LL_VERSION_IND, LL_FEATURE_REQ/RSP, LL_LENGTH_REQ/RSP, ATT_EXCHANGE_MTU_REQ]。I2 = [Paring Request, Pairing Confirm, Pairing Random, LL_ENC_REQ, LL_START_ENC_RSP]



7. BLE漏洞挖掘方法

生成的状态存在.dot文件中，如下所示

```
digraph "cc2640r2f-no-feature" {
s0 [label=s0];
s1 [label=s1];
s2 [label=s2];
s3 [label=s3];
s4 [label=s4];
s5 [label=s5];
s6 [label=s6];
s7 [label=s7];
s8 [label=s8];
s9 [label=s9];
s10 [label=s10];
s0 -> s0  [label="scan_req/Adv"];
s0 -> s1  [label="connection_req/BTLE|BTLE_CTRL|BTLE_DATA|LL_LENGTH_REQ"];
s0 -> s0  [label="length_req/Empty"];
s0 -> s0  [label="length_rsp/Empty"];
s0 -> s0  [label="feature_rsp/Empty"];
s0 -> s0  [label="version_req/Empty"];
s0 -> s0  [label="mtu_req/Empty"];
s0 -> s0  [label="pairing_req/Empty"];
s1 -> s0  [label="scan_req/Adv"];
s1 -> s1  [label="connection_req/BTLE|BTLE_CTRL|BTLE_DATA|LL_LENGTH_REQ"];
s1 -> s1  [label="length_req/BTLE|BTLE_CTRL|BTLE_DATA|LL_LENGTH_RSP"];
s1 -> s4  [label="length_rsp/BTLE|BTLE_DATA"];
s1 -> s1  [label="feature_rsp/BTLE|BTLE_DATA"];
```

7. BLE漏洞挖掘方法

然后是模糊测试时过程，该阶段会生成与执行测试序列

测试序列构造
每个抽象测试序列都由三个部分组成：

1）前缀 (p)：访问序列，用来将设备引导到某个特定状态。

2）模糊输入 (f)：在这个状态下选取一个或多个命令，并有意将其中的某个字段进行“模糊化”（例如选择边界值或随机值），以生成有效值之外的输入。

3）后缀 (s)：继续发送一系列后续指令，用于观察模糊输入后设备的状态迁移和反应。

测试序列=前缀序列+异常序列+后缀序列

7. BLE漏洞挖掘方法

将生成的具体测试序列依次发送给目标设备，并记录设备的响应。 每次测试后，比较实际输出与学习到的模型中对应状态下预期的输出。 如果发现两者不一致（即输出不匹配或状态转移异常），则认为找到了一个“反例”。

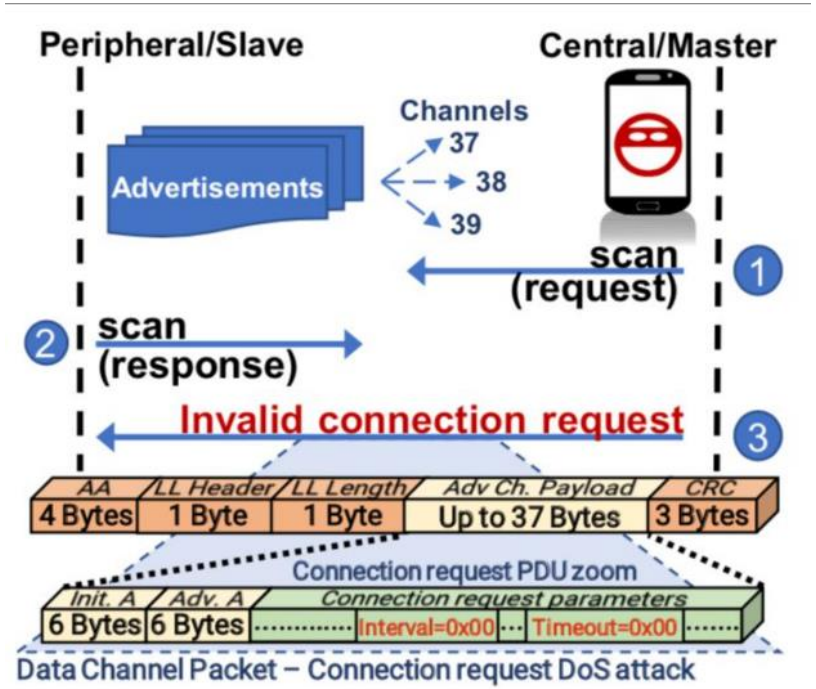
为了排除偶然性错误（比如由于无线传输的延迟或偶发的连接问题），一旦检测到非预期的响应，会重复执行相同的测试序列（例如多达 ncex 次），以确认反常行为是否可重现，确认反例后，利用类似于 W-Method 的技术，结合一个自动导出的表征集，确认这一异常的状态转移是否代表设备进入了未知或错误的状态。

现实漏洞分析

Invalid Connection Request (CVE-2019-19193)

当主机设备尝试连接到 TI CC2540 SDK（v1.5.0及更低版本）时，协议栈无法正确处理非法连接参数，导致从设备进入空闲状态（无广播）。

在 BLE 连接初始阶段，主设备扫描从设备的广播数据包，并发送连接请求，其中包含连接间隔和超时参数。这些参数控制数据交换和超时，必须是非零值。若主设备发送无效请求，且间隔或超时为零，从设备将停止广播。协议栈会发送连接失败事件（bleGAPConnNotAcceptable），SDK默认进入空闲状态，停止广播。



7.参考文献

[1]Pferscher A, Aichernig B K. Stateful black-box fuzzing of bluetooth devices using automata learning[C]//NASA Formal Methods Symposium. Cham: Springer International Publishing, 2022: 373-392.

[2] Cäsar M, Pawelke T, Steffan J, et al. A survey on Bluetooth Low Energy security and privacy[J]. Computer Networks, 2022, 205: 108712.

[3] Garbelini M E, Chattopadhyay S, Wang C. Unleashing Mayhem over Bluetooth Low Energy[EB/OL].(2020-7)

谢谢！

计算机学院 软件学院
网络空间安全学院
School of Computer Science