УДК 681.326

А.А. Волошин, Б.Б. Працюк, Ю.В. Прокопенко, канд. техн. наук

Особенности реализации алгоритма простого сопряжения Bluetooth-устройств

Рассмотрен алгоритм простого сопряжения Bluetooth-устройств, который позволяет легче и безопаснее проходить стадии аутентификации (опознания устройств) и создания общего ключа для дальнейшего шифрования данных. Алгоритм был реализован на основе платформы ТС-3000 и может применяться в Bluetooth-устройствах с различными возможностями (наличие или отсутствие дисплея, возможности ввода текста или цифр и т.п.).

The paper presents Secure Simple Pairing algorithm for Bluetooth-enabled devices, which provides easier and safer pass of authentication and encryption key generation stages. The algorithm was implemented in TC-3000 platform and can be used in Bluetooth devices with different input/output (IO) capabilities such as presence/absence of the display, presence/absence of the text or numbers entering possibility, etc.

Введение

Bluetooth является наиболее распространенным и перспективным представителем технологий беспроводной передачи данных в ближнем радиусе действия. Во время передачи данных информационный поток может подвергаться пассивной (прослушивание и анализ трафика) или активной (возможность модифицирования передаваемого сообщения и вставки своих сообщений) атаке. Модификация потока данных, так называемая атака типа «Man In The Middle» (MITM), представляет наибольшую опасность в таких сферах, как здравоохранение и медицина.

В этих областях используются устройства с ограниченными возможностями ввода и вывода (например: подкожные датчики). Для таких устройств был разработан алгоритм простого сопряжения. Алгоритм простого сопряжения позволяет значительно упростить процесс аутентификации и сделать передачу данных более безопасной по сравнению с традиционным алгоритмом сопряжения.

1. Механизмы защиты информации при передаче данных между Bluetooth-устройствами

Технические требования Bluetooth определяют несколько возможностей защиты информации. Помимо ограниченного радиуса действия и использования скачкообразной перестройки частоты, что чрезвычайно затрудняет перехват сигнала, технические требования Bluetooth определяют также функции аутентификации и шифрования.

Спецификация Bluetooth имеет три основных защитных режима, которые могут использоваться как по отдельности, так и в различных комбинациях.

В *защитном режиме 1* – никаких мер для безопасного использования Bluetooth-устройства не предпринимается. Каждое устройство имеет доступ ко всем сервисам без ограничений.

В *защитном режиме 2* – активируются меры безопасности, основанные на процессах опознания (аутентификации — authentication) и разрешения (авторизации — authorization). В этом режиме определяются различные уровни «доверия» (trust) для каждого сервиса, предложенного устройством.

Защитный режим 3 – требует опознания и шифрования (или encryption) [1].

Защитные режимы 2 и 3 могут использоваться совместно. При этом вначале устанавливается защищённое соединение, а потом устанавливается дополнительная степень защиты в соответствии с требованиями и возможностями конкретной службы.

Основой системы безопасности Bluetooth, используемой в защитном режиме 3, является понятие сеансового ключа. Сеансовый ключ генерируется в процессе сопряжения двух устройств и используется для аутентификации и шифрования передаваемых данных. Для генерации ключа могут использоваться самые различные составляющие, от заранее известных обоим устройствам значений до физических адресов устройств.

С точки зрения безопасности особенно слабым аспектом Bluetooth является процесс сопряжения (pairing) устройств, при котором происходит обмен ключами в еще незакодированных каналах. При перехвате данных в процессе сопряжения возможно получение ключа инициализации путем расчетов и подбора с последующим определением ключа связи.

Алгоритм безопасного простого сопряжения позволяет легче и безопаснее проходить стадии аутентификации и создания общего ключа для дальнейшего шифрования.[2]

2. Реализация алгоритма простого спряжения Bluetooth-устройств

Безопасное простое сопряжение (Secure Simple Pairing) использует простые формы криптографии с открытым ключом, и включает 5 основных этапов, состоящих из 13 шагов (рис. 1):

Этап 1: Обмен общедоступными ключами
Этап 2: Аутентификация, стадия 1
Этап 3: Аутентификация, стадия 2
Этап 4: Вычисление общего ключа канала
Этап 5: LMP-аутентификация и шифрование

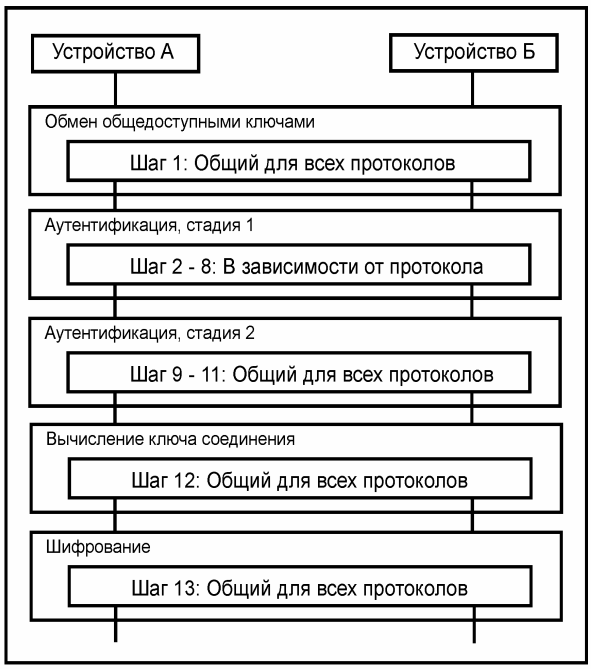

Рис. 1. Стадии безопасного простого сопряжения, где устройство А – инициатор связи, а устройство В – принимающее устройство

В зависимости от возможностей устройства (наличие или отсутствие дисплея, возможности ввода текста или цифр и т.п.) используют разные протоколы для прохождения второго этапа. Все остальные этапы идентичны для всех устройств.

Этап 1: Обмен общедоступными ключами

Для безопасного простого сопряжения двух устройств на первом этапе используется метод эллиптических кривых Диффи-Хеллмана [3]. С открытым и закрытым ключами. Эта пара ключей должна быть сгенерирована вначале сопряжения и обычно используется до конца, и первое и второе устройство в любой момент времени могут отказаться от пары сгенерированных ключей и создать новую пару, начав при этом весь процесс сопряжения сначала.

Обмен ключами с использованием эллиптических кривых выполняется следующим образом. Сначала выбирается простое число p, не превышающее 2^{180}, и параметры a и b для уравнения эллиптической кривой. Эти параметры задают кривую E_p (a,b). Затем выбирается генерирующая точка $G = (x_1,y_1)$, лежащая на этой кривой. Параметры E_p (a,b) и G криптосистемы известны всем участникам. В спецификации Bluetooth v.2.1 используется кривая NIST P192 [4].

Обмен ключами между устройствами А и В производится по следующей схеме:

1. Устройство А выбирает целое число SK_a, не превышающее ограничивающего значения r, которое задается вместе с параметрами кривой. Это число является закрытым ключом устройства А. Затем устройство А вычисляет открытый ключ $PK_a = SK_a \times G$, который представляет некоторую точку на кривой E_p (a,b).
2. Аналогично, устройство В выбирает закрытый SK_b и вычисляет открытый ключ PK_b.
3. Участники обмениваются открытыми ключами (стадии 1 a и 1 b, рис. 2), после чего общий секретный ключ DH_{key} вычисляется следующим образом:

$DH_{key} = P192 (SK_a, PK_b) = SK_a \times PK_b$, для участника А,

$DH_{key} = P192 (SK_b, PK_a) = SK_b \times PK_a$, для участника Б.

Криптографическая стойкость алгоритма обмена ключами при использовании метода эллиптических кривых Диффи-Хеллмана определяется сложностью вычисления секретного ключа SK_a по известным значениям DH_{key} и PK_b). В математике эта задача рассматривается как дискретное логарифмирование на эллиптической кривой,и которая для больших простых чисел считается неразрешимой либо несопоставимой по времени со временем прохождения данного этапа [5].

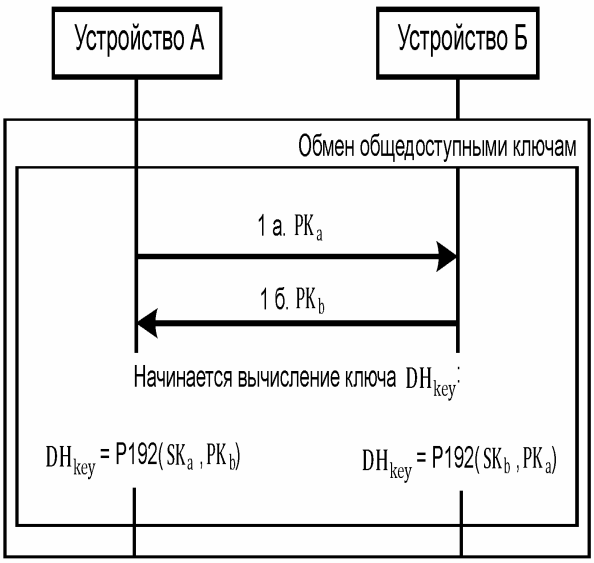

Рис. 2. Обмен общедоступными ключами, где устройство А – инициатор связи

Этап 2: Аутентификация, *стадия 1*

Основываясь на возможности ввода и вывода устройств, различают три типа протоколов:

- протокол численного сопоставления (Numeric Comparison): используется, если оба устройства имеют дисплей и возможность ввода значений да или нет. При этом на каждом из устройств отображается шестизначный цифровой код, а пользователь должен сравнить отображаемые номера и, если они идентичны, подтвердить сопряжение устройств.
- протокол входа с ключом доступа (Passkey Entry): используется между устройством с дисплеем и устройством с цифровой клавиатурой, или для двух устройств с цифровыми клавиатурами. В первом случае дисплей используется, чтобы показать 6-значный цифровой код для пользователя, который затем вводится на клавиатуре. Во втором случае пользователь каждого из устройств вводит одинаковый 6-значный номер. В обоих случаях предоставляется защита от MITM атак.
- внеполосной протокол (Out-Of-Band): Этот метод использует внешние средства связи (например, Near field Communication) для обмена информации, используемой в процессе сопряжения.

Протокол численного сопоставления

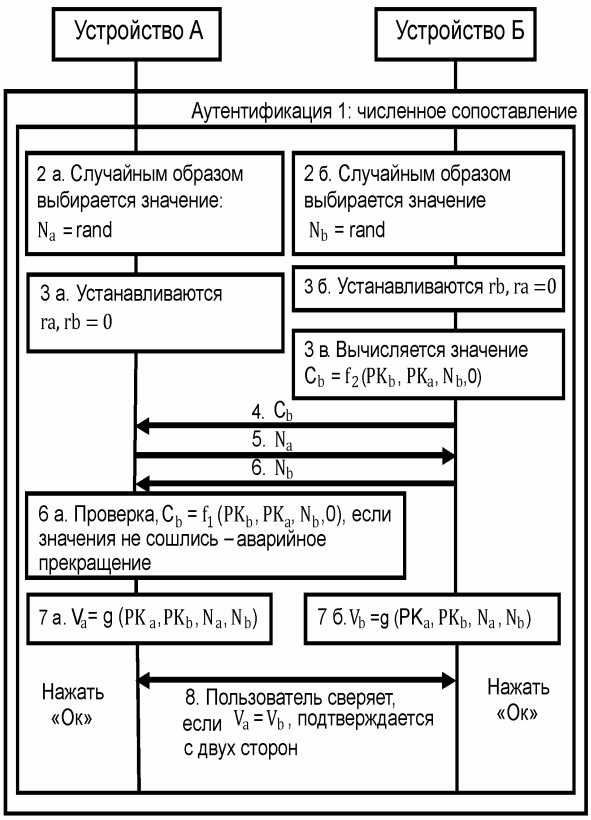

Рис. 3. Алгоритм аутентификации стадии 1: шаги в протоколе численного сопоставления, где устройство А – инициатор связи

После обмена общедоступными ключами каждое устройство выбирает псевдослучайное 128-битное число N_a, N_b (шаг 2, рис. 3). Это значение используется для предотвращения повторяющихся атак и обязательно обновляется при каждом новом сопряжении.

Отвечающее устройство по трем известным значениям: PK_b, PK_a и N_b вычисляет значение C_b (шаг 3в, рис. 3) при помощи функции f_2 (PK_a,PK_b,N_b,0), которая описана после алгоритма. Затем, вычисленное значение передается устройству А и сравнивается с требуемым значением (шаг 6а, рис. 3). Несоответствие сравниваемых значений указывает на наличие атаки или ошибки при передаче данных. При этом нужно повторить второй этап сначала, или начать сопряжение с первого этапа.

В случае получения верных значений оба устройства вычисляют значение V_a при помощи функции g (PK_a,PK_b,N_a,N_b), которая описана после алгоритма. Для вывода шестизначного числа на экран, его берут по mod10^6. Числа выводятся на экраны пользователей, и сравниваются (шаг 7а, 7б и 8, рис. 3). В случае соответствия пользователь подтверждает совпадение.

Благодаря качествам хэш-функции при использовании данного протокола атаки MITM выявляются с вероятностью 0.999999.

Протокол входа с ключом доступа

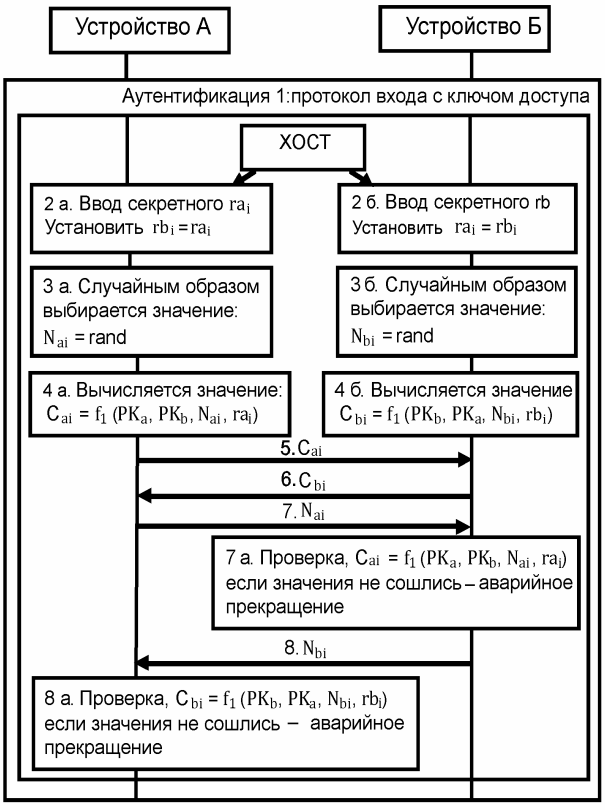

Рис. 4. Алгоритм аутентификации стадии 1: шаги в протоколе с ключом доступа, где устройство А – инициаторсвязи

Пользователем с хоста вводится код доступа, идентичный для обоих устройств. После чего вычисляются при помощи функции f_1, которая описана ниже, значения $C_{ai}=f_1(PK_a, PK_b, N_{ai}, r_{ai})$ и $C_{bi} = f_1(PK_b, PK_a, N_{bi}, r_{bi})$ для дальнейшего сопоставления и проверки правильности вычислений.

Внеполосной протокол

Этот алгоритм относится к симметричным, что дает возможность принимающему устройству в любой момент стать передающим, соответственно передающее устройство становится принимающим.

Этот алгоритм не является стойким к атакам MITM. В случае необходимости повышения устойчивости к атакам MITM используют протокол численного сопоставления.

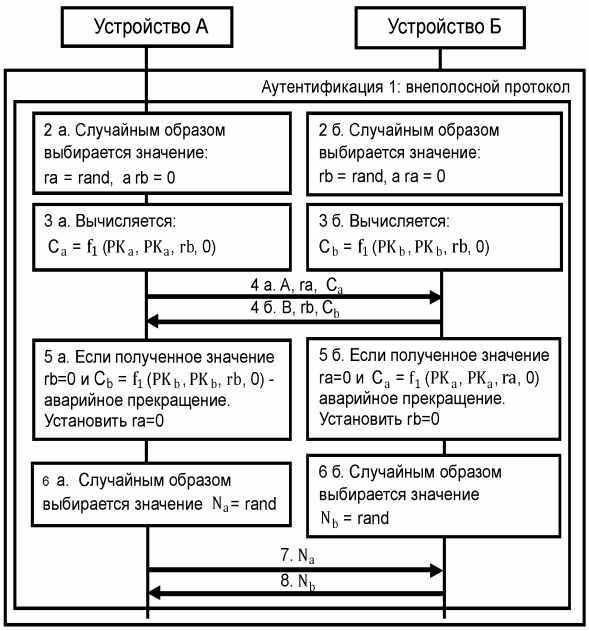

Рис. 5. Алгоритм аутентификации стадии 1: шаги в внеполосовом протоколе

Этап 3: Аутентификация, стадия 2

Вторая стадия аутентификации подтверждает, что оба устройства успешно завершили обмен данными. Эта стадия одинакова для всех трех вышеприведенных протоколов.

Каждое из устройств вычисляет новое значение E, при помощи функции f_3, которая описана ниже, подтверждающее правильность ранее полученных значений (шаг 9, рис. 6). После этого значение $E_a = f_3(DH_{key}, N_a, N_b, rb, IOcapA, A, B)$ устройства А передается устройству В, где оно проверяется. Если проверка неудовлетворительна, то передающее устройство не проходит сопряжение и обязано повторить процедуры. Таким же образом значение устройства Б проверяется устройством А.

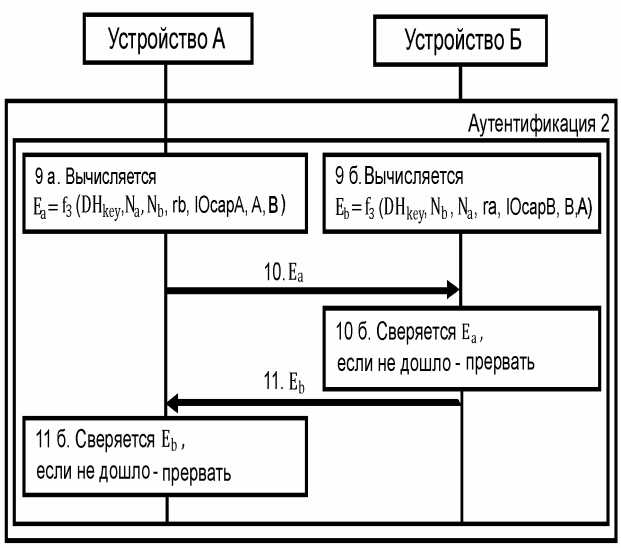

Рис. 6. Алгоритм аутентификации стадии 2: для всех протоколов, где устройство А – инициатор связи, IOcapA, IOcapB – значения, указывающие на способность устройств к вводу/выводу

Если значения были переданы без ошибок, то переходят к шагу 12 (рис. 7).

Этап 4: Вычисление общего ключа канала

После того, как оба устройства прошли стадию сопряжения, вычисляется общий ключ канала, который состоит из общего ключа DHKey и информации о устройствах.

Рис. 7. Завершающая стадия алгоритма: вычисление ключа соединения

Этап 5: LMP идентификация и шифрование

Завершающая стадия в простом сопряжении – обмен сообщениями на уровне LMP (Link Manager Protocol), подтверждающие успешное прохождение стадии аутентификации, что позволяет сгенерировать общий ключ соединения и использовать его для дальнейшего шифрования данных.

Функции, используемые в алгоритме

$$g(U,V,X,Y) = SHA\text{-}256(U//V//X//Y)\bmod 2^{32} \quad (1)$$

где U, V – 192-битные значения, X,Y – 128-битное

$$f_1(U,V,X,Z) = HMAC\text{-}SHA\text{-}256_X(U||V||Z)\bmod 2^{128} \quad (2)$$

где *U,V* – 192- битные значения, *X* – 128 битное, *Z* – 8-битное.

$$f_2(W,N_1,N_2,A_1,A_2)=HMAC\text{-}SHA\text{-}256_x(W \,||\, N_1||N_2||A_1||A_2)/2^{128} \quad (3)$$

где *W* – 192- битные значения, N_1,N_2 – 128 битные, A_1, A_2 – 48-битные.

$$f_3(W,N_1,N_2,R,IOcap,A_1,A_2)=HMAC\text{-}SHA\text{-}256_x(W \,||\, N_1||N_2||R \,||\, IOcap||A_1||A_2)/2^{128} \quad (4)$$

где *W* – 192- битные значения, N_1 ,N_2 – 128 битные, *R* – 128-битное, *IOcap* – 24-битное, A_1 ,A_2 – 48-битные.

Хэш-функция – односторонняя функция, предназначенная для получения дайджеста файла, сообщения или некоторого блока данных. Безопасный хэш-алгоритм (Secure Hash Algorithm) SHA-256 выполняются следующим образом. Входное значение (сообщение, файл и т.п.) рассматривается как последовательность блоков размером n-бит. Входное значение обрабатывается последовательно, блок за блоком, и создается значение хэш-кода, который имеет размер m-бит и осуществляет продольный избыточный контроль.

Один из способов обеспечения целостности данных - это вычисление кода MAC (Message Authentication Code). В данном случае под кодом MAC понимается аутентификатор, являющийся контрольной суммой. В нашем случае используется алгоритм HMAC, который описан в RFC 2104 [6]. Он основан использования хэш-функции и состоит в том, чтобы определенным образом добавить секретное значение к сообщению, которое подается на вход хэш-функции.

Функции SHA-256, HMAC-SHA-256$_x$, а также способ генерации ключей с использованием эллиптических кривых, являются открытыми и представлены в криптографическом пакете с открытым исходным кодом для работы с SSL/TLS, а именно в OpenSSL 0.9.8 [7].

3. Апробация алгоритма

Описанный алгоритм был реализован на основе платформы Bluetooth-тестера TC-3000. Основные этапы обмена сообщениями на уровне LMP представлены на рис. 8. Приведенные LMP-сообщения подтверждают правильность прохождения стадии аутентификации. Пройдя эту стадию, был сгенерирован общий ключ соединения для дальнейшего шифрования данных. Зашифрованная информация была успешно дешифрованы принимающей стороной.

Таблица 1. Основные характеристики хэш-функции SHA - 256

Алгоритм	Длина сообщения (в битах)	Длина блока (в битах)	Длина дайджеста (в битах)	Безопасность (в битах)
SHA - 256	$<2^{64}$	512	256	128

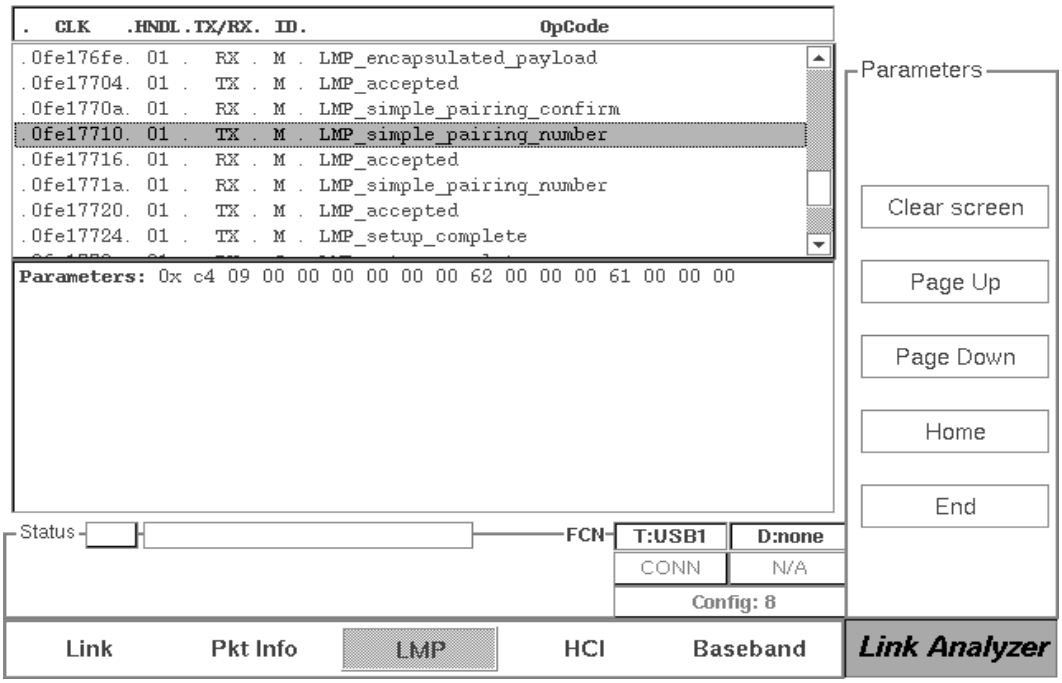

Рис. 8. Последовательность LMP-сообщений при простом сопряжении, зарегистрированная Bluetooth-тестером TC-3000

Выводы

Защита Bluetooth базируется на трех основных процессах: опознание, разрешение и шифрование. Из этого можно сделать вывод, что сильный ключ соединения (сеансовый ключ) вместе с стойким шифрующим алгоритмом способны защитить передаваемую информацию от перехвата и прослушивания.

Сеансовый ключ генерируется в процессе сопряжения двух устройств.

Алгоритм безопасного простого сопряжения позволяет легче и безопаснее проходить стадии идентификации (опознания устройств) и создания общего ключа для дальнейшего шифрования.

Алгоритм был апробирован и может применяться в Bluetooth-устройствах с различными возможностями ввода и вывода в зависимости от выбранного протокола, что позволяет применять его в устройствах с очень ограниченными возможностями. В случае использования протокола численного сопоставления, атака типа «MITM» будет выявлена с вероятностью 0.999999.

Литература

1. *Bouhenguel R., Mahgoub I., Ilyas M*. Bluetooth Security in Wearable Computing Applications. - HONET 2008. International Symposium on Volume, Issue, 18-20. - Nov. 2008 - p.182 – 186
2. *Sharmila D., Neelaveni R., Kiruba, K*. Bluetooth Man-In-The-Middle attack based on Secure Simple Pairing using Out Of Band association model. - INCACEC 2009.International Conference on Volume Issue , 4-6. - June 2009 – p. 1 – 6.
3. www.intuit.ru/department/security/networksec/11/
4. Specification of the Bluetooth System, v.2.1 / SIG. - 2007. – 1400 p.
5. *Бабенко Л.К., Курилкина А.М.* Алгоритмы «распределенных согласований» для оценки вычислительной стойкости криптоаглоритмов. – М.:ДМК Пресс, 2008. - 112с.
6. http://tools.ietf.org/html/rfc2104
7. www.openssl.org
8. *Chatschik Bisdikian*. An Overview of the Bluetooth Wireless Technology // IEEE Communications Magazine. – Dec. 2001. - Vol. 39, p. 86-94.