Preliminary study:
BLUETOOTH SECURITY

Sil Janssens
Sil.Janssens@vub.ac.be

January 9, 2005

Last updated 22/11/2004

<table>
<thead>
<tr>
<th>Date</th>
<th>Author</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>22/11/2004</td>
<td>Sil Janssens</td>
<td>Additions and corrections</td>
</tr>
<tr>
<td>after remarks of Robert Maier and Dave Singelee</td>
<td></td>
<td></td>
</tr>
<tr>
<td>09/11/2004</td>
<td>Sil Janssens</td>
<td>Additions and corrections</td>
</tr>
<tr>
<td>04/11/2004</td>
<td>Sil Janssens</td>
<td>Adaptations after remarks of Philip Cara</td>
</tr>
<tr>
<td>27/10/2004</td>
<td>Sil Janssens</td>
<td>Adaptations after remarks of Dave Singelee</td>
</tr>
<tr>
<td>18/10/2004</td>
<td>Sil Janssens</td>
<td>Corrections</td>
</tr>
<tr>
<td>17/10/2004</td>
<td>Sil Janssens</td>
<td>Corrections and additions</td>
</tr>
<tr>
<td>16/10/2004</td>
<td>Sil Janssens</td>
<td>Corrections and additions</td>
</tr>
<tr>
<td>14/10/2004</td>
<td>Sil Janssens</td>
<td>Corrections and additions</td>
</tr>
<tr>
<td>10/10/2004</td>
<td>Sil Janssens</td>
<td>Corrections and additions</td>
</tr>
<tr>
<td>6/10/2004</td>
<td>Sil Janssens</td>
<td>First Draft</td>
</tr>
</tbody>
</table>

Table 1: Version History
# Contents

1 Introduction 3
   1.1 Purpose and scope 3
   1.2 Definitions, acronyms and abbreviations 3
   1.3 References 7

2 Bluetooth overview 14
   2.1 Bluetooth specifications 14

3 Overall security description 15

4 Bluetooth security overview 16
   4.1 Security Mode 1: Nonsecure mode 16
   4.2 Security Mode 2: Service-level enforced security mode 17
   4.3 Security Mode 3: Link-level enforced security mode 17
   4.4 Bluetooth Key Generation from PIN 17
   4.5 Bluetooth Authentication 18
   4.6 Bluetooth Encryption Process 19
   4.7 Problems with the Bluetooth Standard Security 20

5 Bluetooth security attacks 22
   5.1 Impersonation attack by inserting/replacing data 22
   5.2 Bluejacking 22
   5.3 Bluetooth Wardriving 23
   5.4 Nokia 6310i Bluetooth OBEX Message DoS 24
   5.5 Brute-Force attack 24
   5.6 Denial-of-Service attack on the device 24
   5.7 Disclosure of keys 24
   5.8 Unit key attacks 25
   5.9 Backdoor attack 25
   5.10 Pairing attack 25
   5.11 BlueStumbling = BlueSnarfing 25
   5.12 BlueBug attack 26
   5.13 PSM Scanning 26
   5.14 Off-line PIN (via Kinit) recovery 26
   5.15 On-line PIN cracking 27
   5.16 Off-line encryption key (via Kc) 27
   5.17 Attack on the Bluetooth Key Stream Generator 27
   5.18 Reflection Attack 27
   5.19 Replay attacks 27
   5.20 Man-in-the-middle attack 27
   5.21 Denial-of-Service attack on the Bluetooth network 27
   5.22 A man-in-the-middle attack using Bluetooth in a WLAN interworking environment 27
   5.23 Impersonate original sending/receiving unit 28
   5.24 Correlation attacks 28
1 Introduction

Bluetooth wireless technology is a short-range radio technology that is designed to fulfill the particular needs of wireless interconnections between different personal devices, which are very popular in today’s society. The development of Bluetooth started in the mid-1990s, when a project within Ericsson Mobile Communications required a way to connect a keyboard to a computer device without a cable. The wireless link turned out to be useful for many other things, and it was developed into a more generic tool for connecting devices. A synchronous mode for voice traffic was added and support for up to seven slaves was introduced. In order to gain momentum for the technology and to promote acceptance, the Bluetooth Special Interest Group (SIG) was founded in 1998. The group consists of many companies from various fields. By joining forces, the SIG members have evolved the radio link to what is now known as Bluetooth wireless technology.

1.1 Purpose and scope

The purpose of this document is to provide an introduction to the security aspects of Bluetooth and the existing attacks on the security of Bluetooth.

1.2 Definitions, acronyms and abbreviations

- ACL: Asynchronous connection-oriented (logical transport).
- ACO: Authenticated ciphering offset. A parameter binding devices to a particular authentication event.
- AES: Advanced Encryption Standard
- AG: Audio gateway. A mobile phone or other outloud-playing device (connected to a headset).
- BB: Baseband. This is the lowest layer of the Bluetooth specification.
- BD_ADDR: Bluetooth device address
- BER: Bit error rate. Average probability that a received bit is erroneous.
- CAC: Channel access code. A code derived from the master device address in a Bluetooth connection
- CAK: Common access key. A common key that can be used when connecting to different access points belonging to a particular network provider.
- CID: Channel identifier. End points at an L2CAP channel.
- COF: Ciphering offset. Additional secret input to ciphering key generation procedure.
- CPU: Central processing unit
• CRC: Cyclic redundancy check. A checksum added to the payload by the sender that the receiver can use to detect transmission errors.

• DAC: Device access code. A code derived from a specific slave device in a Bluetooth connection

• DH: Diffie-Hellman. The name of the first public key exchange scheme.

• DoS: Denial of service. Incident in which a user or organization is deprived of the services of a resource they would normally expect to have.

• DSP: Digital signal processor. Editing of sounds in order to produce different sound effects.

• DT: Data terminal

• E0: Bluetooth ciphering algorithm built around four independent linear feedback registers and a finite state machine as a combining circuitry. The final state machine is needed to introduce sufficient nonlinearity to make it difficult to recompute the initial state from observing key stream data.

• E1: Bluetooth authentication function build around SAFER+. E1 is called a Message Authentication Code (MAC) algorithm.

• E21: Bluetooth unit key algorithm, used for unit key derivation, build around a slightly modified SAFER+ algorithm. Because of this, the algorithm E21 cannot be used directly as an invertible encryption algorithm.

• E22: Bluetooth initial key algorithm. Used for initial key derivation and also build around a slightly modified SAFER+ algorithm. E21 and E22 are very similar, this simplified the implementation.

• E3: Bluetooth encryption algorithm

• EAP: Extensible authentication protocol. An authentication protocol standardized by the IETF organization.

• EAPOL EAP: encapsulation over LANs

• ECDH: Elliptic-curve Diffie-Hellman

• eSCO: Enhanced synchronous connection-oriented. A logical channel for transport of prioritized synchronous user data.

• FEC: Forward error correction. Another notion for an error correcting code.

• FH: Frequency hopping. Sending transmissions over a different carrier frequency at different times.

• FHSS: Frequency hop synchronization

• FHSS: Frequency Hopping Spread Spectrum. The FHSS carrier will hop on a predetermined, pseudo random pattern defined using a pool of 79 1MHz sub-channels defined across the entire band changing frequency about 1600 times per second. Each channel is used in 625 microseconds followed by a hop in a pseudo-random order to another channel. Bluetooth uses FHSS to solve interference problems with numerous other technologies that also operate in the 2.4GHz-2.4835GHz ISM frequency band.
- GAP: Generic access profile. A Bluetooth profile that determines common connection handling functions for all other Bluetooth profiles.
- GSM: Global Mobile System
- HC: Host controller
- HCI: Host controller interface
- HS: Headset
- IAC: Inquiry access code
- ICC: Integrated circuit card
- ID: Identifier
- IEEE: Institute of Electrical and Electronics Engineers. A nonprofit technical professional association for engineers in this area.
- IETF: Internet Engineering Task Force
- IIR: Infinite impulse response
- IKE: Internet key exchange. An IETF protocol used to authenticate IP connections and to exchange IPSEC keys.
- IP: Internet protocol.
- IPSEC: IP security protocol. An IETF security protocol used to protect IP packets.
- ISM: Industrial, scientific, and medical. A part of the radio spectrum reserved for these kinds of applications.
- L2CAP: Logical link communication and adaptation protocol.
- LAN: Local area network
- LAP: Lower address part. Bits 0 to 23 of the unique 48-bit IEEE device address BD_ADDR.
- LC: Link controller. Entity that implements the baseband protocol and procedures.
- LFSR: Linear feedback shift register
- LM: Link manager. Entity that sets up and maintains the Bluetooth link.
- LMP: Link manager protocol
- LSB: Least significant bit
- LT_ADDR: Logical transport address. A logical 3-bit address assigned to each slave in a piconet.
- MAC: Message authentication code. E1 is a MAC algorithm.
- MANA: Manual authentication
- MSB: Most significant bit
- NAcP: Network access point
- NAP: Nonsignificant address part. Bits 32 to 47 of the unique 48-bit IEEE device address.
- OpCode: Operation code. A code used to identify different types of PDUs.
- PAN: Personal area network
- PCD: Personal certification device
- PDA: Personal digital assistant
- PDU: Protocol data unit
- PIN: Personal identification number
- PKI: Public key infrastructure
- PSM: Protocol/service multiplexor. An identifier used by L2CAP during channel establishment to route the connection request to the right upper layer protocol. Several protocols can be multiplexed over L2CAP.
- QoS: Quality of service. Defines the specific requirements on the link (e.g., with respect to bit rate, delay, latency) needed by certain applications.
- RFCOMM: A serial cable emulation protocol based on ETSI TS 07.10
- RS-code: Reed-Solomon code.
- RSA: Rivest, Shamir, and Adleman. The name of a public-key cryptosystem for both encryption and authentication.
- SCO: Synchronous connection-oriented. A logical channel for transport of synchronous user data.
- SDP: Service discovery protocol. A protocol for locating services provided by or available through a Bluetooth device.
- SIG: Special Interest Group. The organization owning the Bluetooth trademark, also responsible for the evolution of Bluetooth wireless technology.
- SIM: Subscription identity module. An ICC used in the GSM mobile telephony system. The module stores subscription and user data.
- TLS: Transport layer security. An IETF security protocol used to authenticate peers, exchange keys, and protect TCP traffic.
- UAP: Upper address part. Bits 24 to 31 of the unique 48-bit IEEE device address.
• UART: Universal asynchronous receiver/transmitter. An integrated circuit used for serial communication with the transmitter and receiver clocked separately.

• USB: Universal serial bus

• WLAN: Wireless local area network

1.3 References


• http://www.tml.hut.fi/~candolin/Publications/BT/: A Security Evaluation of Bluetooth Profiles, Candolin


• http://www.pentest.co.uk/about.shtml: About Pentest


• http://th.informatik.uni-mannheim.de/people/Armknecht/TalkEWSCS.pdf: An Algebraic Attack on the Bluetooth Key Stream Generator


• http://www.benhui.net/bluetooth/btbrowser.html: Benhui.Net

• http://www.tdksystems.com/download/applications/bluealert.exe: BlueAlert Download

• http://trifinite.org/trifinite_stuff_bluebug.html: BlueBug

• http://www.bluejackq.com/bluesnarfing.asp: BluejackQ Bluesnarfing

• http://www.bluejackq.com/: bluejackQ.com - Home page

• http://trifinite.org/trifinite_stuff_blueprinting.html: BluePrinting

• http://news.zdnet.co.uk/communications/wireless/0,39020348,39146427,00.htm:
  Bluesnarfing tools 'spreading quickly' - ZDNet UK News

• http://bluesniff.shmoo.com/:
  Bluesniff - Proof of Concept Bluetooth Wardriver

• http://www.tomshardware.com/business/200408021/images/bluesniper.zip:
  BlueSniper Rifle short movie

• http://www.mulliner.org/palm/bluespam.php:
  BlueSpam

• http://studwww.ira.uka.de/~soden/18c3/index.html:
  Bluetooth - Table of Contents

• http://developer.axis.com/software/bluetooth/:
  Bluetooth @ developer.axis.com

• http://www.holtmann.org/linux/bluetooth/:
  Bluetooth and Linux

• http://www.telecomvergelijker.nl/thread/4562:
  Bluetooth attack N6310i - SET610 - TelecomVergelijker.nl

• http://www.y3kgroup.com/news_arc/171103.html:
  Bluetooth attack vector for mobile phones

• http://www.betaversion.net/btdsd/:
  bluetooth device security database btdsd

• http://www.pentest.co.uk/documents/bt_dongle_mod/bt_dongle_mod.html:
  Bluetooth dongle modification

  Bluetooth Hacking Step by Step Guide News

• http://news.mobile9.com/category/bluetooth/:
  Bluetooth Mobile9 News

  Bluetooth phone hacking tools spreading quickly News

• http://news.zdnet.co.uk/communications/wireless/0,39020348,39145881,00.htm:
  Bluetooth phones at risk from 'snarfing' - ZDNet UK News

• http://www.eweek.com/article2/0,1759,1591184,00.asp?kc=EWRSS03129TX1K0000605:
  Bluetooth Scare a Load of Hooey

  Bluetooth Stack introduction
• http://www.palowireless.com/infotooth/tutorial.asp:
  Bluetooth Tutorial - Start

• http://www.pentest.co.uk/documents/wicon_2004.pdf:
  Bluetooth Vulnerabilities Fact and Fiction WiCon 2004

• https://www.bluetooth.org/spec/:
  Bluetooth.org - The Official Bluetooth Membership Site

• http://www.bluez.org/:
  BlueZ - Official Linux Bluetooth protocol stack

• http://www.unrooted.net/hacking/bluez.html:
  BlueZ Bluetooth stack

• http://www.bookpool.com/.x/RRRRRR/sm/1580535046:
  Bookpool Bluetooth Security

• http://www.hpl.hp.com/personal/Jean_Tourrilhes/bt/\#stacks:
  Brainboxes BlueTooth hardware and Linux

• http://www.mulliner.org/palm/btclass.php:
  BTClass

• http://www.betaversion.net/btdsd/:
  btdsd bluetooth device security database

  article\&sid=177:
  Daily Wireless Wireless News, Research, Insight, and Analysis - ‘BlueSnarfing’
  Revisited

  article\&sid=116:
  Devices Vulnerable to ‘Bluesnarf’ attack

• http://www.wavewireless.com/classroom/whitepapers/FHSSvDSSS.
  pdf:
  Direct Sequence vs. Frequency Hopping

• http://www.e2xgear.com/:
  E2X Gear - Extract 2 Transmit

• http://www.extremetech.com/article2/0,1558,1643836,00.
  asp:
  Ericsson Halts Bluetooth Development

• http://news.zdnet.com/2100-1009_22-5197200.html:
  Expert Gaps still pain Bluetooth security Tech News on ZDNet

  html:
  Flaws in Digital Wireless Technology Said to Allow Eavesdropping
• http://www.mediaplace.nl/forum/about2381-0.html: Free Nokia Cellular Phones Bluetooth hack


• http://www.phonearena.com/htmls/readarticles.php?id=59: Hackers can attack Bluetooth devices (phoneArena)


• http://www.betaversion.net/btdsd/download/: Index of -btdsd-download

• http://www.securiteam.com/securityreviews/5OP050UC0U.html: Man-In-The-Middle Attack Using Bluetooth In A WLAN Interworking Environment

• http://www.net-cell.com/MP/dl.html: MeetingPoint - Downloads


• http://www.pocketpc-club.nl/content.php?id=3429: Nederlandse Pocket PC-club


• http://news.zdnet.co.uk/communications/wireless/0,39020348,39145886,00.htm: Nokia admits multiple Bluetooth security holes - ZDNet UK News

• http://news.zdnet.co.uk/internet/security/0,39020375,39161309,00.htm: Nokia releases 'invisible' Bluetooth security fix - ZDNet UK News

• http://www.betaversion.net/btdsd/db/Nokia6310i.txt: Nokia6310i.txt
● http://www.palowireless.com/: palowireless Wireless Resource Center


● http://www.eweek.com/article2/0,1759,1542546,00.asp: Pulling for Bluetooth


● http://www.computerworld.com/mobiletopics/mobile/story/0,10801,89495,00.html: Securing your Bluetooth devices - Computerworld


● http://renegade.w3xs.com/droll.htm: SMan Renegade’s Ericsson Mobile Phone Site

● http://www.betaversion.net/bt/dsd/db/SonyEricsssonT610.txt: SonyEricsssonT610.txt


● http://www.computerworld.com/mobiletopics/mobile/story/0,10801,74152,00.html?from=story_picks: Study Bluetooth security should raise red flags - Computerworld


● http://www.eweek.com/article2/0,1759,1530755,00.asp: The Death of Bluetooth Intel Moves to Ultrawideband

● http://compnetworking.about.com/b/a/092578.htm: The first mobile phone worm - Cabir

● http://edu.lut.fi/LutPub/web/nbnfi-fe20011206.pdf: The role of Bluetooth technology in transforming the value network ...

● http://www.eweek.com/article2/0,1759,1614794,00.asp: This Time, Cell Phone Virus Is for Real

- http://trifinite.org/blog:
  trifinite.blog - the weblog of the trifinite.group
- http://trifinite.org/trifinite_stuff.html:
  trifinite.org - the home of the trifinite.group
- http://www.unrooted.net/hacking/bluez-rfcomm-obex.html:
  UNrooted.net Hacker Area
- http://www.viruslist.com/eng/index.html?tnews=461485\&id=2025573:
  Viruses, anti-viruses, events, facts and commentary
- http://www.lucretia.ca/portal/modules.php?name=News\&file=article\&sid=64:
  Vulnerability in Bluetooth Explained
  War Nibbling Bluetooth Insecurity
- http://searchstorage.techtarget.com/gDefinition/0,294236,sid5_gci952393,00.html:
  What is bluesnarfing - a definition from the leading computer technical encyclopedia, dictionary and glossary
- http://searchnetworking.techtarget.com/sDefinition/0,,sid7_gci525695,00.html:
  What is frequency-hopping spread spectrum - a definition from the leading computer technical encyclopedia, dictionary and glossary
- http://66.45.42.84/Support/index.asp:
  WIDCOMM Bluetooth Connectivity Solutions
- http://developers.sun.com/techtopics/mobility/midp/articles/bluetooth1/:
  Wireless Application Programming with J2ME and Bluetooth
  Wireless Handheld Security Articles - Bluefire Security Technologies
- http://www.xonio.com/features/feature_unterseite_11876822.html:
  Xonio - 23 Handys im Hacker-Test - Denial of Service-Attacke Absturz mitten im Gesprach
- http://news.zdnet.co.uk/internet/security/0,39020375,39162400,00.htm:
  Year-old Bluetooth vulnerability invites mobile worm - ZDNet UK News
- http://www.stanford.edu/%7Ejayasena/ee272/proj_desc.htm:
  Implementation details of the data (payload) encryption algorithm specified in the Bluetooth specification


• Dipl.-Ing.(FH) Herfurt M., *Bluesnarfing @ CeBIT 2004 ‘Detecting and Attacking bluetooth-enabled Cellphones at the Hannover Fairground’*, Salzburg Research Forschungsgesellschaft mbH, Austria


• Sun J., Howie D., Koivisto A., And Sauvola J., *Design, implementation, and evaluation of Bluetooth security*, University of Oulu, Finland


• Jakobsson, Markus and Wetzel, Sussane, *Security Weaknesses in Bluetooth*, Lucent Technologies - Bell Labs,

• Bundesamt fr Sicherheit in der Informationstechnik, *Bluetooth Gefährdungen und Sicherheitsmaßnahmen*

• Senese B., *Implementing Bluetooth Wireless Technology In An Embedded Device*, Extended Systems Inc.


• Vittitoe S., *Bluetooth Security*, SANS Institute, March 2004

• Whitehouse O., *War Nibbling: Bluetooth Insecurity*, @stake, October 2003

• Miller M., *Discovering Bluetooth*, Sybex, 2001

• Whitehouse O., *CanSecWest core04 Vancouver*, @stake, April 2004


• http://th.informatik.uni-mannheim.de/People/Lucks/papers/e0.ps.gz
  Analysis of the E0 Encryption System

• http://www.informatics.indiana.edu/markus/papers/bluetooth.ps
  Security Weaknesses in Bluetooth

• http://www.jonsguides.com/bluetooth/index.html
  Jon’s Bluetooth Guide
2 Bluetooth overview

Bluetooth is a standard developed by a group of electronics manufacturers that allows any sort of electronic equipment - from computers and cell phones to keyboards and headphones - to make its own connections, without wires, cables or any direct action from a user. Bluetooth is intended to be a standard that works at two levels:

- It provides agreement at the physical level (radio-frequency standard).
- It also provides agreement at the next level up, where products have to agree on when bits are sent, how many will be sent at a time and how the parties in a conversation can be sure that the message received is the same as the message sent.

The companies belonging to the Bluetooth Special Interest Group, and there are more than 1,000 of them, want to let Bluetooth’s radio communications take the place of wires for connecting peripherals, telephones and computers.

2.1 Bluetooth specifications

Here are some specification details:

- The devices in a piconet share a common communication data channel. The channel has a total capacity of 1 megabit per second (Mbps). Headers and hand-shaking information consume about 20 percent of this capacity.

- In the United States and Europe, the frequency range is 2,400 to 2,483.5 MHz, with 79 1-MHz radio frequency (RF) channels. In practice, the range is 2,402 MHz to 2,480 MHz. In Japan, the frequency range is 2,472 to 2,497 MHz with 23 1-MHz RF channels.

- A data channel hops randomly 1,600 times per second between the 79 (or 23) RF channels.

- Each channel is divided into time slots 625 microseconds long.
• A piconet has a master and up to seven slaves. The master transmits in even time slots, slaves in odd time slots.

• Packets can be up to five time slots wide.

• Data in a packet can be up to 2,745 bits in length.

• There are currently two types of data transfer between devices: SCO (synchronous connection oriented) and ACL (asynchronous connectionless).

• In a piconet, there can be up to three SCO links of 64,000 bits per second each. To avoid timing and collision problems, the SCO links use reserved slots set up by the master.

• Masters can support up to three SCO links with one, two or three slaves.

• Slots not reserved for SCO links can be used for ACL links.

• One master and slave can have a single ACL link.

• ACL is either point-to-point (master to one slave) or broadcast to all the slaves.

• ACL slaves can only transmit when requested by the master.

• The official Bluetooth logo:

![Bluetooth logo]

### 3 Overall security description

Risks are inherent to any wireless technology. Some of these risks are similar to those of wired networks; some are exacerbated by wireless connectivity; some are new. Perhaps the most significant source of risks in wireless networks is that the technology’s underlying communications medium, the airwave, is open to intruders, making it the logical equivalent of an Ethernet port in the parking lot.

Specific threats and vulnerabilities to wireless networks and handheld devices include the following:

• All the vulnerabilities that exist in a conventional wired network apply to wireless technologies.

• Malicious entities may gain unauthorized access to an agency’s computer network through wireless connections, bypassing any firewall protections.

• Sensitive information that is not encrypted (or that is encrypted with poor cryptographic techniques) and that is transmitted between two wireless devices may be intercepted and disclosed.
DoS attacks may be directed at wireless connections or devices.

- Malicious entities may steal the identity of legitimate users and masquerade as them on internal or external corporate networks.
- Sensitive data may be corrupted during improper synchronization.
- Malicious entities may be able to violate the privacy of legitimate users and be able to track their movements.
- Malicious entities may deploy unauthorized equipment (e.g., client devices and access points) to surreptitiously gain access to sensitive information.
- Handheld devices are easily stolen and can reveal sensitive information.
- Data may be extracted without detection from improperly configured devices.
- Viruses or other malicious code may corrupt data on a wireless device and subsequently be introduced to a wired network connection.
- Malicious entities may, through wireless connections, connect to other agencies or organizations for the purposes of launching attacks and concealing their activities.
- Interlopers, from inside or out, may be able to gain connectivity to network management controls and thereby disable or disrupt operations.
- Malicious entities may use third-party, untrusted wireless network services to gain access to an agency’s or other organization’s network resources.
- Internal attacks may be possible via ad hoc transmissions.

4 Bluetooth security overview

Bluetooth has three different modes of security. Each Bluetooth device can operate in one mode only at a particular time. The three modes are the following:

- Security Mode 1: Nonsecure mode
- Security Mode 2: Service-level enforced security mode
- Security Mode 3: Link-level enforced security mode

4.1 Security Mode 1: Nonsecure mode

A device will not initiate any security procedures. In this nonsecure mode, the security functionality (authentication and encryption) is completely bypassed. In effect, the Bluetooth device in Mode 1 is in a promiscuous mode that allows other Bluetooth devices to connect to it. This mode is provided for applications for which security is not required, such as exchanging business cards.
4.2 Security Mode 2: Service-level enforced security mode

In the service-level security mode, security procedures are initiated after channel establishment at the Logical Link Control and Adaptation Protocol (L2CAP) level. L2CAP resides in the data link layer and provides connection-oriented and connectionless data services to upper layers. For this security mode, a security manager (as specified in the Bluetooth architecture) controls access to services and to devices. The centralized security manager maintains policies for access control and interfaces with other protocols and device users. Varying security policies and trust levels to restrict access may be defined for applications with different security requirements operating in parallel. Therefore, it is possible to grant access to some services without providing access to other services. Obviously, in this mode, the notion of authorization – that is the process of deciding if device A is allowed to have access to service X – is introduced.

4.3 Security Mode 3: Link-level enforced security mode

In the link-level security mode, a Bluetooth device initiates security procedures before the channel is established. This is a built-in security mechanism, and it is not aware of any application layer security that may exist. This mode supports authentication (unidirectional or mutual) and encryption. These features are based on a secret link key that is shared by a pair of devices. To generate this key, a pairing procedure is used when the two devices communicate for the first time.

4.4 Bluetooth Key Generation from PIN

The link key is generated during an initialization phase, while two Bluetooth devices that are communicating are "associated" or "bonded." Per the Bluetooth specification, two associated devices simultaneously derive link keys during the initialization phase when a user enters an identical PIN into both devices. The PIN entry, device association, and key derivation are depicted conceptually in Figure 1. After initialization is complete, devices automatically and transparently authenticate and perform encryption of the link. It is possible to create a link key using higher layer key exchange methods and then import the link key into the Bluetooth modules. The PIN code used in Bluetooth devices can vary between 1 and 16 bytes. The typical 4-digit PIN may be sufficient for some applications; however, longer codes may be necessary.
4.5 Bluetooth Authentication

The Bluetooth authentication procedure is in the form of a “challenge-response” scheme. Two devices interacting in an authentication procedure are referred to as the claimant and the verifier. The verifier is the Bluetooth device validating the identity of another device. The claimant is the device attempting to prove its identity. The challenge-response protocol validates devices by verifying the knowledge of a secret key (a Bluetooth link key). The challenge-response verification scheme is depicted conceptually in Figure 2. As shown, one of the Bluetooth devices (the claimant) attempts to reach and connect to the other (the verifier).

The steps in the authentication process are the following:

1. The claimant transmits its 48-bit address (BD_ADDR) to the verifier.
2. The verifier transmits a 128-bit random challenge (AU_RAND) to the claimant.
3. The verifier uses the E1 algorithm to compute an authentication response using the address, link key, and random challenge as inputs. The claimant performs the same computation.
4. The claimant returns the computed response, SRES, to the verifier.
5. The verifier compares the SRES from the claimant with the SRES that it computes.
6. If the two 32-bit SRES values are equal, the verifier will continue connection establishment.

**Figure 2: Bluetooth Authentication**


### 4.6 Bluetooth Encryption Process

The Bluetooth specification also allows three different encryption modes to support the confidentiality service:

- **Encryption Mode 1**: No encryption is performed on any traffic.

- **Encryption Mode 2**: Broadcast traffic goes unprotected (not encrypted), but individually addressed traffic is encrypted according to the individual link keys.

- **Encryption Mode 3**: All traffic is encrypted according to the master link key.
4.7 Problems with the Bluetooth Standard Security

- Strength of the challenge-response pseudorandom generator is not known:
  The Random Number Generator (RNG) may produce static number or periodic numbers that may reduce the effectiveness of the authentication scheme.

- Short PINS are allowed:
  Weak PINs, which are used for the generation of link and encryption keys, can be easily guessed. Increasing the PIN length in general increases the security. People have a tendency to select short PINs.

- An elegant way to generate and distribute PINs does not exist:
  Establishing PINs in large Bluetooth networks with many users may be difficult. Scalability problems frequently yield security problems.

- Encryption key length is negotiable:
  The Bluetooth SIG needs to develop a more robust initialization key generation procedure.

- Unit key is reusable and becomes public once used:
  A unit key is a link key that one unit generates by itself and uses as a link key.
with any other device. Unit keys can only be safely used when there is full trust among the devices that are paired with the same unit key. This is because every paired device can impersonate any other device holding the same unit key. Since Bluetooth version 1.2, the use of unit keys is not recommended. But, for legacy reasons, unit keys have not been completely removed from the specification.

- The master key is shared:
The Bluetooth SIG needs to develop a better broadcast keying scheme.

- No user authentication exists:
Device authentication only is provided. Application level security and user authentication can be employed.

- Attempts for authentication are repeated:
The Bluetooth SIG needs to develop a limit feature to prevent unlimited requests. The Bluetooth specification requires a time-out period between repeated attempts that will increase exponentially.

- E0 stream cipher algorithm is weak:
The stream cipher E0 has its roots in the so-called summation combiner stream cipher. This was a stream cipher that was proposed by Massey and Rueppel in the mid-1980s. The most powerful attacks on this type of stream ciphers are the correlation attacks in combination with exhaustive search over a limited key space (this is sometimes also referred to as initial guessing). Recent cryptanalysis shows that the E0 cipher is weaker than this.

- Key length is negotiable:
A global agreement must be established on minimum key length.

- Unit key sharing can lead to eavesdropping:
A corrupt user may be able to compromise the security between (gain unauthorized access to) two other users if that corrupt user has communicated with either of the other two users. This is because the link key (unit key), derived from shared information, is disclosed.

- Privacy may be compromised if the Bluetooth device address (BD_ADDR) is captured and associated with a particular user:
Once the BD_ADDR is associated with a particular user, that user’s activities could be logged, resulting in a loss of privacy.

- Device authentication is simple shared-key challenge-response:
One-way-only challenge-response authentication is subject to man-in-the-middle attacks. Mutual authentication is required to provide verification that users and the network are legitimate.

- End-to-end security is not performed:
Only individual links are encrypted and authenticated. Data is decrypted at intermediate points. Applications software above the Bluetooth software can be developed.

- Security services are limited:
Audit, nonrepudiation, and other services do not exist. If needed, these can be developed at particular points in a Bluetooth network.
5 Bluetooth security attacks

5.1 Impersonation attack by inserting/replacing data

When no encryption is activated, this can easily be achieved by correctly setting the CRC check data in the payload after the data in the payload has been changed. When ciphering is activated, the attacker can compute how to modify the CRC to make it agree with modifications in the encrypted data bits. In a practical system were encryption is activated, it is not at all easy to make something useful of this attack beyond the point of just disrupting the communication. The attacker must somehow know the context of the payload data to conduct changes that are meaningful or effective.

5.2 Bluejacking

Although known to the technical community and early adopters for some time, the process now known as “Bluejacking” has recently come to the fore in the consumer arena, and is becoming a popular mechanism for exchanging anonymous messages in public places. The technique involves abusing the Bluetooth “pairing” protocol, the system by which Bluetooth devices authenticate each other, to pass a message during the initial “handshake” phase. This is possible because the “name” of the initiating Bluetooth device is displayed on the target device as part of the handshake exchange, and, as the protocol allows a large user defined name field - up to 248 characters - the field itself can be used to pass the message. This is all well and good, and, on the face of it, fairly harmless, but, unfortunately, there is a down side. There is a potential security problem with this, and the more the practice grows and is accepted by the user community, and leveraged as a marketing tool by the vendors, the worse it will get. The problem lies in the fact that the protocol being abused is designed for information exchange. The ability to interface with other devices and exchange, update and synchronize data, is the reason of existence of Bluetooth. The Bluejacking technique is using the first part of a process that allows that exchange to take place, and is therefore open to further abuse if the handshake completes and the “bluejacker” successfully pairs with the target device. If such an event occurs, then all data on the target device becomes available to the initiator, including such things as phone books, calendars, pictures and text messages. As the current wave of PDA and telephony integration progresses, the volume and quality of such data will increase with the devices’ capabilities, leading to far more serious potential compromise. Given the furore that erupted when a second-hand Blackberry PDA was sold without the previous owner’s data having been wiped[3], it is alarming to think of the consequences of a single bluejacker gathering an entire corporate staff’s contact details by simply attending a conference or camping outside their building or in their foyer with a Bluetooth capable device and evil intent. Of course, corporates are not the only potential targets - a Bluejacking expedition to, say, The House of Commons, or The US Senate, could provide some interesting, valuable and, who’s to say, potentially damaging or compromising data.
This is also called and OBEX Push Attack: OBEX allows you to PUSH items anonymously in some cases between devices.

Impact:

- Annoying, no real security impact
- Possible extensions to this idea is around sending vCard’s with common names such as ‘Home’ or ‘Work’ in an attempt to overwrite an existing phone book entry in the recipients cell/smart phone

5.3 Bluetooth Wardriving

Map the physical whereabouts of users carrying Bluetooth-enabled devices. Since each Bluetooth device is freely broadcasts its unique 48-bit address, it is possible to track the user movements.

To protect a device against location tracking, an anonymity mode is needed. Devices operating in anonymous mode regularly update their device address by randomly choosing a new one.

Different types of location tracking attacks are possible:

- **Inquiry attack** The attack distributes one or more Bluetooth devices throughout a region to locate Bluetooth users. If the potential victim of such an attack has left his device in discoverable mode, attacking device can simply interrogate the area using frequent inquiry messages for devices and maintain a log of all the device addresses that are discovered.

- **Traffic monitoring attack** This attack succeeds even if the victim device is not in discoverable mode. The attacker simply monitors the communication between two trusted devices belonging to the victim. These devices will communicate using a specific CAC. This CAC is computed from the device address of the master device in the piconet.

  Furthermore, the whole device address is sent in the FHS packets of the devices, allowing an attacker to uniquely determine the identity of a device. But the FHS packets are only used at connection establishment.

- **Pagin attack** This attack allows the attacker to determine if a given device with a known BD_ADDR or DAC is present within range. The attack requires that the victim device is connectable. The attacking device pages the target device, waits for the ID packet to be returned, and then does not respond. If an ID is returned, then the attacker knows that the victim device is present. The target device, waiting for the response, will just time out and the incident will not be reported to the application layer.

- **Frequency hopping attack** The frequency hopping scheme in Bluetooth is determined by a repeating hopping sequence. The hopping scheme is calculated from different input parameters, such as an address and the master clock. In the connection state, the LAP and the four least significant bits in the UAP of the master device are used. In the page state, the LAP/UAP of the paged unit is used. Thus, it is (at least theoretically) possible to get information of the LAP and four bits in the UAP based on the observed hopping scheme.

- **User-friendly name attack** A Bluetooth device can request the user-friendly name anytime after a successful baseband paging procedure. The name request command can be used to mount a location tracking attack.
5.4 Nokia 6310i Bluetooth OBEX Message DoS

Nokia 6310i contains a flaw that may allow a remote denial of service. The issue is triggered when invalid Bluetooth OBEX messages are sent by an attacker, and will result in loss of availability for the phone.

Impact:

- Small, since only the phone will be shut-down without loss of data

5.5 Brute-Force attack

Brute-force attack on the BD_ADDR (MAC address) of a device while not in discoverable mode. Some manufacturer’s claim this would take an unreasonable amount of time (eg, 11 hours). However, a multi-threaded version of @stake’s RedFang could simultaneously utilize up to 8 USB Bluetooth devices which would reduce the 11hrs to approximately 90 minutes (based on one vendor’s range).

Impact:

- Can take a long time before the correct BD_ADDR is discovered
- Once the BD_ADDR is discovered, a Bluesnarf attack could be set up, while the user thinks he/she is safe because the device is set to hidden mode

5.6 Denial-of-Service attack on the device

When the Bluetooth authentication fails, a certain amount of time must elapse before the verifier will initiate a new attempt to the same claimant and before the claimant sends a response to an authentication attempt by a unit using the same identity as the unit that notified an authentication failure. For each additional authentication failure, the waiting interval should be exponentially increased until a certain maximum value is obtained. The attacker prevents or prohibits the normal use or management of communications facilities. The resulting system degradation can, for example, be the result of the system being fully occupied by handling bogus connection requests. If the attacker simulates a trustable device during these DoS, making the system decline trustable devices.

5.7 Disclosure of keys

- A Bluetooth device attached to the computer may be exchanged for a false one, whose only purpose is to ‘suck’ out link keys from the host.
- A rightful USB plug or PCMCIA card may be removed from the owners computer and inserted into a corresponding slot of the adversarys computer. On this computer, one or more keys stored on the Bluetooth controller can be read out. Once the list of keys has been read out, the USB plug (or card) is returned to its proper owner, who may be completely unaware.
- Malicious software

A Trojan horse disguised as something quite innocent can send the key database to some place where the adversary can access it. If this malicious code is distributed through a virus or worm, the attack can quickly spread to a large number of computers.
Once the link key of a computer and phone (and the BD_ADDR of the computer) is known, the adversary can silently connect to the mobile phone, impersonate the computer, and make use of any service the phone offers over Bluetooth.

5.8 Unit key attacks

A unit that uses a unit key is only able to use one key for all its secure connections. Hence, it has to share this key with all other units that it trusts. Consequently, a trusted device (a device that possesses the unit key) that eavesdrops on the initial authentication messages between two other units that utilize the unit key will be able to eavesdrop on any traffic between these two units. The unit will be able to impersonate the unit distributing the unit key.

The potential risks with units keys have also been recognized by the Bluetooth SIG. Originally, the unit key was introduced in order to reduce memory requirements on very limited devices and remains part of the standard for backward compatibility reasons.

5.9 Backdoor attack

The Backdoor attack involves establishing a trust relationship through the "pairing" mechanism, but ensuring that it no longer appears in the target's register of paired devices. In this way, unless the owner is actually observing their device at the precise moment a connection is established, they are unlikely to notice anything untoward, and the attacker may be free to continue to use any resource that a trusted relationship with that device grants access to. This means that not only data can be retrieved from the phone, but other services, such as modems or Internet, WAP and GPRS gateways may be accessed without the owner’s knowledge or consent. Once the Backdoor is installed, the Bluesnarf attack will function on devices that previously denied access, and without the restrictions of a plain Bluesnarf attack.

5.10 Pairing attack

The Bluetooth 1.1 specification is sensitive to passive and active attacks on the pairing procedure. The attacks only work if the attacker is present at the pairing occasion, which typically only occurs once between one pair of devices. If pairing is performed in public places during a connection to an access point, point-of-sale machine, or printer, this can be a dangerous threat.

5.11 BlueStumbling = BlueSnarfing

It is possible, on some makes of device, to connect to the device without alerting the owner of the target device of the request, and gain access to restricted portions of the stored data therein, including the entire phonebook (and any images or other data associated with the entries), calendar, realtime clock, business card, properties, change log, IMEI (International Mobile Equipment Identity, which uniquely identifies the phone to the mobile network, and is used in illegal phone 'cloning'). This is normally only possible if the device is in "discoverable" or "visible" mode, but there are tools available on the Internet that allow even this safety net to be bypassed. They refuse to say how the attacks actually works, but presumably it exploits a flaw whereby a default 'pairing' password (probably only four characters) is guessed and the handset owner has left the device with Bluetooth switched on and visibility set to
Also called an OBEX Pull Attack: OBEX allows you to PULL items anonymously in some cases between devices.

Impact:

- A number of Nokia, Ericsson & Sony Ericsson handsets are susceptible, so many popular phones are vulnerable for this attack
- Very much dependent on vendor’s implementation of OBEX/Bluetooth stack
- Information obtainable can include calendar, real time clock, business card, properties, change log, IMEI
- CeBIT paper showed that a lot of devices are vulnerable

5.12 BlueBug attack

The BlueBug attack creates a serial profile connection to the device, thereby giving full access to the AT command set, which can then be exploited using standard off the shell tools, such as PPP for networking and gnokii for messaging, contact management, diverts and initiating calls. With this facility, it is possible to use the phone to initiate calls to premium rate numbers, send sms messages, read sms messages, connect to data services such as the Internet, and even monitor conversations in the vicinity of the phone. This latter is done via a voice call over the GSM network, so the listening post can be anywhere in the world. Bluetooth access is only required for a few seconds in order to set up the call. Call forwarding diverts can be set up, allowing the owner’s incoming calls to be intercepted, either to provide a channel for calls to more expensive destinations, or for identity theft by impersonation of the victim.

Bluesnarf attack does allow the unauthorized downloading of items via the OBEX protocol, while the loophole identified in BlueBug allows to control the device via a plain serial connection.

5.13 PSM Scanning

Works on the idea that not all PSM (Protocol/Service Multiplexer) ports are registered with the local SDP (Service Discovery Protocol). So if we bypass the SDP database and try and connect to PSM’s sequentially we may locate hidden functionality

Impact:

- No PSM’s found to-date that offer other than advertised services
- Idea could be used to create a ‘knock’ style backdoor for Bluetooth devices

5.14 Off-line PIN (via Kinit) recovery

Sniff* the initial ’RAND’ transfer between two devices which occurs in clear-text (effectively the first stage of the bond)
Sniff the XOR’d ’RAND’(s) used for LinkKey generation
Sniff the AUTH RAND and AUTH SRES which both occur in clear-text (the last stage of the bond)
Do some number crunching and have enough data in order to be able to recover the
PIN, LinkKey and all inputs used for both
Needed to sync the frequency hopping or capture entire 2.4ghz spectrum and do off-line

5.15 On-line PIN cracking
Attack possible if fixed PIN exists in device (i.e. same PIN is used for every connecting device)
Need to change the Bluetooth address each time and try different PINs
Will bypass the ever increasing delay between retries counter measure
The specifications do not provide solution to this problem

5.16 Off-line encryption key (via Kc)
Extends on from the Kinit recovery attack
Very similar method as 2 of 3 needed seeds are known (i.e. master clock and Kc), simply sniff the EN_RAND in addition

5.17 Attack on the Bluetooth Key Stream Generator
Break the security of the cipher. Algebraic attack on the Linear Feedback Shift Register
Work effort circa $2^{67,58}$ operations.

5.18 Reflection Attack
A hacker can capture the MIN and ESN and pretend to be someone else
Stealing the Unit Key
Highlights weakness of only authenticating the device and not the user

5.19 Replay attacks
A hacker can record Bluetooth transmissions in all 79 frequencies and then in some way figure out frequency hopping sequence and then replay the whole transmission.

5.20 Man-in-the-middle attack
Intervention of traffic during pairing
Bluetooth authentication does not use public key certificates to authenticate users.

5.21 Denial-of-Service attack on the Bluetooth network
Not very feasible would require the jamming of the whole ISM band

5.22 A man-in-the-middle attack using Bluetooth in a WLAN interworking environment
A man-in-the-middle attack may be possible on the Bluetooth link in a WLAN interworking environment. The attacker lures the victim to connect to a malicious WLAN access point. The attack does not require to know the Bluetooth link key. The attacker can repeat this attack on the same victim many times in any WLAN network.
5.23 Impersonate original sending/receiving unit

This attack requires the attacker to provide the correct response on the authentication challenge of a correspondent. Currently, no attack on the SAFER+-based E1 authentication function is known that achieves this within any realistic computational effort.

5.24 Correlation attacks

References:


The complexity of the attacks by Courtois is $O(2^{49})$. The attack requires $2^{23.4}$ operations can be performed in about 35 hours, however, the result by Courtois shows that the core in E0 is not cryptographically strong.

There are two possibilities to obtain an actual attack on E0 to recover $K'_c$:

1. make the algebraic attack work with only 2,744 output bits (max number of known plaintext bits encrypted with the same $K_p$)

2. find a way to utilize that there exists a relation between the consecutive blocks of 2,744 output bits. This is a result of the fact that the output blocks are generated with the same constraint key $K'_c$, $BD_{ADDR}$ and $RAND$ values, but different clock timer values. This is not infeasible, because the relation between the initial state values (only differ in clock timer values) satisfies a linear relation over the finite field $GF(2)$. It is possible to rewrite the relations in terms of a specific initial state, the clock bits and the output bits.

Both possibilities have not been done currently. No complexity estimates for such attacks are known.