

available at www.sciencedirect.comjournal homepage: www.elsevier.com/locate/diin
**Digital
Investigation**

Digital detective – Bluetooth

Adam Laurie

The Bunker Secure Hosting

ARTICLE INFO

Article history:

Received 16 January 2006

Revised 24 January 2006

Accepted 24 January 2006

Keywords:

Bluetooth

BlueSnarf

BlueBug

Mobile Phone Hacking

ABSTRACT

In November 2005, the Bluetooth SIG announced that shipments of Bluetooth enabled devices had passed the 9.5 million units per week threshold. Given these volumes, it is not surprising that some serious security issues have emerged, albeit in a relatively small number of models. However, when you are dealing with numbers as huge as this, even “a relatively small number of models” can represent a large number of actual devices at risk, and, should those issues affect the integrity of data on mobile phone handsets, could be statistically significant enough to pose a serious threat to the validity of forensic evidence gathered from them. This paper looks at the currently known Bluetooth security issues, and how they could potentially impact on a forensic examination.

© 2006 Elsevier Ltd. All rights reserved.

In November 2003, serious flaws in the Bluetooth security mechanisms built in some popular models of mobile phone were discovered. These flaws allowed unauthorised access to personal information stored on the handsets, such as the contents of the phonebook, calendar, and some other technical information including the IMEI (International Mobile Equipment Identity – the unique number used to individually identify phones). These flaws were dubbed “BlueSnarfing” (<http://www.thebunker.net/security/bluetooth.htm>). Shortly afterwards, further problems were found which extended the range of data that could be read to include SMS text messages, and, in some cases, the entirety of the storage areas on the phone, including removable memory cards. In many cases, it was also possible to take control of the phone via the built-in modem, allowing calls to be placed, SMS messages to be sent, and even turning it into a ‘bug’ or tracking device. In order to understand how all this could be possible, it is worth taking some time to look at how Bluetooth works, and what it is designed for.

Bluetooth is a “wire replacement technology” – its purpose is to provide short-range connectivity for low-bandwidth connections, where one would normally have a short cable, by using low power radio waves instead. A good example of a Bluetooth application would be a headset for a mobile

phone; instead of having a wire dangling from your ear to the phone, the Bluetooth radio provides a connection between the phone and headset, even if the phone is safely tucked away in your pocket or handbag. As well as headsets, it is now becoming common to find Bluetooth in digital cameras, PDAs, printers, MP3 players, etc.

Its principals are very similar to the now familiar wireless networking “Wi-Fi” technology, but, although they share the same frequency spectrum (2.4 GHz), it differs in some important technical details. Unlike Wi-Fi, where all devices transmit on the same shared frequencies, Bluetooth operates in what they call a ‘piconet’, in which all member devices constantly change frequency, in a coordinated manner, allowing them to talk to each other without interfering with, or being visible to, other piconets in the area. This is called “Frequency Hopping”, and in the case of Bluetooth, the frequency change (“Hop”), occurs 16,000 times per second, over 79 frequencies (“Channels”). The hopping sequence is determined by the “Master” device in the piconet (usually the one that initiates the connection), and is followed by up to seven “Slaves”, thereby giving a maximum piconet size of eight active devices, although a Slave can be a member of more than one Piconet, and this is known as a “Scatternet” (see Fig. 1). This feature makes Bluetooth much harder to eavesdrop than Wi-Fi, as it

E-mail address: adam.laurie@thebunker.net

URL: <http://www.thebunker.net>

1742-2876/\$ – see front matter © 2006 Elsevier Ltd. All rights reserved.

doi:10.1016/j.diin.2006.01.011

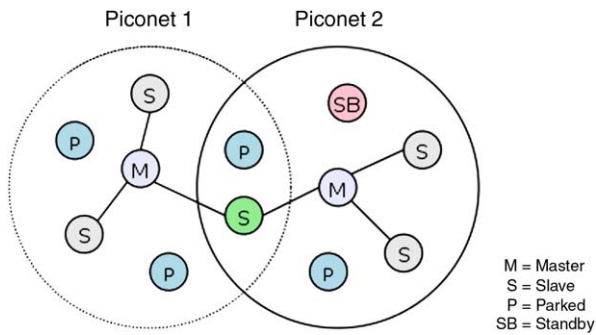


Fig. 1 – Bluetooth Piconet/Scatternet topology.

is not possible to simply monitor one frequency and capture all the traffic – in order to follow a specific Bluetooth “conversation”, one would have to effectively become a member of the piconet and follow the same hopping sequence.

Bluetooth currently comes in three “classes”, determined by their radio transmitter’s power, which, for descriptive purposes, is normally translated into a nominal “range” value, the lowest being a “Class 3” device, with a nominal range of 10 m. “Class 2” and “Class 1” devices are 40 and 100 m, respectively. With a specialist antenna, it has been shown that this range can be extended to over a mile in ideal conditions (http://trifninite.org/trifninite_stuff_bluetooone.html).

Operationally, Bluetooth can be thought of simply as a conduit through which services are provided, so, for example, if a device has a file transfer or “ftp” facility, normally activated by placing it in a cradle attached to a PC, this could also be accessed via Bluetooth without having to use the cradle. Now this is where the security issues start to come to light – in the original scenario where a device had to be placed in a cradle, there was an implicit “trust” between the two devices: since the device had to be physically in the possession of the owner of the cradle, it could be reasonably assumed that they were authorised to connect to it, and no further security was really deemed necessary. In the case of Bluetooth, since the connection could potentially take place without the device having left the owner’s pocket, a layer of authorisation and/or authentication needed to be put in place, even if only to prevent the accidental connection of an unauthorised or unwanted device. To achieve this, Bluetooth uses the concept of “pairing”, in which a PIN number is exchanged between the two devices, allowing verification that the connection is both authorised and desired at the time. Once matching PINs have been exchanged, they can be (and usually are) used to generate cryptographic “Link Keys” which are used to encrypt all further communications, as well as provide authentication for future connections without the owner having to manually enter any further PIN codes. There are some known issues with the cryptography in the Bluetooth standard, most notably that if the initial key exchange can be witnessed that it is possible to reverse engineer the PIN and Link Keys and thereby be in a position to eavesdrop future conversations (<http://www.cansecwest.com/csw04/csw04-Whitehouse.pdf> and <http://www.eng.tau.ac.il/~yash/shaked-wool-mobisys05/>). However, even in the unlikely event that a key exchange were witnessed, as noted earlier, the

practical difficulties of intercepting the radio traffic make this attack not feasible in the majority of cases.

So, in theory, this is all well and good, and should have covered all the bases, but in practice, poor implementation has often meant that the pairing process could be spoofed or bypassed altogether, giving immediate and unauthorised access to devices up to a mile away. The implications for the world of forensics are potentially quite severe: how reliable is evidence gathered from a device that may have been compromised by a third party? It is beyond the scope of this paper to discuss the legalities, but we can look at the specifics of what could have been done to the device in each case, and, more importantly, the data on it. Although there are now a wide variety of attacks against Bluetooth devices, the main ones likely to be relevant to the Forensic investigator are “BlueSnarf”, “BlueSnarf++” and “BlueBug”, and it is these that we will look at in detail.

1. BlueSnarf

Description: reading of device data via “OBEX PUSH”.

Reference: http://trifninite.org/trifninite_stuff_bluesnarf.html.

This is a passive attack that reads data from the device, bypassing the authentication process by connecting to the OBEX PUSH service, and then performing a “PULL” instead. Since OBEX follows standards laid down by the Infrared Data Association (IrDA), the file naming conventions specified in the IrMC (Infrared Mobile Communication protocol – <http://www.irda.org/displaycommon.cfm?an=1&subarticlenbr=7>) can be used to request specific objects such as, for example, the phonebook, which will map to ‘telecom/pb.vcf’. The contents of the device itself are left untouched, and so a device compromised in this manner can be considered uncontaminated. Indeed, there may be cases where collection of data by this method would be preferable to a more conventional approach, as it can be guaranteed that the phone will not change its state or log any unusual activity (although, having said that, care should be taken to only perform this operation on devices known to be vulnerable, as in other cases unexpected OBEX commands have been known to cause devices to crash or reset).

Bluesnarfing, in its original form, can therefore be discounted as a threat to forensics on the device itself, although the fact that the device is vulnerable may in itself be noteworthy as a possible route via which information could have leaked to other parties in the case, so it is still of interest to the investigator.

2. BlueSnarf++

Description: read and write files via “FTP” service.

Reference: http://trifninite.org/trifninite_stuff_bluesnarfpp.html.

This attack is similar to the Snarf attack, except that full read/write access is obtained, usually to the entire storage areas including removable memory cards etc. Local data can therefore be completely compromised by addition, removal or modification. Typical data that can be accessed in this way include camera images/video and sound files such as ring tones, voice recordings, MP3 etc. In some cases, data files from local applications such as word processors or notepads

