A Survey on Security Threats and Vulnerability attacks on Bluetooth Communication

Trishna Panse[#], Prashant Panse[*]

[#]*Department of Information Technology, RGPV*
Sushila Devi Bansal College of Technology, Indore, India

[*] *Department of Information Technology, RGPV*
Swami Vivekanand College of Technology, Indore, India

Abstract— In this article we present a survey on threats and vulnerability attacks on Bluetooth security mechanism. Bluetooth is the personal area network (PAN). It is the kind of wireless Ad hoc network. Low cost, low power, low complexity and robustness are the basic features of Bluetooth. It works on Radio frequency. Bluetooth Technology has many benefits like replacement of cable, easy file sharing, wireless synchronization and internet connectivity. As Bluetooth Technology becomes widespread, vulnerabilities in its security protocols are increasing which can be potentially dangerous to the privacy of a user's personal information. Security in Bluetooth communication has been an active area of research for last few years. The article presents various security threats and vulnerability attacks on Bluetooth technology.

***Keywords*— Bluetooth security; security protocol; vulnerability; security threats; bluejacking; eavesdropping; malicious attackers.**

I. INTRODUCTION

Bluetooth technology uses various types of protocol as key agreement protocol. Generating keys for Bluetooth technology is very decisive part, so our main focus is on functioning of key agreement protocol. For example if two devices want to communicate securely to each other first of all they want to generate a secret key because initially they do not have shared secret key, because of this they use the key agreement protocol. When this protocol performed the link key and encryption keys are generated. The encryption key is used in E0 stream cipher and the link key is used in challenge response technique which is used for authentication in Bluetooth. Link key is of two types: unit key and combination key. Unit key: same key is use for authentication for all the connection. Combination key: is specific to one pair of Bluetooth device [1]

II. PROTOCOL STACK OF BLUETOOTH

A protocol stack is a combination of software/hardware implementation of the actual protocols specified in the standard [2]. It also defines how the devices should communicate with each other based on the standard. The Bluetooth protocol stack is shown in Fig. 1.

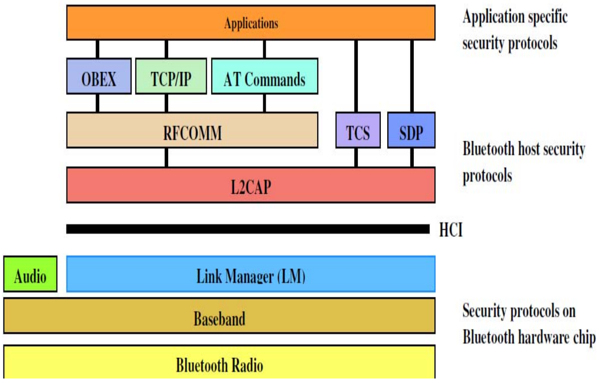

Applications
OBEX
TCP/IP
AT Commands
RFCOMM
TCS
SDP
L2CAP
HCI
Audio
Link Manager (LM)
Baseband
Bluetooth Radio
Application specific security protocols
Bluetooth host security protocols
Security protocols on Bluetooth hardware chip

Fig. 1 Bluetooth Protocol Stack [2]

1. Radio Frequency (RF) Layers

The radio layer is the physical wireless connection. In order to reduce collisions with other devices using the ISM range, the radio uses frequency mapping to separate the range into 79MHz bands, starting at 2.402GHz and stopping at 2.480Hz and uses this spread spectrum to hop from one channel to another, up to 1600 times per second.

2. Base band layer

The base band allows the physical connection between devices. It is responsible for controlling and sending data packets over the radio link. When a Bluetooth device connects to another Bluetooth device, they form a small network called a piconet. A piconet is a small network of Bluetooth devices, where every device in the network can be in one of the following states.

Master: The Bluetooth device that initiates communication. The master sets the time and broadcasts its clock to all slaves providing the hopping pattern, in which they hop frequency at the same time.

Slaves: The state given to all devices that are connected to another. The device can be an active slave if it actively transmits or receives data from the master, or a passive slave if it is not currently sending or receiving any information. The passive slaves check if there is a connection request from the master by enabling their RF receivers periodically.

Standby: All devices that are not connected to a master (i.e. not slave) are called standby devices. When searching for other devices, a device enters the inquiry state. When a device starts creating a Bluetooth link, it enters the page state. Also a device can go to a low power mode to save power.

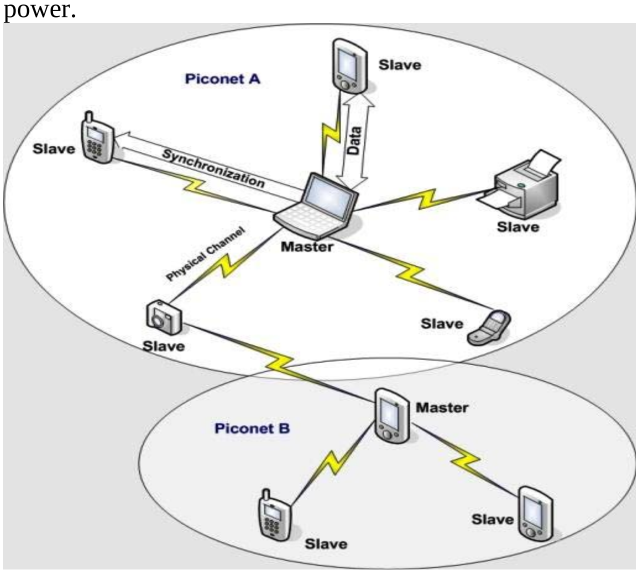


Fig 2: Typical Scatternet[

3. Link 2 Manager Protocol (LMP)
   The LMP protocol uses the links set up between devices by the base band to establish logical connection responsibilities of the LMP. It also includes security aspects and device authentication.
4. Logical Link Control and Adaptation Protocol (L 2CAP)
   The L2CAP is responsible for receiving applicative data from the upper layers and translates it to the Bluetooth format so that it can be transmitted to the higher layer protocol over the base band.
5. Radio Frequency Communication Protocol (RFCOMM)
   The RFCOMM is used to emulate serial connections over the base band layer to provide transport capabilities for upper level services and avoiding direct interface of the application layer with L2CAP.
6. Service Discovery Protocol (SDP)
   The SDP protocol is used to discover services, providing the basis for all the usage models.
7. Telephony Control and Signaling layer (TCS)
   The TCS protocol defines the call control signaling for the establishment of speech and data calls between Bluetooth devices. TCS signaling messages are carried over L2CAP.
8. Application Layer
   The application layer contains the user application. The applications interact with the RFCOMM protocol layer to establish an emulated serial connection. [3]

III. Bluetooth Security Architecture

Security for Bluetooth is provided on the radio paths, which means that link authentication and encryption may be provided, but true end-to-end security is not possible without providing security solutions for the higher layers of Bluetooth. Basically, Bluetooth addresses the three security services:

Confidentiality: The first goal of Bluetooth is confidentiality or privacy. This service prevents an eavesdropper from reading critical information. In general, with this security service only the authorized user can access the data.

Authentication: Providing identity verification of the communicating devices is the second goal of Bluetooth. Authentication allows the communicating devices able to recognize each other; hence communication aborts if the user is not authorized.

Authorization: The third goal of Bluetooth is to control access to the resources. This is achieved by determining the users who are authorized to use the resources.

Keys used in Bluetooth security

Unit Keys: The authentication and encryption mechanisms based on unit keys are the same as those based on combination keys. However, 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, all trusted devices are able to eavesdrop on any traffic based on this key. A trusted unit that has been modified or tampered with could also be able to impersonate the unit distributing the unit key. Thus, when using a unit key there is no protection against attacks from trusted devices. [3][4]

Combination Keys: The combination key is generated during the initialization process if the devices have decided to use one. Both devices generate it at the same time. First, both of the units generate a random number. With the key generating algorithm E21, both devices generate a key, combining the random number and their Bluetooth device addresses. After that, the devices exchange securely their random numbers and calculate the combination key (K_{ab}) to be used between them as shown in Fig 3

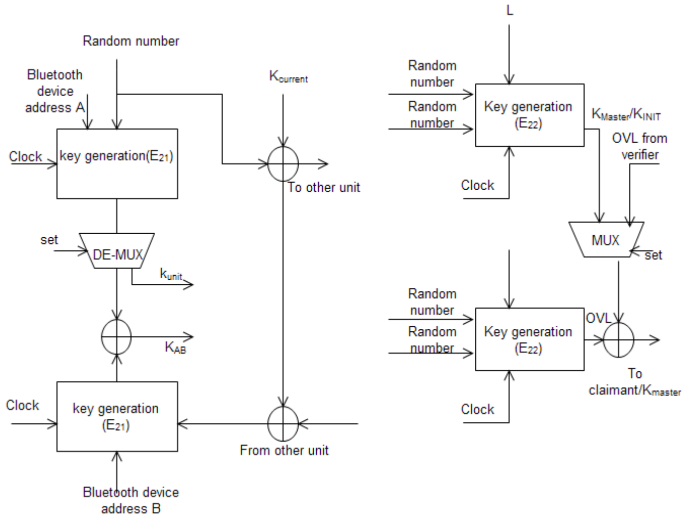


Fig 3: Link Key generation

Encryption keys: The encryption key is generated from the current link key, a 96-bit Ciphering Offset Number (COF) and a 128-bit random number. The COF is based on the Authenticated Ciphering Offset (ACO), which is generated during the authentication process. When the Link Manager (LM) activates the encryption, the encryption key is generated. It is automatically changed every time the Bluetooth device enters the encryption mode.[2]

IV. Vulnerability Attacks on Bluetooth

In Today's Era Bluetooth devices are frequently used, malicious security violations are common events now and it is expected to increase in near future. So Bluetooth architecture needs a constant upgrading to prevent new unknown threats.

Like any other wireless communication system Bluetooth transmission can be deliberately jammed or intercepted. False or modified information could be passed to the devices by the

cyber criminals. Security threats in Bluetooth can be divided into three major categories [5] as follows:

- Disclosure threat: The information can leak from the target system to an eavesdropper that is not authorized to access the information.
- Integrity threat: The information can be deliberately altered to mislead the recipient.
- Denial of Service (DoS) threat: The users can be blocked to get access to a service by making it either unavailable or severely limiting its availability to an authorized user. [2]

Nowadays, it is also possible to transform a standard Bluetooth dongle into a full-blown Bluetooth Sniffer. Tools for reverse engineering the firmware of Bluetooth dongles are also available. The tools include a disassembler for the official firmware, and an assembler that can be used for writing a custom firmware. With these tools one can now write a custom firmware for CSR based Bluetooth dongles to include raw access for Bluetooth sniffing. Moreover, the techniques for finding hidden (i.e., non-discoverable) Bluetooth devices in an average of one minute will be ported onto a standard CSR dongle via a custom firmware. This will open new doors for practical Bluetooth security research and it will also provide a cheap basic weapon to all attackers for Bluetooth sniffing. Therefore, Bluetooth sniffing has become a very popular sport among attackers and hackers. Thus making Bluetooth security becomes even more alarming. There are some other threats that have been reported in the literatures fall outside of these three categories. Some of the threats are presented in the following section. [2]

V. Some other Vulnerability Attacks on Bluetooth

Bluetooth threats have evolved since those days, and while they aren't extremely dangerous they can still be quite serious. Bluetooth attacks depend on exploiting the permission request/grant process that is the backbone of Bluetooth connectivity. Here are a few examples of the mobile security threats in which Bluetooth makes us vulnerable, along with tips to secure your mobile workforce devices.

1. **Bluejacking** is basically Bluetooth spam. Bluetooth users can send what is essentially an e-business card to other Bluetooth users within a 30-foot radius of their device; if downloaded that e-card can add the contact to the now-infected user's address book. That contact then can send messages to this infected device. And some Bluejackers make the spam delivery even more simple, putting the spam directly into the Blutooth device name so when the recipient receives a message that a new device tried to connect, the requesting device is identified by the smap message — " '30% Off Name-Brand Prescriptions' is trying to connect to your device," for instance. Such attacks can be increased to a 300-plus-foot radius if the attacker is using a directional antenna and amplifier. [7]

2. The **Car Whisperer** is software that lets attackers send audio to, and receive audio from, a Bluetooth-enabled car stereo. This means that these attackers can listen to your calls and chime in, if they want to.[7]

3. Bluebugging, is a bit more dangerous than the first two, allowing attackers to remotely access a user's phone and use its features, including listening to calls, forwarding incoming calls, placing calls and sending text messages — and the user doesn't realize what's happening. This can result in expensive phone bills if it's used to make premium or international calls. [7]

4. General software vulnerabilities

Software in Bluetooth devices – especially those using the newer Bluetooth 4.0 specification – will not be perfect. It's unheard of to find software that has zero security vulnerabilities.

To combat this threat: Switch off your Bluetooth when you're not using it. [6]

5. Eavesdropping

Bluetooth – named after the Viking king, Harald Bluetooth Gormsson, thanks to his abilities to make 10th-century European factions communicate – is all about wireless communication. Just like with Wi-Fi, Bluetooth encryption is supposed to stop criminals listening in to your data or phone calls. In other words, eavesdropping shouldn't be a problem. However, older Bluetooth devices use versions of the Bluetooth protocol that have more security holes. Even the latest specification (4.0) has a similar problem with its low-energy (LE) variant.

To combat this threat: Ban devices that use Bluetooth 1.x, 2.0 or 4.0-LE. [6]

6. Denial of service

Malicious attackers can crash your devices, block them from receiving phone calls and drain your battery.

To combat this threat: Again, switch off your Bluetooth when you're not using it. [6]

7. Bluetooth range is greater than you think

Bluetooth is designed to be a "personal area network." That is to say, devices that are more than a few feet away should not be accessible via Bluetooth.

However, you're not safe if you simply ensure there's distance between you and a potential attacker; hackers have been known to use directional, high-gain antennae to successfully communicate over much greater distances.

To combat this threat: Once again, switch off your Bluetooth! [6]

8. MAC Spoofing Attack

Among all passive attacks, the most frequently reported attacks are classified as MAC spoofing and PIN cracking attacks. Malicious attackers can perform MAC spoofing during the link key generation while Piconets are being formed. Assuming the attack is made prior to successful pairing and before encryption is established attackers can easily intercept data intended for other devices. Attackers, with specialized hardware, can easily use spoofing to terminate legitimate connections or capture and/or manipulate data while in transit. Bluetooth SIG did not provide a good solution to prevent this type of attack. They only advised the users to do the pairing process in private settings. They also suggested that a long, random, and variable PIN numbers should be used. [2]

9. PIN cracking attack

Using a Bluetooth frequency sniffer (or protocol analyzer) and acquisition of a FHS packet, attackers can attempt to acquire IN_RAND, LK_RAND and the initialization key during the

entire pairing and authentication processes. The attacker would have to list all of the possible permutations of the PIN. Using the acquired IN_RAND and BD_ADDR they would need to try possible permutations as input in the E22 algorithm. Eventually they would be able to find the correct initialization key. The next step is to hypothesize and test possibilities of the shared session link key using all of the previous data. Assuming the right information is collected, the proper equipment is used, and enough time is allowed, PIN cracking becomes a fairly simple task. The proposed solutions for these types of attacks involve different pairing and authentication schemes that involves using a combination of public/private keys.

10. Man-in-the-Middle/Impersonation Attack

Man-in-the-Middle and impersonation attacks actually involve the modification of data between devices communicating in a Piconet. A Man-in-the-Middle attack involves relaying of authentication message unknowingly between two devices in order to authenticate without knowing the shared secret keys. By forwarding the message of two devices trying to pair, an

attacker will relay two unique link keys. By acting between two devices an attacker can trick two devices into believing they are paired when in fact they have paired with the attacker. The suggested solutions to this kind of attack involve incorporating more Piconet specific information into the pairing process. For example, timestamps and nested mutual authentication can be used to determine the legitimacy of a device’s challenge before responses are sent in return. [2]

11. BluePrinting Attack

A BluePrinting attack is used to determine the manufacturer, device model and firmware version of the target device. An attacker can use Blueprinting to generate statistics about Bluetooth device manufacturers and models, and to find out whether there are devices in the range of vulnerability that have issued with Bluetooth security, for example. BluePrint 0.1 is a tool for performing BluePrinting attack. It runs on Linux and it is based on the BlueZ protocol stack. BluePrinting attacks work only when the BD_ADDR of the target device is known.[2]

12. Blueover attack

Blueover and its successor Blueover II are derived from Bluetooth. However, because they run on handheld devices such as PDAs or mobile phones and are capable of stealing sensitive information by using a BlueBugging attack. A Blueover attack can be done secretly, by using only a Bluetooth mobile phone with Blueover or Bluover II installed. Bluleover and Bluover II run on almost every J2ME (Java 2 Micro Edition) compatible handheld device. They are intended to serve as auditing tools which can be used for checking whether Bluetooth devices are vulnerable or not, but they can be used for attacking against Bluetooth devices as well. A Blueover attack is dangerous only if the target device is vulnerable to BlueBugging. Moreover, an attacker has to know the BD_ADDR of the target device.[2]

13. Off-Line PIN Recovery Attack

An off-line PIN recovery attack is based on intercepting the IN_RAND value, LK_RAND values, AU_RAND value and SRES value, and after that trying to calculate the correct SRES value by guessing different PIN values until the calculated SRES equals the intercepted SRES. It is worth noting that SRES is only 32 bits long. Therefore, a SRES match does not necessarily guarantee that an attacker has discovered the correct PIN code, but the chances are quite high especially if the PIN code is short.[2]

14. Brute-Force Attack

A brute-force BD_ADDR scanning attack uses a brute-force method only on the last three bytes of a BD_ADDR, because the first three bytes are publicly known and can be set as fixed. A brute-force BD_ADDR scanning attack is perhaps the most feasible attack when target devices are Bluetooth mobile phones, because millions of vulnerable Bluetooth mobile phones are used every day all over the world.[2]

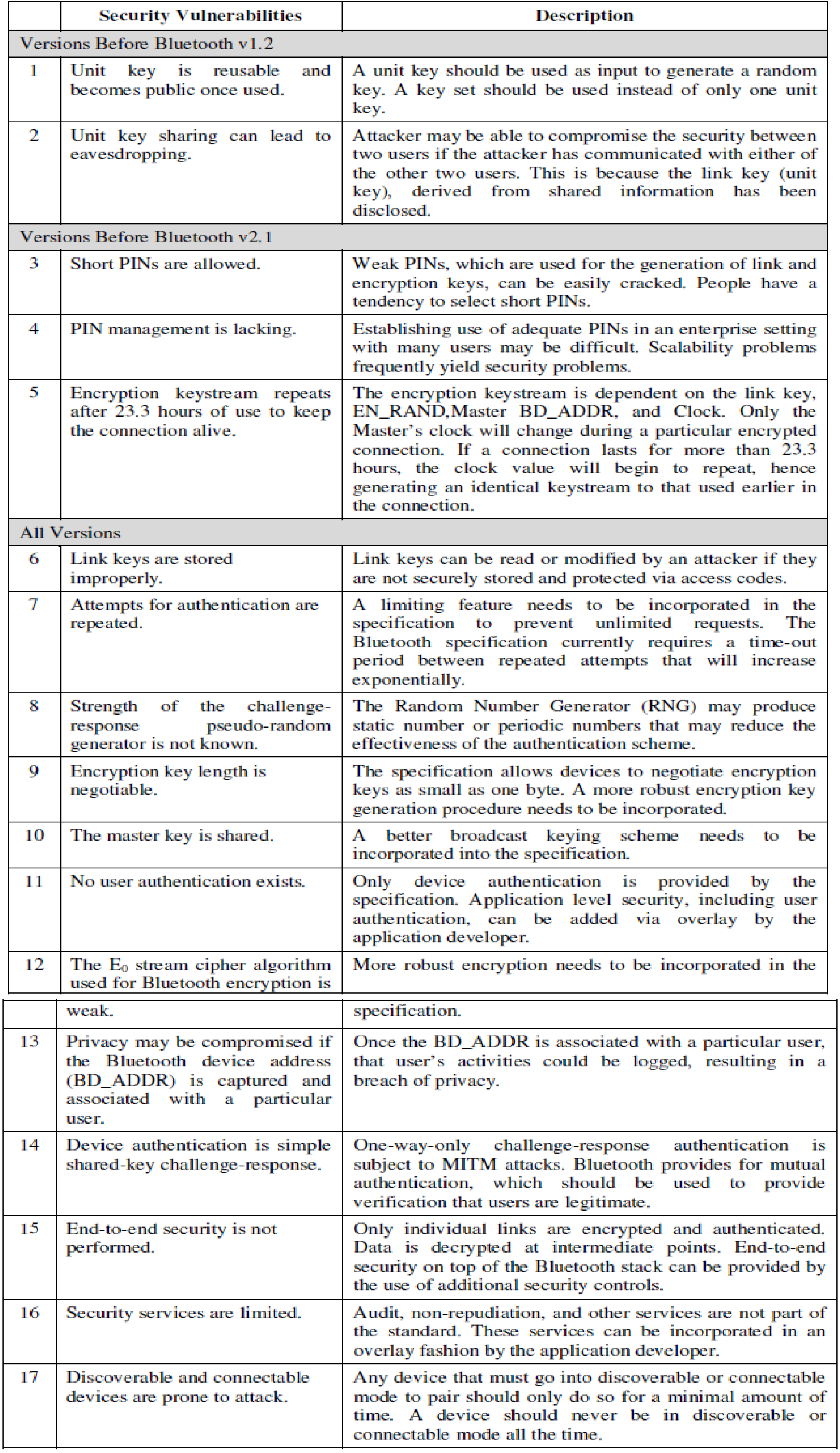
15. Reflection Attack

Reflection attacks (also referred to as relay attacks) are based on the impersonation of target devices. An attacker does not have to know any secret information, because the attacker only relays (reflects) the received information from one target device to another during the authentication. Hence a reflection attack in Bluetooth can be seen as a type of a MITM attack against authentication, but not against encryption.[2]

16. 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 monitoring their devices at that moment, a connection is established. The attacker may continue using the resources that a trusted relationship with that device grants access to until the users notice such attacks. The attacker can not only retrieve data from the phone, but other services such as modems, Internet, WAP and GPRS gateways may be accessed without the owner’s knowledge or consent. A backdoor attack works only if the BD_ADDR of the target device is known. Moreover, the target device has to be vulnerable to a backdoor attack. [2]

VI. COUNTER MEASURES



	Security Vulnerabilities	Description
Versions Before Bluetooth v1.2		
1	Unit key is reusable and becomes public once used.	A unit key should be used as input to generate a random key. A key set should be used instead of only one unit key.
2	Unit key sharing can lead to eavesdropping.	Attacker may be able to compromise the security between two users if the attacker has communicated with either of the other two users. This is because the link key (unit key), derived from shared information has been disclosed.
Versions Before Bluetooth v2.1		
3	Short PINs are allowed.	Weak PINs, which are used for the generation of link and encryption keys, can be easily cracked. People have a tendency to select short PINs.
4	PIN management is lacking.	Establishing use of adequate PINs in an enterprise setting with many users may be difficult. Scalability problems frequently yield security problems.
5	Encryption keystream repeats after 23.3 hours of use to keep the connection alive.	The encryption keystream is dependent on the link key, EN_RAND,Master BD_ADDR, and Clock. Only the Master's clock will change during a particular encrypted connection. If a connection lasts for more than 23.3 hours, the clock value will begin to repeat, hence generating an identical keystream to that used earlier in the connection.
All Versions		
6	Link keys are stored improperly.	Link keys can be read or modified by an attacker if they are not securely stored and protected via access codes.
7	Attempts for authentication are repeated.	A limiting feature needs to be incorporated in the specification to prevent unlimited requests. The Bluetooth specification currently requires a time-out period between repeated attempts that will increase exponentially.
8	Strength of the challenge-response pseudo-random 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.
9	Encryption key length is negotiable.	The specification allows devices to negotiate encryption keys as small as one byte. A more robust encryption key generation procedure needs to be incorporated.
10	The master key is shared.	A better broadcast keying scheme needs to be incorporated into the specification.
11	No user authentication exists.	Only device authentication is provided by the specification. Application level security, including user authentication, can be added via overlay by the application developer.
12	The E_0 stream cipher algorithm used for Bluetooth encryption is weak.	More robust encryption needs to be incorporated in the specification.
13	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 breach of privacy.
14	Device authentication is simple shared-key challenge-response.	One-way-only challenge-response authentication is subject to MITM attacks. Bluetooth provides for mutual authentication, which should be used to provide verification that users are legitimate.
15	End-to-end security is not performed.	Only individual links are encrypted and authenticated. Data is decrypted at intermediate points. End-to-end security on top of the Bluetooth stack can be provided by the use of additional security controls.
16	Security services are limited.	Audit, non-repudiation, and other services are not part of the standard. These services can be incorporated in an overlay fashion by the application developer.
17	Discoverable and connectable devices are prone to attack.	Any device that must go into discoverable or connectable mode to pair should only do so for a minimal amount of time. A device should never be in discoverable or connectable mode all the time.

Table 1: Bluetooth Security Vulnerability [2]

SECURITY TIPS

- Enable Bluetooth only when you need it.
- Keep the device in non-discoverable (hidden) mode.
- Use long and difficult to guess PIN key when pairing the device.
- Reject all unexpected pairing requests.
- Update your mobile phone firmware to a latest version.
- Enable encryption when establishing BT connection to your PC.
- Update your mobile antivirus time to time to keep pace with the new emerging viruses and Trojans.[3]

REFERENCES

[1] Trishna Panse, Vivek Kapoor, Prashant Panse, *"A Review on Key Agreement Protocols used in Bluetooth Standard and Security Vulnerabilities in Bluetooth Transmission"* International Journal of Information and Communication Technology Research, Volume 2 Number 3, March 2012. Available online: http://www.esjournals.org

[2] Nateq Be-Nazir Ibn Minar, Mohammed Tarique, *"BluetoothSecurity Threats and Solutions: A Survey"* International Journal of Distributed and Parallel Systems, volume 3, No. 1, January 2012

[3] Trishna Panse, Vivek Kapoor, *"A Review on Security Mechanism of Bluetooth Communication"*, International Journal of Computer Science and Information Technologies, Vol. 3 (2) , 2012.

[4] Christian Gehrmann, *Bluetooth™ Security* White Paper, Bluetooth SIG Security Expert Group.

[5] *"The BlueBug"*, a Bluetooth virus, available at: http://trifinite.org/trifinite_stuff_bluebug.html

[6] *"A Review of Bluetooth Attacks and How to Secure Mobile Workforce Devices"* available at:http://www.webroot.com/us/en/business/resources/articles/corporate-security/a-review-of-bluetooth-attacks-and-how-to-secure-mobile-workforce-devices

[7] *"Bluetooth Connectivity Threatens Your Security"* available at: http://blog.kaspersky.com/bluetooth-security/#