Bluetooth And Its Inherent Security Issues

Tu Niem

Bluetooth And Its Inherent Security Issues

By
Tu C. Niem
SANS GIAC Security Essentials Certification (GSEC) v1.4b
11/04/2002

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

Bluetooth technology is making a strong comeback despite much disappointment when it was first introduced in 1998. Like most newly developed defacto standards, Bluetooth was plagued by delayed rollouts, design flaws, and cost prohibitiveness [8]. This is no longer the case according to Karen Peterson, a journalist for 10Meters.com. The Frost & Sullivan report projects that chipset shipments will rise from 9.23 million in 2001 to over 971 million chipsets per annum in 2006 [9]. Research firm In-Stat also made the same projection [3]. In fact the technology is already being implemented in many areas according to European technology correspondent for InfoWorld Lucas Van Grinsven [3]:

"Already, major Japanese laptop computer makers have fitted their top-of-the-range models with Bluetooth so they can wirelessly hook up with printers, cell phones, mice or even a digital camcorder from Sony Corp. Microsoft has announced it will soon ship Bluetooth mice and keyboards.

Toshiba has taken the concept one step further by showing a fridge and a microwave that wirelessly connect to a tablet computer. The 'talking fridge' which has haunted the Internet community for half a decade became reality here when the Toshiba machine reported it ran out of beer. Meanwhile recipes on the screen activated freezer programs."

As a matter of fact there have already been discussions in regards to the use of the Bluetooth enabled devices on factory floors for systems control [11], in retail stores for purchase transactions, in hospitals for accessing patient data, and in hotel chains to replace the smart card room keys. The possibilities for such a technology as Bluetooth are limitless.

Despite the limitless possibilities and already-gained market share, the Bluetooth specification does have a flaw in its native security implementation. This flaw can subject a user to such threat vectors as default configuration, theft and loss, eavesdropping and impersonation, person-in-the-middle attack, piconet/service mapping, and denial-of-service attacks. These inherent security issues, they can be mitigated quite easily via implementation of application layer security, device configuration guidelines, enforcement policies, and methods for protecting identifying data. We must be aware of how Bluetooth functions, where it can be used, and the risks involved in using these products in order to make informed decisions on the appropriateness of the usage thereof. In order to reach this understanding, one must understand the Bluetooth technology, possible threat vectors, and risk mitigation. It is noted that despite the issues surrounding the Bluetooth native security implementation, it is not likely that the adoption of this new standard is likely to be hindered by it [1].

INTRODUCTION TO BLUETOOTH

Purpose

Bluetooth was designed as a cable replacement technology. It is a short-range radio link designed to connect portable and/or fixed electronic devices. The effective range, to date, is thirty feet or ten meters. It is employed in interconnecting cellular phones with headsets, cellular phones with PDA's, PDA's to mobile or desktop computers, and the like wirelessly. With over 1500 companies in the Bluetooth Special Interest Group (SIG) developing features and capabilities, over sixty profiles have emerged [1]. Bluetooth profiles are specific characteristics or uses for the Bluetooth standard. These profiles come with slightly differing protocol stacks to handle the type of communication necessary for the particular application. For example, the serial profile is used to emulate RS-232 communications over radio frequency (RF). Examples of other profiles are dialup networking (DUN), local area network (LAN), Service Discovery (SDP), headset, and synchronization; just to name a few.

Figure 1. Bluetooth Ad Hoc Topology [6: section 4 page 4]

How It Works

Bluetooth is designed to operate over the 2.4 GHz Industrial Scientific Medicine (ISM) band. The frequency range for the United States, Europe, and most other countries is 2.400 to 2.4835 GHz. Some countries have national limitations in the operating frequency range, for example France uses the operating frequency range 2.4465 to 2.4835 GHz. For this reason, the Bluetooth SIG has introduced a special frequency hopping algorithm to overcome some of these limitations. It

is important to note that devices employing the reduced frequency range will not work with devices employing the full frequency range [10:20].

The Bluetooth protocol employs a combination of circuit and packet switching technology. It can support an asynchronous data channel, up to three simultaneous synchronous voice channels, or a channel which simultaneously supports asynchronous data and synchronous voice. For more detailed information on the specification and data rates please refer to the Bluetooth specification [10:41].

The Bluetooth system consists of a radio unit, a link control unit (Link Controller), and a support unit for link management and host terminal interface functions (Link Manager Protocol). The Link Controller (LC) is responsible for Baseband protocols and other low-level link routines. The Link Manager Protocol (LMP) is responsible for link set-up and control [10:41].

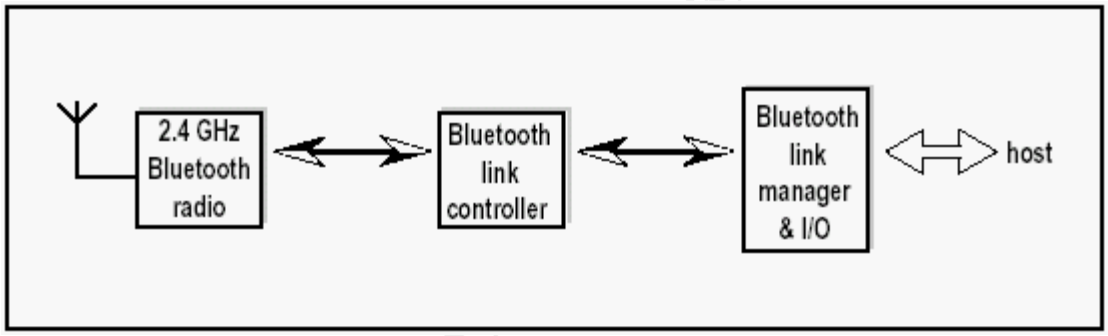

Figure 2. Different functional blocks in the Bluetooth system [10:41].

The Bluetooth system provides a point-to-point (only two Bluetooth devices) or a point-to-multipoint (more than two Bluetooth devices) connection to establish communications between Bluetooth enabled devices. Bluetooth devices sharing the same channel in this way form a piconet. In a piconet, one Bluetooth device acts as the master while the other units as slaves. There can be up to seven active slaves on a piconet. While many more slaves can exist attached to a master, they will not be active [10:41-42]. Communications are established over two defined link types for the master and slaves. The link types are Synchronous Connection-Oriented (SCO) and Asynchronous Connection-Less (ACL) link. The SCO link is used for point-to-point while the ACL link is used for point-to-multipoint connections. In the SCO link type, packets are never retransmitted while in the ACL link type they are retransmitted to ensure data integrity [10:45-46]. These are the two major differences between the link types.

The following protocols and profiles are used predominantly in communications over the SCO and ACL links.

Link Manager Protocol (LMP)

The LMP messages are used to set-up links, maintain security, and maintain control in these piconets. The LMP is responsible for the pairing procedure and handles the challenge response procedure for authentication purposes. The messages are transferred in the payload as opposed to the Logical Link Control and Adaptation Layer Protocol (L2CAP). LMP messages are filtered and interpreted by the Link Manager (LM) on the receiving side. These messages are not propagated to the higher layers. The Link Manager messages have priority over user data and thus will not be delayed by L2CAP traffic. It can be delayed by high retransmissions of individual baseband packets however [10:189-251].

Logical Link Control and Adaptation Layer Protocol (L2CAP)

L2CAP resides in the data link layer. It is layered over the Baseband Protocol of the Bluetooth specification. It provides connection-oriented and connectionless data services to upper layer protocols. Some features are protocol multiplexing, segmentation and reassembly operation, and group abstractions. L2CAP interfaces with other communications protocol such as the Service Discover Protocol and RFCOMM. L2CAP is defined for only the Asynchronous Connection-Less (ACL) links and not the Synchronous Connection-Oriented (SCO) links of the Baseband specification [10:257-329].

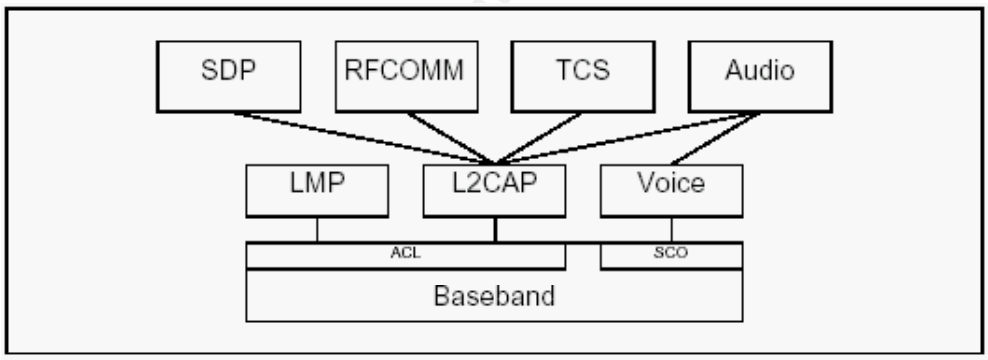

Figure 3. L2CAP in the Bluetooth Protocol Architecture [10:259]

Service Discovery Protocol (SDP)

The Service Discovery Protocol (SDP) provides a mechanism for applications to discover which services are available and to determine the characteristics of those services. This protocol empowers portable Bluetooth enabled devices to cope with the dynamically changing Bluetooth environment when in motion. In essence, this allows a Bluetooth device to discover services available to it as it approaches radio frequency proximity of other Bluetooth devices. Take, for example, a corporate user armed with a PDA roaming the corporate campus can get mail no matter which building or floor they happen to be in as long as there is a Bluetooth device in proximity of that employee that allows them access to the corporate LAN. Since the connection between Bluetooth devices is point-to-point or point-to-multipoint, the roaming Bluetooth device would need to know the

existence of another Bluetooth device to establish communications with it [10:335-391].

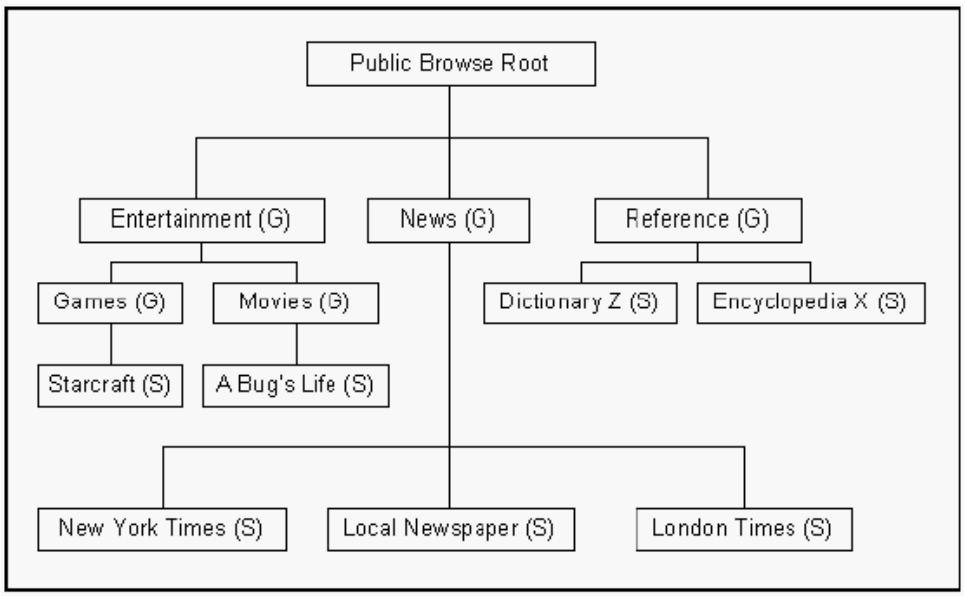

Figure 4. Sample Service Browsing Using SDP [10:347].

RFCOMM

The RFCOMM protocol provides Bluetooth devices with serial port emulation. The protocol functions over L2CAP and can emulate the nine circuits of RS-232 serial ports. The protocol supports up to sixty simultaneous connections between two Bluetooth devices. This protocol enables such applications as connecting a PDA to a computer for data synchronization, a computer to a modem, or a PDA to a cellular phone [10:397-424].

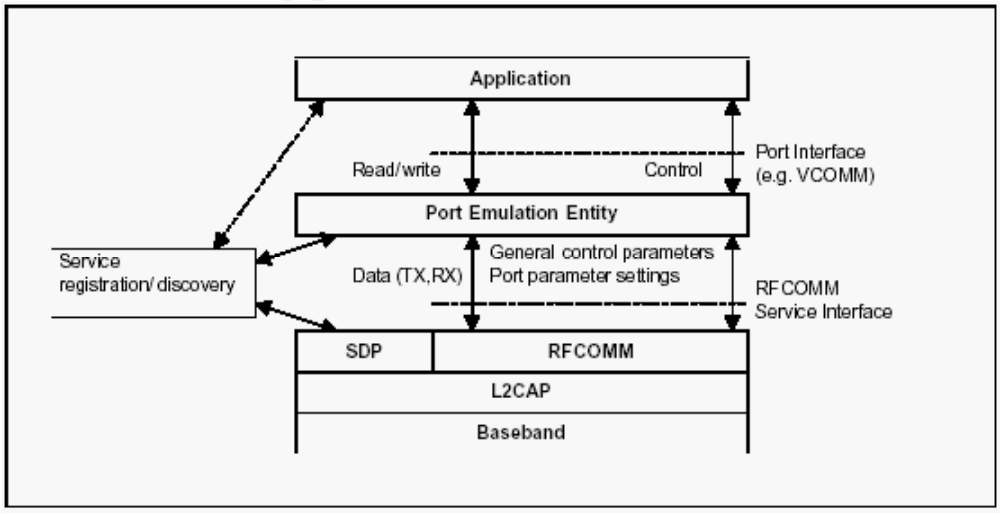

Figure 5. RFCOMM reference model [10:403].

The aforementioned protocols and profiles lend to the functionality of the Bluetooth system. Much of the developed and most used profiles depend on other profiles as depicted in Figure 6. These dependencies also mean that the profiles inherit properties of those profiles that they depend on. For example, the Synchronization Profile depends on the Generic Object Exchange Profile which depends on the Serial Port Profile which depends on the Generic Access Profile. The Serial Port Profile will inherit properties from the Generic Access Profile. The Generic Object Exchange Profile will inherit properties from the Serial Port Profile and so on. Such properties range from security to communication methods.

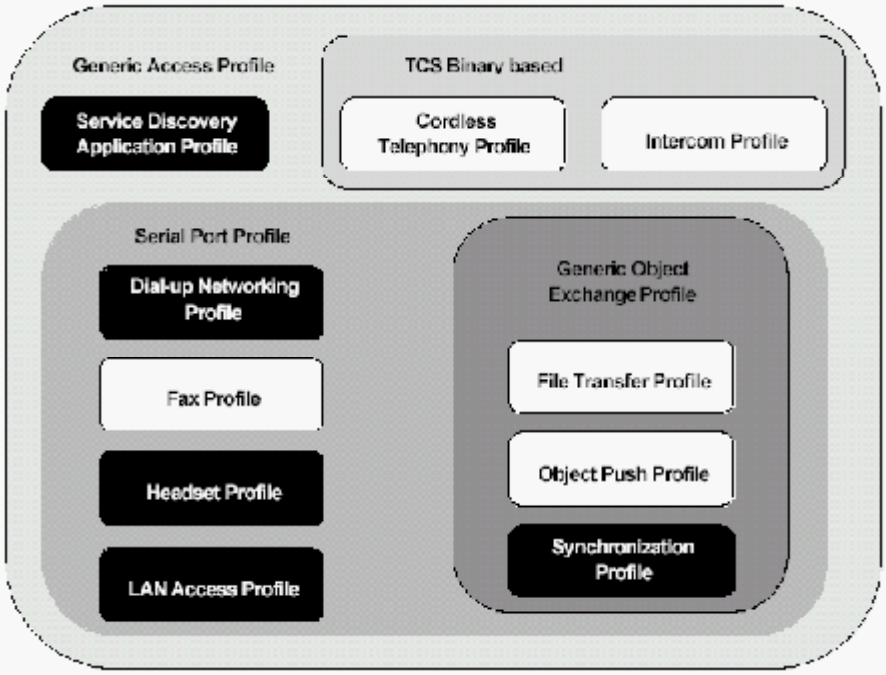

Figure 6. Bluetooth Profiles [2:6].

Market Segments

Bluetooth can support many types of applications. With its many profiles, it is not limited to connecting headsets to cellular phones, cellular phones to PDA's, or PDA's to computers. Developers and vendors have come up with many new and innovative uses for the profiles.

One such use is in human interface devices. These devices include, but are not limited to, keyboards, mice, game controllers, digital cameras, and printers [9]. These devices primarily employ the Personal Area Network (will be replacing the LAN profile), PAN for short, profile of the Bluetooth specification. This may very well be one of the largest market segments due to the pure number of individuals that own a computer system. Demand will be a big factor, of course. The success of this market segment may determine whether Bluetooth will make it in the mainstream. Large technology vendors such as Microsoft, 3Com, Apple,

Hewlett-Packard, and Sony are designing products as well as support in products for Bluetooth technology [3;7;8].

A second use just recently announced is in automobiles enabling many new features such as hands-free operation of a cellular phone. Other features being discussed are automation of climate control, seating, sound system, and ignition. Automotive giants such as Daimler-Chrysler, Ford, and BMW are manufacturing modules/options for their brand of vehicles. There is no news, however, if the Bluetooth devices will be available to limited models [3].

A third and final use we will discuss is in the factory for such tasks as systems control, inventory, systems tuning and troubleshooting, as well as automation. Abb has designed assembly line equipment employing Bluetooth to replace easily damaged cables. The integration of Bluetooth technology in the factory means improved productivity and flexibility [3;11].

Security Features and Modes

The Bluetooth specification defines security at the link level, allowing flexibility in the application security design. This flexibility, however, can come with a price if application designers do not take care in the design process. This link level security is also referred to as Baseband level security and employs authentication and encryption mechanisms [2:22]. The Bluetooth system provides for three basic security services: 1) Confidentiality – addresses information compromise issues from eavesdropping, 2) Authentication – addresses the issue of being able to confirm the authenticity of the identity of devices with whom we are communicating with, and 3) Authorization – addresses the issue of whether the device in question is allowed to access the specific information requested [6: section 4 page 7].

All Bluetooth devices are manufactured with a unique 48-bit identifier known as the Bluetooth device address (BD_ADDR). The device address is publically known and can be obtained via an inquiry routine performed by any Bluetooth device [10:148]. The BD_ADDR is used in establishing all communications. The BD_ADDR of the master device is used to derive the device access code (DAC) and the channel access code (CAC), which are transmitted in clear-text [2;4;6;10;12]. The CAC is also known as the channel identifier used in the Frequency-Hopping-Spread-Spectrum (FHSS) algorithm. The FHSS is used to change between the seventy-nine available bands, twenty-three only in France and Spain, during the communication between Bluetooth enabled devices to: 1) comply with operating frequency ranges of different locale, 2) minimize interference from other devices using the 2.4 GHz range of the ISM band, and 3) avoid possibility of eavesdroppers listening in to the communications. The FHSS hopping rate is 1600 hops per second [10:43].

Initial communication between two Bluetooth enabled devices are established using a method known as pairing [2;10]. Pairing (the term bonding is often used)

is the process by which two Bluetooth enabled devices establish a relationship, either trusted or non-trusted, by means of a key exchange mechanism. The key exchange mechanism is used for authentication and encryption of subsequent communications between the paired Bluetooth devices. Encryption of the communications channel is carried by the E0 stream cipher. The E0 stream cipher consists of three parts: 1) initialization or generation of the payload key, 2) key stream bits generation, and 3) encryption and decryption of the payload [10:158]. The devices, prior to the pairing, do not know of each other's existence or what services are offered. The level of security, or the security modes used, will depend on the application and/or requirement of the specific scenario. It is important to note that the pairing procedure is the weakest process in the Bluetooth Baseband level security specification since all data is transmitted in clear-text until an initialization key is established [2;4]. It is also important to note that only the packet payload is encrypted after the link key and encryption keys have been derived [10:158].

The pairing procedure takes place as follows:

1. **Establishment of Initialization Key**. A temporary initialization key is established for the encryption and decryption of information used in the link key generation process. One of the devices involved in the communication setup chooses a random number and transmits it to the second device. Both devices generate an initialization key as a function of a shared PIN, the BD_ADDR of the receiving device, and the random number. Once the initialization key is generated, a verification process is performed on both devices using a challenge response scheme. The challenge response scheme is as follows: one of the Bluetooth devices generates a random number and computes a function of the second device's BD_ADDR, random number, and newly generated initialization key and transmits it to the second device; the second device computes the function using the same information as the first device and transmits the information back to the first device. If the results are matched by both devices then the mutual verification process is considered successful [10:148-176;6:8-9].
2. **Link Key Generation**. The link key is used for subsequent encryption of the communications between Bluetooth devices. The established initialization key is used to encrypt the cipher text during the link key generation process. There are two types of link key generation processes, the selection of which depends on whether one of the devices involved in the process is limited by memory resources. Devices constrained by memory lack the complexity or resources to generate or store other information in a more complex link key generation process. In this case the device that is memory constrained will request that the first type of link key generation process be used. That is, the unit key of the memory constrained device will be encrypted using the cipher text and transmitted to the

other device to be decrypted and used as the link key. If the devices involved in the communication set-up are not memory constrained then the second type of link key generation process is employed. In this link key generation process, both devices select a random number and transmits it to the other device as encrypted cipher text using the previously established initialization key. Each device then computes a number as a function of both random numbers and their own unique BD_ADDR. Since the devices know each others BD_ADDR they can compute each other's resultant number from the both random numbers and the other party's BD_ADDR. Both units then compute the link key as a function of both computed numbers. The link keys are then passed to the same type of verification procedure as in the initialization key generation process [10:148-176;6:9].

The Bluetooth specification has three main security modes. These modes are used singularly or in combination for most of the communications between Bluetooth devices. Each security mode has its advantages and disadvantages which should be considered carefully for the type of application/configuration intended. We will discuss these security modes and their features as well as limitations.

Security Mode 1: A Bluetooth device will not initiate any security. This is a non-secure mode. In essence the authentication and encryption security procedures are bypassed allowing any Bluetooth device to connect to it [6: section 4 page 7].

Security Mode 2: A Bluetooth device does not initiate security procedures before channel establishment on the L2CAP level. This mode allows different and flexible policies for applications, especially running applications with different security requirements in parallel. This is a service level enforced security mode. The concept of a security manager is introduced in this mode to control access to services. The centralized security manager maintains access control policies and is responsible for interfacing with other protocols and device users. The access control policies allows for more robust control over who has authorization to access certain services. Authentication, confidentiality, and authorization are supported in this mode [6: section 4 page 7].

Security Mode 3: A Bluetooth device initiates security procedures before the link set-up on the LMP level is completed. This is a link level enforced security mode and is fixed. Since this security mode is fixed it is not aware of any application layer security. Authentication (unidirectional or mutual) and encryption are supported in this mode. Authentication and encryption are accomplished using a shared secret link key that is derived during the pairing process [6: section 4 pages 7 through 8].

In addition to the three security modes, a Bluetooth device can be in one of discoverable or non-discoverable modes. Additionally, a Bluetooth device can be in one of connectable or non-connectable modes. A device in the discoverable mode will respond to inquiries from other Bluetooth devices. The opposite is true in the non-discoverable mode. A device in the connectable mode will respond to messages from already discovered Bluetooth devices. Again, the opposite is true of the non-connectable mode [4:1].

As noted thus far, the security features of the Bluetooth specification can leave a user open to many security risks if not managed properly. The pairing, or bonding, procedure of the initial communications set-up is noted to be the weakest process in the Bluetooth security procedures. Since most of the keys are computed as a function of, or is, the Bluetooth unique device address (BD_ADDR) and this information is transmitted in clear text; this can opportune a host of different security vulnerabilities and threat vectors.

POSSIBLE THREAT VECTORS

The inherent security features of the Bluetooth system can leave the devices and data stored on them vulnerable to attacks. The Bluetooth radio itself is also vulnerable. Attacks against confidentiality, data integrity, and availability are possible. Specifically default configuration, theft and loss, eavesdropping and impersonation, person-in-the-middle, piconet/service mapping, and denial-of-service attacks can be carried out with relative ease. It is important to note that default configuration and theft/loss are not truly attacks in the sense that a malicious individuals launch against another. These two threat vectors are important to understand since they play an important role in realizing the other threat vectors. They also make it that much easier for an attacker to accomplish their intended deeds.

Default Configuration

It is too frequent that individuals abandon proper setup of systems/units for the convenience of "out-of-the-box" configurations. Bluetooth, like any other technology, can be quite complex and cumbersome to configure correctly. This becomes especially difficult when you add the requirements of security. It is quite often that individuals neglect to consider what it is the device is for and what the device may contain that is worth protecting. Often it is only until something unimaginable happens that puts the purpose of security into perspective.

Bluetooth enabled devices are becoming more prominent in everyday life. Cell phones, headsets, PDA's, digital cameras, Bluetooth accessories such as PCMCIA cards, and mobile computers are just a few of these devices. Together, they encompass a significant portion of an individual's everyday life. The average person may not be aware of what Bluetooth technology is or is too busy to be concerned with proper setup. This leaves them as well as the information contained on these devices vulnerable to attacks.

Some Bluetooth enabled devices have the Bluetooth radio turned on by default and performing inquiries when in radio proximity of other Bluetooth enabled devices. While in the inquiry substate, the Bluetooth enabled device will disclose its BD_ADDR and clock. This information can be used for impersonation, eavesdropping, or location correlation for profiling. The BD_ADDR of a Bluetooth enable device is used to deterministically derive the link key which is used for encryption of communication between devices. The clock and BD_ADDR of the master device of a piconet is used to calculate the frequency hopping sequence for the communication in the piconet. The frequency hopping sequence, as describe previously, is used to minimize interference with other devices operating in the frequency range of Bluetooth and to avoid the possibility of someone eavesdropping on the communication.

Bluetooth enabled devices typically have a minimal security level, security mode 1, set by default. This poses a serious threat especially when combined with the fact that Bluetooth enabled devices have the Bluetooth radio turned on by default whenever the devices are powered on. Recall that in security mode 1 authentication and encryption are not required. Any Bluetooth device can connect to it and request data without the knowledge of the user. Link keys are also not deleted after a specifiable period of time. This poses great risk in the case of theft or loss of the device. This aspect will be discussed in greater detail in the following section. An attacker may be able to initiate the pairing procedure and gain unauthorized information from the unsuspecting victim. All that is required is radio proximity of the two parties with their Bluetooth enabled devices. This unauthorized information could be of the identity of the victim's Bluetooth device, identity of the victim, data of some sort, or Personal Information Management data such as contacts and scheduled events. All of this information can be used in identity attacks (impersonation) or espionage of some form.

Default configurations of devices are intended to make initial use and setup of the device simple and convenient. It is the manufacturer's way of providing some simplicity to an otherwise complex technology. It is with good intention that this is done but it comes at a price if individuals are not diligent in becoming familiar with the functionality of the product and the dangers of not properly configuring the device.

Theft and Loss

Electronic and computing devices continue to become smaller and more powerful as technology advances. Portability is of great convenience and the power affords much productivity and use. Greater portability means the potential for loss and theft is greater. The devices can perform more functions and store more data. This, however, makes them great instruments for attackers whether owned by the attackers or acquired by some means from a victim.

According to Kevin Jonah, a journalist for the Government Computer News: "A Treasury Department report in January, for example, revealed that the IRS had lost or misplaced 2,332 notebook computers over the last three years, potentially compromising taxpayer data." Portable devices equipped with Bluetooth radios are becoming increasingly abundant and more powerful [5]. Cellular phones, wireless headsets, and PDA's are just of few examples of these. Bluetooth wireless technology makes it a great convenience for using a wireless headset with a cellular phone, synchronize data between PDA's and personal computers, and program a cellular phone from a PDA or personal computer. Not only are the data contained on these devices at risk if they fell into the wrong hands but any device that these Bluetooth enabled devices have paired with are also vulnerable.

Bluetooth devices that previously have established a trust relationship (paired with another Bluetooth device) will keep this trust relationship and store the

respective keys in non-volatile memory unless configured to delete these keys after a specified period of time. This becomes a great risk to other Bluetooth enabled devices owned by the victim who lost the Bluetooth device or others who have paired with the lost/stolen device. Since the keys, such as link keys and unit keys, of other Bluetooth enabled devices which have paired with the lost/stolen device are stored on that device it is possible to: 1) use the device to eavesdrop on communications of devices that have paired with it previously, 2) establish communication with the unsuspecting devices it has paired with previously and obtain unauthorized data, 3) use the unit keys to program a more powerful Bluetooth enabled device to impersonate or spoof that device to accomplish more sophisticated attacks, 4) obtain or determine personal identification numbers of the device or devices that it has paired with previously, or 5) determine some kind of relationship between the victim and others who own devices that have paired with the lost/stolen device. We note that the latter would require a bit more reconnaissance and correlation work but it is not impossible to derive given the type of data stored on these portable devices and the type of information disclosed by a Bluetooth device during communications [2;4;6;10].

Eavesdropping and Impersonation

Eavesdropping is not a new concept in information warfare as well as everyday life. Some common devices used for eavesdropping on communications are scanners for cordless/mobile phones and network sniffers (whether software or hardware). These devices allow an individual to intercept or listen in on communications between two or more parties. To prevent such a probability on Bluetooth communications, the Bluetooth SIG designed a frequency-hopping-spread-spectrum algorithm into the Bluetooth protocol [4;6;10]. The airways are essentially open. There is no need to locate wires to tap into. All that is required is a device designed or modified to listen on those frequencies. These devices are readily available and are manufactured with good intensions.

The Bluetooth SIG designed a frequency-hopping-spread-spectrum algorithm to prevent the probability of eavesdropping on and interference with Bluetooth communications. The Bluetooth devices calculate and agree upon a frequency hopping sequence during communication establishment. The seed of the frequency hopping sequence is calculated from the BD_ADDR and clock of the master device in the piconet. Once a seed has been calculated and agreed upon the devices then hop between the seventy-nine frequencies about 1600 times per second. Encryption is also built into the Bluetooth protocol to provide protection from eavesdropping [4;6;10]. This is a rather good idea for deterring eavesdropping except, as in the case of most deterrence strategies, all that is required is time and determination to derive a countermeasure against such a strategy.

Firstly, we not that if the device was lost or stolen as discussed in the previous section, the information contained on this device can be used to eavesdrop on

future communication of devices it has paired with. This is possible due to the fact that the keys sent and derived from the pairing procedure of two devices are stored on both devices establishing a trust relationship. Take, for example, a Bluetooth headset that was paired with a victim's cellular phone. If this headset were to be lost or stolen, it could be used to eavesdrop on the victim's conversations after the incident. A trust relationship has already been established and the headset already contains the necessary authorization and authentication data to establish communications with the cellular phone. The necessary keys for encryption/decryption of the communication between the two devices are already stored on the headset and cellular phone. On the same token, the unit number of the headset can be used to impersonate the headset itself on a more powerful Bluetooth enabled device to provide the attacker with increased functionality and tools. Since the headset also stores information about the cellular phone that it has paired with, that information can also be used to impersonate the cellular phone to other devices such as the victim's mobile computer or PDA to further violate the victim's right of a confidentiality, information availability and data integrity. If the victim were to delete the keys from the cellular phone upon lose then it would still be possible to eavesdrop on the communications given that the identification of the cellular phone are stored on the headset. This information could be extracted and used to perform offline crunching of the PIN to be used towards determining the initialization key and, subsequently, the link key and encryption keys. This scenario can also be applied to other portable Bluetooth enabled devices as well.

Secondly, the frequency hopping algorithm can be circumvented using a Bluetooth listening device that is modified to listen on all frequencies or by determining the seed of the frequency hopping sequence in use between the devices in communication. Bluetooth listening devices are available through vendors as devices for diagnosing Bluetooth communications issues. These devices act as sniffers to capture data in Bluetooth communication. Software sniffers for the Bluetooth protocol are also available. This allows an attacker to capture some of or the entire communication between Bluetooth devices and decode them offline. Even without such Bluetooth listening devices, it is still possible to circumvent the frequency hopping sequence. This is accomplished by determining the seed of the frequency hopping sequence by using the BD_ADDR and clock of the master device in the piconet. Keep in mind that there is always one master in a piconet and the frequency hop between the masters to any device is different from the master to any other device. Recall that Bluetooth devices send identifying information about themselves in the packet header in clear text such as the BD_ADDR and their respective clocks. An attacker can scan the inquiry frequencies to determine which device is the master device and its BD_ADDR and clock. This is accomplished by observing the response messages. A Bluetooth device responding to an inquiry reveals its DB_ADDR and clock. A Bluetooth device that is the master of a piconet reveals his identity and clock during paging. We note that if encryption is used in the established Bluetooth communication, the ability to eavesdrop may prove fruitless without the

encryption keys to decrypt the packets. This is covered in the next method [4:10-11].

Finally, we note that encrypting the Bluetooth communication channel may not necessarily afford the users much protection given weak PIN's and initial pairing of two Bluetooth devices are conducted in clear text. The pairing procedure is necessary to establish a relationship between the devices as well as exchange keys for encrypting the communications channel. If this pairing procedure is conducted in a public place then it is possible that an attacker can intercept the communication and gather the necessary data to decrypt the payloads. Until the devices are paired, they know only of their own keys and local data. Recall that an initialization key has to be generated from a shared PIN, the BD_ADDR of the receiving device, and a random number. The random number and BD_ADDR are sent in clear text. The shared PIN is also sent in clear text if it is not communicated out of band or application layer encryption is not employed. Also recall that the initialization key is used in the link key generation procedure to encrypt communications for deriving the link key. The link key is used to generate subsequent encryption keys. As long as the attacker is present during the pairing procedure to obtain the initialization key, the attacker can also determine the link key with great confidence and be able to determine the subsequent encryption keys. Even if the attacker was not present during a pairing procedure, it would still be possible to obtain the initialization key. This is possible via stealing by participation or offline brute forcing of the PIN used to generate the initialization key. The PIN can be anywhere from 8 to 128 bits and is usually four decimal digits in length [10]. In the stealing by participation scenario, an attacker initiates a pairing procedure with the victim device and guesses the PIN. The attacker records all communications to obtain the random number and the challenge response transcript of the verification procedure. The attacker performs step one and step two of the initialization key procedure against the guessed PIN. If the output of the verification procedure (step two of the initialization key procedure) is correct then the attacker has the correct PIN and continues with generating the initialization key. If the output is incorrect, then the attacker guesses another PIN, performs step one and two of the initialization key procedure locally, and continues until the procedure outputs correct. Keep in mind that the attacker has recorded the entire communication thus the function for generating the initialization key is known as well as the challenge response pair. As before, once the initialization key is obtained the subsequent keys can also be determined [4:10-11].

Person-in-the-Middle Attack

This attack leverages the vulnerabilities in the Bluetooth Baseband specification as well as the vulnerabilities exhibited in the eavesdropping and impersonation attack scenarios. In this attack, an attacker who has already obtained the link keys and unit keys (BD_ADDR) of two Bluetooth devices can intercept the communication and initiate new communications to both devices posing as the other. The attacker impersonates the victim devices to each other thus the victim

devices believe they are communicating directly with each other. The unsuspecting victim devices are effectively silenced since the communication believed to be to each other is going to a third party, the attacker, and is possibly manipulated for the attacker's needs. The attacker sets up two point-to-point communications, one to each device, and negotiates one of two scenarios to avoid radio frequency interference: 1) both victim devices are masters or 2) both victim devices are slaves. Recall that there has to be a master and slave relationship setup in a piconet. In this instance there are essentially two piconets established. This allows for separate frequency hopping sequences [4:11].

In another person-in-the-middle attack scenario, vulnerabilities involving memory constrained devices are exploited. Memory constrained devices rely on its unit key for encryption to reduce the number of keys it is required to store. An untrusted device, call it C, can establish communications with the memory constrained device, call it A. This connection may have other purposes other than obtaining the unit key for the purposes of the attack. In any case, the memory constrained device, A, has shared its unit key with the untrusted device, C. When device A initiates communications with a differing device, call it B, device C can use the obtained unit key and spoof an address to monitor the communications between device A and B without the either party realizing it. The figure below illustrates this [6: section 4 page 15 and 16]

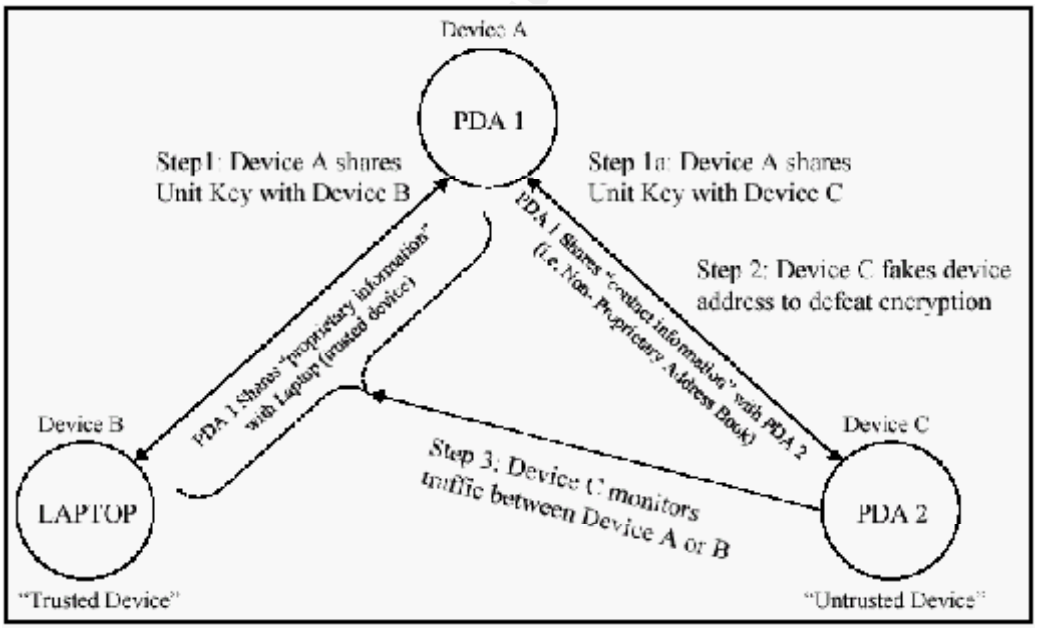

Figure 7. Person-in-the-Middle attack scenario [6: section 4 page 16].

Piconet/Service Mapping

Bluetooth devices that have not paired with other Bluetooth devices need a mechanism in which to inquire about what services are being offered by the other device. This is especially true in a business or cyber café offering wireless LAN

access through the use of Bluetooth technology. The Service Discovery Protocol (SDP) is used for this purpose. The SDP depends on the L2CAP protocol to establish and manage connections. This is important since the basis for security for SDP transactions is the initial pairing of the Bluetooth devices. The SDP does not require authentication or encryption for any of its transactions outside of the application layer. The SDP protocol is not a means to access those services that it discloses but it is enough for an attacker to know what is being offered [2;10].

The first step in any attack is to determine who and what is there. Successful reconnaissance will provide valuable information to an attacker in determining the who, what, why, and how of an attack. With the SDP protocol, an attacker can enumerate the Bluetooth devices and services offered by those devices. This can be in a single or network of Bluetooth devices. Recall that a Bluetooth device responding to an inquiry from another Bluetooth device will reveal its identity and clock. This in turn can be used to map the location of Bluetooth devices in a corporate LAN or cyber café offering Bluetooth services. A service in this respect refers to the different profiles a Bluetooth device can support [2;4;10].

Visualize a corporate site where LAN Access Points (LAP's) are strategically deployed across the campus. These LAP's incorporate the LAN profile of the Bluetooth specification and connect to the LAN infrastructure of the corporation allowing point-to-multipoint Bluetooth connections. Other services or profiles such as Printer and RFCOMM may also be offered by these LAP's. The corporate user armed with a mobile device such as a PDA or laptop computer is empowered to be productive while roaming the site. An executive can manage e-mails and PIM data while he/she is on their way to a meeting without the constraints of a wired connection. And would it not be convenient to be able to see only the devices such as printers near you at that particular moment and access them?

This convenience and flexibility can equate to increased risk if not managed correctly. An attacker can gain information about what services are being offered around the corporate infrastructure simply by roaming the campus with a Bluetooth enabled device. The attacker can find the LAP's deployed around the campus, the services being offered, and correlate them to a specific location. These services may be offered by other Bluetooth enabled devices located around the LAP's. Granted that due to the relatively short range of the Bluetooth radio, an attacker may not be able to conduct such an attack from a parking lot or perimeter. It is possible, however, from inside the campus in the case of poor physical security or an inside attacker. Physical security is non-existent in cyber cafés since they provide the service to the public. This leaves the cyber café open to attack and their devices can possibly be used as agents of far greater schemes. An attacker that maps a target's network and services has the knowledge to plan the attack against those devices or services. Known

vulnerabilities of these devices and/or services can be determined and used to mount an attack since the existence of these devices and services are known.

Denial-of-Service Attack

Denial-of-service (DoS) is possible on the Bluetooth system eventhough there have not been any documented successful cases. DoS attacks can be conducted against the Bluetooth radio, communications channel, or battery (in the case of portable devices). These attacks result in the device losing its ability to access other Bluetooth resources or other Bluetooth devices to be able to access it [6: section 4 page 17].

Bluetooth devices operate over the 2.4GHz ISM band and thus shares the bandwidth with microwave ovens, cordless phones, and other wireless network devices [6: section 4 page 17]. This makes the Bluetooth devices vulnerable to signal jamming. Although the Bluetooth specification has a FHSS mechanism designed into it to minimize interference from such devices, it is not entirely impossible that the signal could be jammed in some of the frequency hops given the ideal environment. Additionally, it is not inconceivable to build or modify a device that can broadcast on all frequencies in the 2.4GHz ISM band jamming signals within a certain radius. Again, it is speculation at this point without proven case studies but not entirely impossible.

Another possible avenue to DoS a Bluetooth device is on the communications channel. Recall that a Bluetooth device can establish communications on one of two physical links the SCO (does not retransmit packets to ensure data integrity) and ACL (does retransmit packets to ensure data integrity) links [10]. Also recall that the theorectical bandwidth of a Bluetooth communications link is about 1 MBps [10;6]. A Bluetooth device can also have a maximum of simultaneous active connections. It is therefore conceivable to DoS a target Bluetooth device by consuming the bandwidth or consuming the maximum allowable simultaneous active connections. In the consuming bandwidth scenario, an attacker can pair with the victim device or spoof a trusted device to request data and never acknowledge receipt of the packets. The communications link, as long as it is not time-based such as voice, will be established over the ACL physical link type. The ACL will retransmit the packet if it does not receive acknowledgement of receipt from the other party in the piconet. If an attacker has enough devices and request sufficiently large amounts of data to tie up the bandwidth on the victim device, then the victim device will not be able to communicate with any other Bluetooth device. In maximum allowable simultaneous active connections scenario, all that is required is that the attacker has a device or devices that take up all seven connections of a target device and keep it occupied with bogus requests. Any other Bluetooth device wishing to establish a connection will be put in the parked state with respect to the victim device and will only be allowed to synchronize the channel to the master [10:41-42].

A third and final DoS attack results when an attacker attempts to exhaust the battery on a portable Bluetooth device. This is accomplished by flooding the targeted device with requests for data transfer or to create connections to the point where the target device is drained of power. The attacker would need to be in radio proximity in order to accomplish such an attack and may need to work around some of the security roadblocks in the case of requesting data. This is not at all difficult given the vulnerabilities and threat vectors previously discussed [6: section 4 page 17].

Although this threat vector does not compromise security of the information on the Bluetooth device, it does prevent the user of the device from conducting normal tasks with the device. This impacts the productivity of the individual or organization as is the original intent of DoS attacks.

RISK MITIGATION

There are several methods to guard against the attacks detailed. The methods are not too terribly complex and do not require modification to the Bluetooth device on the hardware layer. Until further improvements are made in the realm of security in the Bluetooth specification, the methods detailed here for risk mitigation will serve to minimize the risks. Much of the risks can be mitigated by an understanding of the technology, strong security policies, enforcement of the security policies, strong system/node configuration guidelines, and strict adherence to those guidelines. It is important to note that the methods described are by no means the best known methods nor will they guarantee complete protection.

Firstly, always configure the device and software according to established policies. If this device is for personal use and not linked in any way to a business, find out what are some of the best know practices for configuring such a device. The default configuration of a device or software from a manufacturer is designed to allow the user to quickly start using the product. Ease and simplicity, not security, are sole driving forces for most of these manufacturer default configurations. Customizing the setup to a specific need whether security conscience or not is always a good start to minimizing risks.

Secondly, the PIN needs to be protected from interception or cracking by an attacker. This involves carefully choosing PIN's that are sufficiently strong and entering the PIN's out of band. This means that the PIN's have to be sufficiently long and random. This makes it computationally more difficult for an attacker to attempt to guess the PIN. If the PIN were to be entered out of band as opposed to being transmitted between the Bluetooth devices then the attacker can not intercept the PIN. Recall that the PIN is transmitted in cleartext during the pairing procedure. Protecting the PIN will reduce the risk of exposing the communications link to eavesdropping [4;6].

Thirdly, the device identifying data and keys must be protected. Device identifying data refers to the Bluetooth device's unit key (or BD_ADDR) and clock. If at all possible, avoid using a Bluetooth device's unit key as the link key. It is also recommended that Bluetooth devices be set to the non-discoverable mode until a pairing is necessary and set back to the non-discoverable mode after the pairing. This will prevent the Bluetooth devices from responding to queries by unknown Bluetooth devices. It is also recommended that this pairing, if at all possible, be conducted in a private place to prevent attackers from intercepting the communication. This will not prevent an attacker with a Bluetooth listening device that listens on all bands from following the conversation, but at least it would be difficult for attackers with standard Bluetooth devices from doing so. We also note that if the attacker has not determined the link keys and encryption keys, the attacker would not be able to

decrypt the payload. The unit key would be harder to obtain if the device where put into non-discoverable mode. The unit key can be used to impersonate a trusted device. The stored link keys on Bluetooth devices need to be deleted after a certain period especially if the pairing was only for that one time communication of data. If the device were to be lost or stolen, the device could be used to eavesdrop on future communications of devices that it has paired with. Performing the tasks explained thus far will reduce the risk from lost/theft, eavesdropping, impersonation, and middle-person attacks [2;4].

Fourthly, employ application layer security and a public key infrastructure to provide additional security measures. Employing application layer security and a public key infrastructure limits the Bluetooth devices that have access to certain infrastructure services and provides a means of authentication/authorization above that which Bluetooth provides. One possible application layer security is to require a password from the user to authenticate the user in addition to using the Bluetooth device authentication. This helps mitigate against lost/theft, eavesdropping, middle-person attacks, and piconet/service mapping. By promoting a defense-in-depth concept and authentication, only those devices authenticated as being who they really are can enumerate services and access them [4;6].

Lastly, establish device configuration guidelines, security policies, and enforcement mechanisms for the use of Bluetooth devices in the environment. These guidelines and policies should incorporate the mitigation methods detailed above and employ some method of enforcing them. One such method of enforcement could be denial of access at the Bluetooth access devices if the Bluetooth devices operated by the individuals are not configured correctly. There are many ways of enforcing these policies and guidelines but to discuss all of the possibilities would be a whole other whitepaper.

CONCLUSTION

Bluetooth technology is slowly becoming more popular but the security built into the specification is a cause for concern. More and more consumer products used in normal daily life are manufactured with Bluetooth systems such as cell phones, PDA's and mobile computers. It is important that consumers understand the technology and the risks involved in the use thereof. Some of those risks range from loss of productivity to loss of confidentiality and can stem from default configuration, theft and loss, eavesdropping and impersonation, person-in-the-middle, and DoS attacks. Most of these risks can be easily mitigated by following device configuration guidelines and security policies when it comes to the use of a Bluetooth device. Personal as well as corporate users will need to take the initiative to understand the technology and secure their devices until the Bluetooth SIG can work these issues out of the Bluetooth design.

INFORMATION SOURCES

1. 10Meters.com. "Bluetooth Chugging Ahead, Security Won't Derail Adoption." 13 February 2002. http://www.10meters.com/blue_frost_security.html

2. Gehrmann, C. "Bluetooth Security White Paper." Bluetooth Security Expert Group. Revision 1.00. 19 April 2002. http://www.bluetooth.com/upload/24Security_Paper.PDF

3. Grinsven, L. "Bluetooth goes mass market in phone, car and fridge." *InfoWorld*. 14 June 2002. http://www.infoworld.com/articles/hn/xml/02/06/14/020614hnbluetooth.xml

4. Jakobsson, M. & Wetzel, S. "Security Weaknesses in Bluetooth." Lucent Technologies – Bell Labs. 2001. http://www.rsasecurity.com/rsalabs/staff/bios/mjakobsson/bluetooth/bluetooth.pdf

5. Jonah, K. "Securing the airways." *Government Computer News*. Volume 21 Number 10. 6 May 2002. http://www.gcn.com/21_10/guide/18571-1.html

6. Karygiannis, T. & Owens, L. "Wireless Network Security 802.11, Bluetooth and Handheld Devices." National Institute of Standards and Technology. Special Publication 800-48. 24 July 2002. http://csrc.nist.gov/publications/drafts/draft-sp800-48.pdf

7. Lawson, S. "PC, Mac OS updates may spark Bluetooth." *InfoWorld*. 1 August 2002. http://www.infoworld.com/articles/hn/xml/02/08/01/020801hnbluetooth.xml

8. Maier, M. "The Second Coming of Bluetooth." *Business 2.0*. 11 June 2002. http://www.business2.com/articles/web/0,1653,41334,FF.html

9. Peterson, K. "Bluetooth's Future? It's Chips Ahoy!" *10Meters.com*. 20 April 2002. http://www.10meters.com/blue_fs_chipset.html

10. Bluetooth Special Interest Group. "Specification of the Bluetooth System." Version 1.1. 2 February 2001. http://www.bluetooth.com/pdf/Bluetooth_11_Specifications_Book.pdf

11. Teresko, J. "The Real Bluetooth Wireless Payoff." *IndustryWeek*. 1 April 2002. http://www.industryweek.com/CurrentArticles/asp/articles.asp?ArticleID=1223

12. Xydis, T. & Blake-Wilson, S. "Security Comparison: Bluetooth Communications vs. 802.11." Bluetooth Security Expert Group. 1 February 2002. http://www.bluetooth.com/upload/14Bluetooth_Wifi_Security.pdf

Last Updated: March 25th, 2019

Upcoming SANS Training

Click here to view a list of all SANS Courses

SANS Cyber Security Middle East Summit	Abu Dhabi, AE	Apr 04, 2019 - Apr 11, 2019	Live Event
SANS London April 2019	London, GB	Apr 08, 2019 - Apr 13, 2019	Live Event
Blue Team Summit & Training 2019	Louisville, KYUS	Apr 11, 2019 - Apr 18, 2019	Live Event
SANS Riyadh April 2019	Riyadh, SA	Apr 13, 2019 - Apr 18, 2019	Live Event
SANS Boston Spring 2019	Boston, MAUS	Apr 14, 2019 - Apr 19, 2019	Live Event
SANS Seattle Spring 2019	Seattle, WAUS	Apr 14, 2019 - Apr 19, 2019	Live Event
FOR498 Battlefield Forensics Beta 1	Arlington, VAUS	Apr 15, 2019 - Apr 20, 2019	Live Event
SANS FOR585 Madrid April 2019 (in Spanish)	Madrid, ES	Apr 22, 2019 - Apr 27, 2019	Live Event
SANS Northern Virginia- Alexandria 2019	Alexandria, VAUS	Apr 23, 2019 - Apr 28, 2019	Live Event
SANS Muscat April 2019	Muscat, OM	Apr 27, 2019 - May 02, 2019	Live Event
SANS Pen Test Austin 2019	Austin, TXUS	Apr 29, 2019 - May 04, 2019	Live Event
Cloud Security Summit & Training 2019	San Jose, CAUS	Apr 29, 2019 - May 06, 2019	Live Event
SANS Bucharest May 2019	Bucharest, RO	May 06, 2019 - May 11, 2019	Live Event
SANS Security West 2019	San Diego, CAUS	May 09, 2019 - May 16, 2019	Live Event
SANS Perth 2019	Perth, AU	May 13, 2019 - May 18, 2019	Live Event
SANS Milan May 2019	Milan, IT	May 13, 2019 - May 18, 2019	Live Event
SANS Dublin May 2019	Dublin, IE	May 13, 2019 - May 18, 2019	Live Event
SANS Stockholm May 2019	Stockholm, SE	May 13, 2019 - May 18, 2019	Live Event
SANS New Orleans 2019	New Orleans, LAUS	May 19, 2019 - May 24, 2019	Live Event
SANS Northern VA Spring- Reston 2019	Reston, VAUS	May 19, 2019 - May 24, 2019	Live Event
SANS Autumn Sydney 2019	Sydney, AU	May 20, 2019 - May 25, 2019	Live Event
SANS MGT516 Beta Two 2019	San Francisco, CAUS	May 20, 2019 - May 24, 2019	Live Event
SANS Amsterdam May 2019	Amsterdam, NL	May 20, 2019 - May 25, 2019	Live Event
SANS Hong Kong 2019	Hong Kong, HK	May 20, 2019 - May 25, 2019	Live Event
SANS Krakow May 2019	Krakow, PL	May 27, 2019 - Jun 01, 2019	Live Event
SANS Atlanta 2019	Atlanta, GAUS	May 28, 2019 - Jun 02, 2019	Live Event
SANS San Antonio 2019	San Antonio, TXUS	May 28, 2019 - Jun 02, 2019	Live Event
Security Writing NYC: SEC402 Beta 2	New York, NYUS	Jun 01, 2019 - Jun 02, 2019	Live Event
SANS London June 2019	London, GB	Jun 03, 2019 - Jun 08, 2019	Live Event
SANS Zurich June 2019	Zurich, CH	Jun 03, 2019 - Jun 08, 2019	Live Event
Enterprise Defense Summit & Training 2019	Redondo Beach, CAUS	Jun 03, 2019 - Jun 10, 2019	Live Event
SANS Kansas City 2019	Kansas City, MOUS	Jun 10, 2019 - Jun 15, 2019	Live Event
SANS 2019	OnlineFLUS	Apr 01, 2019 - Apr 08, 2019	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced