Security Issues in Mobile (Wireless) Ad Hoc Networking

Christopher Levari
Department of Computing
Macquarie University
Sydney, NSW, Australia
clevari@cox.net

Abstract

Technology and communication have rapidly evolved over the last decade to permeate all facets of our lives. We now use technology in our homes, our work, and at play. With the advent of these devices to fulfill our need, we've discovered that we now want these devices to communicate and share our information. Since we are a mobile society, these communications need to be wireless. This need has helped expand the field of wireless ad hoc communications (networking). Expanding in parallel is the need to secure these communications and keep our data private. Security in ad hoc networks is handled differently; therefore the security issues are different. This paper will define the main security issues in ad hoc networking using Bluetooth as a primary example of ad hoc technology. After which, this paper will offer some insight into these issues and how to combat them.

1 Introduction

Mobile ad hoc networks (MANET) are entirely self-configuring networks that may have any number of devices attached to them. Currently, ad hoc networks are deployed in military tactical operations, disaster management and rescue situations. In order to support the delivery and routing of critical applications as well as to meet the demands of next-generation business applications, security is vital in MANET architecture.

1.1 Goals

This paper will take you through some important topics in networking. Specifically, we will cover Bluetooth technology as it pertains to ad hoc networking. This paper will describe how the Bluetooth standard handles security and show some of the security issues within ad hoc networking. The three main security issues in ad hoc networking are: Key Management, Trust Management and Secure Routing.

1.2 Approach

The approach to the three main problems of ad hoc networking is to first examine current texts written on Key Management, Trust Management and Secure Routing in ad hoc networks. After examining the texts I researched a current technology used in MANETS so that we can learn how it fits with the main security issues and point out weaknesses and possible solutions to the problems..

1.3 Motivation

We will focus on Bluetooth technology instead of some of the others (i.e. 802.11x WiFi) because this is where technology is headed. Devices are more and more portable as the years go by. Portability is a key benefit of ad hoc networking, devices can join or leave a network on a whim. So, as you can see it is beneficial for portable devices to be ad hoc. The problem with portable devices is the lack of static power, so they run off of batteries. Bluetooth is designed to be low power, so it is ideal for portable devices.

2 Related Work

In this section we will cover some of the concepts behind the three main security topics in ad hoc networking. These ideas, concepts, and solutions are based on other professionals in the field as well as my personal understanding of the topics.

2.1 Key Management

The key management issue in ad hoc networking is that there is no centralized key management or distribution in ad hoc environments. Networks are created 'on the fly' so there is no server to handle the keys. Mishra says, *To be able to protect nodes against eavesdropping by using en-*

cryption, it is necessary that the nodes must have made a mutual agreement on a shared secret key or have exchanged public keys. (Mishra, 2008) Mishra has an interesting option for key management that he calls "distributed asynchronous key management service". "Asynchronous key" is the same as Public Key Encryption that I have discussed above. In Mishra's scheme every node in the network carries the public key (K), but the private key (k) is split up evenly between the nodes. Each node can encrypt the message with their piece of the private key (k1,k2, or k3, etc) as long as a certain number of nodes (the threshold) sign the message with their key piece; when the messages are fed through the "combiner" a completed signed message emerges. This message can then be decrypted with the public key and read

2.2 Trust Management

An important issue for ad hoc networks is an issue of trust. We need a method to tell whether a node/device has permission to join the network. Is that device trustworthy? Does it have permission to access the data over the network? According to IIyas, one way to establish trust in ad hoc networks is to use certificates and a master/slave framework to enforce the security policy. This method lends itself to Bluetooth piconets since there is always one master in the network. The master node changes in a round robin fashion, so that at some point every node has been the master/certificate authority. To have this method work properly an Administrator has to install the certificates on each device. Another method is to establish trust as the piconet is formed. As Node A joins with Node B they share keys and say they "trust" each other. Now as another Node "C" joins with Node B and shares its key and around to the other Nodes. Once the network is complete every node has everyone else's key.

2.3 Secure Routing

Secure routing is an important topic in ad hoc networking. Even if a Node does not read the data, does that Node have the appropriate trust and security capabilities to relay the data? This brings up the area of security aware routing. To facilitate secure routing ,some sort of metric is needed to identify device capabilities and trust level. For example, Trust Level- high, medium, low; and Security Capabilities- a metric assigned to a node based on whether the node can process any of the security mechanisms (like the ones I stated above). A security aware protocol would take these metrics into account when calculating routes. For instance, the header on the data packet has a requirement of high trust and a security capability requirement of 5 (meaning it needs to be able to do all of the mechanisms, encryption, timestamp, etc). So when a route discovery message is sent, the nodes give a route reply with each node's security metric. The protocol would then calculate the route only sending the data through nodes with a metric of "high, 5". It is important to note that the calculated route may not be the shortest route. Also, the device must have the processing power to calculate the route in a timely fashion.

3 Background

In order to relate the three main security issues to Bluetooth, we needed to research the Bluetooth specification. We have given a background of Bluetooth and its security mechanisms below.

3.1 Bluetooth

Bluetooth was created in 1994 by Ericsson as a replacement for short range ad hoc networks. (IIyas, 2003) These small short range networks are called piconets. In a Bluetooth piconet there are two types of nodes or devices attached to the network. These nodes are either a master or a slave; the piconet has one master and up to seven slaves. Two or more piconets create a scatternet. In order to connect piconets, a slave node in one network is shared between a second network and becomes the master in that second network. (See Figure 1 for Piconet topology.) All communication to the different nodes only goes through the master. Hearing this method, it sounds more like infrastructure architecture than ad hoc architecture. Bluetooth is more ad hoc than not, because the protocol is designed to allow nodes to join or leave the piconet with ease. The master node is nominated in round robin fashion based on a timer. If the master node leaves the network another node is nominated as the master node..

3.2 Piconets

A Bluetooth node is in either of two states: Standby or Connection. Standby mode means the device is waiting for a connection and not part of a piconet. Connection mode means simply that the device is connected to a piconet. To form a piconet the master transmits an ID packet over 32 of the 79 channels. Devices in the Stand-

by state periodically scan for this packet. If it hears it, the device sends its address and timing info to the master. The device then waits for the master to page it. When the master is satisfied that it has identified all the devices in its range it starts to form the piconet. It pages each device with its own device access code (DAC) using a frequency hopping sequence based on the slaves address. When the slave hears this it sends a confirmation packet. On the next slot the master sends the slave the master DAC. The slave then enters the Connection state. The master does this for all the slaves in the piconet then it enters the Connection state itself. (Dunne, Roche, O'Loghlin, Rhatigan, 2009).

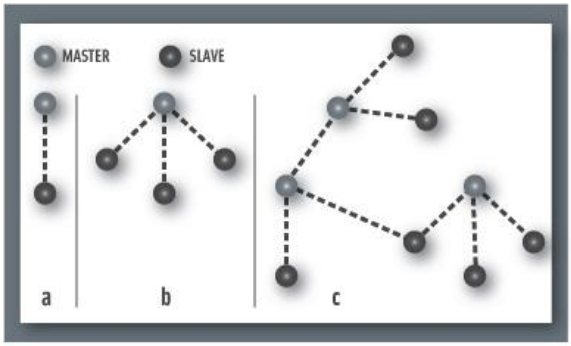

Figure 1: Piconets

3.3 Bluetooth Security

The Bluetooth (IEEE 802.15.1) specification has mechanisms built-in to handle authentication and encryption. In order to facilitate this, each device is given a "Bluetooth Device Address (BDA*)" at manufacture time. This 48 bit ID is akin to a network card's MAC address. This unique hexadecimal number identifies the manufacturer and a unique ID for each device. The BDA is part of the basis for key generation.

3.4 Keys

The Bluetooth standard has two categories of keys: Link keys and Encryption keys (Used for end to end communication). The link keys are used for just that, creating a link between two or more devices. Below are the link keys followed lastly by the encryption key.

Unit key, K_A, is created the first time device A is used. The Unit key is created by combining a 128 bit random number and the 48 bit BDA into an algorithm to create the 128 bit Unit key. This key is often used when device has little memory, instead of using/creating K_{AB}.

Combination key, K_{AB}, The first step to create this key is each device (A,B) create their respective Unit keys and then they exchange their keys and the random number that was generated. Each device combines the Unit keys of A and B with the two random numbers to create K_{AB}. This key is only used for communication between A and B. If device A wants to communicate and authenticate to C, a new combination key K_{AC} has to be created and stored. This key (128 bit) is generated for each pair of devices and is used when more security is needed. This requires more memory, since device has to store one combination key for each connection it has.

Master key, K_{master}, is used when the master device wants to transmit to several devices at ones. It over rides the current link key only for one session. The Master key (128 bit) is created by feeding two random numbers into the same algorithm used to create the initialization key. The algorithm can be found at (Standards, p. 458).

Initialization key, K_{init}, is used in the initialization process. This key protects initialization parameters when they are transmitted. This key is formed from a random
number, a passkey or PIN code, and the BDA of the initiating device.

Encryption key, K_E, Encryption key is derived from the current link key. Each time encryption is needed the encryption key will be automatically changed. The purpose of separating the authentication key and encryption key is to facilitate the use of a shorter encryption key without weakening the strength of the authentication procedure. (Standards, 2005).

3.5 Authentication and Authorization

Bluetooth has three different levels of security, which are called service levels. These levels are:

Service Level 1 - Requires authorization and authentication. Automatic access is granted only to trusted devices; untrusted devices need manual authorization.
Service Level 2 - Requires authentication only; authorization is not necessary. Access to an application is allowed only after an authentication procedure.
Service Level 3 - Open to all devices, with no authentication required. Access is granted automatically. (Standards, 2005)

Authentication is handled by a challenge and response mechanism. In this mechanism, Device A sends a random number to Device B. Device B then calculates a response by feeding the random number from A, its BDA*, and the key it has into

an algorithm. Device B then sends the calculated response back to A. Device A then calculates the response by feeding that same random number, the BDA of B, and the key into the algorithm. If the two responses match, Device B is authenticated and they both share the same key. (See Figure 2 for an authentication example.) If they do not match, Device B must wait a given time interval before trying again. This time interval increases exponentially with each attempt.

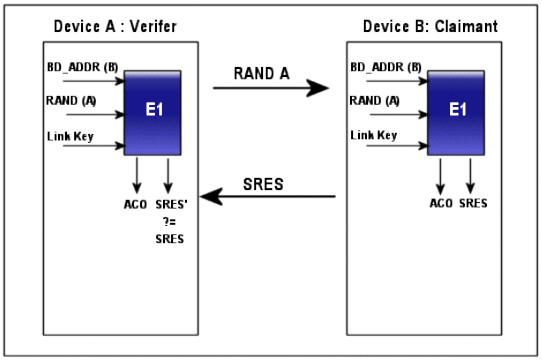

Figure 2: Challenge Response Mechanism

Bluetooth handles authorization very simply. If a device is authorized to access the content then it is known as "trusted", if not, then it is known as "untrusted". When a device is trusted it has been authenticated and its link key is stored.

3.6 Data Encryption

This Bluetooth data encryption is handled by a stream cipher. This cipher first calculates a "payload key" this key is generated using K_E, the BDA, the synchronization clock value, and a random number. After that the payload key is fed into the key stream with the data to be transmitted and out comes the encrypted text/cipher text. The receiving device then decrypts the data by feeding it back through the stream cipher (See Figure 3).

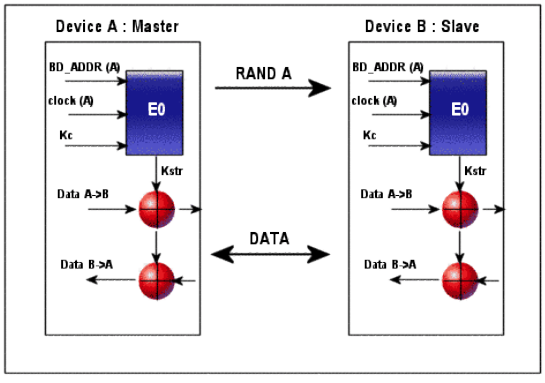

Figure 3: Data Encryption Mechanism

4 Implementation and Observation

Now that now that we have examined the technical details of Bluetooth and understood them, let's look at how it does or does not address the three main security issues of ad hoc networking.

4.1 Implementation Key Management

Bluetooth does handle key management efficiently since the keys for each device are stored only on a semi-permanent basis. This eliminates the problem of constantly authenticating for every communication attempt. The semi permanency of the keys helps alleviate the burden of storing all keys for every device even if that device never joins the network again. The problems start when first "pairing" the device. Bluetooth pairing is equivalent to joining the network. The new device pairs with the master node in the piconet and the initialization key is created once a PIN is shared. How does the PIN get shared? Most likely from a company Network Administrator. The device PIN has to be entered by the user or the PIN is installed on the device by the Administrator. This method is not "truly" ad hoc, as centralized management of the PIN/"key" is handled by the Admin. The other issue I see, is a big security issue. If the device uses its Unit key for communication it becomes a security threat. The Unit key is created for the life of the device, this means the same key will be used over and over for communication. The more times a key is used, the more it has to become compromised and used by a hacker. Also, if Device A is communicating with multiple other devices, for example Device B and C, then Device B could eavesdrop on Device C because they are both using Device A's Unit Key.

4.2 Implementation: Trust Management

The Bluetooth specification does handle the issue of trust but only on a very basic level. Either a device is trusted or it is not, there are no provisions for levels of trust such as top secret, secret, classified, or non-classified.
Bluetooth gets around the issue of trust by sharing a secret PIN. If the PIN is known and entered into the device, the device is trusted. This is a weak method of trust because the PIN could be compromised. If this happens then the proof of trust is void. This method also does not address the trust level of the user. Any user could randomly use a device where the PIN has already

been entered. Therefore some other layer of authentication which establishes trust needs to be implemented.

4.3 Implementation: Secure Routing

This paper will take you through some important topics in networking. Specifically, we will cover Bluetooth technology as it pertains to ad hoc networking. This paper will describe how the Bluetooth standard handles security and show some of the security issues within ad hoc networking. The three main security issues in ad hoc networking are: Key M Although Bluetooth does not have any mechanism for secure routing, ad hoc routing in general have some solutions. Take the AODV (Ad hoc On demand Distance Vector) routing protocol for example. This routing protocol is a reactive protocol because it only requests a route when needed. This is what meant by "on-demand". What is interesting is it doesn't require nodes to maintain routes that are not actively used. Routes are only maintained between nodes that need to communicate. (IIyas, 2003) This helps keep the route tables smaller. Next we will go through the steps of the protocol. When Node A wants to send data to Node B, Node A checks its route table for a route to Node B, if it does it sends the data to the next hop in the route. This process is repeated until the data reaches the destination. If the Node does not have a route to the destination, route discovery is initiated.

Route Discovery: When Node A needs route information it floods the network with a Route Request Message (RREQ). The RREQ packet contains the following: Source address, destination address, sequence number (keeps track of repeat messages), and broadcast ID. After flooding the network, Node A sets a timer to wait for a reply. When a Node receives a RREQ message it checks to see if has received a previous RREQ it checks the source address and ID pair, if it has, it discards the message.

It has been suggested by (Varadharajan, et al, 2009) to secure AODV by signing the data packets with a key. This method would work nicely with Bluetooth's Unit key for each device. Combining this with the capabilities model suggested in section 2.3 would go a long way to applying secure routing to the Bluetooth protocol.

5 Conclusion and Further Work

In order to address the key management issue, the Unit key must not be used for communication. Only use the Combination key. This will solve the security issue, because the Combination key is only used for the lifetime of the communication between two devices, therefore no other device besides that pair can read the data. It is also important to note that these keys only authenticate the device, not the user. To rectify this authentication must take place at the application level, requiring users to login to access the data. Further work on this topic will address trust mechanisms and secure routing in ad hoc networking with focus on Bluetooth. The goal will be to use some method to apply the general ad hoc security solutions to the Bluetooth specification.

Reference

Aggelou, George. *Mobile Ad Hoc Networks: From Wireless LANs to 4G Networks*. McGraw Hill: New York, 2005.

Biagioni, E. & Chen, S. *A Reliability Layer For Ad Hoc Wireless Sensor Network Routing*. University of Hawaii: Hawaii, 2004.

Davis, Carlton R. *A Localized Trust Management Scheme for Ad hoc Networks*. McGill Uni versity: Montreal, 2009.

IIyas, Mohammad. *The Handbook of Ad Hoc Wireless Networks*. CRC Press: New York, 2003.

Mishra, Amitabh. *Security and Quality of Service in Ad Hoc Wireless Networks*. Cambridge University Press: Cambridge, 2008.

Muller, Thomas. *Bluetooth Security Architecture*. Nokia: USA, 1999.

Papadimitratos, P. & Haas, Z. *Secure Routing for Mobile Ad Hoc Networks*. Cornell University: Ithaca, 2002.

Roche, E. & Dunne, K. & O'Loghlin, D. *Bluetooth For Ad Hoc Networking*. http://ntrg.cs.tcd.ie/undergrad/4ba2.05/group3/index.html Retrieved on 23 March 2009.

Scarfone, K. & Padgette, J. *NIST: Guide to Bluetooth Security*. NIST Special Publication: Gai thersburg, 2008.

Standards Committee. *IEEE Standard 802.15.1: Revi sion 2005*. IEEE: New York, 14 June 2005. (pp. 437-459)

Toh, C. K. *Ad Hoc Mobile Wireless Networks: Proto cols and Systems*. Prentice Hall: New Jersey, 2002.

Traskback, Marjaana. *Security of Bluetooth: An Overview of Bluetooth Security*. Helsinki: 2009.

Varadharajan, V. & Shankaran, R. & Hitchens, M. *Securing the Ad Hoc On-demand Distance Vector Protocol.* Macquarie University: Aus tralia, 2009.

Illustrations

Figure 1obtained at:
http://www.easycom.com.ua/data/netlan/712162057/img/piconets1.jpg

Figure 2 obtained at:
http://www.palowireless.com/bluearticles/cc1_security1_files/image002.gif

Figure 3 obtained at:
http://www.palowireless.com/bluearticles/cc1_security1_files/image003.gif