An Analysis of Current Mobile Mesh Network Applications

Computer Security - Tufts Comp116

Author: Daniel Griffin
Email: daniel.griffin@tufts.edu
Mentor: Ming Chow

Abstract

Wireless mesh networks(WMN) are becoming a growing presence in the mobile community for communication and sharing internet connection. This is because unlike current mobile systems, WMNs have reliable service coverage, are easily maintained, and have a low upfront cost. Although there are several advantages to WMNs, many security concerns arise as the data is expected to pass through multiple nodes which cannot be assumed to be friendly. This paper will examine some theoretical attacks on WMNs with a focus on mobile ad hoc networks(MANETs), and proposed ways to prevent these attacks. Following this examination will be an analysis of two current applications using MANETs, FireChat and the Serval Project.

Introduction

The importance of WMN has been highlighted in recent events do to government agencies monitoring internet traffic and Internet Service Provider's (ISP) modifying traffic such as the recent incident with Comcast and Netflix[11]. By using a WMN people are able to communicate with one another directly instead of through an ISP theoretically achieving increased privacy. Maintaining privacy and bypassing restrictions has already been shown as an important use case of WMN in Baghdad during the government cracking down on internet usage. In this period many citizens turned to FireChat to communicate in order to work around these restrictions[10]. Due to the sensitivity of this use case, security of these networks is of the utmost importance.

Network Structure and Terminology

Before examining the security concerns, we must first cover the structure of a WMN. The current specification for mesh network communication and topology is IEEE 802.11s[9]. The most important aspect in regards to this paper is a mesh point which supports a peer to peer link and can connect to its neighbors[9]. A neighbor refers to another mesh point within a distance which is determined using several different methods, although it is also common to treat all visible nodes as neighbors. In regards to data transfer, a hop is transferring data from one node to its neighbor, so to reach a node that is not a direct neighbor is a multihop path. A final note of importance is that the routing in a mesh network is typically done using MAC addresses instead of IP addresses[9].

To the Community

Do to the sensitivity of the information being passed with WMNs, it is import for the weaknesses of these networks to be highlighted. This information must be widely available for developers so that when utilizing this growing field, such as with Apple's new Multipeer Connectivity Framework[12], there is an understanding of the limitations of the systems. Thus it is possible to throttle the information passed through the network or utilize an external encryption method accordingly. Furthermore the users of these applications should have full understanding of the lengths needed to extract information from their messages when using an application and throttle the information they put through the application accordingly.

Mesh Network Attacks

i. Universal Network Issues

Some universal issues that networks and software face are session hijacking, traffic injection, logic errors, buffer overflows, privilege escalation, device tampering and denial-of-service attacks. While theses are sever security concerns, they are not specific to mesh networks and are out of the scope of this paper. Furthermore with an attack such as a denial-of-service, it “does not seem to be feasible to prevent [it] in a network that uses wireless technology"[8] and as such will not be discussed.

ii. man-in-the-middle attack (MITM)

A MITM is an attack in which an attacker will be an intermediary between two parties having a conversation with the intent of either altering the conversation or listening to the conversation. While a MINT can occur in a normal network setting it is an extremely prevalent issue in WMNs as it is expected that all traffic will pass through at least one third party. Thus all that differentiates a MITM from a WMN is the ability of the middle party to alter or listen to the conversation occurring.

In order to perform a MITM attack in a WMN the attacker will often attempt to deceive the network of its distance from other nodes. By counterfeiting a lower distance the malicious node increases its chance of being included in the information path and thus being able to listen to the conversation. One manner of determining the distance between two nodes in a WMN is time of arrival (TOA) in which the round trip time of a packet is calculated and assuming the packet has a direct path with a velocity of the speed of light, a distance can be calculated. This method has been shown to be insecure however, as a contacted node can modify the timestamp on the packet in a manner such as to convince the sending node it is at any given distance[3]. One proposed methods for preventing a node from lying about its distance is packet leashes which restrict packet travel to a defined geographical area. However, this requires gps or synchronized clocks which is difficult. Furthermore, in the case of civilian GPS, the signals used to calculate a location are unauthenticated signals and have been shown to be forgeable and thus insecure as an authentication method[1].

Another method for detecting a MITM attack is to use Rivest and Shamir’s interlock protocol in which half of the message is sent encrypted and the other half is sent unencrypted. The desired effect of this methodology is for the man in the middle to accidently double encrypt a message which would be a flag to the users that there is a MITM attack occurring[1]. The next proposed method for a stable WMN is the preshared key(PSK) method in which all initial nodes are given a phrase or key in advance. This methodology is not feasible within a MANET and furthermore open source tools (such as coWPAtty) which perform dictionary attacks in order to break these passwords are widely available[2]. Another method is to use a public key with a certificate authority, however joining the network can become difficult as there are

issues with the initial contacts with the certificate authority. Furthermore in MANET applications it may be difficult to have a trusted certificate authority[2].

A successful manner for detecting a MITM attack in a MANET is positive acknowledgement in which a node must respond within a timeframe that is below the amount of latency that is introduced by a MITM attack[1]. Another manner is to send information on agreed upon silent frames of communications, if a response occurs to one of these messages then a man in the middle is present[1]. A final method to mitigate the effects of the MITM attack is to use end-to-end encryption.

iii. Wormholes

A wormhole is a specialized MITM attack which connects two distant nodes in a network's topology such that the two nodes appear to be close. In this manner information traveling from one side of the network is likely to travel through the wormhole allowing the attacker to access the information. This can be forged by connecting to two nodes on disparate sections of the network and connectingly them privately through a separate, usually encrypted, network. A hardship with detecting wormholes is that the information as well as the transmission must be examined for authenticity[2]. As such the above Rivest and Shamir's interlock protocol is unable to detect a wormhole leading to the development of other methods.

One successful means of identifying this in a stable network is fingerprinting in which sudden deviations in signal strength and round trip times can be used as flags to detect an attack[1]. Additional manners include visualizing the network topology, watching for increases in the number of links for a station, and investigating higher latency hops as these all can indicate a potential wormhole. An issue with the above suggestions is that all work exclusively with static networks and as such are not portable to MANETs. While detection of wormholes in MANETS is still an active area of research the effect of a wormhole can be largely mitigated, assuming the wormhole relays messages quick, with the use of end-to-end encryption.

iii. Black and Gray Holes

Both a black hole and a gray hole work on the principle of attracting traffic to themselves and modifying the amount of output information. In the case of a gray hole the information that the node passes on is based on a condition while in the case of a black hole it merely will not pass on any information[2]. A reputation-based system can be used to successfully detect a black hole as the neighboring nodes can monitor if the node is passing on any information. In the case of a gray hole it can be more difficult to notice the issue as it will allow some information to pass through. In both cases a reputation-based model will not work if the neighbors cannot detect the signal or channel that the node is outputting[2]. This should not be an issue in the case of most MANETS, especially those examined in this paper as both use bluetooth and wifi which should be visible on most mobile devices.

iiii. Rushing Attacks

A rushing attack is an attempt to subvert the normal route discovery process in order to increase the chance of including the malicious node in a route[2]. One manner of preventing this is the Neighbor-verification protocol (NVP) which uses timing and power information to detect a rushing attack. The issue with an NVP is that is it not cryptographically secure[2]. Another methodology is to use Secure Ad Hoc On-Demand Vector Routing (SAODV) which uses hop counts protected by a hash chain in order to preserve the route discovery integrity[8].

With an understanding of some of the fundamental attacks that occur in the context of WMN this paper will analyze current applications utilizing MANETs for security concerns.

Applications

i. Applications doing it wrong: An analysis of FireChat

FireChat is an applications which allows users three manners of communication, all of which are text based. The first of which is everyone which relies on server communications to open a chatroom to all users called “everyone”. Another option is a “FireChat” which is similar to “everyone” but capped to 1000 people and has a topic of conversation[13]. Finally, the mode which will be examined is “nearby” which allows communication over wifi or bluetooth. Nearby is a “mode for off-the-grid communications, up to 200 feet of your location”[13]. Although FireChat is states that this communication mode is for off-the-grid communication the only security precaution taken is to warn users of the insecurity of the system. Christophe Daligault, Vice President of Sales and Marketing at Open Garden which produces FireChat, stated that “People need to understand that this is not a tool to communicate anything that would put them in a harmful situation if it were to be discovered by somebody who's hostile...It was not meant for secure or private communications”[10].

When examining the system it is very important to note that no encryption is used in the software, or plainly stated that all messages are passed in plain text[7]. Furthermore there is no authentication system meaning that a two contiguous messages from the same username may not be the same user[7]. While these are largely concerning issues a saving grace is that “there is also very little information leaked that could de-anonymize users”[7]. This is due to the fact that the unique identifiers passed with the messages are merely random strings that are unique to the message, not the user. However, another issue with sending the messages is that even when sending the messages by Bluetooth, if Wi-Fi is active all messages will be multicast unencrypted on the connected Wi-Fi[7]. Finally, if a physical device was acquired all of the messages sent and received, in all modes, are stored unencrypted on the device allowing the anonymous messages to be tied to an identity[7]. FireChat is open to a MITM attack as there is no encryption and the state of its vulnerability to the other issues is unknown. While FireChat is a very insecure system Open Garden

has been very open about these issues and could still be a good tool for messaging when security is not as much of a concern.

ii. Applications doing it right: An analysis of the Serval Project

The Serval Project is a project to create infrastructure independent mobile communications with a target application of disaster response and community resilience[4]. As sensitive information is extremely likely to be transferred, especially in the latter case, security is a major concern for the project. The structure of the network is such that a Serval ID(SID) is used for routing instead of IP. Furthermore the SID is also a public key in an Elliptic curve cryptography system[5]. Using this public key information is passed with end-to-end encryption to help mitigate MITM attacks. Furthermore as the public key is the SID, there is no need for key exchange. The Serval Project utilizes the Mesh Datagram Protocol which uses connections similar to UDP meaning that duplicate packets can be sent and packet delivery is not guaranteed[5].

One important aspect of the Serval Project is the Rhizome system for file transfers. The first aspect of Rhizome is a store-and-forward method in which all data that passes through is stored and forwarded to all neighboring nodes[4]. The data is passed in bundles which includes meta-data and the file as well as strong authentication and encryption of the payload. When transferring files, tampering can be determined by using a cryptographic hash[4]. Furthermore, the bundles can be set to auto-delete at a specified time[4]. In a streaming case, the bundles grow in size progressively, and smaller bundles are prioritized for sending which helps to preserve data order. This methodology is used to help send SMS-like text messages[4]. Furthermore, as the data is stored on the nodes, if a node is moved it will broadcast the message to all new nodes at its destination[4].

When trying to contact a to a new node the only information available is a phone number referred to as a DID. In order to prevent a MITM attack, a secure method of mapping DID to SID is needed[6]. In order to perform this a specific verification secret (SSVS) is needed which visualizes a random number which is seeded with a number generated from public and private keys. Both screens will display the generated image and the users are able to verify if the images are identical in person. If this is the case then a permanent mapping can be stored and used for future reference[6]. This method can also be replicated over a live phone call in which random words are pulled from a wordlist and the users can verify the same words occur on the screen in the same order[6]. An important aspect of the Serval Project is that the user's private key is stored partially on the SD card and partially on the local machine. This is an attempt to prevent the loss of the private key if the SD card is acquired[6]. As a final note of a security feature is that a third party is unable to indicate what SID created a given bundle[6].

While the Serval Project has several strengths it is important to acknowledge its weaknesses. It is of further importance to note that the following is not a complete list of weaknesses and that many others are noted in "A Framework for a Robust

Architecture Offering Pervasive Security in Isolated and Infrastructure-Deprived Networks" by Paul Gardner-Stephen. The first issue is that real-time point-to-point links difficult in large mesh. Another issue is that if a large bundle, such as a video is being sent to a single user the store it forward methodology is inefficient as storage capacity is $O(n^2)$[4]. While these are implementation concerns, the main security concern arises with the used cryptographic method which currently does not have a mathematical proof of its strength. It does however claim that its 256-bit keys offer a comparable security strength to a 3,072 bit key in the RSA cryptographic system[6].

Conclusion

Although mesh networks are an emerging method of communication it is important to consider the security concerns when passing messages. Even though these networks are relatively new to the common user it has quickly become apparent that they will be used extensively to spread sensitive material. As such it is the onus of developers to fully understand the security risks of the networks they are using whether it be working on Project Several, using Apple's Multipeer Connectivity Framework or connecting with Open Garden's P2P API. Upon understanding the strengths and limitations of the select method it is essential to clearly and concisely convey this to the user such as occurred with FireChat's disclosure so that the community can be conscious of what information is private and what is public.

References

1. Stephen Glass, Vallipuram Muthukumurasamy, and Marius Portmann. *Detecting Man-in-the-Middle and Wormhole Attacks in Wireless Mesh Networks*
    i.
2. Stephen Glass, Vallipuram Muthukumurasamy, and Marius Portmann. *Securing Wireless Mesh Networks*
    i.
3. Stephen Glass, Vallipuram Muthukumurasamy, and Marius Portmann. *The Insecurity of Time-of-Arrival Distance-Ranging in IEEE 802.11 Wireless Networks*
    1.
4. Dr. Paul Gardner-Stephen.*Serval Rhizome Store-and-Forward & Serval Mesh Datagram Protocols*

5. Bettison, Andrew. "Mesh Datagram Protocol (MDP)." *The Serval Project Wiki*. Serval Project, 5 Feb. 2014. Web. 26 Oct. 2014. <http://developer.servalproject.org/dokuwiki/doku.php?id=content:tech:mdp>.

6. Gardner-Stephen, Paul. "A Framework for a Robust Architecture Offering Pervasive Security in Isolated and Infrastructure-Deprived Networks." 10, December 2012.

7. Dranka, Andrei, Jakub Dalek, Adam Senft, and Philip Winter. "Iraq Information Controls Update: Analyzing Internet Filtering and Mobile Apps." *The Citizen Lab*. Citizen Lab, 24 July 2014. Web. 26 Oct. 2014. <https://citizenlab.org/2014/07/iraq-information-controls-update-analyzing-internet-filtering-mobile-apps/>.

8. M. Zapata, N. Asokan. *Securing ad hoc Routing Protocols*

9. *"IEEE 802.11s." Linux Wireless. Web. 8 Dec. 2014. <http://wireless.kernel.org/en/developers/Documentation/ieee80211/802.11s>.*

10. Christophe Daligault (as cited in Baraniuk, Chris. "FireChat Warns Iraqis That Messaging App Won't Protect Privacy (Wired UK)." *Wired UK*. 25 June 2014. Web. 9 Dec. 2014. <http://www.wired.co.uk/news/archive/2014-06/25/firechat>).

11. "Netflix Performance on Verizon and Comcast Has Been Dropping for Months." *Ars Technica*. 10 Feb. 2014. Web. 9 Dec. 2014. <http://arstechnica.com/information-technology/2014/02/netflix-performance-on-verizon-and-comcast-has-been-dropping-for-months/>.

12. "Multipeer Connectivity Framework Reference." *Multipeer Connectivity Framework Reference*. Web. 9 Dec. 2014. <https://developer.apple.com/library/ios/documentation/MultipeerConnectivity/Reference/MultipeerConnectivityFramework/index.html>.

13. "FireChat." *App Store*. Web. 10 Dec. 2014. <https://itunes.apple.com/us/app/firechat/id719829352?mt=8>.