Vulnerability Analysis and Countermeasures for WiFi-based Location Services and Applications

Jun Liang (Roy) Feng and Guang Gong
Department of Electrical and Computer Engineering
University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada
{jlfeng,ggong}@uwaterloo.ca

Abstract—Wireless LANs, or WiFi, are very common in any household or business today. Their wireless nature allows mobility and convenience for the user and that opens up a lot of new possibilities in mobile devices such as smartphones and tablets. One application that makes use of wireless LANs is positioning, which can be used in areas where Global Positioning Systems may have trouble functioning or not at all. However, a drawback of using wireless communication is that it is susceptible to eavesdropping and jamming. Once the wireless signal is jammed, an attacker can set up fake access points on different channels or frequencies to impersonate a legitimate access point. In this paper, this attack is performed specifically to trick WiFi-based location services that are becoming very popular nowadays, especially for mobile devices. Our paper shows that the attack works on Skyhook, Google, Apple and Microsoft location services, four of the major location service providers, and works on dual-band hardware. We also present some potential countermeasures to such an attack.

Index Terms—WiFi, location, location service, positioning, attack, jamming, impersonation.

I. INTRODUCTION

MOBILE devices are now a big part of our life thanks to wireless LANs, also known as WiFi. WiFi is a technology that allows devices to communicate wirelessly [13], and the most common use is accessing the Internet. The use of wireless technology allows us to access the Internet wherever we are, whether it’s on a laptop at a coffee shop, a tablet at home or a smartphone at a mall. This opens up many new types of applications that were unimaginable using wired network connections. One of these new applications is using WiFi to get the location of the device. The location of the device provides context information that enables new classes of service to the user.

However, bringing wireless technology into all of our devices creates big security problems. Due to the fact that data is being transmitted over the air, anyone can snoop in on the transmissions if it is not encrypted. Plus, all of the problems related to wired technology also applies in the wireless world, so there is a lot of research on securing wireless communications, like in [23], [37], [41], [45].

On the physical layer, jamming attacks can be carried out which involves an attacker sending noise signals that makes it hard for legitimate users to decode their own signals. For wired networks, an attacker would need access to the physical network in order to carry out the attack, but since data is transmitted over the air in wireless networks, no such restriction exist. An attacker can send the attack signal freely and anyone in range of the signal will be affected. This can have many implications on services that organizations have build that rely on wireless communications. In general, jammers jam the control signals which do not use any anti-jamming protection mechanisms in practice [23], [32], [37], [41], [45]. The research on anti-jamming mechanisms have recently looked at using cryptographic related methodologies and approaches, see [38], [39], [46], [47], just to list a few.

In the WiFi case, one such service that is affected is that is very popular today is location services.

Location services, as the name suggests, allows a device to get its location. Traditionally, when one thinks about a device that can get its location is one that uses the Global Positioning System, or GPS [16]. GPS devices is mainly used in cars to provide directions from one place to another. It can achieve this because satellites are launched into space, and these satellites together can see the whole world. Signals are sent from GPS devices on the ground to satellites that are in range and replies are received and calculations are performed to determine the exact position in the world. Due to the use of satellites, GPS has a number of limitations, and one of them that has people looking for alternate methods of getting location is that it does not work indoors. WiFi-based location service is supposed to solve that problem. Plus, the nature of WiFi means that one should be able to get a more accurate measurement of a device’s position. Combined, WiFi-based location service enables brand new applications that were not possible with GPS-based location service. However, this also means more services are compromised if the underlying WiFi infrastructure is compromised. We present our implementation of such attack is illustrated in this paper. This attack was conducted on four major location service providers: Skyhook, Google, Apple and Microsoft. GPS can also be compromised in a similar way as demonstrated in [21]. An analysis of the requirements of carrying out a successful GPS spoofing attack was conducted in [39].

Our contributions in this paper are the following:

• We implemented a more powerful attack on location services than known previous methods. The attack has the following improvements:
  – The attack not only compromises location services but also WiFi infrastructure with working rogue access points, which increases the threat dramatically.
We showed that the attack works on four of the major location service providers on the market today, and they are: Skyhook, Google, Microsoft and Apple.
- We analyzed the impact on many location-enabled applications.
- We noted and warned that the attack can very likely also work on Bluetooth location services.
- We proposed several potential countermeasures to our attack. These countermeasures mostly require a simple software update on the client so that it is feasible, and this can be enhanced with updates on the access point and server.

The rest of the paper is organized as follows: Section II provides related works; Section III gives background information on WiFi and WiFi-based location services; Section IV gives a vulnerability analysis of four major location service providers: Skyhook, Google, Apple and Microsoft; Section V shows the attack environment, procedure and results from the attack; Section VI analyzes the impact of attack on location services and various location-enabled applications; Section VII talks about the impact of the attack on WiFi infrastructure; Section VIII explains some of the potential countermeasures; and lastly Section IX concludes the paper.

II. RELATED WORKS

In [40], the authors performed an attack on WiFi-based location services using simulated fake access points. The authors used a software defined radio (USRP) to send a uniform noise signal to jam some WiFi bands. In the environment that the authors worked in, the legitimate access points occupies channels 6, 10 and 11, which are all channels in the 2.4 GHz band, and they were all jammed with the uniform noise signal from the USRP. For impersonating access points, the authors wrote a script using a program called Scapy running on a Linux computer. The script listens for client probe requests that asks for the available WiFi access points in the area, then sends responses corresponding to access points that are in a far away location. This information is freely available from WiGLE, also known as the Wireless Geographic Logging Engine [10], which is a service that tracks WiFi access points and their locations. The fake access point uses channel 2. The authors performed the attack on a first generation iPhone that uses Skyhook location services, and laptops using the Loki browser plugin, and was able to get the location service to return a false location that is based on the fake access points. In addition, the authors also conducted the experiment with GSM-based positioning, which showed that when WiFi positioning fails, the iPhone uses GSM-based positioning to calculate its location, but GSM-based positioning provides a much less precise location, which renders the applications that use location services useless.

In [14], the attacker used a slightly different method to attack WLAN-based location services. Without jamming the existing access points, the attacker created many fake access points to trick the location service client. So, it does not matter what channels the legitimate access points are operating in because no jamming is needed. In this setup, the attacker had to create many fake access points. The attacker estimates that in order to perform this attack, he would have to create double the amount of legitimate access points in the area in one case. In the two examples that the attacker gave, in a less WiFi-dense area with 6 legitimate access points, he had to create around 20 fake access points, and in a denser area with 25 legitimate access points, he had to create around 65 fake access points to make it work. The attacker used access point information from WiGLE [10] which is the same database used in our own experiment. The hardware used is actually very cheap, with just a Raspberry Pi and a USB WiFi card, which cost around $100 in total with power supply and SD card for storage. The software used are Aircrack-NG and MDK3, which are software dedicated to cracking and manipulating WiFi. The attacker saves the access point information he gets from WiGLE into a text file and launches the software to advertise that information to its surroundings.

In [21], the authors constructed a portable civilian GPS spoofer. The authors note that there was concern with potential vulnerability of civilian GPS receivers if their signals are spoofed as early as the early 2000s. Some examination of spoofing was done and countermeasures were proposed. Amplitude discrimination and time-of-arrival discrimination involve changes to software but is only effective against the simplest attacks. Consistency of navigation inertial measurement unit cross-check, polarization discrimination, and angle-of-arrival discrimination require additional hardware. A portable receiver-spoofing would defeat most of these countermeasures, which is what the authors concentrated on. The portable GPS spoofer consists of a receiver, which, as the name suggests, receives the legitimate GPS signals and the spoofer, which sends the fake GPS signals. The spoofer takes inputs from the receiver module, giving the start times of the kth C/A code period, estimate of beat carrier phase, estimate of the Doppler frequency shift, estimate of the signal amplitudes and the receiver-spoofing’s current 3D position and velocity. The spoofer tries to align the code phase then increase its amplitude, and at which point the target should be under the control of the spoofer.

III. BACKGROUND

This section gives some background on WiFi and WiFi-based location services. The way WiFi works is that it broadcasts or advertises its name and unique identifier and clients can get that information and connect to the access point. WiFi-based location services utilize that information to create a database and associate a location to it.

A. WiFi

WiFi infrastructure generally involves client devices, a WiFi access point, and a router to connect to the Internet [13]. A
client device associates with the access point and all communications from the client device passes through the access point. Access points broadcast a beacon message periodically to advertise the name of the network, which is the Service Set Identifier (SSID), and its fixed address, which is the MAC address of the WiFi radio interface. Clients would scan the airwaves to see available access points and can optionally send probe requests. The user would choose the network based on the name to connect to, enter authentication credentials and authenticate itself with the access point. After authentication is complete, the user can access the network. In subsequent connections, the device would connect to that access point with the same SSID but not necessarily MAC address without user interaction as it believes it is connecting to the same access point.

In a lot of new devices on the market today, WiFi cards are dual-band, meaning they can operate on 2.4 GHz frequencies or 5 GHz frequencies but not both. For the 2.4 GHz band, channel one operates at a centre frequency of 2.412 GHz, and channel eleven operates at a centre frequency of 2.462 GHz [3]. Each channel occupies a bandwidth of 20 MHz, so the whole 2.4 GHz band usable in North America has a total bandwidth of 70 MHz. The close centre frequencies and a 20 MHz bandwidth means the maximum number of channels that can be used without interference is three, namely channels 1, 6 and 11 [3]. The channels in the 2.4 GHz band and their spacing are shown in Figure 1 and the non-overlapping channels are shown in Figure 2. For the 5 GHz band, there are many more channels available, and their separation is spaced out so that they are non-overlapping so there are a lot more channels in the 5 GHz band that can be used without interference. The non-overlapping channels in the 5 GHz band are given in Figure 3. Notice that the 5 GHz band has twenty-three channels that are non-overlapping, which is way more than the three in the 2.4 GHz band.

B. WiFi-based Location Services

WiFi-based location services uses WiFi information to provide location information to devices. Information on WiFi access points, including their SSID and MAC address, is collected by the WiFi-based location services companies, usually using vehicles that is equipped with devices that can collect that information. Many of these vehicles drive around in towns and cities around the world so that it can collect as much WiFi access point information as possible and build a database with that information. The process of determining a device’s location is illustrated in Figure 4. In the diagram, access points are denoted as APs, user device is denoted as UD, and the location service is denoted as LSP. In step 1, the client broadcasts probe requests to ask for surrounding access point information. In step 2, access points that receive this request reply with its information, including its SSID and MAC address. In step 4, the client collects all the responses, along with signal strength information, and send them together to a WiFi-based location services company over the Internet. The servers at the service provider will process that information by looking up the access point information in its database, make calculations on the position, taking into account the signal strengths, then in step 5, return a position with longitude and latitude information that corresponds to what it believes to be the device’s location.

IV. Vulnerability Analysis of Location Services Providers

In our experiments, we focused on four location services providers: Skyhook, Google, Apple and Microsoft. Skyhook was founded in 2003 and is one of the earliest companies to offer WiFi-based location services. Skyhook location services was used on Apple iPhone, iPod Touch and iPad until iPhone OS software release 3.2. It is also used in numerous other smartphones. Skyhook also has a browser plugin that allows
any computer that can use the plugin to determine its location using WiFi location services. Google’s location services is used on Android operating system-based devices, and in Firefox and Chrome browsers. Apple’s location services is more recent, and is used by all of Apple’s current generation devices. Their location service is built into their Mac OS X operating system that runs on computers, and iOS operating system that runs on their mobile devices such as the iPhone, iPad and iPod Touch. Microsoft’s location services is built into their operating systems, Windows and Windows Phone.

To see what kind of information that needs to be sent to determine a location, a proxy was used to capture the traffic between the client and the location services server. Since all of the services use SSL, it was not possible to use just Wireshark to sniff the packets as that would only show the encrypted traffic. So, a tool called sslmeat [1] was used. sslmeat is an SSL proxy. It works like an HTTP proxy in that it sets up two connections to establish communications: one between the client and the proxy and one between the proxy and the destination server. However, in addition, the client needs to trust proxy by trusting the proxy’s server certificate in order for the SSL proxy to work. Then, when a client needs to access the Internet, traffic is first passed from the client to the proxy, encrypted using the proxy’s public key. The proxy decrypts the data, encrypts it again using the destination server’s public key and forwards it to that server. The fact that the proxy can decrypt the client’s data allows analysis of the traffic protected by SSL.

The captured information shows the format of the server queries and what exactly is being sent to these servers. For Skyhook, the query is an HTTP POST request in XML format. The MAC address and signal strength of surrounding access points are sent to Skyhook’s servers and a sample query is shown in Figure 5. For Google, the query is an HTTP POST request, with the MAC address and signal strength of each access point sent to the server, and a sample query is shown in Figure 6. For Apple, the query is an HTTP POST request, and it includes all the MAC addresses in the surrounding area, and a sample query is given in Figure 7. For Microsoft, we were not able to determine the format and information sent to their servers but according to [12], Microsoft location service query is a SOAP request that returns the location of a MAC address in XML format.

The information captured here shows that the only uniquely identifiable information collected is the MAC address. The signal strength is just a number that one can arbitrarily choose from the range of acceptable signal strength values. So, only the MAC address needs to be spoofed to impersonate an access point.

V. DUAL-BAND ATTACK ON WiFi-BASED LOCATION SERVICES

A. Adversary Model

In our experiments, the environment is a residential area consisting of houses separated only by a few metres or two backyard lengths. When a scan is done on the airwaves to see available WiFi access points, around ten access points can be

```xml
POST /wps2/location HTTP/1.1
Accept: */*
User-Agent: WPS API (3.4.2.28)
Accept-Language: en-us
Accept-Encoding: gzip, deflate
Host: api.skyhookwireless.com
Content-Length: 182

DNT: 1
Connection: Keep-Alive
Cache-Control: no-cache
Content-Type: text/xml

</LocationRQ>

Fig. 6. Google location services query

```
To carry out the attack, we need to do two things: (1) jam all legitimate access points so that the location services clients cannot determine its location and (2) set up fake access points so that the location services clients will use incorrect information to determine its location and thus get the wrong location. Procedure 1 gives the steps for the attack.

### Procedure 1 WiFi-based location services attack

1. Use GNU Radio and USRP to send a White Gaussian Noise signal centred at 2.422 GHz to jam about half of the 2.4 GHz band
2. Use GNU Radio and USRP to send a White Gaussian Noise signal centred at 2.452 GHz to jam the rest of the 2.4 GHz band
3. Use WiGLE.net’s database [10] to get the MAC address of access points (more than 2, the more the better) anywhere in the world
4. Set up Internet connection and routing on the embedded system running OpenWRT
5. Set up impersonated access points using these MAC addresses on a 5 GHz channel
6. Connect each location services client to any of the impersonated access points to get an Internet connection so that they can make queries to their respective location services server
7. Have each location services client determine its location

---

**Fig. 7.** Apple location services query

---

**Fig. 8.** Location services attack procedure

---

A diagram of this attack is given in Figure 8. Again, the access points are denoted as APs, user device is denoted as UD, and location service provider is denoted as LSP.

The attack was performed on the four WiFi-based location service providers mentioned in section IV: Skyhook, Google, Apple, and Microsoft. For all of the WiFi-based location service providers, the first step of the attack was successful, that is, jamming all legitimate access points resulted in the device not being able to get a location, thus denying the service. For the second step of the attack, the attack was also successful on all four location service providers as they all displayed the location of the spoofed access points, which is a location in Bellevue, Washington, U.S. in this case and the attack was conducted in Toronto, Ontario, Canada. The result of this step is shown in Figure 9 for Google, Figure 10 for Skyhook, Figure 11 for Apple, and Figure 12 for Microsoft.

To summarize, in Section IV, we first conducted an analysis of some of the location service providers. This analysis showed
Fig. 9. Access point impersonation attack on Google location service

that the unique identifiers used in determining location is the MAC address of access points. While the signal strength is also sent, that is more dependent on the radio hardware and the distance the client is from the access point. The client determines the signal strength when it scans for surrounding access points so it does not need to be spoofed. Next, in Section V-A, we talked about the environment that the experiment was conducted in, which is a residential area consisting of many houses. A scan at one home in the residential area shows around six broadcasting access points spread out over the 2.4 GHz. No 5 GHz band access points were available. In addition, the hardware and software used in the experiment was revealed, which consisted mainly of computers with Ubuntu Linux to control the USRP software defined radios, an embedded system for creating fake access points and several client devices to test each location service provider. Lastly, in Section V-B, we outlined the procedure of the attack and showed the results of our attack, which was successful for all four major location service providers tested.

VI. IMPACT OF ATTACK ON LOCATION SERVICES, LOCATION-ENABLED APPLICATIONS AND BLUETOOTH

With the gaining popularity of location services, there are a lot of applications that are affected by such an attack. This section provides an analysis on some of those applications and an analysis of cost and how it compares to a previous attack on WiFi-based location service in [40]. Depending on the application, the effect of such an attack would range from being a minor inconvenience for the user or disrupt major services.

A. Photo geotagging

Photos contain a lot of metadata that can be used by photographers to perform processing. These kinds of information include the date and time, ISO, aperture size and lens. Photos
can be tagged with GPS coordinates that shows exactly where the photo was taken. This information can be useful to the user if they forgot where the picture was taken or possibly want to make a video based on the location information like in Apple’s iPhoto that generates a video based on that information [6]. If such information contained in the photo is missing or wrong, then it would not be detrimental to the user, but would just cause some minor inconvenience.

B. Social network geotagging

When people make a post on a social network such as Facebook and Twitter, they have the option of providing a location with the post. Location services information would possibly add some value to the posts but it is not critical to the operation of the social networks so having a wrong location displayed would only be a minor annoyance. In extreme cases, a person can see the wrong location of a post made by a significant other and start questioning where that person is, but that is not a technology issue. Other than posts, services like Foursquare allow people to “check-in” to places they are at or have been (for example, a restaurant) based on location information [17]. Having wrong location information would cause minor inconvenience to the user as the user still has the option of manually inputting that information into the check-in.

C. Location-based weather

Weather applications can now provide weather information based on the location of the user, such as the application from the Weather Network [44]. This is a very convenient feature that allows a user to check local weather no matter where he/she is as long as he/she has a data connection. Without location information, the user is forced to enter that information manually, which is a minor inconvenience.

D. Locating friends

There are a number of services on the market that allows a user to share his/her location to those he/she gives permission, such as echoecho’s location sharing service [15]. The whole service relies on location information so having no location information, or worse, having wrong location information, would make the service worse than useless.

E. Device recovery services

With the increasing use of mobile devices, the amount of thefts have gone up in many cities [28] [42] [36] since people carry them wherever they are. Also, mobile devices are small, so people can easily lose them, or forget them, for example, at a coffee shop. A number of device recovery services have been created to solve that problem, including Orbicule’s Undercover service [30]. These services generally works as follows: an application is first installed on the device that a user wants to protect; then, when a device is lost or stolen, the user would report that to the service provider; the application that is installed on the device frequently checks the service provider to see if the device is reported lost or stolen and begins transmitting information such as screenshots and location to the service provider; then the service provider attempts to recover the device for the user, for example, by contacting law enforcement in that area. It is clear that location information is key to recovering the device, and that information is used to call law enforcement so missing location information would make recovery harder, but wrong location information can cause law enforcement to go to the wrong place, which can impact innocent people.

F. Passbook

Passbook is an application on Apple’s iOS 6-based devices that allows users to put tickets, membership cards, coupons and more in one place [8]. This application can be location enabled so that, for example, when a person is in a store at which he/she has a membership card, the card will pop up on the lockscreen of the device to provide easy access to the card. Another example is people going to the movies with tickets in Passbook, and Passbook will automatically show a ticket available on the lockscreen to provide easy access to the ticket. If location information is missing or wrong, then the user still has the option of unlocking the device, opening Passbook, then select the membership card, or ticket or coupon that he/she wants, so it is just a minor inconvenience.

G. Location-based reminders

Smartphones nowadays have reminders that can be based on location, like Apple’s Reminders application on iOS 6 devices [7]. This works by setting up a “geofence” around the area in which the user wants a reminder, and he/she can set the smartphone to remind them of something when they leave or enter this area. This can be used in conjunction with a time-based reminder. If location information is missing or wrong, then the reminder will not work, and a user relying on such reminder will miss the reminder, which can have big consequences depending on the reminder.

H. Search

One feature that has been added to search engines is the ability to find places nearby like Google [20]. This can be very convenient to a user who wants to find, for example, the closest store that is open twenty-four hours. If location information is missing or wrong, then the user would need to input location information manually, which is a minor inconvenience.

I. Bluetooth and Bluetooth-based location service

Bluetooth operates in the same 2.4 GHz band as WiFi, so Bluetooth suffers the same issue as WiFi with respect to jamming. Also, Bluetooth only uses the 2.4 GHz band, so it is worse than WiFi in that it only uses one band instead of two in the WiFi case, which means it is easier to attack Bluetooth than WiFi. The impact of jamming Bluetooth is a little different than jamming WiFi. In WiFi, jamming would mainly affect the traffic to and from the access point and client devices since that is the main use of WiFi. Ad-hoc traffic will also be affected. In Bluetooth, different types of traffic is
affected. Bluetooth is mainly used in short range applications [4] so devices like mice, keyboards and headsets would be affected when a jamming attack is performed on Bluetooth, and we experienced such an issue while we were performing our experiment when our computer was not able to register any movements from the Bluetooth mouse that was paired with the computer.

One area of development in recent years is Bluetooth Low Energy or Bluetooth Smart [4]. This is a specification in Bluetooth version 4.0 that allows devices to communicate with very low energy, thus decreasing energy use and increasing battery life on mobile devices. There are many applications of this new specification, and a lot of companies have already come up with many devices to take advantage of the low power consumption, and many Bluetooth profiles have been created for the various types of applications. Companies have already started taking advantage of the low-power Bluetooth specification and have come up with products such as fitness trackers [22] and smart watches [25] that are connected to one’s smartphone. These Location services based on Bluetooth have also been proposed like in [11], [19], [34]. The combination of Bluetooth Low Energy and location services can potentially provide a very low-power indoor positioning system. However, as mentioned above, Bluetooth is also susceptible to jamming, and since WiFi and Bluetooth share a lot of the 2.4 GHz spectrum, jamming of WiFi also affects Bluetooth, allowing an attacker to disable both location services.

VII. IMPACT OF ATTACK ON WiFi INFRASTRUCTURE

The attack carried out in this experiment not only affected location services but also WiFi infrastructure. With jamming, existing access points are blocked and clients are denied access. With access point impersonation, unsuspecting users may be tricked into connecting to an access point controlled by the attacker and all the traffic going through that access point will be accessible to the attacker.

Our experiment mainly focused on WiFi-based location services. However, there is one big difference between the implementation of the attack in this experiment than in [40] that makes our implementation more powerful. In [40], the authors wrote a custom script using a program called Scapy to impersonate the access points. Scapy is a tool that allows users to easily create various packets or frames, including 802.11 frames. With the proper WiFi card and modified drivers for the Linux platform, the user can use packet injection to send these custom frames. The authors in [40] wrote a script that listens to requests from clients asking for surrounding access point information. When it receives such a request, the script would create a fake response packet and sends it to the client. This script is limited in that not many services other than location services scans for surrounding access points so the use of this script in other attacks is limited. In our implementation, the fake access points are access points that are like any other access point with its own SSID and MAC address. Clients can connect to these access points, get an IP address and can communicate with others on the network and use the Internet. In [40], when clients actually try to connect to these access points, it would fail because the script was not written to accept clients, communicate with others on the network, or route traffic to the Internet, making that attack only suitable for location services. In our implementation, the fake access points can take over existing, legitimate access points and users would not notice. In the implementations in various operating systems [9], [26], the WiFi network manager remembers the SSID of various access points that the user connects to and will attempt to reconnect when it sees that SSID. This can be a big issue since a lot of users go to coffee shops or restaurants, for example, that have free WiFi. An unsuspecting user may connect to the fake access points and not realize that it is doing so, and all of the traffic can be collected by an adversary and the privacy and confidentiality of the user would be compromised.

Another important consequence of using actual access points for the attack is that it is much easier to set up fake access points than to write a script to send out broadcast beacon packets. A person with malicious intentions can easily set up fake access points on a Linux device with the right hardware, or get a router and put one of the several supported custom firmwares that are freely available on the Internet [2] [27] [29]. Since router setups are fairly user friendly with a web interface, it does not take much technical knowledge to set up a fake access point. However, in the attack in [40], the authors used a Scapy script, which requires knowledge of the language used to write Scapy programs and some knowledge of the parameters of a broadcast beacon packet. In order to use Scapy, the attacker needs to get a compatible WiFi card, plus install a special driver or patch an existing driver to support packet injection, which takes even more effort and technical knowledge. So, our attack is much easier to implement, and users can carry out the attack with much less technical knowledge.

The cost of the experiment consist of two things: the cost for jamming, and the cost for access point impersonation. For jamming, we used USRPs and the total cost for these USRPs with daughterboards and antennas is thousands of dollars. They were used since they were readily available in our lab. However, USRPs are software defined radio. Their purpose is not specifically to jam WiFi but rather a platform for people to build and test radio enabled applications among possible other uses. There are specialized hardware out there on the market (possibly illegal) that is designed specifically for jamming WiFi and other wireless devices that operate in the 2.4 GHz for around a few hundred dollars or less [5], which is inexpensive, and require little technical knowledge to operate. For the people who have more technical knowledge, there are video tutorials (e.g. [24]) and open source designs for WiFi jammers (e.g. [18]) that are freely available with a simple search on the Internet and the attacker would just have to buy the parts and put them together. So, an attacker can perform jamming without much technical knowledge and at a relatively low cost. These devices can be very small, which means they can be hidden, making them hard to find.

For access point impersonation, there are a number of ways to achieve this. We used an embedded system with a WiFi card in our experiment because this hardware was readily available
to us. Potential attackers can use the hardware that they already have, or buy new. Hardware like off-the-shelf routers or WiFi cards for desktops can all work. The requirement for these WiFi hardware is that they support customizable MAC address, and be able to act as an access point. To broadcast multiple access points, one can have multiple WiFi hardware or optionally a WiFi card or router that supports multiple wireless networks, which would definitely be more cost effective than having multiple hardware. The WiFi card in the embedded system we used for our experiment supports multiple wireless networks and was specifically picked for this reason.

Together, to perform an attack, an attacker would only need to purchase several hundred dollars worth of equipment. For those who have more technical knowledge, they may already have the hardware to perform the attack or they can follow the tutorials and build their own, which brings down the cost dramatically. So, a jamming and access point impersonation attack can be easily achieved.

VIII. COUNTERMEASURES

In this section, we present a number of countermeasures. They are using multiple frequency band ranges, using multiple location technologies and performing fingerprinting. These countermeasures are meant to increase the cost of an attack for the attacker rather than completely defend against the attack since it is not possible as of yet to mitigate jamming an entire band.

A. Use multiple bands

In our experiments, the environment consisted of access points spread across the entire 2.4 GHz band with no access points on the 5 GHz band, so to jam these access points, we needed to jam the entire 2.4 GHz band. We had to use two USRPs each with RFX2400 daughterboards to accomplish this, and the fake access points needed to be on a 5 GHz channel. However, many client devices are dual-band capable nowadays, meaning they can operate on the 2.4 GHz band or the 5 GHz band. We can take advantage of this feature to increase the cost of the attack by requiring access point information from both the 2.4 GHz band and 5 GHz band. The attacker would then need to jam legitimate access points on the 5 GHz band too if the attacker is to carry out the attack successfully on a client that is dual-band capable. The cost is incurred on the additional hardware that is needed to send the jamming signal, which means additional USRPs with a different daughterboard, or another jamming device. When a client is trying to use WiFi-based location services, the location service software should not display any messages if access points from both 2.4 GHz and 5 GHz bands are detected, and should display messages indicating less accuracy if only access points from one band are detected. Also, since 2.4 GHz bands are more common, location services software should look out for empty 2.4 GHz bands but non-empty 5 GHz band because it is a good indication of an attack like the one carried out in this experiment. This may change over time as more and more access points start using the 5 GHz band. This method is not foolproof as the attacker can choose to attack both bands, but the cost of the attack increases and effectiveness of the attack decreases if both bands are used.

The cost of implementing such change is low. It requires no modifications to hardware since a lot of devices are now dual-band capable. The change that needs to be made on the client side is in software. The algorithm for the entire locating process needs to be modified so that additional information is gathered when scanning for nearby access points. In addition to the MAC address and signal strength, the client should also collect band information, that is, whether the broadcasting access point is using the 2.4 GHz band or the 5 GHz band. If there is no mixture of 2.4 GHz and 5 GHz band access points, then the probability of an attack is higher than if access points exist on both bands and the client can display a message to the user indicating such a possibility, or at least point out that the location displayed may not be entirely accurate. This should be a trivial change in the algorithm as the scanning process already goes through all possible frequencies in all bands that the device is capable of, so for dual-band capable devices, access points broadcasting in both 2.4 GHz and 5 GHz bands can be seen as shown in Figure 13.

B. Use multiple location service technologies

There are three common types of location service technologies: GPS, WiFi and cellular. We assumed that GPS is not available since we are in an indoor environment so only WiFi and cellular service are available. Using both WiFi and cellular locating methods together will increase the cost of the attack, since an attacker now needs to jam all WiFi frequencies that legitimate access points use, and all frequencies that legitimate cellular networks in the surrounding area use. This means an attacker would need all jamming equipment used in this experiment, plus jamming equipment for jamming the 5 GHz WiFi band, and cellular bands. Since there are more cellular bands than WiFi, the cost of the attack increases dramatically. Note that cellular-based location service method will likely be used as a verification of WiFi-based location service results since cellular-based location service is much less precise. This method is also not foolproof since the attacker can choose to jam all of the frequencies, and can set up fake cellular base stations, which is available in open source [33], but the cost of the attack increases dramatically.

C. Noise level as part of fingerprinting

In [40], the authors proposed using unique access point characteristics such as traffic and signal fingerprints. The fingerprints would be stored in the service provider’s location database as another parameter that the client needs to send to get accurate location information. If the fingerprint does not match, then no location data is returned, or the most likely location is returned but with a warning about the accuracy of the location.

We propose an additional characteristic to fingerprint, that is, the noise level. Such fingerprinting would be similar to that proposed for traffic and signal fingerprints in [40], where the location service provider would perform fingerprinting and
store that information in the database. When clients make a location request, the noise level measurement similar to that described in [35] would be sent along with the MAC address, signal strength and the traffic and signal fingerprints. Fingerprinting of noise level can involve measuring the normal noise level, and interference that may occur in the environment from time to time. For example, in residential areas, since microwaves and some cordless phones operate in the 2.4 GHz band, this can be part of the fingerprinting process to create a more unique fingerprint. Denoting the transmitted signal component as $s(t)$ and noise component as $n(t)$, Equation 1 denotes what is received at the receiver.

$$r_1(t) = s(t) + n(t)$$

In digital communications theory, it is possible to recover the signal component $s(t)$ by using the minimum distance method if the noise component $n(t)$ follows a Gaussian distribution [31], or other methods can be used for different noise distributions. After getting the signal component back, it can be subtracted from the received signal to get the noise component. The recovered noise component can be processed and used as a fingerprint. To initially populate the database with noise fingerprinting values, a known signal can be sent by a transmitter, and subtracted at the receiver. The resulting noise component can be processed and stored in the location database as a fingerprint. This can be done several times to account for variations.

D. Analyze spectrum for detecting jamming

Since jamming requires an attacker to send a noise signal, a spectrum analyzer would be able to detect if the surrounding area is being jammed. In normal WiFi operation, we would see spikes in the frequencies at which data is being transmitted. If a frequency range is being jammed, then we will see spikes in surrounding frequencies and a lot of variation in the amplitude of all the frequencies. Basic spectrum analyzers can be written in software and used to measure the noise level detected in the surrounding area. Such software programs are already available on desktop computers (e.g. iStumbler [43]). While they provide very basic noise detection and not very accurate, it should be enough for the purpose of detecting the presence of jamming. This can be enhanced to notify users whether the application detects jamming or not. This can also be integrated with the operating system’s WiFi connection manager interface to alert users of potential jamming attacks.

If we denote the transmitted signal component as $s(t)$, noise component as $n(t)$ and the jamming component as $j(t)$, then the formula for the received signal is represented by Equation 2.

$$r_2(t) = s(t) + n(t) + j(t)$$

Like the noise signal $n(t)$, the jamming signal $j(t)$ distorts the original signal. However, the purpose of jamming is mainly to introduce a lot of errors such that it is very difficult or even impossible to decode the original intended signal, so a calculation of the bit error rate at the client side can show the presence of jamming.

E. Summary of countermeasures

Table I shows a summary of the countermeasures based on whether the countermeasure requires client or server side changes, cost for attacker and cost to implement.

One important thing to note is that a lot of devices on the market today are dual-band capable, and mobile devices,
especially smartphones, already have multiple technologies built into the phone (WiFi, cellular and GPS) so most of the cost is the time, effort and money spent on updating the software algorithms to take advantage of the additional hardware and information.

IX. CONCLUSION AND FUTURE WORK

In this paper, we have shown that Skyhook, Google, Apple and Microsoft location services can be denied and manipulated to provide the wrong location information, and we presented an alternative implementation of the WiFi-based location services attack that is done on dual-band hardware. We have also looked at many services that rely on location services and how the manipulation of location services can impact these services. Some services use location information just for added information while some rely on them to work. Our attack extends to WiFi infrastructure, as legitimate access points are disabled and attacker-controlled access points are used. Users might unknowingly connect to these access points and any traffic passed between the access point and the user’s device can be captured by an adversary for malicious purposes, which makes our attack more powerful than [40].

We proposed a number of countermeasures to the attack. These are using multiple bands, multiple location service technologies and fingerprinting. Many WiFi cards nowadays operate in both 2.4 GHz band and 5 GHz band so changes only need to be made in software to take advantage of it. Having more than one frequency range increases the jamming cost for the attacker. Smartphones now have WiFi, cellular and GPS technology built-in so each one can be used as verification of the others. Lastly, fingerprinting increases the cost dramatically for the attacker since the attacker needs to collect or fake fingerprint information, which is non-trivial.

The method of the attack used our experiment can theoretically be applied to non-WiFi-based location services. As noticed during the experiment, the Bluetooth wireless mouse that was used to operate the computer was having issues registering movements on the computer, which shows that Bluetooth was also affected by the jamming. Our method can be used to gather Bluetooth location service data, create fake Bluetooth access points or beacons and then trick devices. In addition, spoofing attacks on GPS has already been accomplished like in [21], and we believe that it is not hard to do the same to cellular-based location service, which uses cell tower triangulation to get an approximate location. So, with the cost and complexity of jamming plus impersonation attacks decreasing, it is imperative that more research is performed to provide secure solutions.

REFERENCES

[12] E. Burshtein. Using the microsoft geolocation api to retrace where a windows laptop has been.


