**Wireless/Mobile Networks, Key Exchange, and Web Servers**

Adam Livingston

University of Arizona

CYBV 326: Introductory Methods of Network Analysis

Professor Jonathan Martinez

12 December 2021

**Wireless/Mobile Networks, Key Exchange, and Web Servers**

Wireless networks at home, in public, and in private enterprises have greatly increased flexibility in how users connect to each other and the internet. The convenience of not requiring a hardwired internet connection through the use of cell phones or laptops is “evident to all—anywhere, anytime, untethered access to the global telephone network via a highly portable lightweight device.” (Kurose & Ross, p.519). Not only has this convenience been realized through cellular devices, but through home devices as well. There is rarely a home, coffee shop, hotel, or other venue in the United States that one will visit and not ask, “what is the WiFi password?” in order to achieve a connection to the network. The increased capabilities of microchips in conjunction with size decreases has allowed almost anything and everything to be connected to networks, including but not limited to thermostats, home security systems, refrigerators, and even meat smokers. This “internet of things” (IoT) continues to expand in breadth, depth, and complexity and there could be as many as 25 billion devices connected to the internet right now (Kurose & Ross, p.11).

The convenience that this has brought to the lives of many has also broadened the attack surface and avenues available to attackers to compromise such networks. Wireless networks require unique solutions to address the “challenges posed by networking these wireless and mobile devices, particularly at the link layer and the network layer.” (Kurose & Ross, p.519). The types of attacks are different from traditional wired networks as well as the mitigation strategies undertaken by cybersecurity professionals. This paper will explore a couple of these types of attacks as well as secure key exchange, client and server-side attacks, and secure connections with web servers.

**Wireless/Mobile Network Attacks**

The risk to wireless and mobile networks from attackers without proper security allows attackers, with the aid of their own wireless device, to have increased physical separation and decreased physical connection from their intended target. This means that physical controls, such as locks, guards, or motion sensors (Pfleeger, p.29) are less if not entirely useless and a defender is limited to the use of technical controls (network protocols, firewalls, etc.) to prevent attacks from occurring.

**Attack 1: Wardriving.** Perhaps the easiest way for an attacker to gain access to a network is if that network is unsecure. A method that attackers use to find and gain access to such unsecured networks is known as “Wardriving”. Quite simply, an attacker travels around with a mobile device, software to crack passwords or decrypt networks, and a GPS system (Norton). Along their route, they find WiFi access points, map them, and gather data on them.

Although benign at face value since anyone can find wireless networks and plot them on a map, an attacker with motive “could…gain access to any device connected to your home network.” (Norton). If an attacker gains access to devices on one’s home network, the victim could be subject to identity theft, fraud, and other crimes based on the information gathered from computers and other devices. While devastating for individuals and families, this type of attack has the potential to be dangerous for organizational networks, especially if the attacker gains access to public infrastructure such as electric or water supplies putting entire communities at risk.

According to Norton, there are ways for individuals to mitigate such an attack. A person may choose to turn off their router while they are away from home or not using it. However, it can be surmised this is too inconvenient for most people and having instantaneous internet connection via their WiFi router without an extra step of physically turning it on again is preferable to the alternate. Since most home networks use their own router or an Internet Service Provider (ISP)-supplied router, it is recommended to change the password to access it since “for most of the routers, the default username and password is “admin” and “admin”” (Software Testing Help). Since “password strength is determined by how many guesses are required” (Pfleeger p.42), and the fact that these are commonly known router passwords, an attacker can quickly infer what a likely router password could be. Other methods to protect one’s home network is to use the highest level of encryption possible such as WiFi Protected Access 2 (WPA2) or WPA3, ensure all software security patches are up to date, and to use a firewall to block unauthorized access to the network (Norton).

**Attack 2: Bluesnarfing.** Most cell phones, laptops, and other mobile devices are equipped with “An IEEE 802.15.1 network operates over a short range, at low power, and at low cost” (Kurose & Ross, p.549) known as Bluetooth. This network allows individuals to create a Wireless Personal Area Network (WPAN) of devices without the need of network infrastructure such as access points. “Slave” devices (smart watches, mice, keyboards, headphones, etc.) are tethered to a “Master Device” (PC, cell phone) into a piconet ran by the master device.

A bluesnarf attack uses this “Bluetooth technology to access restricted areas of a users’ device without their knowledge or approval for the purpose of capturing data e.g. contacts, images, lists of called missed, received or dialed, calendars, business cards and the device’s International Mobile Equipment Identity (IMEI).” (IJCSNS). To do so, the attacker uses a built-in capability of Bluetooth known as Object Exchange protocol (OBEX), typically used to push electronic business cards. According to IJCSNS, instead of pushing information, the attacker utilizes a “get request” to retrieve restricted files which highlights an enormous privacy concern. By also obtaining a device’s IMEI, an attacker can (illegally) clone the victim’s phone and conduct illegal activity while masked as the victim on the cellular network, therefore hiding the true identity of the perpetrator of such activity.

Thankfully, Bluetooth firmware updates have corrected much of this issue but motivated attackers can still accomplish this attack with increasingly powerful software (IJCSNS). According to CPI solutions, the simplest mitigation strategy, like mentioned above in “Wardriving”, is to turn off the Bluetooth functionality when not in use. Pairing Bluetooth devices also implements a Bluetooth pairing request prompt seen on the end of either the master or slave device. If one receives such a prompt from an unknown device, it is advised to never accept the request as an attacker may send unsolicited files to gain access. Finally, as many have become familiar with the concept of social distancing due to COVID-19, it is good practice to mind your distance since Bluetooth WPANs are limited by physical space (CPI). Combining these techniques is best practice to reduce the risk of becoming victim to a Bluesnarfing attack.

**Diffie-Hellman Key Exchange**

Throughout history in order to exchange messages with confidentiality, a core tenet of the CIA Triad, messages have been encoded with ciphers to exchange secrets between parties without fear of unwanted ears or eyes seeing it. This concept has been applied to modern-day computing through the use of what is known as the Diffie-Hellman Key Exchange and is used in modern-day public key cryptography systems (Kurose & Ross, p.604). There are two main types of encryptions, one of which is symmetric encryption where one key both encrypts and decrypts. But the problem of doing so despite this method being simple and secure is the secure exchange of that secret key over a public space, i.e. the internet. The solution is Diffie-Hellman, an asymmetric encryption method meaning that one key encrypts and a different key decrypts the message (Pfleeger, p.89). Diffie-Hellman Key Exchange ensures that two parties communicating with each other preserve the confidentiality and integrity of their message. To describe how this works, please reference the diagram and example below from Lecture 6 by Vanhoy, J.

In this example, Alice wants to communicate with Bob and for simplicity’s sake and without diving too far into mathematics, an analogy using paint will be used.

**1**

**Step 1**. Alice and Bob agree to use a public yellow key.

**Step 2**. Alice uses red as her private key, then sends Bob the yellow and red key mixed together, an orange key.

**3**

**2**

**Step 3**. Bob uses green as his private key, then sends Alice the yellow and green key mixed together, a blue mixed key.

**Step 4**. Alice mixes her red private key with Bob’s mixed blue key, obtaining a brown key.

**6**

**Step 5**. Bob mixes his green private key with Alice’s mixed orange key, obtaining the same brown key.

**5**

**Step 6**. They now share a secret (brown key).

To describe this a different way, assume the public key is variable G. Alice has private key, A and Bob has private key B. Alice mixes, creating AG (step 2) and sends to Bob. Bob mixes, creating BG (step 3) and sends to Alice. When they both add their private keys, they both get ABG (step 4, 5, and 6).

**4**

Since the private keys are unavailable to the public, an intruder that might intercept the message between Alice and Bob has no way of decoding it without both private keys, hence preserving the message’s confidentiality and integrity portions of the CIA Triad. As the example states, separating the encoded message without both private keys is expensive or nigh impossible. To ensure that both Bob and Alice know that the message came from the intended sender, also known as nonrepudiation or end-point authentication, a digital signature is needed (Kurose & Ross, p.606).

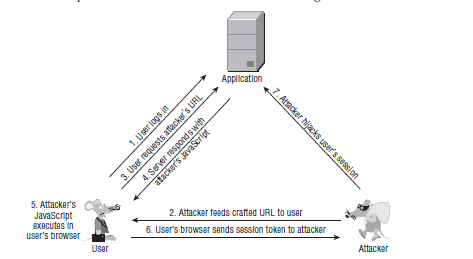
**Client and Server-Side Attacks**

Client-Side and Server-Side attacks are attacks in which harm can “come to an individual user interacting with Internet locations.” (Pfleeger, p.232). These attacks can be against web browsers, against and from websites, can seek sensitive data, and can occur through email in conjunction with social engineering.

**Attack 1: Content Spoofing (Client-Side).** According to OWASP, content spoofing “also referred to as content injection, “arbitrary text injection” or virtual defacement, is an attack targeting a user made possible by an injection vulnerability in a web application.” An attacker can exploit a web application that does not properly handle user-supplied data and can reflect a modified webpage back to a user, even while displaying the web page as a trusted domain (OWASP). This mainly presents a risk to organizations or businesses as a malicious actor can abuse the exploit by sending false information to customers or employees and cause them to switch to competitors. This is achieved by the implied trust a client has when connecting to a (seemingly) reputable website. An attacker may also target client-specific details by modifying a webpage to display a login form on a page that doesn’t have one. Used in conjunction with phishing or other social engineering techniques (Pfleeger, p.274), an attacker can find a site vulnerable to HTML injection, send a URL via email to their victim(s), the victim visits the trusted domain and fills out a form with username and password details, and that info is sent to the attacker’s server (Infosec). By retrieving the victim’s login information, an attacker can gain access to a victim’s banking information and siphon funds, leaving the victim penniless.

As good programming practice would dictate, as well as according to the Infosec Institute, the best mitigation strategy for such an attack is held within the programming phase. Ensuring that there is validation of user input(s), encoding user inputs that will result in an output by the web application, and to scan for vulnerabilities before deploying the web server. Another method is to use POST method (Infosec). When a user visits a webpage using TCP protocol, an HTTP request message packet (such as a GET request) is exchanged with the server to establish the connection. When a user enters their data into a field on the server, such as login information, the *Method* field of the packet request line can use the value of POST (Kurose & Ross, p.104). By using POST instead of the GET method, the user entering in their data will visit a version of the website specifically curated for their user and do not remain in the user’s browser history. This is opposed to the GET method which caches the information and can remain in the user’s browser history (W3schools). By taking this into account, a POST method for HTTP request messages is far more secure than GET methods.

**Attack 2: Cross-Site Scripting (XSS) Cookie Theft (Server-Side).** XSS “occurs when malicious web code is sent or executed, usually in script form, from the browser on the victim’s computer, using their web applications.” (Computer Networks). The intent behind such an attack is to gain personal user information or steal cookies to hijack the identity of a user to create a fraudulent session. Often, it is desirable for a website to identify users and cookies are the mechanism to maintain a session between users and the website (Computer Networks) and are in essence, like a memory for websites. According to Stuttard, D., & Pinto, M., stealing cookies involves seven steps: (1) Victim logs in as normal and receives a cookie, (2) through some means, the attacker feeds a crafted URL to the victim, (3) the victim then requests the attacker’s URL from the server, (4) the server responds to the victim’s request with the code the attacker created, (5) the victim’s browser receives the attacker’s code (6) and executes it, causing session token to be sent to attacker, (7) and finally with the user’s cookies the attacker can now hijack the user’s session.



There are many ways in which an XSS attack can manifest and the above example is one known as a “Reflected XSS Attack”. To prevent such an attack as described by Stuttard, D., & Pinto, M., every instance within the application where user-controllable data is being copied into responses needs to be identified. Once those instances are identified and to prevent vulnerabilities from being exploited, inputs need to be validated, outputs need to be validated, and dangerous insertion points must be eliminated. User-supplied inputs that may be copied need to be analyzed and confirmed such as email addresses, names, account numbers, or any other type of data parameters. Outputs where the application “copies into its responses any item of data that originated from some user or third party, this data should be HTML-encoded to sanitize potentially malicious characters.” (Stuttard, D., & Pinto, M., p.493). Finally, dangerous insertion points where user-controllable data is injected into script code need to be eliminated. An attacker can exploit this through creating their own malicious code and in essence attaching it to the application’s code. (Stuttard, D., & Pinto, M., p.495).

**Establishing a Secure Connection with a Web Server**

Establishing secure connections with a web server has shown widespread implementation of cryptographic techniques to “provide confidentiality, data integrity, and end-point authentication to a specific application, namely, e-mail.” (Kurose & Ross, p.631). There are three main ways in which those secure connections can be established: Secure Sockets Layer (SSL)/Transport Layer Security (TLS), IP security protocol (IPsec), and Secure Shell (SSH) (ScienceDirect).

**Method 1: SSL/TLS.** According to Stapko, “The SSL protocol provides blanket security for network communications by utilizing the advantages of both public-key cryptography and symmetric-key cryptography.” (p.68). Throughout the different versions of SSL being implemented, SSL v3 is called TLS (Kurose & Ross, p.631) hence why both SSL and TLS are under one method. To establish the connection, there is a handshake that begins like a normal TCP connection with a 3-way handshake. But once that happens there is an SSL handshake that takes place in order to verify that the users are actually who they say they are. During this handshake, the client and host perform a key exchange to generate an “Encrypted Master Secret” so that only the client and host have access to the session (Kurose & Ross, p.633).

**Method 2: IPsec.** IPsec is a suite of protocols that run at the network layer that provides cryptography to both IPv4 and IPv6 (ScienceDirect) and is commonly used in Virtual Private Networks (VPN). It has two main protocols, Authentication Header (AH) and Encapsulation Security Payload (ESP) where AH “provides source authentication and data integrity but does not provide confidentiality. The ESP protocol provides source authentication, data integrity, and confidentiality.” (Kurose & Ross, p.640). ESP can run in two modes: Transport or Tunnel. “Transport mode is intended for host-to-host connection and doesn’t hide the original packet’s header information. In comparison Tunnel mode fully encapsulates the IP packet inside a new IP packet, with completely new headers. ESP tunnel mode is the choice when maximum security and confidentiality are required.” (ScienceDirect).

To establish a secure connection, the client sends the remote host its identification in the form of a certificate and is signed by the client to assert its origin. The host signs the data and sends it back to the client, each party computes session keys, and finally exchanges the keys to establish the connection (ScienceDirect).

**Method 3: (SSH).** “SSH provides an authenticated and encrypted path to the shell or operating system command interpreter” in order to negotiate encryption between local and remote sites (Pfleeger, p.438). The client using SSH establishes the connection to the SSH server by using public key cryptography, first initiating the connection in which then the server sends the public key. The client then negotiates the parameters to open a secure channel and finally being able to log in to the server. To maintain confidentiality and integrity, symmetric encryption and hashing algorithms.

**Conclusion**

Wireless networks at home, in public, and in private enterprises have greatly increased flexibility in how users connect to each other and the internet. That flexibility has created an atmosphere of convenience but at the same time creates new avenues of attack and vulnerability for attackers to exploit. To further increase network strength and maintain the CIA Triad intact, secure connections must be established through cryptography and key exchanges. Not only is cryptography important, client and server-side attacks are an ever-present threat that must be mitigated through defensive programming and proper application of protocols in the transport-layer segments. In the constant struggle of defense against attacks, both attackers and defenders are only limited by their ingenuity in patching or exploiting vulnerabilities.

**References**

Kurose, J. F., & Ross, K. W. (2022). *Computer networking: A top-down approach*. Pearson.

Pfleeger, C. P. (2015). *Security in computing*.

Johansen, A. G. (2020, November 24). Wardriving: What it is and how to help protect your network. Norton. Retrieved December 4, 2021, from https://us.norton.com/internetsecurity-id-theft-wardriving-what-it-is-and-how-to-help-protect-your-network.html

Default Router Login Password For Top Router Models (2021 List). (2021, November 29). Software Testing Help. Retrieved December 4, 2021, from https://www.softwaretestinghelp.com/default-router-username-and-password-list/

Case Study on the Bluetooth Vulnerabilities in Mobile Devices. (2006a). IJCSNS International Journal of Computer Science and Network Security, 6(4).

Allen, G. (2021, March 17). How to Prevent a Bluesnarfing Attack. CPI Solutions. Retrieved December 4, 2021, from https://www.cpisolutions.com/blog/how-to-prevent-a-bluesnarfing-attack/

Vanhoy, J. (2021). Lecture 6: Data Hiding and Cryptography [PowerPoint Slides]. University of Arizona CYBV 301 Fundamentals of Cybersecurity.

Content Spoofing Software Attack | OWASP Foundation. (n.d.). Owasp.org. https://owasp.org/www-community/attacks/Content\_Spoofing

Content Spoofing. (2020, October 13). Infosec Resources. Retrieved December 6, 2021, from https://resources.infosecinstitute.com/topic/content-spoofing/

‌HTTP Methods GET vs POST. (2019). W3schools.com. Retrieved December 6, 2021, from https://www.w3schools.com/tags/ref\_httpmethods.asp

Rodríguez, G. E., Torres, J. G., Flores, P., & Benavides, D. E. (2020). Cross-site scripting (XSS) attacks and mitigation: A survey. Computer Networks, 166, 106960.

Stuttard, D., & Pinto, M. (2011). *The Web Application Hacker’s Handbook: Discovering and Exploiting Security Flaws. Second Edition.* Wiley Publishing, Inc.

‌Stapko, T. (2008). *Practical Embedded Security*. (2008). Elsevier. Newnes.

Secure Connection - an overview | ScienceDirect Topics. (n.d.). Www.sciencedirect.com. Retrieved December 7, 2021, from https://www.sciencedirect.com/topics/engineering/secure-connection

SSH Protocol- Secure Remote Login and File Transfer. (n.d.). Retrieved December 6, 2021, from https://www.ssh.com/academy/ssh/protocol#how-does-the-ssh-protocol-work

‌

‌

‌

‌