How Secure is WiFi MAC Layer in Comparison with IPsec for Classified Environments?*

Stanley L. Cebula , Aftab Ahmad, Luay A. Wahsheh, Jonathan M. Graham, Sandra L. DeLoatch and Aurelia T. Williams {s.l.cebula@spartans.nsu.edu}, {aahmad, lawahsheh, jmgraham, sjdeloatch, atwilliams}@nsu.edu
Information Assurance Research, Education and Development Institute (IA-REDI), College of Science, Engineering and Technology (CSET), Norfolk State University (NSU)
700 Park Avenue, Norfolk State University, Norfolk VA 20504

Keywords: WiFi, IPsec, WLANs, Information Assurance

Abstract

The IEEE 802.11-2007, like its earlier versions, provides a robust MAC layer with the help of mandatory CCMP and comprehensive key-generation, derivation, and distribution mechanisms. However, the physical layer continues to be without any protection from signal privacy attacks and anonymity of attacker within the WLAN, and it has no solution in the standard. The MAC layer, too, has not quite achieved the confidence of the networking community at the level of IPsec as of yet. In this paper, we present the results of a study that looks at the IEEE 802.11-2007 MAC security in juxtaposition to IPsec. We thus compare the attacks that can be thwarted by IEEE 802.11-2007 MAC as well as IPsec and the manners in which they can be thwarted by each security layer. The results can be used to propose enhancements to the IEEE 802.11-2007 MAC layer in order for it to gain the same level of confidence as obtained by IPsec. The purpose is not to choose between IPsec and IEEE 802.11-2007 security (as both are required at different layers), rather it is to help understand what could be added to the IEEE 802.11-2007 standard to make it as secure as IPsec.

1. INTRODUCTION

The privacy promised by WEP in the early editions of the IEEE 802.11 suite of standards was invariably taken as a security measure. Its failure thus resulted in the WLANs being branded as insecure. The later editions of the standard and the WiFi Protected Access (WPA) from WiFi Forum did restore some confidence; however the DoD community and large businesses took their time before fully trusting WLANs. On the contrary, IPsec has enjoyed the trust of the networking community from the beginning. It therefore makes sense to juxtapose the two suites of security protocols with the intention of evaluating WiFi medium access control (MAC) layer security in light of IPsec. In order to get a clear picture, we make assumptions throughout the paper that make the comparison look scenario-driven.

The primary application of the work reported in this paper is for classified environments; however, the insight provided is applicable to any networking instance.

The remainder of this paper is organized as follows: a summary of the security of the IEEE 802.11 standard is provided in Section 2. Specific attacks against WLANs are examined and discussed in Section 3. A brief overview of the security of IPsec is in Section 4. Specific attacks against IPsec are examined and discussed in Section 5. A comparison and suggestions are made in Section 6. We conclude the paper in Section 7.

2. IEEE 802.11-2007 MAC SECURITY

This section will summarize the security provided by the IEEE standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications as it applies to classified environments. The standard mandates the use of the Robust Secure Network Association (RSNA) that accompanies security algorithm framework, security association management, and keys and key distribution.

2.1. RSNA Security Algorithm Framework

The security of RSNA can be affected by the environment in which it is used and the correctness of assumptions and constraints. RSNAs can be established in an extended service set that is based on authentication and key management and an extended service set that is based on a preshared key. RSNAs also have a set of assumptions and constraints that increase security: each station can generate cryptographic-quality random numbers, mutual authentication is used (prevents man-in-the-middle attacks), the mutual authentication method must be strong (just strong enough to make impersonation attacks computationally infeasible), the access point and authentication server has a secure channel between them (preventing exposure of cryptographic keys), the authentication server keeps the symmetric key used by the access point and station a secret (ensuring the authentication server is never compromised), a station keeps the common symmetric key used with its peer a secret (preventing attackers from gaining access to a means of breaking the cipher used), the station's supplicant and authenticator generate a new pairwise transient key for each new session (meaning no keys are reused), and ARP (address resolution protocol) or ICMP (internet control message protocol) is used to ensure the destination station is the correct party (making sure nothing is sent by accident to a different party) [1]. All of these assumptions and constraints are important to making RSNA algorithms as secure as possible. The environment that RSNAs are used and the use of constraints and assumptions increase the security of 802.11 networks.

2.2. RSNA Confidentiality Protocols

The IEEE standard defines two data confidentiality protocols for 802.11 networks: TKIP (Temporal Key Integrity Protocol) and CCMP (counter mode with cipher-block chaining

message authentication code protocol). CCMP must be used in all devices that claim they are RSNA compliant. TKIP is only used when communicating with devices that are not able to communicate with CCMP (older devices based off pre-RSNA security). Since our paper focuses on a classified environment where every device is technologically up-to-date, TKIP will never be used. Therefore, this paper will only discuss CCMP.

The purpose of CCMP is to provide confidentiality, integrity, and authentication on RSNA devices. CCMP is based on the counter mode with cipher-block chaining message authentication code of the AES (advanced encryption standard) encryption algorithm. It combines a counter mode for data confidentiality and cipher-block chaining message authentication code for authentication and integrity. This protects the integrity of the MAC protocol data unit (PDU) data field and some parts of the MAC protocol data unit header. Figure 1 shows the MAC PDU when using CCMP. The sections of the MAC protocol data unit highlighted in red represent the additions CCMP makes to the original data unit. Figures 2 and 3 show CCMP encapculation and decapsulation.

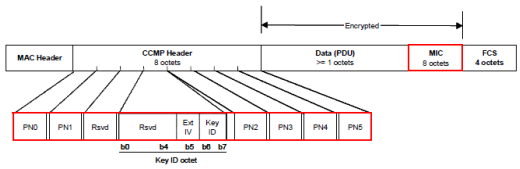

Figure 1. MAC Protocol Data Unit Using CCMP [1]

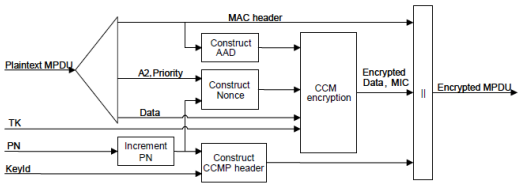

Figure 2. CCMP Encapsulation Process

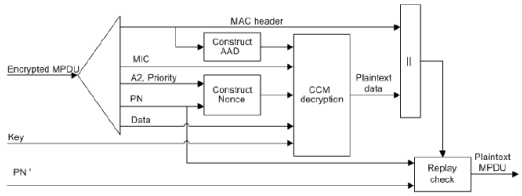

Figure 3. CCMP Decapsulation Process

2.3. RSNA Security Association Management

Security association means secure operations. A security association is a set of policies and keys used to protect information. Each party in the security association stores this information. There are two types of security associations supported by an RSN station that will be used in a classified environment: pairwise master key security association and pairwise transient key security association.

A pairwise master key security association occurs upon a successful IEEE 802.1X exchange, preshared pairwise master key information, or when the pairwise master key is cached [1]. This security association is bidirectional, which means both parties can send and receive information. This security association is also used to create a pairwise transient key security association.

A pairwise transient key security association results upon a successful four-way handshake. This security association is also bidirectional. There is only be one pairwise transient key security association between the same supplicant and authenticator MAC addresses.

In order to establish an RSNA with other stations, a station must advertise its capabilities and specify all the authentication and cipher suites enabled by their policies. The policy selection, authentication, and key management processes are all important when considering the security of the communication between two stations.

The station attempting to connect to another station performs RSNA policy selection. In order for communication to be successful, the connecting station must use these policies. Stations that wish to authenticate will only do so to other stations that they choose to connect to. This prevents unwanted and distrusted connections. Before authentication is completed, all communications between the authenticator and supplicant are completed through the IEEE 802.1X uncontrolled port. After authentication is successful, authenticators and supplicants can communicate using the IEEE 802.1X controlled port. After authentication has been completed, the station and authentication server will share a secret key called a pairwise master key. After the pairwise master key has been shared, a key confirmation handshake is initiated by the four-way handshake in order to: confirm the existence of the pairwise master key for the peer, ensure the security association keys are fresh, synchronize the installation of temporal keys into the MAC, transfer the group temporal key from the authenticator to the supplicant, and confirm the selection of cipher suites that will be used [1]. Policy selection, authentication, and key management are all important concepts that contribute to the overall security in a security association.

3. ATTACKS AGAINST WLANS

There are many different attacks that have been used to cripple WLANs so attackers could gain access to information or prevent machines from doing their jobs. This section will look at session hijacking attacks, denial-of-service attacks, man-in-the-middle attacks, forgery attacks, and other simple attacks that the IEEE 802.11 standard defends against.

3.1. Session Hijacking Attacks

Session hijacking is gaining unauthorized access to information or services in a computer system. The most common form of session hijacking in a classified environment occurs when an attacker is trying to gain access to a network or information. In both cases, the attacker tries to gain control of a robust security network association. In order for this attack to work successfully, the attacker has to time the attacks carefully. As described in [2], "the supplicant and the authenticator engage in the authentication process, which results in the supplicant being authenticated." An attacker then sends a disassociate message to the supplicant pretending to be the authenticator (using the MAC address of the authenticator). If the supplicant trusts the message, the supplicant will disconnect from the authenticator. However, as stated in [2], "since this disassociate message was sent by the attacker, the real access point does not

know about it." This attack must occur after message three is sent from the supplicant to the authenticator and before message four is sent from the authenticator to the supplicant. Therefore, the attacker can then use the supplicant's MAC address to connect to the authenticator, successfully hijacking the original session.

There are two security features built-in CCMP that prevent this attack from occurring successfully. CCMP encapsulation and encryption counter a session hijacking attempt [3] [1]. As explained in section 3.2, CCMP encrypts the plaintext MAC protocol data unit and then encapsulates the cipher text. In order for an attacker to be able to read anything transmitted in the CCMP MAC protocol data unit, the attacker would have to successfully decapsulate and decrypt it. This is not possible, because the keys needed to decapsulate and decrypt the MAC protocol data unit are only known to the authenticator and supplicant. Due to the encryption and encapsulation, the attacker does not have access to the ANonce or RSN information element. As stated in [1], if the key replay counter field, ANonce value, or RSN information element is not exactly what is expected, the message will be discarded silently. Discarding message silently prevents the attacker from realizing the attack attempt has failed. This would prevent the attacker from being able to send messages to the authenticator posing as the supplicant.

3.2. Denial-of-Service Attacks

A denial-of-service attack attempts to make a computer resource unavailable to other users. The most common denial-of-service attack that would occur in a classified environment is an attempt to connect over and over to the authentication server until it crashes due to network volume. This would cause all of the devices that were previously connected to the authentication server to disconnect as well. It is also reasonable to assume a denial-of-service attack could occur against a specific supplicant in order to remove them from the network. As stated in [4], "the vulnerability [of a denial-of-service attack] results from the lack of any authentication in message 1." Denial-of-service attacks against a specific supplicant is rare, but they can occur. In order for a denial-of-service attack to occur against a supplicant, message one would be sent over and over again.

There is a constraint in the structure of RSNA that protect authenticator from denial-of-service attacks. A denial-of-service attack against the authenticator will fail, because the authenticator can only have one active handshake in progress for each supplicant it is connected to. As explained in [5], the authenticator can "discard an unexpected response and retry the previous message or terminate the handshake if the expected response is not received during a given time interval and certain number of retries." Since each authenticator can only have one active handshake per supplicant, it is impossible for a denial-of-service attack to take place.

For denial-of-service attacks against supplicants, nonce values can be reused for more protection. The 802.11i (now section 8 of the 2007 Edition) adopted a modification that allows supplicants to re-use nonce values. As asserted in [4], "re-using the nonce until one four-way handshake completes allows the supplicant to avoid storing state, which prevents memory exhaustion." Even though reusing nonce values goes against the purpose of using nonce values (to create a new value every time), it can be successful in preventing denial-of-service attacks against the supplicant. In other words, a supplicant will not create and store a new nonce value until the original four-way handshake that created the nonce value has been completed. This will prevent hundreds and thousands of nonce values from being stored in the supplicant, which will prevent the denial-of-service attack. However, the system administrators will need to specify this option manually.

3.3. Man-in-the-Middle Attacks

A man-in-the-middle attack is generally a form of eavesdropping that takes place when an attacker fools the authenticator and supplicant into making independent connections with them acting as a midpoint. In a classified environment, eavesdropping can be very detrimental to an organization. As mentioned in [3], the attacker "fools users and other access points forcing them to send data through the unauthorized device." However, the authenticator and supplicant think they are actually only talking to each other. The attacker can then control the entire conversation. In order for a successful man-in-the-middle attack to occur, the attacker must manipulate address resolution protocol (ARP). As explained in [6], "the attacker sends the victim (supplicant) ARP replies that wrongly associate the internet protocol (IP) address of the victim's (supplicant's) default gateway (authenticator) with the attacker's MAC address." This leads the supplicant to believe they are sending packets to the authenticator. In reality, the supplicant is actually sending packets destined to the authenticator to the attacker. Furthermore, "the attacker also sends the gateway (authenticator) ARP replies that that wrongly associate the victim's (supplicant's) IP address with the attacker's MAC address [6]." This action will cause the authenticator to send messages intended to the supplicant through the attacker. Through successful manipulation of ARP, the attacker has setup a man-in-the-middle attack.

Using random nonce values and a message integrity code in the four-way handshake prevents man-in-the-middle attacks. Specifically in [1], "with unpredictable nonces, a man-in-the-middle attack that uses the supplicant to precompute messages to attack the authenticator cannot progress beyond message 2, and a similar attack against the supplicant cannot progress beyond message 3." If an attacker modifies the ANonce value or the address, it will show in the message integrity code. This will cause the supplicant to drop every packet coming from the attacker trying to pose as the authenticator. Also, the same is true in reverse. This forces the attacker to try to pre-compute the nonce values used by the authenticator and supplicant. The use of random nonce values for the ANonce and SNonce will prevent the attacker from guessing or precomputing correctly. Authors in [6] conclude, "such verification thwarts man-in-the-middle attacks."

3.4. Forgery Attacks

A forgery attack consists of an attacker using false information to gain access to services or information on a computer system. The most common form of forgery in a classified environment consists of an attacker stealing a supplicant's information (normally on the classified network), and using it to connect to an authenticator to steal information. The authenticator will think the attacker is actually the

supplicant. If the attacker commits any crimes or suspicious behavior, it will be logged as coming from the supplicant as opposed to the attacker. Forgery attacks are usually used in other types of attacks. Specifically, an attacker would try to steal a supplicant's MAC address or IP address in order to use it to connect to the authenticator. This is possible, because the standard "does not provide authentication of control messages for establishing association between station and access point [7]." An attacker could intercept a control message in order to determine the MAC address or IP address of the supplicant and authenticator.

Forgery attacks are prevented due to the structure of the pairwise transient key and the use of nonce values used in the encapsulation process. As stated in [1], "pairwise key support with CCMP allows a receiving station to detect MAC address spoofing and data forgery." Even though control messages can be forged, illegitimate control messages prevent progress of the next stage because the standards allows mutual authentication [7]. Authors in [1] continue to explain, "message 1 of the four-way handshake can be forged, however the forgery attempt will be detected in the failure of the four-way handshake." The nonce values that are used in the four-way handshake are created using a random number, init counter, and local MAC address concatenated with the time. Therefore, "if an attacker creates a MAC protocol data unit with a spoofed transmitting address, then the decapsulation procedure at the receiver will generate an error [1]." Since there will be an error in the decapsulation process, the attacker will not be able to successfully communicate with the authenticator.

3.5. Other Attacks

There are many attacks that are prevented through the use of a message integrity code throughout the four-way handshake process. The types of attacks that are thwarted include: bit-flipping attacks, data truncation, concatenation, and splicing attacks, fragmentation attacks, iterative guessing against the key, redirection by modifying the MAC PDU destination address or receiver address field, and impersonation attacks by modifying the MAC PDU address or transmitter address field [1]. Bit-flipping attacks modify the cipher text in order to look for a pattern that arises in the plaintext. Data truncation attacks cut data off the end in order to cause errors for the receiver of the data. Data concatenation attacks put data together in order to confuse the receiver of the data. Data splicing attacks join different parts of data in order to cause errors in the decryption process. Fragmentation attacks break up a single packet into smaller individual datagrams, which can cause overlapping, buffer overflow, and overwriting. Iterative guessing against the key consists of guessing what the value of the keys could be. Redirection attacks involve editing the destination address in the MAC protocol data unit to send packets other places besides the original destination. Impersonation attacks consist of editing the source address to make the receiver think the packet came from somewhere else.

The IP address of the machine, a message priority, and the unencrypted data are used to calculate the message integrity code. Any change to either of these between the time the sender sends the packet and the receiver receives the packet results in a different message integrity code. Bit-flipping attacks, data truncation attacks, data concatenation attacks, and data splicing attacks all modify data in a MAC PDU. Fragmentation attacks alter the IP addresses used in the MAC PDU. Iterative guessing against the key modifies some keys in the MAC PDU. Modification of keys will result in an error when the attacker tries to compute the message integrity code. Redirection attacks modify the destination address, and impersonation attacks modify the source address. Therefore, by including the message integrity code in the four-way handshake, all of these attacks are prevented.

4. SECURITY PROVIDED BY IPSEC

IPsec has been evolved for a more customizable security. The Authentication Header (AH) and Encapsulating Security Payload (ESP) protocols within IPsec both support protection of data in the IP packet payload (transport mode) and the entire IP packet (tunnel mode) [8] [9].

4.1. Security Associations

IPsec uses the concept of security associations to represent the bundle of algorithms and keys used to provide authentication during communication from one machine to another. It relies on many different sources for different parts of the security association: IKEv2, Security Policy Database (SPD), Security Association Database (SAD), and Peer Authorization Database (PAD). IKEv2 is the protocol that is used to set up a security association. The SPD is a database that contains policies, which control the services provided to IP datagrams. The use of the SPD represents a form of access control. The SAD is a database that contains security associations and the specific characteristics for each. The PAD is a database that contains the different peers or groups that are allowed to communicate with a certain IPsec entity [8]. All of these sources work together in order to provide authentication during communication for two or more machines.

In order to create a security association, two hosts need to participate in the IKEv2 process. Figure 5 shows how two machines set up a security association. This example assumes that a security association does not already exist between the two machines.

5. ATTACKS AGAINST IPSEC

There are a few attacks that have been used to take advantage of IPsec vulnerabilities so attackers could gain access to information or prevent machines from doing their jobs. This section will look at a denial-of-service attack using ICMP (internet control message protocol) messages, attacks against the initialization vector to recover data, an attack against the initialization vector for denial-of-service, and other thoughts on attacks and security of IPsec.

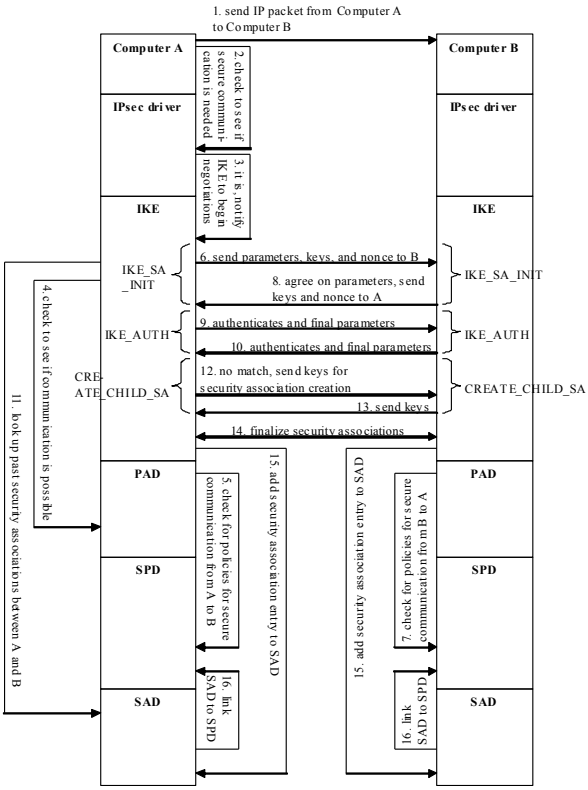

Figure 4. Security Association Process

5.1. Denial-of-Service Attack Using CCMP

In his attack, ICMP messages are used to deliver error messages (destination unreachable) or non-error messages (echo) between two networked computers. ICMP messages that are error messages are not secure in any way. ICMP messages that are non-error messages are supposed to be governed using a security association in the SPD [8]. In an ICMP flood attack, the attacking node sends large amounts of ICMP packets to the victim with their source targeting at another IPv6 node or an invalid IPv6 address [10]. This will attempt to waste the resources of the node being attacked in order to try to make it unavailable to other users. In order for this attack to succeed, two computers need to complete the process of developing a security association. Then, the attacker will start the attack, sending ICMP echo messages to the victim over and over again. Eventually, the victim will not be able to respond to other requests, because all of their resources are tied up trying to process the attacker's requests.

According to [8], "disposition of non-error, ICMP messages (that are not addressed to the IPsec implementation itself) MUST be explicitly accounted for using SPD entries." This guarantees that upon reception of non-error ICMP messages, the receiver can trust the sender. After testing the ICMP-flood attack on IPsec, [10] found the IPsec mechanism is ineffective against distributed denial-of-service attacks without a spoofed source address. Also, the IPsec mechanism is effective in defending against distributed denial-of-service attacks using ICMP messages with a spoofed source address [10]. In any case, these attacks were conducted after a security association was set up between the attacker and victim. Obviously, the culprit will be easily located. Not only does the attacker have to be physically connected to the network, but the attacker also has SPD entries tying themselves to the victim. This attack will only occur in rare, espionage situations.

5.2. Attacks Against the Initialization Vector

Probable plaintext and bit-flipping attacks can be used to try to recover data from an IPsec packet that has been secured using the ESP protocol with encryption and without authentication where the encryption algorithm is of CBC (cipher block chaining) mode. According to [11] a probable plaintext attack works by looking at certain bit positions for which a likely value can be predicted. Next, a comparison engine counts the number of matches and determines if further analysis could reveal more information from the captured packet. Next, a second-stage engine could use more probable plaintext techniques or human analysis [11]. Bit-flipping attacks take advantage of the initialization vector used in an encryption algorithm that employs CBC mode. Three of the four supported algorithms in the ESP protocol support CBC mode. The algorithm splits the plaintexts into blocks and each block is encrypted based on the value of the block before it. A picture depicting the decryption process of a CBC mode algorithm is depicted below.

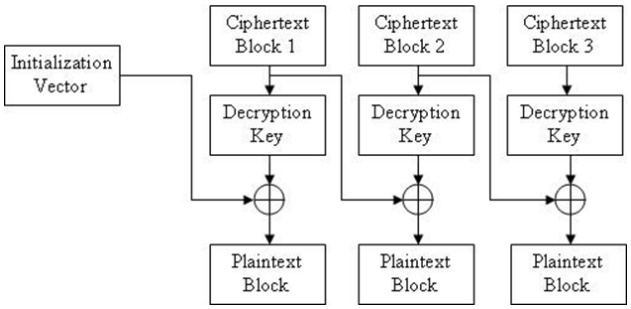

Figure 5. CBC Mode Encryption Scheme

As shown above, the ciphertext of block 1 is used in the exclusive OR function under ciphertext block 2 (that's why referred to as a chain of blocks). However, there needs to be values inserted into the first block of ciphertext. This data is called the initialization vector. The initialization vector is sent without an authentication mechanism; therefore, an attacker can change the value. The initialization vector may be carried explicitly in the payload field and usually is not encrypted per se [12]. Flipping bits in the initialization vector in a controlled way will affect bits in the plaintext blocks. With the combination of probable plaintext attacks and bit-flipping attacks on the initialization vector, the decryption key and entire plaintext blocks can be deduced.

In order to prevent attacks against the initialization vector, there are a few steps that can be taken. First, computers could avoid the use of CBC mode encryption algorithms when setting up a security association. However, this is unrealistic, because only one algorithm is not a CBC mode algorithm (AES-CTR). Second, a constant initialization vector could be agreed upon. If the initialization vector was a constant value, it would never have to be modified. However, this limits the security of the

encryption. Lastly and more successfully, the ESP protocol should not be used with encryption and without authentication. The use of authentication would completely prevent the modification of the initialization vector being successful [12]. This is the best solution to this attack.

5.3. Denial-of-Service Attacks Against the Initialization Vector

An attack proposed by [13] combines the attacks discussed in sections 5.1 and 5.2. This attack also assumes use of the ESP protocol with encryption (CBC mode) and without authentication. In this case, the initialization vector is attacked in order for the attacker to attempt a denial-of-service attack. First, the initialization vector is changed in such a way that the first decrypted block of plaintext is changed in a predictable way [13]. If the CBC mode encryption block's size is 192 bits, the source address will be in the first decrypted block of the plaintext in an IPv6 header. An attacker can then modify the source address in the IPv6 packet. In order to guarantee these changes will be accepted by the receiver of the packet, a new initialization vector needs to be calculated to offset the changes made to the source address [13]. In order to be able to change values in different protocols (TCP, UDP, and RTP), larger block sizes will need to be used. For example, the TCP Checksum field appears between bits 288 and 304 (block size would need to be at least 310 bits). If this were the case, the checksum could also be modified in order to guarantee the receiver would still accept the packet. In order for the attack to work, a packet would have to be captured, modified, and re-injected into the network. According to [13], "for any legitimate encapsulating security payload datagram, the attacker can generate up to 2^{32} false source addresses and hence so many false encapsulating security payload datagrams, which will have the same content as the legitimate one but will claim to come from different sources." By successfully pulling off this attack, one can create over four billion false datagrams. This many datagrams will successfully take down most machines on a network.

In order for this attack to be successful, it is assumed connections use the ESP protocol with encryption (CBC mode) and without authentication. As with the attacks in 5.2, all of the safeguards discussed in 5.2 will successfully protect machines from harm. While three solutions were discussed, the best solution is to use the ESP protocol with encryption and authentication. Authentication will prevent attackers from successfully modifying data.

5.4. Other Thoughts on Attacks and Security of IPsec

As discussed in [13], the authors of [14] have exposed even more serious weaknesses of the encryption-only configuration on IPsec. Authors of [14] have successfully and efficiently completed cipher-text only attacks (no knowledge or guessing of plaintext) to reveal encryption keys. In combination with the attacks discussed in this paper, we conclude with [15] that, "it is quite clear that encryption without integrity checking is all but useless." Furthermore, [11], [12], [14], and [15] all agree in recommending authentication be a mandatory part of IPsec, or encryption-only configuration should be banned from IPsec.

6. COMPARISON (RESISTANCE TO ATTACKS)

Based on the summary and analysis of the security provided by IPsec and WiFi 802.11, we can compare vulnerabilities based on their type. This section will compare IPsec and WiFi 802.11's resistance to denial-of-service attacks, attacks against encryption keys, and the ease of attempting attacks.

6.1. Denial-of-Service Attacks

The denial-of-service attack against the supplicant in WLANs in section 3.2 attempts to make the supplicant unusable through the use of control messages. The denial-of-service attack against IPsec in section 5.1 also attempts to make a computer unusable (through the use of ICMP messages). Both of these attacks have the same goal, therefore, we can compare their successes and develop a general comparison. The denial-of-service attack against the supplicant in WLANs in section 3.2 can be attempted under any configuration of security under the 802.11 protocols. This attack will succeed unless extra help is given to the supplicant by the system administrator. As stated in section 3.2, "the vulnerability [of a denial-of-service attack] results from the lack of any authentication in message 1 [4]." The denial-of-service attack in IPsec described in section 5.1 only occurs between two machines that are in a security association. In order to set up a security association in IPsec between two machines, different types of authentication must be completed in the IKEv2 process. Therefore, IPsec's inclusion of authentication between two machines prevents a denial-of-service attack from successfully occurring. This leads us to state IPsec defends against denial-of-service attacks better than 802.11. 802.11 should add authentication or a constraint for previous authentication during message 1 of the four-way handshake to prevent denial-of-service attacks against the supplicant.

6.2. Attacks Against the Encryption Key

In order for the attacks against WLANs in sections 3.1, 3.3, and 3.4 to complete successfully, the encryption key will need to be recovered. The attack against IPsec described in 5.2 is based on attacking the initialization vector that can result in obtaining the encryption key. These attacks have a common goal; recovering the encryption key in order to decrypt ciphertext. Based on this common goal, we can compare the attacks against WLANs in section 3.1, 3.3, and 3.4 to attacks against IPsec in section 5.2. In WLANs as stated in section 3.2, the presence of the message integrity code prevents the successful modification of a packet. In IPsec as stated in 5.2 and 5.3, the presence of the integrity check value also prevents the successful modification of a packet. Based on these common characteristics, both WiFi and IPsec prevent the modification of packets with the intent of recovering the encryption key. We can conclude WiFi and IPsec protect against attacks against the encryption key with the same level of resistance (successfully with no outside help).

6.3. Difficulty to Attack

In order for attacks to occur against WLANs, the attacker must be in range of an access point on the network. In order for attacks to occur against IPsec, the attacker must be connected to a machine on the network. More specifically, if an attacker wanted to attack a bank's WiFi network, he/she could conduct

the attack in many places: a vehicle outside the bank, a dumpster in an alley, or from the bank's restrooms (as long as there was WiFi access there). If the bank's network was not wireless and protected with IPsec, the attacker would need to be at a machine on the bank's network. IPsec makes attacks much harder to accomplish. This leads us to conclude it is easier to attack WiFi 802.11 protected networks as opposed to IPsec protected networks. In order to solve this problem, the implementation of additional software could be used. If system administrators had a map of their network with every host plotted that is connected to the WLAN, attacks would be possibly located. After the attack is located, security officers can be deployed to stop the attack. If system administrators can locate attacks with ease, less people will try to attack WLANs.

Below is a picture comparing the results of our comparison between the security of WiFi and the security of IPsec:

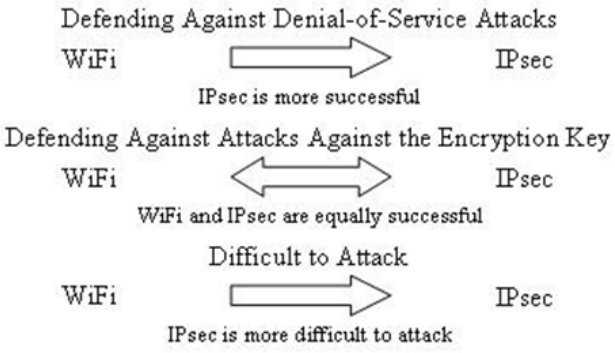

Figure 6. WiFi and IPsec Comparison

7. CONCLUSION

WLANs have become popular, because they are easy to setup, maintain, and use. Classified environments manage very important information and services. Due to these facts, the security of WLANs as they apply to classified environments has been examined to determine if the 802.11 standard protects against specific attacks. Based on our research, we conclude that the security provided by the 802.11 standard is successful in defending against many popular attacks including: session hijacking, denial-of-service attacks against the authenticator, man-in-the-middle attacks, forgery attacks, data manipulation attacks, fragmentation attacks, iterative guessing attacks, redirection attacks, and impersonation attacks. While the IEEE 802.11 standard protects against all of these attacks, denial-of-service attacks against the supplicant are still possible to achieve. It is not acceptable to consider the reuse of nonce values as a defense against this attack. Upon examining the security of IPsec, we conclude WLANs can become more secure by implementing IPsec concepts. Adding authentication to control messages in any part of the RSNA would successfully prevent denial-of-service attacks. We also conclude that both IPsec and the 802.11 protocol defend against attacks on the encryption key successfully. Lastly, it is easier to attack WLANs because of their mobility and ease of use. With the help of additional software, such as network mapping, attackers will be less inclined to attempt attacks. If attackers still attempt attacks, system administrators will know exactly where it is coming from.

References

[1] IEEE. "Std 802.11-2007, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Section 8—Security".

[2] Zahur, Y. and T. A. Yang. "Wireless LAN Security and Laboratory Designs". Consortium for Computing Sciences in Colleges. 2003. 44-60.

[3] Sriram, V. S. S. and G. Sahoo. "Securing IEEE 802.11 Wireless LANs – A Mobile Agent Based Architecture". IEEE International Advanced Computing Conference. Patiala, India, 2009. 1187-1191.

[4] He, C., et al. "A Modular Correctness Proof of IEEE 802.11i and TLS". ACM Conference on Computer and Communications Security. Alexandria, Virginia, USA, 2005. 2-15.

[5] He, C. and J. C. Mitchell. "Analysis of the 802.11i 4-Way Handshake". ACM Workshop on Wireless Security. Philadelphia, Pennsylvania, USA, 2004. 43-50.

[6] Brustoloni, J. C. "Laboratory Experiments for Network Security Instruction". ACM Journal on Educational Resources in Computing (n.d.): 1-18.

[7] Hori, Y. and K. Sakurai. "Security Analysis of MIS Protocol on Wireless LAN Comparison with IEEE802.11i". Mobile Technology, Applications, and Systems. Bangkok, Thailand, 2006. 1-5.

[8] Kent, S. and K. Seo. "Security Architecture for the Internet Protocol". December 2005.

[9] Cheng, P. C., et al. "A Security Architecture for the Internet Protocol". IBM Systems Journal (1998): 42-60.

[10] Yang, X., T. Ma and Y. Shi. "Typical DoS/DDoS Threats Under IPv6". Proceedings of the International Multi-Conference on Computing in the Global Information Technology. 2007.

[11] Bellovin, S. M. "Probable Plaintext Cryptanalysis of the IP Security Protocols". Proceedings of the Symposium on Network and Distributed System Security. 1997. 155-160.

[12] McCubbin, C. B., A. A. Selcuk and D. Sidhu. "Initialization Vector Attacks on the IPsec Protocol Suite". Proceedings of the 9th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises. 2000. 171-175.

[13] Nikov, V. "A DoS Attack Against the Integrity-Less ESP (IPSec)". Proceedings of the International Conference on Security and Cryptography. 2006. 192-199.

[14] Paterson, K. G. and A. K. L. Yau. "Cryptography in Theory and Practice: The Case of Encryption in IPsec". EUROCRYPT. Springer-Verlag, 2006. 12-29.

[15] Bellovin, S. M. "Problem Areas for the IP Security Protocols". Proceedings of the Sixth Usenix UNIX Security Symposium. San Jose, 1996. 205-214.

Biography

Stanley Cebula is a student pursuing a Master of Science degree in Computer Science at Norfolk State University. He received his Bachelor of Art's degree from Virginia Wesleyan College in Computer Science and Criminal Justice. His research interests include: security of wireless networks, ethical hacking, and biometrics. He is also conducting his thesis research on securing the physical layer for IEEE 802.11 LAN. He has one publication, "Computer Ethical Hacking: An Education Perspective."

Aftab Ahmad is an Associate Professor in the Computer Science Department since 2003. His current research interests are in Wireless Networks for healthcare.

Luay Wahsheh is an Assistant Professor in the Computer Science Department. His research interests are in security policy and access control in medical records.

Jonathan Graham is an associate professor in the Computer Science Department and the Director of NSU IA-REDI. His research interests are in digital forensics and intrusion detection.

Sandra DeLoatch is the Dean of College of Science, Engineering and Technology (CSET) and the principal investigator of the Massie Chair grant, of which the two main projects are secure cloud computing and WLAN security in classified environments.

Aurelia Williams is an Assistant Professor in the Computer Science Department. Her current research interests are in digital forensics and cloud computing security.

---

[*] *Acknowledgement:* "This material is based upon work supported by the Department of Energy National Nuclear Security Administration] under Award numbered DE-FG52-09NA29516/A000." *Disclaimer:* "This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. References herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof."