Network Security

Professor Gaurav S. Kasbekar
Department of Electrical Engineering
Indian Institute of Technology, Bombay

Week - 06

Lecture - 35

Securing Wireless LANs : Part 1

Hello, in this lecture and the next few lectures, we will discuss the security of wireless Local Area Networks (LANs). In particular, we will discuss the security of Wi-Fi, which is the most popular wireless LAN standard today. So, the technical name of Wi-Fi is IEEE 802.11. In a Wi-Fi network, mobile nodes, such as smartphones and laptops, communicate over the wireless channel with an access point, which is a base station-like device. In this picture, this is the access point, which is like a base station, and mobile devices, such as smartphones and laptops, communicate wirelessly with the access point.

So, the wireless channel is a shared medium. Whatever one node transmits, that information reaches all the other nodes in the vicinity. So, these different nodes have to share the medium among themselves using an appropriate medium access control protocol. Now, why is the name of Wi-Fi 802.11? So, it's an IEEE standard.

IEEE has a series of standards with the numbering 802.something for different kinds of networks. For example, 802.3 is a series of wired local area network standards, 802.11 is a series of Wi-Fi standards, and 802.16 is a series of WiMAX standards. So, it was a cellular technology or metropolitan area network technology, which was an alternative to LTE Advanced. All these standards have numbers starting from 802.something. Another example is the standard on which Bluetooth is based, that is 802.15.1. Hence, the name 802.11 for Wi-Fi.

One feature of Wi-Fi is that it operates in the Industrial, Scientific, and Medical (ISM) bands. There is one band around 2.4 GHz, one band around 5 GHz, another recently introduced band around 60 GHz, and so on. So, all these ISM bands were assigned for different industrial, scientific, and medical purposes, as the name suggests. So, they were originally meant for purposes other than telecom. For example, the radiation emitted by microwave ovens is in these frequency bands.

But later on, when commercial wireless communication became popular, these same bands, the ISM bands, were allowed for unlicensed use by wireless devices, such as Wi-Fi and Bluetooth. These devices, such as Wi-Fi and Bluetooth, are allowed to operate in an unlicensed manner on these bands. We can just use any certified Wi-Fi or Bluetooth equipment and operate them using these bands. There is no need to obtain any license from the spectrum regulator. So, these are unlicensed bands.

So, we discussed that the wireless medium is a shared medium, so we require some appropriate medium access control protocol to share the medium among the different nodes in the vicinity. In the case of Wi-Fi, a binary exponential back-off-based medium access control protocol is used. So, it is known as CSMA-CA or Carrier Sense Multiple Access-Collision Avoidance. So, it is a randomized MAC protocol. During our review of basic communication networks, we discussed two types of MAC protocols: channel partitioning and randomized MAC protocols.

So, the binary exponential back-off based MAC protocol used in Wi-Fi is a randomized protocol, where each node waits for a random amount of time before transmitting to avoid collisions with other nodes. The wireless channel is quite prone to errors. So, when the destination receives an error-free packet from a source, it sends a small packet called an acknowledgement packet to the source. So, this is to acknowledge the correct receipt of the packet. Consider the source that sends a packet to a destination.

If the destination receives the packet correctly, then it sends an acknowledgement to the source. And when the source receives the acknowledgement, it knows that the packet has been correctly transferred. If the source does not receive an acknowledgement, then it means that either some error happened on the channel or some collision may have occurred with other transmissions. In any case, the destination did not receive the packet correctly. So, the source retransmits the packet.

And then the destination sends an acknowledgement for the retransmitted packet, assuming that it received the retransmitted packet correctly. These retransmissions keep on proceeding until the destination receives the packet correctly. So, our focus is on the security in Wi-Fi or 802.11. Security is particularly important in 802.11. The medium is a wireless medium.

So, the medium used is the wireless medium. So, one implication is that the radio waves that carry the wireless transmissions can propagate beyond the building that contains the access point and mobile devices, such as laptops and smartphones. So, these radio waves

are transmitted using antennas, and they propagate everywhere in the vicinity. In particular, these transmissions can penetrate walls, and they can go out through windows and so on. So, these transmissions propagate beyond the building which contains the network consisting of access points and mobile devices.

So, this transmitted signal can be intercepted using packet sniffers placed near the network. Earlier, we have discussed packet sniffers, which are devices that can be used to intercept signals that are being sent on a particular communication medium. So, if there is an access point and associated mobile devices in an area, then an intruder can place a passive sniffer near the network and sniff all the packets that are being exchanged on the network. So, in particular, if the packets are not encrypted, then the intruder can get the information that is being sent. Another implication of the fact that the wireless medium is used is that an intruder can transmit packets pretending to be an access point or a legitimate mobile device.

So, one does not have to be connected to any cable for transmitting packets on the medium. Since this is a wireless medium, an intruder can just use a wireless transmitter and transmit in the band in which the network is operating. So, it's very easy to pretend to be an access point or legitimate mobile device and transmit packets on the medium. For this reason, we require authentication mechanisms. A mobile device needs to be convinced that it is communicating with the legitimate access point and vice versa.

For these reasons, security is particularly important in 802.11. So, security has evolved in the 802.11 standards. So, in the original 802.11 standard, which was adopted in the late 1990s, there were a set of security mechanisms which were known collectively as Wired Equivalent Privacy (WEP). These security mechanisms were included in the original 802.11 standard in the late 1990s. But later on, several security flaws were found in WEP.

- In the original 802.11 standard (adopted in late 1990s), a set of security mechanisms known collectively as **Wired Equivalent Privacy (WEP)** were included
  - later several security flaws were found in WEP
- In 2004, **802.11i**, a more secure standard for 802.11 security was adopted; in 2009, another security related amendment, **802.11w**, was introduced
- Next: we discuss WEP, some of the flaws in it, 802.11i, and 802.11w

We'll discuss these security flaws. Because of these flaws, in 2004, a more secure standard for 802.11 security was adopted. This was known as 802.11i. And later on, in 2009, another security-related amendment called 802.11w was introduced. So, about the numbering of

these standards; 802.11 is a series of standards, and there are alphabets that indicate different amendments and different standards within this broad family.

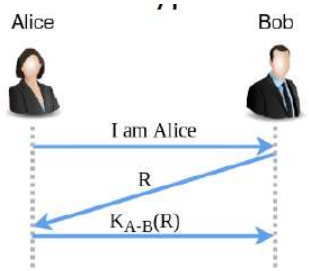
For example, there are names like 802.11a, 802.11b, 802.11g, 802.11n is a series of MIMO standards, 802.11ac, 802.11ax, and so on. So, there are alphabets that denote different standards in this family. So, 802.11i and 802.11w are security-based standards. So, next, we discuss WEP and then we discuss some of the flaws in WEP. Later on, we'll see how 802.11i remedied these flaws.

And then we'll also discuss 802.11w. So, we start with our discussion of WEP. It was designed to provide authentication and data encryption between a mobile device or host and the access point. So, how are keys distributed in WEP? WEP assumes that a symmetric shared key exists between the host and the access point.

No key management algorithm is provided by WEP. In particular, WEP does not use any public key algorithms to distribute keys. It is assumed instead that the host and the access point have somehow agreed on the symmetric key to be used. One typical way in which they can agree on the symmetric key is that a user may manually input a key provided by the system administrator, and the system administrator inputs the same key into the access point. So, this way, the access point and host agree on the symmetric key to be used.

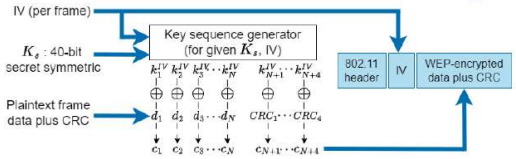
So, we first discuss WEP's authentication protocol, and then we'll discuss its encryption protocol. So, WEP's authentication protocol is the same as the authentication protocol ap4.0 that we studied earlier. This ap4.0 is shown in this figure. It's a one-way authentication protocol. A host requests authentication by an access point.

That request is this message, 'I am Alice'. So, the host is Alice in this case, and the access point is Bob. So, the host sends this message, 'I am Alice', to Bob. Then, the recipient Bob, which is the access point, responds with a 128-byte nonce value. We denote it by R in this example.



And then, the host Alice encrypts the nonce using the symmetric key that it shares with the access point and sends the encrypted nonce to the access point. So, that encrypted nonce is $K_{A\text{-}B}(R)$. So, this is sent to Bob, that is, the access point. And then, the access point decrypts and verifies the encrypted nonce sent by the host. So, Bob decrypts this value and checks whether the decrypted value equals the nonce R that he sent to Alice. So, this ap4.0 protocol, which we studied earlier, is the authentication protocol used in WEP.

- A 40-bit symmetric shared key, K_S, assumed to be known by both host and AP
- A 24 –bit *Initialization Vector* (IV) appended to K_S to get a 64-bit key that is used to encrypt a single frame
  - IV changes from frame to frame (*e.g.*, selected randomly)
- Suppose plaintext data is N bytes in length; 4-byte CRC computed for it
- The 64-bit key used to generate a stream of key values (1 byte each), k_1^{IV}, k_2^{IV}, k_3^{IV},... using RC4 stream cipher (details omitted)
- Ciphertext obtained by XORing (plaintext data+CRC) with the key value stream
- IV included in plaintext form in header of WEP-encrypted frame



We now discuss WEP's data encryption protocol. A 40-bit symmetric shared key, K_S, is assumed to be known by both the host and the access point. A 24-bit initialization vector (IV) is appended to the key K_S to get a 64-bit key that is used to encrypt a single frame, and this IV changes from frame to frame. For example, it may be selected randomly. We combine the 40-bit static key, K_S, with a 24-bit IV, which is different for different frames, and that way, by combining these, we get a 64-bit key, and that is used to encrypt a single frame.

And this 64-bit key changes from frame to frame, since the IV part of the key is different for different frames. Now, we will discuss encryption. Suppose the plaintext data of the packet is N bytes in length. Then, a 4-byte CRC or cyclic redundancy check is computed for it. The CRC is a kind of checksum.

So, if any bits in the plaintext data get modified, then the CRC indicates that some bits have been modified. This shows the encryption procedure. This is K_S, which is the 40-bit symmetric secret, and we add an IV of 24 bits to it. So, the result is a 64-bit key, and this 64-bit key is used in a block called the key sequence generator. It generates a stream of key values, 1 byte each, which are denoted here by $k_1^{IV}, k_2^{IV}, k_3^{IV}$, and so on up to k_{N+4}^{IV}.

And this stream of key values is generated using the RC4 stream cipher. We omit the details of how exactly this key stream is generated. RC4 is a stream cipher, which stands for Rivest Cipher 4. So, for our purposes, the input to this key sequence generator is the 64-bit key, which is a combination of the 40-bit key K_S and the IV of 24 bits. And this key sequence generator is like a pseudo-random function.

So, it takes this 64-bit key as input and generates these bytes, which look like random bytes. So, this is like a pseudo-random generator. It generates this key stream, which is a sequence of random-looking bytes. So, for someone who doesn't know the secret key that is used, that is K_S, this key stream looks like a random sequence of bytes. Now, how is the ciphertext obtained?

So, we have the N bytes of the plaintext data in the message, that is d_1 to d_N, and the 4 bytes of CRC, that is CRC_1 to CRC_4. So, all these are XORed with the key value stream consisting of N+4 bytes. So, after this XOR, we get the ciphertext bytes, C_1, C_2, up to C_{N+4}. So, these are the ciphertext blocks. These are the ciphertext bytes corresponding to the plaintext data in the message and the CRC.

There is a header in an 802.11 packet that is shown over here. It has different fields, including, for example, the source MAC address, destination MAC address, and so on. And then the IV is put in plaintext form in the 802.11 packet. So, that is shown over here. So, the IV is put in plaintext form in the packet, and this is where the WEP encrypted data plus CRC is put.

So, these are the ciphertext bytes obtained using the procedure that we discussed by XORing the key stream with the plaintext frame data plus the CRC. So, this goes into this place in the packet. This is how a packet is created in 802.11 when WEP is used. And the IV is included in plaintext form in the header of the WEP-encrypted frame. So, as shown here, this IV is a part of the header.

So, this is the encryption procedure that is used in WEP. Now, we discuss the reasons for using the IV. Why do we add a 24-bit IV, which is different for different packets, before generating the key stream? Recall that K_S is a 40-bit secret symmetric key, which is shared between the host and the AP, and we add an IV of 24 bits to it. So, this way, we get a 64-bit key that is used to encrypt a single frame, and this IV changes from frame to frame.

For example, it is selected randomly. So, the reasons for using the IV are as follows. One reason is to ensure that two identical plaintext messages do not produce the same

ciphertext. So, assume that the IV was not used, and the key stream was just a function of K_S. In that case, this key stream would be the same for different frames.

So, if the plaintext in two different frames were the same, then the ciphertext blocks would be the same in the two frames in the absence of an IV. So, that's one reason for using the IV. By looking at these identical ciphertext blocks, the intruder would come to know that the corresponding plaintext messages are the same. So, we don't want to leak any information about the plaintext messages. So, we want identical plaintext messages to map to different ciphertext blocks.

So, that's one reason for using the IV. Since an IV is used, even if the plaintext messages are the same in two different frames, the IV will be different with high probability. Hence, the ciphertext blocks will be different. Another reason for using the IV is that for every frame, the RC4 algorithm, which is used to generate the keystream, is initialized with the key value prior to the start of the keystream generation. Now, if the IV were not used, then the key value would be the same for every frame, and the RC4 algorithm would be initialized to the same state in every instance.

Hence, the keystream produced would be the same for every frame. So, in this case, the (plaintext + the CRC) would be XORed with exactly the same keystream in every frame. So, this would be a serious weakness because if an attacker found out the keystream somehow, for example by guessing the plaintext in a frame, then the attacker would be able to decipher every frame by XORing the frame with the keystream. So, assume that the attacker guesses some of the plaintext bytes; in that case, the intruder can XOR these plaintext bytes with the corresponding ciphertext bytes which are sent on the channel. So, if we XOR these plaintext bytes with the ciphertext blocks, then we get the corresponding keystream bytes.

So, because keystream XOR plaintext is equal to ciphertext, hence ciphertext XOR with plaintext is equal to the keystream. So, if the intruder guesses some of the bytes in the plaintext, then they would be able to deduce the keystream from it. And after deducing some of the keystream bytes, the intruder can take future ciphertext frames and XOR them with the guessed keystream bytes to recover the plaintext. So, the intruder would be able to decipher every frame by XORing the frame with the keystream. So, in the extreme case, where the attacker finds out the entire keystream by guessing the entire plaintext in a frame, the intruder would be able to decipher every frame completely by XORing the frame with the keystream.

Even if one frame's plaintext becomes known, in that case, the entire security is compromised because of this algorithm. For this reason, we use a different IV for different frames. These are the reasons for using a different IV in every frame. This concludes the discussion of WEP's encryption protocol. So, to summarize, in this lecture, we introduced the topic of Wi-Fi security.

We discussed that there are different security standards for Wi-Fi. The original standard introduced in the 1990s had WEP as the security mechanism. Then several flaws were found in WEP. Later on, other security standards, namely 802.11i and 802.11w, were introduced. We have discussed the authentication and encryption protocols used in WEP.

In the next lecture, we'll discuss the flaws in WEP that have been discovered. Thank you.