InteropNet Labs

5 in a series

What's Wrong With WEP?

WEP is the privacy protocol specified in IEEE 802.11 to provide wireless LAN users protection against casual eavesdropping. WEP stands for "Wired Equivalent Privacy" referring to the intent to provide a privacy service to wireless LAN users similar to that provided by the physical security inherent in a wired LAN.

When WEP is active in a wireless LAN, each 802.11 packet is encrypted separately with an RC4 cipher stream generated by a 64 bit RC4 key. This key is composed of a 24 bit initialization vector (IV) and a 40 bit WEP key. The encrypted packet is generated with a bitwise exclusive OR (XOR) of the original packet and the RC4 stream. The IV is chosen by the sender and should be changed so that every packet won't be encrypted with the same cipher stream. The IV is sent in the clear with each packet. An additional 4 byte Integrity Check Value (ICV) is computed on the original packet using the CRC-32 checksum algorithm and appended to the end. The ICV (be careful not to confuse this with the IV) is also encrypted with the RC4 cipher stream. WEP has been widely criticized for a number of weaknesses.

Weakness: Key Management and Key Size

Key management is not specified in the WEP standard, and therefore is one of its weaknesses, because without interoperable key management, keys will tend to be long-lived and of poor quality. Most wireless networks that use WEP have one single WEP key shared between every node on the network. Access Points (APs) and client stations must be programmed with the same WEP key. Since synchronizing the change of keys is tedious and difficult, keys are seldom changed.

In addition, the size of the key---40 bits---has been cited as a weakness of WEP. When the standard was written in 1997, 40 bit keys were considered reasonable for some applications. Since the goal was to protect against "casual eavesdropping" it seemed sufficient at the time. The US did not tightly control exports of 40-bit encryption, and the IEEE wanted to ensure exportability of wireless devices.

The 802.11 standard does not specify any WEP key sizes other than 40 bits. Most vendors have implemented a de facto standard, simply extending the key size to 104 bits, with excellent interoperability. You will often see this called “128-bit” WEP key (because it sounds better than a 104-bit key), but that is not a fair comparison. This is why you enter 13 characters (or 26 hexadecimal digits) instead of 16 characters when you set up a long WEP key. In either case (40-bits or 104-bits), the RC4 encryption key includes a 24-bit IV (as described above). Obviously, 104-bit keys are more resistant to brute-force attacks than 40-bit keys. But brute-force attacks on 104-bit keys---which would take billions of years---are not considered the primary weakness of WEP.

Weakness: The Initialization Vector (IV) is Too Small

WEP’s IV size of 24 bits provides for 16,777,216 different RC4 cipher streams for a given WEP key, for any key size. Remember that the RC4 cipher stream is XOR-ed with the original packet to give the encrypted packet which is transmitted, and the IV is sent in the clear with each packet. The problem is IV reuse. If the RC4 cipher stream for a given IV is found, an attacker can decrypt subsequent packets that were encrypted with the same IV, or, can forge packets. This means that you don’t need to know the WEP key to decrypt packets if you know what the key stream was used to encrypt that packet. They sound like similar problems, but it’s actually much easier to discover the key stream than it is to discover the WEP key.

Since there are only 16 million IV values, how the IV is chosen makes a big difference in the attacks based on IV. Unfortunately, WEP doesn’t specify how the IV is chosen or how often the IV is changed. Some implementations start the IV at zero and increase it incrementally for each packet, rolling over back to zero after 16 million packets have been sent. Some implementations choose IVs randomly. That sounds like a good idea, but it really isn't. With a randomly chosen IV, there is a 50% chance of reuse after less than 5000 packets.

Additionally, there are many methods for discovering the cipher stream for a particular IV. For example, given two encrypted packets with the same IV, the XOR of the original packets can be found by XORing the encrypted packets. If the victim is on the Internet, the attacker can simply ping the victim or send an email message. If the attacker is able to send the victim packets and observe and analyze those encrypted packets, he can deduce the cipher stream.

Weakness: The Integrity Check Value (ICV) algorithm is not appropriate

The WEP ICV is based on CRC-32, an algorithm for detecting noise and common errors in transmission. CRC-32 is an excellent checksum for detecting errors, but an awful choice for a cryptographic hash. Better-designed encryption systems use algorithms such as MD5 or SHA-1 for their ICVs.

The CRC-32 ICV is a linear function of the message meaning that an attacker can modify an encrypted message and easily fix the ICV so the message appears authentic. Being able to modify encrypted packets provides for a nearly limitless number of very simple attacks. For example, an attacker can easily make the victim's wireless AP decrypt packets for him. Simply capture an encrypted packet stream, modify the destination address of each packet to be the attacker's wired IP address, fix up the CRC-32, and retransmit the packets over the air to the AP. The AP will happily decrypt the packets and forward them to the attacker. (The attack is slightly more complex than that, but to keep this paper short, we've skipped some of the details.)

The biggest problem with IV and ICV-based attacks is they are independent of key size, meaning that even huge keys all look the same. The attack takes the same amount of effort.

Weakness: WEP's use of RC4 is weak

RC4 in its implementation in WEP has been found to have weak keys. Having a weak key means that there is more correlation between the key and the output than there should be for good security. Determining which packets were encrypted with weak keys is easy because the first three bytes of the key are taken from the IV that is sent unencrypted in each packet. This weakness can be exploited by a passive attack. All the attacker needs to do is be within a hundred feet or so of the AP.

Out of the 16 million IV values available, about 9000 are interesting to the most popular attack tool, meaning they indicate the presence of weak keys. The attacker captures "interesting packets", filtering for IVs that suggest weak keys. After that attacker gathers enough interesting packets, he analyzes them and only has to try a small number of keys to gain access to the network. Because all of the original IP packets start with a known value, it's easy to know when you have the right key. To determine a 104 bit WEP key, you have to capture between 2000 and 4000 interesting packets. On a fairly busy network that generates one million packets per day, a few hundred interesting packets might be captured. That would mean that a week or two of capturing would be required to determine the key.

The best defense against this type of attack is not to use those weak IV values. Most vendors are now implementing new algorithms that simply do not choose weak IVs. However, if just one station on the network uses weak keys, the attack can succeed.

Weakness: Authentication Messages can be easily forged

802.11 defines two forms of authentication: Open System (no authentication) and Shared Key authentication. These are used to authenticate the client to the access point. The idea was that authentication would be better than no authentication because the user has to prove knowledge of the shared WEP key, in effect, authenticating himself. In fact, the exact opposite is true: if you turn on authentication, you actually reduce the total security of your network and make it easier to guess your WEP key.

Shared Key authentication involves demonstrating the knowledge of the shared WEP key by encrypting a challenge. The problem is that a monitoring attacker can observe both the challenge and the encrypted response. From those, he can determine the RC4 stream used to encrypt the response, and use that stream to encrypt any challenge he receives in the future. So by monitoring a successful authentication, the attacker can later forge an authentication. An advantage of Shared Key authentication is that it reduces the ability of an attacker to create a denial-of-service attack by sending garbage packets (encrypted with the wrong WEP key) into the network. Shared Key authentication also allows a wireless client to quickly determine if they know the correct WEP key, which is a nice "user friendly" configuration---and allows a malicious client to try a dictionary attack on the wireless network.

Open system gives you better network security. Most network managers should turn off shared key authentication and depend on other authentication protocols, such as 802.1X, to handle the task of properly authenticating wireless users.