Wireless Security

Jason Bonde
University of Minnesota, Morris
bond0107@morris.umn.edu

ABSTRACT

Wireless internet has become a popular way of accessing the Internet. While wireless internet does have the advantage of being inexpensive and highly convient, it is potentially risky. This paper discusses three algorithms for wireless security: Wired Equivalent Protection (WEP), Wi-Fi Protected Access (WPA), and WPA's successor, WPA2. The main focus is WPA, WPA's Temporal Key Integrity Protocol (TKIP), and the Beck-Tews attack on TKIP. A related earlier attack on WEP known as chopchop attack is covered to provide background to better understand the Beck-Tews attack. A brief explanation of the latest security standard, WPA2, and its defense against the Beck-Tews attack is also offered.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless communication*; C.2.0 [**Computer-Communication Networks**]: General—*Security and protection*

General Terms

Security

Keywords

Wireless security, WEP, WPA, WPA2, RC4, TKIP, Beck-Tews attack, Chopchop attack

1. INTRODUCTION

As the popularity of accessing the internet via wireless increases, the amount of protection must also increase. On a wireless network, communication is done by transmitting and receiving messages through the air by radio waves. It is easy for any outsider to intercept these waves. If the messages being sent are not encrypted securely, there is great risk. On a network with inadequate protection, an intruder could recover passwords, social security numbers, credit card numbers, or other private information.

UMM CSci Senior Seminar Conference Morris, MN.

There are three security algorithms for encryption that this paper will cover. Wired Equivalent Protection (WEP) will be covered in Section 3, Wi-Fi Protected Access (WPA) in Section 4, and WPA's successor, WPA2, in Section 6. The strengths and weaknesses of each algorithm will be covered in their respective sections. The chopchop attack on WEP networks is described in Section 5.1. The highly effective Beck-Tews attack that can be performed in just twelve minutes is covered in 5.2. Section 5.2 details the attack as well as some countermeasures against it. Section 2 covers the relevant background information needed.

2. BACKGROUND

In this section we will define the key concepts used later in the paper.

2.1 Shared-key Encryption

Shared-key, or symmetric-key, algorithms are a class of algorithms used in cryptography that use identical, or trivially related cryptographic keys for both encryption and decryption [9]. By trivial it is meant that a simple transformation is required to go between the two keys. The two keys represent a shared secret between two or more parties. WEP, WPA, and WPA2 all use shared-key encryption. For example, if the shared secret was "each letter is replaced by the letter after it in the alphabet", the encryption of "password" would be "qbttxpse". To decrypt the message, the recipient would replace each letter with the letter that occurred before.

2.2 Keystreams

A keystream is a psuedorandom stream of bits, bytes, numbers, or letters. Keystreams are used in many cryptographic protocols. A keystream can be combined with a plaintext message by adding, subtracting, or by performing a bitwise exclusive OR (XOR) using modular arithmetic to produce an encrypted message, or ciphertext. Keystreams are used in most stream ciphers, including Rivest Cipher 4, described in more detail next. An important property of a keystream is that if the stream is genuinely random, the resulting encrypted message will also be genuinely random. The encrypted message will have no patterns which can be used to decipher the original plaintext.

An example using XOR is provided in Figure 1. The plaintext is the top sequence of bits and the keystream is the middle; the result of the XOR is at the bottom. Using XOR, if the two binary values are both 1 or both 0, the result is 0. If and only if one value is 1, then the result is 1. To retrieve

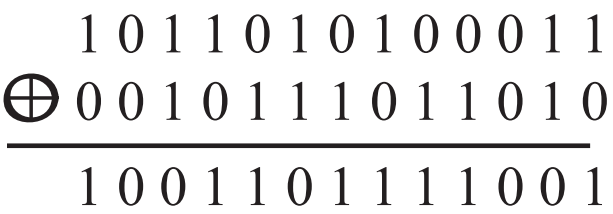

Figure 1: An example of bitwise exclusive OR being performed on a plaintext (top) and a keystream (middle) to produce an encrypted message (bottom).

the plaintext, the keysteam is XORed with the encrypted message.

2.3 Rivest Cipher 4

A commonly used component of some encryption protocols is Rivest Cipher 4, more commonly referred to as RC4. RC4 generates a keystream, which can then be used for encryption. Generating the keystream requires two parts: a permutation of all 256 possible bytes (S), and two 8-bit index-pointers (i and j).

The permutation is initialized with a variable length *key*, which is usually between 40 and 256 bits, using the key-scheduling algorithm (KSA) shown in Alg. 1. The KSA first sequentially assigns the values 0 through 255 to the array S. It then scrambles the values by incrementing the variable b by a value based on the key and the key length mod 256, and swapping the ath and bth position in the S array. At the end of this process S contains the values 0 through 255 randomized in a way that can be easily replicated with the key, but without the key is very difficult to guess or recreate.

Once the KSA is finished, the stream of bits is generated using the pseudo-random generation algorithm (PRGA) that is shown in Alg. 2. The PRGA cycles i and j through the values 0 through 255, i increments by 1 on each pass, while j increments by a value from the now-shuffled array S. After each increment the values at the ith and jth position within the array S are swapped. The value that is returned, K, is a byte, which contains 8 bits for use in a keystream.

While RC4 is quick and simple, it does have vulnerabilities. The KSA that RC4 uses has two significant weaknesses. The first weakness is the existence of large class of weak keys, where a small part of the key determines a large number of bits of the KSA output. The PRGA translates the patterns of the initial permutation (the KSA output) into patterns in the prefix of the output stream. Therefore, the initial outputs of the weak keys are disproportionally affected by a small number of key bits. The defining property of a weak key is their length. A weak key has a length which is divisible by some non-trivial power of two. The exact mathematical details are not the purpose of this paper, and can be found at the beginning of [5]. The second weakness of KSA happens when part of the key is exposed to an attacker. When the same secret part of the key is used with multiple different exposed values, an attacker can analyze the initial word of the keystreams and rederive the secret part. Cryptographic methods, such as WEP, that use RC4 without taking precautions against these weaknesses can be very insecure.

Algorithm 1 Key-Scheduling Algorithm

```
for a = 0 to 255 do
    S[a] = a
end for
b = 0
for a = 0 to 255 do
    b = (b + S[a] + key[a mod keyLength]) mod 256
    swap values of S[a] and S[b]
end for
```

2.4 Hash Function

A hashing algorithm takes an input and transforms it into a new value, called the hash value. The importance of a hash value is that it is very difficult, often near impossible, to derive the original input value even if the hashing function is known. An example of a hash function would be $Input*3+1$. Inputting 3 into the function would return a hash value of 10. This example is poor because it is easy to inverse. A good hash function would be a one-way function that is very difficult to inverse.

2.5 Packets

A packet is the unit of data that is routed between an origin and a destination on a network. When any file is sent from one place to another on the Internet, the file is divided into efficiently sized packets for routing. Each packet is individually numbered and includes the Internet address of the destination. Once all of the packets have been received, the destination reassembles them into the original file. There are attacks, such as chopchop and Beck-Tews, that take advantage of the structure and contents of packets.

2.6 IP Address

An Internet Protocol (IP) address is a unique address which is used to locate and verify a device on a network. The most widely used IP is Internet Protocol version 4 (IPv4). IPv4 uses 32-bit (four-byte) addresses. The addresses are usually represented in dot-decimal notation where each byte is separated by a dot. An example of an IPv4 address is 146.57.92.50. An IP address has two parts: a network part and a machine-specific part. The first three bytes are the network part, and the last is the machine-specific part. In

Algorithm 2 Pseudo-Random Generation Algorithm

```
i = 0
j = 0
while GeneratingOutput do
    i = (i + 1)  mod 256
    j = (j + S[i])  mod 256
    swap values of S[i] and S[j]
    K = S[(S[i] + S[j])  mod 256]
    return K
end while
```

the example, 146.57.92 would be the network, and 50 would be a specific computer.

2.7 Quality of Service

One of the conditions of the Beck-Tews attack on WPA, covered in Section 5.2 is that the network being attacked supports the IEEE 802.11e Quality of Service (QoS) features. The quality of service feature allows eight or sixteen different channels for data flow. The attack on WPA requires the ability to change between different channels.

2.8 Cyclic Redundancy Check

WEP and WPA use a cyclic redundancy check (CRC) to ensure message integrity. A CRC is a class of "checksum" algorithms that treat any message as a large binary number and then divide that number by a fixed constant [3]; the remainder is the "checksum". An example of a checksum for input "The red fox jumps over the blue dog" is 2367213558. If the input were changed to "The red fox jumps oevr the blue dog", the result would be 1321115126. When a message is sent, the CRC is computed, and appended to the message. When a message is received, it is easy to compute the checksum of the message, and then check to see if the numbers match. If they do not, the receiver knows that the message has been altered.

CRCs requires no authentication, allowing an attacker to edit a message and recalculate the CRC without the substitution being detected; this remains true even when the CRC is encrypted. The chopchop attack specifically exploits this vulnerability and recalculates the CRC in its steps to decrypt a plaintext. CRCs are designed to protect against common types of errors on communication channels. CRCs are not suited for protecting against the intentional altering of data, which makes them a poor choice for use as an integrity check in a security protocol.

3. WIRED EQUIVALENT PRIVACY

Wired Equivalent Privacy (WEP) is a security algorithm that implements the IEEE 802.11 security standard. It was introduced as a part of the original 802.11 standard in 1999. The purpose of WEP was to provide data confidentiality which was comparable to that of a wired network. It uses the RC4 stream cipher for confidentiality and the CRC-32 mechanism for integrity.

WEP constructs the cipher text by performing an XOR operation on the plaintext and an RC4 keystream. The seed for generating the keystream is a 40-, 104-, or 232-bit key concatenated to a 24-bit initialization vector (IV). A larger key size provides more security because more packets are required to crack a longer key. However, a longer key does not prevent IV collision, which occurs when an IV is repeated. The purpose of an IV is to prevent repetition. A 24-bit IV is simply too short to ensure that a collision will not happen. After 5000 packets, there is a 50 percent probability of a repeated IV.

With the ratification of the full IEEE 802.11i standard in 2004, the IEEE declared WEP-40 and WEP-104 as deprecated. WEP-40 is WEP with a 40 bit key size; WEP-104 has a 104 bit key size. There have been many published attacks for use against WEP [7]: the Fluhrer, Mantin and Shamir (FMS) attack, the KoreK attack, the Pyshkin, Tews and Weinmann (PTW) attack, and the chopchop attack. Section 5.1 will describe the chopchop attack. A more sophisticated version of the chopchop attack is used by [7] to break WPA.

4. WI-FI PROTECTED ACCESS

The Wi-Fi Protected Access, or WPA, protocol was introduced by the Wi-Fi (Wireless Fidelity) Alliance in 2003 to comply with the pending IEEE 802.11 standard. WPA was meant to solve the cryptographic problems of WEP without requiring new hardware.

WPA has three primary improvements over WEP [3]:

- The Temporal Key Integrity Protocol (TKIP) provides improved data encryption. More details are given in the next section.
- TKIP uses a new algorithm called "Michael" to compute its new Message Integrity Code (MIC). MIC is computed in order to detect data content errors, which may be due to errors or purposeful alterations [6]. In theory, there is only a one in one million chance of guessing the correct MIC.
- The extensible authentication protocol (EAP) provides user authentication which WEP lacks. For more information on EAP, refer to [4] or [2].

The introduction of WPA also brought along other, less vital, features, including:

- Key management is an issue in WEP; WPA has built-in secure key management.
- The IV length has been increased to 48 bits from 24 bits to reduce the likelyhood of reusing keys, which is a major security flaw in WEP. To protect against the replaying of data, IVs are used as sequence counters for the TKIP Sequence Counter (TSC). Each time a packet is sent, the IV is increased by one.
- WPA avoids using known weak IV values.
- Each packet is encrypted with a different secret key.

5. TKIP

Because TKIP was designed to be compatible with existing hardware using WEP there were many constraints in its design. These constraints are divided into three major parts [1]:

- The fixes must be completely deployed through software upgrades only. Upgrading the existing hardware would probably cost more than purchasing new hardware.
- The new algorithm must run on existing low-end processors that are already deployed in wireless hardware. Vendors use the cheapest processors available, leaving few CPU cycles for more operations. In older access points, traffic management can consume up to 90 percent of available CPU. An access point is a device, usually a router, that allows wireless devices to connect to a wired network.

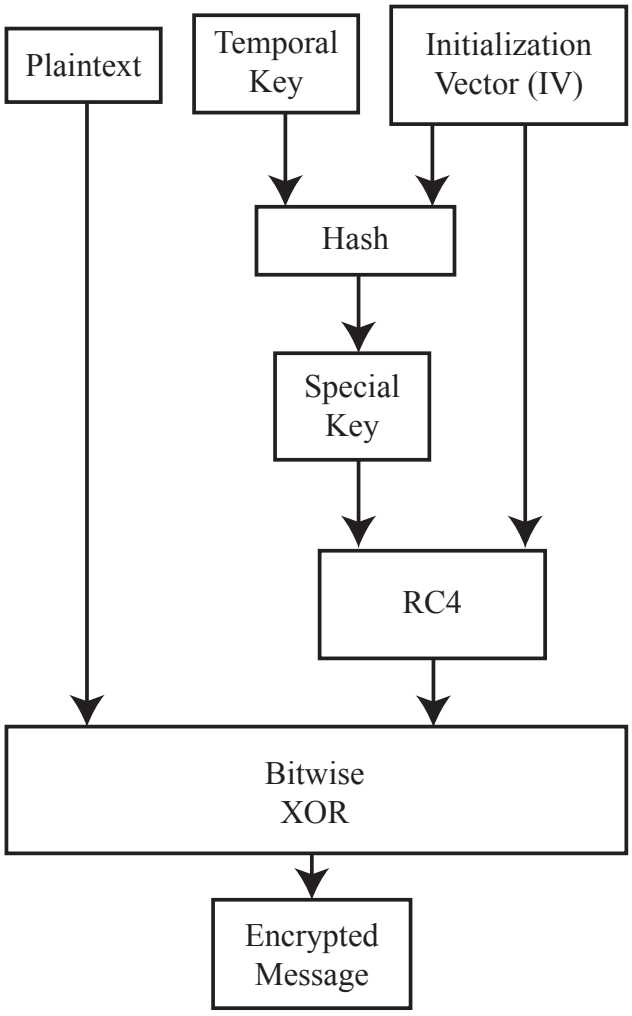

Figure 2: Temporal Key Integrity Protocol

- The new algorithm also has to use the existing hardwired encryption function that is in deployed wireless hardware. Because of the limited CPU cycles, manufacturers included custom WEP hardware for the encryption and decryption operations. The encryption function expects the shared key, an IV, and the packet to encrypt or decrypt. The operation is done on a per-packet basis.

With all of those constraints taken into consideration, TKIP was created as an enhancement to WEP. In order to be run on WEP hardware, TKIP uses the RC4 stream cipher for encryption and decryption. TKIP scrambles the keys using a hashing algorithm. An integrity-checking feature was added to ensure that the keys have not been tampered with. All parties must share the same secret key, called the "Temporal Key", and the key must be 128 bits. To fix the vulnerabilities of WEP without hardware change, TKIP has:

- A key mixing function which operates on a per-packet basis.
- A sequence counter which is used to prevent replay attacks. Whenever a packet is correctly received, the counter is updated. If a packet has a lower value than the current counter (the packet is received out of order), then it is discarded.
- A message integrity code named Michael which prevents packet modifications and injections.

Figure 2 shows the steps TKIP takes to encrypt a message. The IV and the Temporal Key are both put into a hash function, which returns a new, different key. The new key is then used along with the IV as input for the RC4 stream cipher. The hash function is the additional step WPA introduced to make RC4 more secure. The resulting RC4 keystream is then XORed with the plaintext to produce the encrypted message.

5.1 Chopchop Attack

The Beck-Tews method for cracking WPA uses a modified chopchop attack [7]. The chopchop attack allows an attacker to decrypt the last m bytes of a plaintext of an encrypted packet by sending, on average, $m * 128$ packets to the network. The attack exploits the insecurity of the four byte CRC-32 checksum, which is appended on the packet's data. The checksum is named the integrity check value (ICV).

The majority of access points can be used to distinguish encrypted packets with a correct and an incorrect checksum. If the access point receives a packet with a correct checksum from an unauthenticated client, the access point will generate an error message. If the packet had an incorrect checksum, it is silently discarded.

An attacker selects a captured packet for decryption. The attacker then guesses the last byte of a packet, R, and corrects the checksum. The packet is then sent to the access point to determine if the guess for R was correct. On a correct guess, the attacker knows the last byte of plaintext and can continue with the second to last byte. If the guess was incorrect, the attacker makes different guess for R. It will take at most 256 guesses, and on average 128 guesses, to guess the correct value of R. The exact mathematics can be found in [7].

5.2 Beck-Tews Attack

The first published work on cracking WPA encryption was by Martin Beck and Erik Tews, in 2008. The Beck-Tews attack requires several reasonable conditions. First, the network that is being attacked is using TKIP for client to access point communication. The IPv4 protocol must also be used. The attacker must know most of the bytes of the IP addresses (for example, 190.162.0.X). There must also be a long re-keying interval for TKIP, such as 3600 seconds. The re-keying interval is how often TKIP establishes a new Temporal Key. The network must support the IEEE 802.11e QoS feature. This feature allows 8 different channels for different data flows.

In order to attack a network, the attacker first captures traffic until the encrypted ARP (Address Resolution Protocol) request or response is found. The exact details of ARP are unimportant for this paper. What is important is the structure of ARP packets. ARP request or response packets are easily detected because of their characteristic length and the destination always being the broadcast address. WEP and TKIP do not protect the source and destination Internet

addresses, so they are always sent to the broadcast address of the network. The majority of the plaintext of this packet is known to the attacker. The attacker does not know the last byte of the source and destination IP addresses, the 8 byte Michael MIC, and the 4 byte ICV checksum. Michael and the ICV form the last 12 bytes of the packet.

In order to decrypt the unknown plaintext, an attacker can launch a modified chopchop attack. TKIP has two primary countermeasures against chopchop-like attacks. The first countermeasure is based on the ICV and the MIC. If a packet with an incorrect ICV is received by the client, TKIP assumes a transmission error and silently discards the resulting packet. If the ICV is correct, but the MIC verification fails, an attack is assumed. The client then notifies the access point by sending a MIC failure report frame. If there are two or more MIC failure reports in 60 seconds, communication is shut down[1]. All keys are then renegotiated after a 60 second penalty period.

The second countermeasure TKIP employs is its sequence counter (TSC). When a packet is correctly received, the TSC for the channel the packet was received on is updated. If a packet with a lower value than the current counter is received then it is discarded.

To execute a chopchop attack, the attacker must use a different QoS channel from the one the packet was originally received on. There is usually a channel with little to no traffic where the TSC is lower. If the guess for the last byte during the chopchop attack was incorrect, it is silently dropped. If the guess was correct, a MIC failure report frame is sent, but the TSC is not increased. If the attacker waits at least 60 seconds after trigging a MIC failure report frame, the TKIP countermeasures can be circumvented. To decrypt the last 12 bytes, the MIC and the ICV, will then take little more than 12 minutes since each byte takes a little more than 60 seconds. Once this is completed, the attacker can decrypt the exact sender and receiver IP address by guessing the values and checking them against the decrypted ICV.

Once the attacker knows the MIC and the plaintext of the packet, the Michael algorithm can be reversed to recover the MIC key, which protects packets being sent from the access point to the client. Michael was not designed as a one-way function, and reversing it is just as efficient as calculating it forward. The attacker would now have recovered the MIC key, and know the keystream for access point to client communication. The attacker can now generate a frame that will pass the MIC check. The attacker can send a custom packet on every QoS channel where the TSC is lower than the value used for the captured packet. For most networks, all traffic is simply transmitted to channel 0. This means that the TSC from the captured packet, from channel 0, will be much higher than the other channels. Because of this, the attacker can send a custom packet across each of the channels 1-7 without being detected. In some cases it is possible to also use channels 8-15, which allow for 8 more custom packets to be sent. One example of what an attacker could do with the custom packets is reroute traffic using fake ARP responses.

Subsequent keystreams can be derived more quickly, in four to five minutes, once the attacker knows the MIC key. Only the ICV needs to be derived from the chopchop method. The attacker can then guess the IP, checking the MIC result with the ICV locally.

Some countermeasures are suggested by [7]. A shorter rekeying time, such as 120 seconds, would only allow the attacker to decrypt parts of the ICV at the end of the packet. If the rekeying time is short enough, by the time an attacker recovers the key it will have changed. Another solution is to disable the sending of MIC failure report frames. If the MIC failure report frames are never sent, the attacker would never know when they made a correct guess, preventing the attack entirely.

[1]It is stated in [7] that communication is shut down when there are more than two MIC failure reports. There is evidence in [7], as well as elsewhere, that communication shuts down after only two MIC failure reports.

6. WPA2

WPA2 was released in 2004 as a replacement for WPA [8]. WPA2 was not designed to be hardware compatible with WEP, as WPA was. It implements the mandatory elements of IEEE 802.11i-2004 standard. Similar to WPA, WPA2 also uses EAP for authentication.

The most important feature of WPA2 is the introduction of CCMP (Counter Mode with Cipher Block Chaining Message Authentication Code Protocol) which uses the Advanced Encryption Standard (AES) block cipher. CCMP was created to replace TKIP and WEP. CCMP currently provides the highest level of integrity, confidentiality, and replay protection available in the 802.11 standard. There are no known feasible attacks against the CCMP algorithm currently, except for brute force attacks attempting to discover weak passwords.

7. CONCLUSION

Wireless users should be aware of their current security algorithm. It has been shown that WEP is not secure. WEP has a short IV which allows for collisions and uses the CRC-32 mechanism for integrity. WPA is insecure because of the inherent flaws in TKIP. It can be made more secure by having a quicker rekeying interval. Users on a WEP or WPA network should take care to not transmit important data, such as credit card information or social security numbers, over the wireless network. If vital data needs to be transmitted over a wireless network, a protocol using CCMP, such as WPA2, should be used instead of WEP or WPA.

Acknowledgements

I would like to thank my professors Nic McPhee and Elena Machkasova for advising me throughout this paper.

8. REFERENCES

[1] K. Benton. The evolution of 802.11 wireless security. `http://itffroc.org/pubs/benton\_wireless.pdf`, April 2010.

[2] M. Bhakti, A. Abdullah, and L. Jung. EAP-based authentication with EAP method selection mechanism: Simulation design. In *Research and Development, 2007. SCOReD 2007. 5th Student Conference on*, pages 1 –4, dec. 2007.

[3] H. I. Bulbul, I. Batmaz, and M. Ozel. Wireless network security: comparison of WEP (wired equivalent privacy) mechanism, WPA (wi-fi protected access) and RSN (robust security network) security protocols. In *Proceedings of the 1st international conference on*

Forensic applications and techniques in telecommunications, information, and multimedia and workshop, e-Forensics '08, pages 9:1–9:6, ICST, Brussels, Belgium, Belgium, 2008. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[4] A. Chiornita, L. Gheorghe, and D. Rosner. A practical analysis of EAP authentication methods. In *Roedunet International Conference (RoEduNet), 2010 9th*, pages 31 –35, june 2010.

[5] S. Fluhrer, I. Mantin, and A. Shamir. Weaknesses in the key scheduling algorithm of RC4. `http://www.crypto.com/papers/others/rc4\_ksaproc.pdf`.

[6] A. Lashkari, M. Danesh, and B. Samadi. A survey on wireless security protocols (WEP, WPA and WPA2/802.11i). In *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*, pages 48 –52, 2009.

[7] E. Tews and M. Beck. Practical attacks against WEP and WPA. In *Proceedings of the second ACM conference on Wireless network security*, WiSec '09, pages 79–86, New York, NY, USA, 2009. ACM.

[8] Wikipedia. IEEE 802.11i-2004 - Wikipedia, the free encyclopedia. `http://en.wikipedia.org/wiki/IEEE\_802.11i-2004`, 2004. [Online; accessed 24-April-2011].

[9] Wikipedia. Symmetric-key algorithm - Wikipedia, the free encyclopedia. `http://en.wikipedia.org/wiki/Symmetric-key\_algorithm`, 2004. [Online; accessed 24-April-2011].